

Pflichtenheft

Wireless Controller for Smart Systems

BACHELOR THESIS - ANKLIN, BOBST, HORATH
25. Februar 2020

Fachcoach:

Matthias Meier
Manuel Di Cerbo

Team:

Raffael Anklin
Robin Bobst
Cyrill Horath

Studiengang:

Elektro- und Informationstechnik

Semester:

Frühlingssemester 2020

Inhaltsverzeichnis

1 Übersicht	1
1.1 Ausgangslage	1
1.2 Ziel der Arbeit	1
2 Lösungskonzept	2
2.1 Punkt zu Punkt Testinfrastruktur	2
2.2 Test Mesh Netzwerke	2
2.2.1 Bluetooth Mesh	2
2.2.2 Thread	2
2.2.3 Zigbee	2
2.3 Steuer und Auswertesoftware	2
3 Projektziele und Lieferobjekte	3
3.1 Punkt zu Punkt Testinfrastruktur	3
3.2 Test Mesh Netzwerke	3
3.3 Steuer und Auswertesoftware	3
3.4 Lieferobjekte	5
4 Projektmanagement	6
4.1 Projektaufteilung	6
4.2 Projektplan	6
4.3 Risikoanalyse	6
4.4 Projektvereinbarung	6
Literatur	8
A Aufgabenstellung	9
B Terminplanung	14
C Risikoanalyse	15

1 Übersicht

Das vorliegende Dokument stellt das Pflichtenheft der Bachelorthesen von Raffael Anklin, Robin Bobst und Cyrill Horath an der Fachhochschule Nordwestschweiz Brugg-Windisch im Studiengang Elektro- und Informationstechnik dar. Im kommenden, ersten Kapitel soll eine Übersicht über die Ausgangslage sowie das Ziel dieser Arbeit gegeben werden und somit die Rahmenbedingungen abgesteckt werden.

1.1 Ausgangslage

Cyrill

Unter den standardisierten Low Power Mesh Netzwerk Protokollen im freien GHz ISM-Band konkurrenzieren sich derzeit vorrangig Bluetooth Mesh, Zigbee sowie Thread. Bezüglich MAC und Physical Layer basieren Zigbee und Thread auf IEEE 802.15.4 wogegen Bluetooth Mesh auf Bluetooth Low Energy (BLE) basiert. Jedes dieser Netzwerkprotokolle hat gewisse Vorzüge: Bluetooth Mesh, dass BLE mittlerweile von jedem Smartphone und Notebook unterstützt wird, Thread aufgrund seiner IPv6 Basis und damit einfachem Übergang ins Internet sowie Zigbee aufgrund seiner etablierten Verbreitung im Smart-Lampenbereich durch Philips, IKEA und Osram. Hauptproblem aller drei Mesh Netzwerkprotokolle ist nebst physikalisch und distanzbedingter Absorption und Reflexion die Störbeeinflussung durch WLAN (WiFi) und andere Netzwerke im GHz Frequenzbereich.

Im Vorprojekt *P5 - Bluetooth-Mesh Plattform für IoT Anwendungen* zu dieser Arbeit wurde das Bluetooth-Mesh Protokoll bereits vertieft betrachtet und dessen Vor- und Nachteile aufgezeigt. Basierend auf diesen Erkenntnissen und der oben beschriebenen Problematik soll das Bluetooth-Mesh Protokoll mit den Alternativen Thread sowie Zigbee verglichen werden.

1.2 Ziel der Arbeit

Cyrill

In der vorliegenden Arbeit soll zuerst ein praxistaugliches, einheitliches Testframework für alle drei Mesh Netzwerke erstellt werden, wonach die Tauglichkeit aller drei Mesh Netzwerke unter realitätsnahen Bedingungen ermittelt und verglichen werden soll. Zwecks besserer Vergleichbarkeit sollen alle drei Testnetze das gleiche Radio-Interface als Grundlage verwenden. Aufgrund der guten Unterstützung aller drei Mesh Protokolle als auch dem im vergangenen P5 gesammelten Wissens, sollen hierfür die nRF52840 SoCs der Firma Nordic eingesetzt werden. Die zu erstellende Testinfrastruktur soll aus den drei folgenden Teilen bestehen:

- Punkt-Punkt Testinfrastrukturen auf MAC-Ebene
- Test Mesh Netzwerke für BT Mesh, Zigbee und Thread
- Steuer- und Auswertesoftware

Die genauen Anforderungen sind einerseits

2 Lösungskonzept

Raffael

Im Zentrum soll die Entwicklung einer *Bluetooth Mesh* Plattform stehen. Diese soll für ein weiterführendes Projekt einsetzbar sein (*Home Automation, Agriculture* oder *Industrie*).

2.1 Punkt zu Punkt Testinfrastruktur

Raffael

2.2 Test Mesh Netzwerke

2.2.1 Bluetooth Mesh

Raffael

2.2.2 Thread

Robin

2.2.3 Zigbee

Cyrill

2.3 Steuer und Auswertesoftware

Raffael

3 Projektziele und Lieferobjekte

Robin

In den beiden Tabellen 3.1 und 3.2 sind die Pflicht- resp. Wunschziele für dieses Projekt festgehalten.

3.1 Punkt zu Punkt Testinfrastruktur

Robin

3.2 Test Mesh Netzwerke

Robin

3.3 Steuer und Auswertesoftware

Robin

Pflichtziele		
Nr.	Ziel	Beschrieb
P1	Bluetooth-Mesh-Netzwerk	Eine variable Anzahl an BLE-Nodes bauen ein Mesh-Netzwerk auf um darin Datenaustausch zu ermöglichen.
P2	UPN	Der Universal-Peripheral-Node kann je nach Einsatz als Sensor oder Aktor konfiguriert und bestückt werden.
P3	Low Power	Die UPN sind bezüglich Hardware und Software energiesparend konzipiert um sie autonom betreiben zu können.
P4	Security	Das Mesh-Netzwerk ist gegen unerlaubten Zugriff und sonstigen Angriffen geschützt.
P5	Netzunabhängig	Durch Versorgung mittels Batterie und Energy-Harvesting können die UPN komplett netzunabhängig betrieben werden.
P6	Energy-Harvesting	Für die Versorgung der UPN werden verschiedene Varianten für das Energy-Harvesting entwickelt. Das Ergebnis wird eine Variantenstudie sein.
P7	Gateway	Zur Konfiguration des Bluetooth-Mesh-Netzwerks steht ein Gateway basierend auf Standard Hardware (Raspberry-Pi + nRF52840 USB Dongle o.ä.) zur Verfügung.
P8	LAN/WLAN	Für die Integration in TCP/IP basierte Systeme bietet der Gateway eine entsprechende Schnittstelle.
P9	CLI	Mittels Command-Line-Interface kann das Mesh-Netzwerk verwaltet werden.

Tabelle 3.1: Pflichtziele

Wunschziele		
Nr.	Ziel	Beschrieb
W1	UPN Konfiguration via Mesh	Einstellungen des UPN können via Mesh Netzwerk angepasst werden und somit z.B. die Peripheriekonfiguration verändert werden.
W2	Firmwareupgrade via Mesh	Die Firmware der UPN wird via Mesh-Netzwerk aktualisiert.
W3	BLR und BLE	Bluetooth Long Range (BLR) und Bluetooth Low Energy (BLE) ergänzen das Bluetooth Mesh um die Reichweite zu vergrößern oder den Energieverbrauch nochmals zu vermindern.
W4	Dedizierte Hardware UPN	Das UPN ist als dedizierte Hardware realisiert und somit einsatzbereit.
W5	Datenschnittstelle	Mittels passender Datenschnittstelle auf dem Gateway können Fremdsysteme wie Apple Homekit, Google Home oder KNX angebunden werden.
W6	Datenschnittstelle ohne Zwischenspeicherung	Damit keine Daten auf dem Gateway zwischen gespeichert werden müssen können die Nodes mittels verbindungslosem Protokoll (MQTT, CoAP, usw.) direkt aus dem Mesh Netzwerk mit einem Fremdsystemen kommunizieren.
W7	HMI	Ein Human-Machine-Interface in Form einer Webapplikation unterstützt den User bei der Konfiguration des Mesh-Netzwerks und ermöglicht die Anbindung an Fremdsysteme.
W8	Dedizierte Gateway Hardware	Der Gateway ist auf einer dedizierten Hardware umgesetzt.
W9	Onboard Bluetooth	Da der Raspberry-Pi 4 bereits ein Bluetooth 5 Chip besitzt soll direkt dieser verwendet werden anstelle eines angeschlossenen Dongles.
W10	Mobiltelefon	Anstelle oder ergänzend zum Gateways kann ein Mobiltelefon ins Mesh-Netzwerk eingebunden werden um Konfigurationen vorzunehmen oder Daten aus zu lesen.
W11	GSM/LTE	Für Feldanwendungen besitzt der Gateway ein GSM/LTE Modul.
W12	Versuchsaufbau Energy-Harvesting	Erfolg versprechende Energy-Harvesting-Systeme werden in einem Versuchsaufbau auf deren Tauglichkeit weiter geprüft.

Tabelle 3.2: Wunschziele

3.4 Lieferobjekte

Robin

Zusätzlich zu den Projektzielen, folgen in diesem Kapitel die Lieferobjekte mit dem jeweiligen Datum. In der Tabelle 3.3 sind diese aufgelistet.

Nr.	Datum	Lieferobjekt
1	07.10.2019	Abgabe Pflichtenheft, 1. Version
2	14.10.2019	Abgabe Pflichtenheft, definitive Version
3	13.01.2020	Projektpräsentation
4	13.01.2020	Abgabe Fachbericht
5	13.01.2020	Abgabe Testaufbau Mesh-Netzwerk

Tabelle 3.3: Lieferobjekte

4 Projektmanagement

Cyrill

Schlankes Projektmanagement mit Projektplan im Anhang. 3 Teile einzeln plus ein Teil gemeinsam.

Alle 2 Wochen soll eine Projektsitzung mit den Dozenten abgehalten werden.

4.1 Projektaufteilung

Cyrill

Evtl. Tabelle mit Definition der Aufteilung. Wer ist für welchen Teil zuständig.

4.2 Projektplan

Framework -> Raffi

Einzelprojekte der jeweils Zuständige

Projektpläne erstellen

Verweis auf die Projektpläne Framework und 3 mal Mesh Netzwerke.

4.3 Risikoanalyse

Robin: Risikoanalyse erstellen und in den Anhang einfügen

4.4 Projektvereinbarung

Projektcoach

Di Cerbo Manuel

Ort, Datum:

Unterschrift:

Meier Matthias

Ort, Datum:

Unterschrift:

Projekt: EIT-P-20FS-030

Anklin Raffael

Ort, Datum:

Unterschrift:

Projekt: EIT-P-20FS-031

Bobst Robin

Ort, Datum:

Projekt: EIT-P-20FS-032

Horath Cyril

Ort, Datum:

Unterschrift:

Unterschrift:

Literatur

- [1] K. Michna, *Entwicklungsgeschichte der Bluetooth-Technologie / Wissen*, de, Jan. 2019. Adresse: <https://www.bluetoothtest.de/wissen/geschichte/> (besucht am 30. Sep. 2019).
- [2] Bluetooth SIG, *Our History*, en-US, Bluetooth, Jan. 2019. Adresse: <https://www.bluetooth.com/about-us/our-history/> (besucht am 30. Sep. 2019).
- [3] M. Eckstein, *Neue Bluetooth-SIG-Gruppe für Mesh-Vernetzung im Smart Home*, de, Jan. 2019. Adresse: <https://www.elektronikpraxis.vogel.de/neue-bluetooth-sig-gruppe-fuer-mesh-vernetzung-im-smart-home-a-789161/> (besucht am 9. Okt. 2019).
- [4] M. Woolley, *An Intro to Bluetooth Mesh Part 1*, en-US, Juli 2017. Adresse: <https://www.bluetooth.com/blog/an-intro-to-bluetooth-mesh-part1/> (besucht am 30. Sep. 2019).

Aufgabenstellung P6 / Thesearbeit FS20

Wireless Controller for Smart Systems

Test und Vergleich von GHz Low Power Mesh Netzwerken

Studierende	Raffael Anklin
	Robin Bobst
	Cyrill Horath
Betreuende Dozenten	Manuel Di Cerbo FHNW Studiengang EIT manuel.dicerbo@fhnw.ch
	Matthias Meier FHNW Studiengang EIT matthias.meier@fhnw.ch
Ausgangslage	<p>Unter den standardisierten Low Power Mesh Netzwerk Protokollen im freien GHz ISM-Band konkurrenzieren sich derzeit vorrangig Bluetooth Mesh, Zigbee sowie Thread.</p> <p>Bezüglich MAC und Physical Layer basieren Zigbee und Thread auf IEEE 802.15.4 wogegen Bluetooth Mesh auf Bluetooth Low Energy (BLE) basiert.</p> <p>Jedes dieser Netzwerkprotokolle hat gewisse Vorzüge: Bluetooth Mesh, dass BLE mittlerweile von jedem Smartphone und Notebook unterstützt wird, Thread aufgrund seiner IPv6 Basis und damit einfachem Übergang ins Internet sowie Zigbee aufgrund seiner etablierten Verbreitung im Smart-Lampenbereich durch Philips, IKEA und Osram.</p> <p>Hauptproblem aller drei Mesh Netzwerkprotokolle ist nebst physikalisch und distanzbedingter Absorption und Reflexion die Störbeeinflussung durch WLAN (WiFi) und andere Netzwerke im GHz Frequenzbereich.</p>

Ziel der Arbeit

In der vorliegenden Arbeit soll zuerst ein praxistaugliches, einheitliches Testframework für alle drei Mesh Netzwerke erstellt werden, wonach die Tauglichkeit aller drei Mesh Netzwerke unter realitätsnahen Bedingungen ermittelt und verglichen werden soll.

Zwecks besserer Vergleichbarkeit sollen alle drei Testnetze das gleiche Radio-Interface als Grundlage verwenden. Aufgrund der guten Unterstützung aller drei Mesh Protokolle als auch dem im vergangenen P5 gesammelten Wissens, sollen hierfür die nRF52840 SoCs der Firma Nordic eingesetzt werden.

Die zu erstellende Testinfrastruktur soll jeweils aus folgenden Teilen bestehen:

1. Zweier **Punkt-Punkt Testinfrastrukturen auf MAC-Ebene** (für BLE und 802.15.4)
Diese sollen es ermöglichen, möglichst einfach zwischen zwei beliebigen Standorten kanalweise die Übertragungsbedingungen zu ermitteln. Diese Testinfrastruktur kann somit einerseits in der Planung eines BT Mesh, Zigbee oder Thread Mesh-Netzwerks als Messinstrument dienen, andererseits auch als Stör-Infrastruktur um gezielt Störungen zu generieren, wie sie von konkurrenzierenden resp. interferierenden BLE und/oder 802.15.4 Netzwerken hervorzurufen werden.
2. Dreier **Test Mesh Netzwerke** (für BT Mesh, Zigbee und Thread)
Mit diesen Test-Netzwerken soll die Robustheit und Mesh-Funktionalität der drei Protokoll-Stacks unter realitätsnahen und nachvollziehbaren Bedingungen ermittelt werden, d.h. bei verschiedenen Netzbelastungen, Netztopologien und gezielt eingebrachter Störungen mit unterschiedlichen Störmustern.
3. Einem Leitrechner, auf welchem eine Python basierte **Steuer- und Auswertesoftware** läuft. Für kürzere Tests und während der Entwicklung kann somit hierfür ein Notebook eingesetzt werden, für Langzeittests hingegen ein Raspberry Pi.
Die Steuer- und Auswertesoftware soll möglichst modular aufgebaut und ein einfaches Commandline-Interface haben, mit Ausgabe der verarbeiteten Daten auf Standard Output.

Zwecks einfacher Bedienung und grafischer Anzeige soll darauf aufbauend ein einfaches Python basiertes Web-Interface realisiert werden.

**Eckpunkte der
Punkt-Punkt
Testinfrastrukturen
auf MAC-Ebene
BLE und 802.15.4**

Beide Varianten sollen konzeptionell wie folgt realisiert werden:

- Der Leitrechner steuert via USB oder UART ein Master-Funkmodul an.
- Das Master-Funkmodul sendet gemäss den Vorgaben des Leitrechners regelmässig MAC-Frames an einen oder mehrere Slave-Funkknoten.
- Der oder die batteriebetriebenen Slave-Funkknoten, bestätigen die MAC-Frames zurück.

Folgende Parameter sollen von der übergeordneten Steuer- und Auswertsoftware konfigurierbar sein:

- Anzahl und ID der Slaves
- Welche BLE resp. 802.15.4 Kanäle zyklisch getestet werden
- Einstellbare Framelänge
- Einstellbare Framerate und Kanalwechselrate
- Einstellbare Sendeleistung
- Nur für BLE: Modulationsart resp. Datenrate (2Mbps ... 125kbps d.h. auch Long Range)
- Nur für 802.15.4: mit/ohne Collision Avoidance (CSMA/CA)

Sowohl master- wie auch slaveseitige Erfassung der Verbindungsqualität (RSSI, Package Loss, Collisions, Noise Level, ...). Die Slaves senden hierzu die erfassten Werte im Rückantwortframe dem Master zurück.

Einfaches und für beide Protokolle (BT und 802.15.4) taugliches Protokoll zwischen Master-Knoten und Auswerterechner.

**Eckpunkte der drei
Mesh Testnetzwerke
(BT Mesh, Zigbee,
Thread)**

Erstellen der drei Mesh-Testnetzwerke mit jeweils ca. 10 Netzknoten wovon:

- Ein Master-Node mit wahlweise USB- oder UART Anschluss zum Auswerterechner
- die restlichen Knoten konfigurierbar z.B. 4 Routing und 5 Low Power Sensor Knoten

Die Sensor-Knoten sollen in einem vom Auswerterechner vorgegebenen parametrisierbaren Intervall Sensorwerte simulieren.

Als "Sensordaten" sollen die Netz-Zustandsdaten übermittelt werden (z.B. Paketnummer zwecks erkennen von verlorenen Datenpaketen, Anzahl Retries, Paketverluste, RSSI, Strombedarf resp. aktive CPU- und Radio-Zeiten, ...)

Das Protokoll und Interface zum Leitreechner soll wenn möglich für alle drei Mesh Netze einheitlich sein.

Um die Störimmunität der Netze zu ermitteln sollen auch gezielt "Fremdstörungen" im eingebracht werden, mit definierbarer Tastung und Störframelänge. Hierfür sollen die "Punkt-Punkt Testinfrastrukturen auf MAC-Ebene" eingesetzt werden.

Umfassende Gegenüberstellung und Validierung aller drei Netzwerke in einem FHNW Gebäude, insbesondere Durchsatz, Antwortzeit, Zuverlässigkeit, Einfachheit der Konfiguration (inkl. Routing), Einfachheit der Ermittlung geeigneter Router-Standorte, Sicherheit und Energieverbrauch, etc.

**Teamaufteilung,
Fachberichte und
Bewertung**

- Jedes Teammitglied realisiert eines der drei Test Mesh Netze.
- Bezüglich Aufteilung der restlichen Arbeiten einigen sich die Teammitglieder untereinander.
- Die Thesisnote setzt sich jeweils hälftig aus einer Individualnote und einer Teamnote zusammen.
- Die Fachberichte resp. Dokumentation setzt sich zusammen aus:
 - Dokumentation der **individuell erstellten Teile**
 - Dokumentation der **gemeinsam erstellten Teile**
 - Einem **Paper** mit Dokumentation und Interpretation der durchgeführten Tests
- Alles Sourcen sollen als Open Source auf GitHub oder GitLab veröffentlicht werden.

Pflichtenheft und Projektvereinbarung

Nach Projektstart soll innert ca. 1 Monat ein technisches Pflichtenheft erarbeitet werden beinhaltend:

- Lösungskonzept und Spezifikation des zu erstellenden Systems,
- Formulierung von Arbeitspaketen (typisch 5-15) und Meilensteinen,
- Zeitplan in Form eines Gantt-Diagrammes.

Projektmanagement, Kommunikation, Abgabetermine, Bewertung:

Das Projekt soll von einem schlanken, ergebnisorientierten Projektmanagement begleitet werden.

Arbeitgeber und betreuender Dozent sollen periodisch (mind. alle 3 Wochen) über den Stand der Arbeiten sowie allfälliger Abweichungen zum Pflichtenheft und Projektplan informiert werden.

Es finden mindestens folgende Meetings statt:

- Kickoffmeeting
- Besprechung Pflichtenheft/Projektvereinbarung
- Schlusspräsentation

Bei Bedarf können mehr Meetings durchgeführt werden.

Betreffend Fachbericht, Ausstellung der Thesearbeit (inkl. Erstellen eines Factsheets und Posters) sowie Verteidigung und Bewertung gelten die Vorgaben und Richtlinien der FHNW, Hochschule für Technik.

Termine

Es gelten die offiziellen Termine der FHNW, Hochschule für Technik

- Die Thesearbeit startet KW 8/2020 und umfasst 360 Arbeitsstunden für jeden Probanden (inkl. Erstellung Fachbericht).
- Der Abgabetermin der Arbeit (Fachbericht) ist am Tag der Bachelor-Ausstellung 14.8.2020.
- Die Verteidigung (d.h. Präsentation und Diskussion der Thesis vor den Betreuern und externem Experten) findet gemäss Semesterplanung der FHNW voraussichtlich in der KW36 oder KW37 statt. Der genaue Termin wird noch bekannt gegeben.

B Terminplanung

Hauptaufgabe	Aufgabe	Anfang	Ende	Wer	Review	30. Sep 19	7. Okt 19	14. Okt 19	21. Okt 19	28. Okt 19	4. Nov 19	11. Nov 19	18. Nov 19	25. Nov 19	2. Dez 19	9. Dez 19	16. Dez 19	23. Dez 19	30. Dez 19	6. Jan 20	13. Jan 20
Fachbericht	Strukturierung / Disposition	04.11.2019	18.11.2019	TBD																	
Fachbericht	Schreiben	18.11.2019	06.01.2020	Alle	Alle																
Fachbericht	Korrigieren / Gegenlesen	06.01.2020	13.01.2020	Alle	Alle																
Fachbericht	Drucken	13.01.2020	13.01.2020	CH																	
Fachbericht	Abgabe Fachbericht	13.01.2020	13.01.2020	Alle																	
Präsetation	Präsentation erstellen	06.01.2020	10.01.2020	Alle																	
Präsetation	Präsentation halten	13.01.2020	13.01.2020	Alle																	
Pflichtenheft	Pflichtenheft verfassen	30.09.2019	14.10.2019	Alle																	
Pflichtenheft	Abgabe Pflichtenheft	14.10.2019	14.10.2019	Alle																	
Vorstudie	Lösungsvarianten recherchieren	14.09.2019	21.10.2019	Alle																	
Vorstudie	Varianten Vergleichen in Tabelle und Entscheidung	07.10.2019	14.10.2019	Alle																	
Energie Harvsting	Berechnungen und Simulationen	07.10.2019	11.11.2019	RA																	
Energie Harvsting	Planen der Testaufbauten	21.10.2019	18.11.2019	RA																	
Energie Harvsting	Material Bestellungen für Testaufbauten	11.11.2019	11.11.2019	RA																	
Energie Harvsting	Testaufbauten fertigstellen	25.11.2019	30.11.2019	RA																	
Energie Harvsting	Messungen Durchführen und Vergleichen	25.11.2019	09.12.2019	RA																	
Energie Harvsting	Dokumentieren und evt. Optimierungen durchführen	09.12.2019	23.12.2019	RA																	
Bluetooth Firmware	IDE aufsetzen	30.09.2019	07.10.2019	BOB																	
Bluetooth Firmware	Mesh Netzwerktest mithilfe Nordic Examples	07.10.2019	21.10.2019	BOB																	
Bluetooth Firmware	Low Power node und Freind Node Energiebedarf messen	14.10.2019	21.10.2019	BOB/RA																	
Bluetooth Firmware	Firmware Remote Node Configuration Konzept erstellen	14.10.2019	28.10.2019	CH/BOB																	
Bluetooth Firmware	Firmware Remote Node Configuration programmieren	21.10.2019	25.11.2019	BOB																	
Gateway	Gateway Aufbauen und Basic Setup durchführen	30.09.2019	07.10.2019	CH																	
Gateway	Mesh Interface mithilfe Example Nordic ausprobieren	07.10.2019	25.10.2019	CH																	
Gateway	Webserver (Node Red) aufsetzen	07.10.2019	25.10.2019	CH																	
Gateway	Programm Remote Configuration Konzept erstellen	14.10.2019	28.10.2019	CH/BOB																	
Gateway	Programm Remote Configuration programmieren	21.10.2019	25.11.2019	CH																	
Hardware uDev	Schema und PCB Layout erstellen	25.11.2019	09.12.2019	tbd																	
Hardware uDev	Bestellung Material und PCB	12.12.2019	23.12.2019	tbd																	
Hardware uDev	Fertigen und Funktionstest	30.12.2019	06.01.2020	tbd																	
Platform Bindings	Anbindungs Konzepte erstellen	25.11.2019	09.12.2019	tbd																	
Platform Bindings	Tests durchführen und Dokumentieren	09.12.2019	06.01.2020	tbd																	
Validierung	Gesamttest Aufbauen (Gateway+Nodes+Energy Harv.)	02.12.2019	16.12.2019	Alle																	
Validierung	Gesamttest durchführen (Gateway+Nodes+Energy Harv.)	16.12.2019	23.12.2019	Alle																	
Validierung	Resultate dokumentieren	23.12.2019	06.01.2020	tbd																	
Validierung	Gesamtsystem für Presentation aufbauen und abgeben	06.01.2020	13.01.2020	Alle																	

C Risikoanalyse

Ereignis				Risiko ohne Massnahme			Prävention	Risiko mit Massnahme			Verantwortlich	Indikator
id	Risiko	Ursachen	Konsequenzen	si	pi	pi		si	pi	pi		
A	Teammitglied fällt kurzfristig aus	Unvorhergesehener Termin, leichte Krankheit, leichter Unfall	Weniger Personalressourcen, kleiner Mehraufwand	3	2	6	Reservezeit einplanen, Transparenter Informationsfluss im Team	1	2	2	CH	Abwesenheit
B	Teammitglied fällt längerfristig aus	Militärdienst, schwere Krankheit, Studienabbruch, schwerer Unfall	Grössere Umplanung, Neuverteilung der Arbeiten	3	2	6	Strukturierte Datenablage, guter Kommunikationsfluss	1	2	2		Abwesenheit
C	Datenverlust oder Zugriffsprobleme	Löschung der Projektdaten, Unzugänglichkeit von Onedrive, keine Internetverbindung	Zugriff auf Daten nicht möglich, Sämtliche Projektdaten nicht mehr vorhanden	2	1	2	Regelmässige Backups, Dokumente zusätzlich lokal abspeichern	1	1	1		Arbeiten auf dem Stand des letzten Backups
D	Software kann nicht mehr ausgeführt werden	Datenverlust, Softwareupdate	Schlimmstenfalls Verlust der gesamten Arbeit, vorübergehende Arbeitspause bis Update komplett	2	3	6	Fertige Softwareteile werden zusätzlich im Onedrive gespeichert (Revisionsverwaltung) / Github	2	1	2		Fehlermeldung
Ereignis	Softwarekonzept nicht ausführbar	Mangelnde Vorkenntnisse, schlechte Planung	Überdenken der Arbeit, Verzug der Arbeiten	2	2	4	Mit Software-Fachcoach besprechen	1	1	1		Nicht funktionierendes Skript
F	Softwareprojekt von Node Gerät und provisioner Gerät nicht verknüpfbar	Schnittstelle wurde nicht korrekt eingehalten	Verzögerung der Arbeit, Mehraufwand	2	2	4	Kommunikation zwischen Softwareteam	1	1	1		Softwareteam können Vorhaben nicht weiterführen
G	Zu komplizierter Sachverhalt	Inhalt kann nicht umgesetzt werden	Stillstand der Arbeit, Projekt nicht durchführbar	1	3	3	Früzeitige Besprechung mit Fachcoaches	1	1	1		Kein Weiterkommen
H	Soziale Spannungen im Team	Meinungsverschiedenheiten, schlechte Arbeitsaufteilung, keine Kompromissbereitschaft	Motivation sinkt, Arbeitsmoral sinkt, schlechte Projektarbeit, unzufriedener Arbeitgeber	2	2	4	Gegenseitige Kontrolle, Fehler offen im Team besprechen, Konstruktive Kritik	2	1	2		Schlechte Arbeitsmoral
I	Mangelnde Kommunikation	zu wenig Sitzungen, Angst vor Demütigung	Schlechtes Zusammenspiel, schlechtere Arbeit	2	1	2	Häufigere Sitzungen, höhere Wertschätzung der einzelnen Teammitglieder	1	1	1		Zurückhaltung
J	Nicht Termingerechte Abgabe der Arbeiten	Faulheit, mangelnder Einsatz, falsche Prioritäten, schlechte Projektführung	Terminplan kann nicht eingehalten werden	2	2	4	striktere Projektführung, gegenseitige Kontrolle, frühzeitiges Melden	2	1	2		Schlechte Arbeitsmoral
K	Qualitativ minderwertige Arbeit	Faulheit, mangelnder Einsatz, schlechter Teamgeist	Mehraufwand, Qualitativ ungenügende Arbeit, Zeitliche Probleme	2	2	4	Gegenkontrolle der Arbeiten	2	1	2		Schlechte Arbeitsmoral
L	Schlechte Terminplanung	Aufwand unterschätzt, keine Reserve eingeplant	Mehraufwand, Überarbeitung des Terminplans, Engpässe	3	2	6	Genug Reservezeit einplanen	2	1	2		Terminverzug

si= Eintrittswahrscheinlichkeit

pi= Auswirkung