



**Hochschule für Technik  
und Wirtschaft Berlin**

*University of Applied Sciences*

---

## **Wissensmanagement**

---

**Sommersemester 2023**

Studiengang

**Wirtschaftsinformatik**

**Fachbereich 4**

vorgelegt von

**Ben Rump (s0563490)**

**Kaya Löher (s0564784)**

**Carmen Luu (s0564264)**

Datum:

**Berlin, 05.10.2023**

## Inhaltsverzeichnis

Abbildungsverzeichnis .....	III
1 Einleitung .....	1
1.1 Herausforderung .....	1
1.2 Lösungsansatz.....	2
2 Umsetzung.....	3
2.1 Vorgehensweise.....	3
2.2 Hardware .....	3
3 Ergebnisse .....	6
3.1 Bewässerung .....	6
3.2 InfluxDB.....	6
3.3 Grafana .....	8
4 Schlussbetrachtung.....	11
4.1 Fazit .....	11
4.2 Ausblick.....	11
4.3 Lessons Learned .....	12

## Abbildungsverzeichnis

Abbildung 1: Hardware .....	4
Abbildung 2: HydroTower - Deckel .....	4
Abbildung 3: Sensoren auf Board .....	5
Abbildung 4: LED aus (links)   LED an (rechts).....	6
Abbildung 5: InfluxDB - Organization .....	7
Abbildung 6: InfluxDB - Bucketansicht .....	8
Abbildung 7: Grafana - Datenquelle Einstellungen .....	9
Abbildung 8: Grafana - Dashboard .....	10

# 1 Einleitung

Im Rahmen des Moduls „Wissensmanagement“ im Sommersemester 2023 haben wir die Möglichkeit erhalten, erstmals mit Mikrocontrollern, Sensoren und vergleichbaren technischen Komponenten in Berührung zu kommen. Während dieses Kurses wurden wir vor eine spannende Herausforderung gestellt, die im Zeitraum vom 11. September bis zum 21. September zu bewältigen war. Im nachfolgenden findet sich die Challenge:

## Challenge #2: Food and the City – Stadtraum beackern

Seit der neolithischen Revolution, also seit die Menschen sesshaft geworden sind, spielt der Anbau von Nahrung eine wichtige Rolle. In den letzten Jahrhunderten ist die Welt zunehmend arbeitsteiliger geworden, sodass Städter heute kaum noch aktive Berührungspunkte mit der Landwirtschaft haben. In der Stadt wird viel gegessen, aber nicht viel angebaut. Ackerflächen sind rar, daher sind kreative Lösungen gefragt. Ausgehend von sozialen, ökologischen und wirtschaftlichen Fragestellungen werden wir aktuelle technologische Ansätze in den Fokus nehmen, um die Lebensmittelproduktion im Stadtraum zu befördern. In Kleingartenanlagen, auf Balkonen, Dachflächen, in Hochbeeten, Indoor oder auf schwimmenden Inseln - pflanzliche Lebensmittel können aller Orten wachsen. Natürlich müssen sich die frischen Stadtbauern auch kümmern und nicht alle sind gleich begabt. Kann

Technologie unser neuer grüner Daumen sein?

Für diese Challenge wurde eine Kooperation mit den Gründern des HydroTowers eingegangen. Dank einer automatisierten Bewässerungsfunktion kann der Tower bis zu drei Wochen ohne manuelles Gießen auszukommen. Dies ermöglicht den Anbau einer Vielzahl von Pflanzen, darunter Salate, Kräuter, Chilis, Tomaten und vieles mehr. Zusätzlich ist der HydroTower äußerst Effizienz, was sich in einem 30% bis 50% schnelleren Wachstum und einem um bis zu 90% reduzierten Wasserverbrauch zeigt. Darüber hinaus nimmt der Tower aufgrund seiner kompakten Bauweise lediglich die Fläche eines DIN-A3-Blattes in Anspruch.

## 1.1 Herausforderung

Derzeit erfolgt die Bewässerung der Pflanzen mittels einer festen Zeitsteuerung, welche durch eine Zeitschaltuhr reguliert wird. Dies ist jedoch unabhängig von tatsächlichem Bedarf und vernachlässigt entscheidende Faktoren. Um den gegenwärtigen Entwicklungsstand des HydroTowers zu verbessern, wurden klare Wünsche und Ziele durch die Gründer definiert. Primär wird eine smarte Steuerung der Pumpe angestrebt, welche es ermöglicht, den Bewässerungsprozess effizienter zu gestalten. Hierbei ist es von entscheidender Bedeutung, die

verschiedenen Faktoren wie Lichtverhältnisse, Temperatur und den Wasserstand zu berücksichtigen.

## 1.2 Lösungsansatz

Um die vorliegende Herausforderung anzugehen, wurde die Implementierung einer intelligenten Steuerung angestrebt, die sich durch eine bedarfsangepasste und dynamische Bewässerung auszeichnet. Diese Anpassung soll in Echtzeit erfolgen und sich nach den Faktoren Lichteinfall, Temperatur und Wasserstand richten. Dafür müssen die entsprechenden Sensoren in den HydroTower integriert werden, welche kontinuierlich Daten zu den genannten Faktoren erfassen und an ein Steuerungssystem weiterleiten. Auf Grundlage dieser Informationen soll die Bewässerung automatisch reguliert werden.

Außerdem liegen keine Informationen dazu vor, wie eine optimale Bewässerung aussieht, denn der einzige wirkliche Erfolgsindikator ist die Wuchsgeschwindigkeit der Pflanzen. Diesen muss man über einen längeren Zeitraum beobachten, als im Rahmen des Projektes zur Verfügung steht. Daher werden die gemessenen Daten erst einmal gespeichert und visualisiert, um daraus später die optimale Bewässerung ableiten zu können. Bis dahin werden bestmögliche Schätzwerte verwendet.

## 2 Umsetzung

In diesem Kapitel wird auf die Vorgehensweise und die eingesetzte Hardware eingegangen.

### 2.1 Vorgehensweise

Zu Beginn wurde versucht, mit dem IoT Octopus und LoRaWan zu arbeiten. Jedoch stellte sich heraus, dass die aktuelle Version dieser Technologien die Umsetzung nicht so leicht ermöglichte. Als Alternative fiel die Entscheidung auf den Arduino Nano mit BLE (Bluetooth Low Energy) in Kombination mit einem Tiny Machine Learning Shield. Zudem wurden die Sensoren Temperatur, UV-Strahlung und Ultraschall ausgewählt. Zur Simulation der Pumpenfunktion kam außerdem eine LED zum Einsatz. Anschließend konnten die Sensoren erfolgreich angeschlossen werden, womit es nun möglich war, zuverlässig Daten zu messen. Die Datenübertragung per Bluetooth an eine Bluetooth Handy-App verlief erfolgreich (der Code dafür wurde im finalen Sketch nicht verwendet, kann aber zur Weiterentwicklung hier gefunden werden:

[https://github.com/benkaru/WM-2023/blob/main/testing/testing\\_bluetooth.ino](https://github.com/benkaru/WM-2023/blob/main/testing/testing_bluetooth.ino)).

Allerdings stellte sich heraus, dass das automatische Speichern der Daten nicht sofort möglich war. Um dieses Problem zu umgehen, wurde vorübergehend auf die Kabelverbindung umgestellt. Der Arduino wurde an einen Laptop angeschlossen und es wurde ein Skript entwickelt, um die aktuell gemessenen Daten direkt an InfluxDB zu senden. Auf Grundlage dieser Daten wurde ein aussagekräftiges Dashboard in Grafana erstellt. Der nächste Schwerpunkt lag in der Entwicklung einer intelligenten Logik für die Bewässerung. Es wurden drei Einstellungen für die Pumpe erstellt, auch hier muss der Arduino am Laptop angeschlossen sein. Zur weiteren Optimierung der Bewässerung wurde eine automatische Steuerung für die LED-Beleuchtung entwickelt. Diese basierte auf Informationen von der RTC (Echtzeituhr) und den Sensoren.

### 2.2 Hardware

Für das Projekt wurden die folgenden Hardwarekomponenten verwendet:

- Arduino Tiny Machine Learning Kit
- Arduino Nano 33 BLE Sense Lite
- Grove - Temperature Sensor V1.2
- Grove - Ultrasonic Distance Sensor
- Grove - I2C UV Sensor (VEML6070)
- Grove - LED Socket Kit

- Jumper Kabel

Dies wird in der untenstehenden Abbildung verbildlicht:

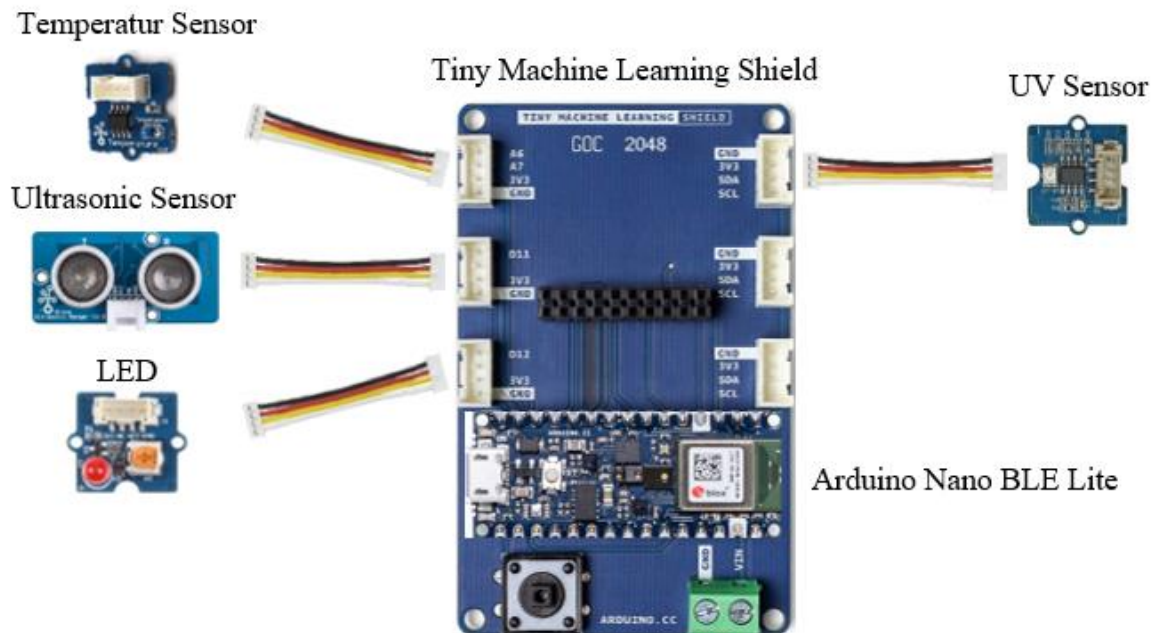
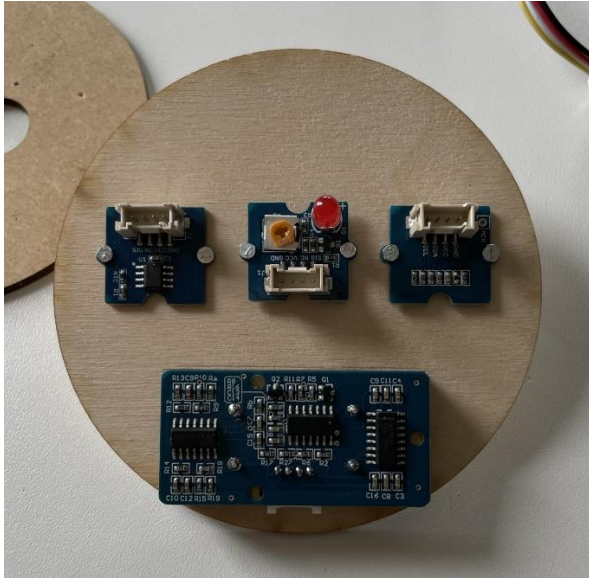


Abbildung 1: Hardware

Um die Hardware an dem HydroTower zu befestigen, haben wir mit einer Laserschneidemaschine einen Deckel aus Holz hergestellt. Dieser besteht aus einer runden Scheibe mit Löchern für die Sensoren (siehe Abbildung 3). Der Deckel ersetzt die bisherige Abdeckung des Wasserkastens des HydroTowers (siehe Abbildung 2). Auf dem Deckel aufsitzend sind der Mikrocontroller und die Wärme- und UV-Sensoren sowie die LED. Der Ultraschallsensor zur Bestimmung der Wasserhöhe im Kanister ist auf der Unterseite des Deckels angebracht.



Abbildung 2: HydroTower - Deckel



*Abbildung 3: Sensoren auf Board*



## 3 Ergebnisse

Dieses Kapitel präsentiert die erarbeiteten Ergebnisse, insbesondere die Umsetzung der LED-Simulation, die Datenverwaltung in InfluxDB und die Visualisierung in Grafana.

### 3.1 Bewässerung

Die Bewässerung beruht auf drei Messungen: Wasserstand, Temperatur und UV. Darauf aufbauend haben wir uns ein regelbasiertes System überlegt, das aus den Messwerten die beste Bewässerungsmenge ermittelt. Die vier Möglichkeiten der Bewässerung, die einem Entscheidungsbaum ähnlich implementiert wurden:

- kein Wasser: wenn es Nacht ist
- wenig Wasser: wenn der Wasserstand niedrig ist, wird automatisch wenig gegossen
- mittel Wasser: wenn genug Wasser da ist und es nicht zu heiß ist
- viel Wasser: wenn genug Wasser da ist und es heiß ist

Die tatsächlichen Werte, was “heiß” ist, wurden geschätzt und müssen durch Versuche bestätigt oder verändert werden.

Die Steuerung der Bewässerung sollte durch ein Relais erfolgen, welches zwischen der Stromzufuhr und der Pumpe geschaltet ist. Dadurch kann der Arduino dynamisch kontrollieren, wann die Pumpe laufen soll und es ist keine fest eingestellte Zeitschaltung nötig. Zu Beginn des Projektes war kein Relais vor Ort verfügbar, später hat eine andere Gruppe das einzige Bauteil gehabt. Da die Steuerung jedoch einfach ist, haben wir uns dazu entschieden, das fehlende Teil mit einer LED zu simulieren, damit es später ersetzt werden kann. Relais und LED haben zwei Zustände, AN und AUS. Ein Vorteil der LED ist die gute Sichtbarkeit während der Entwicklung.

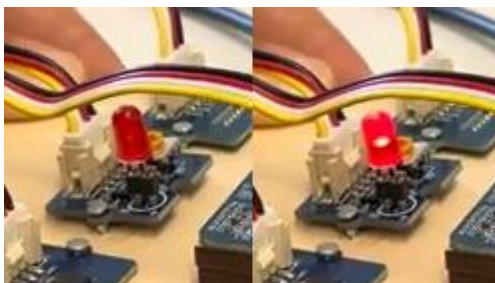


Abbildung 4: LED aus (links) / LED an (rechts)

### 3.2 InfluxDB

Um die gemessenen Daten in InfluxDB zu speichern, muss ein Account über die folgende Seite erstellt werden: <https://cloud2.influxdata.com/signup>. Anschließend wird eine sogenannte

*Organization* (siehe Abbildung 5) erstellt. Die Cluster URL (Host Name) wird im späteren Verlauf für Grafana benötigt.

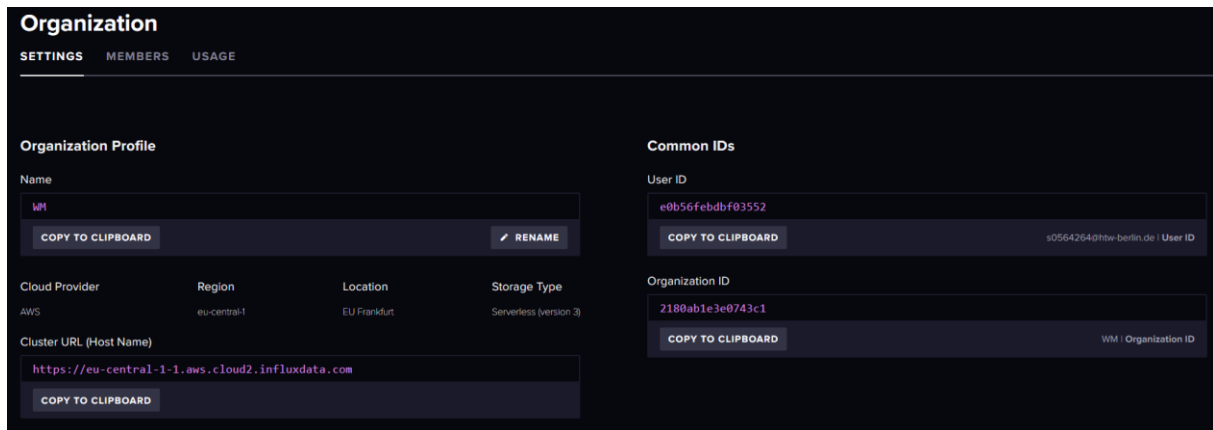


Abbildung 5: InfluxDB - Organization

Im nächsten Schritt muss ein Bucket erstellt werden: Load Data > Buckets > Create Bucket. Ein entsprechender Name muss vergeben und die präferierte Datenspeicherungsdauer festgelegt werden. In unserem Fall heißt der Bucket *hydrotower*.

Um die Daten von dem Mikrocontroller in die Datenbank zu übertragen, haben wir schlussendlich ein lokales Python-Script verwendet, das auf einem angeschlossenen Laptop/PC läuft (s. Kapitel 2.1). Dafür muss zuerst in InfluxDB ein API-Token erstellt werden.

Bucketname, Organisation und API-Token müssen in das Script (<https://github.com/benkaru/WM-2023/blob/main/uploadData.py>) eingetragen werden. Das Script hört auf eintreffende Daten von der Serial-Schnittstelle. Zusammengehörende Daten werden zusammengeführt, teilweise transformiert und anschließend über zwei Wege gespeichert. Zum einen wird eine lokale CSV-Datei und zum anderen wird die InfluxDB SDK verwendet, um die Daten über HTTP in InfluxDB zu sichern.

The screenshot shows the InfluxDB Schema Browser interface. On the left, the 'Bucket' is set to 'hydrotower' and the 'Measurement' is 'arduino\_data'. The main area displays a table of data with the following columns: Temperature (no group, double), time (no group, dateTime:MFC339), Ultrasonic (no group, double), UVS (no group, double), UVS\_readable (no group, string), and Water (no group, double). The table contains 185 rows of data. The data is as follows:

Temperature	time	Ultrasonic	UVS	UVS_readable	Water
23.91	2023-09-20T09:56:56.000Z	0	0	Niedrig	21.667
23.66	2023-09-20T09:57:06.000Z	6	0	Niedrig	18.333
23.26	2023-09-20T09:57:16.000Z	1	0	Niedrig	21.111
23.42	2023-09-20T09:57:26.000Z	1	0	Niedrig	21.111
23.58	2023-09-20T09:57:36.000Z	16	1	Mittel	12.778
22.94	2023-09-20T09:57:46.000Z	19	1	Mittel	11.111
23.02	2023-09-20T09:57:56.000Z	11	1	Mittel	15.556
23.1	2023-09-20T09:58:06.000Z	0	2	Hoch	21.667
22.94	2023-09-20T09:58:16.000Z	0	2	Hoch	21.667
22.3	2023-09-20T09:58:26.000Z	0	2	Hoch	21.667

Abbildung 6: InfluxDB - Bucketansicht

### 3.3 Grafana

Um ein Dashboard basierend auf den Daten von InfluxDB zu erstellen, ist zunächst die Erstellung eines Kontos über <https://grafana.com/> erforderlich. Anschließend muss eine Verbindung zu InfluxDB hergestellt werden. Hierfür muss über *Home > Connections > Data sources > Add data source* navigiert und InfluxDB ausgewählt werden.

Für die Verbindung kann ein beliebiger Name gewählt werden. Die Abfragesprache muss von InfluxQL auf Flux umgestellt werden. Unter dem Reiter *HTTP* wird die zuvor genannte Cluster URL von InfluxDB eingetragen und die Option *Basic Auth* ausgewählt. Die Anmeldedaten für Basic Auth entsprechen den Zugangsdaten von InfluxDB. Im Abschnitt *InfluxDB Details* sind die *Organization* und der *Default Bucket* anzugeben. Anschließend kann die Verbindung gespeichert und getestet werden. Bei erfolgreicher Verbindung werden verfügbare Buckets angezeigt. Dieser Vorgang ist in der Abbildung unten dargestellt.

Abbildung 7: Grafana - Datenquelle Einstellungen

Nun können die Daten von InfluxDB visualisiert werden. Zuerst muss ein neues Dashboard angelegt werden: *Home > Dashboards > New > New Dashboard*. Anschließend wird *Add visualization* ausgewählt und die davor verbundene Datenquelle ausgewählt. Um die Temperatur anzeigen zu lassen muss folgender Code eingefügt werden:

```
from(bucket: "hydrotower")
  |> range(start: -5y)
  |> filter(fn: (r) => r._field == "Temperature")
```

Zunächst ist der *Bucketname* anzugeben. Die *range* bestimmt, wie weit Daten in der Vergangenheit angezeigt werden können und der *filter* legt genau eine Variable fest, die angezeigt werden soll. Daraufhin kann eine passende Visualisierung von den gegebenen Optionen ausgewählt und nach belieben angepasst werden. Somit kann beispielsweise die Einheit Celsius und die Farbe festgelegt werden. Dieser Ablauf muss für alle gewünschten Variablen wiederholt werden. In Abbildung 8 ist das erstellte Dashboard für das Projekt zu sehen.



Abbildung 8: Grafana - Dashboard

## 4 Schlussbetrachtung

Im folgenden Kapitel dieser Arbeit erfolgt eine umfassende Betrachtung des Projektes. Hierbei wird das Fazit gezogen, ein Ausblick auf potentielle Weiterentwicklungen gegeben und die gewonnenen Erkenntnisse reflektiert.

### 4.1 Fazit

Die Untersuchung der verschiedenen Aspekte im Zusammenhang mit dem HydroTower-Projekt ergab eine vielversprechende Grundlage für die Erweiterung und Verbesserung dieses Systems. Die Möglichkeit, den HydroTower gut zu erweitern, ermöglicht eine zukünftige Anpassung an individuelle Bedürfnisse und wachsende Anforderungen.

Die Zeitschaltuhr mit einer einfachen Logik in Arduino zu ersetzen, konnte leicht umgesetzt werden. Die Verwendung von Sensoren wie Temperatur, UV und Ultraschall in der Low-Level-Umsetzung dieser Logik bietet eine schnelle Lösung für die Bewässerung, genauso wie zusätzlich die Überwachung wichtiger Parameter im Hydroponik-System.

Die erfolgreiche Anwendung von Open-Source-Visualisierungstools wie Grafana und InfluxDB ermöglicht eine benutzerfreundliche Darstellung der Daten und erleichtert die Analyse.

Insgesamt bietet das HydroTower-Projekt vielversprechende Möglichkeiten zur Verbesserung der Pflanzenkultivierung und zur Automatisierung von Bewässerungsprozessen. Mit der richtigen Hardware, Sensoren und Anpassungen kann es zu einem effizienten und benutzerfreundlichen System entwickelt werden.

### 4.2 Ausblick

Um das HydroTower-Projekt weiter voranzutreiben, sind einige wichtige Schritte notwendig.

Eine vertiefte Recherche für passende Hardwarekomponenten ist unerlässlich, um die optimale Leistung und Effizienz zu gewährleisten. Die Auswahl eines geeigneten Relais für die Pumpe sollte erfolgen, um eine zuverlässige Steuerung sicherzustellen. Die Evaluation der Sensoren hinsichtlich ihrer Genauigkeit und ihres Preises ist von großer Bedeutung, um die Kosten zu optimieren und gleichzeitig genaue Daten zu erhalten. Die Integration des UV-Sensors würde die Überwachungsfunktionen des Systems erweitern. Die Erweiterung durch einen Feuchtigkeitssensor in den Töpfen ermöglicht eine noch präzisere Bewässerungslogik und kann das Pflanzenwachstum verbessern. Das Ersetzen der LED durch eine Pumpe ist leicht umzusetzen. Eine WLAN-Anbindung würde die Fernüberwachung des HydroTowers

ermöglichen, selbst in Abwesenheit. Des Weiteren würde das Bauen eines Gehäuses die Komponenten vor äußeren Einflüssen schützen und dem System eine professionelle Ästhetik verleihen. Zu guter Letzt stellt die Erweiterung der Logik für die manuelle Steuerung während Urlaubszeiten (unter Berücksichtigung bereits vorhandener Schnittstellen im Code), eine sinnvolle Erweiterung dar.

#### 4.3 Lessons Learned

Die Lessons Learned sollen Einblicke in die Erfahrungen und Herausforderungen des Projekts geben. Sie dienen als Grundlage für zukünftige Projekte, um ähnliche Fehler zu vermeiden und die Planung und Umsetzung zu verbessern.

#### **Hardware**

##### Verschiedene Stärken und Schwächen von Mikrocontrollern:

Die Erfahrung zeigt, dass die Auswahl des richtigen Mikrocontrollers von entscheidender Bedeutung ist. Jeder Mikrocontroller hat seine eigenen Vor- und Nachteile, die die Projektdurchführung erheblich beeinflussen können. Eine gründliche Analyse und Vergleich der verfügbaren Optionen ist unerlässlich, um die bestmögliche Wahl zu treffen.

##### Einarbeitung ist zeitintensiv:

Die Einarbeitung in die Hardwarekomponenten kann zeitaufwendig sein. Es ist wichtig, genügend Zeit für das Erlernen der benötigten Fähigkeiten und Kenntnisse einzuplanen, insbesondere wenn das Team noch keine Erfahrung mit der spezifischen Hardware hat.

#### **Projekt**

##### Smarte und ML-Lösungen erfordern ausreichende Daten:

Das Streben nach smarten oder maschinellen Lösungen ist zwar erstrebenswert, jedoch müssen ausreichend Daten verfügbar sein - um solche Modelle zu trainieren und zu validieren - die in der kurzen Zeit des Projektes nicht vorhanden waren.

##### Zeitraumen für weiterführende Messungen:

Es ist wichtig, realistische Zeiträume für die Durchführung weiterführender Messungen und Experimente zu setzen. In vielen Fällen kann der vorgesehene Zeitraum nicht ausreichen, um umfassende Forschung und Tests durchzuführen. Dies sollte bei der Planung berücksichtigt werden.

## **Rahmenbedingungen**

### Workshops für Hardware erst gegen Ende des Projektes:

Die Planung und Bereitstellung von Schulungen und Workshops für Hardwarekomponenten sollte von Anfang an in Betracht gezogen werden. Eine späte Einführung von Schulungen kann zu Verzögerungen und Schwierigkeiten in der Projektumsetzung führen.

### Wenig Unterstützung:

Wenn das Projekt nur begrenzte Ressourcen und Unterstützung erhält, sollten klare Kommunikationswege und Strategien entwickelt werden, um Herausforderungen zu bewältigen. Dies könnte die Suche nach externer Unterstützung, die Zusammenarbeit mit anderen Teams oder die Anpassung der Projektziele umfassen.

### Ständiger Wechsel des Zeitplans:

Ein sich häufig ändernder Zeitplan kann die Projektumsetzung erheblich beeinträchtigen. Es ist wichtig, eine klare Projektplanung und -organisation aufrechtzuerhalten und den Zeitplan nur bei Bedarf und nach sorgfältiger Abwägung von Risiken und Konsequenzen anzupassen.