

Messen und Analysieren mit IoT und Data Science

Automatisierung und Überwachung einer Bewässerungsanlage für Indoor-Pflanzen

Projektvorfeld

Für die Bewässerung von Pflanzen in einem geschlossenen Raum gibt es nur die Möglichkeit des manuellen Gießens mithilfe einer Gießkanne, da die Pflanzen nicht durch den Regen bewässert werden können, wie es bei Pflanzen ohne Überdachung der Fall ist. Zum Problem wird dies, wenn der Halter der Pflanzen keinen grünen Daumen besitzt und nicht weiß, wann die Pflanzen zu gießen sind. Pflanzen sterben oft schon nach kurzer Zeit ab und die Entsorgung und die Neubeschaffung neuer Pflanzen erfordert wiederum Geld und Zeit. Herkömmliche Systeme zur automatischen Bewässerung sind meist teuer für den Funktionsumfang, den sie bieten. Als Beispiel ist das Micro Drip System von Gardena zu nennen, wo nur die Möglichkeit besteht, die Pflanzen in gewissen Zeitintervallen zu gießen.

Projektziele

Mit dem Projekt "Automatisierung und Überwachung einer Bewässerungsanlage für Indoor-Pflanzen" soll, wie es der Name bereits sagt, eine Bewässerungsanlage für solche Pflanzen in einem Raum erstellt werden. Dabei soll sie von sich selbst mit möglichst wenig Eingriffen einer Person arbeiten und dabei Statistiken über die zu überwachenden Pflanzen sammeln.

Mit dem Eingriff ist gemeint, dass die Pumpen nicht an ein Wassersystem angebunden sind, sondern in einem mit Wasser gefüllten Behälter liegen, weshalb dieser vom Halter selbst wieder befüllt werden muss, wenn der Wasserstand zu gering für die Betreibung der Pumpen ist.

Die Pumpen selbst sollen über einen ESP32 betrieben werden, indem bei der Stromversorgung für die Pumpen ein Relay zwischengeschaltet wird, der wiederum durch den ESP32 gesteuert werden kann.

Für die Statistiken sollen Informationen über die Feuchtigkeit der Pflanzen, die begossen werden, gesammelt werden. Dies wird erreicht durch Feuchtigkeitssensoren, die in der Erde des Topfes der jeweiligen Pflanze stecken. Anhand der Feuchtigkeit der Erde soll entschieden werden, ob die Pflanze gegossen werden muss oder nicht. Bei einer zu geringen Feuchtigkeit sollen die Pumpen eingeschaltet werden.

Darüber hinaus soll die Raumtemperatur gemessen werden, um diese auch in den für den Nutzer bereitgestellten Statistiken anzuzeigen. Da man bei zu hoher Temperatur nicht gießen sollte, soll die Temperatur auch in den Entscheidungsprozess, ob gegossen werden soll oder nicht, miteinfließen.

Auch die Lichteinstrahlung soll gemessen und wie die Raumtemperatur wiedergegeben werden. Auch wie bei einer zu hohen Temperatur soll bei zu starkem Licht nicht gegossen werden, weshalb auch hier die Funktion des Gießens deaktiviert werden soll, wenn eine gewisse Helligkeit überschritten wird.

Zuletzt wird noch der pH-Wert des Wassers aus dem Behälter gemessen. Dies dient nur zur Information für den Nutzer, damit dieser eine für die Pflanzen gesunde pH-Wert des Wassers überwachen kann. Eine Grenze ist hier schwerer zu pauschalisieren, da die benötigten Werte hier stärker von Pflanze zu Pflanze abweichen als bei Temperatur und Helligkeit.

Darüber hinaus sollen noch Knöpfe installiert werden, mit denen man die Bewässerung der Pflanzen manuell ansteuern kann.

Aufbau

Aufgeteilt werden alle Sensoren auf zwei ESP32, da dies für die Benutzung praktischer ist und gefordert wurde, dass die zwei Mikrocontroller kommunizieren sollen. Außerdem können so die Sensoren aufgrund des Nachteils durch die Kabel besser positioniert werden und so lassen sich bessere Messergebnisse für den Lichtsensor bestimmen.

Aufgeteilt werden die Funktionalitäten wie folgt:

- ein Mikrocontroller kümmert sich um die Steuerung der Pumpen und um die Messung des pH-Werts, da die Kabel alle in den Wasserbehälter führen. Außerdem sind noch zwei Knöpfe an den ESP32 angeschlossen, mit denen man die Pumpen manuell starten kann.
- der andere Mikrocontroller kümmert sich um alle restlichen Messgeräte. Dazu gehört der BME280, der die Funktion der Temperaturmessung übernimmt, ein Photoresistor, der die Lichthelligkeit bestimmt und Feuchtigkeitssensoren für die Erde, die im Topf der jeweiligen Pflanze sind.

Die wasserfesten Brushless DC Pumps (AD20P-1230E) laufen mit 12V, welches durch

entsprechende
Netzteile
bereitgestellt wird.
Die Pumpen liegen
im Wasser des
Behälters und
pumpen bei
Aktivierung Wasser
in die 4,6mm
Schläuche, die zu
den Pflanzen führen.
In unserem Projekt
sind es zwei
Pflanzen, die
automatisch
begossen werden.



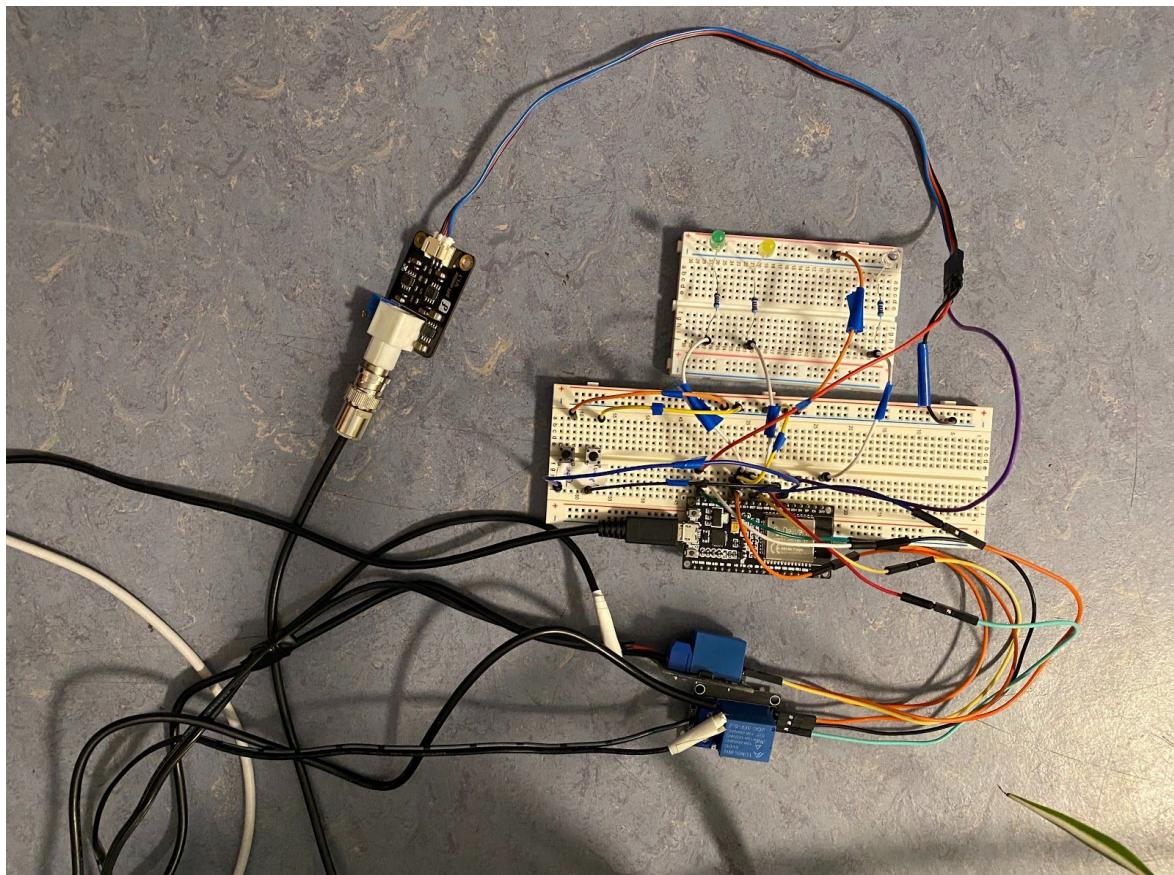
Die Schläuche selbst sind mit T-Stücken verseht, aus denen das Wasser austreten kann. Durch die Anbindung des Ende des Schlauchs an ein T-Stück wird der Endstopfen umgangen. Somit hat man aus jedem T-Stück beinahe dieselbe Menge an Wasser, die austritt. Für die andere Pflanze sind ausreichend T-Stücke im Schlauch vorhanden, dass aus allen ohne eine Endstopfen dieselbe Menge Wasser austritt.



Die Schläuche werden in den Töpfen so positioniert, dass die T-Stücke, aus denen das Wasser fließt, auf die Pflanzen zeigen und diese somit gegossen werden.



Wie bei den Projektzielen bereits erwähnt, wird der zum ESP32 externe Stromkreislauf der Pumpen, wessen Netzteile dauerhaft in der Steckdose sind, durch Relays geschlossen oder geöffnet. Diese Relays wiederum werden von einem Microcontroller gesteuert. Die Relays (blaue Kästen) kann man im nachfolgenden Bild unten erkennen. Diese sind an den 5V Output des ESP32, dem Ground und einem Steuerpin gebunden, was die Steuerung durch Software ermöglicht. Außerdem werden 2 Knöpfe bereitgestellt, die an den ESP32 angeschlossen sind und über die man die zwei Pumpen manuell steuern kann durch gedrückthalten eines Knopfes. Zu sehen sind diese auf der linken Seite des Steckbretts im Bild.



Die in diesem Bild nicht leuchtenden Kontrollämpchen in den Farben weiß (erscheint auf den Bildern blau), grün und gelb werden später im Abschnitt Funktionen noch erklärt.

Der andere ESP32 ist lediglich dazu da, Informationen der Sensoren bereitzustellen, die dann vom andern Microcontroller und Node-Red aufgenommen werden.
Um Messdaten für die Temperatur zu erhalten, wird ein BME280 angeschlossen. Diese Information wird genauso wie alle nachfolgenden alle 2 Sekunden zum MQTT geschickt. Die Feuchtigkeit erhält man durch Feuchtigkeitssensoren, die in der Erde der Topfpflanzen stecken. Hierbei ist zu erwähnen, dass in unserem Aufbau die Pflanze mit der Bezeichnung "Plant 1" 2 Feuchtigkeitssensoren besitzt, wohingegen die Pflanze mit der Bezeichnung "Plant 2" nur einen Sensor hat. Dies erkennt man später im Abschnitt Funktionen auch an dem Screenshot aus Grafana, wo die zu der Feuchtigkeit zu Plant 1 zwei Kurven besitzt.
Zwei Sensoren hat die Pflanze 1 weil der Topf länglicher ist und es aufgrund

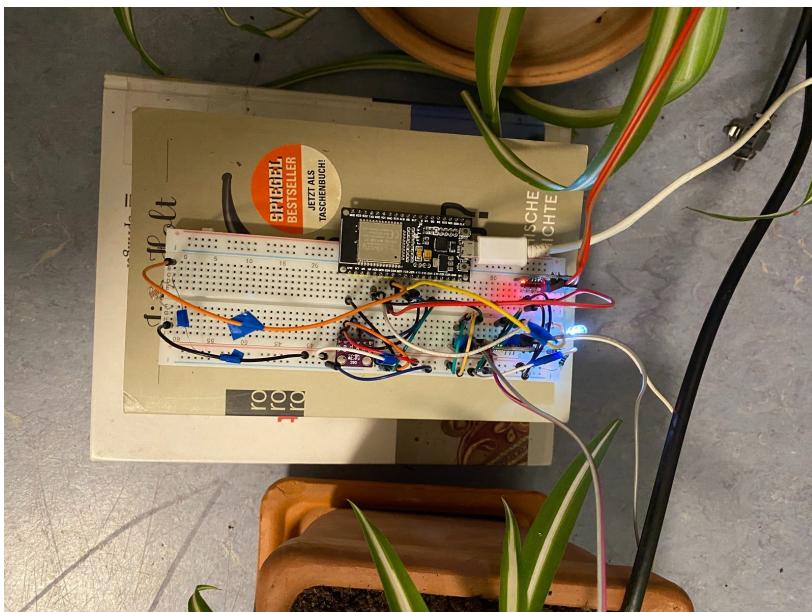
unterschiedlicher
Sonneneinstrahlungen dazu
kommen kann, dass eine Seite des
Topfes trockener ist als die andere.
Dies kann man auch am



nebenstehenden Bild erkennen, wo man die Position der Sensoren vor allem an den Kabeln erkennt, die zu ihnen führen.

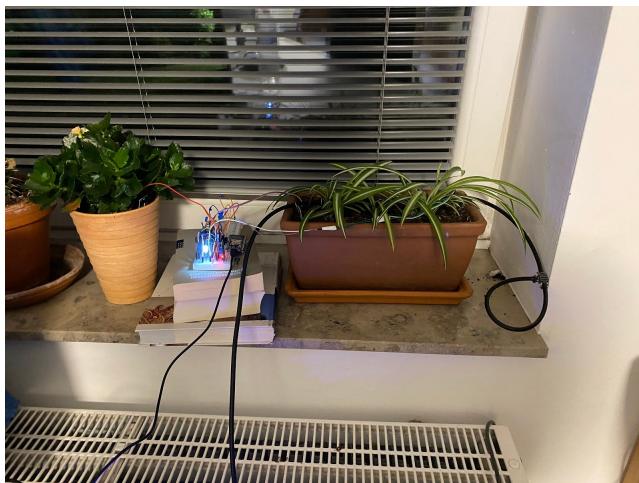
Zuletzt wird noch die Lichteinstrahlung mit einem Photoresistor gemessen. Dieser wird mit einem 5100Ω Widerstand gemessen und genauso wie die anderen Daten im MQTT unter einem passenden Topic veröffentlicht.

Der Aufbau sieht dann wie folgt aus:



Den fertigen Aufbau im Einsatz auf einer Fensterbank sieht man auf dem nächsten Bild. Dabei ist der Microcontroller, der für die Pumpen und zur Überprüfung des pH-Wertes des Wassers zuständig ist, weiter unten positioniert und für den Nutzer dann nicht sichtbar, da

dieser auch keine Sensoren hat, die eine gewisse Positionierung benötigen, wie der ESP32, der für die Messung der Lichthelligkeit zuständig ist. Bei diesem ESP32 ist es nämlich von Vorteil, den Lichtsensor so nah wie möglich an die Pflanzen zu stellen, um ein möglichst sinnvolles Ergebnis zu erhalten.

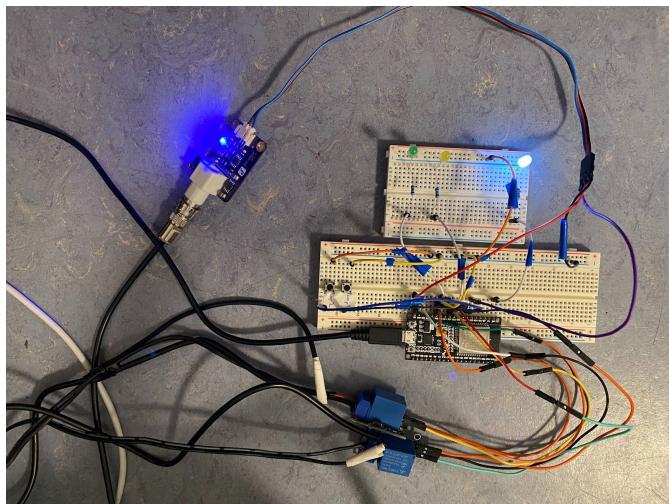


Funktionen

Alle Funktionen in unserem Code können unter dem nachfolgenden Link im Repository eingesehen werden: <https://github.com/jangabow/Automatische-Bewässerungsanlage>

Die gesammelten Informationen von einem Microcontroller über die Temperatur, die Feuchtigkeit der Erde und die Lichthelligkeit werden zusammen unter verschiedenen Topics an einen MQTT-Server gesendet, welcher die Daten abfängt und an alle Subscriber weitersendet. Dies geschieht genauso mit den Werten über den pH-Wert des Wassers, in dem sich die Pumpen befinden. Der Pumpen-Microcontroller fängt die Daten ab und bestimmt dann durch eine im Code definierte Vorgehensweise, wann die Pumpen einzuschalten sind und somit auch wann die Pflanzen gegossen werden sollen.

Der Normalzustand des Microcontrollers, welcher die Pumpen steuert und die LED's steuert sieht wie folgt aus:



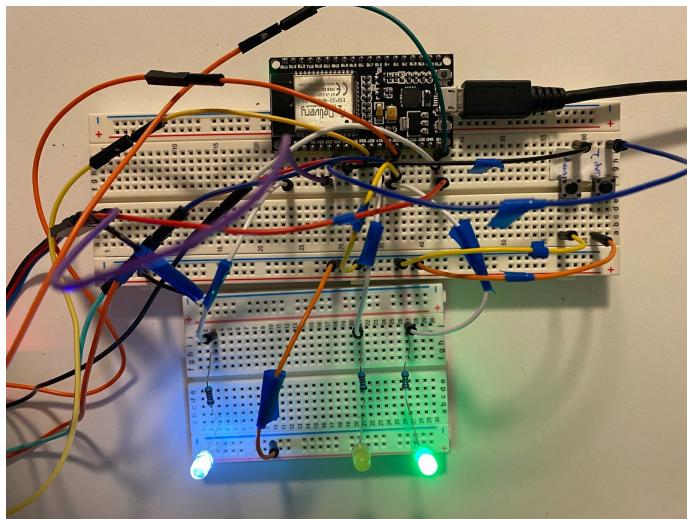
Normalzustand: weiße LED leuchtet dauerhaft.

Die Bewässerung wird gestartet, wenn ein bestimmter Grenzwert der Feuchtigkeit unterschritten wird. In unserem Fall liegt der Grenzwert bei 30% Feuchtigkeit, fällt die Bodenfeuchtigkeit also unter 30%, so wird die Bewässerung gestartet. Hierbei wird jedoch nicht einfach so lange gewässert bis die Feuchtigkeitssensoren wieder über dem Grenzwert liegen, sondern es wird mit einer Zeitsteuerung sichergestellt, dass die Pflanzen nicht übergossen werden. Denn bis die Feuchtigkeitssensoren das gegossene Wasser und die somit zunehmende Feuchtigkeit erkennen, kann einige Zeit vergehen, da sich die Feuchtigkeitssensoren ca. 10cm in der Erde befinden. Damit nun also die Pflanzen nicht übergossen werden sieht die Ablauffolge, wenn ein Feuchtigkeitssensor den Grenzwert unterschreitet, wie folgt aus:

Wenn bei der Pflanze 2 (hat nur einen Feuchtigkeitssensor) der Grenzwert unterschritten wird, wird für 5 Sekunden Wasser zur Pflanze gepumpt. Zeitgleich leuchtet für die Zeit der Bewässerung die grüne LED auf. Nach den 5 Sekunden Bewässerung sorgt ein Timer dafür, dass in den nächsten 55 Sekunden keine Bewässerung an dieser Pflanze stattfinden darf. Dies liefert dem Sensor genug Zeit um die neuen Werte, die durch die Bewässerung entstanden sind genau zu messen und zurückzugeben. Ist nach dieser Zeit die Erde immer noch zu trocken, so wird die Bewässerung wieder freigegeben und der Prozess wiederholt sich.

Bei Pflanze 1 (hat zwei Feuchtigkeitssensoren) läuft dieser Prozess ähnlich ab, jedoch wird hier erst bewässert, wenn beide Feuchtigkeitssensoren den Grenzwert unterschreiten. Danach läuft der Prozess genau wie bei Pflanze 2 ab, zuerst wird 5 Sekunden lang

gewässert, zeitgleich leuchtet die grüne LED auf, dann wird 55 Sekunden gewartet und bei Bedarf der Prozess wiederholt.



Auf dem Bild sieht man, dass die grüne LED bei einer Bewässerung aufleuchtet.

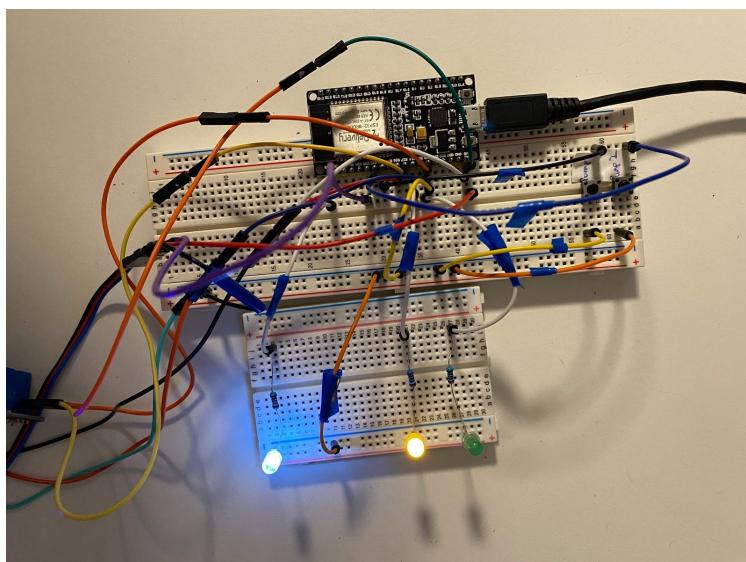
Erklärung der LED's:

weiße LED: Diese LED leuchtet auf, sobald der ESP32 mit Strom versorgt wird und im Betrieb ist

grüne LED: Diese leuchtet auf, solange eine Bewässerung stattfindet.

orangene LED: Diese LED leuchtet immer dann, wenn bestimmte Grenzwerte überschritten wurden. Also leuchtet sie, wenn die Temperatur über 27°C liegt oder wenn die Lichteinstrahlung über 70% liegt.

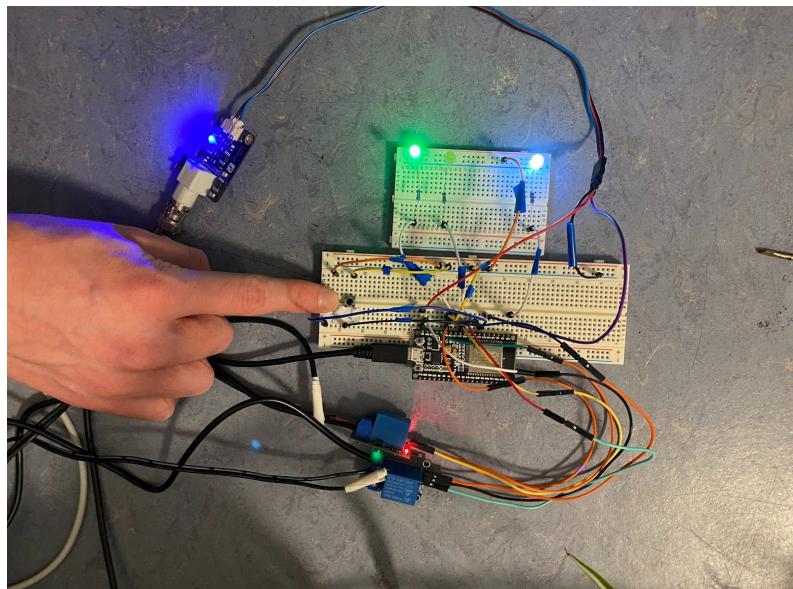
Sobald die Orange LED aufleuchtet, wird die Bewässerung blockiert. Denn es sollte nicht gewässert werden, wenn die Temperatur oder die Lichteinstrahlung zu hoch ist. Wenn die Temperatur zu hoch ist kann das Wasser zu schnell verdunsten, wodurch sich eine effiziente und wassersparende Bewässerung nicht realisieren lässt. Ist die Lichteinstrahlung zu hoch sollte man auch auf eine Bewässerung verzichten, da Wasser auf der Pflanzenoberfläche wie eine Lupe wirken kann und somit kann es passieren, dass die Pflanzen dadurch verbrennen. Nachdem die orangene LED erloschen ist, ist die Bewässerung wieder freigegeben und es kann bei Bedarf wieder gegossen werden.



Orangene LED leuchtet auf, dass heißt die Temperatur oder die Lichteinstrahlung ist für eine effiziente Bewässerung zu hoch.

Manueller Modus:

Alternativ lässt sich die Bewässerung auch mittels zweier integrierter Buttons realisieren. Dabei aktiviert Button 1 die Bewässerung für Pflanze 1 und der Button 2 die Bewässerung für Pflanze 2. Damit kann auch die Bewässerungssperre aufgehoben werden, falls die Temperatur oder die Lichteinstrahlung zu hoch ist. Mit dem manuellen Modus kann man auch beliebig lange und oft wässern, denn solange man den Button drückt, solange wird auch bewässert. Zeitgleich leuchtet die grüne LED wieder auf und signalisiert den Bewässerungsvorgang. Lässt man den Button wieder los, so stoppt die Bewässerung und die grüne LED erlischt.



Beispiel für den manuellen Modus, Button 1 wird gedrückt und somit wird die Pflanze 1 bewässert. Dabei leuchtet die grüne LED für die Zeit der Bewässerung wieder auf.

Betrieb:

Die Funktionsfähigkeit des Bewässerungssystems wird in den folgenden Videos gezeigt. Dabei werden die Pumpen eingeschaltet und zu den T-Stücken im Schlauch gepumpt, die dann das Wasser auf die Erde unter den Pflanzen lassen:

<https://youtu.be/oYmoa6OvaYQ>

https://youtu.be/EnfSFBs5_00

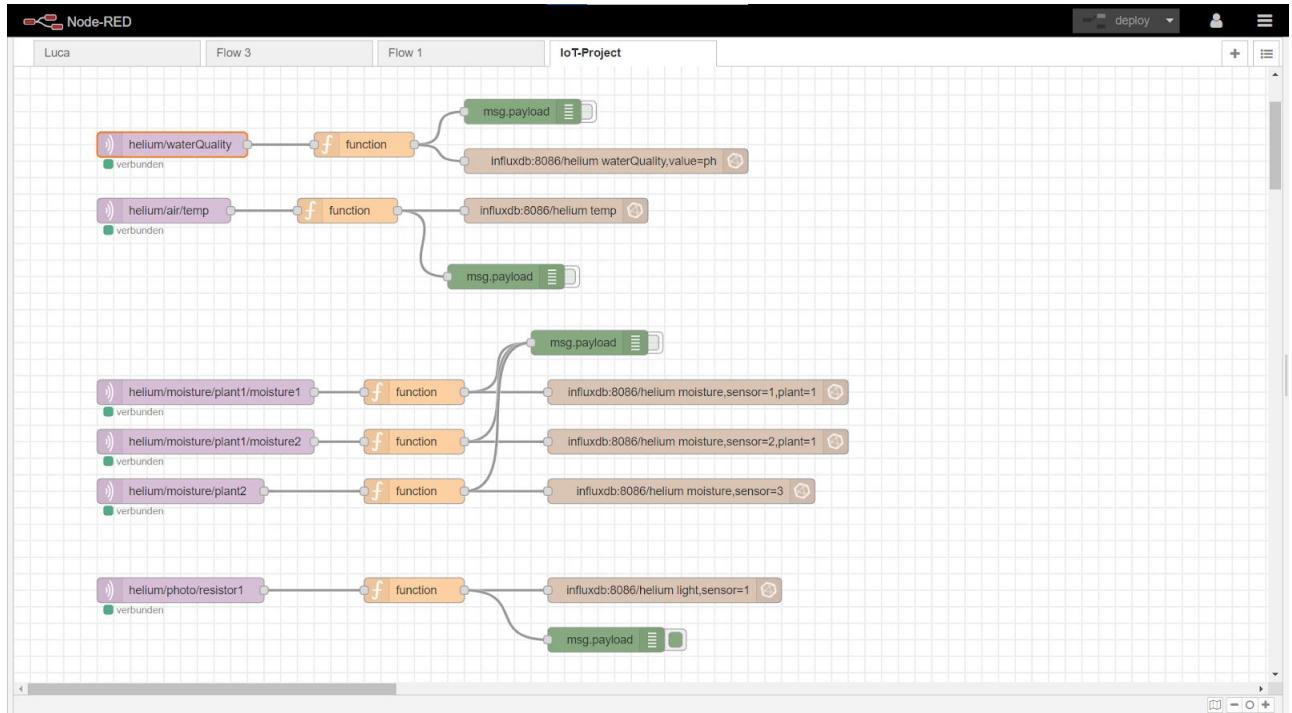
Im folgenden Video wird exemplarisch gezeigt, wie der Gießvorgang abläuft, wenn die Feuchtigkeit zu gering ist: <https://youtu.be/GYZrsNSS4cQ>

Hierbei mussten wir jedoch darauf zurückgreifen, dass wir die Feuchtigkeitssensoren aus der Erde ziehen und somit die Trockenheit simulieren, da eine lange Zeit vergehen kann, bis die Erde einer Topfpflanze eine ausreichende Trockenheit erlangt.

Visualisierung durch Grafana mit Node-Red und InfluxDB:

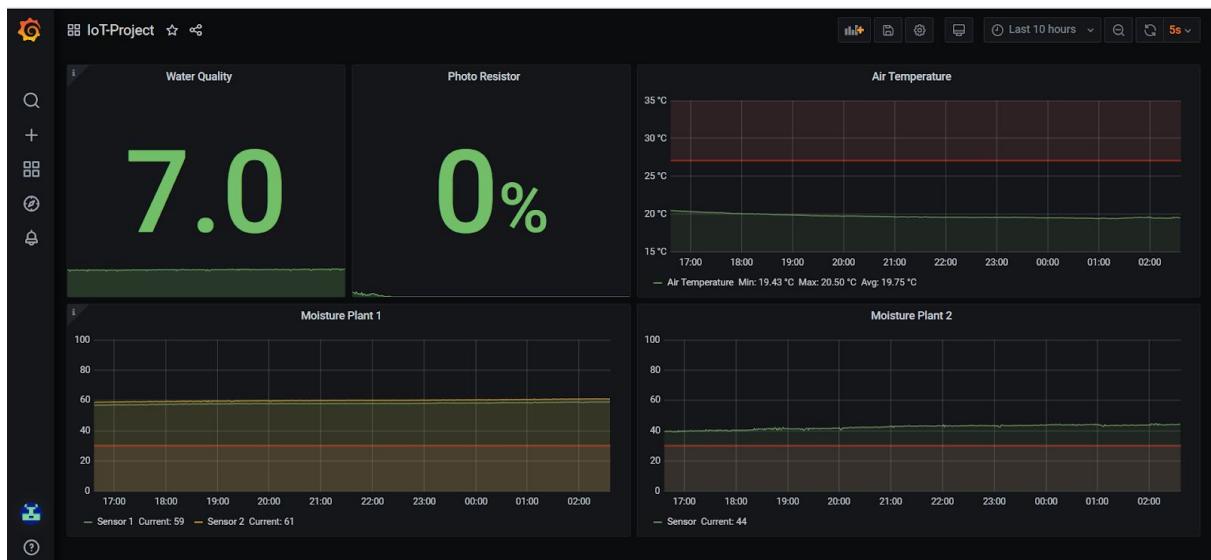
Die Messdaten werden allerdings auch von Node-Red abgefangen. Darauf ist eine Erweiterung für die InfluxDB Datenbank installiert. Man spezifiziert die Daten zu einem Sensor und diese Messwerte werden dann an die erwähnte Datenbank gesendet, wo die Daten gesichert werden. Node-Red mit Anbindung an InfluxDB ist somit die Schnittstelle zwischen den beiden ESP32 und Grafana.

Grafana wird benutzt, um die Daten für den Nutzer zu visualisieren. Dazu werden die Messergebnisse, die in der Datenbank gespeichert werden, hinzugezogen und dann in passenden Zeitgraphen dem Nutzer zur Verfügung gestellt. Dies gibt dem Nutzer die Möglichkeit, den Verlauf der Messergebnisse der letzten Minuten oder auch Tage oder Monate nachzuverfolgen. Dies ist vor allem nützlich, um Störungen aufzuzeigen, die das Gießen behindern, oder auch um das Wohlergehen der Pflanzen zu dokumentieren, was für den Nutzer auch von Interesse sein könnte.



Node-Red Funktionscode.

In Grafana werden die gesammelten Daten visualisiert. Dabei haben wir uns für folgendes Dashboard entschieden:



Unser Grafana Dashboard zeigt fünf unterschiedliche Panels an. Das erste Panel zeigt die Wasserqualität an, also den PH-Wert des Wassers. Das zweite Panel zeigt die Lichteinstrahlung. Das dritte Panel die Temperatur und den Grenzwert (27°C) ab dem nicht

mehr gewässert werden darf. das vierte Panel zeigt die Bodenfeuchtigkeit von der Pflanze 1 (zwei Sensoren) und das fünfte Panel zeigt die Bodenfeuchtigkeit von der zweiten Pflanze. Die empfangenen Werte werden dabei alle 5 Sekunden aktualisiert, um immer die neusten Informationen sehen zu können.



Man erkennt, dass der PH-Wert des Wasser fast Konstant bei 7.0 bleibt, was gut für die Pflanzen ist.



Das Bild zeigt eine 0%ige Lichteinstrahlung (==>Nacht), aber man kann sehen wie sich die Lichteinstrahlung über den Tag verändert und abnimmt.



Ebenfalls lässt sich bei der Temperatur der Verlauf eines Tages rekonstruieren und erkennt, dass es zum Abend hin kühler wird.



Die Bodenfeuchtigkeit nimmt im Verlauf kaum ab, was an einem regnerischen und feuchten Tag liegen kann.



Die Bodenfeuchtigkeit von Pflanze 2 nimmt

ebenfalls nicht ab, sondern zu was auf einen feuchten, regnerischen Tag schließen lässt.

Ausblick

Für die Zukunft ergeben sich nun noch mehrere Möglichkeiten dieses Projekt zu erweitern. Unser Projekt war für Zimmerpflanzen angepasst, jedoch lässt sich das Projekt auch für den Garten anpassen. Dafür ist es zum Beispiel möglich eine Wetter API einzubinden, um schon vorhersagen für die nächsten Tage einbinden zu können. Des Weiteren ist es auch eine Möglichkeit das Wasser direkt aus der Leitung zu verwenden, anstatt mit Pumpen das Wasser aus einem Behälter zu nutzen.

Eine andere Möglichkeit das Projekt zu erweitern besteht auch darin die graphische Visualisierung durch Grafana zu optimieren, sodass man dann die Grenzwerte für die bestimmten Sensoren (Temperatur, Feuchtigkeit, Licht, etc.) direkt über Grafana anpassen kann. Oder auch der manuelle Modus kann durch eine Webseite oder Grafana ermöglicht werden.

Auch kann man ein bestimmtes Zeitintervall festlegen in dem gegossen werden darf, zum Beispiel nur zwischen 18Uhr und 22Uhr.

Eines der Vorteile an dem Projekt ist auch, dass man es beliebig erweitern kann und den individuellen Anpassungen kaum Grenzen gesetzt sind.

Dieses Projekt bietet zudem auch die Möglichkeit mit den gesammelten Daten eine KI zu entwerfen, welche dann je nach Wetter, Tag, Pflanze, und Boden die bestimmten Grenzwerte optimal und individuell berechnet um eine noch größere Effizienz von Wasser, Zeit und Strom zu erreichen.

Fazit

Unser entwickeltes Bewässerungssystem ist eine gute Alternative für die momentan auf dem Markt vorhanden Bewässerungssysteme, wie zum Beispiel von Gardena. Vor allem macht sich dies im Preis deutlich bemerkbar, da wir auf Low-cost Hardware und Open-Source-Software gesetzt haben und somit die Kosten gering gehalten haben. Von Herstellern, die das gleiche Anbieten zahlt man mit Sicherheit mehr als das 3-fache. Außerdem ist diese Art der intelligenten Bewässerung noch nicht so flächendeckend auf dem Markt vorhanden, denn die meisten "automatischen Bewässerungssysteme" setzen meist nur auf eine Zeitsteuerung. Dies ist jedoch nicht so effizient wie unser System, denn wenn es zum Beispiel momentan regnet oder die Erde noch genug Feuchtigkeit gespeichert hat, muss schließlich nicht noch zeitgleich gegossen werden. Mit unserem Bewässerungssystem wird also garantiert, dass die Blumen, Hecken, Bäume und Pflanzen jeder Art nur genau so viel Wasser bekommen, wie sie wirklich benötigen. Zu dem werden die Pflanzen durch unser System geschützt, da auf die Bewässerung bei starker Hitze oder zu hoher Lichteinstrahlung verzichtet wird. Außerdem läuft unser System vollkommen autonom, sodass nie wieder ein Eingreifen des Nutzers nötig ist. Somit ist unser Projekt anderen Systemen oder der manuellen Bewässerung gegenüber nachhaltiger, effizienter und kostengünstig. Nachhaltig ist es dadurch, dass Pflanzen jeglicher Art vor einer Überwässerung oder vor dem Austrocknen geschützt werden und auf Umweltfaktoren (Wetter, Sonneneinstrahlung, Temperatur, etc) geachtet wird. Effizient ist es, da nur so viel Wasser wie nötig an genau die Stelle gebracht wird, wo das Wasser wirklich benötigt wird, nämlich an der Wurzel der Pflanze. Und kostengünstig ist es, weil Wasser gespart wird und

beim System auf Low-cost Geräte gesetzt wird. Nicht zu vergessen ist die viele Zeit die man sich spart und nun mit der Familie oder seinen Hobbys verbringen kann.

Somit ist unser System für jeden Haushalt, egal ob drinnen die Zimmerpflanzen oder draußen der komplette Garten, die richtige Wahl.