

GTA Mechatronik Teil 6

Wir überwachen Sensordaten über das Internet
und lernen dabei viel über IoT und MQTT

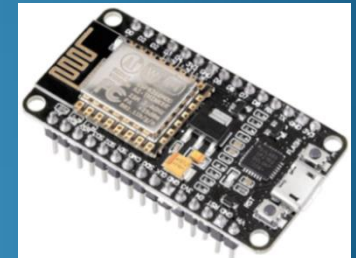
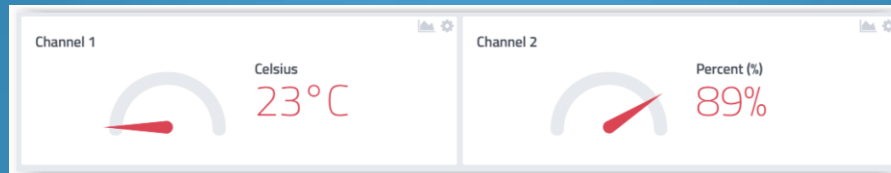


The Messaging and Data Exchange Protocol of the IoT

Cayenne

The world's first drag-and-drop IoT project builder

```
Sketch_MQTT-Cayenne_DHT11 §  
9 #include <CayenneMQTTESP8266.h>  
10 #include <DHT.h>  
11  
12 char ssid[] = "AndroidAP3232";  
13 char wifiPassword[] = "20c93d.....";  
14  
15 /*Cayenne Zugangsdaten (authentication data)  
16 char username[] = "8f0a4cc0-9296-11ed-b193-d  
17 char password[] = "0457a81043fcde45a6fced08f  
18 char clientID[] = "6a7c4740-9297-11ed-8d53-d  
19  
20 DHT dht (D2, DHT11);  
21
```



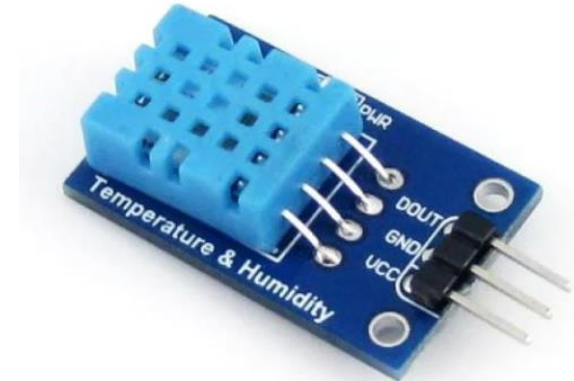
IoT Internet of Things (Internet der Dinge)

https://www.youtube.com/watch?v=NE697_h5nPY



[IoT mit dem Funkstandard LoRaWAN](#)

Temperatur- und Feuchtesensor DHT11 und DHT22



	1x AZDelivery DHT22 AM2302 mit Platine	1 x AZDelivery DHT11
Abmessungen	15 x 25 x 7,7mm (ohne Platine)	15 x 32 x 9mm
Temperatur- und Luftfeuchtigkeitbetriebsbereich	-40~80Celsius / 0-100%RH	0°C - 50°C / 20% - 90%
Genauigkeit (Temperatur und Luftfeuchtigkeit)	+/-0.5°C / +/-2%RH(Max +/-5%RH)	0°C - 50°C mit einer Genauigkeit von ±2°C

MQTT Message Queue Telemetry Transport



<https://mqtt.org/>

MQTT, ist ein äußerst einfach aufgebautes binäres Datenübertragungsprotokoll im Internet und eine Alternative zu HTTP (HyperText Transfer Protocol).

Es eignet sich für den Nachrichtenaustausch

- zwischen Geräten mit geringer Funktionalität, beispielsweise Mikrocontrollern wie dem Arduino
- für die Übertragung über unzuverlässige Netze mit geringer Bandbreite und hoher Latenz.

Mit diesen Eigenschaften spielt MQTT eine wichtige Rolle für das Internet der Dinge (IoT) und in der M2M (Machine To Machine) Kommunikation.

MQTT wurde bereits im Jahr 1999 entwickelt, zur Überwachung von Öl-Plattformen.

Die Anforderungen an das Protokoll waren unter anderem, dass es auch in unzuverlässigen Netzwerken funktioniert, in denen es zu Verbindungsunterbrechungen kommen kann.

Die aktuelle Version ist MQTT Version 5.0.

<https://www.youtube.com/watch?v=kZDhyqrmbJo>

<https://www.youtube.com/watch?v=NXyf7tVsi10>

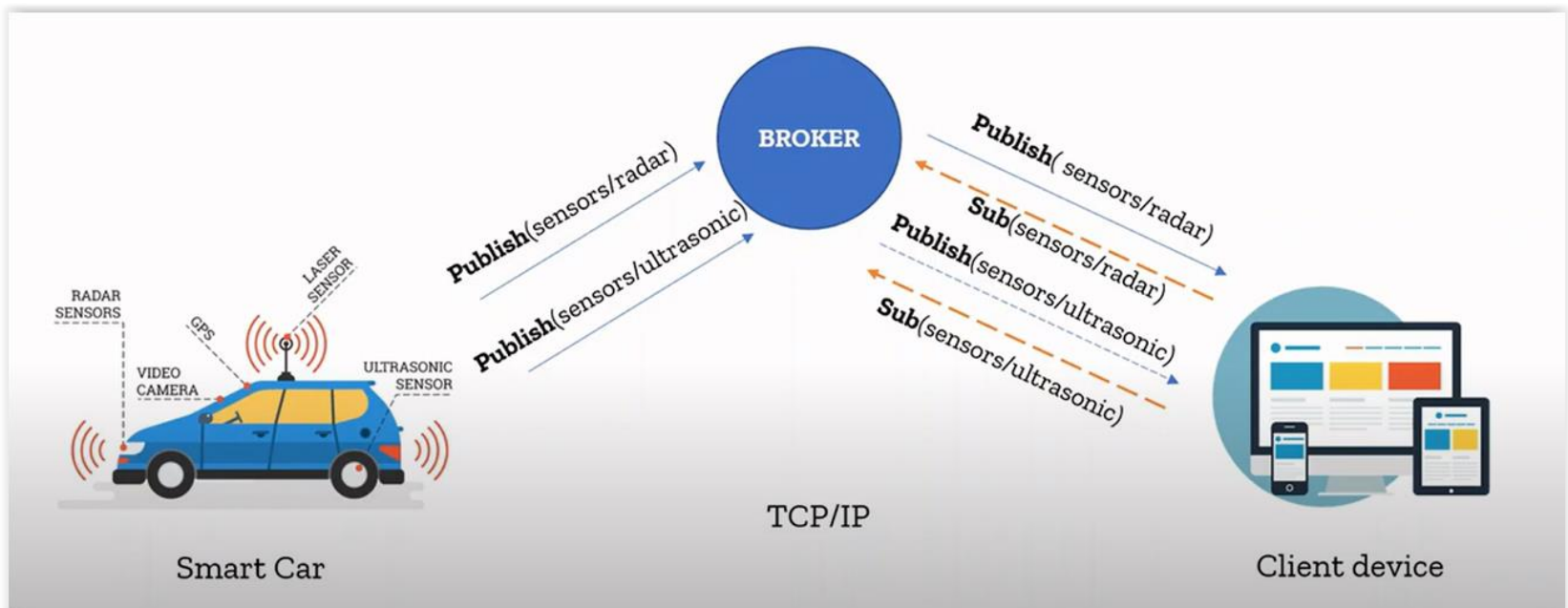
MQTT - Publish/Subscribe Protocol

MQTT verwendet das sogenannte Publish/Subscribe Protokoll, welches auch kurz als Pub/Sub bezeichnet wird.

Die als Publisher (Herausgeber) bezeichneten MQTT-Clients können Nachrichten unter sogenannten Topics an den MQTT-Broker (Vermittler) schicken. Topics sind ähnlich wie URLs hierarchisch aufgebaut, wobei einzelne Hierarchie-Ebenen durch einen Slash getrennt werden, zum Beispiel *factory1/hall2/sensorA*.

Diejenigen MQTT-Clients, die sich für ein Topic registrieren und hierunter auf Nachrichten „lauschen“, werden dagegen Subscriber (Abonnenten) genannt.

Der Broker fungiert dann als „Mediator“ zwischen Publishern und Subscribern: Er empfängt Nachrichten von den Publishern unter den jeweiligen Topics und leitet sie entsprechend an die Subscriber weiter, die sich für das jeweilige Topic registriert haben.



MQTT – Quality of Service

MQTT bietet verschiedene sogenannte Servicequalitätslevel an (QoS - Quality of Service), deren Einsatz abhängig von der Stabilität des Netzwerks sinnvoll sein kann.

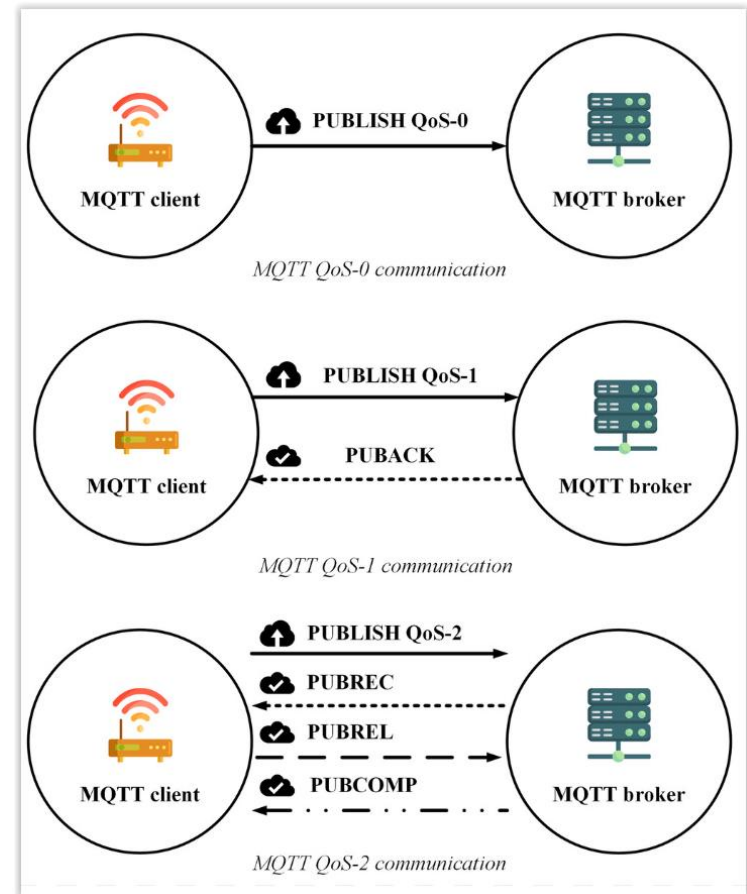
So lässt sich hierüber steuern, ob

- Nachrichten nur einmal (QoS – 0),
- mindestens einmal (QoS - 1)
- exakt einmal (QoS - 2) übertragen werden.

Default (voreingestellter Standard) ist QoS-0.

Um mit Verbindungsabbrüchen zwischen Clients und Dem Server (MQTT-Broker) umgehen zu können, verfügt MQTT zudem über Sessions:

Nachrichten, die aufgrund eines Verbindungsabbruchs nicht zugestellt werden konnten, können mit Hilfe von Sessions zwischengespeichert und zu einem späteren Zeitpunkt gesendet werden, wenn die entsprechende Verbindung wiederhergestellt wurde.



Cayenne am Computer – SIGN UP FREE und SIGN IN

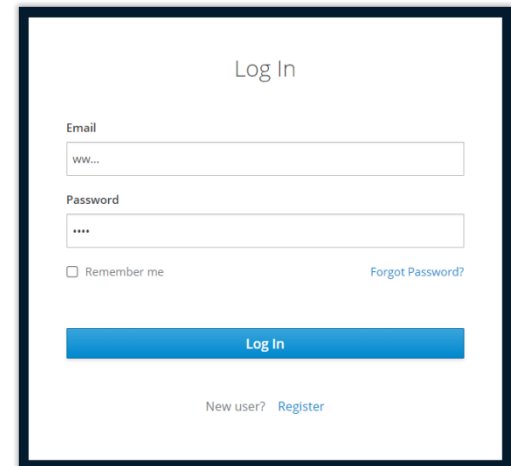
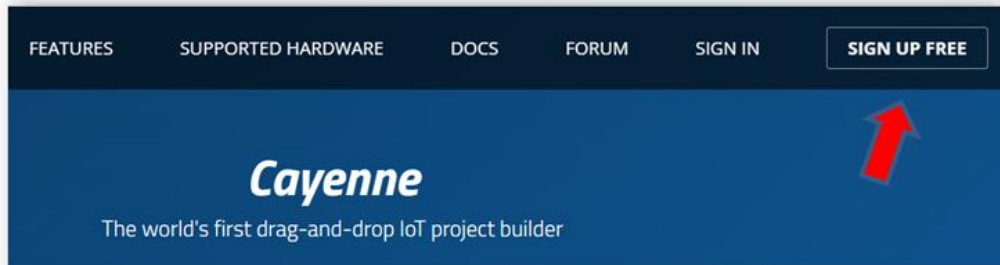
Cayenne ist ein Internetdienst, der MQTT nutzt und von [MyDevices](#) angeboten wird. Er macht es sehr einfach, IoT-Projekte zu visualisieren.

Es wird auch eine kostenlose Version angeboten, allerdings mit eingeschränktem Umfang.

Ein Zugriff auf die Daten ist auch über Smartphone möglich, allerdings gibt es keine App dazu im Apple oder Google Play Store, sondern als sogenannte Progressive Web Application ([PWA 439](#)).

<https://developers.mydevices.com/cayenne/features/>

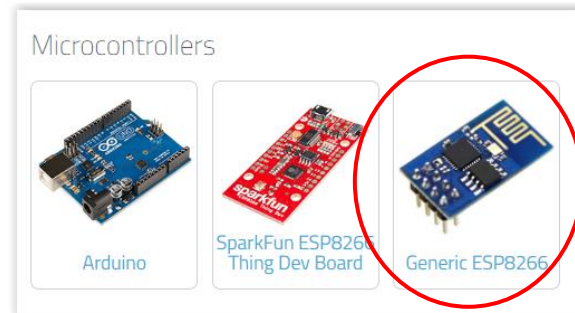
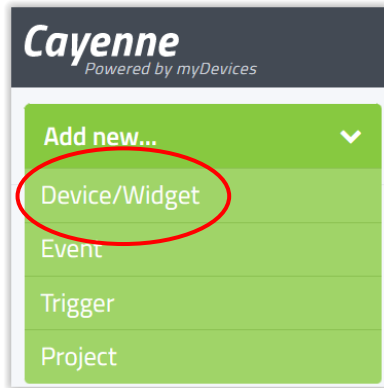
Nach einer Anmeldung mit SIGN UP FREE startet man Cayenne später mit SIGN IN.



Der Start von Cayenne und das Anlegen der folgenden zwei Sketche wird hier ausführlich erklärt:

https://www.youtube.com/watch?v=6YAnqHTm7_o

Device anlegen und Zugangsdaten erhalten



Connect your Device

SELECT YOUR BOARD

Generic ESP8266



STEPS TO CONNECT

- Install the [Arduino IDE](#) and add [Cayenne MQTT Library](#) to Arduino IDE.
- Install the ESP8266 board package to Arduino IDE. [show me how](#)
- Install required USB driver on your computer so you can program the ESP8266. [show me how](#)
- Connect the ESP8266 to your PC/Mac via data-capable USB cable.
- In the Arduino IDE, go to the **tools** menu, select your **board**, and the **port** your ESP8266 is connected to.

MQTT USERNAME:

8f0a4cc0-9296-11ed-b193-d9789b2af62b



MQTT PASSWORD:

0457a81043fcde45a6fced08f7ee2273cfb74947



CLIENT ID:

6a7c4740-9297-11ed-8d53-d7cd1025126a



MQTT SERVER:

mqtt.mydevices.com

MQTT PORT:

1883

NAME YOUR DEVICE (optional):

Generic ESP8266

⚙️ Waiting for board to connect...

Sketche

Sketch_MQTT-Cayenne_LED

Vom Cayenne-Dashboard soll über das Internet die interne LED D0 des ESP8266 ein/ausgeschaltet werden.

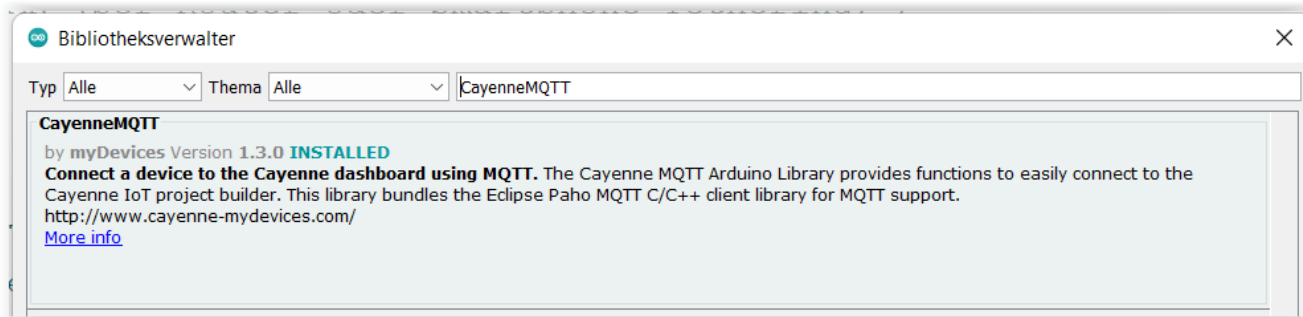
The screenshot shows the Cayenne dashboard interface for adding a new widget. On the left, a sidebar menu has 'Project' circled in red. Below it, 'Custom Widgets' is also circled in red. The main area displays various widget options, with a 'Button' widget selected and circled in red. To the right, the 'Enter Settings' panel for the 'Button' widget is shown, with the 'Name' field set to 'LED D0' and the 'Add Widget' button at the bottom right circled in red.

This screenshot shows the Cayenne dashboard after the 'LED D0' widget has been added. The widget is visible in the 'Overview' and 'Data' tabs, and the 'LED D0' label is circled in red.

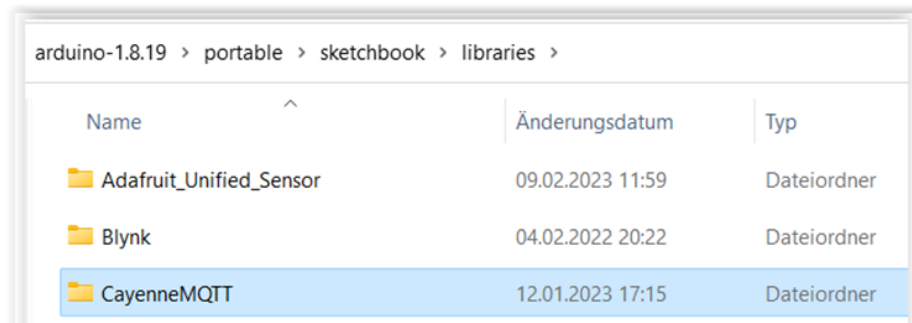
This screenshot shows the Cayenne dashboard after the 'LED D0' widget has been added. The widget is visible in the 'Overview' and 'Data' tabs, and the 'LED D0' label is circled in red.

Sketch_MQTT-Cayenne_LED


Für den Sketch im ESP8266 installieren wir zuerst das Bibliotheksprogramm „CayenneMQTT“.



Es wird dabei ein umfangreiches Paket heruntergeladen.



Wir müssen die Header-Datei

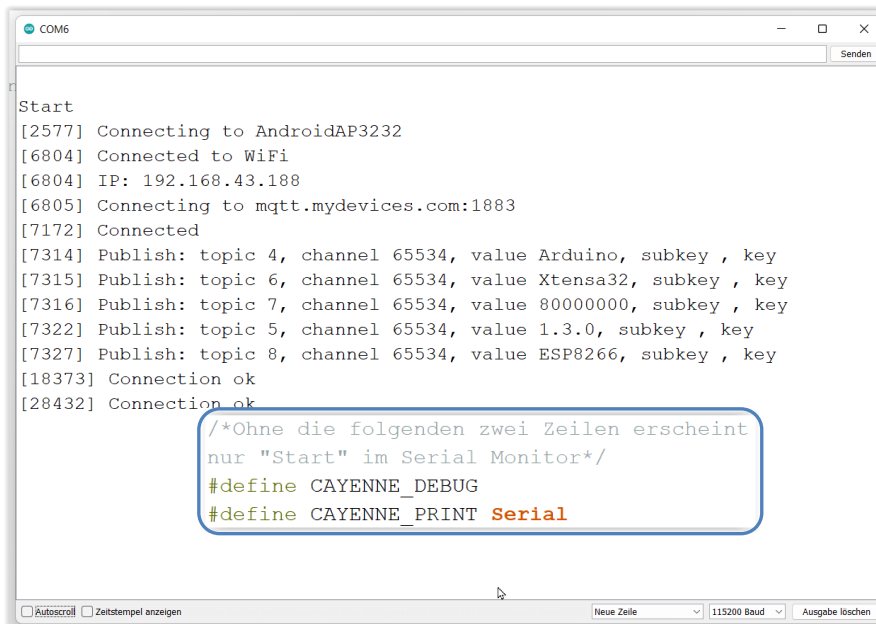
 CayenneMQTTESP8266.h

in den Sketch einbinden.

Sketch_MQTT-Cayenne_LED

Jetzt starten: Sketch_MQTT-Cayenne_LED.

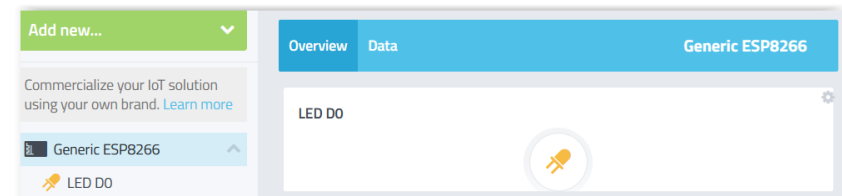
Im Serial Monitor erscheint (wenn der ESP mit dem Internet verbunden ist, z.B. über Tethering des Smartphones):



```
COM6
Start
[2577] Connecting to AndroidAP3232
[6804] Connected to WiFi
[6804] IP: 192.168.43.188
[6805] Connecting to mqtt.mydevices.com:1883
[7172] Connected
[7314] Publish: topic 4, channel 65534, value Arduino, subkey , key
[7315] Publish: topic 6, channel 65534, value Xtensa32, subkey , key
[7316] Publish: topic 7, channel 65534, value 80000000, subkey , key
[7322] Publish: topic 5, channel 65534, value 1.3.0, subkey , key
[7327] Publish: topic 8, channel 65534, value ESP8266, subkey , key
[18373] Connection ok
[28432] Connection ok

/*Ohne die folgenden zwei Zeilen erscheint
nur "Start" im Serial Monitor*/
#define CAYENNE_DEBUG
#define CAYENNE_PRINT Serial
```

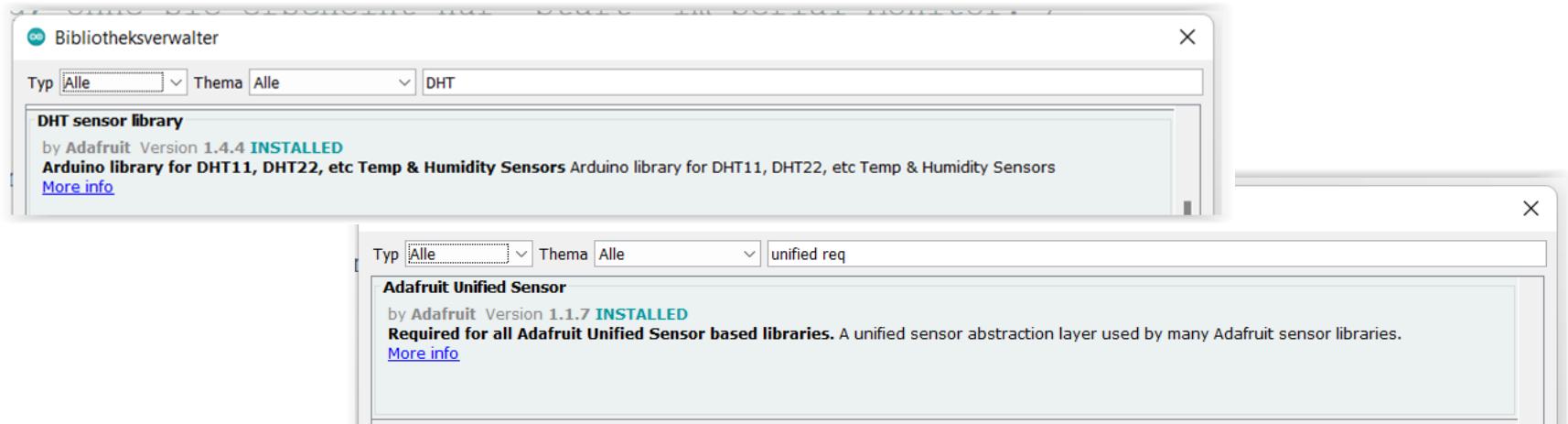
Am Dashboard kann jetzt die LED geschaltet werden:



Sketch_MQTT_Cayenne_DHT11

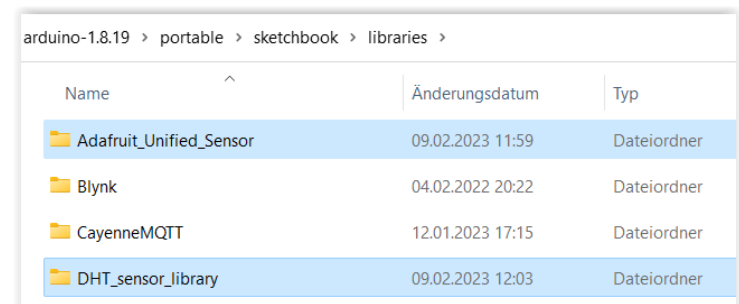
Wir wollen vom Sensor DHT11 die Temperatur und Luftfeuchte messen und über das Internet an das Cayenne-Dashboard übertragen.

Dazu werden zwei weitere Bibliotheksprogramme installiert:



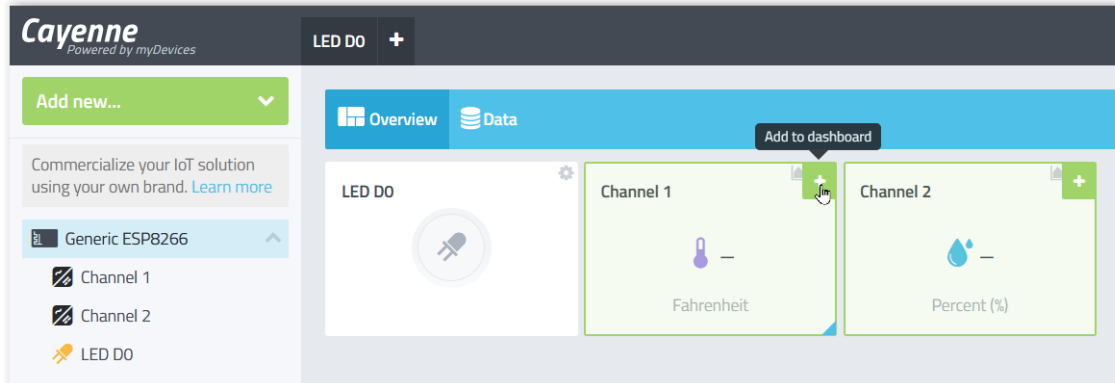
Wir müssen die Header-Dateien in den Sketch einbinden.

```
#include <CayenneMQTTESP8266.h>
#include <DHT.h>
```

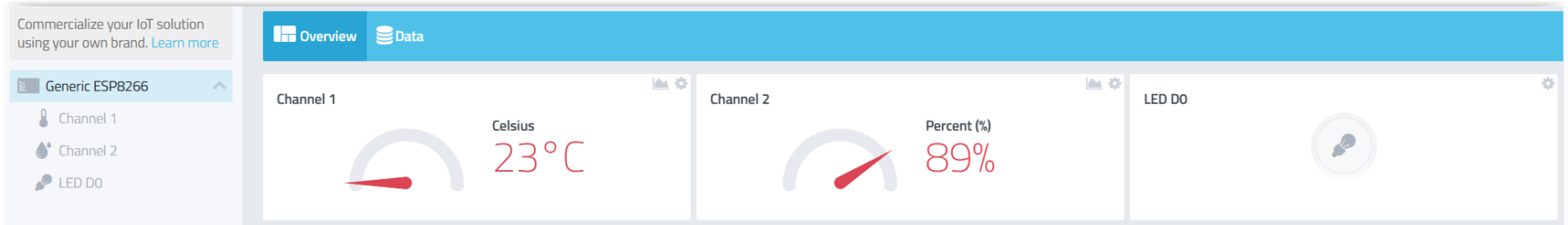
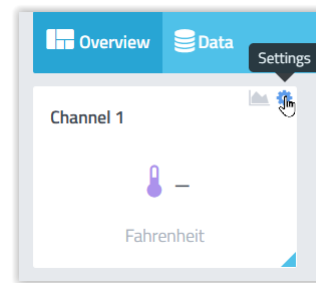


Sketch_MQTT_Cayenne_DHT11

Nach Upload des Sketches erscheint im Cayenne-Dashboard:



Wir können die Settings anpassen:



Anhang

IoT mit dem Funkstandard LoRaWAN

LoRaWAN steht für Long Range Wide Area Network und ist ein internationales Netzprotokoll zur drahtlosen Datenübertragung (wie auch WLAN und Bluetooth).

Es ermöglicht ein energieeffizientes (Sensoren können bis zu 10 Jahren ohne Batteriewechsel betrieben werden) Senden von Daten auch über lange Strecken.

Dies wurde speziell für das Internet of Things (IoT) entwickelt.



[LoraWAN Teil1 – Vorteile - Eigenschaften - Anwendungsmöglichkeiten – YouTube](#)

<https://www.youtube.com/watch?v=rmtEOu-2VN8>

Ende
Teil 6