



**Hochschule für Technik  
und Wirtschaft Berlin**

University of Applied Sciences

---

# Luftfeuchtigkeits-Sensornetzwerk zur zeitnahen Detektion von Wasserschäden auf Basis von LoRa(WAN)

## Projektabschlussbericht

von

**Sidney Göhler und Ilja Buschujew**

Fachbereich 1 – Energie und Information –  
der Hochschule für Technik und Wirtschaft Berlin

im Modul

**Projekt Netzbasierte Systeme**

des Studienganges

**Informations- und Kommunikationstechnik (M. Eng.)**

Tag der Abgabe: 25.02.2022

Technischer Input: Prof. Dr. Thomas Scheffler

Projektmanagement: Hanna Full

---



# Inhaltsverzeichnis

<b>1</b>	<b>Einleitung</b>	<b>1</b>
1.1	Vorstellung der Projektidee . . . . .	1
1.2	Ausgangslage und Zielsetzung . . . . .	2
1.3	Strukturierung des Projektberichtes . . . . .	3
<b>2</b>	<b>Projektplanung</b>	<b>5</b>
2.1	Pflichtenheft . . . . .	5
2.1.1	Vorgangsmodell . . . . .	5
2.1.2	Projektstruktur . . . . .	7
2.1.3	Zeitplanung . . . . .	7
2.1.4	Kostenaufstellung . . . . .	10
2.2	Systemkonzept und theoretische Realisierung . . . . .	10
2.2.1	Systemkonzept . . . . .	10
<b>3</b>	<b>Grundlagen</b>	<b>11</b>
3.1	LoRa und LoRaWAN . . . . .	11
3.2	MQTT . . . . .	14
<b>4</b>	<b>Praktische Umsetzung</b>	<b>15</b>
4.1	Verwendete Hardware . . . . .	15
4.1.1	LoPy4-Development-Board . . . . .	15
4.1.2	DHT11/-22 Sensormodul . . . . .	15
4.1.3	Restliche Hardware . . . . .	15
4.2	Beschreibung der Software . . . . .	15
4.2.1	1. Komponente: Sensoransteuerung und der Versand der Daten mittels LoRa(WAN) . . . . .	16
4.2.2	2. Komponente: Empfangen der Daten und Versand ins Internet	16
4.2.3	3. Komponente: Manuelles abrufen und versenden der veröf- fentlichten Daten . . . . .	16
4.3	Visualisierung der Sensordaten . . . . .	16
4.4	Berechnung der Laufzeit im Batteriebetrieb . . . . .	16

<b>5</b>	<b>Fazit</b>	<b>17</b>
5.1	Projektauswertung . . . . .	17
5.2	Probleme und Herausforderungen . . . . .	17
5.3	Ausblick . . . . .	17
5.4	Abschlusswort . . . . .	17
<b>A</b>	<b>Anhang</b>	<b>19</b>
	Abbildungsverzeichnis	21
	Tabellenverzeichnis	23
	Quelltextverzeichnis	25
	Literaturverzeichnis	27
	Eigenständigkeitserklärung	29

# Kapitel 1: Einleitung

## 1.1 Vorstellung der Projektidee

Die Digitalisierung hat unsere Art und Weise wie die Gesellschaft lebt und wie verrichtete Arbeit wertgeschätzt wird, grundlegend verändert. Es sind nicht mehr die Menschen, sondern Computer und Maschinen, die den Takt vorgeben und die Maßstäbe setzen. Arbeit und soziales Zusammenleben werden in einer von freien Marktwirtschaft geleiteten Gesellschaft durch die Digitalisierung neu bestimmt. Begriffe wie Homeoffice und Telearbeit sind aus unserem heutigen Arbeitsleben kaum mehr wegzudenken, was schlussendlich in unserer globalisierten Welt zu einem Optimierungswahn geführt hat. Weitere Folgen sind unter anderem die Privatisierung von Wissen und Information, sowie die Ausbeutung von Mensch und Natur.

Aus diesen und weiteren Gründen wünschen sich immer mehr Menschen einen Rückschritt zu einer Gesellschaft, bei der moralische Werte über den wirtschaftlichen Erfolg gestellt werden. Sie wünschen sich mehr Selbstbestimmung, unter Rücksichtnahme der vorhandenen natürlichen Ressourcen und beteiligten Personen, um schlussendlich die vorherrschende Ellenbogengesellschaft durch eine sozialere auszutauschen.

Im Diskurs werden unter Anderem Begrenzung Anderer, Grenzsetzung gegenüber Anderen, aber auch Ausgrenzung Anderer bzw. die eigene Ausgrenzung thematisiert und in Frage gestellt, wodurch sich unter anderem die Bewegung der „Urban Commoner“ herauskristallisiert hat.

Urban Commons zielt auf eine Entwicklung von individuellen und gesellschaftlichen Werte und Normen, auf Basis eines Zusammenschlusses einzelner Individuen, um ein bestimmtes Gebiet oder eine bestimmte Ressource unabhängig zu gestalten.

Wir, Ilja Buschujew und Sidney Göhler, möchten mit unserem Luftfeuchtigkeit-Temperatur-Sensor Netzwerk unseren Beitrag dazu leisten, um prinzipiell jedem die Möglichkeit zu bieten, seine eigenen Daten zu sammeln und diese mit seinem Umfeld

zu teilen, um dem sich fortschreitenden Konkurrenzgedanken innerhalb seines Umfeldes entgegen zu wirken ohne dabei auf seine individuellen Bedürfnisse verzichten zu müssen.

## 1.2 Ausgangslage und Zielsetzung

Wie schon im Abschnitt 1.1 beschrieben, versucht unser Projekt „Luftfeuchtigkeits-Sensornetzwerk auf Basis von LoRa(WAN)“ das Thema des „Urban-Commons“, was aus dem englischen kommt und so viel wie: „gesellschaftliches- oder städtisches Gemeingut“ bedeutet, aufzugreifen. Aber was versteht man jetzt genau unter dem Begriff „gesellschaftliches Gemeingut“ eigentlich?

Wir Menschen sind eine soziale und kooperative Spezies, die zu weitaus wundervollen Erzeugnissen fähig ist. Der Begriff der Emergenz stellt ein wunderbares Beispiel dafür dar. Es bezeichnet die Möglichkeit der Herausbildung von neuen Eigenschaften oder Strukturen eines Systems infolge des Zusammenspiels seiner Elemente. So ist das auch in der Gesellschaft; wenn die Menschen zusammen an einer Aufgabe oder einem Projekt arbeiten, können neue Strukturen und Eigenschaften der Gesellschaft daraus wachsen. Das Kollektiv ist also mehr als die Summe der einzelnen Individuen, denn die individuelle Identität ist immer auch Teil kollektiver Identitäten. Es gibt daher kein isoliertes Ich, sondern ein Ich-in-Bezogenheit [BH19]. Unser Identität wird von Anfang an aus Beziehungen zu anderen heraus gebildet.

„Die Welt als Commons zu denken und zu gestalten bedeutet, unsere Kooperationsfähigkeit so zu nutzen, dass sich niemand über den Tisch gezogen fühlt, aber auch niemandem ein Platz am Tisch verweigert wird.“ [BH19].

Unserer Meinung nach betrifft es vor allem, die kooperative Gestaltung des eigenen Wohnumfeldes, welches unabhängig vom Staat, Markt, den sozialen Status, Herkunft, oder dem eigenen Einkommen stattfinden soll. Dabei spielt die Selbstbestimmung eine zentrale Rolle. Daher haben wir uns auch für das Projekt „Luftfeuchtigkeits-Sensornetzwerk auf Basis von LoRa(WAN)“ entschieden, dass an dem Prinzip des Commons anknüpfen soll.

Bei unserem Projekt soll jeder der Lust oder das Bedürfnis hat die Möglichkeit haben, einen eigenen Luftfeuchtigkeits- und Temperatursensor im Keller anzubringen und so bei Tagen an dem z.B. viel Regen fällt, oder es zu einem Rohrbruch im Keller kommt, wo das Wasser sich ansammelt und eventuell zu Sachschäden oder ähnlichen führen kann, zu warnen und mit anderen Menschen dies zu teilen.

Die Sensorwerte sollen über die LoRa-Funktechnik, dessen Frequenzband, genau wie WLAN oder Bluetooth, im unlizenzierten ISM-Band liegt, versendet und damit keine Gebühren für die Nutzung des Frequenzbandes bezahlt werden. Darüber hinaus weist LoRa eine durchaus hohe Reichweite auf, sodass damit auch mehrere Gebiete gleichzeitig im Umkreis von mehreren Kilometern abgedeckt werden und damit mehr Menschen sich an dem Netzwerk anschließen können, welches bei dem LoRaWAN (Longe Range Wide Area Network) der Fall ist. Jedoch beschränken wir uns in unserem Projekt nur auf eine Punkt-zu-Punkt LoRa Kommunikation, die man aber später noch ausbauen und zu LoRaWAN erweitern könnte.

## 1.3 Strukturierung des Projektberichtes

Im nachfolgenden Kapitel 2. wird die herangehensweise der Produktentwicklung, verbunden mit dem Pflichtenheft, welches aus der Auswertung des Umfragebogens heraus entsteht, sowie die resultierende theoretische Realisierung der Projektidee im Form eines Systemkonzeptes und Blockschaltbildes. Zum Schluss wird die Art und Weise des Managements für unser Projekt beschrieben.

Im Kapitel 3. wird auf die Grundlagen, wie der Funktionsweise von LoRa und LoRaWAN, sowie dem MQTT-Protokoll, eingegangen.

Im 4. Kapitel wenden wir uns der praktischen Realisierung zu. Dabei beschreiben wir zunächst einmal die Hardware, die wir für das Projekt verwendet haben. Wir gehen auf die Funktionsweise und die besonderen Eigenschaften der Mikrocontroller, der Development-Boards und den Sensortyp ein. In Form eines Schaltplans wird die Verdrahtung der einzelnen Hardware-Komponenten dargestellt und beschrieben.

Im zweiten Teil der praktischen Realisierung wird die Umsetzung in der Software beschrieben. Dabei gehen wir auf die Umsetzung der Punkt-zu-Punkt LoRa-Kommunikation zwischen dem LoRa-Sender und -Empfänger ein. Es wird die Einbindung des MQTT-Protokolls in der Software beschrieben und die Visualisierung der Sensordaten im Form eines Dashboards dargestellt. Abschließend wird der Stromverbrauch im Batteriebetrieb veranschaulicht und ausgewertet.

Abschließend wird im Kapitel 5. die Arbeit mit einem Fazit, im Form der Projektauswertung, der Probleme und Herausforderungen, die während der Arbeit entstanden sind, sowie ein Ausblick auf zukünftige Verbesserungsmöglichkeiten und eines Abschlusswortes, beendet.





# Kapitel 2: Projektplanung

## 2.1 Pflichtenheft

Auch wenn wir in erster Linie ein Produkt aus freien Inhalten entwickeln wollen, müssen wir an einigen Stellen den Kompromiss zwischen freiem Inhalt und Entwicklungsaufwand eingehen, da wir in der zeitlichen Ressource begrenzt sind. Im Zuge der Projektplanung haben wir eine Umfrage durchgeführt, woraus sich unser Pflichtenheft abgeleitet hat.

- **Anschaffungskosten:** Unser Persona möchte maximal 40 Euro in dieses Projekt investieren, um einen funktionsfähigen Funksensor zu erhalten.
- **Laufende Kosten:** Um die laufenden Kosten gering zu halten, soll der Funksensor möglichst stromsparend und wartungsarm sein. Die Hardware sollte ihren Strom über einen Akku bzw. eine Batterie beziehen.
- **Einsatzgebiet:** Da der Sensor am Mikrocontroller im Bereich von  $-40^{\circ}\text{C}$  . . .  $80^{\circ}\text{C}$  arbeitet und auch in Gebieten mit einer relativen Feuchtigkeit von bis zu 100% zum Einsatz kommen soll, muss das fertige Endprodukt mindestens diesen Anforderungen entsprechen. Die Entwicklung eines Gehäuses erfolgt aber erst nach Erreichen der Serienreife.
- **Datenschutz:** Der Endnutzer möchte selbst bestimmen, ob und mit wem er seine gesammelten Daten teilt. Des weiteren möchte er selbst bestimmen, ab welchem Zeitpunkt/Schwellwert er über den aktuellen Datenstand informiert wird.

### 2.1.1 Vorgangsmodell

Scrum ist ein Vorgehensmodell im Projektmanagement, welches seinen Ursprung in der Softwareentwicklung hat. Der Ansatz von Scrum ist das systematische Sammeln von Erfahrungen (empirisch), das kontinuierliche weiterentwickeln bestehender

Module (inkrementell), sowie dem mehrfachen Wiederholen gleicher oder ähnlicher Prozesse (iterativ) und basiert auf der Erkenntnis, dass viele Entwicklungsprojekte zu komplex sind, um sie in einem vollumfänglichen Plan fassen zu können, was wiederum den Grund hat, dass wesentliche Teile der Ursprungsanforderung bzw. deren Lösungsansätze zu Beginn unklar sind.

Ein weiteres Merkmal von Scrum ist, dass neben dem Produkt auch die Planung kontinuierlich verändert bzw. weiterentwickelt wird, wobei der langfristige Plan, auch Product Backlog genannt, iterativ verfeinert und verbessert wird.

Resultierende Arbeitspakete werden zyklisch in sogenannten Sprints detailliert formuliert und in einem Detailplan, auch Sprint Backlog genannt, zur Bearbeitung abgelegt, sodass diese, fokussiert auf die aktuelle Problemstellung, abgearbeitet werden können.

Ziel ist dabei eine schnelle und kostengünstige Entwicklung hochwertiger Produkte, wobei die jeweiligen Anforderungen aus der Anwendersicht formuliert werden.

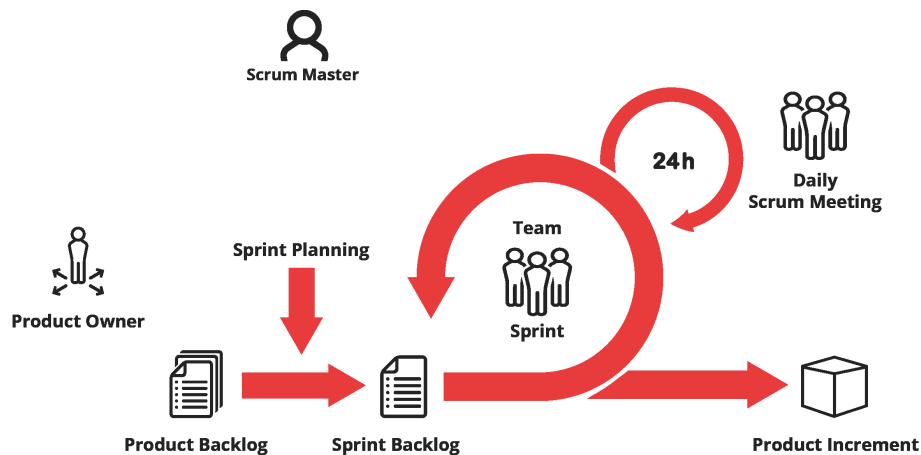


Abbildung 2.1: Agiles arbeiten mit Scrum

Die Verantwortlichkeiten liegen beim sogenannten Scrum-Team, welches sich aus folgenden Rollen ergibt:

Rolle	Besetzung in unserem Projekt	Anmerkung
Product Owner	HTW Berlin	Vertreten durch Prod. Dr. Thomas Scheffler
Scrum Master	-	-
Projektmanager	Sidney Göhler	-
Scrum Team	Sidney Göhler, Ilja Buschujew	-

Tabelle 2.1: Verantwortlichen im Scrum-Team

## 2.1.2 Projektstruktur

Aus den uns gesetzten Pflichten, haben sich für uns die folgende Projektstruktur grob herauskristalisiert, welche im laufenden Prozess immer weiter in Arbeitspakete verfeinert wurde.

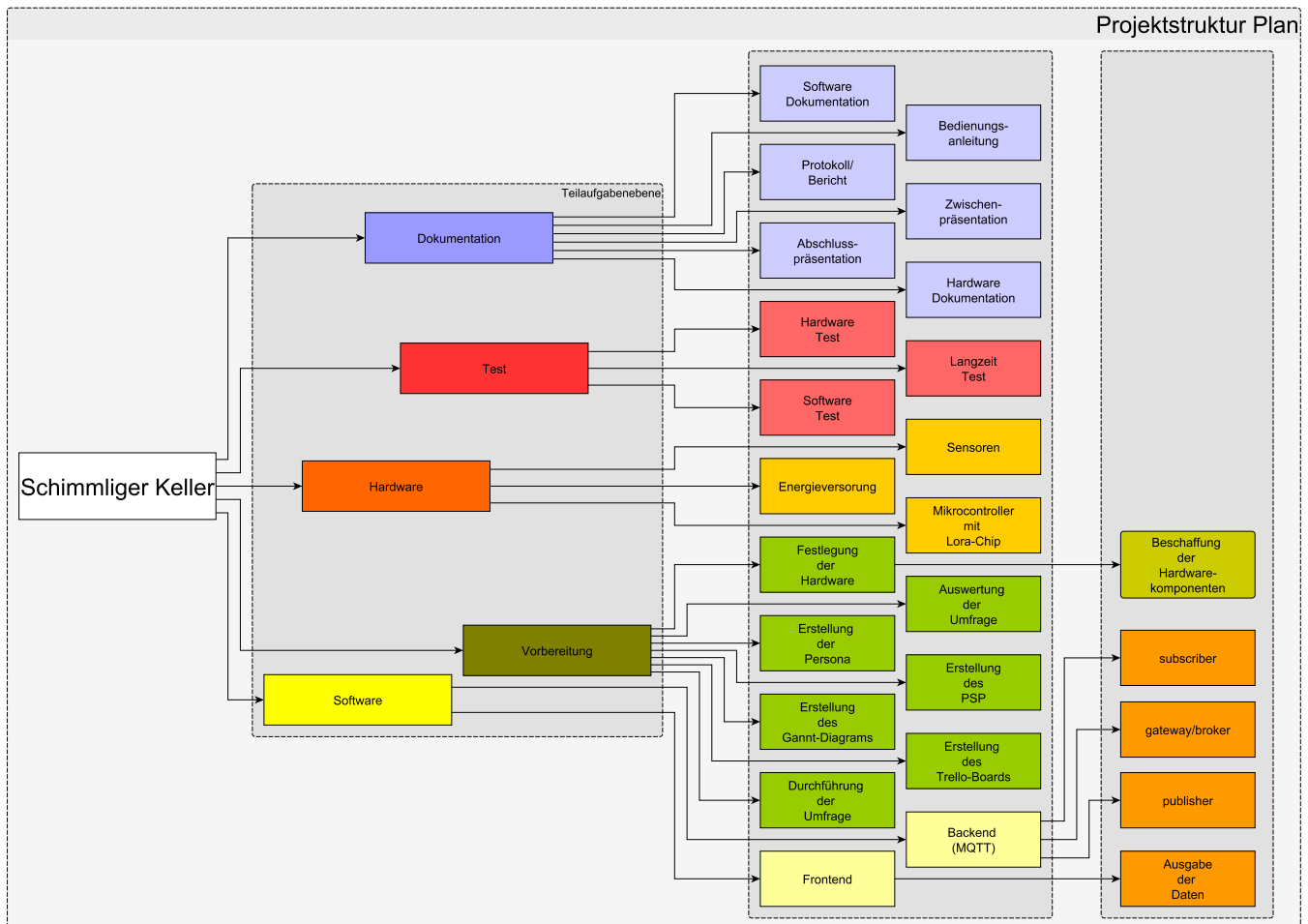


Abbildung 2.2: Projektstrukturplan

Mithilfe des Projektstrukturplanes ließen

## 2.1.3 Zeitplanung

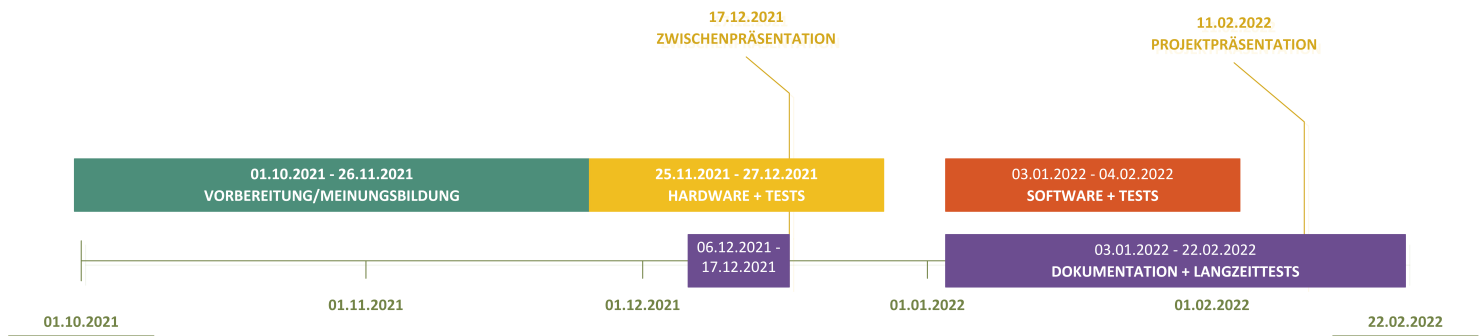


Abbildung 2.3: Übersicht unserer Zeitplanung

Illustriert wird unsere grobe Zeitplanung zu Beginn des Projektes. Wir wollten aufgrund unserer agilen Arbeitsweise nur grobe Zeiträume definieren. Wie sich schlussendlich herausgestellt hat, haben manche Teilaspekte länger, andere wiederum kürzer gedauert.

Nachfolgend wird eine tabellarische Übersicht unserer Zeitplanung aufgeführt, wobei wir die Daten aus unserem Trello-Board entnehmen:

Vorgangsname	Anfang	Ende	Bearbeiter	Arbeitszeit
<b>Vorbereitung</b>				
Sichtung der Quellenlage	01.10.2021	20.02.2022	S + I	20h
Erstellung Trelloboard	01.10.2021	01.10.2021	S	30m
Erstellung eines Git repositories	01.10.2021	01.10.2021	S	30m
Erstellung des PSP	01.10.2021	20.02.2022	I	2h
Erstellung Zeitplan	01.10.2021	20.02.2022	S	2h
Erstellung/Durchführung der Umfrage	01.10.2021	02.12.2021	S + I	20h
Festlegung/Beschaffung der Hardware	01.10.2021	20.02.2022	S + I	20h
Auswahl/Einrichtung Software IDE	01.10.2021	20.02.2022	S + I	20h
Auswertung der Umfrage	01.10.2021	20.02.2022	S + I	20h
Erstellung des Persona	01.10.2021	20.02.2022	S + I	20h
<b>Hardware</b>				
Inbetriebnahme der Hardware	01.10.2021	20.02.2022	S + I	20h
<b>Software</b>				
µC Management	01.10.2021	20.02.2022	S + I	20h
Integration der Sensoren	01.10.2021	20.02.2022	S + I	20h
Einrichtung einer P2P LoRa-Kommunikation	01.10.2021	20.02.2022	S + I	20h
Entwicklung eines MQTT Publishers	01.10.2021	20.02.2022	S + I	20h
Entwicklung eines MQTT Subscribers	01.10.2021	20.02.2022	S + I	20h
Einrichtung einer grafischen Schnittstelle	01.10.2021	20.02.2022	S + I	20h
Softwaretests	01.10.2021	20.02.2022	S + I	20h
<b>Nachbereitung</b>				
Vorbereitung der Zwischenpräsentation	01.10.2021	20.02.2022	S + I	20h
Vorbereitung der Abschlusspräsentation	01.10.2021	20.02.2022	S + I	20h
Dokumentation	01.10.2021	20.02.2022	S + I	20h
Langzeittests	01.10.2021	20.02.2022	S + I	20h

Tabelle 2.2: Übersicht der Arbeitspakete und Arbeitszeiten

### 2.1.4 Kostenaufstellung

Auch wenn wir unser Projekt in erster Linie als Freie Software bereitstellen wollen, Aus unseren Anforderungen geht hervor, dass die meisten potentiellen Nutzer möglichst wenig für unser Produkt bezahlen möchten. Wie bei

PyCom			DIY		
Name	Anzahl	Kosten	Name	Anzahl	Kosten
LoPy4	1	38,45	ESP32 DevKit	1	9,99
Pytrack	1	40,65	Breadboard	1	5,99
Antennen-Kit	1	9,00	LoRa Transceiver + Antenne	1	11,98
3 X 1.5V AAA Batterie-Halterung	1	9,00	3 X 1.5V AAA Batterie-Halterung	1	9,00
DHT22 Sensor	1	9,99	DHT22 Sensor	1	9,99
			Passive Bauelemente		2,00
<b>Gesamtkosten</b>		107,09	<b>Gesamtkosten</b>		48,95

Tabelle 2.3: Kostenaufstellung für das Projekt

Anzumerken ist hier, dass die Entwicklung des Produktes mithilfe der PyCom Plattform zwar deutlich kostenintensiver ist, aber besonders für einen Prototypen doch recht komfortabel, da jeder einzelne Baustein dafür gemacht ist, miteinander zu funktionieren.

Tendenziell ließe sich aber mit einem Selbstbau der Preis um knapp die Hälfte reduzieren.

Neben der PyCom Plattform existieren noch andere Entwicklungsplattformen, wie z.B. das *TOOGOO WiFi ESP-32 Entwicklungs Board*, welches den LoRa Transceiver implementiert und teilweise auch schon eine Halterung für Batterien anbietet.

## 2.2 Systemkonzept und theoretische Realisierung

### 2.2.1 Systemkonzept

# Kapitel 3: Grundlagen

## 3.1 LoRa und LoRaWAN

Der Begriff LoRa steht für **L**ong **R**ange und definiert dabei ein funkbasiertes Übertragungsverfahren auf der Bitübertragungs- (physical layer) und der Sicherungsschicht (MAC layer) im OSI-Schichtenmodell. Es wurde von der französischen Firma Cycleo, welche später von Semtech Corporation abgekauft wurde, entwickelt.

LoRa kann zu den sogenannten **L**ow-**P**ower-**W**ide-**A**rea-**N**etwork (LPWAN) Technologien zugeordnet werden, die einen energiesparsamen Betrieb und eine hohe Übertragungsreichweite aufweisen. Im Vergleich zu WLAN oder Mobilfunk, fällt jedoch die Daten- bzw. Bandbreite bei diesen Technologien relativ gering aus, sodass diese hauptsächlich bei drahtlosen Sensornetzwerken Anwendung finden, wo es darum geht Sensordaten mit einer geringen Datenrate über eine weite Funkstrecke zu übertragen. Ein Vergleich zu den funkbasierten Technologien (WLAN, LPWAN und Mobilfunk) stellt die Abb. 3.1 dar.

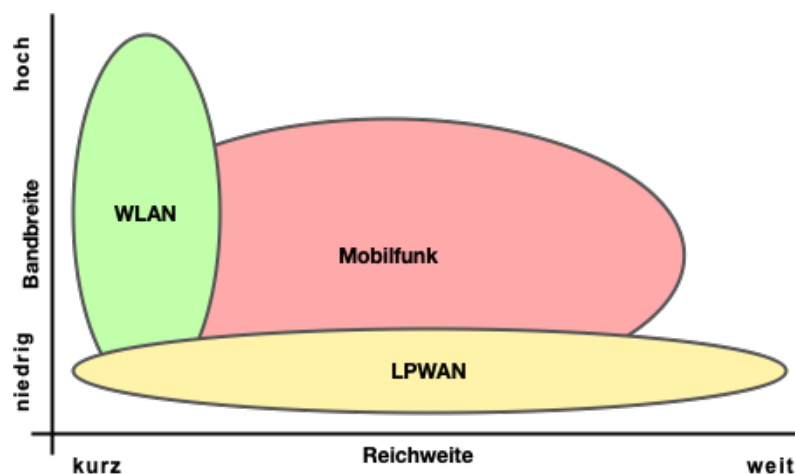


Abbildung 3.1: Vergleich zwischen WLAN, Mobilfunk und LPWAN bezüglich der Bandbreite und Reichweite [kom22]

Die hohe Reichweite, bei gleichzeitig energiesparsamem Betrieb, erzielen die LPWAN Technologien mithilfe von Frequenzen unterhalb des 1 GHz Bereiches. Da LoRa ebenfalls zu den LPWAN Technologien zählt, nutzt diese in Europa die lizenfreien 433 und 868 MHz ISM-Frequenzbänder. Die Frequenzbänder unterscheiden sich je nach Land und Region auf der ganzen Welt. In den USA z.B. liegt der nutzbare Frequenzband bei 915 MHz. Das 433 MHz Frequenzband ist jedoch nur für unidirektionale Übertragung, wo man entweder nur Senden oder Empfangen kann, vorgesehen. Hingegen ist das 868 MHz Band bidirektional, das heißt, dass damit das gleichzeitige Senden und Empfangen gewährleistet ist [kom22].

Die jeweiligen Frequenzbänder haben eine bestimmte Frequenzbandbreite und werden wiederum in sogenannte Funkkanäle unterteilt, mit einer bestimmten Kanalbandbreite, um das gegenseitige stören gleicher Frequenzen zu minimieren. So weist z.B. das 868 MHz Frequenzband eine Frequenzbandbreite zwischen 863 und 870 MHz auf [Sta20]. So können benachbarte Funkknoten, die sich im gleichen 868 MHz Frequenzband befinden, aber einen unterschiedlichen Funkkanal nutzen, gleichzeitig senden und empfangen, ohne sich dabei zu stören.

Darüberhinaus werden weitere regulatorische Maßnahmen ergriffen, wie die Festlegung einer Zeit in Form eines **Duty-Cycles** (DC), welches angibt, wie lange ein Funkknoten auf das Medium pro Tag zugreifen darf. Dieser ist ein prozentualer Wert und liegt bei LoRa normalerweise bei 1% oder 10% je nach Funkkanal und Sendeleistung [Sta20].

Die Abbildung 3.2 zeigt das 868 MHz-Frequenzband mit den jeweiligen Funkkanälen, deren Bandbreite (in kHz), der äquivalenten Strahlungsleistung (in dBm) und des jeweiligen Duty Cycles an .

Weitere Möglichkeiten zur Verhinderung der gegenseitigen Störung sind die definierten Zugriffsverfahren auf das Medium, die bei LPWAN auch als **Polite-Medium-Access** (PMA) heißen. Darunter zählt das sogenannte **Clear-Channel-Assessment** (CCA), welches wiederum in **Adaptive-Frequency-Agility** (AFA) und **Listen-Before-Talk** (LBT) eingeteilt werden kann. Dabei ähnelt das CCA-Zugriffsverfahren sehr stark dem Carrier-Sense-Multiple-Access/Collision-Avoidance (CSMA/CA) Zugriffsverfahren bei z.B. WLAN.

Bei dem AFA Verfahren, wird einerseits die Datenrate hinsichtlich der Kanaleigenschaften angepasst und andererseits ein geeigneter Funkkanal mittels spezieller Algorithmen, die die optimale Lastverteilung zwischen den jeweiligen Funkkanälen



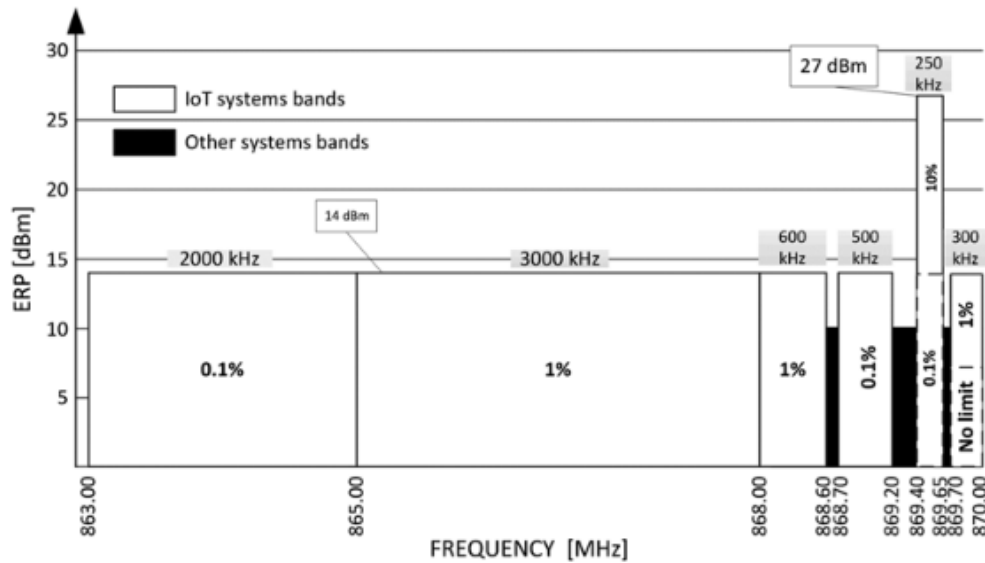


Abbildung 3.2: [Sta20]

berechnen, ausgewählt. Das LBT Verfahren stellt sicher, dass immer nur ein Funkgerät auf einen Funkkanal zugreifen kann, um die gegenseitige Störung zu verhindern. So „lauscht“ (listen) es auf den Funkkanal auf dem es zugreifen möchte zunächst einmal, um festzustellen, ob es frei ist, bevor es darauf zu „sprechen“ (talk) beginnt. Wenn das Funkgerät merkt, dass der Kanal momentan besetzt ist, dann wartet es eine gewisse Zeit, die normalerweise zwischen 5 und 10ms beträgt, ab, bevor es nochmal versucht auf den Kanal zuzugreifen [Sta20].

Die Reichweite von LoRa beträgt zwischen 2-5 km in urbanen und 5-15 km in ländlichen Regionen mit keiner direkten Sichtverbindung (non-line-of-sight). Wenn es eine direkte Sichtverbindung (line-of-sight) zwischen den Funkmasten gibt, dann kann die Reichweite auch weit über 15 km erreichen. Die Datenrate liegt dabei im Bereich zwischen 0.3-5.5 kBit/s und die maximale Übertragungsleistung bei 25 mW [Lie22]. Der aktuelle Rekord, bei dem die Sensorwerte noch empfangen konnten, liegt bei einer unglaublichen Reichweite von 766 km, welcher im Juli 2019 aufgestellt wurde<sup>1</sup>.

Die hohe Reichweite bei gleichzeitig geringem Energieverbrauch ist abgesehen von der niedrigen Frequenz, der speziellen Modulations- bzw. Übertragungsverfahren zu verdanken. LoRa benutzt die sogenannte *Chirp-Spread-Spectrum* (CSS) Modulationstechnik, bei der sich die Frequenz eines Signals (0 oder 1) innerhalb eines definierten Frequenzbereiches gleichmäßig ändert (siehe Abb. 3.3).

<sup>1</sup><https://tech-journal.semtech.com/university-of-zaragoza-breaks-long-range-lorawan-based-signal-record>

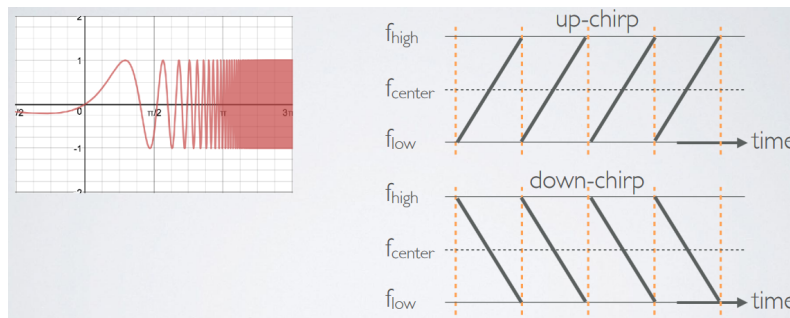


Abbildung 3.3: chirp spread spectrum [lora2020]

Die gleichmäßige Änderung der Frequenz innerhalb eines definierten Frequenzbereiches wird als chirp (zwitschern oder zirpen) bezeichnet und ist in der Natur weit verbreitet. So kommunizieren z.B. Vögel oder Delphine ebenfalls über die zeitliche Änderung der Frequenz. Dies hat den Vorteil, dass äußere Störungen einen geringeren Einfluss auf die Signalqualität ausüben und somit das Signal-Rausch-Verhältnis sich verbessert.

## 3.2 MQTT

...

# Kapitel 4: Praktische Umsetzung

## 4.1 Verwendete Hardware

### 4.1.1 LoPy4-Development-Board

...

### 4.1.2 DHT11/-22 Sensormodul

...

### 4.1.3 Restliche Hardware

...

## 4.2 Beschreibung der Software

Für unser Projekt haben wir drei verschiedene, miteinander interagierende Software Komponenten realisiert, welche über eine Schnittstelle (Interface) miteinander kommunizieren. Der Vorteil einer solchen Architektur ist, dass die einzelnen Komponenten sich unter Umständen wiederverwenden lassen und sich im Idealfall so eine Software modular aufbauen lässt. Da wir als Programmiersprache ausschließlich Python bzw. Micropython verwendet haben, könnte man argumentieren, dass unsere Software automatisch Modular ist, da sich in der Theorie alle programmierten Komponenten in Python wiederverwenden lassen. Dies ist aber sehr verallgemeinert gesprochen, da gerade die Programmierung der Mikrocontroller definitiv auch Individualsoftware benötigt, welche sich aber immerhin nicht nur auf einer einzelnen Mikrocontrollerfamilie ausführen funktionieren würde. Anzumerken ist noch, dass für

unsere finale Version des Projektes vermutlich nur eine einzelne Softwarekomponente notwendig wäre.

### **4.2.1 1. Komponente: Sensoransteuerung und der Versand der Daten mittels LoRa(WAN)**

Für die Programmierung der Mikrocontroller verwenden wir die Programmiersprache Micropython, welche eine schlanke und schnelle Implementation der Programmiersprache Python ist, welche für Mikrocontroller optimiert wurde.

### **4.2.2 2. Komponente: Empfangen der Daten und Versand ins Internet**

...

### **4.2.3 3. Komponente: Manuelles abrufen und versenden der veröffentlichten Daten**

...

## **4.3 Visualisierung der Sensordaten**

...

## **4.4 Berechnung der Laufzeit im Batteriebetrieb**

...

# Kapitel 5: Fazit

## 5.1 Projektauswertung

...

## 5.2 Probleme und Herausforderungen

...

## 5.3 Ausblick

...

## 5.4 Abschlusswort

...



# Anhang A: Anhang

...





# Abbildungsverzeichnis

2.1	Agiles arbeiten mit Scrum . . . . .	6
2.2	Projektstrukturplan . . . . .	7
2.3	Übersicht unserer Zeitplanung . . . . .	8
3.1	Vergleich zwischen WLAN, Mobilfunk und LPWAN bezüglich der Bandbreite und Reichweite . . . . .	11
3.2	Aufteilung des 868-ISM-Bandes in Funkkanäle nach ETSI EN 300 220-2	13
3.3	chirp spread spectrum . . . . .	14



# Tabellenverzeichnis

2.1	Verantwortlichen im Scrum-Team . . . . .	6
2.2	Übersicht der Arbeitspakete und Arbeitszeiten . . . . .	9
2.3	Kostenaufstellung für das Projekt . . . . .	10



# Quelltextverzeichnis



# Literaturverzeichnis

- [BH19] David Bollier und Silke Helfrich. *Frei, fair und lebendig: die Macht der Commons*. Sozialtheorie. Bielefeld: transcript Verlag, 2019, S. 400. ISBN: 978-3-8394-4530-3. DOI: <https://doi.org/10.14361/9783839445303>.
- [kom22] elektronik kompendium. *LPWAN - Low Power Wide Area Network*. <https://www.elektronik-kompendium.de/sites/kom/2207181.htm>. Letzter Zugriff: 22.02.2022. 2022.
- [Lie22] Robert Lie. *LoRa*. <https://lora.readthedocs.io/en/latest/>. Letzter Zugriff: 22.02.2022. 2022.
- [Sta20] Kamil Staniec. *Radio Interfaces in the Internet of Things Systems - Performance studies*. 1. Aufl. Singapore: Springer Nature, 2020. ISBN: 978-3-030-44846-2. DOI: <https://doi.org/10.1007/978-3-030-44846-2>.





# Eigenständigkeitserklärung

Hiermit versichern wir, dass wir das vorliegende Projektabschlussbericht selbstständig und nur unter Verwendung der angegebenen Quellen und Hilfsmittel verfasst haben.

Berlin, den 25.02.2022

Sidney Göhler und Ilja Buschujew