Intro to Making · How to create smart things using the Geeny IoT Platform

20.09.2017

1 **Set up.** Get the software we need for today

2 **Blink.** The Hello World of hardware

3 **Sensors.** Connect the humidity and temperature sensor

4 **MQTT.** Send and receive data

5 **Geeny.** How can we create IoT products which matter?

Setup

1. Get the software we need

1. Install Arduino software from www.arduino.cc
2. Open the Arduino software and go to files and click on the preference in the Arduino IDE
3. Copy the below code in the Additional boards Manager:

   http://arduino.esp8266.com/stable/package_esp8266com_index.json
4. In the Arduino Software go to Tools>Boards>Board Manager
5. Navigate to *esp8266 by esp8266 community* and install the software for Arduino
6. Select now from Tools>Board the NodeMCU 1.0 (ESP12E module)
7. Install USB driver from software collection folder *USB2UART-driver*

   (or https://www.silabs.com/products/development-tools/software/usb-to-uart-bridge-vcp-drivers*)*

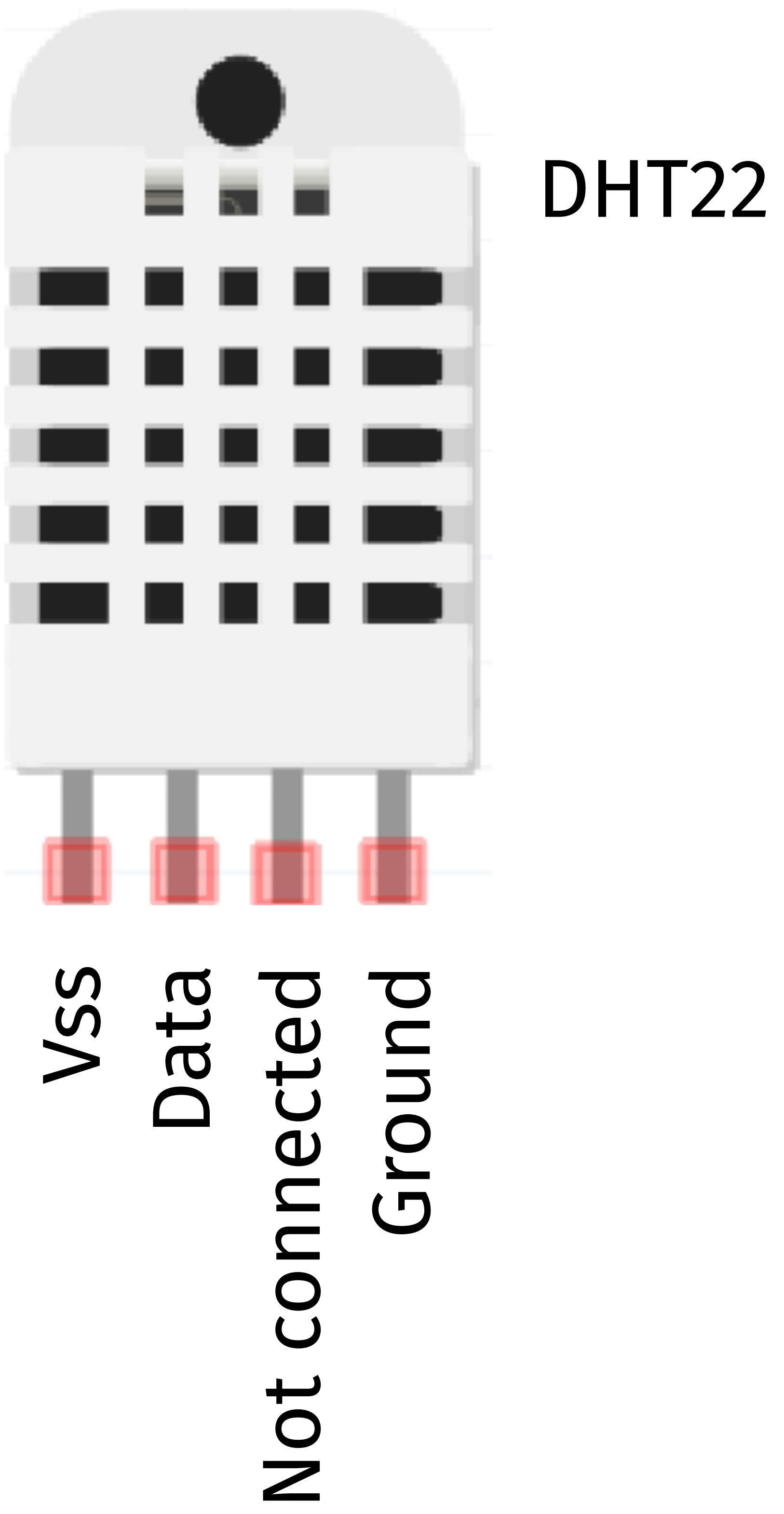
Hello World. aka blink

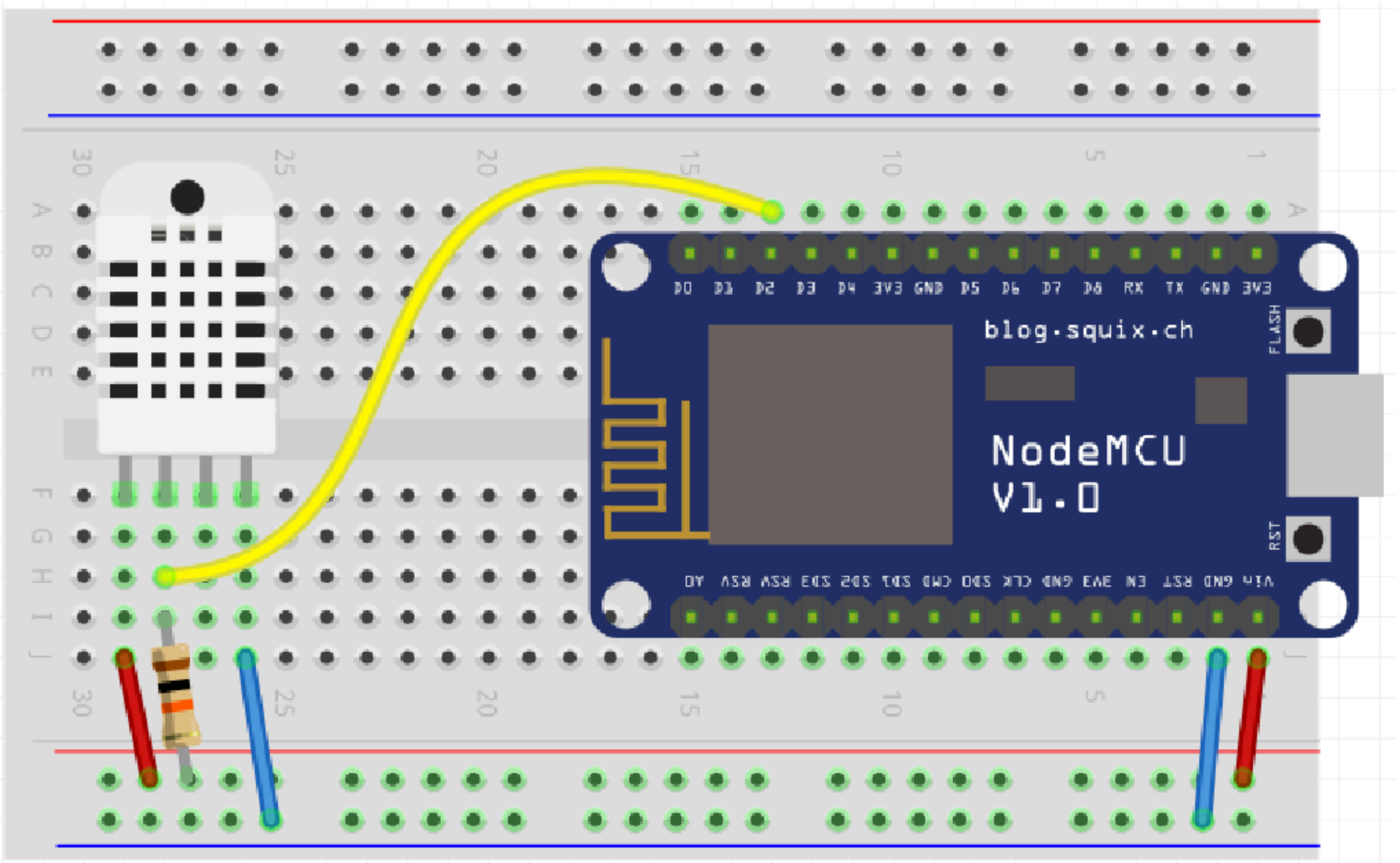
Hello World. aka blink

```
void setup() {
  // initialize digital pin LED_BUILTIN as an output.
  pinMode(LED_BUILTIN, OUTPUT);
}

// the loop function runs over and over again forever
void loop() {
  digitalWrite(LED_BUILTIN, HIGH);   // turn the LED on (HIGH is the voltage level)
  delay(1000);                       // wait for a second
  digitalWrite(LED_BUILTIN, LOW);    // turn the LED off by making the voltage LOW
  delay(1000);                       // wait for a second
}
```

Humidity and temperature

Connect the humidity and temperature sensor





[Made with Fritzing](http://www.fritzing.org)

DHT22Sketch

1. **global**

```
#include <ESP8266WiFi.h>      // general library for ESP8266 module
#include <DHT.h>              // library for DHT22 humidity and temparature sensor

#define DHTPIN D2             // the pin, the DHT22 is connected to
#define DHTTYPE DHT22         // the type of the humidity and temparature sensor (DHT22)
DHT dht(DHTPIN, DHTTYPE);     // initialising humidity and temparature sensor

unsigned long lastMillis = 0;                                    // variable used for timer
```

DHT22Sketch

2. setup() and loop()

```
void setup() {
  Serial.begin(115200);       // establishing serial connection for debugging
  Serial.println("Startup");  // serial output
}

void loop() {
  if (millis() - lastMillis > 1000) {    // timer for reading sensor data
    lastMillis = millis();               // reset timer
    sendSensorData();                    // method call
  }
}
```

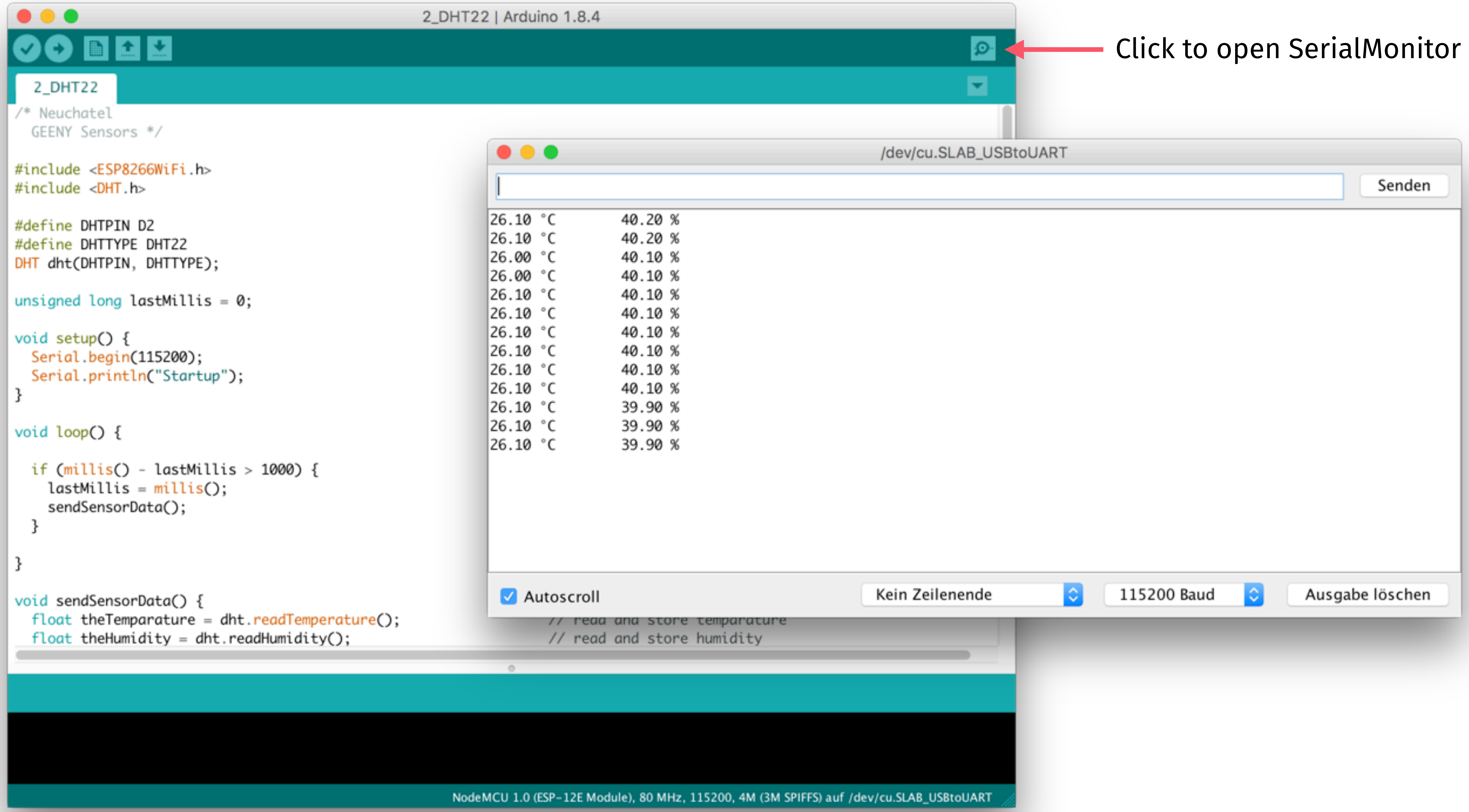
DHT22Sketch

3. sendSensorData()

```
void sendSensorData() {
  float theTemparature = dht.readTemperature();           // read and store temparature
  float theHumidity = dht.readHumidity();                 // read and store humidity
  if (isnan(theHumidity) || isnan(theTemparature)) {      // if data is invalid
    Serial.println("Failed to read from DHT sensor!“);    // serial output
    return;
  }

  Serial.print(theTemparature);   // serial output
  Serial.print(" °C\t");
  Serial.print(theHumidity);
  Serial.println(" %");
}
```

Connect the humidity and temperature sensor

4. **Open the Serial Monitor and check the data**



Click to open SerialMonitor

```
/* Neuchatel
  GEENY Sensors */

#include <ESP8266WiFi.h>
#include <DHT.h>

#define DHTPIN D2
#define DHTTYPE DHT22
DHT dht(DHTPIN, DHTTYPE);

unsigned long lastMillis = 0;

void setup() {
  Serial.begin(115200);
  Serial.println("Startup");
}

void loop() {

  if (millis() - lastMillis > 1000) {
    lastMillis = millis();
    sendSensorData();
  }

}

void sendSensorData() {
  float theTemparature = dht.readTemperature();                 // read and store temparature
  float theHumidity = dht.readHumidity();                       // read and store humidity
```

```
26.10 °C       40.20 %
26.10 °C       40.20 %
26.00 °C       40.10 %
26.00 °C       40.10 %
26.10 °C       40.10 %
26.10 °C       40.10 %
26.10 °C       40.10 %
26.10 °C       40.10 %
26.10 °C       40.10 %
26.10 °C       40.10 %
26.10 °C       39.90 %
26.10 °C       39.90 %
26.10 °C       39.90 %
```

NodeMCU 1.0 (ESP-12E Module), 80 MHz, 115200, 4M (3M SPIFFS) auf /dev/cu.SLAB_USBtoUART

GEENY

DHT22Wifi

DHT22Wifi

1. Change your Wifi on your computer to:

   ...

   Password:

   ...

DHT22Wifi

1. global

```
#include <ESP8266WiFi.h>
#include <DHT.h>
#include <ArduinoJson.h>
StaticJsonBuffer<200> jsonBuffer;
#define DHTPIN D2
#define DHTTYPE DHT22
DHT dht(DHTPIN, DHTTYPE);
const char* ssid     = "geenyworkshop";       // the wifi name (SSID)
const char* password = "g33nyworkshop";       // the wifi password
String sensorSpecifier = "12345";
WiFiClient net;                              // creating wifi object
unsigned long lastMillis = 0;
```

Change your Wifi on your computer to

...

Password:

...

DHT22Wifi

2. connect()

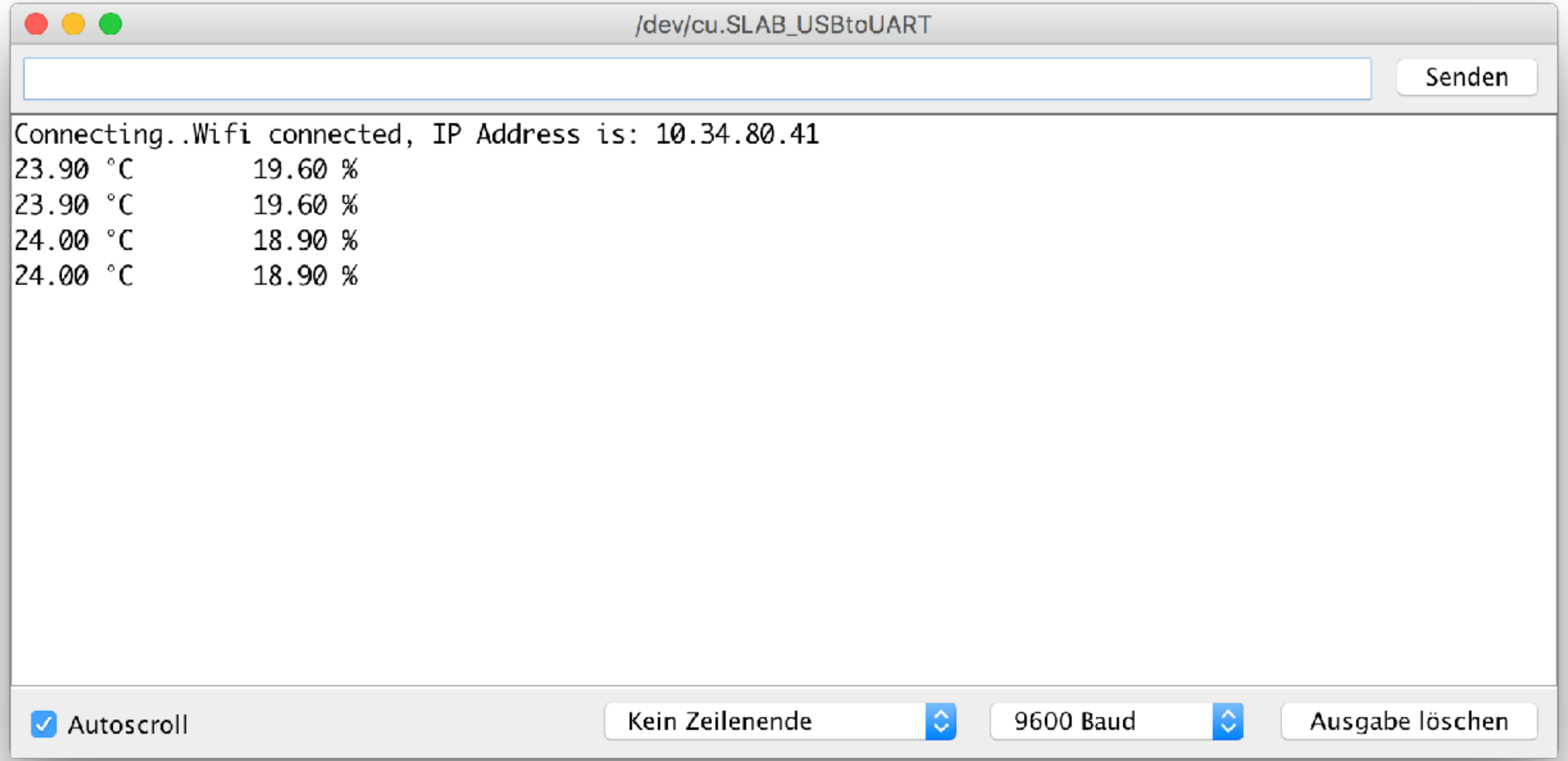
```
void connect() {
  Serial.print("checking wifi...");         // serial output
  while (WiFi.status() != WL_CONNECTED) {   // try to connect to wifi until success
    Serial.print(".");                      // serial output
    delay(1000);                            // wait for next atempt
  }
  Serial.println("\nconnected!");           // serial output
}
```

DHT22Wifi

3. setup() and loop()

```
void setup() {
  Serial.begin(115200);
  Serial.println("Startup");
  WiFi.begin(ssid, password);    // starting wifi connection
  connect();                     // connecting to Wifi
}

void loop() {
  if (millis() - lastMillis > 1000) {
    lastMillis = millis();
    sendSensorData();
  }
}
```

DHT22Wifi



```
Connecting..Wifi connected, IP Address is: 10.34.80.41
23.90 °C        19.60 %
23.90 °C        19.60 %
24.00 °C        18.90 %
24.00 °C        18.90 %
```

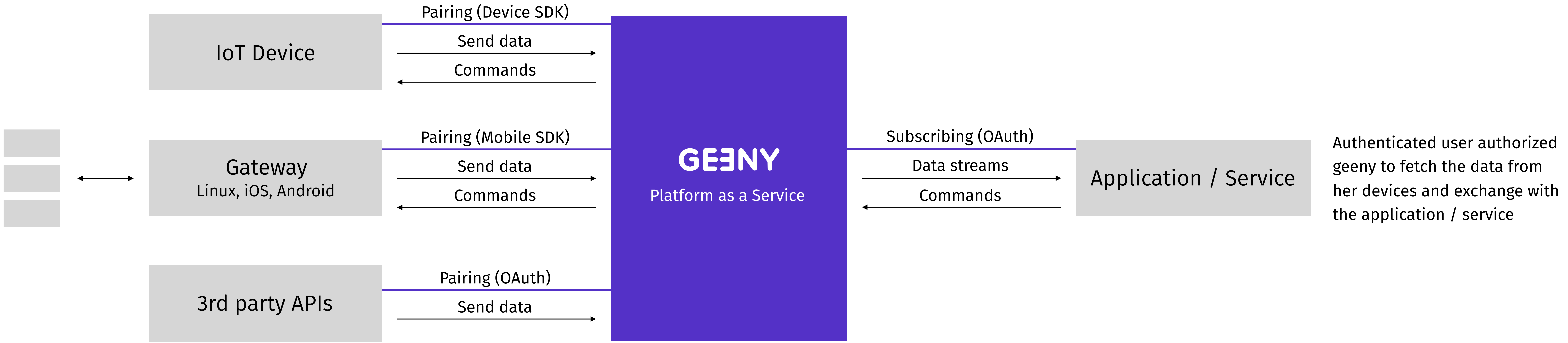
☑ Autoscroll

Kein Zeilenende

9600 Baud

What is Geeny?

Geeny is the consumer IoT platform by Telefónica NEXT



Authenticated user authorized geeny to fetch the data from her devices and exchange with the application / service

Send data to Geeny

First name
Last name
Email
Password
Repeat password
Create Account

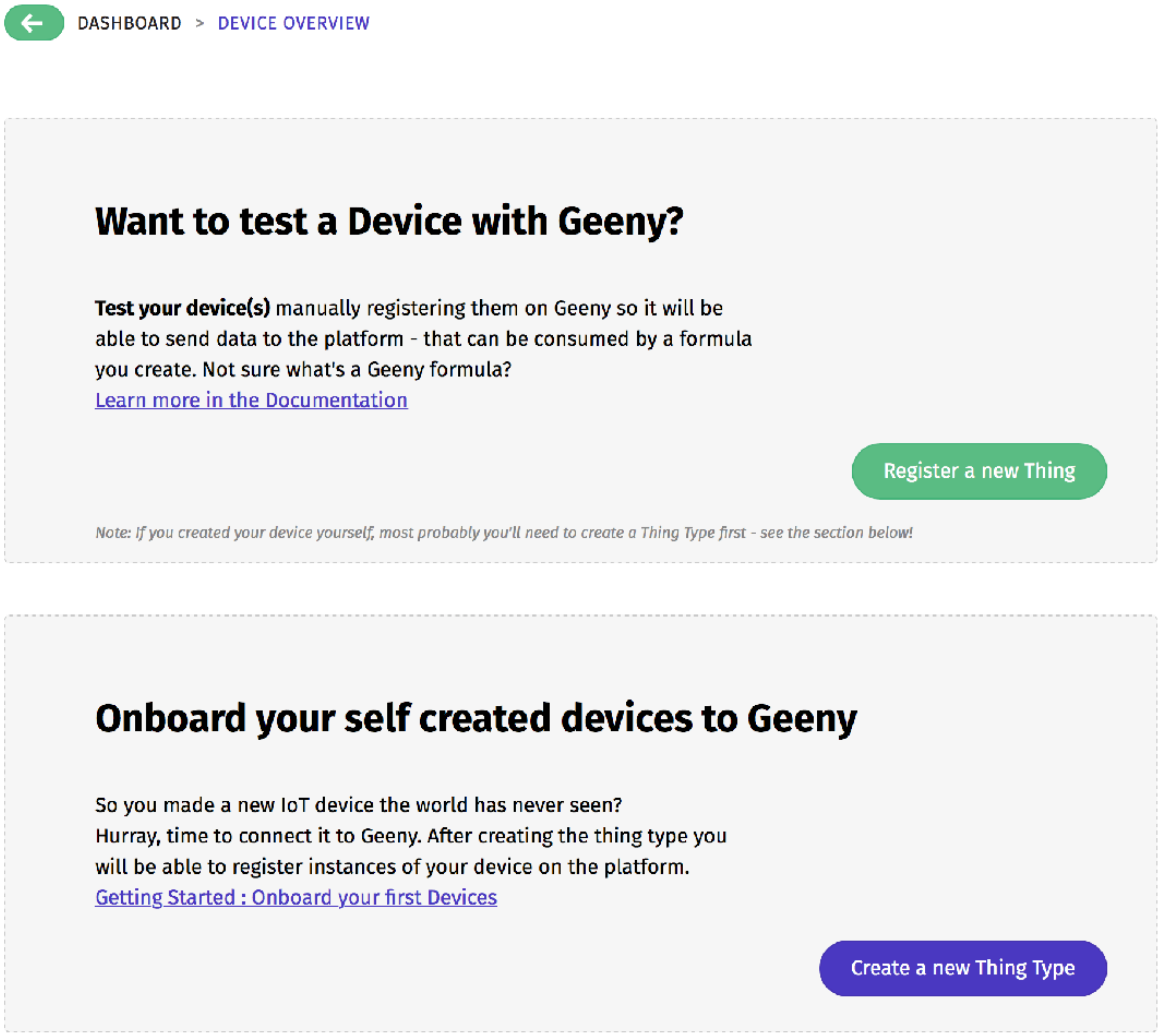
Create your Geeny Account

By creating an account you agree to Geeny's Consumer Terms of Service and Privacy policy.

You already have an account? Login

olicy Imprint Terms and Conditions

This website stores cookies on your computer. We use cookies to improve your website experience and to offer you personalized services. To learn more about the cookies please see our privacy policy. Learn more



Want to test a Device with Geeny?

Test your device(s) manually registering them on Geeny so it will be able to send data to the platform - that can be consumed by a formula you create. Not sure what's a Geeny formula?
Learn more in the Documentation

Note: If you created your device yourself, most probably you'll need to create a Thing Type first - see the section below!

Onboard your self created devices to Geeny

So you made a new IoT device the world has never seen?
Hurray, time to connect it to Geeny. After creating the thing type you will be able to register instances of your device on the platform.
Getting Started : Onboard your first Devices

GEENY / DEV

Let's get started!

Formula Quick Start

Launch a simple Hello World example and modify it to consume IoT data.

Get Started with Formulas

Devices Quick Start

Use your computer as a "virtual Thing" — it will act like an IoT device you register with Geeny and send data.

Get Started with Devices

Formulas

Deploy your Apps & Services onto Geeny platform.

Devices

All steps to onboard your Device to Geeny.

Data Explorer

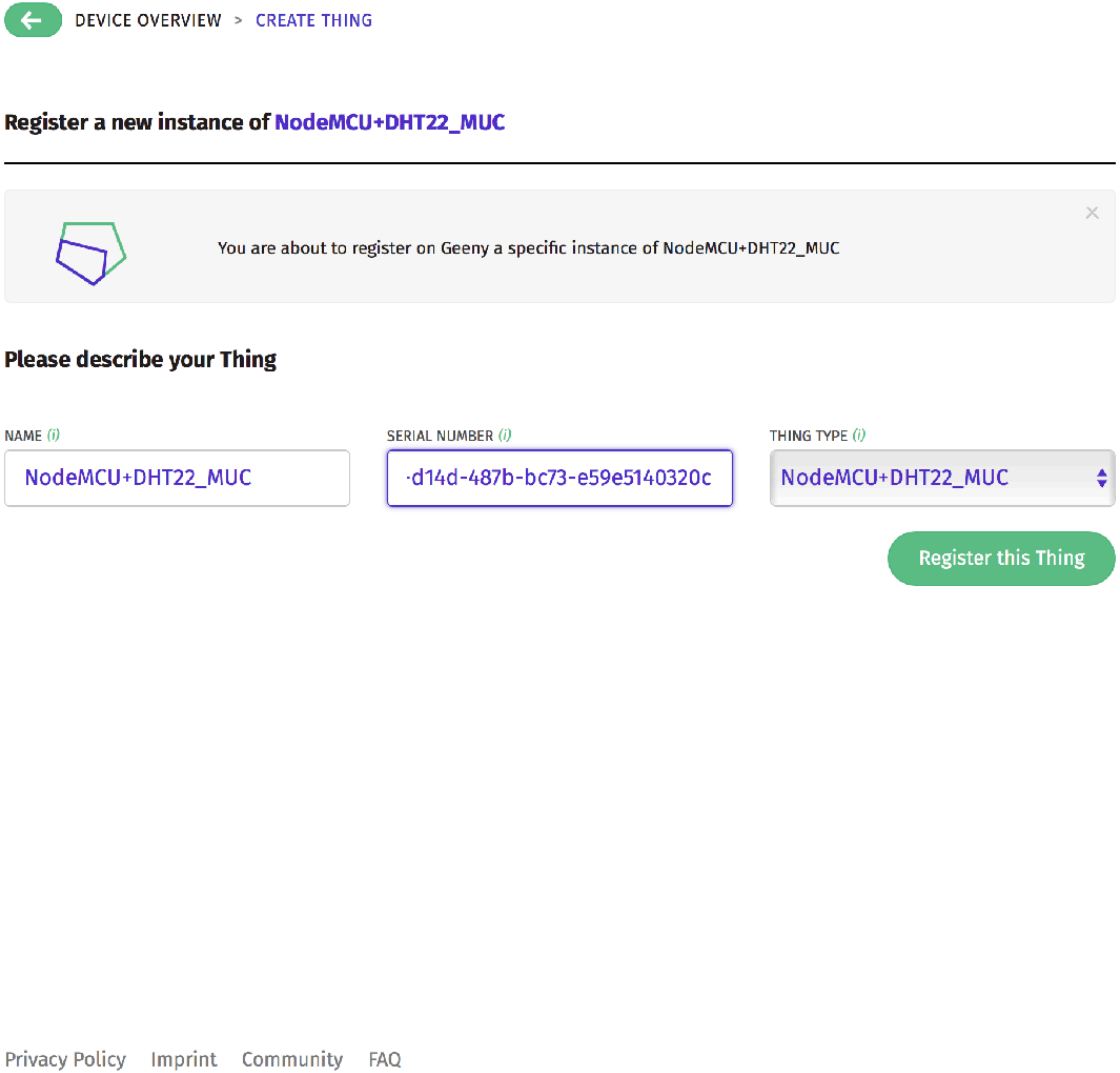
Add Elements & Message Types to your Formula

Documentation Center

Getting Started, Platform Overview, API & SDK Reference

Get Support

From our Community



Register a new instance of NodeMCU+DHT22_MUC

You are about to register on Geeny a specific instance of NodeMCU+DHT22_MUC

Please describe your Thing

NAME (i): NodeMCU+DHT22_MUC

SERIAL NUMBER (i): ·d14d-487b-bc73-e59e5140320c

THING TYPE (i): NodeMCU+DHT22_MUC

Congrats! You are doing great!
But take care, now a very important step 🤓

During communication Thing and Platform will need to authenticate each other via certificates. Make sure you download all the three certificates below and store them on a secure place.

NOTE: The certificates ARE NOT AVAILABLE for later download you MUST download them NOW!

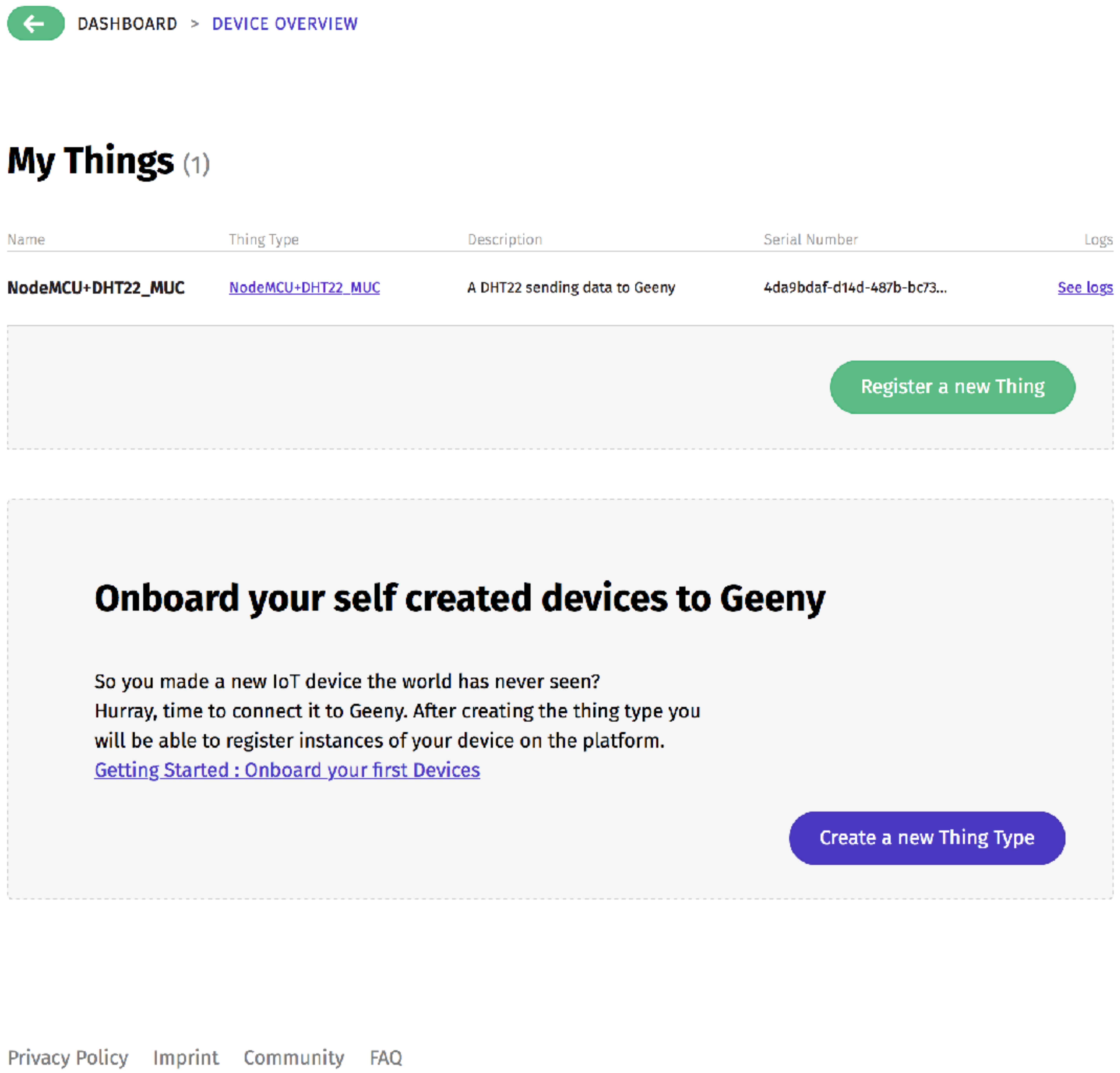
- CRT geeny-ca.crt
- CRT thing.crt
- KEY thing.key

Congrats! You are doing great!
But take care, now a very important step 🤓

During communication Thing and Platform will need to authenticate each other via certificates. Make sure you download all the three certificates below and store them on a secure place.

NOTE: The certificates ARE NOT AVAILABLE for later download you MUST download them NOW!

CRT
geeny-ca.crt
CRT
thing.crt
KEY
thing.key
I have downloaded all the certificates

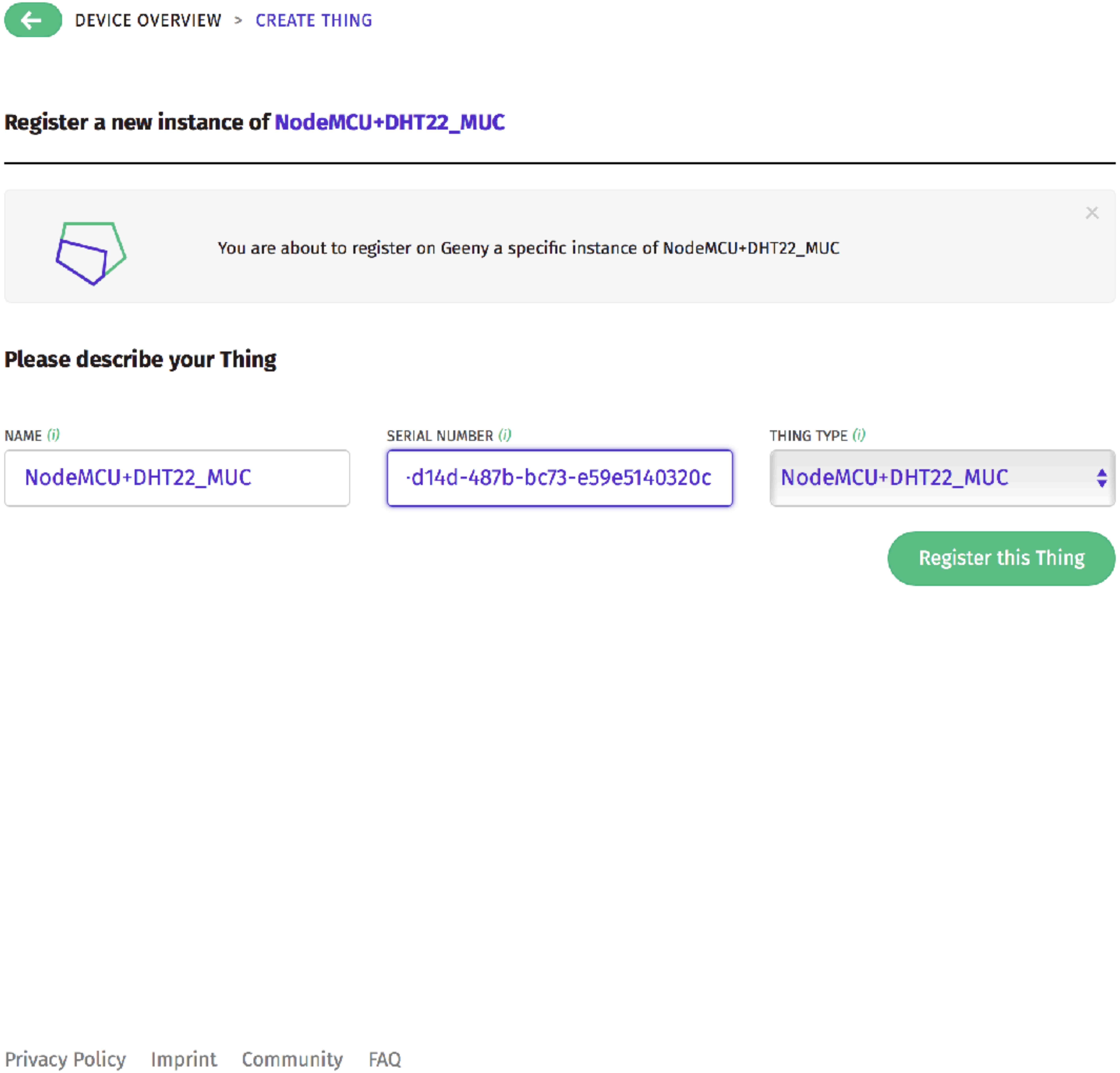


My Things (1)

Name	Thing Type	Description	Serial Number	Logs
NodeMCU+DHT22_MUC	NodeMCU+DHT22_MUC	A DHT22 sending data to Geeny	4da9bdaf-d14d-487b-bc73...	See logs

Onboard your self created devices to Geeny

So you made a new IoT device the world has never seen?
Hurray, time to connect it to Geeny. After creating the thing type you will be able to register instances of your device on the platform.
Getting Started : Onboard your first Devices



Register a new instance of NodeMCU+DHT22_MUC

You are about to register on Geeny a specific instance of NodeMCU+DHT22_MUC

Please describe your Thing

NAME (i): NodeMCU+DHT22_MUC

SERIAL NUMBER (i): ·d14d-487b-bc73-e59e5140320c

THING TYPE (i): NodeMCU+DHT22_MUC

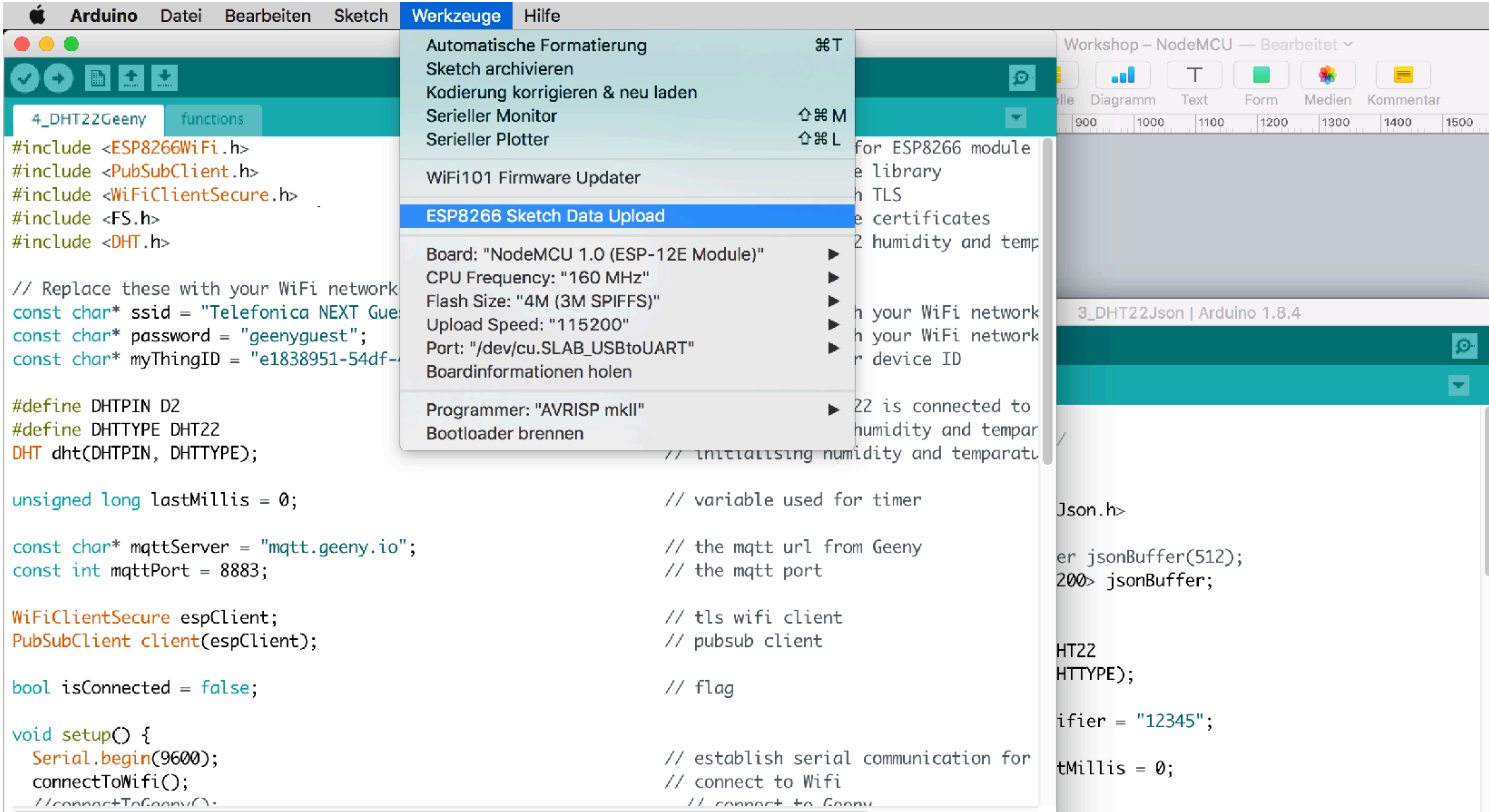
```
openssl rsa -outform der -in thing.key -out thing-key.der
openssl x509 -outform der -in thing.crt -out thing-crt.der
```

Setup

1. **Install ESP8266FS – A upload tool for files to the NodeMCU**

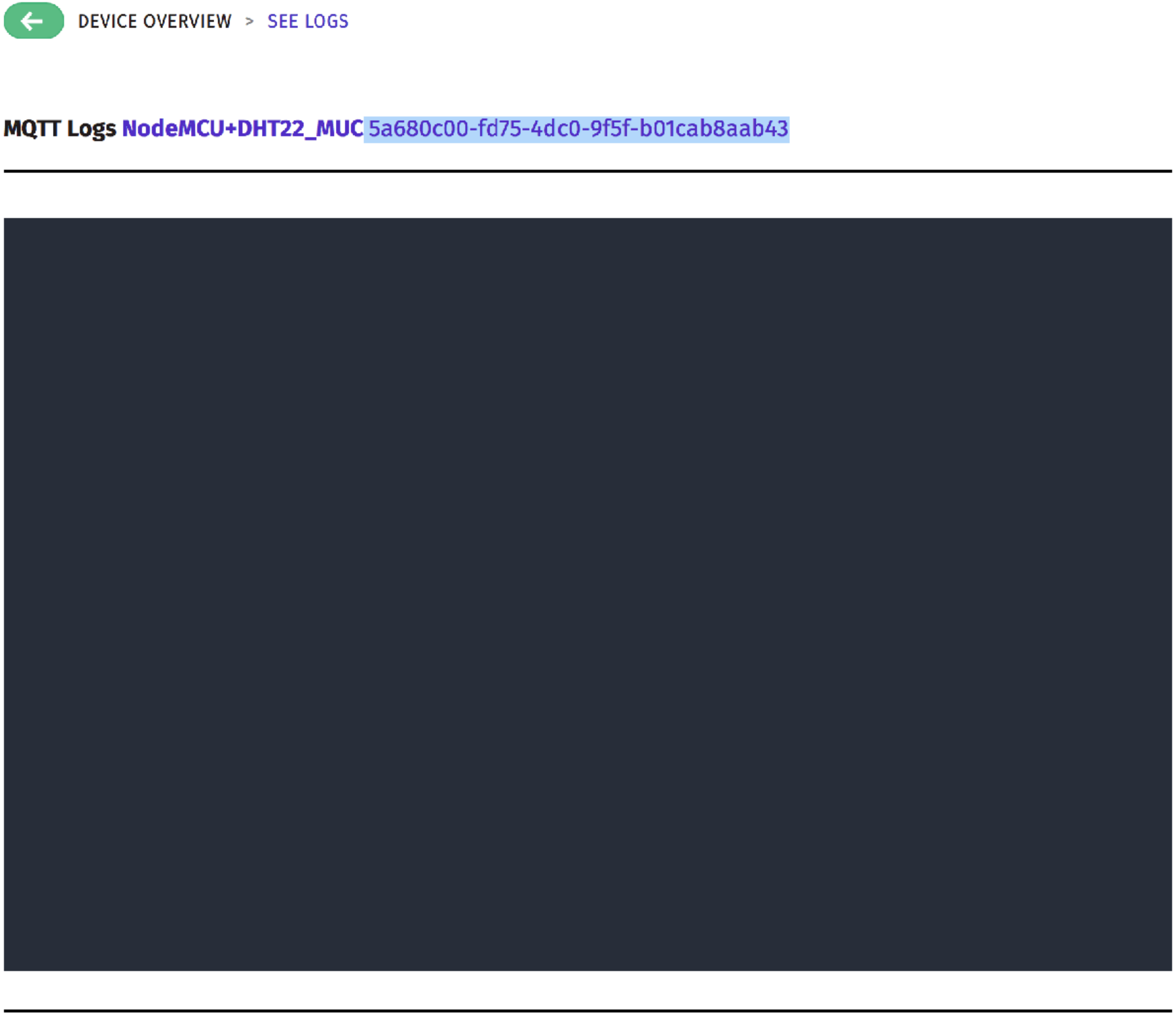
    1. Download the tool: https://github.com/esp8266/arduino-esp8266fs-plugin/releases/download/0.2.0/ESP8266FS-0.2.0.zip
    2. In your Arduino sketchbook directory, create tools directory if it doesn't exist yet
    3. Unpack the tool into tools directory

       (the path will look like<home_dir>/Arduino/tools/ESP8266FS/tool/esp8266fs.jar)
    4. Restart Arduino IDE

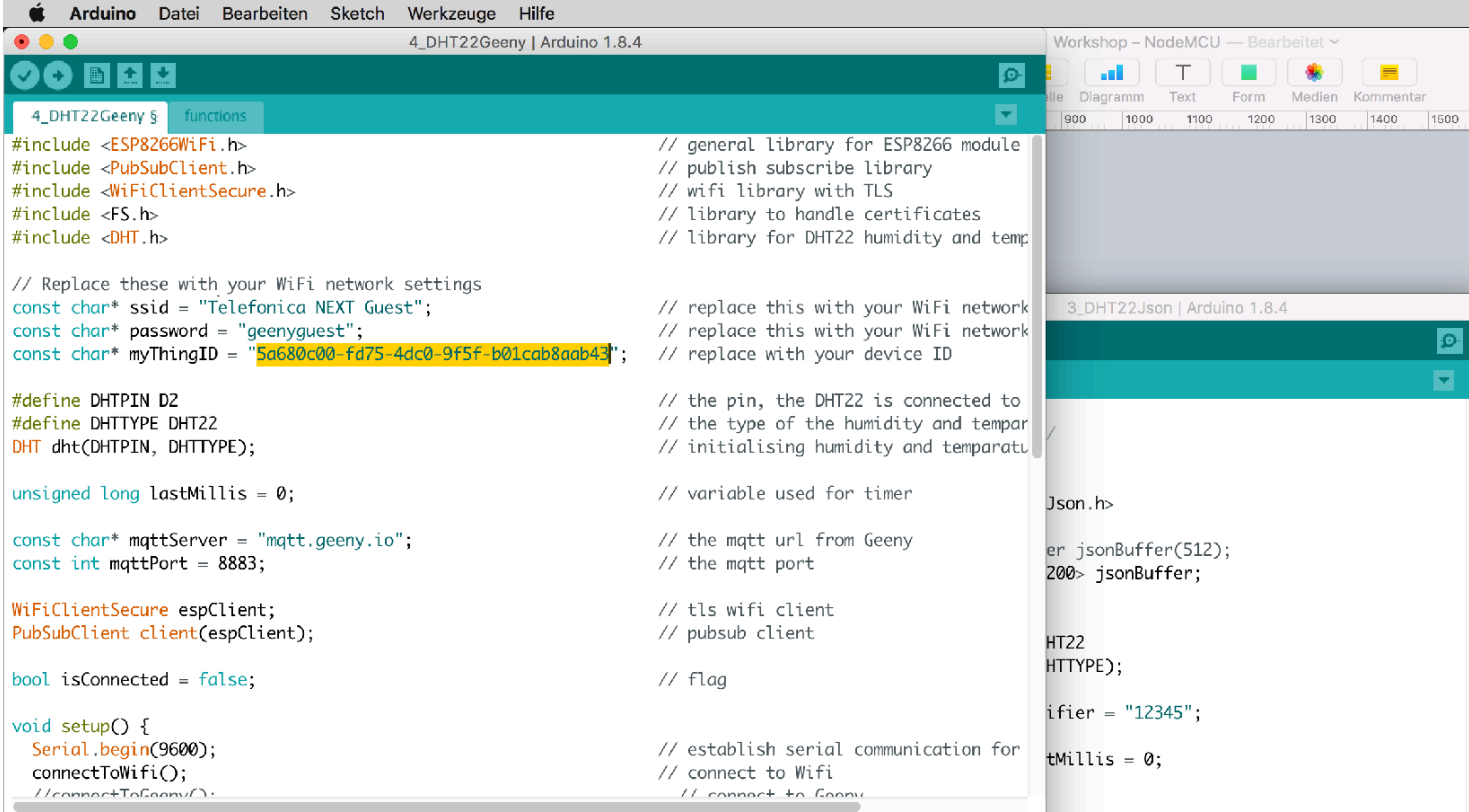


```
#include <ESP8266WiFi.h>                                              for ESP8266 module
#include <PubSubClient.h>                                             e library
#include <WiFiClientSecure.h>                                         h TLS
#include <FS.h>                                                       e certificates
#include <DHT.h>                                                      2 humidity and temp

// Replace these with your WiFi network
const char* ssid = "Telefonica NEXT Gue                               h your WiFi network
const char* password = "geenyguest";                                  h your WiFi network
const char* myThingID = "e1838951-54df-                               r device ID

#define DHTPIN D2                                                     22 is connected to
#define DHTTYPE DHT22                                                 humidity and tempar
DHT dht(DHTPIN, DHTTYPE);                                // initialising humidity and temparatu

unsigned long lastMillis = 0;                            // variable used for timer

const char* mqttServer = "mqtt.geeny.io";                // the mqtt url from Geeny
const int mqttPort = 8883;                               // the mqtt port

WiFiClientSecure espClient;                              // tls wifi client
PubSubClient client(espClient);                          // pubsub client

bool isConnected = false;                                // flag

void setup() {
  Serial.begin(9600);                                    // establish serial communication for
  connectToWifi();                                       // connect to Wifi
  //connectToGeeny();                                    // connect to Geeny
```

```
/
Json.h>

er jsonBuffer(512);
200> jsonBuffer;

HT22
HTTYPE);

ifier = "12345";

tMillis = 0;
```

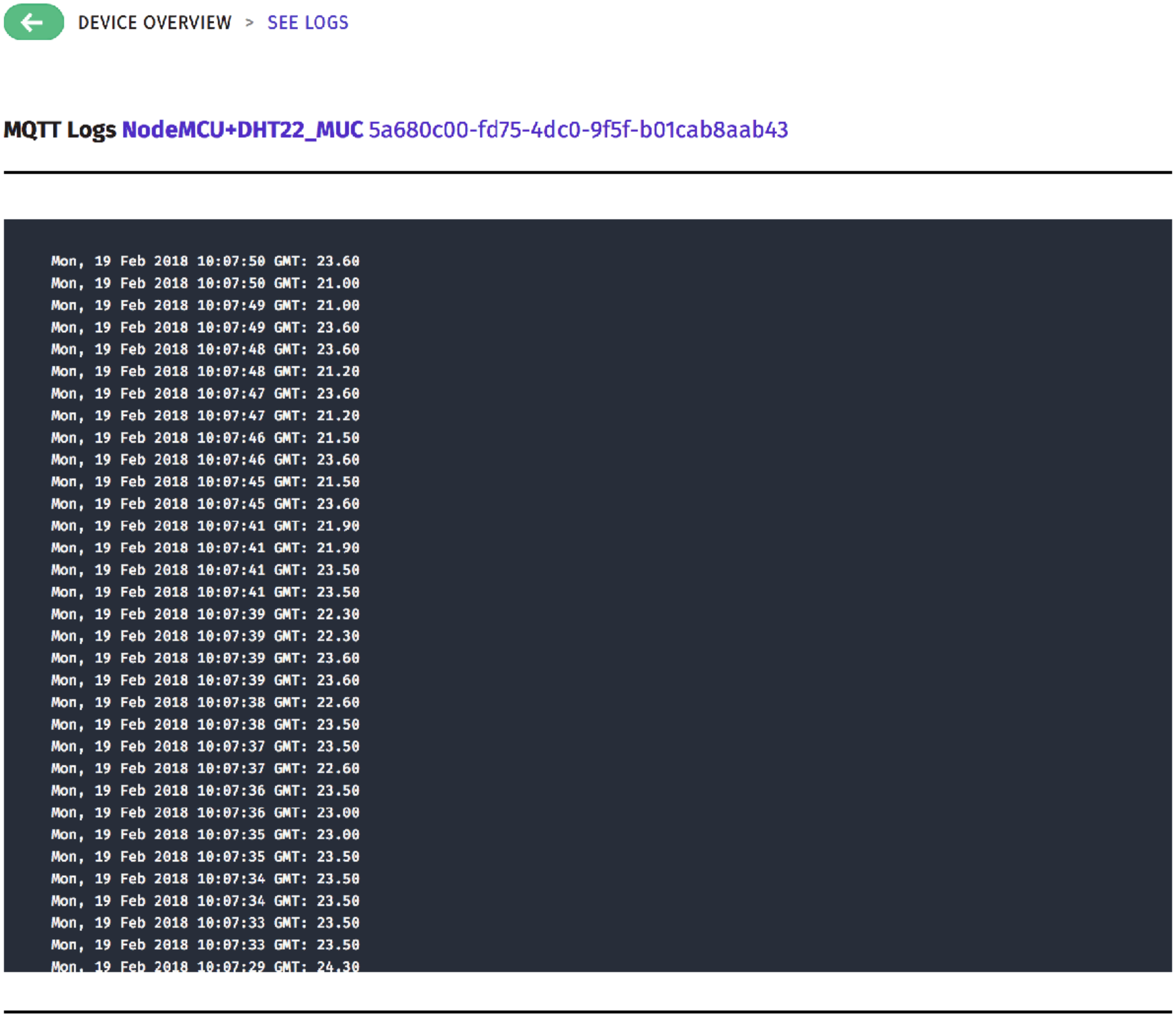


MQTT Logs NodeMCU+DHT22_MUC 5a680c00-fd75-4dc0-9f5f-b01cab8aab43



```
#include <ESP8266WiFi.h>                                          // general library for ESP8266 module
#include <PubSubClient.h>                                         // publish subscribe library
#include <WiFiClientSecure.h>                                     // wifi library with TLS
#include <FS.h>                                                   // library to handle certificates
#include <DHT.h>                                                  // library for DHT22 humidity and temp

// Replace these with your WiFi network settings
const char* ssid = "Telefonica NEXT Guest";                       // replace this with your WiFi network
const char* password = "geenyguest";                              // replace this with your WiFi network
const char* myThingID = "5a680c00-fd75-4dc0-9f5f-b01cab8aab43";   // replace with your device ID

#define DHTPIN D2                                                 // the pin, the DHT22 is connected to
#define DHTTYPE DHT22                                             // the type of the humidity and tempar
DHT dht(DHTPIN, DHTTYPE);                                         // initialising humidity and temparatu

unsigned long lastMillis = 0;                                     // variable used for timer

const char* mqttServer = "mqtt.geeny.io";                         // the mqtt url from Geeny
const int mqttPort = 8883;                                        // the mqtt port

WiFiClientSecure espClient;                                       // tls wifi client
PubSubClient client(espClient);                                   // pubsub client

bool isConnected = false;                                         // flag

void setup() {
  Serial.begin(9600);                                             // establish serial communication for
  connectToWifi();                                                // connect to Wifi
  //connectToGeeny();                                             // connect to Geeny
```

```
/
Json.h>
er jsonBuffer(512);
200> jsonBuffer;
HT22
HTTYPE);
ifier = "12345";
tMillis = 0;
```



MQTT Logs NodeMCU+DHT22_MUC 5a680c00-fd75-4dc0-9f5f-b01cab8aab43

```
Mon, 19 Feb 2018 10:07:50 GMT: 23.60
Mon, 19 Feb 2018 10:07:50 GMT: 21.00
Mon, 19 Feb 2018 10:07:49 GMT: 21.00
Mon, 19 Feb 2018 10:07:49 GMT: 23.60
Mon, 19 Feb 2018 10:07:48 GMT: 23.60
Mon, 19 Feb 2018 10:07:48 GMT: 21.20
Mon, 19 Feb 2018 10:07:47 GMT: 23.60
Mon, 19 Feb 2018 10:07:47 GMT: 21.20
Mon, 19 Feb 2018 10:07:46 GMT: 21.50
Mon, 19 Feb 2018 10:07:46 GMT: 23.60
Mon, 19 Feb 2018 10:07:45 GMT: 21.50
Mon, 19 Feb 2018 10:07:45 GMT: 23.60
Mon, 19 Feb 2018 10:07:41 GMT: 21.90
Mon, 19 Feb 2018 10:07:41 GMT: 21.90
Mon, 19 Feb 2018 10:07:41 GMT: 23.50
Mon, 19 Feb 2018 10:07:41 GMT: 23.50
Mon, 19 Feb 2018 10:07:39 GMT: 22.30
Mon, 19 Feb 2018 10:07:39 GMT: 22.30
Mon, 19 Feb 2018 10:07:39 GMT: 23.60
Mon, 19 Feb 2018 10:07:39 GMT: 23.60
Mon, 19 Feb 2018 10:07:38 GMT: 22.60
Mon, 19 Feb 2018 10:07:38 GMT: 23.50
Mon, 19 Feb 2018 10:07:37 GMT: 23.50
Mon, 19 Feb 2018 10:07:37 GMT: 22.60
Mon, 19 Feb 2018 10:07:36 GMT: 23.50
Mon, 19 Feb 2018 10:07:36 GMT: 23.00
Mon, 19 Feb 2018 10:07:35 GMT: 23.00
Mon, 19 Feb 2018 10:07:35 GMT: 23.50
Mon, 19 Feb 2018 10:07:34 GMT: 23.50
Mon, 19 Feb 2018 10:07:34 GMT: 23.50
Mon, 19 Feb 2018 10:07:33 GMT: 23.50
Mon, 19 Feb 2018 10:07:33 GMT: 23.50
Mon, 19 Feb 2018 10:07:29 GMT: 24.30
```

Thank you very much!

Let's stay in touch.

Geeny

geeny.io
@geenyio

Stefan Hermann

stefan.h@geeny.io
@stefanhermann