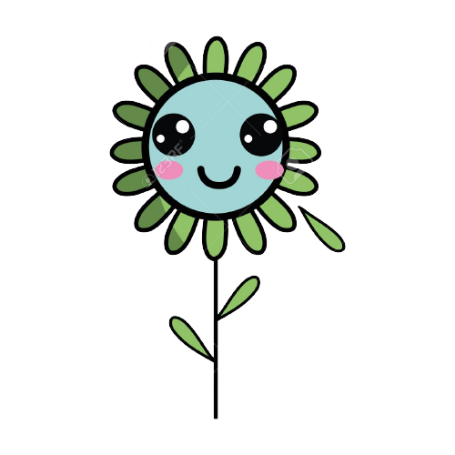
Projektdokumentation

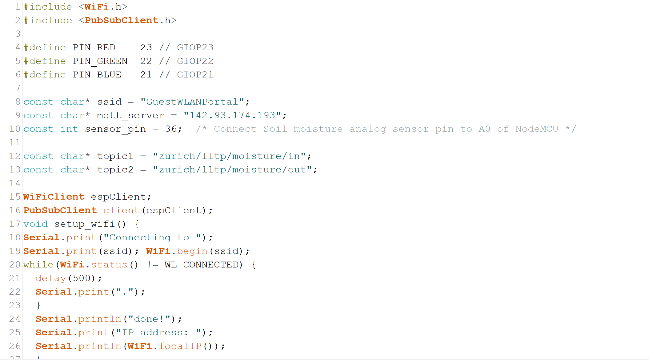
Überbetrieblicher Kurs

Modul 216

Gruppe 6







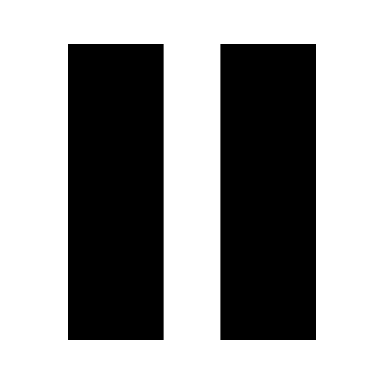


Abbildung 2

Abbildung 1

Inhaltsverzeichnis

[Einstieg 3](#_Toc88726648)

[Anforderungen 3](#_Toc88726649)

[Ressourcen 3](#_Toc88726650)

[Datenfluss 4](#_Toc88726651)

[Tests 4](#_Toc88726652)

[Ablauf 4](#_Toc88726653)

[Probleme 4](#_Toc88726654)

[Reflexion 4](#_Toc88726655)

[Quellen 4](#_Toc88726656)

# Einstieg

Viele Menschen besitzen Zuhause oder im Büro eine oder mehrere Pflanzen.

Dabei kommt es hin und wieder vor, dass das Bewässern vergessen wird und die Pflanze deshalb verwelkt.

In diesem Projektauftrag soll nun ein IoE Messgerät entwickelt werden, welches die Feuchtigkeit einer Pflanze (Topferde) misst und einem signalisiert, dass die Pflanze Wasser braucht.

# Anforderungen

Für das Projekt haben wir uns Anforderungen gesetzt, welche wir bis zur Abgabe erfüllt haben wollen.

* Umsetzung gemäss Bewertungsraster (siehe Dokument «Bewertung» in Microsoft Teams)
* Zeitlicher Verlauf
* Die Feuchtigkeit der Erde soll gemessen werden und bei vorbestimmten Werten soll ein LED-Lämpchen die Farbe wechseln.
* Werte der Pflanze Sichtbar machen (Node-RED)

# Optionale Anforderungen

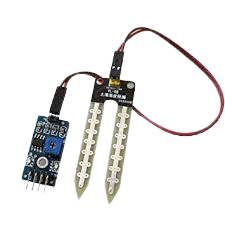
* (Optional bei genügend Zeit) Möglichkeit zur Benachrichtigung über ein Kommunikationskanal (Email, SMS…)
* (Optional bei genügend Zeit) Der Zustand der Pflanze soll über einen OLED Bildschirm veranschaulicht werden.

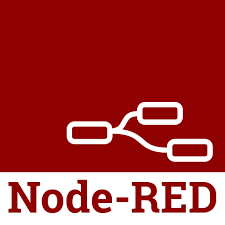
# Ressourcen

|  |  |
| --- | --- |
| Ressource: | Anzahl: |
| ESP 32 Mikrocontroller | 1x |
| Hygrometer Sensor | 1x |
| Jumper Kabel |  |
| RGB LED | 1x |
| Breadboard (klein) | 1x |
| HW 103 | 1x |
| Mikro USB-Kabel | 1x |

# 

# Datenfluss



Ein Bild, das Text, Elektronik, Schaltkreis enthält.

Automatisch generierte Beschreibung

# Drahtlos Silhouette

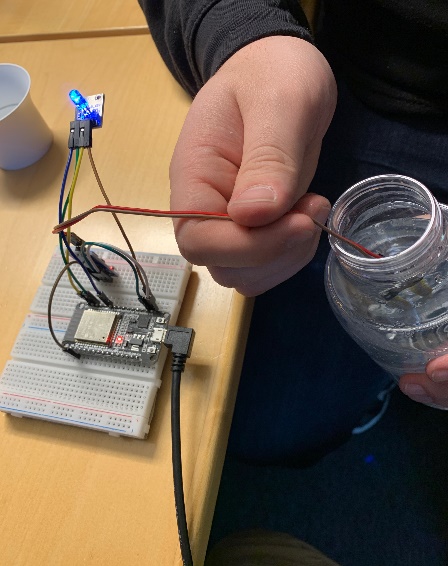
Sensordaten Sensordaten

# 

Status LED

# Realisierung

## Planung und Entscheidung

Vor unserer Realisierung haben wir mit dem Planen und Entscheiden angefangen. Als erstes haben wir uns also zusammengesetzt und entschieden, welches Thema wir als Projekt behandeln möchten. Die Entscheidung lag vor allem bei Mikka, da er schon im Vorhinein die Idee hatte, ein IoT Gerät zu programmieren, welches ihm den Zustand seiner Pflanzen nennt. Zusätzlich war es einer der Vorschläge, zu was wir ein Projekt führen sollten. Nun gingen wir zur Planungsphase über. Wir begannen damit zusammenzutragen, welche Ressourcen wir für unser Projekt benötigen. Auf jeden Fall bräuchten wir einen ESP 32 Mikrocontroller, ein Breadboard und einen Feuchtigkeitssensor. Wir haben uns auch noch vorgestellt, dass wir den Zustand der Pflanze direkt anzeigen lassen können durch ein LED-Lämpchen. Eventuell wollten wir auch noch bei genügend Zeit ein OLED Bildschirm einbinden, welcher durch Emojis den Zustand der Pflanze zusätzlich verdeutlicht. Auch bräuchten wir auf jeden Fall noch ein paar Jumper Kabel, um alles miteinander zu verkabeln.

## Grundaufbau

Nach dem wir die Ressourcen einmal aufgeschrieben und unseren Projektauftrag verdeutlich haben, sammelten wir die ersten Ressourcen zusammen und begannen damit, alles zu verkabeln was wir bisher benötigten. Erst verbanden wir den Sensor mit dem HW 103 Modul, welcher dann mit dem ESP 32 angeschlossen wurde. Bevor wir unsere LED verbinden, wollten wir erst ein Programmbeispiel finden und uns davon Ausschnitte für unser Programm holen. Um nun erste Daten zu bekommen, benutzten wir eine Wasserflasche, in welche wir den Sensor reinlegten.

Über den Serial Monitor bekamen wir nun einige Messdaten des Sensors. Allerdings waren diese Werte in einem negativen Bereich und sobald der Sensor das Wasser berührte, stieg der Negative Wert von -512, auf -304. Für uns war dieser Merkwürdige Wertebereich ein Dorn im Auge. Dafür haben wir uns erstmal angeschaut, welche Zahlen der Sensor ohne Berechnungen versendet. Ohne Berechnung im Programm, erhielten wir einen Konstanten Wert von 4095. Sobald nun der Sensor ins Wasser geriet, sank der Wert drastisch. Um nun logische Werte zu bekommen, benutzten wir eine Mathematische Formel, welche uns den Wert 4095 auf 0% setzte und sobald der Wert durch das Wasser sank, stieg unser Endwert auf über 60%. Somit hatten wir nun eine Logische Skala, 0% ist trocken und 60%+ ist nass.

## LED

Mit den Werten konnten wir nun unsere LED einstellen. Dafür verbanden wir diese mit dem ESP 32 über das Breadboard. Nun schrieben wir im Programm das, sobald der Sensor einen Gewissen Prozent Wert hat, sich die Farbe der LED ändert. Ein Bild, das Text, Elektronik enthält.

Automatisch generierte BeschreibungSchlussendlich leuchtet unsere LED rot für Trocken, bis einem Feuchtigkeitswert von 5%.

Ein Bild, das Text, Elektronik enthält.

Automatisch generierte BeschreibungAb 5% bis 55% leuchtet die LED grün, wobei sich mit 4 Stufen, das Blau intensiviert.

Ab einem Feuchtigkeitsprozent von über 55% wird die LED blau.

Ein Bild, das Text, Elektronik enthält.

Automatisch generierte Beschreibung

Nachdem die LED eingestellt war, fingen wir an, Node RED zu verbinden. Unsere Node RED Instanz war anfangs sehr einfach gehalten. Wir haben unsere Werte simpel in einem Graph wieder. Um nun einige Punkte bei der Evaluation zu integrieren, haben wir für unsere MQTT Verbindung ein Passwort gesetzt. Auch schicken wir nun dem Sensor einen Wert zurück, welcher nun die LED auf dem HW 103 aufblinken lässt, sobald der ESP 32 kein WLAN hat oder die Verbindung sucht.

## Switch

Unsere nächste Idee war, den Sensor über Nose RED deaktivieren zu können. Dafür haben wir in unserer Instanz einen Switch eingebaut welcher nun einen true/false Wert zurückgibt. Dieser Wert soll nun die loop Funktion steuern. Solange unser Switch in Node RED auf true ist, läuft innerhalb der loop Funktion unser Programm zum Sensor und zum Senden der Daten.

# Tests

# Probleme

Sensor Rost

Node RED Port

Sensoren Wert

RGB LED

Keine Pflanze mit Erde

Callback Value in eine Funktion

# Reflexion

# Quellen

Abbildung 1:

<https://media.digikey.com/photos/Schtoeta-Engineering/MFG_ESP32-DEVKITC-32E.jpg>

Abbildung 2: <https://previews.123rf.com/images/stockgiu/stockgiu1706/stockgiu170600557/79584203-kawaii-gl%C3%BCckliche-blume-pflanze-mit-bl%C3%A4ttern-und-bl%C3%BCtenbl%C3%A4tter-vektor-illustration.jpg>

Abbildung 3: <https://i2.wp.com/www.esp8266learning.com/wp-content/uploads/2017/06/Soil-Moisture-Sensor.jpg?resize=355%2C355>

Abbildung 4: <https://encrypted-tbn0.gstatic.com/images?q=tbn:ANd9GcQWgVuppfVqS6ikt0a1vGdYon8IxsL377Le-Q&usqp=CAU>

Abbildung 5: