

# Pflanzensensor Workshop

Willkommen beim Fabmobil Pflanzensensorworkshop! Dieses Dokument fasst die Inhalte und Aufgaben zusammen, die wir in den kommenden 2 Tagen gemeinsam bearbeiten werden und die du benötigst, damit du am Ende einen funktionierenden Pflanzensensor mit nach Hause nehmen kannst.

## Inhaltsverzeichnis

Ablauf.....	1
Tag 1.....	1
Tag 2.....	2
Aufgaben.....	3
Kennenlernen.....	3
Calliope-Programmierung.....	3
Einstieg.....	3
Bonus.....	3
Calliope Pflanzensensor.....	4
Kennenlernen der Arduino-IDE.....	5
Eine kleine Einführung in die C++ Programmierung.....	7
Beispiel 1: LED blinken lassen.....	7
Beispiel 2: Feuchtesensor auslesen.....	7
Bonus.....	8
Pflanzensensor.....	8
Zusammenbau.....	8
Bespielen des ESP8266 mit dem Pflanzensensorprogramm.....	9
Logdateien mitlesen.....	9
Webinterface aufrufen.....	9
Kennenlernen des Webinterfaces.....	14
Kalibrieren der Sensoren.....	14
Einstellen der Grenzen für die LED Ampel.....	15
Bonus.....	15
Wie weiter?.....	16
Verwendete Programme und Links.....	17
Calliope.....	17
Pflanzensensor.....	17
Fabmobil und Silicon Saxony.....	17

## Ablauf

### Tag 1

- Anmeldung, Kennenlernen, Bustour
- Calliope Programmierung:

- Kennenlernen und Programmierchallenges
- Pflanzensensor bauen, programmieren, kalibrieren, testen

## **Tag 2**

- Fertigstellen Calliope Pflanzensensor
- Arduino IDE kennenlernen
- Pflanzensensor zusammenbauen
- Pflanzensensor flashen
- Pflanzensensor kennenlernen und über Webinterface einstellen
- Testen
- Abschluss

# Aufgaben

## Kennenlernen

- macht euch ein Namensschild mit Kreppband: wenn ihr nicht fotografiert werden dürft nehmt ihr einen roten Stift, ansonsten nehmt ihr einen schwarzen Stift
- lasst euch den Bus zeigen und erklären
- meldet euch über das [Anmeldeformular](#) an

## Calliope-Programmierung

### Einstieg

Eure WorkshopleiterInnen werden euch eine Einführung ins Thema geben. Wenn ihr die bekommen habt findet euch in Zweiergruppen zusammen. Pro Gruppe benötigt ihr:


- Einen Laptop inkl. Maus und Netzteil
- Einen Calliope inklusive MicroUSB-Kabel
- Ein Calliope Lernkartenbuch

Eure Aufgabe ist es, dass Calliope Lernkartenbuch durchzuarbeiten. Es erklärt euch alles, was ihr über den Calliope, dessen Programmierung und die dazugehörige Entwicklungsumgebung wissen müsst. Am Ende gibt es ein paar kleine Programmierherausforderungen. Wenn ihr die geschafft habt, seid ihr gut vorbereitet für die Programmierung des Pflanzensensors.

### **Bonus**

Falls ihr früher als die anderen fertig seid, gibt es folgende Bonusaufgabe:

Sucht euch eine andere Gruppe die auch schon fertig ist. Ihr sollt ihr ein Schnick-Schnack-Schnuck Spiel programmieren, was mit zwei Calliopes gegeneinander gespielt werden kann. Hier ein paar Tipps:

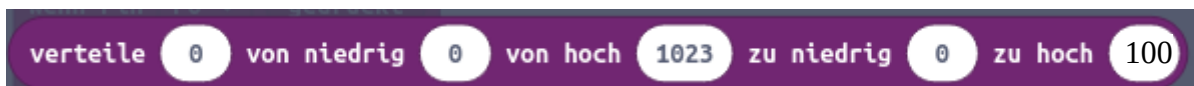
- Zum Senden und Empfangen der Daten zwischen den Calliopes benötigt ihr das Funk-Modul. Ihr findet es hinter dem  Erweiterungen Button im Online-Editor
- als erstes wählt ihr auf eurem Calliope (z.B.: A = Schere, B = Stein, A + B = Papier) und speichert diese Wahl in einer Variable
- dann schickt diese Wahl an den anderen Calliope

- parallel empfängt ihr die Wahl des anderen Calliope und speichert sie in einer anderen Variable
- diese zwei Variablen müssen dann miteinander verglichen werden: Stein schlägt Schere, Schere schlägt Papier usw.
- danach müsst ihr nur noch das Ergebnis ausgeben: habt ihr gewonnen oder verloren?

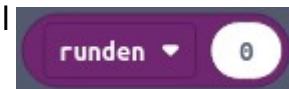
## Calliope Pflanzensensor

Meldet euch dann bei euren WorkshopleiterInnen. Die geben euch einen Bodenfeuchtesensor und die notwendigen Kabel, um sie mit dem Calliope zu verbinden. Außerdem wird euch gezeigt, wie ein funktionierendes Pflanzensensorprogramm auf dem Calliope aussehen kann. Ihr sollt jetzt auch ein solches Programm entwerfen. Nutzt dafür das Wissen, was ihr durch die Lernkarten erworben habt. Wenn ihr nicht weiter kommt, fragt die WorkshopleiterInnen um Hilfe. Hier sind auch noch ein paar Tipps und Tricks:

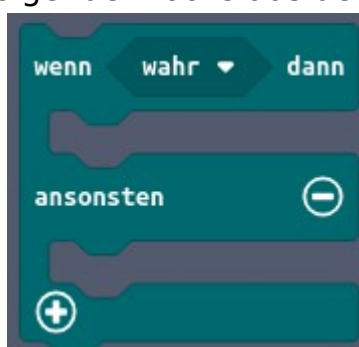
- in der „Beim Start“-Funktion wollt ihr alle eure Variablen und deren Werte definieren. Ihr braucht insgesamt 7: Bodenfeuchte, BodenfeuchteMin, BodenfeuchteMax, TemperaturMin, TemperaturMax, LichtstärkeMin, LichtstärkeMax
- in der „Dauerhaft“-Schleife wollt ihr den Bodenfeuchtesensor an Analogpin P1 auslesen und in einer Variable speichern. Der Sensor arbeitet invertiert: ganz feucht entspricht dem Messwert 1023, ganz trocken dem Messwert 0. Unter den Mathematikblöcken gibt es einen Block der es euch ermöglicht, die Messwerte auf eine 0 – trocken bis 100 – nass Skala umzulegen:

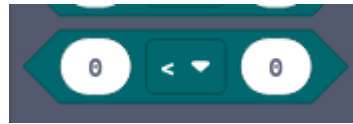


Da diese Funktion eine Kommazahl ausgibt, könnt ihr noch die Rundenfunktion verwenden um das Ergebnis auf eine ganze Zahl zu runden:



- Beim Drücken der Knöpfe A, B, A+B soll jeweils einer der Messwerte (Temperatur, Lichtstärke, Bodenfeuchte) angezeigt werden und die RGB-LED soll grün oder weiss sein, je nachdem ob der Messwert zwischen Minimum und Maximum liegt oder nicht. Für diesen Vergleich benötigt ihr folgende Blöcke aus dem Logik Bereich:





- Beim Drücken von P0 soll der Sensor alle Messwerte hintereinander anzeigen. Denkt euch jeweils ein schönes Symbol auf der LED-Matrix aus, damit klar ist, welcher Sensor gerade angezeigt wird

Wenn euer Calliope Pflanzensensor funktioniert, könnt ihr ihn mit feuchter und trockener Blumenerde kalibrieren und testen.

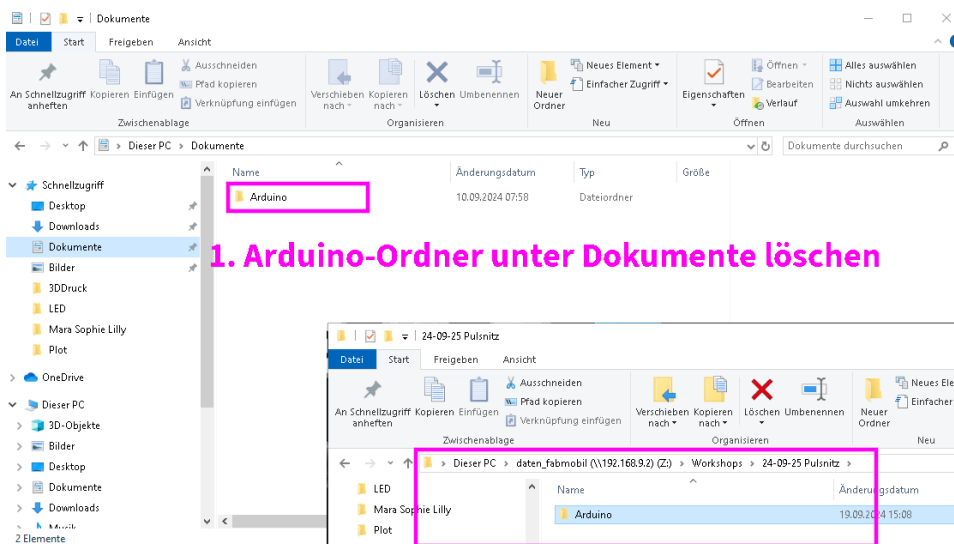
## Kennenlernen der Arduino-IDE

Nachdem ihr mit dem Calliope den Einstieg ins Programmieren gefunden habt, geht es mit der Arduino-IDE und der Programmiersprache C++ weiter.

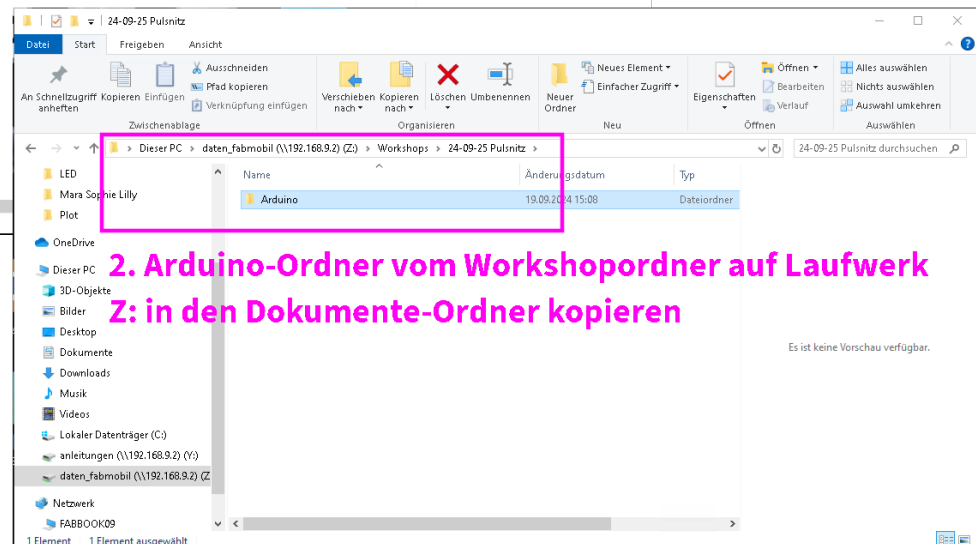
Ihr arbeitet weiterhin in Zweiergruppen und benötigt für diesesen Teil:

- euren Laptop
- einen ESP8266-Microcontroller
- ein MicroUSB-Kabel um den Microcontroller mit eurem PC zu verbinden
- die Software „Arduino IDE“ (sie ist schon auf eurem Laptop installiert)
- den Bodenfeuchtesensor, den ihr schon mit eurem Calliope verwendet habt

Als erstes kopiert ihr euch den Arduino-Ordner aus dem Workshopordner auf dem Netzlaufwerk in den Dokumente Ordner auf eurem Laptop. Dort sollte es schon einen Arduino Ordner geben, löscht ihn vorher damit ihr die aktuellen Versionen des Quellcodes habt:



1. Arduino-Ordner unter Dokumente löschen



2. Arduino-Ordner vom Workshopordner auf Laufwerk Z: in den Dokumente-Ordner kopieren

Dann öffnet ihr die Arduino IDE und schaut euch um. Hier ist ein Bild in dem die wichtigsten Bereiche markiert sind:



serieller Monitor

serieller Plotter

Quellcode kompilieren und hochladen

hier klicken um eine Liste aller Dateien zu sehen

## Eine kleine Einführung in die C++ Programmierung

Der ESP8266 wird mit der C++ Programmiersprache programmiert. Es gibt viele unterschiedliche Programmiersprachen, so wie es auch viele unterschiedliche gesprochene Sprachen gibt. Sie haben alle ihre Vor- und Nachteile. C++ kann gut mit Hardware wie Sensoren sprechen, deshalb wird sie hier benutzt. Sie ist schon ein wenig älter und schwer zu lesen, weshalb sie eher keine gute Sprache für den Einstieg ist. Wenn du dich für Programmieren interessierst empfehlen wir dir, dir mal Python anzuschauen.

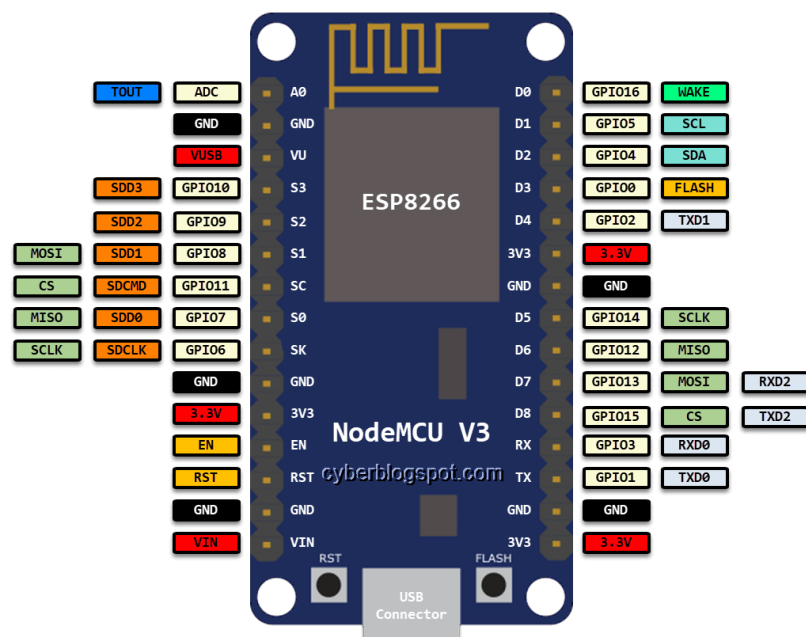
Nichts desto trotz:

### Beispiel 1: LED blinken lassen

- öffne `LED.ino` im LED Verzeichnis in der Arduino IDE
- verbinde den ESP8266 per USB Kabel mit deinem Laptop
- überspiele das Programm auf deinen ESP8266:
  - klicke unter Werkzeuge auf Board, wähle dann ESP8266 Boards und danach `LOLIN(WEMOS) D1 R2 & mini`
  - klicke unter Werkzeuge auf Port und wähle den COM-Port mit der höheren Nummer aus. (Falls mit diesem das Überspielen nicht funktioniert, kann es auch sein, dass es der andere Port der Richtige ist)
- überprüfe, ob die LED blinkt nachdem du das Programm überspielt hast
- schau dir das Programm an: du kannst verschiedene Sachen wiedererkennen:

### Beispiel 2: Feuchtesensor auslesen

- Schließe den Bodenfeuchtesensor mit Hilfe von Jumperkabeln an deinen ESP8266 an: rot kommt an 3V3 (Versorgungsspannung), schwarz an GND (Masse) und Gelb an A0 (Sensorwert). Die Pins sind beschriftet bzw. kannst du sie rechts ablesen.
- öffne `feuchtesensor.ino` in der Arduino IDE
- überspiele das Programm auf deinen ESP8266



- schau dir die Messwerte im seriellen Monitor und im seriellen Plotter an.
- Nimm den Sensor in die Hand und beobachte, wie sich der Wert verändert
- Der Wert liegt immer zwischen 0 und 1024

## Bonus

Wenn du früher als alle anderen fertig bist, kannst du dir ein Arduinobuch aus der Fabmobil Bibliothek nehmen und dir dort ein kleines Arduino-Programmierprojekt herausuchen. Sprich mit deinen WorkshopleiterInnen: fast alle der in den Büchern verwendeten Sensoren haben wir auch im Fabmobil und du kannst sie gern ausprobieren.

Die Arduino IDE hat auch viele Beispielprogramme die ausprobiert und verstanden werden wollen. Klick doch mal auf **Datei** und dann auf **Beispiele** und schau, was du da so findest!

## Pflanzensensor

Endlich ist es so weit und wir kümmern uns um euren ganz persönlichen Pflanzensensor! Du benötigst:

- einen ESP8266 Mikrocontroller
- einen Bodenfeuchtesensor
- ein Display
- ein Pflanzensensor PCB
- einen Laptop
- ein MicroUSB-Kabel
- Abstandshalter und Schrauben aus Plastik
- 3D-gedruckte Hülle für den Bodenfeuchtsensor

## Zusammenbau

Baue als erstes deinen Pflanzensensor zusammen:

- der ESP8266 kommt in die Fassung auf dem PCB. **Achtung:** der USB-Port muss nach unten zeigen!
- Das Display kommt in die Fassung über dem ESP8266. Fixiert es mit Plastikabstandshaltern und Schrauben: es gibt entsprechende Löcher auf dem PCB und im Display



- steckt den Bodenfeuchtesensor an: er kommt an den Port der mit X6 gelabelt ist. Passt auf die Orientierung des Kabels auf: - am PCB muss an GND am Sensor
- Setzt das Gehäuse um den oberen Teil des Bodenfeuchtesensors. Es schützt die elektronischen Bauteile davor, dass sie beim Blumengießen Spritzwasser abbekommen und dadurch eventuell kaputt gehen.
  - Falls das Gehäuse nicht hält könnt ihr es mit Klebeband oder Sekundenkleber zusammenkleben.

## Bespielen des ESP8266 mit dem Pflanzensensorprogramm

Es wird einen oder zwei Arbeitsplätze geben, an denen euer ESP8266 mit dem Fabmobil Pflanzensensor Programm bespielt wird. Geht dort hin und lasst euren Pflanzensensor aufspielen.

## Logdateien mitlesen

Du kannst die Logdateien deines Pflanzensensors mit deinem Laptop mitlesen. Dazu schließt du ihn per USB an deinen Laptop an, öffnest die Arduino IDE und dort den seriellen Monitor:

```

Serieller Monitor x Ausgabe
Nachricht (drücke Enter zum Senden für 'LOLIN(WEMOS) D1 R2 & mini' auf '/dev/ttyUSB0')

21:52:52.246 -> 1297s :I: [SensorP] Analog min/max aktualisiert für ANALOG0. Min: 13, Max: 883, Inverted: 0, Bytes geschrieben: 2355
21:52:54.043 -> 1299s .D: [Multiplexer] Wechsle von Kanal 1 zu 2 (Binär: 110)
21:52:54.075 -> 1299s .D: [Multiplexer] Erfolgreich auf Kanal 2 umgeschaltet nach 1299771ms
21:52:56.065 -> 1301s .D: [MeasurementCycle] Analog Sensor: Wechsel in Verarbeitungszustand
21:52:57.061 -> 1302s .D: [MeasurementCycle] Verarbeite: Feldnamen=8, Einheiten=8, Werte=2, currentResults=2
21:52:57.061 -> 1302s :I: [MeasurementCycle] Analog Sensor Messungen: lichtstaerke=7.47% bodenfeuchte=22.84%
21:53:09.166 -> 1314s .D: [MeasurementCycle] DHT: Wechsel in Verarbeitungszustand
21:53:10.130 -> 1315s .D: [MeasurementCycle] Verarbeite: Feldnamen=8, Einheiten=8, Werte=2, currentResults=2
21:53:10.130 -> 1315s :I: [MeasurementCycle] DHT Messungen: air_temperature=26.00°C air_humidity=45.00%
21:53:15.167 -> 1320s .D: [Multiplexer] Wechsle von Kanal 2 zu 1 (Binär: 111)
21:53:15.167 -> 1320s .D: [Multiplexer] Erfolgreich auf Kanal 1 umgeschaltet nach 1320877ms
21:53:17.187 -> 1322s .D: [Multiplexer] Wechsle von Kanal 1 zu 2 (Binär: 110)
21:53:17.187 -> 1322s .D: [Multiplexer] Erfolgreich auf Kanal 2 umgeschaltet nach 1322898ms
21:53:19.176 -> 1324s .D: [MeasurementCycle] Analog Sensor: Wechsel in Verarbeitungszustand
21:53:20.203 -> 1325s .D: [MeasurementCycle] Verarbeite: Feldnamen=8, Einheiten=8, Werte=2, currentResults=2
21:53:20.203 -> 1325s :I: [MeasurementCycle] Analog Sensor Messungen: lichtstaerke=7.47% bodenfeuchte=22.91%
21:53:20.267 -> 1325s .D: [Memory] Speicher [loop_monitor] Heap:12776/81920 Block:10112 Stack:2664/3332 Frag:20%
21:53:20.267 -> 1325s .D: [main] AP-Modus aktiv, überspringe erneute WiFi-Verbindungsversuche
  
```

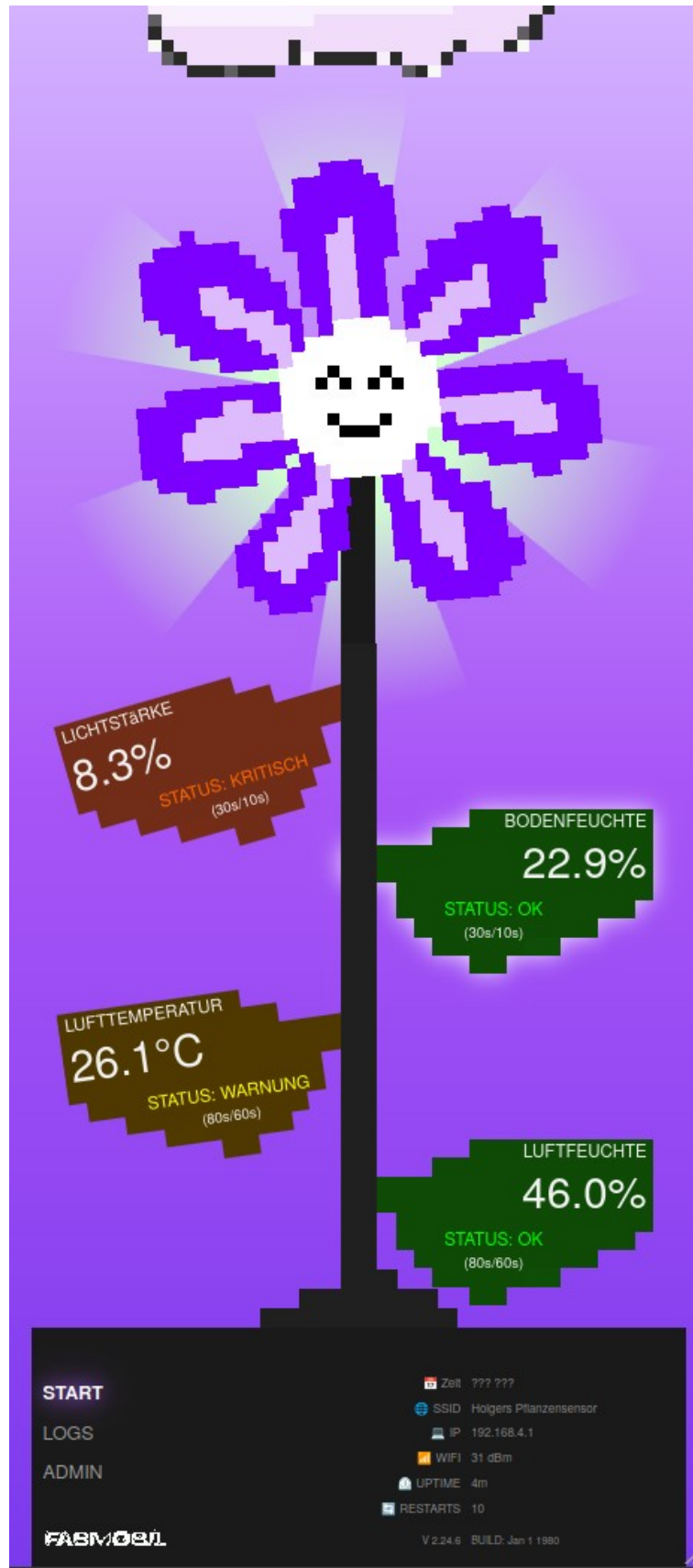
as

## Webinterface aufrufen

Finde die IP Adresse deines Pflanzensensors heraus. Er sollte mit dem Fabmobil WLAN verbunden sein und auf seinem Display seine IP Adresse anzeigen. Auch im Log wird sie regelmäßig angezeigt. Du musst sie in die Adressleiste deines Browsers einfügen:



Dann sollte sich die Pflanzensensorseite öffnen, sie sieht in etwa so aus:



Schau dich um. Deine Pflanze zeigt die Werte der Sensoren an. Da diese noch nicht kalibriert sind, kann es sein das deine Pflanze traurig ist. Kein Problem, wir ändern das gleich!

Unten links gibt es eine Navigationsleiste. Du kannst da auf ADMIN klicken und dann rechts auf Einstellungen. Dann wirst du nach einem Benutzernamen und einem Passwort gefragt. Der Benutzername ist admin und das Passwort ist Fabmobil. Nachdem du das eingegeben hast siehst du die Einstellungsseite.

Dort kannst du:

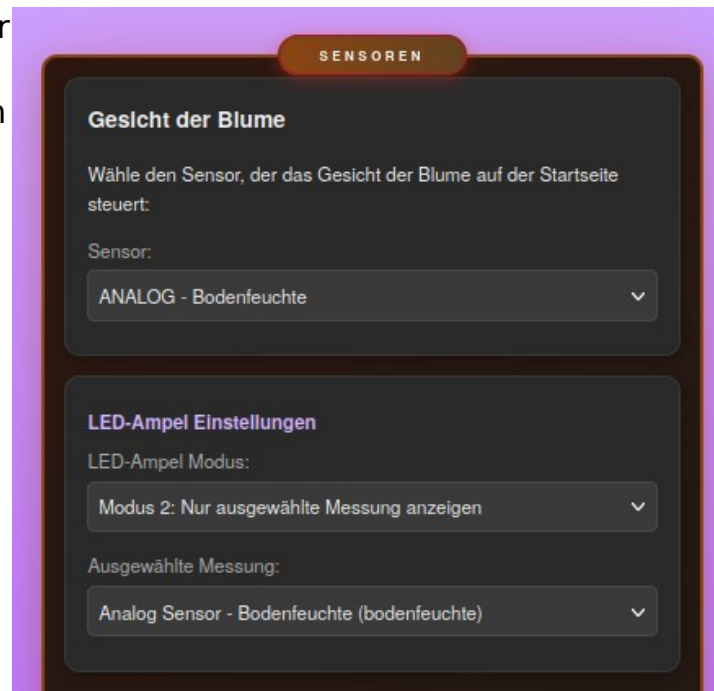
- unter **Systemeinstellungen** den Namen des Geräts ändern und das Adminpasswort ändern
- unter **Systemaktionen** die Einstellungen zurücksetzen, dass Gerät neu starten und deine Sensor- sowie Geräteeinstellungen herunterladen und auch wieder aufspielen
- unter **WiFi Einstellungen** kannst du deine WLAN Daten von zu Hause oder die deines Handyhotspots eingeben, damit sich der Pflanzensensor später damit verbinden kann.
  - **Achtung:** Der ESP8266 den wir verwenden kann kein 5G WLAN.
- In den **Log Einstellungen** ganz unten kann eingestellt werden, wie viele Informationen im Log angezeigt werden.

The screenshot displays the ADMIN interface with a dark theme. At the top, a red 'ADMIN' button is visible. The main content is divided into three sections:

- Systemeinstellungen**: Contains fields for 'Gerätename:' (set to 'Holgers Pflanzensensor'), a checkbox for 'MD5-Überprüfung für Updates aktivieren', 'Administrator Passwort:' (with a 'Neues Admin-Passwort' field), and a 'Passwort wiederholen:' field. A purple 'Passwort speichern' button is at the bottom.
- Systemaktionen**: Features four buttons: 'Einstellungen zurücksetzen' (red), 'Neustart durchführen' (orange), 'Einstellungen herunterladen' (purple), and 'Sensordaten herunterladen' (purple). Below these is a file upload section with a 'Browse...' button, the text 'No file selected.', and a label 'Einstellungen oder Sensordaten hochladen'.
- WiFi Einstellungen**: Includes three sets of fields for 'SSID' and 'Passwort', labeled 'SSID 1:', 'Passwort 1:', 'SSID 2:', 'Passwort 2:', 'SSID 3:', and 'Passwort 3:'. A purple 'WiFi Einstellungen speichern' button is at the bottom.

Suche jetzt nach der Sensor Adminseite. Du findest sie auch unten im Navigationserdhügel. Dort kannst du deine Sensoren konfigurieren:

- Das Gesicht der Blume reagiert immer auf einen Sensor. Ganz oben kannst du einstellen, welcher Sensor das sein soll
- Bei den LED-Ampel-Einstellungen kannst du drei unterschiedliche Einstellungen für die LED Ampel tätigen. Sie sind hoffentlich selbsterklärend.



## Kalibrieren der Sensoren

Damit euer Bodenfeuchtesensor richtig funktioniert, müsst ihr ihn jetzt kalibrieren:

- Stecke den Bodenfeuchtesensor in sehr feuchte Erde.  
**Achtung:** Du solltest den Sensor immer gleich tief in die Erde stecken damit die Messung einigermaßen genau ist. Ein Sensor, der nur halb in der Erde steckt, wird dir andere Messwerte anzeigen als einer, der ganz in der Erde steckt. Ein weißer Strich auf dem Sensor kurz unterhalb des Gehäuses markiert die Stelle, bis zu der ihr den Sensor in die Erde stecken solltet.
- Geh auf die Adminseite deines Pflanzensensors, finde dort den Bodenfeuchte-Bereich und lese den aktueller absoluter Messwert ab. Dieser muss nun darunter bei Maximalwert eingetragen werden.
- Stecke den Bodenfeuchtesensor in sehr trockene Erde.
- Lies wieder den aktuellen absoluten Messwert ab und trage ihn diesmal bei Minimalwert ein.
- Speichere deine Änderungen.
- Jetzt ist dein Bodenfeuchtesensor kalibriert. Falls du allerdings später zu Hause andere Erde hast kann es sein, dass du die Kalibrierung dort noch einmal wiederholen musst.
- Wenn du willst, kannst du auch deinen Helligkeitssensor kalibrieren: bedecke ihn damit es ganz dunkel ist und halte ihn in die Mittagssonne. Die voreingestellten Werte passen aber ganz gut, so dass du das nicht zwingend machen musst.

## Einstellen der Grenzen für die LED Ampel

Die Grenzwerte der LED Ampel für die unterschiedlichen Sensoren können auf der Adminseite deines Pflanzensensors eingestellt werden:

Diese Schwellwerte sind folgendermaßen definiert:



Abhängig davon, was die Pflanze die du mit dem Sensor überwachen willst für Bedürfnisse hat, stellst du jetzt die Skale für die unterschiedlichen Sensoren ein.

## Update der Software

## Einstellungen sichern und wiederherstellen

## Logs im Browser anschauen

## Wie weiter?

Ihr bekommt ja jede\*r einen Pflanzensensor mit nach Hause. Im besten Fall wisst ihr nun, wie ihr ihn konfigurieren und einsetzen könnt.

Unter <https://www.github.com/fabmobil/pflanzensensor> findet ihr die Dokumentation und den Quellcode zu eurem Sensor. Dort werden auch Aktualisierungen des Pflanzensensor-Programms veröffentlicht. Falls ihr in ein paar Wochen vergessen habt, wir ihr das Pflanzensensor Programm flashen könnt, steht das dort auch geschrieben. Wusstet ihr, dass ihr noch mehr Bodenfeuchtesensoren anschließen könnt um mehrere Pflanzen gleichzeitig zu überwachen? Oder das [euer Pflanzensensor euch per Mail oder Telegram Bescheid sagen kann, wenn eure Pflanze euch braucht?](#)

Falls ihr Fragen oder Anregungen habt könnt ihr uns auch gern per Email kontaktieren: [pflanzensensor@fabmobil.org](mailto:pflanzensensor@fabmobil.org)

Vielleicht findet ihr den Pflanzensensor ja aber auch doof und wollt lieber ganz andere Sachen mit eurem WLAN-fähigen Mikroprozessor machen? Kein Problem: zieht ihn einfach aus dem PCB raus und widmet ihn um. Das Internet ist voll mit Tutorials und Anleitungen, wenn ihr Inspiration braucht könnt ihr zum Beispiel hier mal schmökern: <https://polluxlabs.net/esp8266-projekte/>

Viel Spaß ♥

# Verwendete Programme und Links












## Calliope

- Calliope Lernkarten aus dem Fabmobil
- [Calliope Online Editor](#)
- [Einführungs Tutorial auf Youtube](#)

## Pflanzensensor

- [Arduino IDE](#)
- [Pflanzensensor Quellcode und Erklärungen](#)
- Anleitungen im Pflanzensensor Wiki:
  - [Pflanzensensor konfigurieren und bespielen](#)
  - [Erklärungen zum Webinterface](#)
  - [Webhook für Telegram- oder Mailbenachrichtigungen einrichten](#)
  - [make.com Website für Mail- oder Telegrambenachrichtigungen](#)

## Fabmobil und Silicon Saxony

-  [Fabmobil Homepage](#) 
-  [Silicon Saxony](#) 
-  [Bosch](#) 
-  [Global Foundries](#) 
-  [Infineon](#) 
-  [Xfab](#) 