

Diplomarbeit

Vertical Farming

ausgeführt an der
Höheren Abteilung für Mechatronik
der Höheren Technischen Lehranstalt Wien 3 Rennweg

im Schuljahr 2018/19

durch

Philipp Gasser
Noah Gruber
Ulrich Obetzhäuser
Stefan Stetina

unter der Anleitung von

Dipl.-Ing. Martin Meschik
Dipl.-Ing. Dr. techn. Ewald Cekan
B.Ed. Mark-Steven Sauerwein

Wien, 05. April 2019

Kurzfassung

Vertical Farming löst die Probleme der Nachfrage an Lebensmitteln im Bereich „Obst & Gemüse“ durch das vertikale Anbauen von Pflanzen. Die vorliegende Diplomarbeit bringt außerdem die Natur und die Nachhaltigkeit in die eigenen 4-Wände, wodurch es jeder Person möglich ist, einen individuellen Obst- oder Gemüsegarten anzulegen. Da das System auch an sonnenarmen oder dunklen Orten nutzen findet, sind Leuchtdioden über den jeweiligen Anbauflächen montiert, welche die Photosynthese und somit das Wachstum der Pflanzen unterstützen. Um auf Erde und damit verbundene Unreinheiten verzichten zu können, hat sich die Bewässerungsart „Hydroponik“ bewährt. Bei dieser Variante nehmen die Pflanzen die Nährstoffe, statt über den Boden, durch eine Nährstofflösung auf. Die Pflanzen-Setzlinge befinden sich in Samenbehältern, welche so ausgelegt sind, dass die Wurzeln der Pflanzen zum Teil mit Wasser bedeckt sind. Um auf dem aktuellen Stand zu bleiben, hat man per Android-App Einsicht auf die Messwerte des Systems. Außerdem kann man in der App die Bewässerung, als auch die Beleuchtung, steuern. Auf einer Standfläche von nur 0,5 m² ist es somit möglich bis zu 36 Pflanzen, beispielsweise Salat, gleichzeitig anzupflanzen.

Abstract

Vertical Farming solves the problems of food demand in the fruit and vegetables sector by growing fruit and vegetables vertically. This diploma thesis also brings nature and sustainability into one's own four walls, allowing each person to create an individual fruit or vegetable garden. Since the system also finds use in sun-poor or dark places, light-emitting diodes are mounted above the respective cultivated areas. These diodes support the photosynthesis and therefore also the growth of the plants. In order to be able to grow plants without soil and associated impurities, the irrigation method "hydroponics" has proven itself. In this variant, the plants absorb the nutrients, rather than over the soil, through a nutrient solution. The plant seedlings are in seed containers, which are designed so that the roots of the plants are partially covered with water. To keep up to date with the Android app, you can take a look at the current information of the system. In addition to that, you can control the irrigation as well as the lighting in the app. With a stand space of only 0.5 m², it is possible to plant up to 36 plants, for example salad, at the same time.

Ehrenwörtliche Erklärung

Ich erkläre an Eides statt, dass ich die individuelle Themenstellung selbstständig und ohne fremde Hilfe verfasst, keine anderen als die angegebenen Quellen und Hilfsmittel benutzt und die den benutzten Quellen wörtlich und inhaltlich entnommenen Stellen als solche erkenntlich gemacht habe.

Wien, am 05. April 2018

Philipp Gasser

Noah Gruber

Ulrich Obetzhauser

Stefan Stetina

Präambel

Die Inhalte dieser Diplomarbeit entsprechen den Qualitätsnormen für „Ingenieurprojekte“ gemäß § 29 der Verordnung des Bundesministers für Unterricht und kulturelle Angelegenheiten über die Reife- und Diplomprüfung in den berufsbildenden höheren Schulen, BGBl. Nr. 847/1992, in der Fassung der Verordnungen BGBl. Nr. 269/1993, Nr. 467/1996 und BGBl. II Nr. 123/97.

Liste der betreuenden Lehrer

wProf, Dipl.-Ing., Martin Meschik Hauptbetreuer

Prof, Dipl.-Ing. techn. Dr., Ewald Cekan Hauptbetreuer Stellvertreter

FL, B.Ed., Mark Sauerwein Betreuer

Liste der Kooperationspartner:

Radio Höndl & Co.

easynname GmbH

Herz As Media & Communications

Pechstahl

PrintShop Sofortdruck- u. HandelsgmbH

Robotunits GmbH

Bilek + Schüll GmbH

Herbeus Greens GmbH

Beta LAYOUT GmbH

Lias Österreich GesmbH

Hornbach Baumarkt GmbH

Rome International GmbH & Co. KG

Inhaltsverzeichnis

Tabellenverzeichnis	19
Abbildungsverzeichnis	21
1 Ziele	27
1.1 Hauptziele	27
1.2 Optionale Ziele	29
1.3 NICHT Ziele.....	30
1.4 Projektteam	31
1.4.1 Betreuer	31
1.4.2 Schüler.....	31
1.5 Individuelle Aufgabenverteilung der Teammitglieder im Gesamtprojekt.....	32
1.5.1 Philipp Gasser.....	32
1.5.2 Stefan Stetina	32
1.5.3 Noah Gruber	33
1.5.4 Ulrich Obetzhauser	33
1.6 Motivation	34
1.6.1 Philipp Gasser.....	34
1.6.2 Stefan Stetina	34
1.6.3 Noah Gruber	35
1.6.4 Ulrich Obetzhauser	35
2 Umsetzung	37
2.1 Projektmanagement	37
2.1.1 Projektidee	37
2.1.1.1 Ausgangssituation	37
2.1.1.2 Was ist die Intention der Diplomarbeit „Vertical Farming“?.....	37
2.1.1.3 Für was steht diese Diplomarbeit?	38
2.1.1.4 Beschreibung der Idee	38
2.1.1.4.1 Konzept 1: Offenes System mit Kapillarbewässerung.....	39
2.1.1.4.2 Konzept 2: Offenes System mit Hydroponik.....	41
2.1.1.4.3 Konzept 3: Geschlossenes System mit Hydroponik	42
2.1.2 Projektmanagement Methode	46

2.1.3	Projektorganisation	48
2.1.3.1	Erklärung der Projektorganisation	48
2.1.3.2	Grafische Darstellung (Empowered Projektorganisation)	49
2.1.3.3	Projektteam	49
2.1.4	Projektmfeldanalyse.....	50
2.1.4.1	Erklärung der Projektmfeldanalyse.....	50
2.1.4.2	Grafische Darstellung	51
2.1.4.3	Beschreibung der wichtigsten Umfelder	52
2.1.5	Risikoanalyse.....	53
2.1.5.1	Erklärung der Risikoanalyse.....	53
2.1.5.2	Risikoportfolio.....	55
2.1.5.3	Beschreibung der wichtigsten Risiken	56
2.1.5.4	Risikogegenmaßnahmen.....	57
2.1.6	Objektstrukturplan.....	58
2.1.6.1	Erklärung des Objektstrukturplans	58
2.1.6.2	Grafische Darstellung	59
2.1.7	Projektstrukturplan	60
2.1.7.1	Erklärung des Projektstrukturplans.....	60
2.1.7.2	Grafische Darstellung	63
2.1.8	Meilensteinplan	64
2.1.8.1	Erklärung des Meilensteinplans.....	64
2.1.8.2	Darstellung der Meilensteine mit SOLL bzw. IST-Terminen	65
2.1.8.2.1	Meilenstein 1: „Projekt grob geplant“	66
2.1.8.2.2	Meilenstein 2: „Projektplanung abgeschlossen“	66
2.1.8.2.3	Meilenstein 3: „Gesamte Einheit konstruiert“	66
2.1.8.2.4	Meilenstein 4: „Platinenlayouts erstellt“	67
2.1.8.2.5	Meilenstein 5: „Blockschaltbild erstellt“	67
2.1.8.2.6	Meilenstein 6: „Tech. Planung abgeschlossen“	67
2.1.8.2.7	Meilenstein 7: „Zwischenpräsentation abgehalten“	68
2.1.8.2.8	Meilenstein 8: „Gesamte Einheit zusammengebaut“	68
2.1.8.2.9	Meilenstein 9: „Android-App erstellt“	68
2.1.8.2.10	Meilenstein 10: „Gesamte Einheit verkabelt“	68
2.1.8.2.11	Meilenstein 11: „Programmcode erstellt“	69
2.1.8.2.12	Meilenstein 12: „Umsetzungsphase abgeschlossen“	69

2.1.8.2.13 Meilenstein 13: „Testphase abgeschlossen“	69
2.1.8.2.14 Meilenstein 14: „Abschlusspräsentation abgehalten“	69
2.1.8.2.15 Meilenstein 15: „Projekt beendet/abgenommen“	70
2.1.9 Personelle Ressourcen	70
2.1.10 Budget	71
2.1.10.1 Geplante Aufwände für Durchführung des Projektes	71
2.1.10.2 Tatsächliche Aufwände für Durchführung des Projektes	72
2.1.10.3 Kostendeckung	73
2.1.11 Externe Kooperationspartner	74
2.1.11.1 Radio Höndl & Co	74
2.1.11.2 easyname GmbH	74
2.1.11.3 Herz As Media & Communications	74
2.1.11.4 Pechstahl	75
2.1.11.5 PrintShop Sofortdruck- u. HandelsgmbH	75
2.1.11.6 Robotunits GmbH	75
2.1.11.7 Bilek + Schüll GmbH	76
2.1.11.8 Herbeus Greens GmbH	76
2.1.11.9 Beta LAYOUT GmbH	76
2.1.11.10 Lias Österreich GesmbH	77
2.1.11.11 HORNBACH Baumarkt GmbH	77
2.1.11.12 Rome International GmbH & Co. KG	77
2.1.12 Marketing	78
2.1.12.1 Website der Diplomarbeit	78
2.1.12.1.1 Bedienung des 3D-CAD Modells	78
2.1.12.2 Folder	81
2.1.12.2.1 Innenseite (mit 1 mm Schnittrand)	82
2.1.12.2.2 Außenseite (mit 1 mm Schnittrand)	83
2.1.12.3 Visitenkarten	84
2.1.12.3.1 Innenseite (mit 1 mm Schnittrand)	84
2.1.12.3.2 Außenseite (mit 1 mm Schnittrand)	84
2.1.12.4 Textilien	85
2.1.12.4.1 T-Shirts	85
2.1.12.4.2 Sweater	85
2.1.13 Instandhaltungskosten der „Vertical Farming“ Einheit	86

2.1.13.1	Kosten bei Verwendung von 1 Etage	86
2.1.13.2	Kosten bei Verwendung von 2 Etagen	87
2.1.13.3	Kosten bei Verwendung von 3 Etagen	88
2.2	Mechanik & Konstruktion	89
2.2.1	Erstes Konzept.....	89
2.2.1.1	Aufbau: Erstes Konzept.....	89
2.2.1.2	Beleuchtung und Elektronik-Führung: Erstes Konzept	89
2.2.1.3	Bewässerung: Erstes Konzept	89
2.2.1.4	Anpflanzen: Erstes Konzept	90
2.2.1.5	Hintergründe für den ersten Konzeptwechsel.....	90
2.2.2	Zweites Konzept	91
2.2.2.1	Aufbau: Zweites Konzept.....	91
2.2.2.2	Beleuchtung und Elektronik-Führung: Zweites Konzept.....	91
2.2.2.3	Entwicklung: Anpflanzmechanismus	92
2.2.2.4	Entwicklung: Samenbehälter	92
2.2.2.5	Bewässerung: Zweites Konzept	92
2.2.2.6	Entscheidung für den zweiten Konzeptwechsel	93
2.2.3	Drittes Konzept	94
2.2.3.1	Aufbau: Drittes Konzept.....	94
2.2.3.2	Bewässerung: Drittes Konzept	94
2.2.3.3	Beleuchtung und Elektronik- Führung Drittes Konzept.....	95
2.2.3.4	Verbesserung des Anpflanzmechanismus	95
2.2.3.5	Verbesserung des Samenbehälters	96
2.2.3.5.1	Säuberungsbürste	96
2.2.4	Korpus	97
2.2.4.1	Korpus Anpflanzeinheit.....	97
2.2.4.1.1	Abstandshalter	97
2.2.4.2	Korpus untere Gehäuseeinheit.....	97
2.2.4.3	Korpus Zusammenbau	98
2.2.5	Beleuchtungsplatte	98
2.2.5.1	Mechanische Bearbeitung Beleuchtungsplatte.....	98
2.2.5.2	Verkabelung Beleuchtungsplatte	99
2.2.6	Anpflanzwanne	99
2.2.6.1	Biegen der Anpflanzwanne.....	99

2.2.6.2	Schweißen und Dichtheitsprüfung der Anpflanzwanne	100
2.2.6.3	Nachbehandlung	100
2.2.7	Drehmechanismus	101
2.2.7.1	Konstruktion Drehmechanismus.....	101
2.2.7.1.1	Sterndeckel	101
2.2.7.1.2	Rotationsdeckel.....	101
2.2.7.1.3	Zusammenbau Drehmechanismus.....	102
2.2.7.2	Funktion & Anpflanzen im Mechanismus.....	103
2.2.7.2.1	Funktion	103
2.2.7.2.2	Anpflanzen	103
2.2.8	Wasserbehälter.....	105
2.2.8.1	Wasserreservoir	105
2.2.8.2	Deckel Wasserreservoir	105
2.2.9	Platinen Gehäuse	106
2.2.9.1	Platinen Gehäuse erste Version.....	106
2.2.9.2	Platinen Gehäuse zweite Version.....	106
2.2.9.3	Gehäusedeckel	107
2.2.9.3.1	Gehäusedeckel erste Version	107
2.2.9.3.2	Gehäusedeckel zweite Version	107
2.3	Elektronik.....	108
2.3.1	Bauteile auf der Platine	108
2.3.1.1	Microcontroller	108
2.3.1.2	Dioden Dimensionierung	108
2.3.1.3	Schaltregler	109
2.3.1.4	Konstantstromquellen.....	109
2.3.2	Sensoren und Aktoren	110
2.3.2.1	Füllstandssensor	110
2.3.2.2	Status LED	110
2.3.2.3	Feuchtigkeits- und Temperatursensor.....	111
2.3.2.4	Lüfter	112
2.3.2.5	Pumpe	112
2.3.2.6	Power LEDs	112
2.3.2.7	Bluetooth-Modul	113
2.3.2.8	Netzteil	114

2.3.3 Konzepte.....	114
2.3.3.1 Platine für High Power LEDs	114
2.3.3.2 Beleuchtungsplatine	116
2.3.3.3 Beleuchtungsplatte	116
2.3.3.4 LEDs mit Sockel verbinden	117
2.3.4 Berechnung LED Kühlung	118
2.3.5 Platinenfertigung	119
2.3.6 Vorteile Schaltregler im Gegensatz zu Spannungsregler.....	121
2.3.7 Berechnung der gesamten Leistung des Gerätes:.....	122
2.3.8 Verkabelung.....	123
2.3.8.1 Beleuchtungsplatte Verkabelung	123
2.3.8.2 Temperatur-/Feuchtigkeitssensorverkabelung	123
2.3.8.3 Status LED Verkabelung	124
2.3.8.4 Netzteil Verkabelung	124
2.4 Programmierung.....	125
2.4.1 Blockschaltbild	125
2.4.2 Flussdiagramm – Mikrocontroller	126
2.4.3 Flussdiagramm - Android App.....	127
2.4.4 Android App	128
2.4.4.1 Erstes Layout der App und Funktionalität (MIT App Inventor).....	128
2.4.4.2 Verbessertes Layout und angepasste Funktionalität der Applikation 131	131
2.4.4.3 Endgültiges Layout der App (Android Studio).....	132
2.4.4.4 Entwicklung der App mit App Inventor.....	133
2.4.4.4.1 Allgemeines über „App Inventor“	133
2.4.4.4.2 Funktion der App	134
2.4.4.5 Entwicklung der App mit Android Studio.....	137
2.4.4.5.1 Allgemeines über das Programm Android Studio.....	137
2.4.4.5.2 Funktion der App	138
2.4.5 Mikrocontroller (PIC16F1827).....	143
2.4.5.1 MPLAB X DIE	143
2.4.5.1.1 Der MPLAB Code Configurator	143
2.4.5.2 Implementieren des Feuchtigkeit-, Temperatur- und Füllstandssensors.....	144
2.4.5.3 Bluetooth Kommunikation zwischen Mikrocontroller und App	145

2.4.5.4	Bewässerung und Beleuchtung der Einheit.....	145
2.4.5.5	Status LEDs mit dazugehöriger Statusdefinition (inkl. Status LED PIC)	148
2.4.5.6	Lüfter	149
3	Lessons Learned.....	151
3.1	Philipp Gasser	151
3.2	Stefan Stetina.....	152
3.3	Ulrich Obetzhauser.....	153
3.4	Noah Gruber.....	154
Anhang A	155
Anhang B	195
Anhang C	207
Anhang D	211
Literaturverzeichnis	239
Quellenverzeichnis	241

Tabellenverzeichnis

Tabelle 1: Projektteam.....	49
Tabelle 2: Beschreibung der wichtigsten Risiken	56
Tabelle 3: Risikogegenmaßnahmen	57
Tabelle 4: Meilensteinliste SOLL-IST Vergleich.....	65
Tabelle 5: Geplante Aufwände	71

Abbildungsverzeichnis

Abbildung 1: Für was steht diese Diplomarbeit?.....	38
Abbildung 2: Aufbau Konzept 1	39
Abbildung 3: Kapillarbewässerung	40
Abbildung 4: Aufbau Konzept 2	41
Abbildung 5: Aufbau Konzept 3	42
Abbildung 6: RobotUnits Profile.....	42
Abbildung 7: 4-Kant Hohlprofile	42
Abbildung 8: Schematischer Aufbau.....	43
Abbildung 9: NFT (Nutrient Film Technique)	43
Abbildung 10: Drehmechanismus.....	44
Abbildung 11: Anpflanzzyklus bei Drehmechanismus	44
Abbildung 12: Projektphasen.....	46
Abbildung 13: Ablauf Projektmanagement.....	47
Abbildung 14: Stamm-/ Linienorganisation	48
Abbildung 15: Empowered Projektorganisation	49
Abbildung 16: Projektumfeldanalyse	51
Abbildung 17: Erklärung der Risikoanalyse	53
Abbildung 18: Risikoportfolio	55
Abbildung 19: Erklärung des Objektstrukturplans.....	58
Abbildung 20: Objektstrukturplan.....	59
Abbildung 21: Projektphasen.....	61
Abbildung 22: Erklärung des Projektstrukturplans	62
Abbildung 23: Projektstrukturplan	63
Abbildung 24: Personelle Ressourcen SOLL-IST Vergleich	70
Abbildung 25: Tatsächliche Aufwände.....	72
Abbildung 26: Kostenaufteilung	73
Abbildung 27: Logo "Höndl & Co."	74
Abbildung 28: Logo "easyname".....	74

Abbildung 29: Logo "Herz As Media"	74
Abbildung 30: Logo "Pechstahl"	75
Abbildung 31: Logo "PrintShop"	75
Abbildung 32: Logo "Robotunits"	75
Abbildung 33: Logo "Bilek + Schüll".....	76
Abbildung 34: Logo "Herbeus Greens".....	76
Abbildung 35: Logo "Beta LAYOUT".....	76
Abbildung 36: Logo "Liapor"	77
Abbildung 37: Logo "Hornbach".....	77
Abbildung 38: Logo "Rome International".....	77
Abbildung 39: Vertical Farming-Einheit auf B2B.Partcommunity	78
Abbildung 40: Aktionsfelder B2B.Partcommunity	78
Abbildung 41: Additional Functions B2B.Partcommunity	79
Abbildung 42: Virtual Reality B2B.Partcommunity	79
Abbildung 43: Model Position B2B.Partcommunity	79
Abbildung 44: Model Shading B2B.Partcommunity	79
Abbildung 45: Shortcuts B2B.Partcommunity	79
Abbildung 46: Environment B2B.Partcommunity	80
Abbildung 47: Aufbau des Folders.....	81
Abbildung 48: Folder "Vertical Farming" Außenseite	82
Abbildung 49: Folder "Vertical Farming" Innenseite	83
Abbildung 50: Visitenkarte "Vertical Farming" Innenseite	84
Abbildung 51: Visitenkarte "Vertical Farming" Außenseite	84
Abbildung 52: T-Shirts "Vertical Farming".....	85
Abbildung 53: Sweater "Vertical Farming"	85
Abbildung 54: Erstes Konzept	89
Abbildung 55: Nährbodenträgerwanne	90
Abbildung 56: Rechnung Eisenwaren Erstes Konzept	90
Abbildung 57: Zweites Konzept	91
Abbildung 58: Drehmechanismus erster Entwurf.....	92
Abbildung 59: Samenbehälter V1	92
Abbildung 60: Rechnung Eisenwaren erstes Konzept.....	93
Abbildung 61: Drittes Konzept	94

Abbildung 62: Hydroponik/ NFT	94
Abbildung 63: Angepasste Beleuchtung	95
Abbildung 64: Verbesserung des Anpflanzmechanismus.....	95
Abbildung 65: Samenbehälter V2	96
Abbildung 66: Säuberungsbürste	96
Abbildung 67: Anpflanzeinheit	97
Abbildung 68: Abstandshalter	97
Abbildung 69: Untere Gehäuseeinheit.....	97
Abbildung 70: Fertiger Korpus	98
Abbildung 71: Beleuchtungsplatte	98
Abbildung 72: Beleuchtungsplatte mit LEDS	99
Abbildung 73: Hydraulische Biegemaschine.....	99
Abbildung 74: Verschweißte Wanne.....	100
Abbildung 75: Nach Nachbehandlung	100
Abbildung 76: Vor Nachbehandlung	100
Abbildung 77: Sterndeckel.....	101
Abbildung 78: Rotationsdeckel	101
Abbildung 79: Schematischer Zusammenbau Drehmechanismus	102
Abbildung 80: Korpus fertig	102
Abbildung 81: Samenbehälter mit Keimling	103
Abbildung 82: Samenbehälter mit vollendetem Blütephase	103
Abbildung 83: Samenbehälter im äußeren Anpflanzkreis	104
Abbildung 84: Erste und zweite Generation im Wachstum	104
Abbildung 85: Füllstandsensor eingebaut.....	105
Abbildung 86: Deckel mit Polokalrohr	105
Abbildung 87: Platinen Gehäuse V1	106
Abbildung 88: Platinen Gehäuse V1	106
Abbildung 89: Platinen Gehäuse V2	106
Abbildung 90: Layout Programm VSLER	107
Abbildung 91: Layout Programm VSLER	107
Abbildung 92: Platinen Gehäuse mit Deckel.....	107
Abbildung 93: Platinen Gehäuse mit Deckel.....	107
Abbildung 94: Microcontroller	108

Abbildung 95: Beschaltung Microcontroller.....	108
Abbildung 96: Schottkydiode	108
Abbildung 97: Schaltregler.....	109
Abbildung 98: Beschaltung Schaltregler.....	109
Abbildung 99: Konstantstromquelle	109
Abbildung 100: Beschaltung Konstantstromquelle	110
Abbildung 101: Füllstandssensor.....	110
Abbildung 102: Beschaltung Füllstandssensor	110
Abbildung 103: Status LED.....	110
Abbildung 104: Beschaltung Status LED	111
Abbildung 105: DHT22 Sensor	111
Abbildung 106: Beschaltung DHT22 Sensor	111
Abbildung 107: Lüfter	112
Abbildung 108: Beschaltung Lüfter.....	112
Abbildung 109: Pumpe	112
Abbildung 110: Beschaltung Pumpe.....	112
Abbildung 111: Power LED.....	112
Abbildung 112: Spektrum Power LED	113
Abbildung 113: HC-06	113
Abbildung 114: Beschaltung Levelshifter.....	113
Abbildung 115: Netzteil.....	114
Abbildung 116: Schaltplan Konzept 1	114
Abbildung 117: Layout Konzept 1	114
Abbildung 118: Lösung 1 Konzept 1	115
Abbildung 119: Lösung 2 Konzept 1	115
Abbildung 120: Schaltplan Konzept 2	116
Abbildung 121: Layout Konzept 2	116
Abbildung 122: Konzept 3.....	116
Abbildung 123: LED Lötofen.....	117
Abbildung 124: Power LED.....	117
Abbildung 125: Schematik Power LED	117
Abbildung 126: Erwärmungsdiagramm.....	118
Abbildung 127: geätzte Platine	119

Abbildung 128: Platinenverbesserung	119
Abbildung 129: Test 1.....	119
Abbildung 130: Platine Vorderseite	120
Abbildung 131: Platine Rückseite	120
Abbildung 132: Platine 3D	120
Abbildung 133: Reglervergleich.....	121
Abbildung 134: Rückseite Beleuchtungsplatte.....	123
Abbildung 135: Duo-LED	124
Abbildung 136: LED Fassung	124
Abbildung 137: Layout nach Starten der App	128
Abbildung 138: Layout nach Öffnen des Drop-Down-Menüs.....	128
Abbildung 139: Layout der Auswahlmöglichkeit "About".....	129
Abbildung 140: Layout der Auswahlmöglichkeit "Beleuchtung"	129
Abbildung 141: Layout der Auswahlmöglichkeit "Bewässerung"	130
Abbildung 142: Layout der Auswahlmöglichkeit "Sensoren".....	130
Abbildung 143: Verbildlichung der Messung des Füllstandes.....	131
Abbildung 144: Layout der gesamten App.....	131
Abbildung 145: Endgültiges App-Layout.....	132

1 Ziele

1.1 Hauptziele

RE-M 1 Projektplanung und Projektcontrolling

Der Erfolg des Projekts oder Produkts hängt auch von der Planung und dem Controlling ab. Als Planungsmethode für das OSP und PSP wird das Programm „NCH Software“ eingesetzt. Mithilfe der Online-Cloud „Microsoft: SharePoint“ wird das Projekt sowohl koordiniert als auch dokumentiert. Zudem finden Meetings statt, um den IST-Zustand mit dem SOLL-Zustand zu vergleichen und um gegebenenfalls Gegenmaßnahmen zu unternehmen.

RE-M 2 Repräsentation des Projekts nach außen

Das Projekt soll mithilfe einer Internetseite vertreten werden. Diese ist nötig, um auf Rückmeldungen und Verbesserungsvorschläge von Interessenten reagieren zu können. Weiters kann diese von möglichen Sponsoren besichtigt werden, um benötigte Informationen abrufen zu können.

RE-M 3 Gesamte Konstruktion modellieren inkl. Thingmark

Um die Konstruktion Interessenten als Simulation zeigen zu können, muss die Einheit (Grundgerüst + Wasserreservoirs + Anpflanzbehälter) mithilfe des Programmes „CREO Parametric 4.0“ modelliert werden. Weiters ist es nötig ein Thingmark zu erstellen, mit welchen man sich das Vertical Farming System mit Hilfe der App „Vuforia View“ an jedem Ort visuell darstellen lassen kann.

RE-M 4 Sponsoring

Um unser Projekt einerseits leichter finanzieren und andererseits hochwertige Komponenten verwenden zu können, ist es nötig Sponsoren zu suchen. Unser Ziel ist es, dass uns Sponsoren mindestens 25% der voraussichtlichen Kosten übernehmen.

RE-M 5 Konstruktion fertigen

Die Konstruktion besteht aus 4 Kant Profilen (Robotunits), welche das Fundament für die gesamte Konstruktion ist. Das Gerüst erhält zusätzlich ein aus welchem bei Bedarf, Wasser zur Anpflanzeinheit gepumpt wird. Die Pumpe, die Platinen, das Wasserreservoir sowie die gesamte Elektronik sollen im Basisbehälter Platz finden.

RE-M 6 1 Anpflanzeinheit fertigen

Die Anpflanzeinheit besteht aus einem Wasserreservoir (Edelstahl), aus einem Rahmen (Aluminium) und der LED Beleuchtung für die Pflanzen. Das Wasserreservoir erhält einen Deckel mit integriertem Drehmechanismus aus Kunststoff. Weiters werden kleine Anpflanzbehälter gefertigt, die in den Deckel des Wasserreservoirs integriert werden. Auf die Oberseite der gesamten Einheit kommt weiters eine Abdeckplatte, welche mit Lüftungsschlitzten versehen ist.

RE-M 7 Beleuchtungsplatine montieren

Die Beleuchtungsplatine soll über der Anpflanzfläche Platz finden.

RE-M 8 Status LEDs im Programmcode festlegen

Die Farbe der LED's symbolisiert einen spezifischen Zustand des Systems. Die grüne LED weist darauf hin, dass das Wasserreservoir ausreichend gefüllt ist. Die gelbe LED weist darauf hin, dass der Wasserspiegel unter dem grünen Bereich liegt. Die rote LED weist darauf hin, dass der Wasserbehälter vollkommen leer ist.

RE-M 9 Bewässerung und Beleuchtung mit Hilfe der Android App steuern

Die Bewässerung und die Beleuchtung werden über eine Bluetooth-App angesteuert.

RE-M 10 Implementieren von Feuchtigkeits-, Temperatur- und Füllstandssensor im Programmcode

Die Feuchtigkeits- und Temperatursensoren messen die Feuchtigkeit und die Temperatur in einer Anpflanzeinheit. Der Füllstandssensor misst den Wasserspiegel im Wasserreservoir.

RE-M 11 Ansteuerung des PIC über Bluetooth-Schnittstelle

Die Bluetooth-Schnittstelle am Mikroprozessor dient dazu, sich via Bluetooth mit der Vertical-Farming-Einheit zu verbinden und diese auch zu steuern

RE-M 12 Android App programmieren

Mit Hilfe der Android-App werden Feuchtigkeit, Temperatur und Füllstand in einer Anpflanzeinheit ausgelesen. Außerdem werden Bewässerung und Beleuchtung ferngesteuert.

RE-M 13 Erstellen der Platinenlayouts

Die Platinenlayouts für die Hauptplatine und Beleuchtungsplatinen werden erstellt.

RE-M 14 Fertigung der Haupt- und Beleuchtungsplatine

Die Platinen werden gefertigt, bestückt und getestet.

RE-M 15 Verkabelung der gesamten Einheit

Die elektronischen Bauteile werden in die Konstruktion eingebaut und vor Inbetriebnahme, sowie während des Betriebs getestet.

RE-M 16 Recherche der benötigten Sensoren

Geeignete Sensoren und Bauteile werden gewählt.

1.2 Optionale Ziele

RE-O 1 Bewerben des Projekts nach außen

Während der Diplomarbeit sollte das Projekt auf verschiedenen Plattformen, wie zum Beispiel Facebook, Instagram und Twitter vertreten werden. Dies wird benötigt, um auf Rückmeldungen und Verbesserungsvorschläge von Interessenten reagieren zu können.

RE-O 2 Werbevideo

Es wird ein Werbevideo erstellt, welches unsere Projektidee repräsentiert. Die Mindestlänge wurde auf 30 Sekunden festgelegt.

RE-O 3 Folder und Visitenkarten

Es werden Folder erstellt damit Interessenten Informationen über unser Projekt erhalten. Weiters werden Visitenkarten für den Projektleiter erstellt, um unseren Sponsoren bzw. Interessenten Kontakt-Informationen hinterlassen zu können.

RE-O 4 Weitere Anpflanzeinheit fertigen

Optional sollen weitere Anpflanzeinheiten gefertigt werden. Geplant sind maximal 3-4 Einheiten damit das ernten noch bequem möglich ist.

RE-O 5 Vollautomatische Bewässerung

Optional wird eine Vertical-Farming-Einheit vollautomatisch bewässert.

RE-O 6 Manuelle Steuerung

Eine manuelle Steuerung wird mittels Taster und Status LEDs verwirklicht.

1.3 NICHT Ziele

RE-N 1 IOS App erstellen

Da es in den meisten Fällen Probleme gibt auch für IOS eine App zu erstellen, wird nur eine Android App für das Vertical-Farming-Projekt erzeugt.

1.4 Projektteam

1.4.1 Betreuer



Hauptbetreuer
DI Martin Meschik
Betreuer von:
Gruber & Obetshauser



Stv. Hauptbetreuer
DI Dr. Ewald Cekan
Betreuer von: Gasser



Nebenbetreuer
Mark Sauerwein
Betreuer von: Stetina

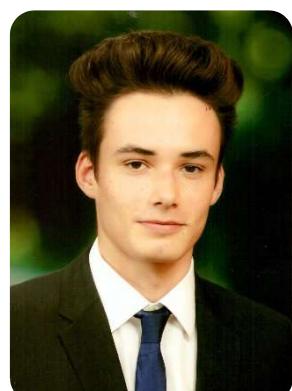
1.4.2 Schüler



Projektleiter
Philipp Gasser



Stv. Projektleiter
Stefan Stetina



Projektmitarbeiter
Ulrich Obetshauser



Projektmitarbeiter
Noah Gruber

1.5 Individuelle Aufgabenverteilung der Teammitglieder im Gesamtprojekt

1.5.1 Philipp Gasser

Themenschwerpunkt	<p>Projektmanagement</p> <p>Als Projektleiter wird Philipp Gasser für die Planung des Projektes, das bedeutet für die Organisation, die Koordination und das Projektcontrolling verantwortlich sein. Im Laufe der Diplomarbeit wird Philipp Gasser für die technische Umsetzung in bestimmten Teilen der Mechanik, den öffentlichen Webauftritt im Sinne einer Internetseite zuständig sein.</p>
Aufgabenstellung Auflistung der einzelnen Ziele und Anforderungen	<ul style="list-style-type: none"> • RE-M 1 Projektplanung und Projektcontrolling • RE-M 2 Repräsentation des Projekts nach außen • RE-M 3 Gesamte Konstruktion modellieren inkl. Thingmark • RE-M 4 Sponsoring • RE-O 1 Bewerben des Projekts nach außen • RE-O 2 Werbevideo • RE-O 3 Folder und Visitenkarten

1.5.2 Stefan Stetina

Themenschwerpunkt	<p>Mechanik</p> <p>Als stellvertretender Projektleiter wird Stefan Stetina den Projektleiter als gut als möglich in beliebigen organisatorischen Bereichen und bei der Koordination innerhalb des Projektes unterstützen. Der Schwerpunkt von Stefan Stetina liegt dennoch in der Mechanik, bei welcher er die Konstruktion berechnet als auch fertigt.</p> <p>.</p>
Aufgabenstellung Auflistung der einzelnen Ziele und Anforderungen	<ul style="list-style-type: none"> • RE-M 5 Konstruktion fertigen • RE-M 6 1 Anpflanzeinheit fertigen • RE-M 7 Beleuchtungsplatine montieren • RE-O 4 Weitere Anpflanzeinheit fertigen

1.5.3 Noah Gruber

Themenschwerpunkt	<p>Programmierung</p> <p>Noah Grubers Aufgabe als Programmierer dieses Teams ist es, den PIC mittels eines Bluetooth-Modules anzusteuern. Weiters wird er sich mit neuen Programmiersprachen auseinandersetzen, um eine benutzerfreundliche App zu gestalten. Mit Hilfe dieser App wird die Bewässerung als auch die Beleuchtung ferngesteuert. Weiters wird er im Programmcode Status LED's festlegen. Durch die verschiedenen Farben der LED's wird jeweils ein anderer Zustand des Systems symbolisiert.</p>
Aufgabenstellung Auflistung der einzelnen Ziele und Anforderungen	<ul style="list-style-type: none"> • RE-M 8 Status LEDs im Programmcode festlegen • RE-M 9 Bewässerung und Beleuchtung mit Hilfe der Android-App steuern • RE-M 10 Implementieren von Feuchtigkeits-, Temperatur- und Füllstandssensoren im Programmcode • RE-M 11 Ansteuerung des PIC über Bluetooth-Schnittstelle • RE-M 12 Android App erstellen • RE-O 5 Vollautomatische Bewässerung • RE-N 1 IOS App erstellen

1.5.4 Ulrich Obetzhauser

Themenschwerpunkt	<p>Elektronik</p> <p>Im Zuge der Diplomarbeit wird Ulrich Obetzhauser im Bereich der technischen Umsetzung mit der Erstellung und der Fertigung passender Platinenlayouts inkl. benötigten Bauelementen beschäftigen. Während des Projektes wird sich Ulrich Obetzhauser mit der Verkabelung der gesamten „Vertical Farming“ Einheit auseinandersetzen.</p>
Aufgabenstellung Auflistung der einzelnen Ziele und Anforderungen	<ul style="list-style-type: none"> • RE-M 13 Erstellen der Platinenlayouts • RE-M 14 Fertigung der Haupt- und Beleuchtungsplatten • RE-M 15 Verkabelung der gesamten Einheit • RE-M 16 Recherche der benötigten Sensoren • RE-O 6 Manuelle Steuerung

1.6 Motivation

1.6.1 Philipp Gasser

Ich war derart entsetzt, als ich bemerkte, dass das Obst und Gemüse in den Supermärkten meist mit chemischen Substanzen, wie Pestiziden oder Herbiziden, behandelt wurde. Nicht nur, dass dies von keinerlei Frische zeigt, sondern auch die immens langen Transportwege belasten die Umwelt und somit die Pflanzen sehr. Da die Transportmittel nicht gerade wenig Treibstoff benötigen, wird die Luft permanent mit reichlich Schadstoffen verschmutzt.

Meine Motivation war es somit, diese Probleme mithilfe meines erlernten Wissens im Bereich des Projektmanagements und der Mechanik zu lösen. Als Projektleiter war es mir außerdem wichtig, diese Diplomarbeit in Zusammenarbeit mit diversen Kooperationspartnern durchzuführen. Nicht nur wegen der Zurverfügungstellung von Kosten oder Materialien, sondern auch wegen dem wichtigen Know-How im Bereich des „Vertical Farmings“.

Ein weiterer wichtiger Aspekt meiner Motivation war es, meine Fähigkeiten und Kompetenzen als Projektleiter zu testen. Für das Team, das Projekt, die Termineinhaltung und etliches verantwortlich zu sein, hat mich sehr motiviert.

Ich möchte das bereits erlangte Wissen mit den Erkenntnissen aus dieser Diplomarbeit verbinden, um somit meinen Horizont im Bereich des Projektmanagements und der Mechanik zu erweitern.

1.6.2 Stefan Stetina

Meine Motivation das Projekt „Vertical Farming“ zu realisieren besteht darin, das erlernte Wissen sowohl aus dem Theorieunterricht als auch der Praxis in die Tat umzusetzen. Die Motivation genau dieses Projekt zu wählen kommt daher, dass das Umweltbewusstsein der heutigen Gesellschaft immer mehr zunimmt doch im Urbanen Raum kein reglicher Platz für den persönlichen Gemüse und Obstanbau herrscht. Da der Wunsch jedoch in der im Menschen verankert ist Eigenanbau zu betreiben habe ich mich mit meinen Kollegen dazu entschlossen diesen in einem Objekt für jedermann zu realisieren.

1.6.3 Noah Gruber

Mit Vertical Farming ist es möglich den Garten von draußen nach drinnen zu holen und verschiedenste Obst- und Gemüsearten in den eigenen vier Wänden anzubauen. Mich persönlich hat dabei das entwickeln einer benutzerfreundlichen App und das Programmieren eines Mikrocontrollers am meisten motiviert. Die Tatsache, dass die Digitalisierung der Welt allgegenwärtig ist und stetig zunimmt hat mich einerseits dazu inspiriert, das Anpflanzen von Obst und Gemüse mit einer einfach gehaltenen Applikation über eine Bluetooth-Schnittstelle fernzusteuern und andererseits dazu, das Überwachen von Temperatur- und Feuchtigkeitswerten, für ein optimales Wachstum der Pflanzen, in einer Anpflanzeinheit zu gewährleisten. Ein zusätzlicher wichtiger Aspekt meiner Motivation war, für mich zuvor unbekannte Gebiete, wie zum Beispiel das Erlernen der Programmiersprachen C und Java, zu erforschen und mich damit aktiv auseinanderzusetzen.

1.6.4 Ulrich Obetzhauser

Meine Motivation das Projekt „Vertical Farming“ zu realisieren besteht darin, das gelernte Wissen sowohl aus dem Theorieunterricht als auch der Praxis in die Tat umzusetzen. Meine Entscheidung genau dieses Projekt zu wählen kommt daher, dass das Umweltbewusstsein der heutigen Gesellschaft immer mehr zunimmt, doch im urbanen Raum kein Platz für die Umsetzung dieses Gedankens in Form eines persönlichen Gemüse- und Obstbaus besteht. Da der Wunsch jedoch im Menschen verankert ist, Eigenanbau zu betreiben, habe ich mich mit meinen Kollegen dazu entschlossen, diesen in einem Objekt für jedermann umsetzbar zu realisieren.

2 Umsetzung

2.1 Projektmanagement

2.1.1 Projektidee

2.1.1.1 Ausgangssituation

Das Projektteam hat sich aufgrund vieler überschneidender Interessensbereiche zu einer gemeinsamen Diplomarbeit, über die Realisierung einer steuerbaren Bepflanzungsanlage mit vertikaler Ausrichtung entschlossen.

Welche Problemstellung gibt es bzw. warum genau „Vertical Farming“?

Aufgrund des zunehmenden Bevölkerungswachstumes, müssen immer mehr Menschen mit Lebensmitteln im Bereich „Obst und Gemüse“ versorgt werden.

Hier gäbe es 4 Möglichkeiten, um dieser Nachfrage gerecht zu werden:

- Die Anbauflächen der Landwirtschaft sollen wesentlich effizienter genutzt werden. Dies ist nicht möglich, da diese bereits zu 80% ausgeschöpft und somit kaum erweiterbar sind.
- Wälder roden, um mehr Platz für diverse Äcker zu schaffen. Da diese Bäume aber lebenswichtigen Sauerstoff ausstoßen, ist dies nicht möglich.
- Den Ertrag der Ernte durch genmodifizierte Pflanzen erhöhen. Es ist unbekannt, wie der menschliche Körper auf solche Mutationen reagieren würde, da es über dieses Thema noch keine wissenswerten Studien gibt.

Die letzte Möglichkeit dieser Nachfrage auf eine ordnungsgemäße Art und Weise gerecht zu werden, ist Vertical Farming.

2.1.1.2 Was ist die Intention der Diplomarbeit „Vertical Farming“?

Dieses Projekt soll den Sinn haben, die Natur und die Nachhaltigkeit in die eigenen vier Wände zu bringen, wodurch es jeder Person möglich sein soll, einen eigenen kleinen Gemüsegarten anlegen zu können.

Weiters wird die Transparenz bei der Herstellung von Obst und Gemüse, wie zum Beispiel Salat unterstützt, wodurch jeder Konsument weiß, dass seine Lebensmittel auf jeden Fall biologisch nachhaltig und frei von jeglicher Art von Pestiziden sowie Herbiziden sind. Man ist auf keine Güter mit langen Transportwegen, sowie auf saisonales Obst und Gemüse angewiesen, da die Anlage an beliebigen Plätzen, nur durch einen gewöhnlichen Stromanschluss betrieben werden kann.

Das aussagekräftigste Argument bleibt dennoch, die Reduktion des Landverzehrs, durch Verringerung der Anbaufläche für die gleiche Menge an Lebensmitteln.

2.1.1.3 Für was steht diese Diplomarbeit?

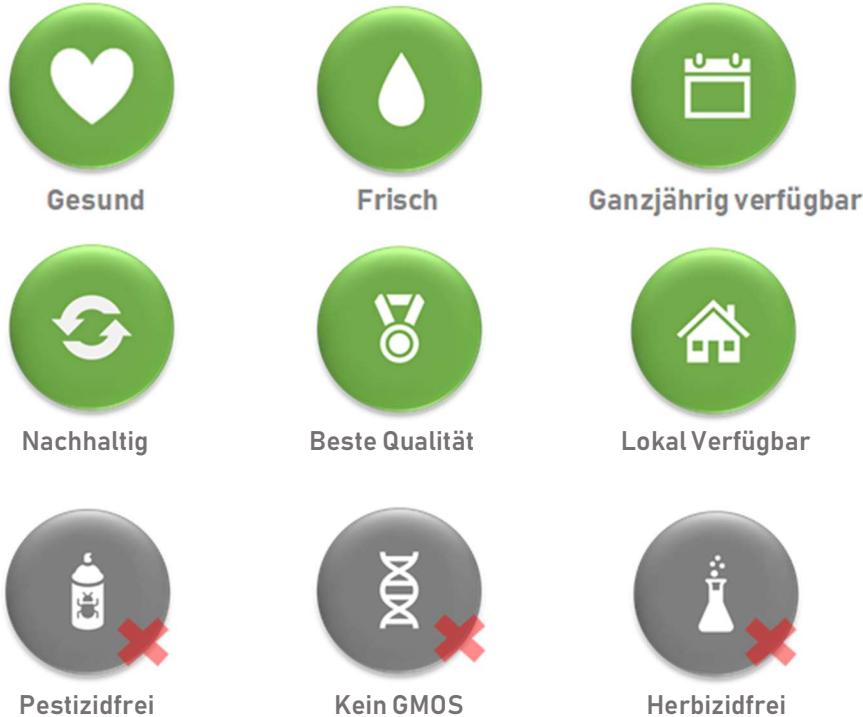


Abbildung 1: Für was steht diese Diplomarbeit?

2.1.1.4 Beschreibung der Idee

Die Idee ist es eine Konstruktion, bestehend aus zumindest einem verstellbaren Anpflanzbehälter, welcher mithilfe einfacher Handgriffe erweitert werden kann, und aus einem Wasserreservoir, zu fertigen. Da das System überall Platz finden soll, kann die Anzahl der Anpflanzbehälter bis auf drei Stück erweitert werden. Um den Nutzer wissenswerte Informationen, wie den Wasserstand, die Temperatur und die Luftfeuchtigkeit anzeigen zu können, sind dementsprechende Sensoren verbaut worden. Außerdem wird die Höhe des Füllstands mithilfe von Status-LEDs direkt am System angezeigt, wobei die Farbe Grün für einen ausreichenden und die Farbe Rot für einen unzureichenden Füllstand steht. Da das System auch an sonnenarmen oder dunklen Orten eingesetzt werden soll, sind Leuchtdioden über den jeweiligen Anbauflächen montiert, welche die Photosynthese der Pflanzen unterstützen. Bei diesen Leuchtdioden handelt es sich um sogenannte „High Power LEDs“, welche eine viel höhere Lichtintensität erzeugen als herkömmliche LEDs. Zudem haben diese LEDs ein Volllichtspektrum, welches auf Pflanzen einen ähnlichen Einfluss hat, wie das Sonnenlicht. In der eigens programmierten Android-App hat der Nutzer Einsicht auf die Messwerte des Systems und kann die Bewässerung als auch die Beleuchtung steuern.

Im Rahmen der Diplomarbeit im Jahrgang 2018/19 wurden neue Bereiche, wie die Kommunikation zwischen verschiedenen Sensoren, die Programmiersprache C für die performante Programmierung des Systems sowie die gezielte und erfolgreiche Vermarktung dieses Projektes erarbeitet.

Um diese Diplomarbeit veranschaulichen zu können, hat sich das Team entschlossen die verschiedenen Versionen dieser Diplomarbeit mithilfe des CAD-Programmes „CREO Parametric 4.0“ zu konstruieren:

2.1.1.4.1 Konzept 1: Offenes System mit Kapillarbewässerung

Das erste Konzept, welches bereits mit Ende des Schuljahres 2017/18 vorlag, unterschied sich maßgeblich von dem tatsächlich umgesetzten Konzept. Bei der Bauform handelt es sich um eine offene Bauform, welche es Krankheiten und Insekten leicht macht, Pflanzen zu infizieren oder zu befallen. Das Grundgerüst bestand aus Vierkant Hohlprofilen und das Wasserreservoir als auch die 3 Anpflanzbehälter aus nichtrostendem Stahl (nichtrostender Stahl). Aufgrund der massiven Bauform wäre ein immens hohes Gewicht zustande gekommen. Außerdem wäre der fertigungstechnische Aufwand erheblich höher gewesen. Dieses Konzept hätte einen geringeren Kostenaufwand benötigt, dennoch wäre die Benutzerfreundlichkeit und die Beleuchtung der Pflanzen nicht gegeben.

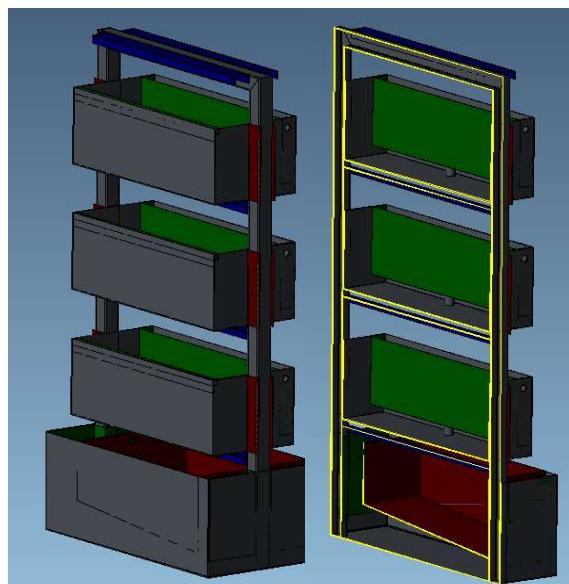


Abbildung 2: Aufbau Konzept 1

2.1.1.4.1.1 Benötigte Materialien

	Stahl	Aluminium	Kunststoff
Verwendung	80%	15%	5%

2.1.1.4.1.2 Bewässerungsmethode

Bei der Bewässerungsmethode des ersten Konzepts handelt es sich um die sogenannte „Kapillarbewässerung“, welches den Pflanzen die Nährstofflösung aus tiefer gelegenen Behältern über Dochte oder Matten ausschließlich durch Kapillarität zuführt. Zur Ergänzung wird eine Pumpe eingesetzt, um das Wasser aus dem Wasserreservoir in die Anpflanzbehälter zu befördern. In diesem Behältern befinden sich Rillen, in welchem sich das Wasser sammelt und mithilfe des Rohres und des kapillaren Effekts zur Pflanze gelangt.

Anhand folgender Grafik kann der Kapillareffekt erklärt werden: Desto geringer der Durchmesser eines Rohres, desto höher ist der Wasserstand. Da der Durchmesser des geplanten Rohres klein genug ist, fließt die Nährstofflösung von dem Zwischenraum, unter den Behältern, direkt in die Behälter. Wenn man hier hingegen die Nährstofflösung direkt in die Behälter pumpt, fließt diese Substanz gleich ab. Mithilfe des Rohres und des Zwischenraumes befindet sich die Nährstofflösung länger im Behälter, da diese nur bis zu einem bestimmten Wasserstand abfließen kann.

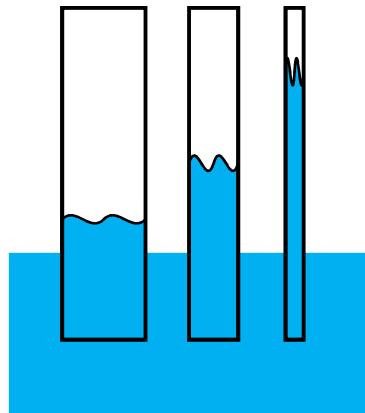


Abbildung 3: Kapillarbewässerung

2.1.1.4.2 Konzept 2: Offenes System mit Hydroponik

Das zweite Konzept dieser Diplomarbeit entsprach schon eher der Letztversion und somit dem Endprodukt. Der gravierende Unterschied war die offene Bauform. Bei einem nicht komplett geschlossenen System bestand das Risiko darin, dass Insekten oder Krankheiten die wachsenden Pflanzen befallen oder anstecken konnten. Da die Konstruktion hier aus Vierkant Hohlprofilen gefertigt worden wäre, wäre zudem ein immenser fertigungstechnischer Aufwand angefallen, welcher außer eventuellen Zeitproblemen auch keine ordnungsgemäße Qualität gewährleisten hätte können. Der Aufbau des Wasserreservoirs und der Anpflanzbehälter weißt hier dennoch schon Ähnlichkeiten mit dem tatsächlich umgesetzten Konzept auf.

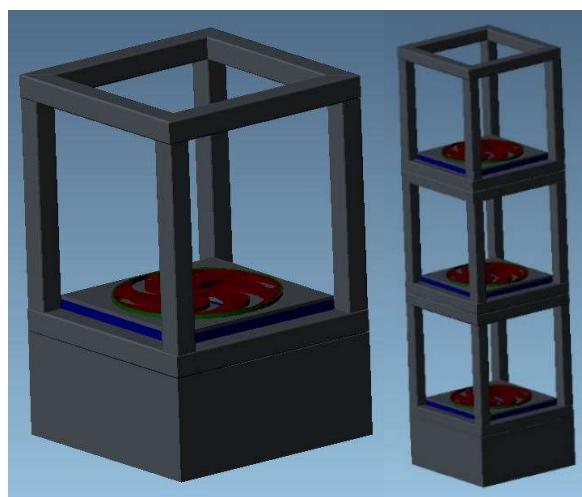


Abbildung 4: Aufbau Konzept 2

2.1.1.4.2.1 Benötigte Materialien

	Aluminium	Kunststoff	Stahl
Verwendung	70%	20%	10%

2.1.1.4.2.2 Bewässerungsmethode

Durch Gespräche mit einigen Sponsoren und Lehrkräften, unter anderem auch mit Unternehmen, welche das vertikale Anbauen von Gemüse wirtschaftlich betreiben, entschied sich das Team eine derzeit durchaus beliebte Bewässerungsmethode namens „Hydroponik“ zu verwenden. Bei Hydroponik handelt es sich um ein erdeloses Kulturverfahren, in welchem Pflanzen die benötigten Nährstoffe aus einer Nährstofflösung extrahieren.

Es gibt einige Arten der „Hydroponik“, doch schlussendlich wurde die sogenannte „NTF-Methode“ (Nutrient Film Technique) verwendet. Bei dieser Art hängen die Wurzeln der Setzlinge in einer Nährstofflösung, währenddessen sich aber der Großteil der Pflanze in der Luft befindet. Durch dies wird gewährleistet, dass die Pflanze mit ausreichend Nährstoffen als auch Sauerstoff versorgt wird.

2.1.1.4.3 Konzept 3: Geschlossenes System mit Hydroponik

Wie bereits erwähnt, ähnelt das tatsächlich umgesetzte Konzept dem zweiten Konzept sehr, mit dem Unterschied, dass die Konstruktion hier eine geschlossene Bauform aufweist. Nach Diskussionen entschloss sich das Projektteam letztendlich doch für dieses Konzept, da der Einsatz von Pestiziden oder Herbiziden nicht ausgeschlossen werden konnte. Außerdem wäre der Zeitaufwand, um die Erweiterbarkeit der Anpflanzbehälter gewährleisten zu können, wesentlich höher gewesen, weshalb die vorherig geplanten Vierkant Hohlprofile durch sogenannte Robotunits-Profile ersetzt wurden. Zudem wären mögliche Änderungen des Projektablaufes, wie die Anpassung der Scharniere der Türen, mit Vierkant Hohlprofilen wesentlich aufwendiger gewesen. Genau dies ist der Vorteil der Robotunits-Profile, da man bei dem Zusammenbau dieser Profile keine Werkzeuge, wie Bohrer oder Gewindeschneider, benötigt. Zudem war es dem Projektteam wichtig, dass auch eine nicht unbedingte Technik affine Person, die Konstruktion aufbauen als auch abbauen können soll. Dennoch sollte auch der Nachteil dieser Entscheidung erwähnt werden, denn die Kosten haben sich immens erhöht, da es sich bei Robotunits-Profilen um qualitativ hochwertige und bereits bearbeitete Halbteile handelt. Das restliche System hat sich aber im Vergleich zu dem zweiten Konzept nicht verändert.

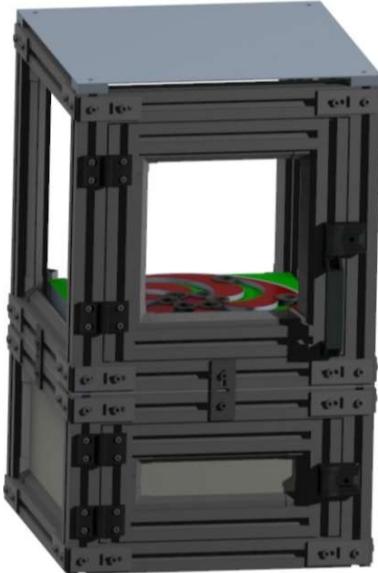


Abbildung 5: Aufbau Konzept 3



Abbildung 7: 4-Kant Hohlprofile

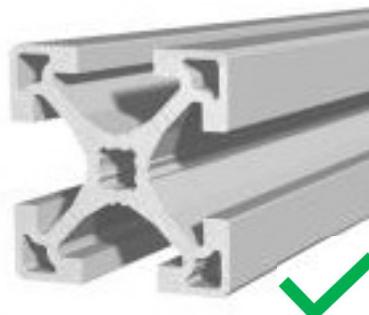


Abbildung 6: RobotUnits Profile

2.1.1.4.3.1 Benötigte Materialien

	Aluminium	Kunststoff	Stahl
Verwendung	60%	30%	10%

2.1.1.4.3.2 Bewässerungsmethode

Die Bewässerungsmethode der umgesetzten Variante entspricht der des zweiten Konzepts. Wie schon erwähnt entschloss sich das Team für die sogenannte Bewässerungsmethode „Hydroponik“, bei welcher auf jegliche Erde verzichtet werden konnte. Letztendlich wurde die „NFT-Methode“ gewählt, bei welcher die Wurzeln der Pflanzen in einer Nährstofflösung hängen, wobei sich der Rest der Pflanze in der Luft befindet. Diese Kombination sorgt für eine ausreichende Versorgung der Pflanzen mit Nährstoffen und Sauerstoff. Bei diesem Bewässerungsverfahren wird die Nährstofflösung vom Wasserreservoir in den Anpflanzbehälter, in welchem sich die Setzlinge befinden, gepumpt. Die Nährstofflösung wird über die Wurzeln der Pflanze aufgenommen, wodurch der Füllstand im Anpflanzbehälter wieder sinkt. Nach einer gewissen Zeit muss die Nährstofflösung erneut hochgepumpt werden, damit die Pflanzen nicht „ersticken“. Eine automatische Bewässerung ist hier eher riskant, da das Zeitintervall des Hochpumpens von Pflanzenart zu Pflanzenart variiert und somit das Risiko besteht, dass alle angesetzten Pflanzen „ertrinken“ können.

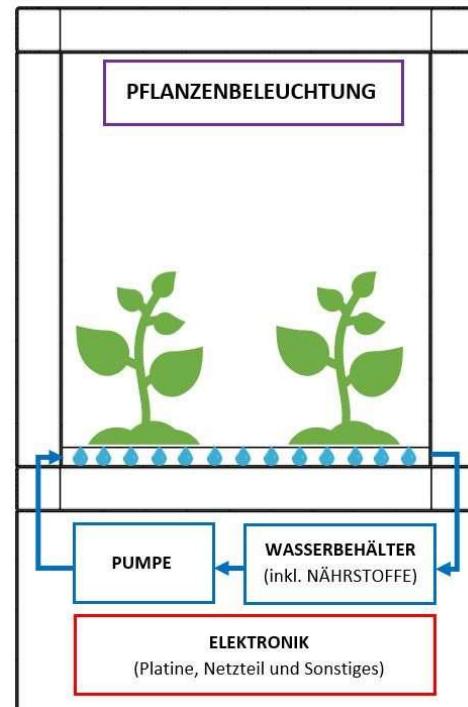


Abbildung 8: Schematischer Aufbau

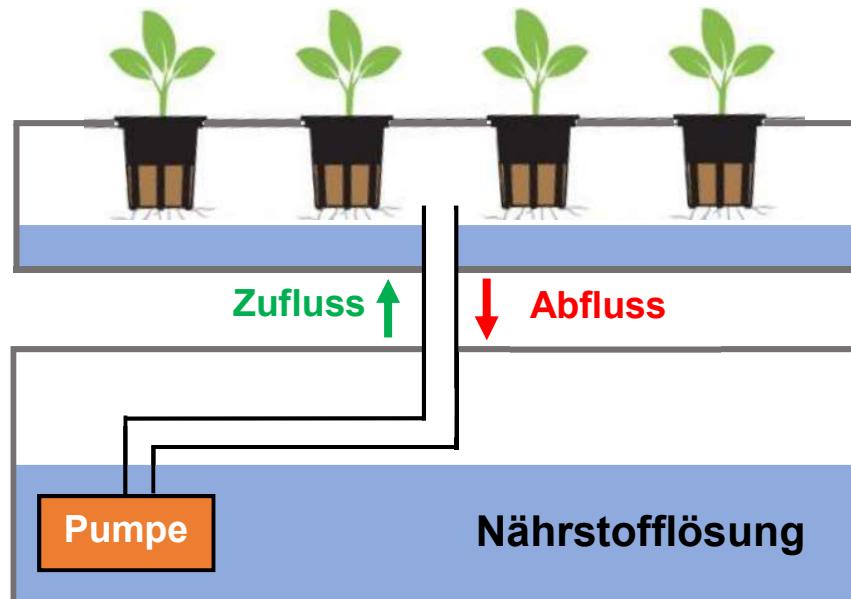


Abbildung 9: NFT (Nutrient Film Technique)

2.1.1.4.3.3 Drehmechanismus

Welchen Zweck hat der Drehmechanismus?

Er vereinfacht das Ernten, da man die bereits erntefertigen Salatköpfe durch eine Drehung von 60° gegen den Uhrzeigersinn nach außen drehen kann.

Um den Mechanismus aber ordnungsgemäß nutzen zu können, beginnt man zuerst die innere Setzling-Reihe zu bestücken.

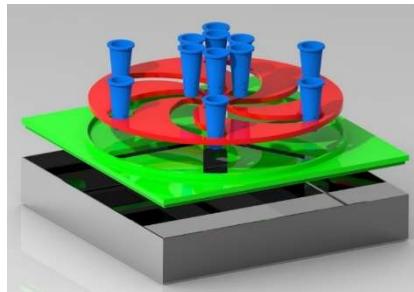


Abbildung 10: Drehmechanismus

Nach der Hälfte des Wachstums kann man die Pflanzen nach außen drehen und auf den inneren 6 „Löchern“ neue, frische Setzlinge einstecken.

Wenn man hierbei wieder ein halbes Wachstum wartet, also die äußeren Pflanzen erntereif sind, kann man diese ernten – die vorherig frischen Setzlinge werden nach außen geschoben und innen anschließend neue Anpflanzbehälter eingesteckt.

Um sich diese textbasierte Erklärung auch visuell vorstellen zu können, wurden folgende Grafiken eingefügt:

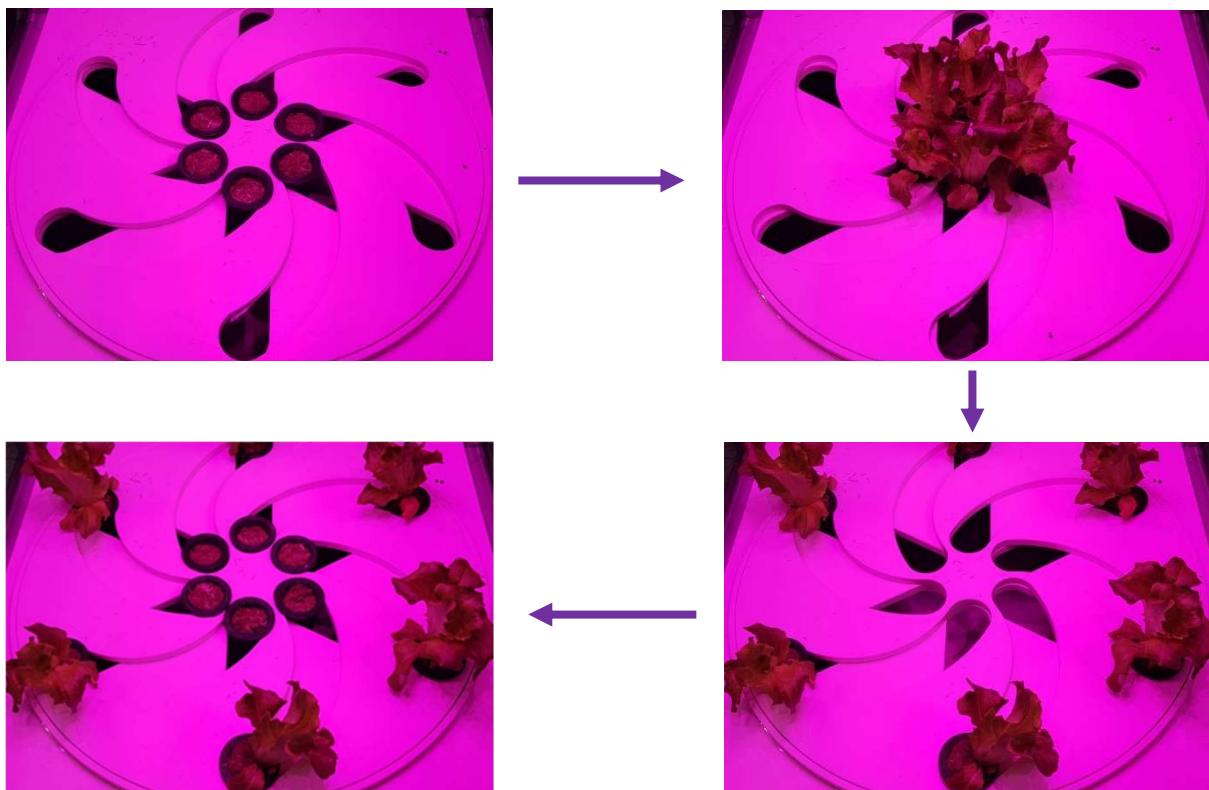


Abbildung 11: Anpflanzzyklus bei Drehmechanismus

Welche Vorteile bietet dieser Mechanismus?

- *Geringer Platzbedarf*

Die Größe des Drehmechanismus wurde ideal auf die maximal benutzbare Anbaufläche ausgelegt.

- *Einfaches Wechseln der Pflanzensorte*

Durch die vorhergesehenen Samenbehälter ist es dem Konsumenten möglich, die Sorte der anzubauenden Pflanze, ohne jegliche Umstände zu wechseln. Beispielweise kann man auf jeder Etage und somit in jedem Behälter eine andere Pflanzensorte anbauen.

- *Verkürzte Wartezeit bis zur Ernte*

Bei einer Verwendung von drei Etagen können drei individuelle Zyklen angegeben werden. Das bedeutet der nächste Zyklus wird jeweils 1/3 nach dem vorherigen Zyklus gesetzt. Wenn sich der Konsument hier zum Beispiel entscheidet Salat anzubauen, wird die durchschnittliche Dauer bis zur Ernte von 30 Tagen auf 10 Tage verkürzt. Das heißt, dass es dem Nutzer möglich ist alle 10 Tage 6 Salate zu ernten.

- *36 Pflanzen auf 0,5 m² Fläche (Drei Etagen)*

Bei einer Verwendung der Vertical-Farming Einheit mit drei Etagen ist es dem Nutzer möglich bis zu 36 Pflanzen auf einer Fläche von 0,5 m² gleichzeitig anzubauen.

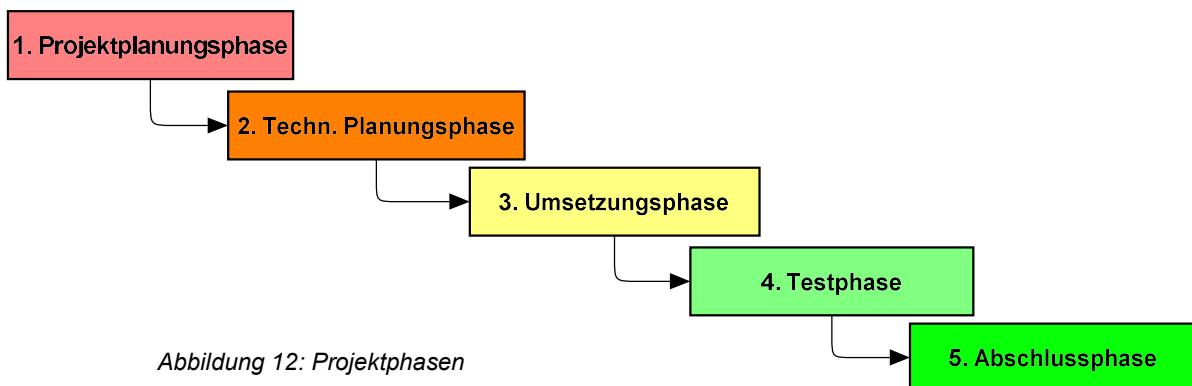
Vorteile von „Vertical Farming“:

- Die Transparenz bei der Herstellung von Obst und Gemüse wird unterstützt, wodurch jeder Konsument weiß, dass seine Lebensmittel auf jeden Fall biologisch nachhaltig und frei von jeglicher Art von Pesti- sowie Herbiziden sind.
- Konsumenten sind nicht mehr auf Güter mit langen Transportwegen und auf saisonales Obst bzw. Gemüse angewiesen.
- Die Reduktion des Landverzehrs, durch Verringerung der Anbaufläche für die gleiche Menge an Lebensmitteln.
- Die Einheit findet an jedem beliebigen Ort Platz, an welchem ein gewöhnlicher Stromanschluss zur Verfügung steht.

2.1.2 Projektmanagement Methode

Bei der Diplomarbeit „Vertical Farming“ wurde die „Wasserfall-Methode“ als Projektmanagement Methode gewählt. Mit dieser Methode wurde das Projekt in verschiedene Phasen bzw. Stufen eingeteilt. Diese Phasen werden am Anfang des Projektstarts, also somit vor Beginn der Umsetzung, definiert.

Anfangs befand sich das Projektteam in der Projektplanungs-, dann in der technischen Planungs-, anschließend in der Umsetzungs-, danach in der Test- und letztlich in der Abschlussphase. Diese Phasen werden auch im PSP, im sogenannten Projektstrukturplan, wiedergefunden. Der Vorteil dieser Projektmanagement Methode ist, dass die Planungssicherheit immens gesteigert wird. Als Folge konnte das Projekt sehr umfangreich als auch präzise geplant werden.



Spontane Änderungen können den Projektablauf gefährden und die Flexibilität des Projektes einschränken. Hierbei würde sich die „agile Vorgehensweise“, beispielsweise in Form eines SCRUM-Modells als vorteilhafter beweisen. Bei dieser Projektmanagement Methode wird das Projekt nicht anhand eines langfristigen, bereits im Vorhinein erstellten Plans, sondern mit Hilfe sogenannter Sprints durchgeführt. Diese Sprints entsprechen mehr oder weniger kurzen Bearbeitungszyklen, in welchen jeweils ein oder mehrere Themenbereiche bearbeitet, getestet und abgeschlossen werden. Jedoch ist es bei der „agilen Vorgehensweise“ im Vergleich zu dem „Wasserfall-Modell“ schwieriger, die Ergebnisse am Ende eines jeden Sprints abzuschätzen. Aus diesem Grund ergeben sich bei der „agilen Vorgehensweise“ gewisse Einschränkungen im Bereich der Planungssicherheit.

Um diesen Nachteil entgegenzuwirken ist es notwendig, das Projekt bereits im Vorhinein gut und detailliert zu strukturieren. Auch wenn der Konzeptionsaufwand relativ hoch war bzw. teils sehr detaillierte Schritte lange Zeit im Voraus geplant werden mussten, hat sich die gewählte Projektmanagement Methode durchaus bewährt. Schwierig war es hierbei, das gesamte Projekt und die damit verbundenen Prozesse im Auge zu behalten, um mögliche Veränderungen bereits im Voraus zu erkennen.

Um diesen teils, nicht ausschließbaren Änderungen entgegenzuwirken wurde unter anderem eine Projektumfeldanalyse, eine Risikoanalyse, ein Objektstrukturplan, ein Projektstrukturplan und eine Meilensteinliste erstellt. Zusätzlich wurden die Projektressourcen, die personellen Ressourcen und ein Kostenplan erstellt, um den Arbeits- als auch den Kostenaufwand abschätzen zu können.

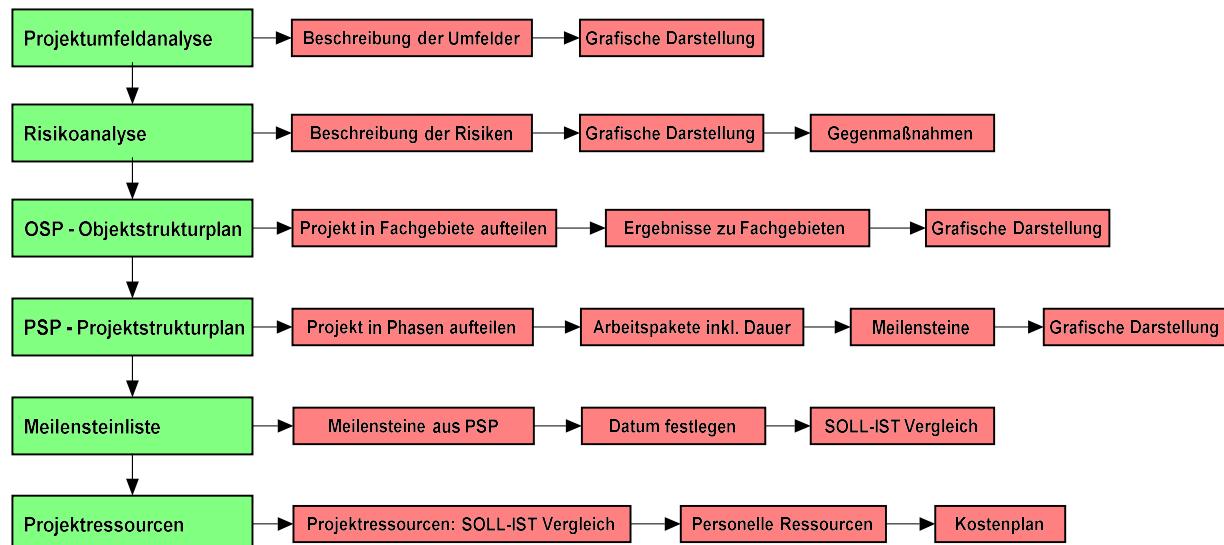


Abbildung 13: Ablauf Projektmanagement

Mit Hilfe obiger Ansicht kann der Ablauf der Projektplanung relativ gut erläutert werden. Zu Allererst wurde eine Projektumfeldanalyse erstellt, in welcher alle positiven als auch negativen Umfelder, wie Schüler, Lehrer oder wichtige Einflussfaktoren, wie Sensoren oder Rohmaterialien definiert, beschrieben und grafisch dargestellt wurden.

Danach wurde eine Risikoanalyse durchgeführt, in welcher alle beliebigen Risiken vermerkt, beschrieben als auch grafisch dargestellt wurden. Weiters wurden dazugehörige Gegenmaßnahmen hinzugefügt.

Anschließend wurde ein sogenannter Objektstrukturplan erstellt, in welchem die jeweiligen Ergebnisse, wie Grundgerüst oder Wasserreservoir, den dazugehörigen Fachgebieten, wie Konstruktion, zugewiesen wurde. Dieser Plan wurde ebenfalls als grafische Darstellung verwirklicht.

Als Folge aus dem Objektstrukturplan wurde der Projektstrukturplan entworfen, in welchem die benötigten Arbeitspakete den dazugehörigen Projektphasen zugeteilt wurden. Basierend auf den Arbeitspaketen wurden hier einige essenzielle Meilensteine definiert. Der Plan wurde ebenso grafisch dargestellt.

Infolgedessen wurde eine Meilensteinliste erstellt, in welcher alle bereits definierten Meilensteine mit einem SOLL-Datum versehen wurden. Während des gesamten Projektes wurden diese Meilensteine mithilfe eines SOLL-IST Vergleiches kontrolliert.

Abschließend wurden die benötigten Projektressourcen und die personellen Ressourcen eruiert sowie ein Kostenplan erstellt.

2.1.3 Projektorganisation

2.1.3.1 Erklärung der Projektorganisation

Jedes Projekt, jedes Unternehmen und jede Organisation hat eine bestimmte Struktur, welche normalerweise als Stammorganisation oder Linienorganisation bezeichnet wird. In dieser sind die Abteilungen eines Projektes hierarchisch in Abhängigkeit gegliedert, wie man in folgender Abbildung erkennen kann:

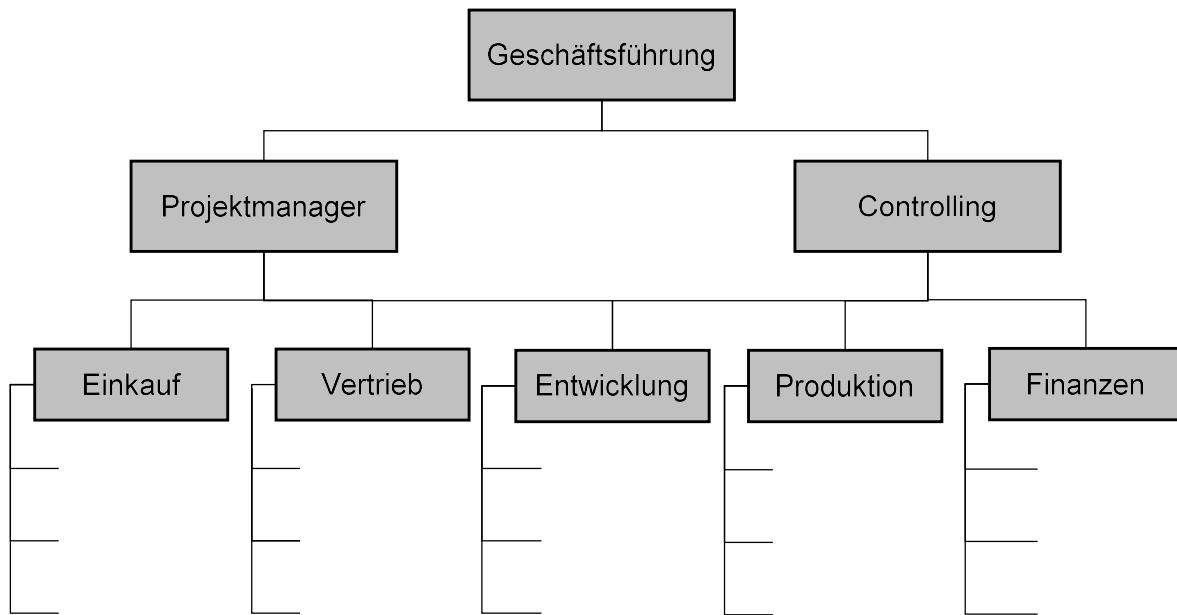


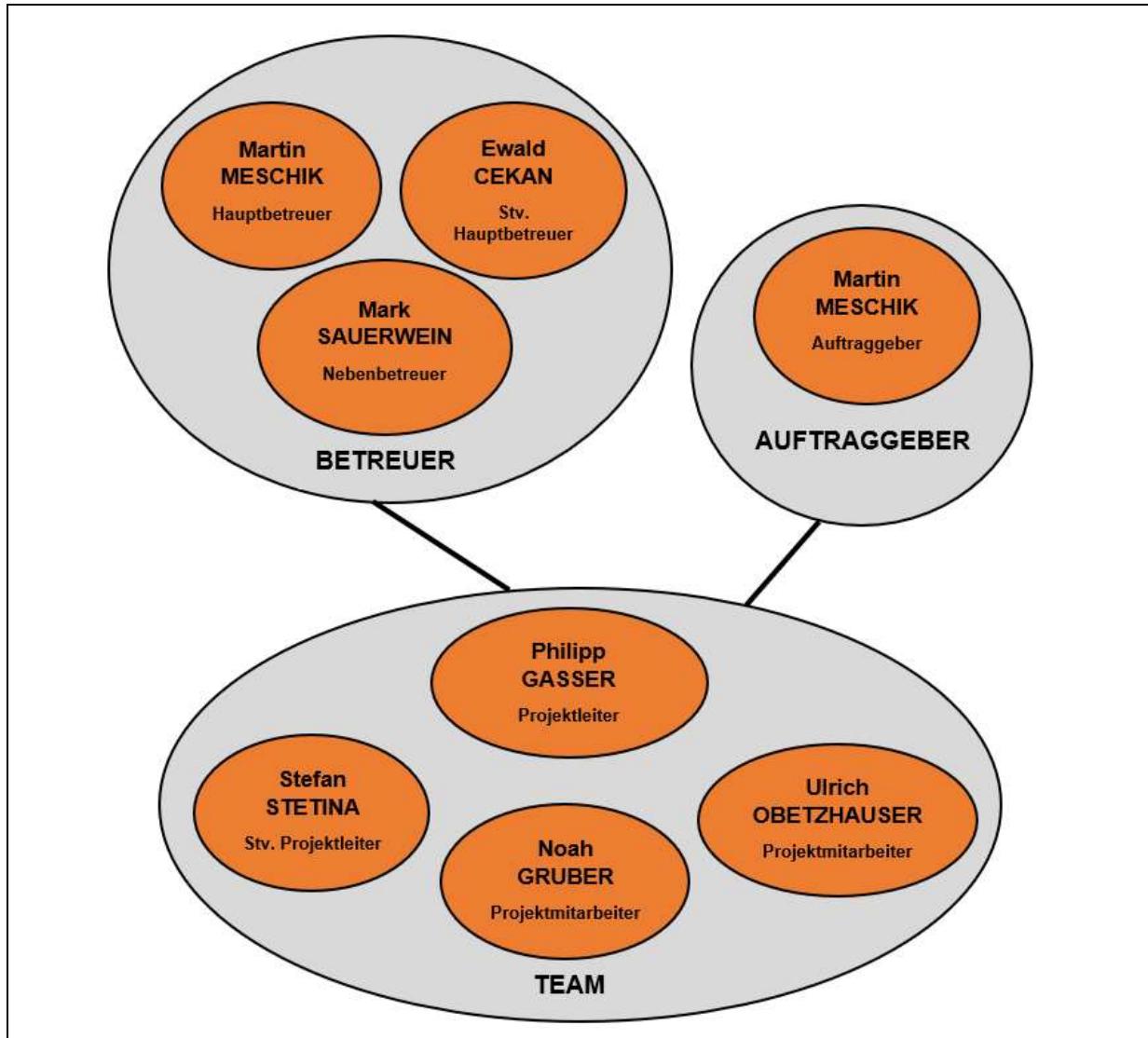
Abbildung 14: Stamm-/ Linienorganisation

Zu beachten ist hierbei, dass sich in einer Stamm- oder Linienorganisation meist nur die Abteilungen, ohne die dazugehörigen Abteilungsleiter oder Mitglieder, befinden.

Aus diesem Grund wird zusätzlich zur Stamm- oder Linienorganisation auch eine Projektorganisation gebildet, welche die Zusammenstellung der Rollen der Projektmitglieder, die Befugnisse und die Schnittstellen eines Projektes kennzeichnet. Bei großen Projekten, in denen viele Personen mitwirken, ist eine solche Projektorganisation essenziell. Die Aufgabenverteilung als auch der Projekterfolg können nur gewährleistet werden, wenn auch alle beteiligten Personen wissen, welche Aufgaben, Befugnisse und Verantwortungen sie innerhalb eines Projektes haben. Das bedeutet sie können sich nur richtig also dementsprechend verhalten, wenn eine Projektorganisation gebildet wurde.

Bei dieser Diplomarbeit wurde aber nur eine Projektorganisation durchgeführt, da das Projektteam lediglich aus 7 Mitgliedern bestand und eine Stammorganisation bzw. Linienorganisation somit überflüssig gewesen wäre. Zudem wurde schon vorab geklärt, welcher Mitarbeiter für welches Themengebiet zuständig ist.

2.1.3.2 Grafische Darstellung (Empowered Projektorganisation)



2.1.3.3 Projektteam

Funktion	Name	Kürzel	E-Mail
PA	Martin MESCHIK	MSK	msk@htl.rennweg.at
PL	Philipp GASSER	4047	4047@htl.rennweg.at
PL Stv.	Stefan STETINA	4070	4070@htl.rennweg.at
PTM	Noah GRUBER	4048	4048@htl.rennweg.at
PTM	Ulrich OBETZHAUSER	4060	4060@htl.rennweg.at

Tabelle 1: Projektteam

2.1.4 Projektumfeldanalyse

2.1.4.1 Erklärung der Projektumfeldanalyse

Als eine Projektumfeldanalyse bezeichnet man die Erfassung, Untersuchung und Dokumentation der äußereren Einflussfaktoren auf das zu betrachtende Projekt. Die Ziele einer solchen Analyse sind folgende:

- Erkennung aller Einflussfaktoren für das Projekt
- Erfassung aller Interessensgruppen am Projekt
- Identifizierung von positiven und negativen Projekteinflüssen (sowohl Bedrohungen als auch Chancen)
- Vorbereiten von Gegenmaßnahmen zur Beeinflussung des Projektumfelds
- Verschriftlichung und Verbildlichung dieser Erkenntnisse für Projektplanung

Zu allererst wurden die Umfelder erfasst und deren Einfluss auf das Projekt identifiziert. Anschließend wurden diese Umfelder mit Nummern versehen und entweder negativ und/oder positiv bewertet. Daraufhin wurde zu jedem Umfeld eine kurze Beschreibung hinzugefügt, welche den Einfluss beziehungsweise das Risiko oder die Chance noch einmal beschreibt. Zu guter Letzt wurden diese erarbeiteten Informationen mithilfe einer grafischen Darstellung verbildlicht.

Anhand der folgenden Abbildung, welche sich auf der nächsten Seite befindet, lässt sich eine Projektumfeldanalyse gut beschreiben: Auf der linken oberen Seite befinden sich die „internen Umfelder“, wie den Projektleiter, die Projektmitarbeiter, die Betreuer und dergleichen. Wobei sich hierbei auf der linken unteren Seite die „externen Umfelder“, wie Know-How oder der Verlauf des Schuljahres. Als dritten und letzten Oberpunkt gibt es die „technischen Umfelder“, wie zum Beispiel Rohmaterial, SharePoint oder Sensoren.

Diese Umfelder befinden sich auf der rechten Seite der Projektumfeldanalyse. Positive Umfelder werden mit einem „+“ und negative Umfelder mit einem „-“ markiert.

Wichtig ist es zudem, während des gesamten Projektverlaufs das Projektumfeld im Auge zu behalten, um auf unvorhersehbare Veränderungen frühzeitig reagieren zu können. Da es sich bei der Diplomarbeit „Vertical Farming“ um ein Entwicklungsprojekt handelt, war das Risiko einer Veränderung niedriger, als beispielsweise bei einem Bauprojekt, bei welchem Wetterbedingungen einen sehr hohen Einfluss haben können.

2.1.4.2 Grafische Darstellung

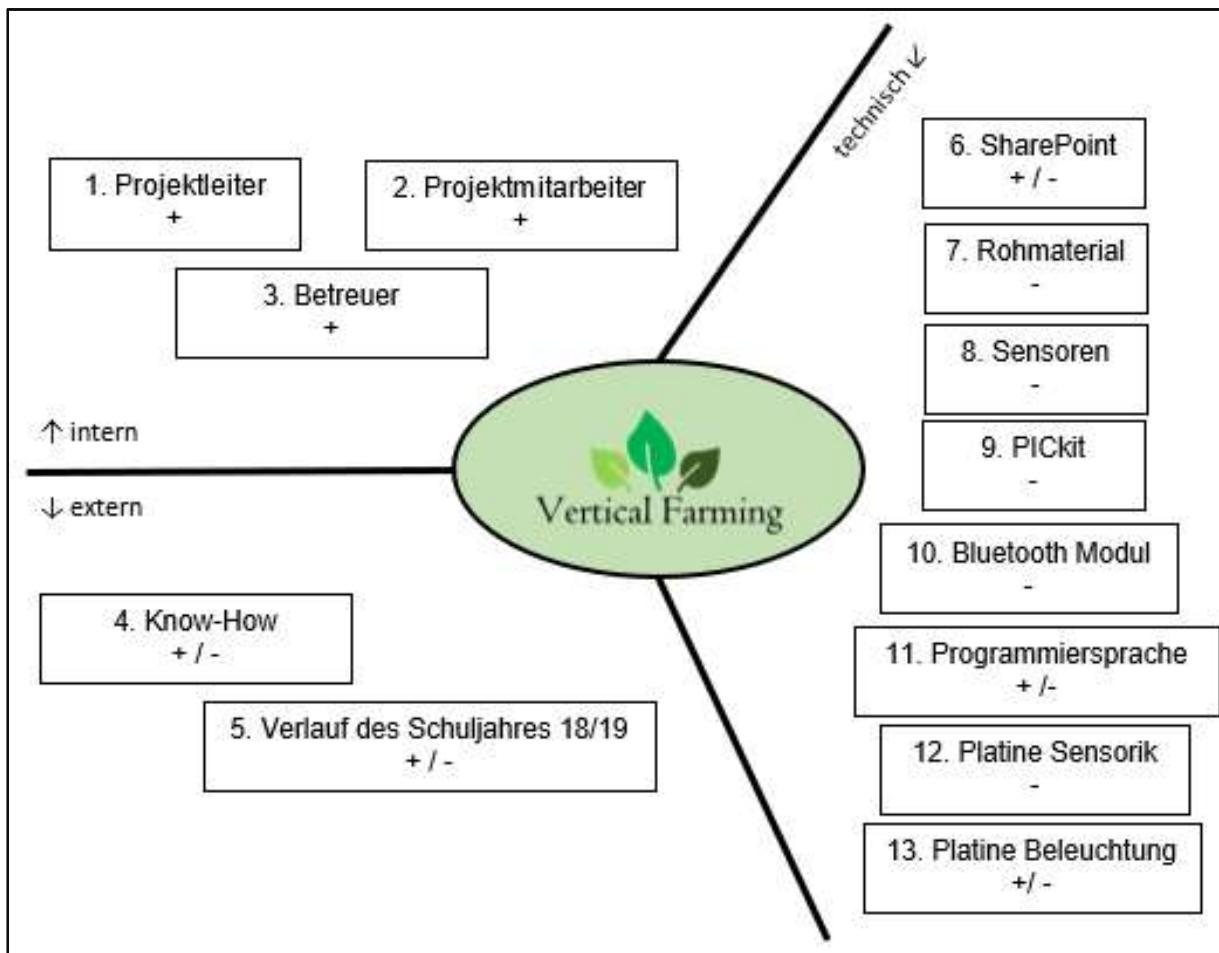


Abbildung 16: Projektumfeldanalyse

2.1.4.3 Beschreibung der wichtigsten Umfelder

#	Bezeichnung	Beschreibung	Bewertung
1	Projektleiter	Motiviert die Projektmitarbeiter und hilft diesen bestmöglich bei offenen Fragen. Ist äußerst engagiert und versucht somit die Organisation des Projektes als Best als möglich zu strukturieren.	+
2	Projektmitarbeiter	Arbeiten engagiert und motiviert an der Diplomarbeit, um mögliche Verzögerungen zu vermeiden.	+
3	Betreuer	Unterstützt das gesamte Diplomarbeitsteam mit fachlicher Beratung.	+
4	Know-How	Das Anwenden von bereits erlernten Kenntnissen kann zeitfördernd sein. Das Erlernen von neuen Fähigkeiten kann hingegen zeitaufwändig sein.	+ / -
5	Verlauf des Schuljahres 18/19	Schularbeiten, Tests oder der normale Unterricht können uns bei Arbeiten an der Diplomarbeit hindern.	+ / -
6	SharePoint	Stellt uns eine Ablagemöglichkeit sämtlicher Dateien zur Verfügung. Bei Kommunikationsfehlern oder gar Ausfällen kann es zu einer erheblichen Verzögerung seitens der Dokumentation geben.	+ / -
7	Rohmaterial	Durch nicht ordnungsgemäßes Verarbeiten des Rohmaterials, könnte es zu einer Knappheit seitens der Ressourcen kommen.	-
8	Sensoren	Durch zu hohe/geringe Spannungs- bzw. Stromversorgung könnte die Sensorik nicht ordnungsgemäß oder gar defekt werden.	-
9	PICkit	Die Anschlüsse des PICkits könnten durch an- und ausstecken bei etwaigen Tests des Programmcodes defekt werden.	-
10	Bluetooth-Modul	Das Bluetooth-Modul könnte durch falsche Verkabelung oder ein falsches Programm, Verbindungsprobleme verursachen.	-
11	Programmiersprache	Die Programmiersprache des PICkits ermöglicht die Entwicklung eines passenden Programmcodes der „Vertical Farming“ Einheit, kann bei sämtlichen technischen Defekten jedoch zu Zeitrückschlägen führen.	+ / -
12	Platine Sensorik	Platine könnte fehlerhaft oder falsch bestückt werden, was massive Zeitaufwände als Folge haben wird.	-
13	Platine Beleuchtung	Platine könnte fehlerhaft oder falsch bestückt werden, was massive Zeitaufwände als Folge haben wird.	-

2.1.5 Risikoanalyse

2.1.5.1 Erklärung der Risikoanalyse

Eine Risikoanalyse wird generell als eine Identifizierung und Bewertung von Risiken bezeichnet. Das Ziel dieser Analyse ist es, die Erreichung aller Projektziele zu gewährleisten und zusätzlich den Schaden sowie die Eintrittswahrscheinlichkeit möglicher Risiken zu minimieren beziehungsweise zu verhindern. Außerdem wird hier die Transparenz, besonders bei Entscheidungssituationen, verbessert. Das heißt, dass es bereits Gegenmaßnahmen zu den jeweiligen Risiken gibt, welche bei Bedürfnis eingesetzt werden können. Wichtig ist es, dass sich alle Projektmitglieder den Risiken bewusst sind, da nicht nur der Projektleiter wissen sollte, was bei welchem Risiko, zu machen ist. Der Grund hierfür ist, dass Projektleiter besonders in deren ersten Projekten eher agieren statt reagieren, das heißt, sie versuchen das Problem eher selbst zu lösen, anstatt die jeweilig betroffenen Projektmitglieder dazu aufzufordern.

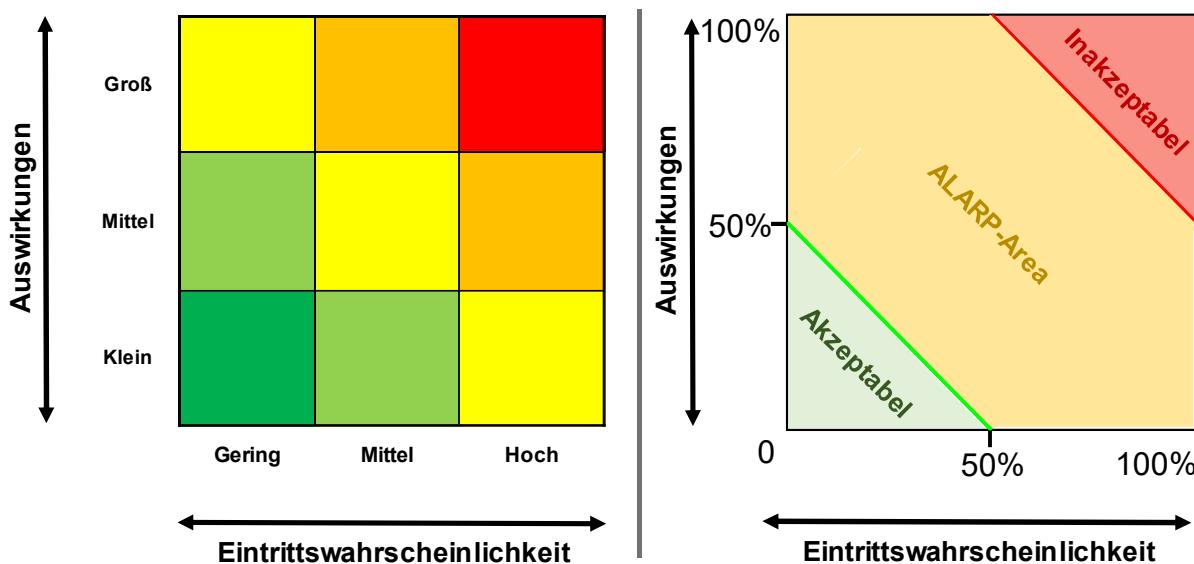


Abbildung 17: Erklärung der Risikoanalyse

Die Farben der obigen Abbildung zeigen die verschiedenen Bereiche eines Risikoportfolios. Im grünen Bereich sind die Risiken akzeptabel und verlangen daher eher selten Gegenmaßnahmen. Der gelbe Bereich wird auch oft als „ALARP-Area“ (as low as reasonably practicable) bezeichnet, in welchem bereits Gegenmaßnahmen getroffen werden sollten.

Im roten Bereich müssen Gegenmaßnahmen getroffen werden, um Risiken mit einer hohen Auswirkung beziehungsweise mit einem hohen Schaden und einer hohen Eintrittswahrscheinlichkeit, entgegenzuwirken.

Nachdem alle Risiken erkannt und niedergeschrieben wurden, werden diese in eine Tabelle eingefügt. Jedes Risiko erhält hierbei eine eigene Nummer, eine Bezeichnung, eine Beschreibung und einen sogenannten Risikofaktor (RF).

Der Risikofaktor ist das Kriterium, nach welchem jedes Risiko beurteilt wird. Dieser Wert setzt sich aus den möglichen Auswirkungen (A) und der Eintrittswahrscheinlichkeit (P) zusammen:

$$RF = P * A$$

Desto niedriger der Risikofaktor ist, desto vernachlässigbarer wird das Risiko, wobei ein Risiko nie unterschätzt werden sollte.

Basierend auf den Bereichen im Risikoportfolio, gibt es drei Möglichkeiten einem Risiko gegenüberzustehen:

1. Akzeptieren

Risiken, die den Projekterfolg kaum oder nur in geringem Maße gefährden, können akzeptiert werden und verlangen nach keinem weiteren Handlungsbedarf. Doch hier ist es ratsam, immer wieder einen Blick auf die anfangs wenig gefährlichen Risiken zu werfen, da sich diese im Laufe der Arbeit an einem Projekt selbstverständlich ändern können. Typischerweise sind dies die Risiken, welche sich im grünen Bereich des Risikoportfolios befinden.

2. Reduzieren

Bei dieser Strategie wird versucht, die Eintrittswahrscheinlichkeit eines Risikos zu senken und / oder das Ausmaß beziehungsweise die Schadenshöhe zu mindern. Diese Methode wird meist bei Risiken in der ALARP-Area verwendet. Zum Beispiel können die benötigten Ressourcen frühzeitig bestellt werden, um einen Zeitverzug aufgrund fern herkommender Ware, zu verhindern. Hierbei wird die Eintrittswahrscheinlichkeit zu spät eintreffender Bauteile minimiert.

3. Vermeiden

Bei Risiken, die einen starken beziehungsweise gravierenden Einfluss auf den Projektablauf haben können, wird es versucht, diesen aus dem Weg zu gehen, um den Gesamterfolg des Projektes nicht zu gefährden. Meist wird diese Methode mit einer Änderung des Projektplanes verbunden, da das Projekt so umgeplant werden muss, dass die Eintrittswahrscheinlichkeit und Auswirkung des gefährlichen Risikos keinen Einfluss mehr auf den Projekterfolg hat. Üblicherweise wird diese Strategie bei Risiken im roten und somit inakzeptablen Bereich eingesetzt.

Außerdem ist es empfehlenswert zu jedem Risiko eine passende Gegenmaßnahme parat zu haben, wie es bei dieser Diplomarbeit der Fall ist. Auch sehr unwahrscheinliche Risiken mit geringer Schadenshöhe, wurden mit einer dazugehörigen Gegenmaßnahme gekennzeichnet.

Es ist besser die Risiken im Laufe einer Risikoanalyse strenger zu beurteilen, als zu locker, da ein Fehler in dieser Phase des Projekts, immense Auswirkungen auf den Projektablauf und somit den Projekterfolg haben könnte.

2.1.5.2 Risikoportfolio

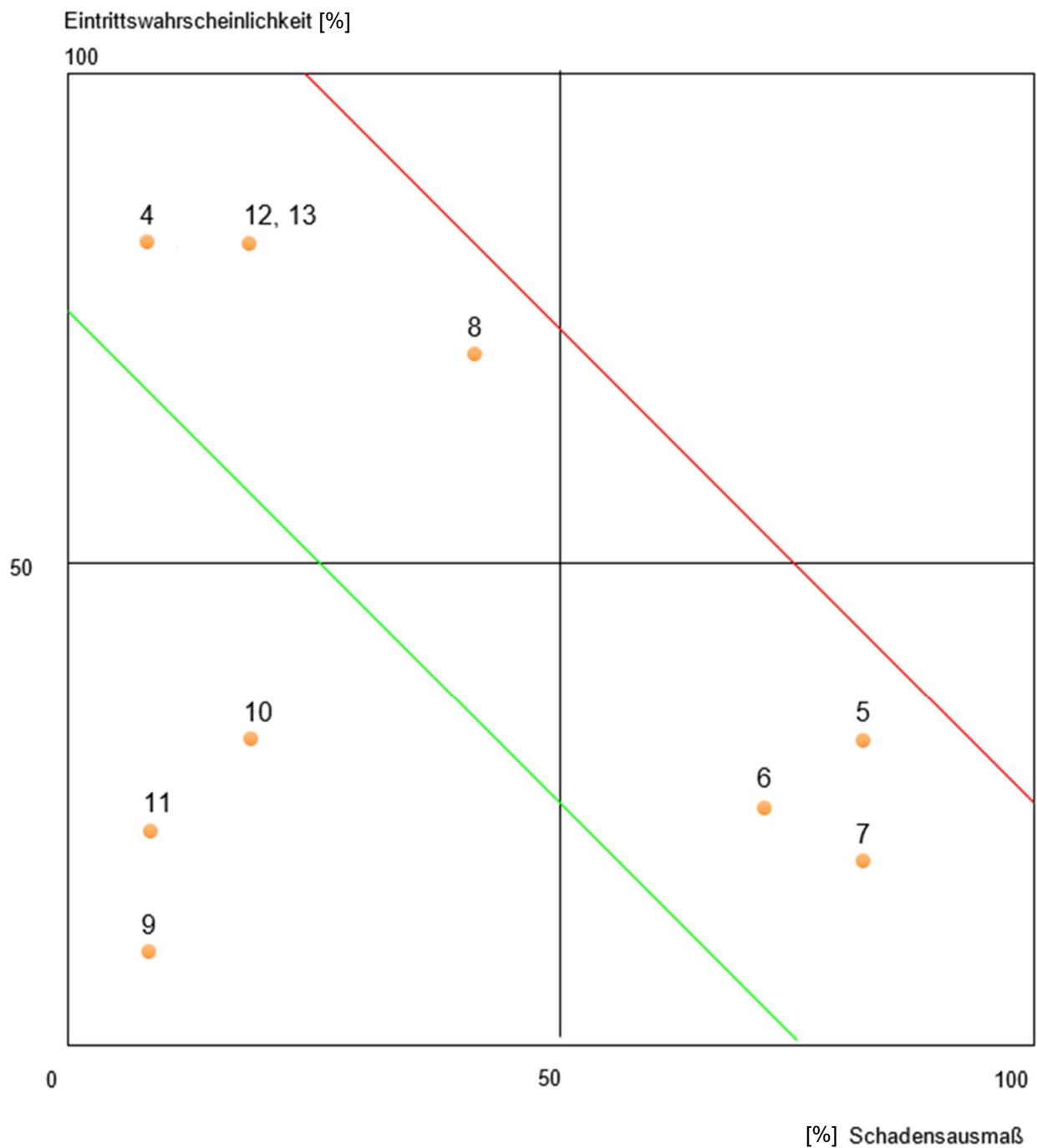


Abbildung 18: Risikoportfolio

2.1.5.3 Beschreibung der wichtigsten Risiken

#	Bezeichnung	Beschreibung des Risikos	P	A	RF
8	Sensoren	Die Sensorik könnte aufgrund von Qualitätsmängeln nicht ordnungsgemäß funktionieren.	70	40	2800
5	Verlauf des Schuljahres 18/19	Schularbeiten, Tests oder der normale Unterricht können uns bei Arbeiten an der Diplomarbeit hindern.	35	80	2800
6	SharePoint	Back-Up Fehler könnten zu Datenverlust führen.	25	75	1875
7	Rohmaterial	Durch nicht ordnungsgemäßes Verarbeiten des Rohmaterials, könnte es zu einer Knappheit seitens der Ressourcen kommen.	20	80	1600
12	Platine Sensorik	Platine könnte fehlerhaft oder falsch bestückt werden, was massive Zeitaufwände als Folge haben wird.	80	20	1600
13	Platine Beleuchtung	Platine könnte fehlerhaft oder falsch bestückt werden, was massive Zeitaufwände als Folge haben wird.	80	20	1600
4	Know-How	Möglicher Zeitaufwand, um nötiges Know-How aufzuarbeiten.	85	10	850
10	Bluetooth-Modul	Das Bluetooth-Modul könnte durch falsche Verkabelung oder ein falsches Programm, Verbindungsprobleme verursachen.	30	20	600
11	Programmiersprache	Kann bei sämtlichen technischen Defekten jedoch zu Zeitrückschlägen führen.	20	10	200
9	PICkit	Die Anschlüsse des PICkits könnten durch an- und ausstecken bei etwaigen Tests des Programmcodes defekt werden.	10	10	100

Tabelle 2: Beschreibung der wichtigsten Risiken

2.1.5.4 Risikogegenmaßnahmen

#	Bezeichnung	Gegenmaßnahme
8	Sensoren	Die zugehörigen Datenblätter sorgfältig durchlesen und um einen möglichen Zeitstress aus dem Weg zu gehen, Sensoren zur Genüge und früh genug bestellen.
5	Verlauf des Schuljahres 18/19	Die Planung der einzelnen Arbeitspakete / „Sprints“ in Zeit zwischen Tests, Schularbeiten und Prüfungen legen.
6	SharePoint	Die Backup-Dateien regelmäßig überprüfen und auf externen Speichermedien sichern.
7	Rohmaterial	Alle Arbeiten werden ordnungsgemäß bzw. erst nach Absprache durchgeführt. Zudem genügend Reservematerial einplanen
12	Platine Sensorik	Vor Fertigung der Platine wird das Platinenlayout inkl. Bauteile mehrmals kontrolliert, einerseits durch das Projektteam und andererseits durch dazugehörige Betreuer.
13	Platine Beleuchtung	Vor Fertigung der Platine wird das Platinenlayout inkl. Bauteile mehrmals kontrolliert, einerseits durch das Projektteam und andererseits durch dazugehörige Betreuer.
4	Know-How	Durch bestmögliche Planung auftretenden Zeitaufwand einplanen. Eventuell auch Hilfe bei dazugehörigen Betreuern einholen.
10	Bluetooth-Modul	Vor Inbetriebnahme des Systems wird die Verkabelung bzw. der Programmcode kontrolliert.
11	Programmiersprache	Genügend Puffer einplanen und bei absoluter Katastrophe früh genug eingreifen und Change-Request schreiben.
9	PICkit	Der Stecker, zum Übertragen des Programmcodes vom Desktop auf den PIC, wird nur auf Anweisung des dazugehörigen Betreuers abgesteckt. Ansonsten stecken wir diesen nie ab.

Tabelle 3: Risikogegenmaßnahmen

2.1.6 Objektstrukturplan

2.1.6.1 Erklärung des Objektstrukturplans

Der OSP – oder auch Objektstrukturplan genannt, orientiert sich an den Ergebnissen eines Projektes. Bei der Erstellung eines Planes dieser Art ist zu beachten, dass die Objektteile anhand zuvor festgelegter Strukturierungskriterien hierarchisch gegliedert werden. Hierbei werden alle Produktkomponenten festgelegt, welche nötig sind, um eine Erfüllung der Projektziele gewährleisten zu können.

Genauer gesagt ist der Objektstrukturplan die Basis für den Projektstrukturplan, in welchem die benötigten Arbeiten als Arbeitspakete definiert werden. Er ist somit eine allemal notwendige Planungshilfe, welche sich um die technische Struktur eines Projektes kümmert. Die Intention dieses Plans ist es, einen ordnungsgemäßen Ablauf eines Projektes zu gewährleisten. Wichtig, bei der Erstellung eines OSP's ist es zudem wichtig nur Nomen und keine Verben zu verwenden.

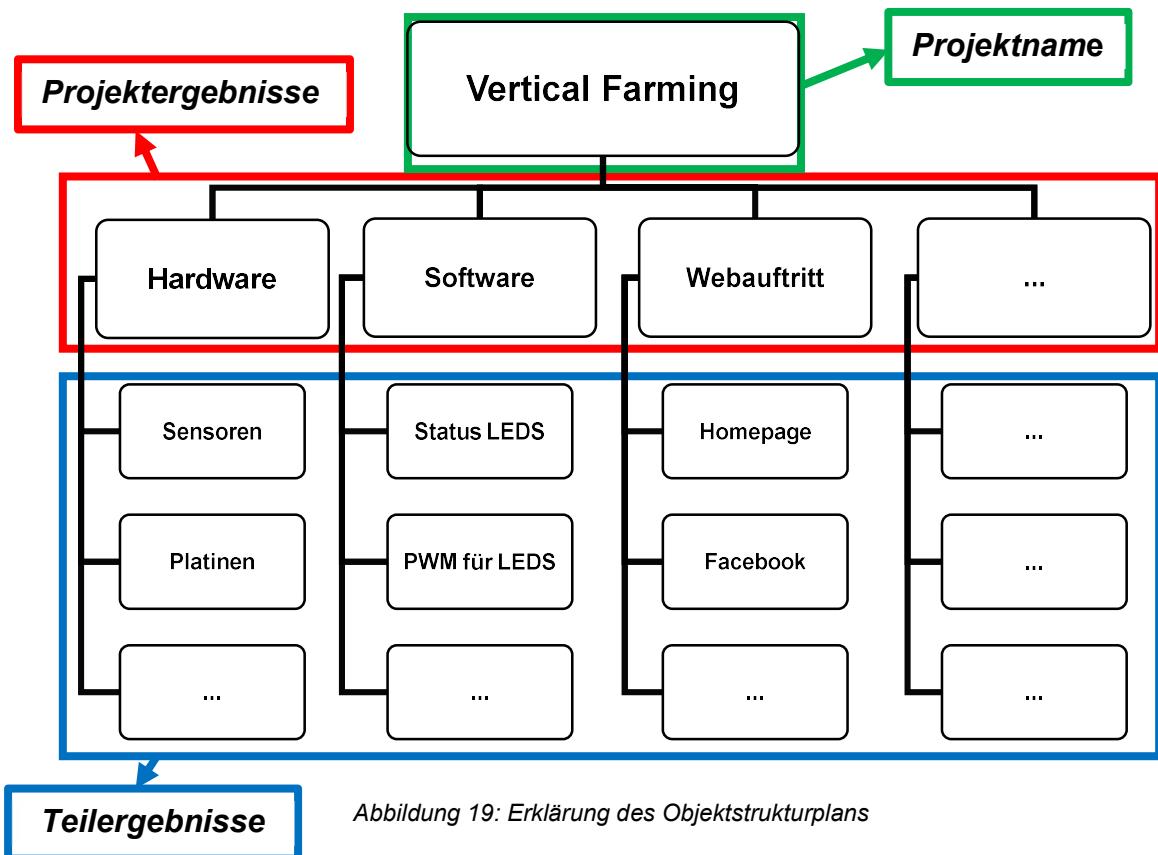


Abbildung 19: Erklärung des Objektstrukturplans

Anhand obiger Abbildung lässt sich der Aufbau eines solchen OSP's gut erklären: An der obersten Position befindet sich der Projektname oder die Projektbezeichnung. Unter diesem Block befinden sich alle Projektergebnisse, welche nötig sind, um das Projekt erfolgreich umsetzen zu können. Von den Projektergebnissen ausgehend, werden die einzelnen Teilergebnisse erstellt und untergeordnet. Diese Teilergebnisse sind sozusagen Unterthemen des Hauptthemas, beispielsweise müssen die Teilergebnisse „Sensoren“ & „Platinen“ erledigt werden, um das Projektergebnis „Hardware“ abschließen zu können.

2.1.6.2 Grafische Darstellung

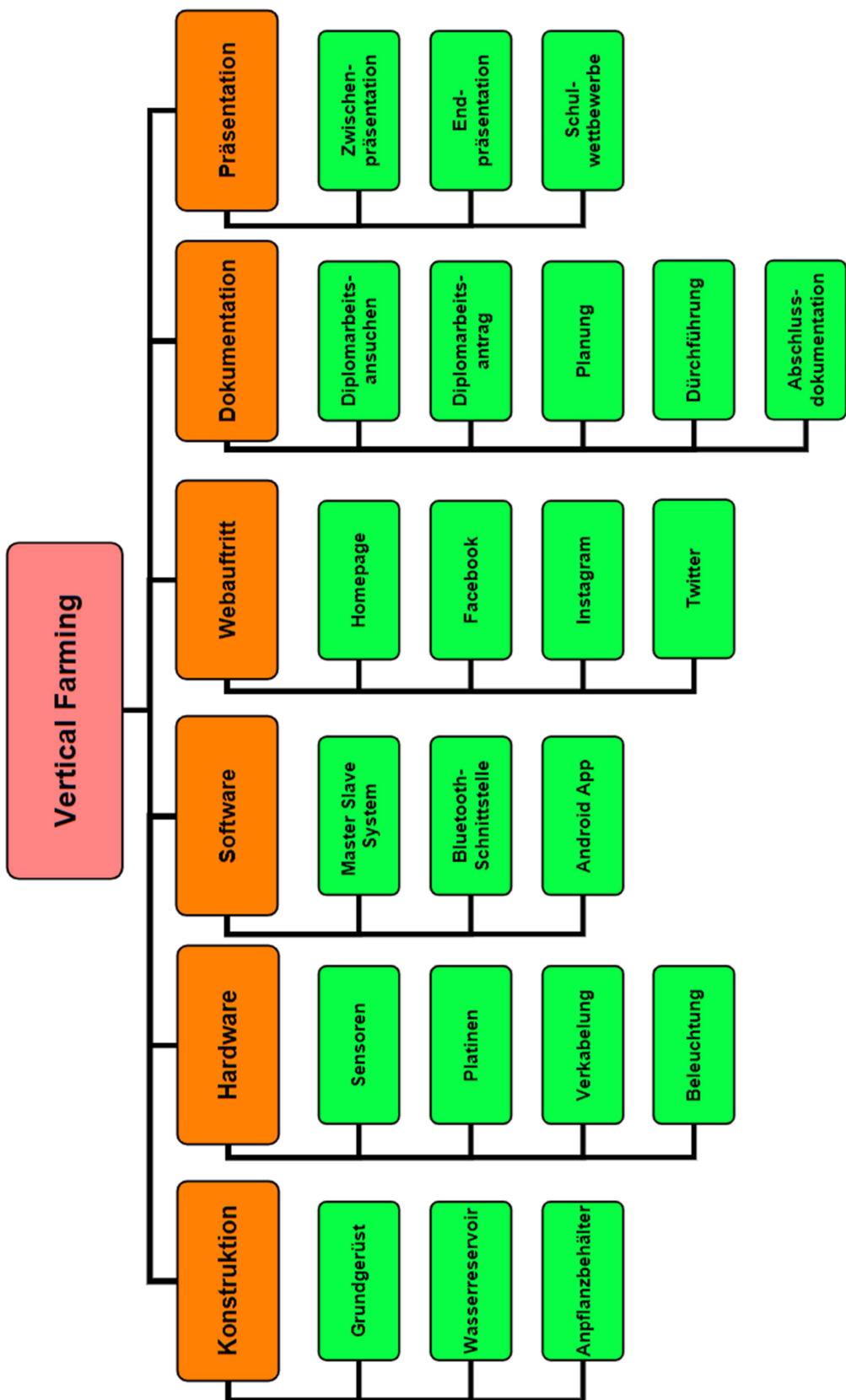


Abbildung 20: Objektstrukturplan

2.1.7 Projektstrukturplan

2.1.7.1 Erklärung des Projektstrukturplans

Der PSP – oder auch Projektstrukturplan genannt, basiert auf dem OSP und ist somit einer der wichtigsten Planungsmethoden im Bereich des Projektmanagements. In einem PSP wird das Projekt in Phasen sowie Arbeitspakete gegliedert. Wichtig ist es hierbei, dass die Erstellung mit dem gesamten Team erfolgt, um Erfahrung und vorhandenes Know-How integrieren zu können.

Bei der Gliederung eines solchen Planes gibt es folgende drei Methoden:

1. Funktionsorientierte Gliederung

Bei der funktionsorientierten Gliederung wird das Projekt, unter Berücksichtigung der Unternehmensfunktionen, in Teilgebiete aufgeteilt. Zum Beispiel gibt es ein Teilprojekt, für die Planung, für die Konstruktion, für die Fertigung und dergleichen. Diese Methode hat den Vorteil, dass sie den Mitwirkenden angenehmer ist, da sich die Vorgesetzten und verantwortlichen Mitglieder bereits kennen. Dennoch wird diese Gliederung eher in großen Unternehmen, als wie in Entwicklungsprojekte, wie es bei uns der Fall war, verwendet.

2. Objektorientierte Gliederung

Diese Gliederungsform wird verwendet, wenn eine Vielzahl von ähnlichen oder identen Objekten das Projekt auf eine ähnliche Art und Weise beeinflussen. Als gutes Beispiel dient eine Veranstaltung oder ein Fest, bei dem mehrere Künstler auf verschiedenen Bühnen auftreten. Hierbei wird das Fest als Gesamtprojekt und die verschiedenen Bühnen als Teilprojekt betrachtet.

3. Phasenorientierte Gliederung

Bei dieser Gliederung wird das Projekt in verschiedene Phasen unterteilt, welche das Projekt durchlaufen soll, bevor das Endprodukt fertig ist. Einen Einsatz findet diese Methode beispielsweise bei Entwicklungsprojekten, wie es auch bei uns der Fall war.

In diesem Projekt wurde die phasenorientierte Gliederung gewählt, bei welcher das Projekt in folgende 5 Phasen, unterteilt wurde:

1. Projektplanungsphase

Hier werden unter anderem Ziele definiert und andere notwendige Maßnahmen getätigt, die die Wahrscheinlichkeit eines Projekterfolges erhöht und gegebenenfalls gewährleistet.

1. Projektplanungsphase

2. Technische Planungsphase

In dieser Phase wird ein hoher Konzeptionsaufwand betrieben, wie zum Beispiel, das Konstruieren eines Grundgerüstes oder eines Wasserreservoirs.

2. Techn. Planungsphase

3. Umsetzungsphase

In der Umsetzungsphase wird eines der zuvor erstellten Konzepte, unter Berücksichtigung der Ziele, gefertigt und somit umgesetzt.

3. Umsetzungsphase

4. Testphase

Wie schon der Name dieser Phase verrät, wird hier das umgesetzte Konzept getestet. Zum Beispiel kann hier die Platine getestet und die Verkabelung überprüft werden.

4. Testphase

5. Abschlussphase

In der letzten auch sogenannten Abschlussphase, werden alle notwendigen Arbeiten erledigt, welche nötig sind, um das Projekt zu präsentieren beziehungsweise die erlangten Erfahrungen zu verschriftlichen.

5. Abschlussphase

Abbildung 21: Projektphasen

Ebenso wie der OSP, ist der PSP hier auch hierarchisch gegliedert und strukturiert. Die Strukturierung des Projektstrukturplanes erfolgte nach dem „Top-Down-Ansatz“:

- Bezeichnung des Projektes
 - z.B.: „1. Vertical Farming“
- Auswahl der geeigneten Phasen für die zweite Ebene
 - z.B.: „1.1 Projektplanungsphase“, usw.
- Aufteilung des Gesamtprojekts in Teilprojekte
- Auflistung der Aufgaben in Form von Arbeitspaketen
 - z.B.: „1.1.1 Ziele definieren“
- Angabe des dazugehörigen Zeitaufwandes
 - z.B.: 2h
- Definieren benötigter Meilensteine
 - z.B.: nach Abschluss von „1.1.2 Projekt grob planen“

Mit dieser Methode beginnt man mit dem Projekt als Ganzen und geht dann immer mehr ins Detail, bis schließlich die Arbeitspakete gebildet sind. Außerdem sollten Meilensteine definiert werden, um den IST-Stand immer wieder mit dem SOLL-Stand vergleichen zu können und gegebenenfalls, bei Terminverzug vorzeitig eingreifen zu können. Zu beachten ist hier weiters, dass alle Arbeitspakete in Form eines Verbes und nicht in Form eines Nomens gestaltet werden müssen.

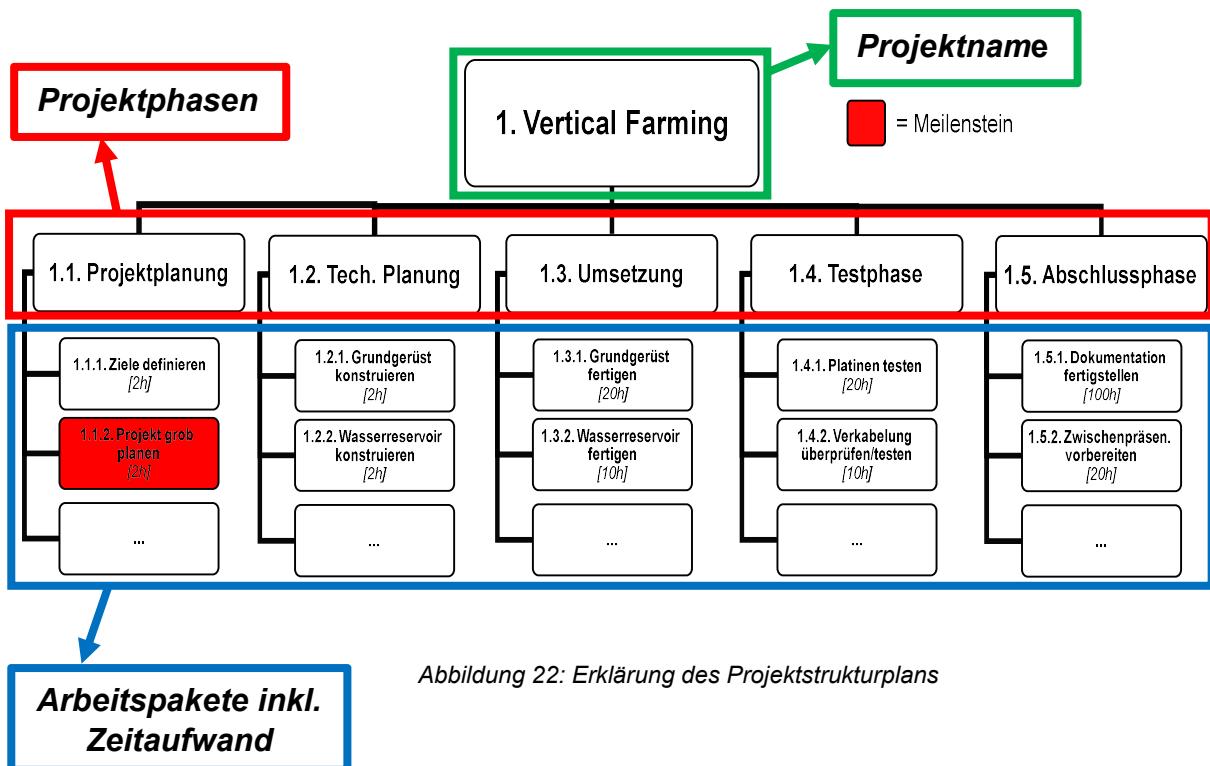
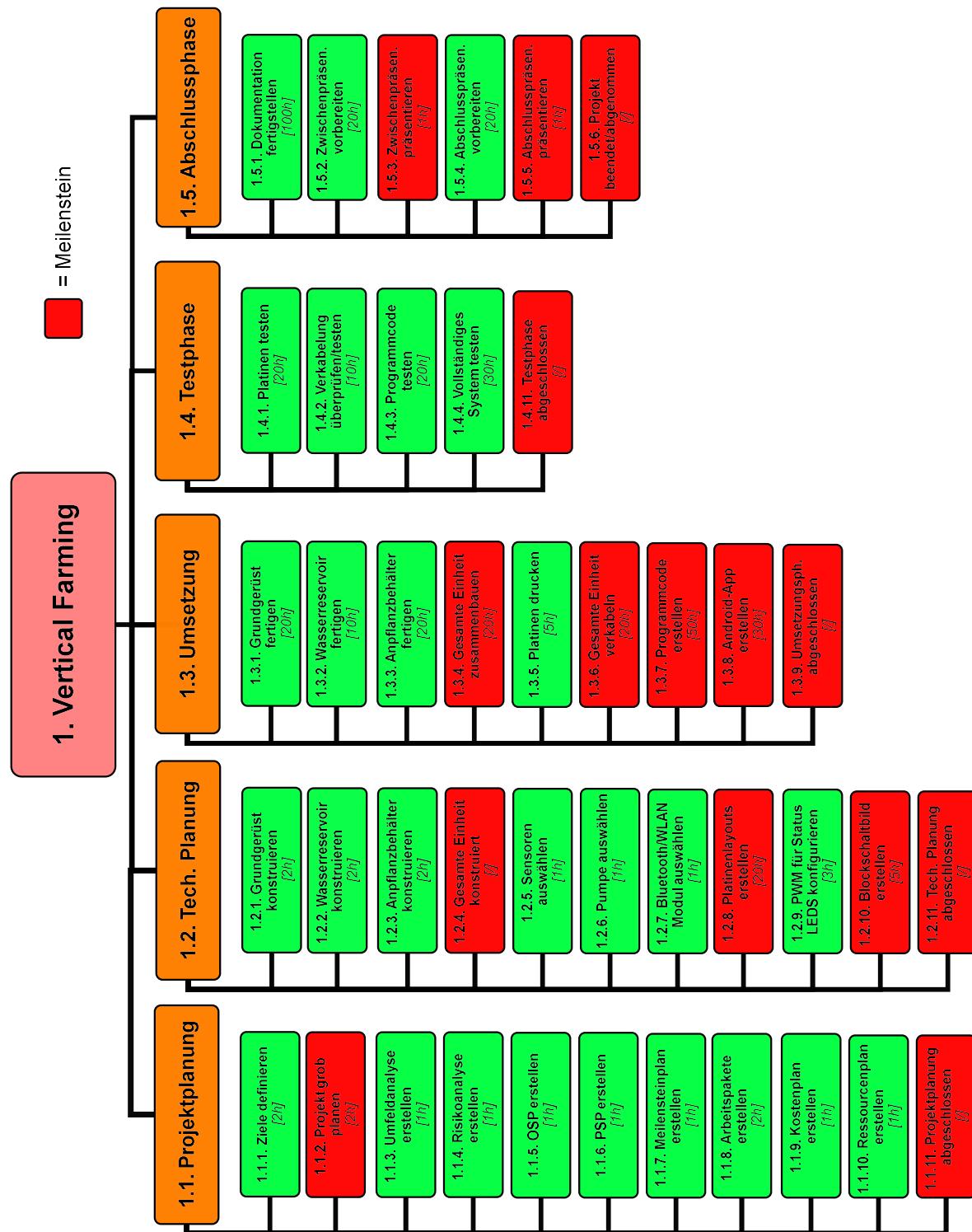


Abbildung 22: Erklärung des Projektstrukturplans

Anhand obiger Grafik kann ein solcher PSP mit Projektnamen, Projektphasen, Arbeitspaketen, Zeitaufwand und Meilensteinen gut dargestellt werden. Vor allem bei einem PSP ist es vorteilhaft, den Projektnamen, die Projektphasen und die darin benötigten Arbeitspakete durchzunummerieren, um den Plan übersichtlich zu halten. Außerdem ist es empfehlenswert, den geschätzten Zeitaufwand direkt unter das jeweilige Arbeitspaket zu schreiben.

Die Meilensteine werden nicht wie gewöhnliche Arbeitspakete illustriert, sondern besonders gekennzeichnet, beispielsweise mit der Farbe Rot. Diese sollten optisch herausstechen, da diese bei dem Meilensteinplan wieder anzutreffen sind. Zudem ist es für das Projektteam wesentlich einfacher dem Meilensteinplan folgen zu können, wenn alle Meilensteine auch im Projektstrukturplan wiederzufinden sind.

2.1.7.2 Grafische Darstellung



2.1.8 Meilensteinplan

2.1.8.1 Erklärung des Meilensteinplans

Bei Projekten, die umfangreich oder eine Durchlaufzeit von einer sehr langen Zeit haben, ist es sehr schwierig den Überblick zu bewahren. In diesem Fall werden solch komplizierte Aufgaben mithilfe von Projektplänen, welche Termine, Kosten Mitarbeiter, Ergebnisse und derartiges beinhalten, gelöst. Je höher die Komplexität und Vernetzung diverser Aufgaben sind, desto schwieriger wird es als Projektleiter durchzublicken. Auch mithilfe von Objekt- und Projektstrukturplänen kann dies sehr schwer zu meistern sein. Aus diesem Grund ist es essenziell einen Meilensteinplan zu erstellen, um die momentane Termsituation des Projektes abrufen zu können.

Die Voraussetzungen für die Erstellung eines solchen Plans sind folgende:

- Start- und Endtermin des Projektes
- Inhalt und Ziele des Projektes
- Wann welche Teilergebnisse vorliegen müssen, muss bekannt sein
- Klarheit über die angenommene Ressourcensituation
- Gewissheit, wann welche Ressourcen in welcher Menge benötigt werden
- Ein mit dem Projektteam besprochener Projektstrukturplan
- Umfeldanalyse (inkl. Stakeholder)

Wichtig ist es außerdem anzumerken, dass ein Meilenstein als Ergebnis oder Ereignis und nicht als Aktivität definiert wird. Zum Beispiel ist bei dem Arbeitspaket „Projekt grob planen“ der dazugehörige Meilenstein „Projekt grob geplant“.

Das Ziel eines Meilensteinplanes sieht somit wie folgt aus:

- Grobe Terminplanung des gesamten Projekts
- Schaffen von Zwischenzielen – ggf. zur Mitarbeitermotivation
- Wichtige Projekttereignisse transparent machen
- Übersicht der Terminverzüge und deren Auswirkung auf andere Termine
- Abrufen des aktuellen Leistungsfortschritts

Der Meilensteinplan legt somit Informationen über zeitkritische Ereignisse dar. Die termingerechte Einhaltung dieser Ereignisse ist unumgänglich, da es sonst zu verspäteten Zwischen- oder Endterminen kommt oder Deadlines nicht eingehalten werden können. Die einzelnen Meilensteine dienen auch als sogenannte Zwischenziele, welche als Zwischenergebnis gewertet werden. Bei Projekten, die sich über einen langen Zeitraum ziehen, kann das Erreichen dieser Zwischenziele eine durchaus benötigte Motivation für das gesamte Projektteam sein. Üblicherweise werden Meilensteine zum Beispiel beim Projektstart und beim Projektende oder am Ende einer Projektphase und/oder eines Arbeitspakets gesetzt. Diese Arbeitspakete sind im Projektstrukturplan ersichtlich. Die Darstellung des Meilensteinplanes kann in Form einer Tabelle oder einer Grafik erfolgen.

2.1.8.2 Darstellung der Meilensteine mit SOLL bzw. IST-Terminen

SOLL	IST	Meilenstein
14.09.2018	14.09.2018	Projekt grob geplant
21.09.2018	21.09.2018	Projektplanung abgeschlossen
28.09.2018	15.01.2019	Gesamte Einheit konstruiert
28.09.2018	07.11.2018	Platinenlayouts erstellt
28.09.2018	02.09.2018	Blockschaltbild erstellt
28.09.2018	07.11.2018	Tech. Planung abgeschlossen
28.11.2018	28.11.2018	Zwischenpräsentation abgehalten
23.11.2018	20.11.2018	Gesamte Einheit zusammengebaut
07.12.2018	27.11.2018	Android-App erstellt
21.12.2018	28.12.2018	Gesamte Einheit verkabelt
01.03.2019	26.02.2019	Programmcode erstellt
01.03.2019	26.02.2019	Umsetzungsphase abgeschlossen
15.03.2019	15.03.2019	Testphase abgeschlossen
02.04.2019	02.04.2019	Abschlusspräsentation abgehalten
05.04.2019	19.03.2019	Projekt beendet/abgenommen

Tabelle 4: Meilensteinliste SOLL-IST Vergleich

Alle Meilensteine wurden etwas früher angesetzt, als sie nötig waren, um den Arbeitsdruck auf einem konstanten Level halten zu können. Einige Meilensteine wurden zwar verspätet aber dennoch zeigerecht abschlossen, da diese auch problemlos zu einem späteren Zeitpunkt angesetzt hätten werden können.

Da die Meilensteine in 4 Unterpunkte und zwar in das Projektmanagement, in die Mechanik, in die Elektronik und in die Programmierung aufgeteilt wurden, kam es kaum zu verspäteten Abgaben, da diese 4 Gebiete meist parallel abliefen.

Lediglich bei Änderungen der ursprünglichen Versionen, zum Beispiel der Konstruktion, des Platinenlayouts oder des Programmcodes kam es zu einem nichtehalten des SOLL-Termines. Der Grund hierbei waren unvorhersehbare Änderungen.

2.1.8.2.1 Meilenstein 1: „Projekt grob geplant“

Bei diesem Meilenstein musste das Projekt bis zum 14.09.2018 „grob“ geplant werden. Das bedeutet, es musste die Planungsphase bereits absolviert sein und ein grober Überblick über das Projekt bestehen.

Da der Meilenstein am 14.09.2018 pünktlich absolviert wurde, wurde keine Gegenmaßnahme unternommen.

2.1.8.2.2 Meilenstein 2: „Projektplanung abgeschlossen“

Wie der Name dieses Meilensteins schon verrät, musste hier bis zum 21.09.2018 die Projektplanung in detaillierter Form und ein exakter Überblick über das Projekt vorhanden sein. Der Grund hierfür war, dass der Antrag zur Durchführung dieser Diplomarbeit ebenfalls bis zum 21.09.2018 fällig war.

Auch dieser Meilenstein wurde fristgemäß bis 21.09.2018 abgeschlossen, wodurch keine Gegenmaßnahme getätigten werden musste.

2.1.8.2.3 Meilenstein 3: „Gesamte Einheit konstruiert“

Bis zum 28.09.2018 musste der Meilenstein „Gesamte Einheit konstruiert“ abgeschlossen werden. Das heißt, die „Vertical-Farming Einheit“ musste hier in dem CAD-Programm CREO Parametric modelliert werden. Alle Einzelteile mussten konstruiert und zu einer größeren Baugruppe zusammengefügt werden.

Zu konstruieren war das zu umsetzende Konzept bis 28.09.2018, das auch zeitgerecht abgeschlossen werden konnte. Jedoch kam es im Laufe der Umsetzungs- und Testphase immer wieder zu kleinen Änderungen der Konstruktion, wodurch es nicht möglich war, den Meilenstein bis zu dem oben genannten Datum abzuschließen. Dennoch wurde nach fristgemäßer Absprache mit den Betreuern beschlossen, dass der Meilenstein erst abgeschlossen werden kann, wenn die Konstruktion fertig ist. Tatsächlich wurde die endgültige Konstruktion am 15.01.2019 modelliert. Da aber dieser Meilenstein keinen Effekt auf den Projektablauf hatte, war diese Verzögerung nicht weiter problematisch. Aus diesem Grund wurde keine Gegenmaßnahme durchgeführt.

2.1.8.2.4 Meilenstein 4: „Platinenlayouts erstellt“

Bei diesem Meilenstein mussten die benötigten Platinenlayouts bis zum 28.09.2018 erstellt werden. Darunter fällt die Hautplatine, welche zur Bewässerung, zur Beleuchtung und zum Auslesen der Sensoren nötig war. Hier handelte es sich lediglich um das Layout und nicht um die Platine mit bestückten Bauteilen.

Dieser Meilenstein wurde am 24.10.2018 und somit mit einer Verzögerung von 26 Tagen fertiggestellt. Als Gegenmaßnahme wurde die Programmierung parallel zu der Elektronik verschoben, sodass das Platinenlayout keinen Einfluss mehr auf den Projektablauf hatte. Zudem wurde beschlossen, dass zusätzliche Zeit investiert werden musste.

2.1.8.2.5 Meilenstein 5: „Blockschatzbild erstellt“

Der Meilenstein „Blockschatzbild erstellt“ musste bis zum 28.09.2018 fertiggestellt werden. Hierbei handelt es sich um eine grafische Darstellung des Systems als auch der zueinanderstehenden Bauteile.

Bei diesem Meilenstein gab es keinen Handlungsbedarf, da er termingemäß am 02.09.2018 abgeschlossen wurde.

2.1.8.2.6 Meilenstein 6: „Tech. Planung abgeschlossen“

Dieser Meilenstein ist abhängig von den Meilensteinen „Gesamte Einheit konstruiert“, „Platinenlayouts erstellt“ und „Blockschatzbild erstellt“. Erst wenn all diese Arbeitspakete erfolgreich abgearbeitet wurden, gilt die technische Planung und somit dieser Meilenstein als abgeschlossen. Der SOLL-Termin war der 28.09.2018.

Wie bereits erwähnt, wurde der Meilenstein „Gesamte Einheit konstruiert“ nach Absprache mit den Betreuern, in die Umsetzungsphase verschoben, da diese erst fertig modelliert werden kann, wenn die Konstruktion fertig ist. Aufgrund des Meilensteines „Platinenlayouts erstellt“ hat sich die Fertigstellung des Meilensteines „Tech. Planung abgeschlossen“ von 28.09.2018 auf 24.10.2018 verschoben.

Hier wurde keine Gegenmaßnahme unternommen, da bereits eine bei dem Meilenstein „Blockschatzbild erstellt“ getroffen wurde. Durch diese Maßnahme konnte der Projektablauf wieder „auf Kurs“ gebracht werden.

2.1.8.2.7 Meilenstein 7: „Zwischenpräsentation abgehalten“

Dieser Meilenstein diente als Erinnerung an die Zwischenpräsentation dieser Diplomarbeit. Hierbei musste eine PowerPoint-Datei erstellt, nötige Informationen erarbeitet und das Projekt präsentiert werden.

Da der Meilenstein, wie auch schon geplant, am 28.11.2019 abgeschlossen wurde, musste keine Gegenmaßnahme unternommen werden.

2.1.8.2.8 Meilenstein 8: „Gesamte Einheit zusammengebaut“

Der Meilenstein „Gesamte Einheit zusammengebaut“ musste bis zum 21.12.2018 fertiggestellt werden. Das bedeutet, es mussten die Robotunits-Profile verschraubt, die Anpflanzwanne geschweißt und dergleichen erledigt werden, damit dieser Meilenstein abgeschlossen werden konnte.

Bei diesem Meilenstein wurde kein Handlungsbedarf benötigt, da die Konstruktion bereits am 20.11.2018 fertiggestellt wurde.

2.1.8.2.9 Meilenstein 9: „Android-App erstellt“

Wie der Name dieses Meilensteins schon verrät, musste hier bis zum 07.12.2018 die zum Steuern der Beleuchtung und der Bewässerung als auch zum Abrufen der Messwerte der Sensoren benötigte Android-App erstellt werden.

Auch bei diesem Meilenstein gab es keinen Handlungsbedarf, da dieser bereits am 27.11.2018 abgeschlossen wurde.

2.1.8.2.10 Meilenstein 10: „Gesamte Einheit verkabelt“

Bis zum 07.12.2018 musste der Meilenstein „Gesamte Einheit verkabelt“ abgeschlossen werden. Das heißt, es mussten die High-Power LEDs, der Füllstandsensor, der Lüfter und der Temperatur- / Feuchtigkeitssensor angeordnet, befestigt und verkabelt werden.

Der Meilenstein wurde am 28.12.2018 und somit 21 Tage zu spät abgeschlossen. Um dieser verspäteten Fertigstellung entgegenzuwirken, wurde dieser Meilenstein und somit dieses Arbeitspaket in die Weihnachtsferien verlegt. Mit dieser Gegenmaßnahme konnte die zeitliche Auswirkung auf den Projektablauf minimiert und der Projektstatus auf „Grün“ gestellt werden.

2.1.8.2.11 Meilenstein 11: „Programmcode erstellt“

Bei diesem Meilenstein musste der Programmcode für den Mikrocontroller in MPLAB und für die Android App mit Android Studio bis zum 01.03.2019 geschrieben werden.

Da bereits am 26.02.2019 der Programmcode erstellt wurde, wurde keine Gegenmaßnahme verwendet.

2.1.8.2.12 Meilenstein 12: „Umsetzungsphase abgeschlossen“

Dieser Meilenstein ist abhängig von den Meilensteinen „Gesamte Einheit zusammengebaut“, „Android-App erstellt“, „Gesamte Einheit verkabelt“ und „Programmcode erstellt“. Erst wenn all diese Arbeitspakete erfolgreich abgearbeitet wurden, gilt die Umsetzungsphase und somit dieser Meilenstein als abgeschlossen.

Der SOLL-Termin war der 01.03.2019, welcher durch den IST-Termin mit 26.02.2019 vorzeitig erledigt wurde.

2.1.8.2.13 Meilenstein 13: „Testphase abgeschlossen“

Der Sinn dieses Meilensteines war es, das finale und somit umgesetzte Konzept zu testen. Das heißt, hier musste die Mechanik, die Elektronik und die Programmierung auf Fehler überprüft werden. Beispielsweise musste hier der Programmcode auf die Platine gespielt werden, um die Kompatibilität von Elektronik und Programmierung zu testen.

Der Meilenstein „Testphase abgeschlossen“ wurde exakt mit dem Datum des SOLL-Termines am 15.03.2019 abgeschlossen.

2.1.8.2.14 Meilenstein 14: „Abschlusspräsentation abgehalten“

Dieser Meilenstein diente als Erinnerung an die Abschlusspräsentation dieser Diplomarbeit. Hierbei musste eine PowerPoint-Datei erstellt, nötige Informationen erarbeitet und das Projekt präsentiert werden.

Da der Meilenstein, wie auch schon geplant, am 02.04.2019 abgeschlossen wurde, musste keine Gegenmaßnahme unternommen werden.

2.1.8.2.15 Meilenstein 15: „Projekt beendet/abgenommen“

Der Meilenstein „Projekt beendet/abgenommen“ musste bis zum 05.04.2019 abgeschlossen werden. Hierbei musste das Projekt fertiggestellt, das Diplomarbeitsbuch geschrieben und die Abschlusspräsentation abgehalten werden.

Da dieser Meilenstein das Ende der Diplomarbeit kennzeichnet und dieses bereits seit Projektanfang vorgegeben war, ist hier keine Terminverzögerung möglich. Entweder wurde das Projekt erfolgreich oder negativ abgeschlossen. In dem Fall dieser Diplomarbeit, konnte das Projekt dank passender Gegenmaßnahmen sogar bereits am 19.03.2019 fertiggestellt werden.

2.1.9 Personelle Ressourcen

Anhand folgender Grafik können die personellen Ressourcen, welche zur Umsetzung der Diplomarbeit „Vertical Farming“ nötig waren, abgelesen werden:

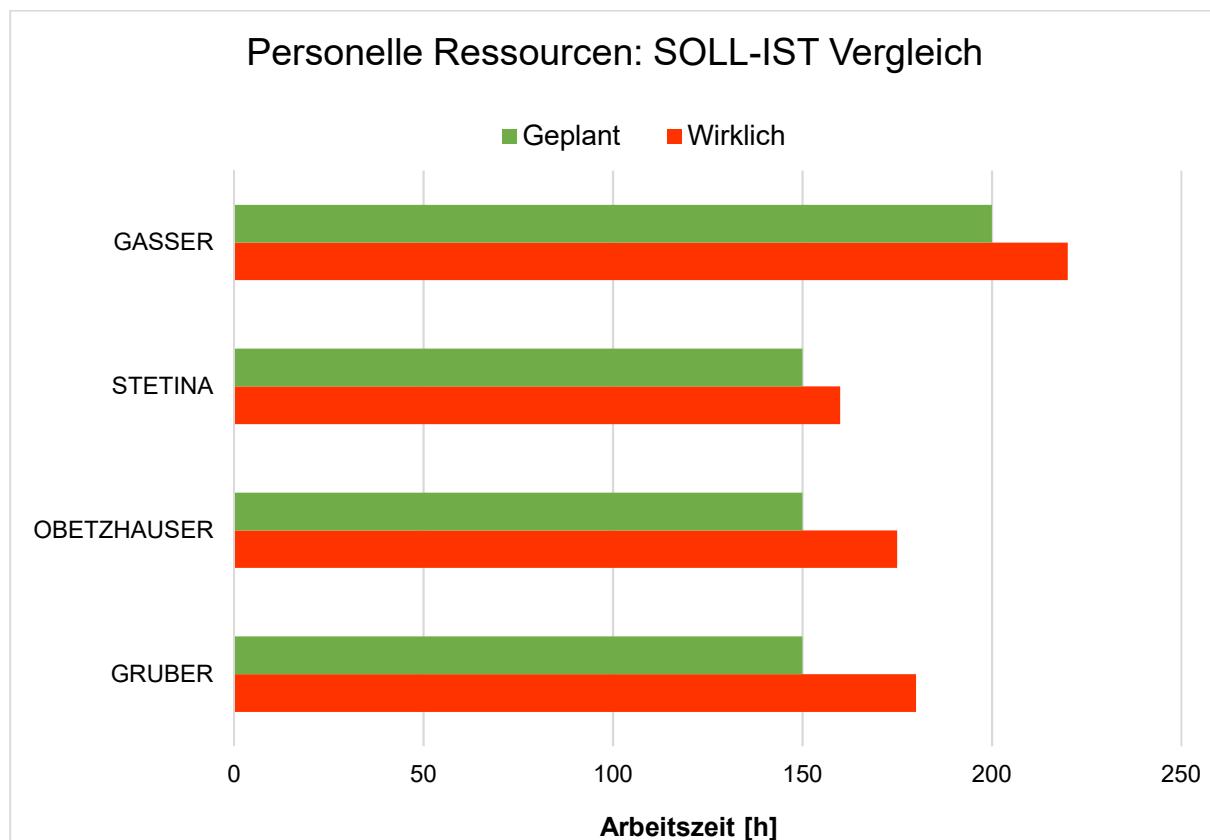


Abbildung 24: Personelle Ressourcen SOLL-IST Vergleich

Wie der SOLL-IST Vergleich der Arbeitsstunden bereits veranschaulicht, wurden die geplanten Aufwände unterschätzt. Dennoch war die Differenz zwischen SOLL und IST nicht so immens, dass dies einen Effekt auf den Projektlauf hatte. Aus diesem Grund kann gesagt werden, dass das gesamte Projektteam die jeweiligen Aufwände gut eingeschätzt hat.

2.1.10 Budget

2.1.10.1 Geplante Aufwände für Durchführung des Projektes

Anhand folgender Tabelle können die geplanten materiellen Aufwände zur Durchführung der Diplomarbeit „Vertical Farming“ abgelesen werden:

Pos.	Bezeichnung des Aufwands	Kosten	Kumuliert
1	Konstruktion	€ 1500	€ 1500
2	Elektronik	€ 200	€ 1700
3	PIC	€ 20	€ 1720
4	Bluetooth Modul	€ 50	€ 1770
5	Binden der endgültigen Version	€ 120	€ 1890
6	Druckkosten für 500 Flyer	€ 40	€ 1930
-	Gesamtkosten		€ 1930

Tabelle 5: Geplante Aufwände

2.1.10.2 Tatsächliche Aufwände für Durchführung des Projektes

DIPLOMARBEIT "Vertical Farming": Kostenübersicht						
	Produktkategorie	Berechnung	Anzahl	Einzelpreis	Kosten (primär)	Kosten (tatsächlich)
Marketing	Website	Hosting + Domain für 1 Jahr DIN Lang Zack-Zack-Falz, 6-Seitig	1 €	15,00 €	15,00 €	- €
	Folder	Visitenkarte 8,5 x 5,5 cm, Beidseitig	1000 €	0,26 €	262,80 €	- €
	Visitenkarten	B&C E190, Weiss, Digitaldirektdruck 4x L, 1x XL	1000 €	0,15 €	149,90 €	692,70 €
	T-Shirts	B&C ID.002 80/20, Heather Grey, Digitaldirektdruck 3x L, 2x XL	5 €	20,00 €	100,00 €	- €
	Sweater	TOFT, Innovation Day und Aushang	5 €	30,00 €	150,00 €	- €
	Plakate	Stückliste siehe SharePoint "2.2.1_Pecnstahl Niro Stahl"	3 €	5,00 €	15,00 €	- €
	Niro Stahl	Stückliste siehe SharePoint "2.2.2_Robotunits Konstruktion V2"	1 €	350,00 €	350,00 €	295,20 €
	Robotunits Konstruktion (1 Etage)	Akryl	1 €	1.004,92 €	1.004,92 €	717,80 €
	Wasserbehälter	3D-Druck	1 €	15,00 €	15,00 €	- €
	Behälter für Säzlinge	3D-Druck	12 €	4,77 €	50,00 €	- €
Mechanik	Austrandschalter	Bilek & Schüll ausAkryl	4 €	2,25 €	9,00 €	- €
	Drehmechanismus	Deckel für Platinengehäuse	1 €	222,90 €	222,90 €	1.716,54 €
	Rollen	Anschluss für Wasserzuführ	4 €	8,99 €	35,96 €	- €
	Silikon	Deckel für Platinengehäuse	1 €	8,99 €	8,99 €	- €
	Plexiglas	Anschluss für Wasserzuführ	1 €	3,59 €	3,59 €	- €
	Fitting	Deckel für Platinengehäuse	1 €	13,99 €	13,99 €	13,99 €
	Polykarbonat	Anschluss für Wasserzuführ	1 €	2,19 €	2,19 €	- €
	High Power LED 3W	High Power LED 3W 12V DC, 1,3A	100 €	0,30 €	30,20 €	30,20 €
	Tauchpumpe	Max Schalt-I: 0,5A, Max. Schalt-U: 220V DC, Temp.: -10- + 85 Grad	1 €	13,99 €	13,99 €	- €
	Füllstandssensoren	MagiDeal 3W/2W 5-35V 700mA, PWM dimmbar	3 €	10,73 €	32,20 €	32,20 €
Elektronik	Konstantstromquellen	Kabel	10 €	2,14 €	21,40 €	21,40 €
	Schaltzeitteil	27V DC, 7,5A, 200W, IP20	1 €	20,00 €	20,00 €	20,00 €
	Leiterplattenklemme 2-polig	WAGO 2081-1202	2 €	1,61 €	3,22 €	3,22 €
	Leiterplattenklemme 6-polig	WAGO 2081-1226	2 €	2,01 €	4,02 €	4,02 €
	Leiterplattenklemme 12-polig	WAGO 2081-1232	4 €	2,71 €	10,84 €	- €
	Diverse Widerstände	Beta Layout	20 €	0,15 €	3,00 €	- €
	Platine	LED inkl. Fassung	1 €	89,90 €	89,90 €	- €
	Status-LED	2 Versch Bauteile	1 €	3,89 €	3,89 €	- €
	Lüfter	MOSFET	1 €	31,41 €	31,41 €	31,41 €
	Holzgeflecht	Infineon Technologies IRL520NPBF MOSFET 1 N-Kanal 48 W TO-220AB	4 €	2,58 €	10,31 €	- €
Etwäiges	Nährstoffpräparat	Von Herbeus Greens (1kg)	1 €	20,00 €	20,00 €	- €
	Nährstoffpräparat	BioBizz Bio Heaven (1L)	1 €	49,90 €	49,90 €	108,61 €
	Messbehälter für Präparat	BioBizz Bio Heaven (250mL)	1 €	18,79 €	18,79 €	- €
	Sponsoring	gesund.co.at	8 €	2,49 €	19,92 €	- €
Geldmittel	Sponsoring	Liapor Österreich	€ -	€ -	€ 600,00	600,00 €
	Sponsoring	Growland	€ -	€ -	€ 500,00	500,00 €
	Sponsoring	Hondl & Co.	€ -	€ -	€ 300,00	300,00 €
Summe: € 2.794,47						Summe: € 181,25

Abbildung 25: Tatsächliche Aufwände

Mithilfe des folgendem Kreisdiagrammes kann die Kostenaufteilung zur Durchführung dieser Diplomarbeit veranschaulicht werden. Der größte Teil der Ausgaben ist in dem Bereich der Mechanik und des Marketings mit ungefähr 75% wieder zu finden. Die Kostenaufwände für die Elektronik und für etwaig benötigte Sachmittel belief sich hierbei hingegen nur auf ungefähr 25%.

Dank der Unterstützung von Kooperationspartnern konnten 106,5% der Kostenaufwände gedeckt werden, d.h. es blieben neben den ausgegebenen Kosten noch 181,25€ für einen möglichen Verschleiß von Bauteilen über.

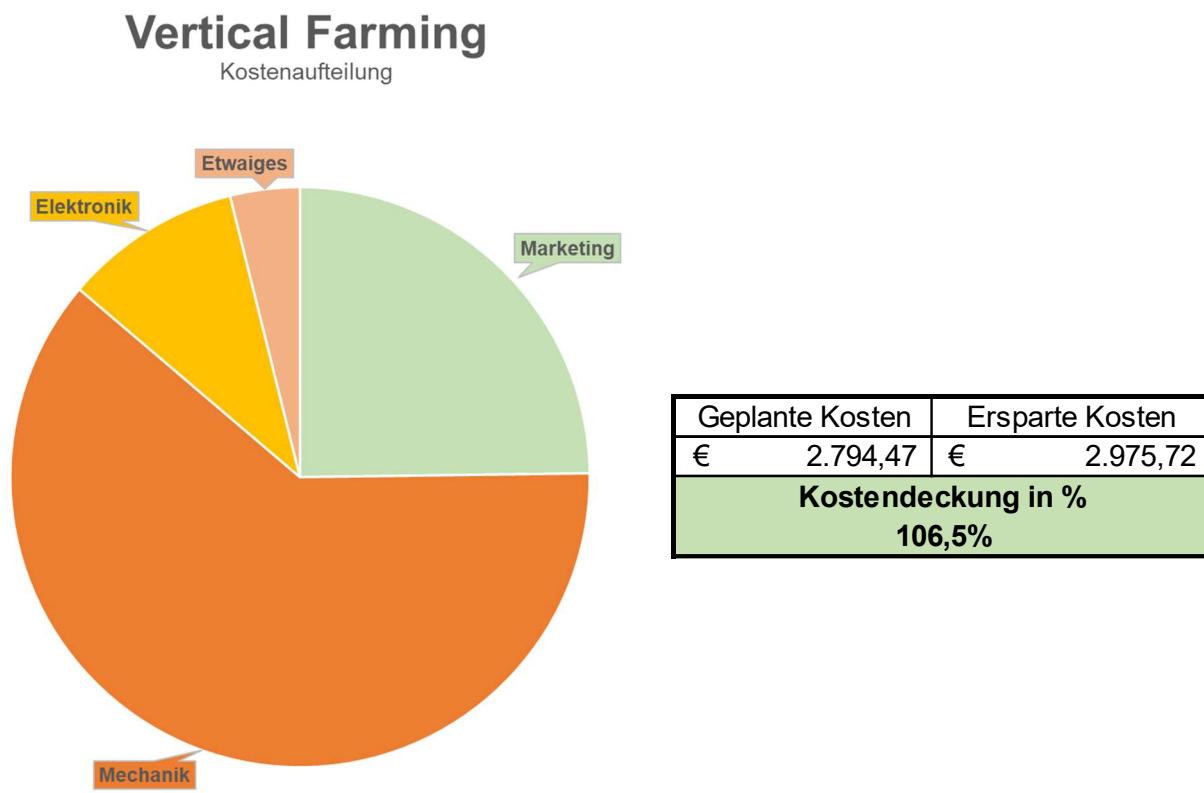


Abbildung 26: Kostenaufteilung

2.1.10.3 Kostendeckung

Die Kostendeckung wurde durch Sponsoren bzw. aus den Privatmitteln des Projektteams gewährleistet.

2.1.11 Externe Kooperationspartner

Unser Projekt wird von folgenden Unternehmen gesponsert:

2.1.11.1 Radio Höndl & Co.

Schlickgasse 4
A-1090 Wien
Österreich

office@hoendl.at



Abbildung 27: Logo "Höndl & Co."

Ein großer Dank geht an die Firma „Radio Höndl & Co.“ für eine finanzielle Unterstützung.

[Im Wert von 100,00€]

2.1.11.2 easynname GmbH

Fernkorngasse 10/3/501
A-1100 Wien
Österreich

office@easynname.com



Abbildung 28: Logo "easynname"

Das Projektteam bedankt sich herzlichst bei der Firma "easynname" für die Zurverfügungstellung der Domain: "vertical-farming.at". Dadurch konnte einerseits diese Website erstellt und auf das Projekt angepasste E-Mail-Adressen generiert werden.

[Im Wert von 14,90€]

2.1.11.3 Herz As Media & Communications

Arnikaweg 99
A-1220 Wien
Österreich
k.atefie@herz-as.com



Abbildung 29: Logo "Herz As Media"

Einen essenziellen Beitrag zum Projekt leistete die Unternehmensgruppe "Herz As Media". Diese unterstützte die Diplomarbeit mit einem Geldbetrag in der Höhe von 500€ zuzüglich 20% Umsatzsteuer und mit einem einheitlich gebrandeten Outfit (T-Shirt und Sweater).

Dadurch konnte sich das Projektteam in der Kostenplanung auf andere Bereiche konzentrieren. Durch das Branding wurde das Projekt noch besser repräsentiert.
[Im Wert von 800,00€]

2.1.11.4 Pechstahl

Friedrich Ganglberger Str.1-3
A-2514 Traiskirchen
Österreich
pech@pechstahl.at



Abbildung 30: Logo "Pechstahl"

Ein großes Dankeschön geht an das Unternehmern Pechstahl für den Verkauf der benötigten Rohteile zu Eigenerwerbspreisen. Sie ermöglichen dem Projektteam eine schnelle Zustellung als auch einen kostengünstigen Erwerb der Materialien.
[Im Wert von 150,00€]

2.1.11.5 PrintShop Sofortdruck- u. HandelsgmbH

Stadiongasse 5
A-1010 Wien
Österreich
kolarik@printshop.at



Abbildung 31: Logo "PrintShop"

PrintShop stellte der Diplomarbeit insgesamt 1000 Folder und 1000 Visitenkarten zur Verfügung. Dadurch konnte einerseits das Interesse an etlichen Präsentationen gesteigert werden, gleichzeitig konnte das Projektteam aber auch verschiedenste Unternehmen auf das Projekt aufmerksam machen.

[Im Wert von 463,00€]

2.1.11.6 Robotunits GmbH

Dr. Walter Zumtobel Str. 2
A-6850 Dornbirn
Österreich
michael.poszert@robotunits.com



Abbildung 32: Logo "Robotunits"

Ein großer Dank gilt außerdem dem Maschinenbauunternehmen Robotunits, welches die Diplomarbeit mit einem 40%-Rabatt auf Profil- und Verbindungstechnik unterstützt hat. Mit Hilfe dieser Bauteile konnten die Diplomarbeit den hohen Ansprüchen an Fertigungsqualität gerecht werden.

[Im Wert von 345,00€]

2.1.11.7 Bilek + Schüll GmbH

Seybelgasse 12a
A-1230 Wien
Österreich
service@bilek.at



Abbildung 33: Logo "Bilek + Schüll"

Außerdem möchte sich das Projektteam bei dem Unternehmen "Bilek + Schüll" für einen 20%-Rabatt auf unseren Drehmechanismus bedanken. Außerdem ermöglichen Sie es, die Lieferdauer sehr gering zu halten.

[Im Wert von 85,00€]

2.1.11.8 Herbeus Greens GmbH

Pysdorf 2
A-2281 Raasdorf
Österreich
ci@herbeusgreens.at



Abbildung 34: Logo "Herbeus Greens"

Herzlichen Dank an das Unternehmen "Herbeus Greens", welches die Diplomarbeit mit notwendigem Know-How unterstützt hat. Zudem stellten sie auch deren Holzgeflecht zur Verfügung, mit welchem getestet werden konnte, welchen Effekt die Diplomarbeit auf verschiedenste Pflanzen hat.

[Im Wert von 25,00€]

2.1.11.9 Beta LAYOUT GmbH

Im Aartal 14
D-65326 Aarbergen
Deutschland
mathias.bierwage@pcb-pool.com



Abbildung 35: Logo "Beta LAYOUT"

Das Projektteam bedankt sich zudem bei "Beta Layout" für die Bereitstellung von professionell gefertigten Platinen. Durch ihre Unterstützung ist es möglich die Platinen auf die benötigten Ansprüche anpassen zu können.

[Im Wert von 500,00€]

2.1.11.10 Lias Österreich GesmbH

Fabrikstraße 11
A-8350 Fehring
Österreich
bernd.hoerbinger@liapor.at



Abbildung 36: Logo "Liapor"

Wir wollen uns außerdem bei der Firma "LIAPOR Österreich" für deren finanzielle Unterstützung bedanken. Durch diese Unterstützung konnten wir qualitativ hochwertigere Komponenten verwenden.

[Im Wert von 500,00€]

2.1.11.11 HORNBACH Baumarkt GmbH

IZ NÖ-Süd, Str. 3, Obj. 64
A-2355 Wiener Neudorf
Österreich
rene.loeschnigg@hornbach.com



Abbildung 37: Logo "Hornbach"

Das Projektteam freut sich besonders über die Kooperation mit dem Baumarkt Hornbach. Dieser stellte der Diplomarbeit einerseits Werkzeuge, wie Bohrer und anderseits Materialien, wie Schläuche bzw. Klemmen zur Verfügung, wodurch eine zuverlässige Fertigung gewährleistet wurde.

[Im Wert von 180,00€]

2.1.11.12 Rome International GmbH & Co. KG

Hammer Deich 6-10
D-20537 Hamburg
Deutschland
ag@romeint.com



Abbildung 38: Logo "Rome International"

Der Webshop "Growland" unterstützt das Projekt mit finanziellen Mitteln als auch mit wichtigen Nährstoffen, welche für den Anbau der Pflanzen wichtig sind. Durch ihre Unterstützung war es dem Projektteam möglich, Produkte, welche für die Umsetzung des Projektes benötigt wurden, erwerben zu können.

[Im Wert von 300,00€]

2.1.12 Marketing

2.1.12.1 Website der Diplomarbeit

2.1.12.1.1 Bedienung des 3D-CAD Modells

Auf „B2B.Partcommunity“ wurde ein 3D-CAD Modell generiert, mit welchem es dem Nutzer möglich ist, sich den groben Aufbau der Vertical-Farming-Einheit von allen Perspektiven anschauen zu können. Der Vorteil hierbei ist, dass dies ohne jegliches Programm erfolgt, da das Fenster per Embedded-Code auf jede Internetseite eingefügt werden kann. Folgende Grafik ist auf der Projektwebsite zu sehen:

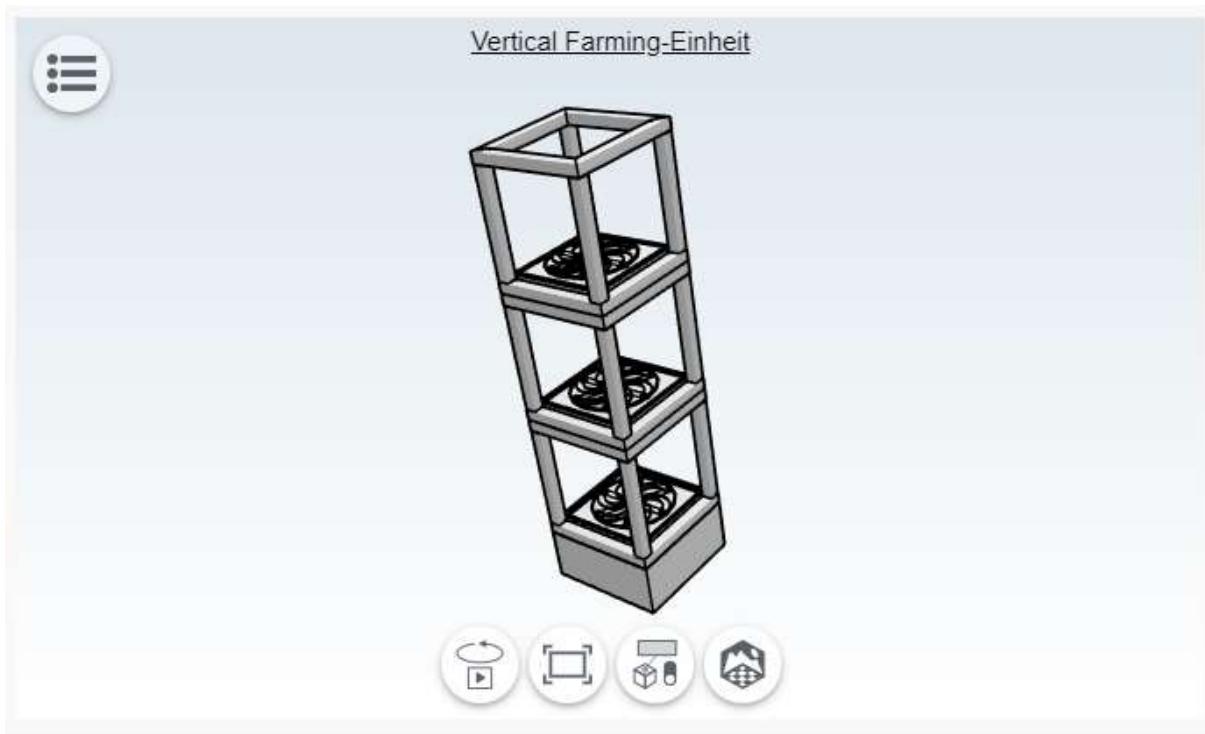


Abbildung 39: Vertical Farming-Einheit auf B2B.Partcommunity

Weiters ist es dem Interessenten möglich, folgende Adjustierungen in der 3D-Darstellung vorzunehmen.

Hierzu klickt man auf das Symbol, welches sich am linken oberen Fensterrand befindet. Anschließend sollte sich folgendes Menü öffnen:

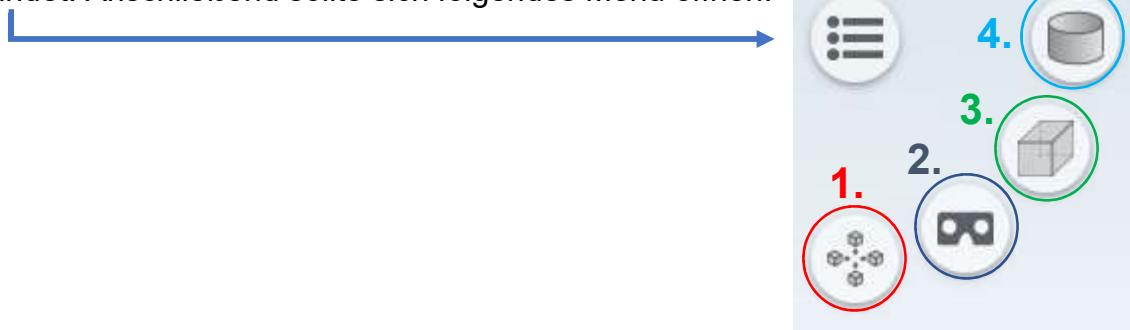


Abbildung 40: Aktionsfelder B2B.Partcommunity

- **1. Additional Functions**

- Cut
- Exploded View
- Fullscreen
- Teleport-Mode



Abbildung 41: Additional Functions B2B.Partcommunity

Durch obige Befehle ist es möglich sich das Modell im Detail anzusehen. Mit „Cut“ wird ein Koordinaten-System erstellt, welches einen Schnitt an einer beliebigen Stelle erstellt. Unter „Exploded View“ kann man sich die Konstruktion als Explosionsansicht darstellen lassen. Mit „Fullscreen“ und „Teleport-Mode“ kann man die Größe und die Position der Kamera ändern.

- **2. Virtual Reality**

- Anaglyph
- Dreamoc HD3
- Dreamoc XL2
- Holographic Pyramid
- Bluebox



Abbildung 42: Virtual Reality B2B.Partcommunity

Hier kann man sich das Modell in verschiedenste Virtual Reality-Ansichten konfigurieren lassen.

- **3. Model Position**

- Isometric View
- Animation



Abbildung 43: Model Position B2B.Partcommunity

Unter „Isometric View“ kann man sich das Modell aus einer vorgenerierten Perspektive ansehen. Der Befehl „Animation“ dient dazu, dass sich das Modell permanent um 360° dreht.

- **4. Model Shading**

- Edges
- Shaded
- Shaded with edges



Abbildung 44: Model Shading B2B.Partcommunity

Mit obigen Befehlen kann man die Darstellung des Modells ändern. Bei „Edges“ sieht man die Kanten, bei „Shaded“ die Schattierung und bei „Shaded with edges“ die Schattierung mit Kanten des Modells.

Weiters befinden sich am unteren Bildschirmrand sogenannte „Shortcuts“, mit welchem es dem Nutzer einfacher fällt, die wichtigsten Befehle auf einen Blick zu haben.



Abbildung 45: Shortcuts B2B.Partcommunity

- **1. Animation**

Wie schon beschrieben, dient der Befehl „Animation“ dazu, sich das Modell dauerhaft um 360° rotieren zu lassen.

- **2. Fullscreen**

Mit „Fullscreen“ kann das Fenster auf Vollbild umgestellt werden.

- **3. Show Hotspots**

Hier kann man sich eine Info-Box des Modells ausgeben lassen, welcher den Link des Modells zu B2B.Partcommunity beinhaltet.

- **4. Environment**



Abbildung 46: Environment B2B.Partcommunity

Unter „Environment“ kann die Umgebung, der Hinter- als auch der Untergrund geändert werden.

Da ein detaillierter Aufbau der Konstruktion wesentlich schwerer nachvollziehbar wäre, wurde nur der grobe Entwurf als 3D-Modell hochgeladen. Dennoch befinden sich Bilder, mit den genauen Komponenten direkt darunter.

2.1.12.2 Folder

Um am Tag der offenen Tür, bei Wettbewerben als auch bei Kooperationspartnern Interesse wecken beziehungsweise Informationen hinterlassen zu können, wurden auf das Projekt angepasste Folder erstellt. Die Intention dieser Drucksorten war es, jeder Person einen kleinen Überblick über diese Diplomarbeit geben zu können, ohne diesen etwas erklären zu müssen. Bei der Art des Folders handelt es sich um den „Folder DIN lang 6-Seitig Zick-Zack-Walz“.

Anhand folgender Grafik kann der Aufbau des Folders nachvollzogen werden:

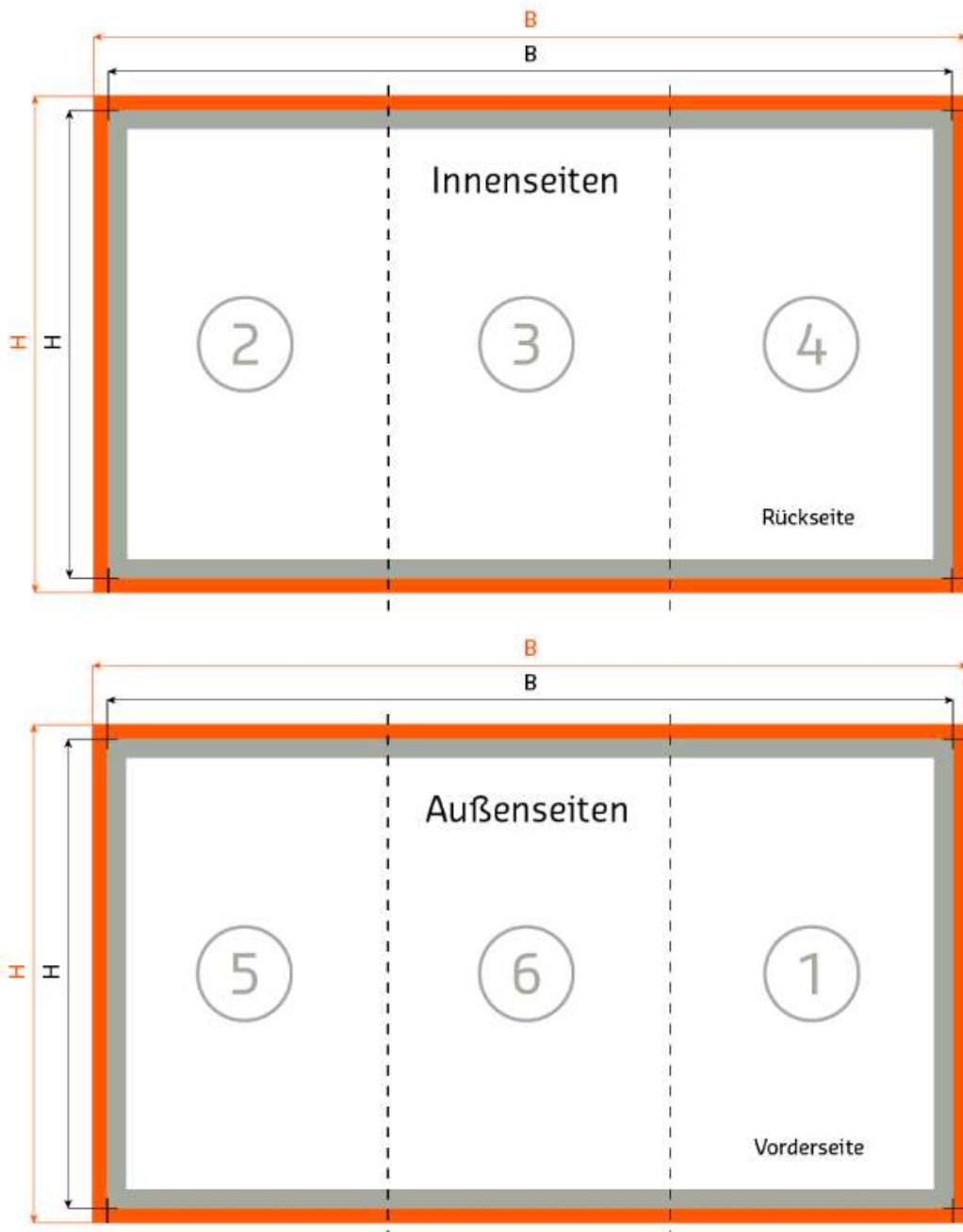


Abbildung 47: Aufbau des Folders

2.1.12.2.1 Innenseite (mit 1 mm Schnittrand)

Wie funktioniert unser Projekt?

Konstruktion individuell anpassbar

Die Konstruktion besteht aus einem Grundgerüst, einem Wasserreservoir, sowie zumindest einem Anpflanzbehälter. Es wird möglich sein, die Konstruktion dem Einsatzzort anzupassen. Man kann sich hier zwischen einem, zwei oder drei Anpflanzbehältern entscheiden.

(Max. Abmaße: B 50cm x L 50cm x H 150cm)

Alles im Überblick dank Android-App

In dem Wasserreservoir befinden sich Sensoren, welche den Wasserstand angeben. Wenn zu wenig Wasser im Reservoir vorhanden ist, wird eine Warnung auf das Handy geschickt oder direkt am Gerät mit Hilfe von Status-LED's angezeigt.

Per Handy steuerbar

Die Bewässerung sowie die Beleuchtung der Anpflanzbehälter, sollen per Bluetooth über eine Handy-App steuerbar sein und optional vollautomatisch erfolgen.

Für was steht unsere Diplomarbeit?

	Frisch		Gesund		Lokale Produktion
	Nachhaltig		Ganztägig verfügbar		Herbizidfrei

Warum genau „Vertical Farming“?

Aufgrund des zunehmenden Bevölkerungswachstumes, müssen immer mehr Menschen mit Lebensmitteln, unter anderem auch mit „Obst und Gemüse“, gesättigt werden.

Gibt es außer „Vertical Farming“ andere Möglichkeiten dieser Nachfrage gerecht zu werden?

- Kann man die Anbauflächen der Landwirtschaft noch effizienter nutzen bzw. vergroßern?
Leider ist dies nicht möglich, da diese bereits zu 80% ausgeschöpft und somit kaum erweiterbar sind.
- Kann man Wälder roden, um mehr Platz für diverse Acker und Anbauflächen zu schaffen?
Da diese Bäume lebenswichtigen Sauerstoff ausstoßen, ist dies nicht möglich.
- Kann man den Ertrag der Ernte durch genmodifizierte Pflanzen erhöhen?
Man weiß nicht, wie der menschliche Körper auf solche Mutationen reagieren würde, da es über dieses Thema noch keine wissenschaftlichen Studien gibt.
Aufgrund von möglichen Nebeneffekten wird diese Möglichkeit auch ausgeschlossen.

Druck: www.printshop.at

Abbildung 48: Folder "Vertical Farming" Außenseite

2.1.12.2.2 Außenseite (mit 1 mm Schnittrand)

Konnten wir Ihr Interesse wecken?	<p>Was macht unser Projekt innovativ?</p> <p><u>Schwierigkeiten beim Vorstellen?</u></p> <p>Unter folgendem „Thingmark“ können Sie sich, unsere Konstruktion als 3D CAD-Modell ansehen. Hier können Sie unsere Vertical Farming-Einheit aus beliebig vielen Perspektiven beobachten.</p> <p>Laden Sie sich „Vuforia View“ aus Ihrem Appstore herunter und scannen Sie folgendes Symbol:</p> 	<p>Landwirtschaft für Jedermann</p> 	<p>Eigens entwickelte Erntetechnik</p> <p>Sie vereinfacht das Ernten, da man die bereits erntefertigen Salatköpfe durch eine Drehung von 60° gegen den Uhrzeigersinn nach außen drehen kann. In den inneren 6 „Löchern“ können anschließend neue Sämlinge angepflanzt werden.</p> 	<p>Warum ist dieser Zyklus so wichtig?</p> <p>Der Zyklus wird benötigt um mehr Pflanzen auf engem Raum züchten zu können. Da unser System vertikal aufgebaut ist, wird „Platz sparen“ bei uns großgeschrieben!</p>	<p>Welche Vorteile hat der Mechanismus?</p> <ul style="list-style-type: none"> • Geringer Platzbedarf • Einfaches Wechseln der Pflanzensorte • Mehr Anpflanzmöglichkeiten bei gleichbleibender Fläche • Bis zu 36 Pflanzen auf 0,5m² Standfläche (Bei 3 Etagen)
<p>Abbildung 49: Folder "Vertical Farming" Innenseite</p>	<p>Philip Gasser Projektleiter Email: philipp.gasser@vertical-farming.at</p> <p>Wir bieten Ihnen bzw. Ihrem Unternehmen attraktive Gegenleistungen!</p> <p> www.vertical-farming.at  support@vertical-farming.at</p>	 Vertical Farming Eine Diplomarbeit der HTL Rennweg			

2.1.12.3 Visitenkarten

Um Kooperationspartnern und Interessenten eine Möglichkeit zur Kontaktaufnahme bieten zu können, wurde eine Visitenkarte erstellt. Da es die Aufgabe eines Projektleiters ist, mit externen Personen und Unternehmen in Kontakt zu treten, wurden auf der Visitenkarte nur Informationen des Projektleiters vermerkt.

Durch Drucksorten, wie Folder und Visitenkarten, konnte die Seriosität des Projekts maßgeblich angehoben werden.

2.1.12.3.1 Innenseite (mit 1 mm Schnittrand)



Abbildung 50: Visitenkarte "Vertical Farming" Innenseite

2.1.12.3.2 Außenseite (mit 1 mm Schnittrand)



Abbildung 51: Visitenkarte "Vertical Farming" Außenseite

2.1.12.4 Textilien

2.1.12.4.1 T-Shirts

Um diese Diplomarbeit auf Messen und diversen Veranstaltungen Best als möglich repräsentieren zu können, wurden T-Shirts mit dem Projektlogo und dem Namen des jeweiligen Projektmitglieds bedruckt. Bei der Druckart handelt es sich um einen Digitaldruck.

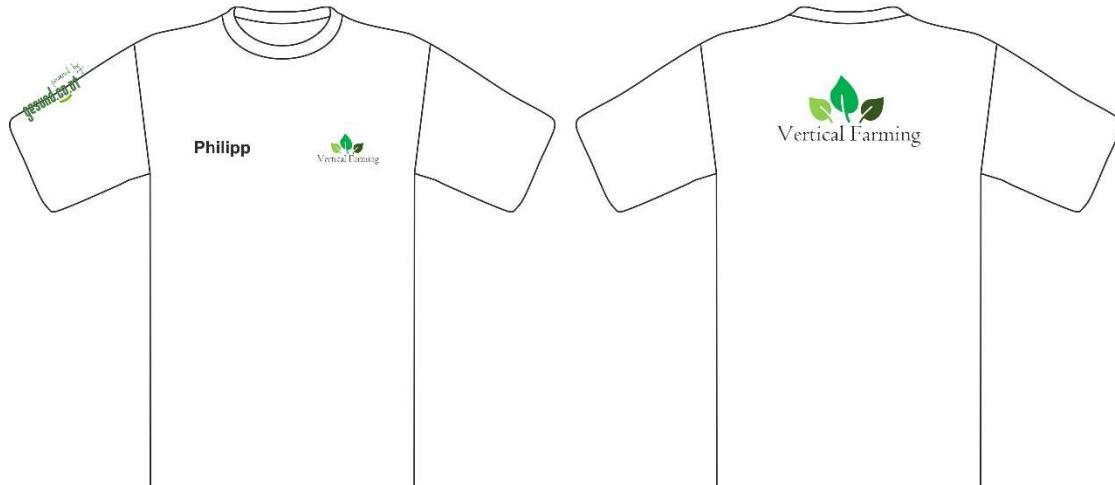


Abbildung 52: T-Shirts "Vertical Farming"

2.1.12.4.2 Sweater

Neben T-Shirts wurden auch Sweater, ebenfalls mit dem Projektlogo versehen, hergestellt. Der Sweater wurde jedoch nicht dem Namen des jeweiligen Projektmitgliedes bedruckt. Bei der Druckart handelt es sich, wie auch bei den T-Shirts, um einen Digitaldruck.



Abbildung 53: Sweater "Vertical Farming"

2.1.13 Instandhaltungskosten der „Vertical Farming“ Einheit

Neben den Materialkosten ist es auch wichtig die Instandhaltungskosten dieser Diplomarbeit zu eruieren, um den Nutzer die Kosten für den Betrieb mitteilen zu können. Neben den Instandhaltungskosten wird sich in den folgenden Berechnungen auch der Umsatz und der Gewinn/Verlust berechnet. Dies ist nötig, da sich der Nutzer dazu entschließen könnte, diese Diplomarbeit auf längere Dauer zu nutzen und damit statt einem Privatkonsum einen wirtschaftlichen Effekt erzielen möchte. Neben den Energiekosten wurden auch die Kosten des Nährstoff-Substrates miteinbezogen.

2.1.13.1 Kosten bei Verwendung von 1 Etage

Zuallererst musste die maximale Leistung eruiert werden, welche sich auf 0,2kW (200W) belief. Anschließend wurde diese Leistung bei 1 Etage mit 1/3, bei 2 Etagen mit 2/3 und bei 3 Etagen mit 3/3 multipliziert. Das Produkt aus dieser Berechnung ergibt die Leistung bei der Verwendung von einer, zwei oder drei Etagen:

$$P_{Max} := 0.2 \text{ kW} \cdot \frac{1}{3} = 0.067 \text{ kW}$$

Nachdem die benötigte Leistung feststeht, müssen die durchschnittlichen Kosten für 1kWh recherchiert werden:

$$Kosten_{1kWh} := 0.27 \frac{\text{€}}{\text{kW} \cdot \text{hr}}$$

Anschließend werden die Stunden errechnet, welche die „Vertical-Farming“ Einheit benötigt, bis sie 1kWh verwendet hat. Hierbei wird 1kWh durch die maximale Leistung dividiert:

$$t_{1kWh} := \frac{1 \text{ kW} \cdot \text{hr}}{P_{Max}} = 15 \text{ hr}$$

Danach muss der Verwendungszeitraum bestimmt werden, in diesem Fall läuft die Einheit einen Monat lang permanent durch:

$$t_{1Monat} := 30 \text{ day} = 720 \text{ hr}$$

Um dann wiederum auf die Wattstunden, also die kWh zu kommen, wird die Dauer des Monats mit 1kWh multipliziert und durch die Dauer, welche benötigt wird, um auf 1 kWh zu kommen, dividiert:

$$P_{1\text{Monat}} := \frac{t_{1\text{Monat}} \cdot 1 \cdot \mathbf{kW} \cdot \mathbf{hr}}{t_{1\text{kWh}}} = 48 \mathbf{kW} \cdot \mathbf{hr}$$

Die Aussage obiger Formel erläutert, dass die „Vertical-Farming“ Einheit, unter Verwendung einer Etage maximal 48 kWh benötigt.

Um nun schlussendlich zu den Kosten für den Stromverbrauch eines Monates zu kommen, werden die bereits vorher berechneten Wattstunden mit den Kosten pro kWh multipliziert. Als Ergebnis erhält man 12,96€.

$$\mathbf{Kosten}_{Energie1\text{Etage}} := P_{1\text{Monat}} \cdot \mathbf{Kosten}_{1\text{kWh}} = 12.96 \mathbf{\epsilon}$$

2.1.13.2 Kosten bei Verwendung von 2 Etagen

Ebenso, wie bei der Berechnung der Kosten einer Etage, gelten hier alle vorherig angeschriebenen Berechnungen, bis auf die maximale Leistung P_{Max} , abhängig davon die benötigte Leistung bei der Verwendung eines Monats P_{1Mo} und folglich davon die Kosten.

$$P_{Max} := 0.2 \mathbf{kW} \cdot \frac{2}{3} = 0.133 \mathbf{kW}$$

$$P_{1\text{Monat}} := \frac{t_{1\text{Monat}} \cdot 1 \mathbf{kW} \cdot \mathbf{hr}}{t_{1\text{kWh}}} = 96 \mathbf{kW} \cdot \mathbf{hr}$$

$$\mathbf{Kosten}_{Energie2\text{Etagen}} := P_{1\text{Monat}} \cdot \mathbf{Kosten}_{1\text{kWh}} = 25.92 \mathbf{\epsilon}$$

Letztendlich müssen 25,92€ pro Monat bezahlt werden, um den benötigten Stromverbrauch für die Verwendung von 2 Etagen abdecken zu können.

2.1.13.3 Kosten bei Verwendung von 3 Etagen

$$P_{Max} := 0.2 \text{ kW} \cdot \frac{3}{3} = 0.2 \text{ kW}$$

$$P_{1\text{Monat}} := \frac{t_{1\text{Monat}} \cdot 1 \text{ kW} \cdot \text{hr}}{t_{1\text{kWh}}} = 144 \text{ kW} \cdot \text{hr}$$

$$\boxed{Kosten_{Energie3Etagen} := P_{1\text{Monat}} \cdot Kosten_{1\text{kWh}} = 38.88 \text{ €}}$$

Bei der maximalen Auslastung des Systems, also der Verwendung von 3 Etagen, müssen 38,88€ pro Monat an Stromkosten berücksichtigt werden.

2.2 Mechanik & Konstruktion

2.2.1 Erstes Konzept

2.2.1.1 Aufbau: Erstes Konzept

Das erste Konzept bestand aus ca. 80% Stahl, 15% Aluminium und etwa 5% Kunststoff bzw. verschiedenen Verbundmaterialien. Es war geplant, die Konstruktion aus Stahlprofilen in einer Bügelform zu verschweißen und anschließend U-Profile als Halterungsmöglichkeiten für die Anpflanzbehälter auf deren Oberfläche anzubringen. Durch den verschweißten Leiterrahmen lag die maximale Anzahl an Behälter für den Anbau bei 3 Stück. Die Anpflanzbehälter waren als gebogene und verschweizte Boxen aus Niro geplant, welche zur Stabilisierung des Korpus dienten und Halt für die späteren Nährbodenträgerwannen gaben.

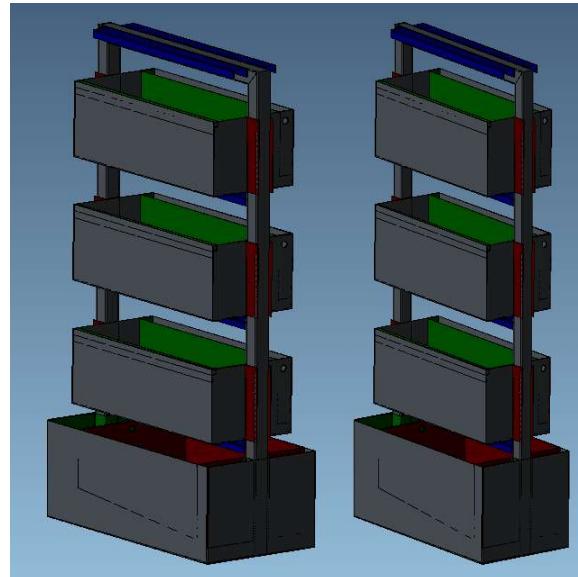


Abbildung 54: Erstes Konzept

2.2.1.2 Beleuchtung und Elektronik-Führung: Erstes Konzept

Die gesamte Elektronik sowie das Wasserreservoir befand sich im unteren Teil in einer Aluminium Wanne, welche auch als Halterungsgrund für das Bügelgerüst diente. Der Zugang zur Elektronik und Bewässerungseinheit war durch eine Öffnung in der Aluminium Wanne geplant. Des Weiteren waren diverse Öffnungen für die Führung der Elektronik der Wachstumsbeleuchtung bzw. der Bewässerung im Bügelrahmen vorgesehen. Beleuchtet sollte das System über Wachstumslampen werden, welche sich zur Kühlung in Aluminium Hohlprofilen befanden und mittig über jedem Anpflanzbehälter angebracht waren.

2.2.1.3 Bewässerung: Erstes Konzept

Der Bewässerungstyp dieser Einheit stellte die sogenannte Kapillarbewässerung dar, welche das Wasser aus dem Wasserreservoir in den obersten Anpflanzbehälter förderte. Dort wurde dieser bis zu einem gewissen Level gefüllt und anschließend konnte das Wasser in den darunter Liegenden abfließen. Durch die Füllung des Anpflanzbehälters auf ein gewisses Maß sollte gewährleistet werden, dass der Nährboden befeuchtet wird und durch seine kapillaren Eigenschaften das Wasser bis zu den Samen bzw. den Wurzeln der daraus entstandenen Pflanzen fördert.

2.2.1.4 Anpflanzen: Erstes Konzept

Bei dieser Umsetzungsvariante sollte der Anbau etwaiger Gemüse und Obstsorten in eigenen PVC Behältern erfolgen - den sogenannte Nährbodenträgerwannen. Hierbei war ein Grundmodell in der Anpflanzwanne befestigt, in welches zusätzlich noch 4 kleine kippbare herausnehmbare Container, welche mit dem Nährboden geflechtet befüllt waren, in das Grundmodell gesteckt worden.



Abbildung 55: Nährbodenträgerwanne

2.2.1.5 Hintergründe für den ersten Konzeptwechsel

Die Hauptgründe, warum sich das Projektteam für einen Wechsel entschied, sind einerseits das enorme Gewicht von knappen 140kg ohne zusätzliches Gewicht von Elektronik, Wasser und Nährstoffmaterial und andererseits die hohen Materialanschaffungskosten von **350,40€**. Zuzüglich zu den Kosten für die diversen Eisenwaren kommen hierbei selbstverständlich die Kosten für die Elektronik bzw. für etliche Nebenmaterialien, die für die Fertigung benötigt würden hinzu. Ein weiterer Grund für die Änderung des Konzepts ist der Innovationsgrad, da das Projekt nicht wirklich benutzerfreundlich bzw. vermarktbar erschien und somit den Nutzen einer solchen Einheit nicht erfüllen konnte. Der endgültige Ausschluss des Projekts ist dem massiven Arbeitsaufwand geschuldet,

Pos	Beschreibung	Mwst	Einzelpreis	Rabat	Menge	Summe
1	Blech 1.4301 kaltgewalzt 2B (IIIC) EN 10088-2 1 TFL 2500 x 1250 x 2,0 mm (für Pos.1+3)	3,00	0,00	49 kg	€ 147,00	
2	Blech 1.4301 kaltgewalzt 2B (IIIC) EN 10088-2 1 TFL 2000 x 1000 x 2,0 mm (für Pos.2+9)	3,00	0,00	32 kg	€ 96,00	
3	AL-U-Profil EN-AW 6060/6063 4 x Stabänge 704 mm U 20 x 60 x 20 x 2 mm (für Pos.4 ?)	0,00	0,00	1,5 kg		
4	Sägekosten	0,00	0,00	4 Stk.		
5	Formrohr aus Aluminium EN-AW 6060/6063 2 x Länge 1500 mm 1 x Länge 804 mm 50 x 50 x 2,0 mm (für Pos.6+7)	0,00	0,00	4 kg		
6	Sägekosten	0,00	0,00	3 Stk.		
7	Feinblech S235 / DC01, kaltgewalzt EN 10025, EN 10130 1 TFL 2500 x 1250 x 2,0 mm (für Pos.8+10)	1,00	0,00	49 kg	€ 49,00	
						Netto 292,00 € 20% MwSt 58,40 € Gesamtbetrag € 350,40 €

Abbildung 56: Rechnung Eisenwaren Erstes Konzept

da allein für die Mechanik des Konzepts der Großteil der vorhandenen Zeit verloren gegangen wäre, da jedes Teil von dem Projektteam extra angefertigt werden müsste.

2.2.2 Zweites Konzept

2.2.2.1 Aufbau: Zweites Konzept

Dieses Konzept bestand aus ca. 70% Aluminium, 20% Kunststoff und etwa 10% Stahl. Hierbei war geplant, dass das Gerüst aus einer unteren Einheit und ein bis zu drei erweiterbaren oberen Einheiten besteht. Die untere Einheit sollte eine Box aus Aluminium sein, in welcher das Wasserreservoir aus Niro und die Platine mit diverser Elektronik Platz finden sollte. Zugänglich sollte die Elektronik und der Wasserbehälter über eine Zugangstür ermöglicht werden. Die obere Einheit sollte eine quadratische Form haben und deren Gerüst aus verschweißten

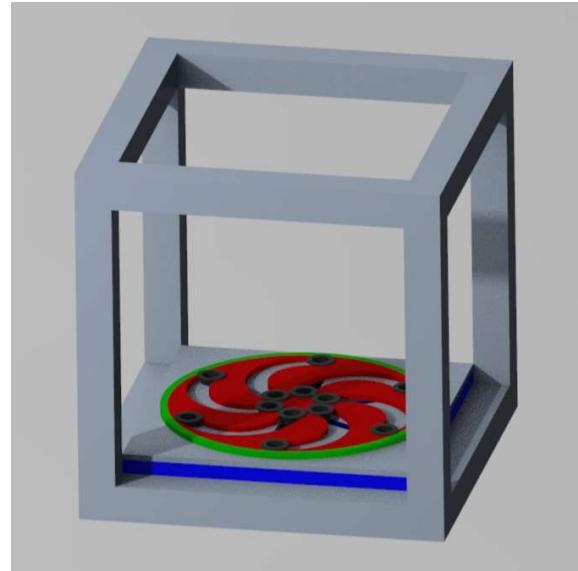


Abbildung 57: Zweites Konzept

Aluminium Profilen bestehen. Es war geplant die Wände mit Plexiglasplatten zu verkleiden, wobei eine herausnehmbar sein sollte, um den Zugang zu den Pflanzen zu ermöglichen. Weiters sollte auf die Oberseite der oberen Einheit eine Platte montiert werden, unter welcher sich die Beleuchtung und Sensorik zur Ermittlung der Temperatur und Luftfeuchtigkeit befand.

2.2.2.2 Beleuchtung und Elektronik-Führung: Zweites Konzept

Da ein Sponsoring einer deutschen Firma vorhanden war, welche ein fertiges Beleuchtungssystem zur Verfügung gestellt hätte, wäre die Beleuchtung der Pflanzen bei diesem Konzept nicht vom Projektteam entwickelt worden.

Das System dieser Firma hat ein quadratisches Außenmaß von 250x250mm, mit einer Leuchtleistung von 600W. Diese Leuchtleistung wäre für dieses Projekt jedoch völlig überdimensioniert gewesen. Des Weiteren hätte es durch die interne Programmierung und Verkabelung keine Möglichkeit für das Team gegeben, ohne Schaden der Originalsoftware in das System einzugreifen, um wie im Antrag beschrieben die Helligkeit und Laufzeit der Beleuchtung zu steuern. Die Kabelführung des Temperatur- und Luftfeuchtigkeitssensors sowie die Stromzufuhr für die Beleuchtung sollte bei diesem System in den Aluminium Profilen verlaufen und seine Zufuhr über die Platine in der unteren Einheit bekommen.

2.2.2.3 Entwicklung: Anpflanzmechanismus

Für das Konzept wurde, um die Fläche optimal nutzen zu können, ein eigener Anpflanzmechanismus - das „Drehsystem“ - entwickelt. Basierend auf dem Mechanismus eines Tresorschlosses, bestand es aus zwei Teilen, einem Unteren, bei dem die Linienführung gerade war und einem Oberen, bei dem die Linienführung ein angenäherter Kreisbogen war, in welchen anschließend die Samenbehälter gesteckt wurden. Diese gegensätzliche Linienführung ermöglichte es, dass sich bei einer Drehung der oberen Platte gegen den Uhrzeigersinn die Samenbehälter von innen nach außen bewegten. Der Kreisbogen wurde bei dieser Umsetzung mit zu kleinen Radien angenommen, wodurch bei der Rotation die Komplikation auftrat, dass die inneren Samenbehälter bei der Drehung nach außen aneckten. Dies lag daran, dass die Tangential bzw. Reibkraft zwischen Samenbehälter und der oberen Drehplatteneinheit zu groß wurde, wodurch der ganze Drehmechanismus seinen Nutzen verlor. Aufgesetzt wurde das Ganze auf eine aus Niro gebogene Wanne, in die das Wasser für die Bewässerung der Pflanzen gepumpt wurde.

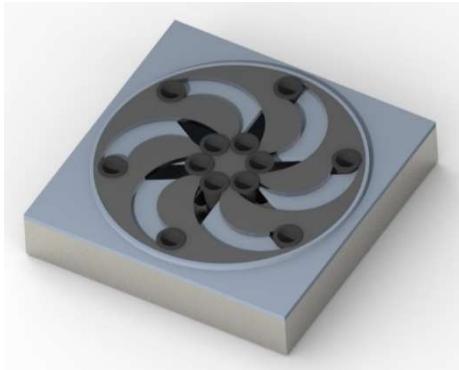


Abbildung 58: Drehmechanismus erster Entwurf

2.2.2.4 Entwicklung: Samenbehälter

Die erste Version des Samenbehälters hatte eine Höhe von 70mm und wurde mittels 3D-Drucker gefertigt. Seitlich hatte er 12 Löcher, die für die Bewässerung zuständig gewesen waren, jedoch wurden diese zu hoch angenommen, wodurch eine zu hohe Menge an Wasser zur Pflanze gelangt ist. Die Auflagefläche, mit der der Halt des Samenbehälters auf dem Drehsystem gewährleistet wird, war für die Last, mit der dieser befüllt wird zu groß gewählt was auf Kosten des Anpflanzvolumens ging. Eine weitere Verkleinerung des Anpflanzvolumens brachte die Wandstärke des Samenbehälters mit sich, da diese bei der ersten Version bei 3,5mm lag.

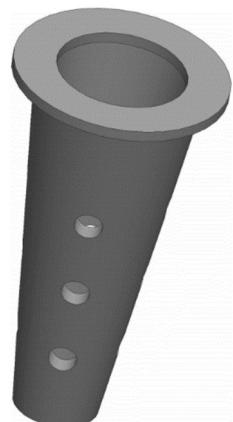


Abbildung 59:
Samenbehälter V1

2.2.2.5 Bewässerung: Zweites Konzept

Bei diesem Konzept hätte die Bewässerung der Pflanzen zwar dem Hydroponik Prinzip entsprochen jedoch wären die Schläuche im Profil vom Wasserreservoir bis zur obersten Einheit verlaufen. Das Wasser sollte über einen Anschlussstutzen, welcher sich im Deckel der Rotationseinheit befunden hätte in die Anpflanzwanne fließen, dort angelangt füllt es die Anpflanzwanne 5-10mm über die Unterkante des Samenbehälters. Um in die darunter liegende Einheit zu gelangen hätte das Wasser ein kleineres Loch passieren müssen, welches so gewählt war, dass die benötigte Wasserhöhe erreicht wird. Wenn es schließlich alle drei Einheiten passiert hätte wäre es erneut in das Wasserreservoir geflossen.

2.2.2.6 Entscheidung für den zweiten Konzeptwechsel

Die Gründe weshalb sich für einen erneuten Konzeptwechsel entschieden wurde, sind zum einen der extreme Preis für die Anschaffung der Rohmaterialien von 392,40€ und zum anderen, das obwohl Aluminium verwendet wurde, hohe Gewicht von 159kg. Der Grund für die Gewichtzunahme im Vergleich zum ersten

Konzept ist, dass hierbei zwar leichteres, im gleichen Zuge allerdings auch viel mehr Material verbaut werden sollte. Hierbei ist das Gewicht allerdings wieder im leerzustand, also ohne Elektronik, Wasser und Nährstoffmaterial bemessen worden. Wie beim ersten Konzept fallen auch hier wieder Kosten für die Fertigung, die Elektronik oder Nebenmaterialien an. Der endgültige Ausschluss dieser Projektidee ist hierbei erneut dem massiven Arbeitsaufwand geschuldet, da allein für die Mechanik des Konzepts der Großteil der vorhandenen Zeit verloren gegangen wäre, weil jeder Teil des Korpus bzw. andere mechanische Elemente eigens gefertigt werden hätten müssen.

Pos	Beschreibung	Mwst	Einzelpreis	Rabatt	Menge	Summe
1	Formrohr aus Aluminium EN-AW 6060/6063 24 x Länge 510 mm 12 x Länge 410 mm 50 x 50 x 5 mm (für Pos.1+2)	2,00	0,00	42 kg	€ 84,00	
2	Sägekosten	0,00	0,00	36 Stk.		
3	Blech 1.4301 kaltgewalzt 2B (IIIC) EN 10088-2 1 TFL 2500 x 1250 x 2,0 mm (für Pos.3,4,6)	3,00	0,00	49 kg	€ 147,00	
4	Blech 1.4301 kaltgewalzt 2B (IIIC) EN 10088-2 1 TFL 2000 x 1000 x 2,0 mm (für Pos.5)	3,00	0,00	32 kg	€ 96,00	
						Netto 327,00 € 20% MwSt 65,40 € Gesamtbetrag € 392,40 €

Abbildung 60: Rechnung Eisenwaren erstes Konzept

2.2.3 Drittes Konzept

2.2.3.1 Aufbau: Drittes Konzept

Das dritte Konzept ähnelt in Aufbau und Form dem zweiten, da auch dieses eine rechteckige Standfläche mit 50x50cm aufweist, allerdings nur aus 60% Aluminium, 30% Kunststoff und 10% Stahl besteht. Um den Arbeitsaufwand zu minimieren und die Montierbarkeit für den Endverbraucher für eine etwaige spätere Vermarktung zu erleichtern, entschied sich das Projektteam, anstatt der 4-Kant Holprofile sogenannte Robot Units Profile zu verwenden. Die optionale Erweiterung auf mehr Etagen bzw. eine Einheit zu erhöhen, um größere Pflanzen zu setzen wird ebenfalls durch die schnelle Montierbarkeit erleichtert. Bei diesem Konzept wurde für den oberen Korpus anstatt einer herausnehmbaren Plexiglasscheibe eine Zugangstür geplant, welche allerdings wie die umliegenden Seiten mit Plexiglasscheiben verkleidet wird. Die untere Korpus Einheit verfügt ebenfalls über eine Zugangstür, um zu den Elektronik Teilen und dem Wasserbehälter zu gelangen.

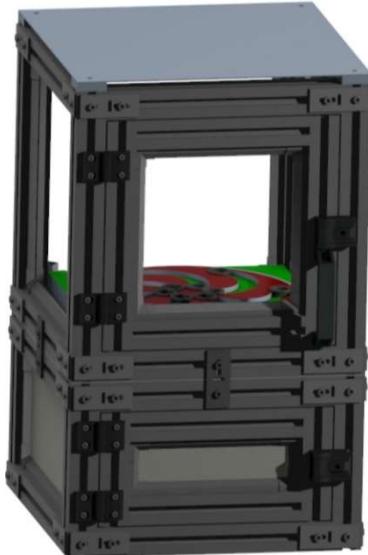


Abbildung 61: Drittes Konzept

2.2.3.2 Bewässerung: Drittes Konzept

Bei dem dritten Konzept wird ebenfalls das Hydroponik Bewässerungskonzept umgesetzt, allerdings wird hierbei auch die „NFT-Methode“ angewandt. Das heißt, das auf Erde verzichtet werden kann, die Wurzeln nur in einem Nährstoff angereichten Wasser hängen und die Pflanze an sich in einem Nährbodengeflecht wächst, somit bekommt die Pflanze genug Wasser, Sauerstoff und Nährstoff. Die Wasserversorgung wird durch Pumpen gewährleistet, sobald die Pumpe ihren Betrieb beginnt, wird Wasser von unten über einen Anschlussstutzen in die Anpflanzwanne gefördert. Dort angelangt füllt sie diese bis knapp über das zweite Bewässerungsloch der Samenbehälter, anschließend soll sie mit einer gewissen Nachlaufleistung ihren Betrieb fortsetzen, was wiederum sicherstellt, dass das Wasser nicht zu schnell abfließen kann.

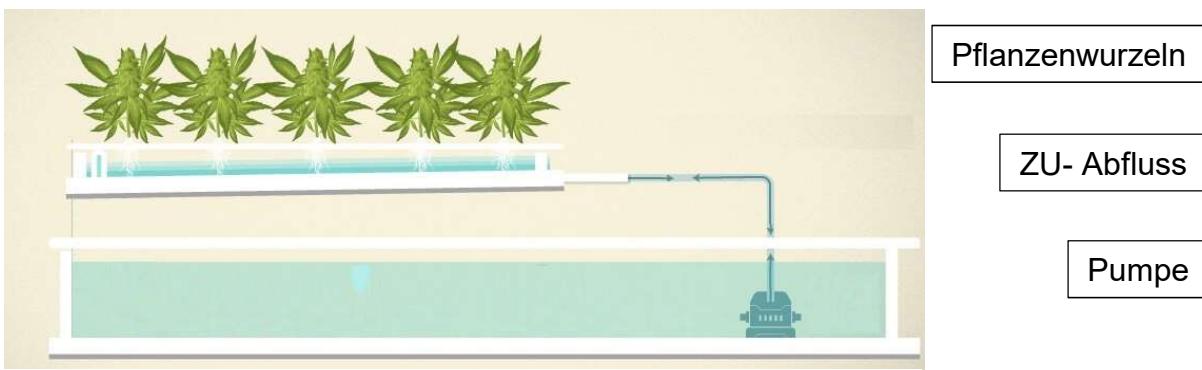


Abbildung 62: Hydroponik/ NFT

2.2.3.3 Beleuchtung und Elektronik- Führung Drittes Konzept

Die Beleuchtung der Pflanzen wurde bei diesem System auf die Pflanzen bzw. das Drehsystem angepasst geplant, dass heißt es wurde so konzipiert, dass sich über jeder Pflanze eine Beleuchtungs- LED befindet. Es sollen sogenannte High Power Leds verwendet werden, welche mittels Wärmeleitendem Doppelseitigem Klebeband auf eine Aluminium Platte aufgetragen werden. Auf der Oberseite der



Abbildung 63: Angepasste Beleuchtung

Platte soll sich zur Kühlung der Leds bzw. des Systems ein Lüfter befinden und auf einem der senkrechten Teile des oberen Korpus ein Temperatur und Luftfeuchtigkeitssensor eingebaut werden, welcher Daten des Systems auf eine Handy App übertragen soll. Die Gesamtleistung der Beleuchtung kann durch den Einbau der gewählten Leds im Vergleich zum vorherigen Konzept auf knapp 31W herabgesenkt werden, somit sollte das gesamte System mit einer Anpflanzleinheit eine Leistung von rund 61W verbrauchen. Die Kabelführung für die diversen elektronischen Bauteile des Systems soll sich in einer der Profilnuten der Robot Units Profile befinden und die Spannungsversorgung der Bauteile mit der Platine in der unteren Korpus Einheit sicherstellen. Durch die eigens konstruierte Beleuchtung ist es bei diesem Konzept auch möglich mittels Handy App die Beleuchtungsdauer bzw. deren Intensität zu steuern.

2.2.3.4 Verbesserung des Anpflanzmechanismus

Die überarbeitete Version des Drehmechanismus besitzt größere Radien bei der Linienführung des Kreisbogens, da diese bei der Vorgänger Version zu klein angenommen wurden wodurch das ganze System nicht zu gebrauchen war. Dadurch, dass die Radien eineinhalb Mal so groß sind wie bei der ersten Version erreichen die Samenbehälter bei der Drehbewegung des Rotationsdeckels eine höhere Geschwindigkeit, dadurch muss vor allem bei sehr Bewegungsempfindlichen Pflanzen darauf geachtet werden, dass diese nicht zu „hart“ im äußeren Anpflanzkreis angelangen. Der darunter liegende Sterndeckel kann bei dieser Umsetzungsart jedoch gleich gelassen werden, da dieser die Funktion des Drehmechanismus nicht maßgeblich beeinflusst. Weiters wurde überlegt ob der Radius des Systems erweitert werden sollte, jedoch wurde diese Idee wieder verworfen, da maximal ein bis zwei Zentimeter zwischen den Pflanzen hinzugekommen wären, und da diese Umsetzung mit erneuten Kosten verbunden gewesen wäre.

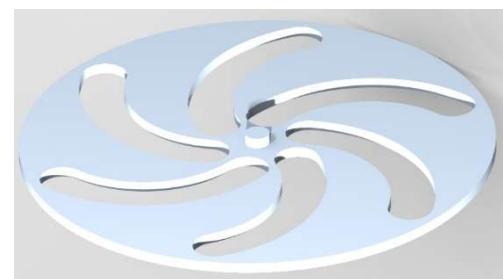


Abbildung 64: Verbesserung des Anpflanzmechanismus

2.2.3.5 Verbesserung des Samenbehälters

Die zweite Version des Samenbehälters hat ebenfalls eine Höhe von 70mm und wurde genau wie sein Vorgänger mittels 3D-Druckers gefertigt. Hierbei wurden jedoch bei dessen Konstruktion einige Veränderungen vorgenommen. Zum einen wurde die Wandstärke von 3,5mm auf 2,5mm herabgesetzt bzw. der Flankenwinkel verkleinert, jedoch musste im Umkehrschluss der Winkel in den unteren 10mm verkleinert werden, damit das Nährbodenmaterial durch den Wasserfluss nicht verloren geht. Eine weitere Änderung zu Gunsten des Anpflanzvolumens besteht darin, dass die Auflagefläche aufgrund des geringen Gewichts welche diese zu tragen hat erheblich verkleinert wurde. Für eine effizientere Nutzung des Wassers wurden die Zulauflöcher von 12 auf 8 Stück herabgesetzt bzw. deren Durchmesser und Höhe verkleinert. Für eine kommerzielle Nutzung müsste das Konzept des Samenbehälters mittels Spritzgusses umgesetzt werden, da für einen 3D Druck zu viel Zeit (3h/Behälter) beansprucht werden würde.



Abbildung 65:
Samenbehälter V2

2.2.3.5.1 Säuberungsbürste

Durch das Wachstum der Pflanzen im Nährboden kann nicht verhindert werden, dass sich einzelne Pflanzenwurzeln mit dem Material des Samenbehälters verbinden. Um ihn anschließend von diesen Verunreinigungen zu befreien wird eine konisch geformte Bürste mit Kunststoffborsten verwendet. Diese wird mit Zugabe von ein wenig Wasser in den Behälter eingeführt und anschließend durch eine Drehbewegung die Verunreinigungen entfernt.



Abbildung 66: Säuberungsbürste

2.2.4 Korpus

2.2.4.1 Korpus Anpflanzeinheit

Der Aufbau des Korpus der Anpflanzeinheit erfolgt mit Robot Units Profilen, welche mit Verbindungselementen zusammengeschraubt werden. Drei der vier Seitenwände wurden mit Plexiglasplatten versehen welche durch spezielle Halterungsschienen und 90° Eckverbindungselementen in Position gehalten werden, diese müssen mittels Bandsäge auf Länge gebracht werden. Auf der vierten Seitenwand wird eine Tür eingebaut, welche ebenfalls mit einer Plexiglasplatte versehen wird, eingebaut wird diese mit zwei Rasterscharnieren, einer Türraste und einem Haltegriff für ein leichteres Öffnen und Schließen der Einheit. Als Abschluss für die Oberseite, muss eine Platte mit 500x500mm zugeschnitten werden, welche mit M10 Gewindeschrauben befestigt wird. Zwischen die Platte und den Profilelementen müssen auf allen 4 Eckpunkten noch Abstandshalter befestigt werden, um die Luftzufuhr in das System zu gewährleisten.

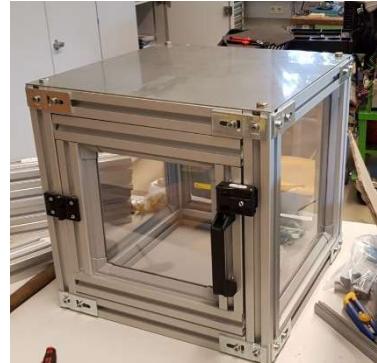


Abbildung 67: Anpflanzeinheit

2.2.4.1.1 Abstandshalter

Der Abstandshalter ist ein Quadratisches 3D -Druck Teil mit den Außenmaßen 50x50mm, welches die Aufgabe hat die obere Abschlussplatte auf Distanz zu halten, damit die Luft in das System mittels Lüfter gelangen kann und um die Beleuchtungsplatte der LEDS zu kühlen. Er hat vier quadratische Zäpfchen, die mittels Presspassung in die Stirnkante des Robot Units Profil gesteckt werden. Pro Anpflanzeinheit werden vier Stück dieses Abstandshalters benötigt damit die Abschlussplatte bzw. die nächste Einheit waagerecht angebracht werden kann.

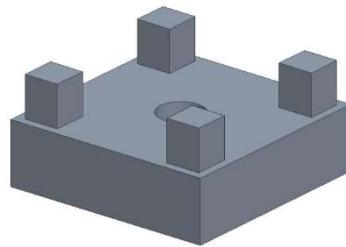


Abbildung 68: Abstandshalter

2.2.4.2 Korpus untere Gehäuseeinheit

Der Aufbau des Korpus der unteren Gehäuseeinheit erfolgt ebenfalls mittels Robot Units Profilen, welche durch Verbindungsbleche zusammengehalten wird. In ihr finden die Platine samt Gehäuse Platz bzw. der Wasserbehälter mit Pumpe und Füllstandsensor. Für einen leichteren Zugang zu diesen Elementen wird eine Tür eingebaut, auch hier kommen wieder die eine Türraste und ein Rasterscharnier zum Einsatz. Verkleidet wird die Einheit mit undurchsichtigen Polycarbonatplatten. Auf der Unterseite sind Gummistandfüße bzw. optional Rollen angebracht, um den Transport zu erleichtern.



Abbildung 69: Untere Gehäuseeinheit

2.2.4.3 Korpus Zusammenbau

Der Zusammenbau der beiden Korpus Einheiten erfolgt ebenfalls mittels Verbindungsblechen, hierbei muss jedoch auf das Richtige Anzugsdrehmoment von 50Nm geachtet werden, da ein verschieben der beiden Teile gegeneinander vermieden werden sollt. Zwischen die Korpus Teile wird eine Platte befestigt, um der Anpflanzwanne einen stabilen Untergrund zu bieten. Diese dient in weiterer Folge des Zusammenbaus auch als Zentrierbefestigung für den Anschlussstutzen, welcher das Wasser über einen Schlauch in die Anpflanzwanne leitet. Aufgrund von möglicher auftretender Undichtheit zwischen Anpflanzwanne und Zwischenplatte, muss das Gewinde der Schraube des Anschlussstutzens vor dem Einbau noch mit Silikon abgedichtet werden.



Abbildung 70: Fertiger Korpus

2.2.5 Beleuchtungsplatte

2.2.5.1 Mechanische Bearbeitung Beleuchtungsplatte

Der Zuschnitt der Aluminiumplatte welche als Kühlung für die Power LEDs dient erfolgt mittels Motortafelschere. Die Platte wird im Nachhinein angerissen und mit den nötigen Kabeldurchführungen für die LEDs versehen. Für die Frischluftzufuhr in das System wird die Platte zusätzlich noch mit Luflöchern versehen und ein Platz für einen Lüfter in deren Mitte gelassen, wenn die Luft im System zu warm werden sollte. Aus optischen Gründen wurde die Platte zuletzt noch einer Oberflächenbehandlung unterzogen. Dadurch Aluminium ein weiches Material ist konnte der X-Schliff der Oberflächenbehandlung mit einer Handelsüblichen Stahlwolle erfolgen. Um eine schöne Kante schleifen zu können muss zunächst das Muster mit einem Handelsüblichen Klebeband abgeklebt werden, anschließend kann mit den Schleifarbeiten begonnen werden. Dafür darf immer nur vom Rand in Richtung Mitte geschliffen werden, da ansonsten ein ungleichmäßiges Muster entsteht.

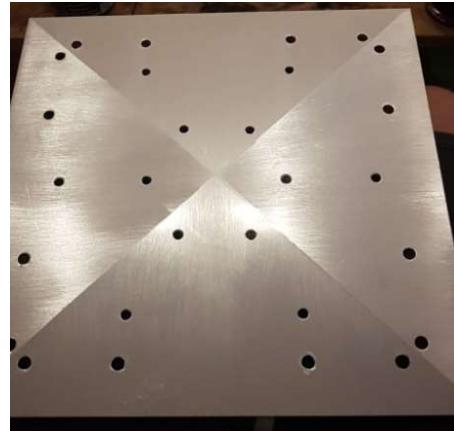


Abbildung 71: Beleuchtungsplatte

2.2.5.2 Verkabelung Beleuchtungsplatte

Die Beleuchtung wurde mittels High-Power-Star LEDs realisiert, welche mit einem wärmeleitenden doppelseitigen Klebeband auf die Beleuchtungsplatte aufgetragen wurden. Anschließend wurden jeweils sechs LEDs in Serie miteinander verbunden, um einen äußeren und inneren Beleuchtungskreis zu erhalten. Die benötigte Stromzufuhr erfolgt mittels einer Konstantstromquelle pro Kreislauf welche jeweils den Strom auf 700mA begrenzt.

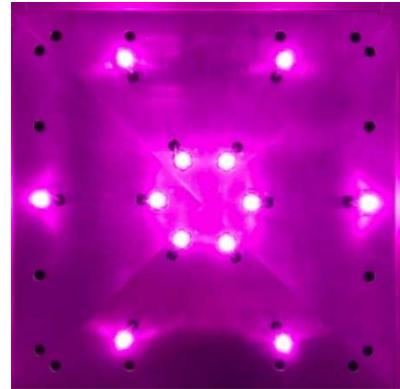


Abbildung 72: Beleuchtungsplatte mit LEDs

2.2.6 Anpflanzwanne

2.2.6.1 Biegen der Anpflanzwanne

Die Anpflanzwanne besteht aus Niro 1.4301 mit einer Blechstärke von 2mm. Gewählt wurde dieses Material wegen seiner Korrosionsbeständigkeit und guten Schweißbarkeit bzw. seiner guten Hygienebeständigkeit. Nach dem Zuschnitt der Platte mittels Motortafelschere für die Anpflanzwanne auf 560x560mm, erfolgt das Anreißen der Biegelinien bzw. der Eckpunkte, welche vor der Biegung ausgeschnitten werden müssen. Da das Biegen des Niro Blechs mit der herkömmlichen Biegemaschine nicht funktioniert wird die Form der Wanne entlang der Biegelinien mit einer Hydraulischen Biegemaschine mit einem Druck von ca. 140t gebogen. In die Maschine müssen vor der Biegung die Parameter des Materials eingestellt werden bzw. der Anschlag sowie Ober- und Unterwange befestigt werden.

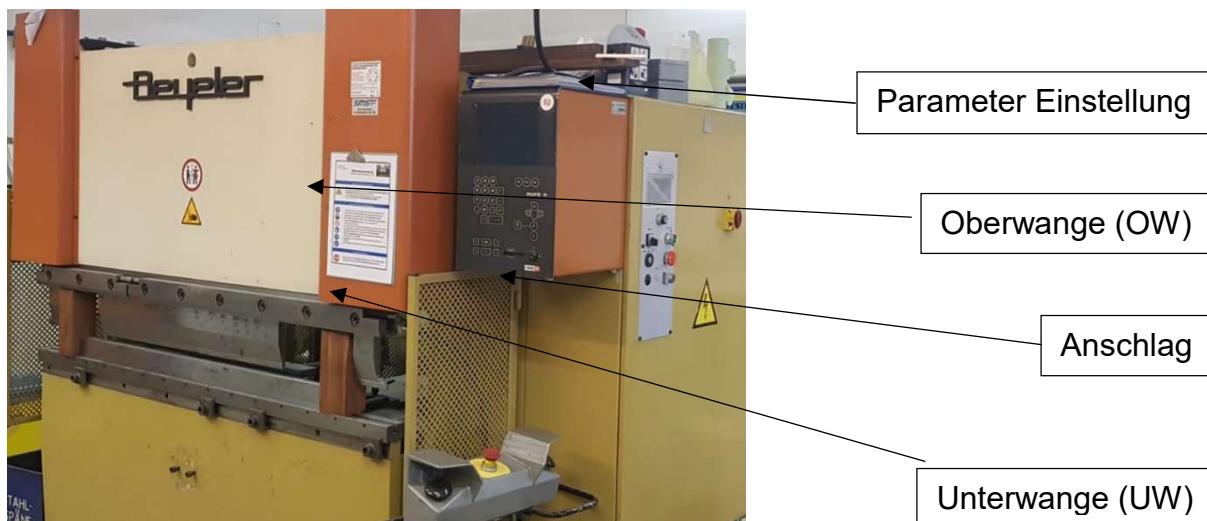


Abbildung 73: Hydraulische Biegemaschine

2.2.6.2 Schweißen und Dichtheitsprüfung der Anpflanzwanne

Die zuvor gebogene Anpflanzwanne wurde mittels eines WIG Schweißgerätes verschweißt und anschließend alle Schweißstellen mit einem Winkelschleifer bearbeitet, um eine homogene Oberfläche zu erhalten. Da die Wanne im späteren Gebrauch ca. bis zur Hälfte mit Wasser gefüllt sein wird, wurde diese auf Dichtheit geprüft, dies



Abbildung 74: Verschweißte Wanne

geschah mittels einer Füllung der Wanne bis zum obersten Rand und anschließender Trockenprüfung aller verschweißten Elemente.

2.2.6.3 Nachbehandlung

Nach Beendigung der Schweiß- und Biegearbeiten, konnte mit der Nachbehandlung der Anpflanzwanne sowie der Abschlussplatte begonnen werden. Diese wurde mit einer Exzентerschleifmaschine umgesetzt, zunächst mit einer groben Körnung des Schleifblattes um die tiefen Kratzer oder Beschädigungen aus der Oberfläche zu schleifen und anschließend mit einer etwas feineren Körnung, um eine feine matte Oberfläche zu erhalten.



Vorher ↔ Nachher



Abbildung 76: Vor Nachbehandlung

Abbildung 75: Nach Nachbehandlung

2.2.7 Drehmechanismus

2.2.7.1 Konstruktion Drehmechanismus

2.2.7.1.1 Sterndeckel

Der Sterndeckel besteht aus einer 8mm dicken Acryl Platte und wird mittels CNC Fräsen gefertigt. Er besitzt 6 Linienförmige ausschnitte, welche vom Zentrum des Bauteils ausgehen. Die Ausschnitte besitzen eine Länge von 147mm und eine Breite von 17mm, wodurch die Richtung der Samenbehälter vorgegeben wird. In der Mitte des Deckels befindet sich eine Bohrung mit 20mm Durchmesser, welche für die Zentrierung des Rotationsdeckels verantwortlich ist, weiters befindet sich auf der Oberseite eine kreisrunde Führung, welche dafür sorgt, dass sich der Rotationsdeckel perfekt in den Sterndeckel einfügen bzw. auf ihm verdrehen lässt. Zuletzt befindet sich auf der Unterseite noch eine rechteckige Führungskante, welche für einen sicheren Halt des Sterndeckels in der darunter liegenden Anpflanzwanne sorgt.

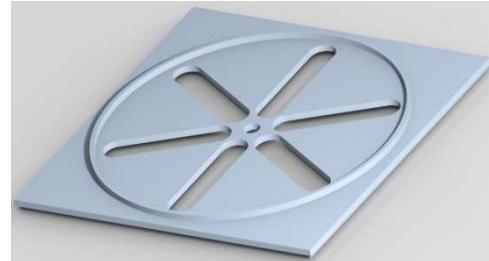


Abbildung 77: Sterndeckel

2.2.7.1.2 Rotationsdeckel

Wie auch der Sterndeckel wird der Rotationsdeckel aus einer 8mm starken Acrylplatte mittel CNC-Fräsen gefertigt. Er hat einen Außendurchmesser von 360mm und besitzt 6-Kreisbogenförmige Aussparungen welche in einem 60° Versatz von der Mitte des Deckels ausgehen. Diese Aussparungen dienen als Führung der Samenbehälter in den äußeren Anpflanzkreis und besitzen eine Breite von 17mm. Sowohl der Außen- als auch der Innenradius der Ausschnitte ist wichtig da bei einem zu geringen Radius die Tangential bzw. Reibkräfte zu groß werden, aus diesem Grund wird ein Außenradius von 114mm und ein Innenradius von 80mm gewählt. In der Mitte des Deckels befindet sich ein Zapfen einem Durchmesser von 19mm welcher wie bereits erwähnt zur Zentrierung des Rotationsdeckels dient.

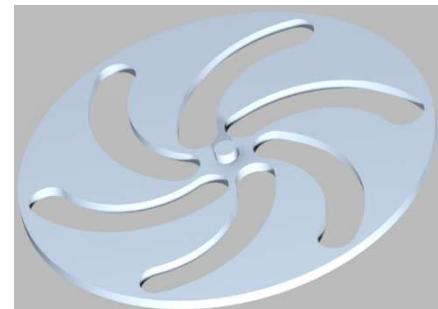


Abbildung 78: Rotationsdeckel

2.2.7.1.3 Zusammenbau Drehmechanismus

Der Zusammenbau des Drehmechanismus erfolgt anschließend wie schematisch dargestellt, es muss zunächst der Sterndeckel mittels Presspassung mit dem Anpflanzbecken verbunden werden. Der nächste Schritt besteht darin, den Rotationsdeckel mittels zentrierzapfen auf den Sterndeckel aufzusetzen, anschließend können schon die Samenbehälter des inneren und äußeren Anpflanzkreises in dem Zusammenbau eingefügt werden. Sobald der Drehmechanismus fertig zusammengebaut ist wird er samt Anpflanzbecken in die obere Korpus Einheit gestellt und das Anpflanzen kann beginnen.

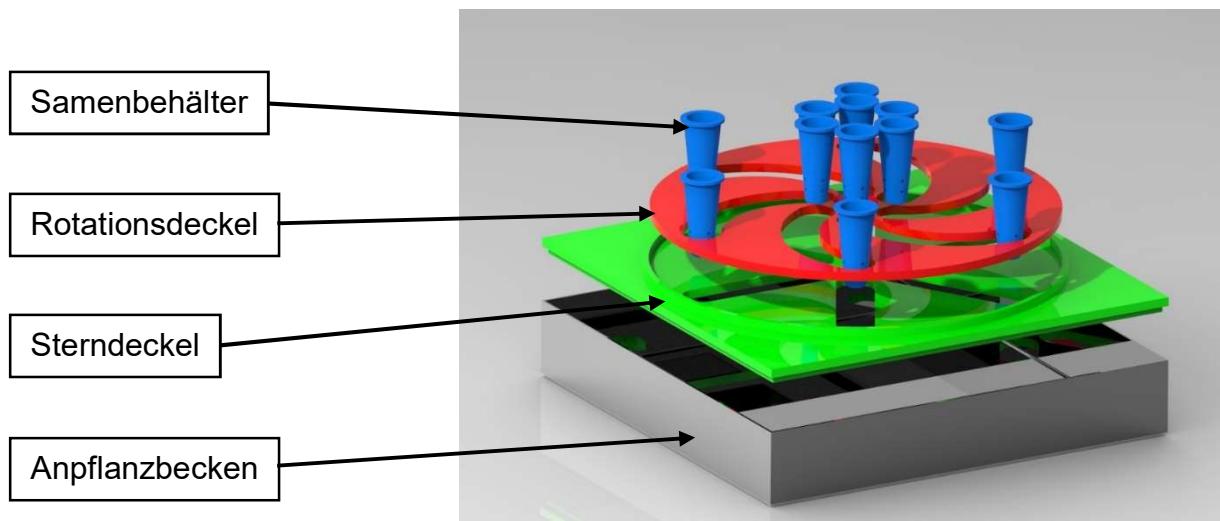


Abbildung 79: Schematischer Zusammenbau Drehmechanismus

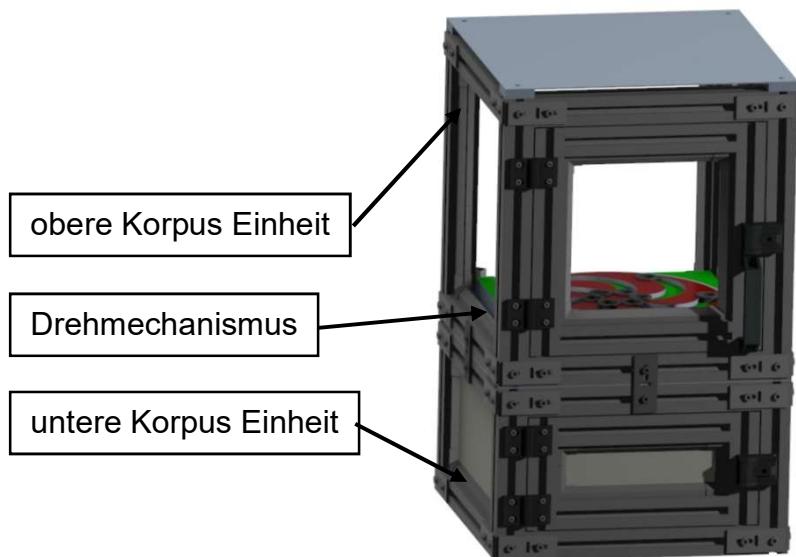


Abbildung 80: Korpus fertig

2.2.7.2 Funktion & Anpflanzen im Mechanismus

2.2.7.2.1 Funktion

Der Drehmechanismus ist ein sogenannter 60° Mechanismus, dass heißt, dass beim verdrehen des Rotationsdeckel um 60° gegen den Uhrzeigersinn ein halber Anpflanzzyklus vorrangeschritten ist. Diese 60° Bahnen sind durch den Rotationsdeckel bzw. den darunterliegenden Sterndeckel vorgegeben, wodurch die Samenbehälter einer angenäherten Kreisbogen Bahn folgen müssen.

2.2.7.2.2 Anpflanzen

Das Anpflanzen funktioniert in dem System wie folgt:

(bei den erklärten Schritten wird von einem komplett leeren System ausgegangen)

ERSTER SCHRITT:

Zunächst werden 6 Samenbehälter inklusive Samen und Nährboden in den mittleren Anpflanzkreis der Rotationsplatte gesteckt, anschließend benötigen diese 6 Pflanzen je nach Pflanzenart eine gewisse Wachstumsdauer bis die Blütephase vollendet ist. In diesem Blütezyklus erfährt die Pflanze eine intensive

Beleuchtung da die Wachstums LEDS in einem gebündelten Lichtkreis über jedem Samenbehälter angebracht sind, zuzüglich wird der Nährboden noch von der Unterseite bewässert wodurch das Wasser besser zum Keimling (unausgereifter Samen) gelangen kann.

ZWEITER SCHRITT:

Hierbei ist zu sehen wie die Pflanze (in diesem Fall Salta) nach der Vollendung der Blütephase im System aussieht. Da Lebensmittelpflanzen wie Salat eine sehr hohe Feuchtigkeit in ihrem Blattwerk gespeichert haben können die Pflanzen in ihrer jetzigen Position nicht weiterwachsen, da sich die Blätter der verschiedenen Pflanzen berühren wodurch die Gefahr von Schimmelbakterien gegeben ist.



Abbildung 81: Samenbehälter mit Keimling



Abbildung 82: Samenbehälter mit vollendetem Blütephase

DRITTER SCHRITT:

Da wie bereits erwähnt, die Pflanzen im inneren Anpflanzkreis nicht deren volles Potential ausschöpfen können und die Gefahr von Bakterien entsteht muss, sobald die Blütephase vollendet ist die Rotationsplatte um 60° gegen den Uhrzeigersinn verdreht werden. Durch diese Bewegung wandern die Samenbehälter in den äußeren Anpflanzkreis wo die Pflanzen wieder genügend Platz zum Wachsen haben und sich deren Blätter nicht mehr berühren. In dem äußeren Anpflanzkreis ist das Licht auch nicht mehr so konzentriert wodurch „Verbrennungen“ der Pflanze vermieden werden.



Abbildung 83: Samenbehälter im äußeren Anpflanzkreis

VIERTER SCHRITT:

Zuletzt kann, durch das Verschieben der hervorgehenden Generation in den äußeren Anpflanzkreis eine neuen Generation Pflanzen angepflanzt werden. Das System kann wieder gedreht werden sobald die hervorgehende Generation Pflanzen ihre Wachstumsphase vollendet haben und bereit zur Ernte sind. Ist dies der Fall so können die Samenbehälter mit den ausgewachsenen Pflanzen aus dem System entfernt werden und der Rotationsdeckel kann wieder um 60° verdreht werden. Dies Funktioniert dadurch, dass sobald die hervorgehende Generation ihr Wachstum abgeschlossen hat, die nachfolgende Generation im gleichen Zuge ihre Blütephase vollendet hat.



Abbildung 84: Erste und zweite Generation im Wachstum

2.2.8 Wasserbehälter

2.2.8.1 Wasserreservoir

Das Wasserreservoir besteht im Grundlegenden aus einer Aufbewahrungsbox aus Polypropylen mit dazugehörigem Deckel. In eine der schmalen Seitenwände der Aufbewahrungsbox muss im unteren Drittel, mittels eines Holzfräzers eine Bohrung mit Durchmesser ...mm für den Füllstandsensor gebohrt werden und anschließend dieser samt Flachdichtung in die Bohrung geführt und schließlich mit einer 3/8 Zoll Mutter festgezogen werden. Die Bohrung muss zusätzlich noch mit Silikon abgedichtet werden, da ansonsten Wasser austreten kann was bei einer etwaigen Dichtheitsprüfung ersichtlich wird. Für einen besseren halt der Dichtmasse muss die Oberfläche des Wasserreservoirs aufgeraut werden, da sie ansonsten nicht fähig wäre den Kleber zu fixieren.



Abbildung 85: Füllstandsensor eingebaut

2.2.8.2 Deckel Wasserreservoir

Der Deckel besteht ebenfalls aus Polypropylen und wird auf das Wasserreservoir aufgesetzt. In die hintere rechte Ecke muss ein Loch geschnitten werden da hier die Halterung für die Pumpe Platz finden soll. Beim Schneiden muss eine langsame Schnittgeschwindigkeit gewählt werden, da ansonsten das Polypropylen des Deckels hinter der Schnittkante erneut verschmilzt. Sie wird mittels Polokalrohr verwirklicht welches durch einen Komponentenkleber mit der zuvor aufgerauten Oberfläche des Deckels verklebt wird. Das Polokalrohr hat in den unteren zwei Zentimetern



Abbildung 86: Deckel mit Polokalrohr

Halbkreisrunde Einschnitte damit das Wasser zur Pumpe gelangen kann, welche sich im aufrechten zustand in diesem befindet.

2.2.9 Platinen Gehäuse

2.2.9.1 Platinen Gehäuse erste Version

Das Platinen Gehäuse dient zum Schutz und zur Halterung der Platine im unteren Korpus des Projekts und es wird mittels 3D Druck gefertigt. Basierend auf den äußereren Abmaßen bzw. den Platzierungen der Bohrungen der Platine wurde in Creo ein Gehäusemodell mit einer Wandstärke von 3mm erstellt. Im Inneren des Gehäuses befinden sich in jeder Ecke jeweils ein 10mm hoher Befestigungssockel mit einer M3 Gewindebohrung, welche als Verschraubpunkte für die Platine dienen. Auf der oberen Gehäuseseite befinden sich auf allen Eckpunkten ebenfalls M3 Gewindebohrungen, diese dienen allerdings nicht zum Halt der Platine selbst, sondern für den Deckel der Einheit, welche die Elektronik vor etwaigen Wasserspritzern bzw. Teilen von Oben schützen soll. Aufgrund des Feinheitsgrades des 3D- Druckers können jedoch keine M3 Bohrungen mit Gewinde gedruckt werden da dieser eine Abweichung bei Kreisdrucken von ca. 0,5mm hat, deshalb muss man ein mittels eines Profils in Creo runde Vertiefung mit Durchmesser 3mm erzeugen.

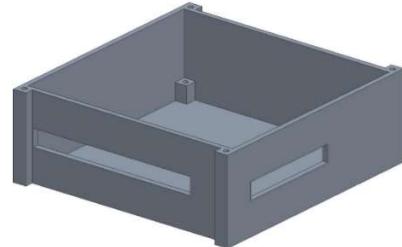


Abbildung 87: Platinen Gehäuse V1

Da sich die Platine im Anschluss in einer geschlossenen Einheit befindet, ermöglichen zwei rechteckige Durchgangslöcher den Zugang für die Kabel in das Innere des Gehäuses. Hier werden diese mit den Anschlussklemmen der Platine verbunden, welche wiederum über die Programmierung andere elektronische Bauteile steuern.

2.2.9.2 Platinen Gehäuse zweite Version

Die zweite Version des Platinen Gehäuses besitzt dieselbe Aufgabe wie sein Vorgänger und wird ebenso mittels 3D- Drucker gefertigt. Für eine optimale Passform und Nutzung des Gehäuses müssen jedoch zwei kleine Veränderungen vorgenommen werden. Zunächst wird die Höhe des Gehäuses von 50mm auf 70mm angehoben da für das Bluetooth Modul nicht genug Platz vorhanden war und dieses über den Deckel der Einheit herausragt ist. Die zweite Veränderung besteht darin den Abstand der Bohrungen auf der schmäleren Seite von 117mm auf 116mm zu reduzieren. Jedoch entschied sich das Projektteam letztendlich, doch die erste Version des Gehäuses zu verwenden, da die zweite Version durch die angehobene Bauhöhe den Zugang zur Platine und somit zum Herzstück des Projekts erschwert. Dies liegt daran, dass die Höhe der unteren Korpus Einheit nur 200mm beträgt und hier noch mit Werkzeug zu hantieren ist nahezu unmöglich. Das Problem mit der Bauhöhe des Bluetooth Moduls wird durch eine Änderung des Gehäuse- Deckels gelöst, bzw. wird hier ebenfalls der Abstand der Bohrung der schmäleren Seite auf 116mm reduziert.

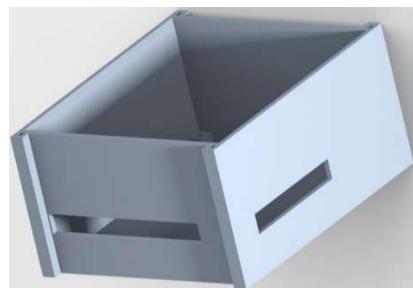


Abbildung 89: Platinen Gehäuse V2

2.2.9.3 Gehäusedeckel

2.2.9.3.1 Gehäusedeckel erste Version

Der Gehäusedeckel wird aus Plexiglas mit einer Glasstärke von 8mm gefertigt. Dafür wird zunächst mit dem Programm Corel Draw dessen Außen form konstruiert und anschließend die gezeichneten Linien rot eingefärbt. In der Mitte des Deckels soll aus optischen Gründen noch der Projektschriftzug



Abbildung 90: Layout
Programm VSLER

integriert werden. Wenn die Konstruktion abgeschlossen ist, wird



Abbildung 92: Platinen
Gehäuse mit Deckel

der gezeichnete Deckel in das Programm VSLER übertragen. Dort muss vor der Fertigung noch das Material Acryl zugewiesen werden, die dicke auf 8mm gestellt werden und die Leistung des Lasers auf 80% gedrosselt werden. Zuletzt müssen alle Bohrungen mit einem Senkbohrer nachbearbeitet werden, da, um den Deckel zu montieren Senkschlitzschrauben zur Verwendung kommen.

2.2.9.3.2 Gehäusedeckel zweite Version

Die überarbeitete Version des Gehäusedeckels ähnelt seiner Vorgängerversion, jedoch wird hierbei Platz für die Bauhöhe des Bluetooth Moduls gelassen, da dieses über die Höhe des Gehäuses hinausragt. Dies geschieht mittels eines Ausschnittes im Deckel, welcher es ermöglicht, dass das Modul ebenfalls mit dem Gehäuse verbaut werden kann. Da es eine eigene Isolierung gegen Spritzwasser besitzt ist es nicht von Nöten, dass es komplett unter dem Deckel liegt, lediglich dessen Kontakte sind durch den Deckel vor Wassereintritt geschützt.



2.3 Elektronik

2.3.1 Bauteile auf der Platine

2.3.1.1 Microcontroller

In das System ist ein PIC16F1827 integriert. Dieser verfügt über 15 I/O Pins, welche sowohl als Eingang, als auch als Ausgang genutzt werden können. Er ist über den Schaltregler mit 5V versorgt und mit einem 220nF Kondensator gegen Spannungseinbrüche entkoppelt. Der gewählte Microcontroller hat noch 4 weitere Anschlüsse, welche im Falle einer Änderung der Schaltung verwendet werden können. Zur Programmierung des PICs gibt es ein PICKit, welches den Microcontroller über USB mit dem Computer verbinden kann. Auf der Platine muss deshalb ein Anschluss für das PICKit eingeplant sein.



Abbildung 94:
Microcontroller

Beschaltung:

Der gewählte PIC16F1827 ist zentral in die Schaltung integriert. Wie in Abbildung 95 zu sehen, sind 14 von 18 Ausgänge des Microcontrollers belegt. Die nicht Verwendeten sind für eventuelle Erweiterungen vorenthalten und können jederzeit genutzt werden. Bei der aktuellen Schaltung wird ein Augenmerk auf die PWM-fähigen gelegt, da man nicht weiß, ob beispielsweise die Status LED gedimmt werden soll.

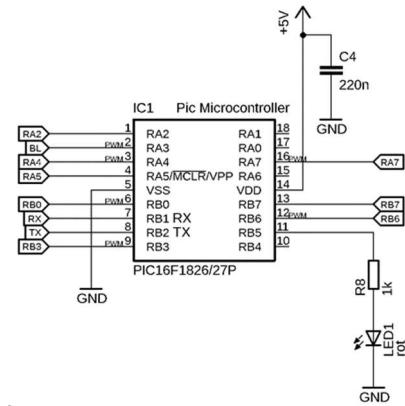


Abbildung 95: Beschaltung Microcontroller

2.3.1.2 Dioden Dimensionierung

Außerdem müssen die passenden Dioden ausgesucht werden, denn am Schaltplan steht alles fest, außer den Dioden. Um eine Geringere Verlustspannung zu erhalten und die ausreichende Schnelligkeit bei Freilaufdioden zu erzielen, werden Schottky Dioden ausgewählt. Diese haben im Normalfall einen Spannungsabfall von unter einem Volt, sind jedoch teurer. Gewählt sind die Dioden:

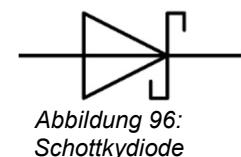


Abbildung 96:
Schottkydiode

PRLL5819, Freilaufdioden für Lüfter: max. 40V, 1A

SSB44, Freilaufdiode für Pumpe: max. 40V, 4A

2.3.1.3 Schaltregler

Da man die Komponenten eines Schaltreglers meist nicht als Einzelstück beziehen kann, wird nach anderen Lösungen gesucht. Nach mehreren Recherchen findet man eigene Bauteile, welche die zusätzliche Beschaltung schon integriert haben zu kaufen. Dabei handelt es sich um kleine Platinen, welche mit der Platine mittels Stiftleisten verbunden werden können.



Abbildung 97:
Schaltregler

Beschaltung:

Die Schalregler sind unmittelbar nach den Eingangsklemmen, welche direkt vom Netzteil versorgt werden, platziert. Das hat den Vorteil, dass die Leiterbahnen nicht zu lang sind und die Breite geringer sein kann als bei langen Leitungen mit gleichem Spannungsabfall. Weiters ist der Footprint der 2 Step-down Module selbst konstruiert, da die Module zugekauft sind und es für diese keine Eagle Datei gibt.

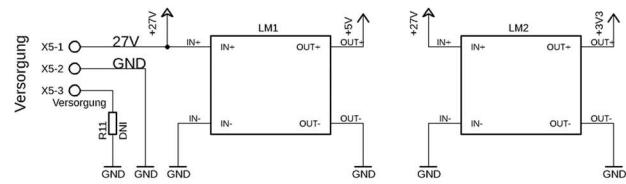


Abbildung 98: Beschaltung Schaltregler

2.3.1.4 Konstantstromquellen

Konstantstromquellen werden benötigt, um die LEDs anzusteuern. Da diese konstant 700mA Strom brauchen, wird dieses Modul gewählt. Es ist vorkonfiguriert auf 700mA, also sind Widerstände perfekt angepasst. Weiters ist es möglich, die LEDs mittels PWM zu dimmen und sogar ganz auszuschalten. Hierbei gilt die invertierte Logik, denn legt man die PWM auf Masse, leuchten die LEDs zu 100%. Es ist möglich, an den Ausgang bis zu 10 LEDs anzuschließen. Da wir nur 6 LEDs pro Stromquelle benötigen, bringen wir diese Platine nicht ans Limit. Auf eine aktive Kühlung kann somit verzichtet werden.



Abbildung 99: Konstantstromquelle

Beschaltung:

Da das System mit bis zu drei Etagen erweiterbar sein soll, werden 6 Stück Konstantstromquellen benötigt. Pro Etage befinden sich 12 LEDs mit jeweils 6 Stück in Serie geschaltet. Da pro LED bis zu 3,6V Spannung abfallen können und das gesamte System nicht über 30V versorgt werden soll, sind nur 6 Dioden in Serie verbunden. Gesteuert werden die Konstantstromquellen mit dem PIC über einen PWM fähigen Pin. Da die Logik der Konstantstromquelle active-low ist, leuchten die LEDs, wenn keine PWM angelegt ist.

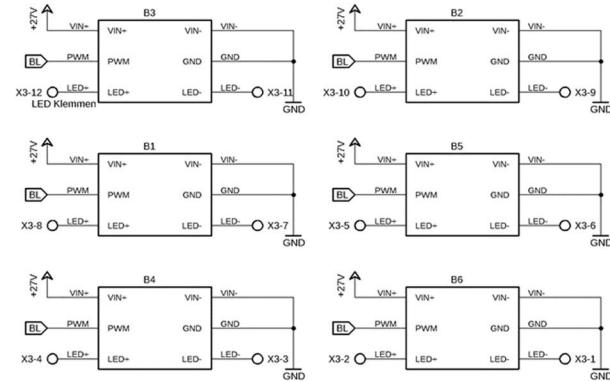


Abbildung 100: Beschaltung Konstantstromquelle

2.3.2 Sensoren und Aktoren

2.3.2.1 Füllstandssensor

Um einen Trockenlauf der Pumpe ausschließen zu können, muss ein geeigneter Sensor gewählt werden. Dieser sollte leicht einzubauen sein und leicht in das Programm eingebunden werden können. Schlussendlich wird für einen einfachen Schwimmerschalter entschieden, welcher bei einem zu kleinen Füllstand das Signal unterbrechen kann.



Abbildung 101:
Füllstandssensor

Beschaltung:

Da der Füllstandssensor ein Schwimmer ist, kann er wie ein normaler Schalter durchschalten oder unterbrechen. Die Versorgungsspannung beträgt 5V. Das Signal des Sensors ist mit dem Pin RA4 verbunden. So kann man erkennen, ob der Füllstand in Ordnung ist oder nicht. Zur Sicherstellung, dass kein undefinierter Zustand am PIC anliegt, ist ein Pull-Down Widerstand in der Schaltung integriert.

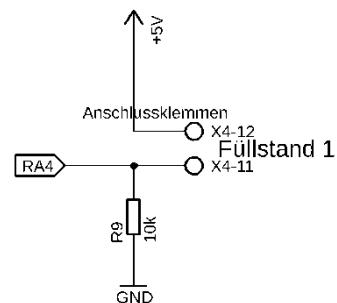


Abbildung 102: Beschaltung
Füllstandssensor

2.3.2.2 Status LED

Die Status LED soll den Füllstand des Gerätes veranschaulichen. Ist dieser in Ordnung, leuchtet die LED grün, ist der Füllstand jedoch unter dem Limit, unterbricht der Sensor und die LED leuchtet rot. Bei der LED handelt es sich um eine Duo LED welche zwei Anoden (rot, grün) und eine Kathode (Masse) besitzt.



Abbildung 103: Status LED

Beschaltung:

Die LEDs sind über zwei verschiedene PWM fähige Pins vom PIC versorgt, da anfangs nicht klar war, ob die LEDs für den Anwendungsbereich zu hell leuchten. Die Kathoden der LEDs werden an den anderen Pins mit Masse verbunden.

Vorwiderstand LED rot:

$$R_V = \frac{V_{CC}}{I} = \frac{5V - 1,8V}{20mA} = 160\Omega$$

Vorwiderstand LED grün:

$$R_V = \frac{V_{CC}}{I} = \frac{5V - 2,2V}{20mA} = 140\Omega$$

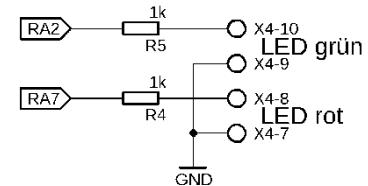


Abbildung 104: Beschaltung Status LED

Da die Status LED nicht zu hell leuchten soll, sondern nur den Füllstand anzeigen soll, ist für $1k\Omega$ entschieden. Damit ist auch sichergestellt, dass die LED nicht zu viel Spannung bekommt und kaputt werden.

2.3.2.3 Feuchtigkeits- und Temperatursensor

In die Vertical Farming Einheit ist ein DHT22 Sensor integriert. Dieser besitzt die One-Wire Technologie und benötigt somit nur 3 Pins, um Daten zu liefern und versorgt zu werden. Der Sensor liefert Temperaturen zwischen -40 und +80°C (Genauigkeit von 0,5°C), sowie 0-100% Luftfeuchtigkeit (Genauigkeit von 2-5%). Dieser Sensor deckt den erforderlichen Bereich für Temperatur und Luftfeuchtigkeit.

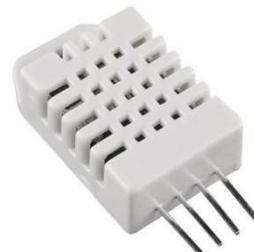


Abbildung 105:
DHT22 Sensor

Beschaltung:

Die Beschaltung ist wie im Datenblatt aufgebaut. Dort ist ein 4,7k Ohm Widerstand empfohlen, der einen unbestimmten Zustand verhindert, wenn zwischen Output sowie Input gewechselt wird. Außerdem hat der DHT22 vier Pins, wovon nur drei in Verwendung sind. Nach einer Recherche, ob der Pin einen Nutzen hat, kam heraus, dass dieser teilweise mit Masse verbunden wird. Das Programm Eagle überprüft ob die Pins verbunden sind, bei keiner Verbindung wird eine Fehlermeldung ausgegeben. Da der Pin „NULL“ in der Abbildung 107 auf manchen Datenblättern mit Masse verbunden ist, wird er vorsichtshalber mit Masse verbunden.

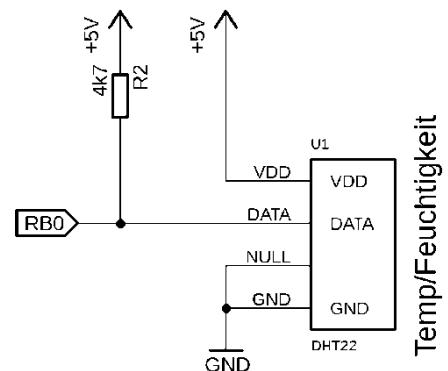


Abbildung 106: Beschaltung DHT22 Sensor

2.3.2.4 Lüfter

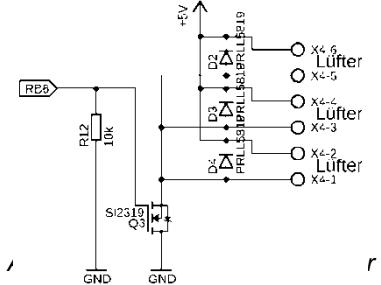
Der gewählte Lüfter bewegt $92,75 \text{ m}^3$ pro Stunde. Dies ist für das Projekt jedenfalls ausreichend, wie die Tests gezeigt haben. Der Lüfter ist mit 5V versorgt und nahezu geräuschlos. Mit einer Leistung von 1,6W ist er auch sehr effizient.



Abbildung 107: Lüfter

Beschaltung:

Die Lüfter werden über einen Mosfet geschalten und mit Freilaufdioden betrieben. Da das System bis zu 3 Lüfter beinhalten kann, muss es dafür auch Anschlüsse, sowie Freilaufdioden geben. Der Pull-Down Widerstand wird benötigt, um einem undefinierten Zustand am PIC aus dem Weg zu gehen.



2.3.2.5 Pumpe

Die Pumpe fördert maximal 10l pro Minute bis zu einer Höhe von 6m. Das reicht ebenfalls völlig aus, da das System mit den drei Anpflanzeinheiten 1,80m hoch ist. Die Fördermenge wird nie völlig verwendet, da der Wasserzulauf sonst einem Springbrunnen gleicht. Die Pumpe ist eine Tauchpumpe, was wiederrum heißt, dass sie im Wasserbehälter sitzt und mittels eines Schlauchs das Wasser abtransportiert.



Abbildung 109: Pumpe

Beschaltung:

Die Pumpe ist mit einer dreipoligen Klemme auf der Platine integriert. Zwischen den zwei benötigten Anschlüssen befindet sich eine Schottky Diode, welche als Freilaufdiode eingebaut ist. Betrieben wird die 12V Pumpe mit einer PWM (max.25%, da Versorgung der Pumpe 24V ist). Um die 24V zu schalten, wird ein Mosfet und ein Pull-down Widerstand benötigt. An dem dritten Pin der Klemme befindet sich ein Widerstand, welcher die Möglichkeit für Erweiterungen offenlässt und nicht implementiert ist.

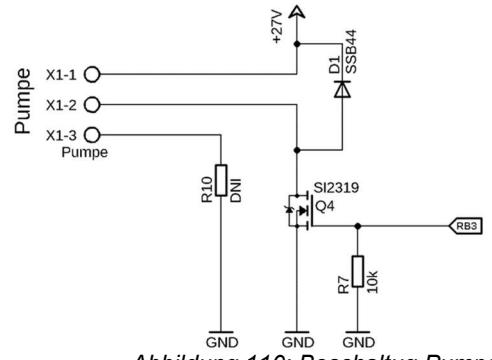


Abbildung 110: Beschaltung Pumpe

2.3.2.6 Power LEDs

Da die gewählten LEDs ziemlich teuer und meist in nur fertigen Systemen angeboten werden, werden diese aus China bezogen. Die LEDs benötigen zwischen 3,2V und 3,6V bei einem Strom von 700mA. Das Farbspektrum ist eigens für solche Zwecke ausgewählt und besteht aus rotem, sowie blauem Licht.



Abbildung 111: Power LED

Farbspektrum:

Um Pflanzen großziehen zu können, benötigt man viel rotes, sowie auch ein wenig blaues Licht. Die Wellenlängen sind bei Rot ca. 660nm und bei blau ca. 450nm. Das blaue Licht ist für das Wachstum in die Länge, sowie in die breite und für die Photosynthese notwendig. Rotes Licht wird verwendet, um die Pflanzen zur Blütenphase zu bringen.

Deshalb werden LEDs mit größtenteils diesen Wellenlängen gewählt.

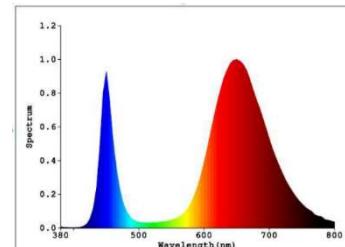


Abbildung 112: Spektrum Power LED

2.3.2.7 Bluetooth-Modul

Es wird das HC-06 Modul gewählt, da die Kommunikation zwischen App und Vertical Farming Einheit drahtlos mittels Bluetooth gegeben sein soll. Um den Levelunterschied zwischen PIC und HC-06 auszugleichen, ist ein Levelshifter notwendig. Die Kommunikation erfolgt mittels RX und TX Pins bei Modul und PIC.

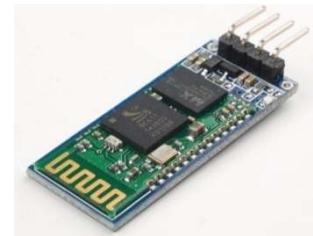


Abbildung 113: HC-06

Levelshifter:

An der TX Seite des Bluetooth Moduls ist ein Mosfet in Gateschaltung. Dieser zieht den Pegel des Signals vom Modul kommend von 3,3V auf 5V, welchen der PIC benötigt. Bei der Schaltung wird Eingangs-, sowie Ausgangsschaltung mittels Pull-up Widerstand auf die jeweilige Referenzspannung gezogen. Das dient dazu, dass bei fehlender Kommunikation, sowie bei einem High Signal der Pegel ausgangsseitig 5V beträgt, eingangsseitig jedoch aufgrund des Mosfets nicht durchgeschalten wird.

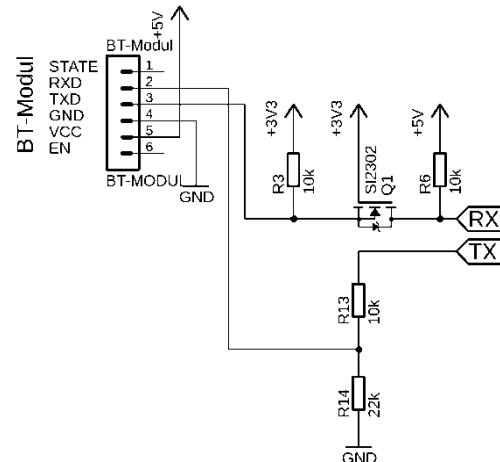


Abbildung 114: Beschaltung Levelshifter

Auf der anderen Seite muss der Pegel von 5V auf 3,3V gebracht werden, dies geschieht mit einem Spannungsteiler aus zwei Widerständen.

Berechnung

Spannungsteiler:

$$U_a = 3,3V \quad \frac{U_a}{U_e} = \frac{R_2}{R_1+R_2}$$

$$U_e = 5V$$

$$R_1 = 10k\Omega \quad \frac{U_a}{U_e} = \frac{22k\Omega}{10k\Omega+22k\Omega} = 0,68$$

$$U_{a_{vorh}} = U_e * \frac{U_a}{U_e} = 3,44V \quad R_2 \text{ gewählt: } 22k\Omega$$

2.3.2.8 Netzteil

Das Netzteil muss die Spannung liefern, die die Konstantstromquellen benötigen, um die gewünschten 700mA der LEDs zu bekommen. Wäre das nicht der Fall, würde man Langzeittests machen müssen, da sich das Spektrum leicht verändern könnte. So benötigt man 27V. Das gewählte Netzteil liefert bis zu 240W. Die LEDs bei drei Etagen benötigen bei 100% Leuchtkraft ca. 57W.



Abbildung 115: Netzteil

2.3.3 Konzepte

2.3.3.1 Platine für High Power LEDs

Schaltplan:

Der Schaltplan besteht aus 6 LEDs in Serie geschaltet, welche mit zwei Pads testweise an die Konstantstromquelle angelötet werden.

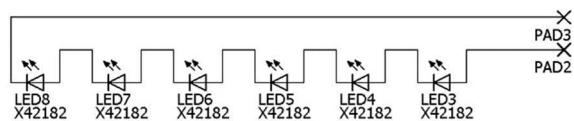


Abbildung 116: Schaltplan Konzept 1

Verwendungszweck:

Bevor die LEDs verbaut werden, soll die Funktion, sowie der Effekt auf Kleinpflanzen wie Kresse und Salat getestet werden. Um die Funktionalität testen zu können, werden ausschließlich 6 LEDs verwendet. Es sollen genau 6 Stück sein, da bei der Beleuchtungsplatte später auch 2 Schaltkreise mit jeweils 6 LEDs verwendet werden. Durch die Fertigung eines Schaltkreises zur Testung wird Zeit für Tests und für die Nachbestellung von kaputten Teilen gespart.

Um den Effekt bzw. die Auswirkung der LEDs auf Pflanzen wie z.B.: Kresse und evtl. Salat sehen zu können, soll die Platine so klein wie möglich und mit möglichst geringen Kosten bzw. Aufwand gefertigt sein. Es wird zwar Zeit bzw. Geld für diese Testplatine benötigt, dies rentiert sich aber, da man sich dadurch vergewissern kann, dass die Komponenten (LEDs, Konstantstromquellen und Netzteile) einwandfrei funktionieren. Abgesehen davon soll anhand des Projekts Studien bestätigt werden, welche besagen, dass Vertical Farming einen dreimal schnelleren Ernteertrag als herkömmliche Landwirtschaft abwirft.

Platinenlayout:

Die Anschlüsse Pad 2 bzw. Pad 3 sind mit den Konstantstromquellen verbunden, welche sich auf der gefertigten Platine befinden. Diese Konstantstromquellen sind mit dem Netzteil versorgt, welches eine Spannung von 27V erzeugt.

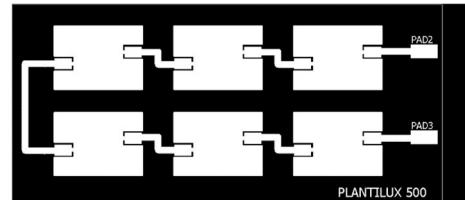


Abbildung 117: Layout Konzept 1

Test:

Um herauszufinden, welches Netzteil benötigt wird, muss die Durchflussspannung der LEDs gemessen und diese mit der Spannung aus dem zugehörigen Datenblatt verglichen werden. Da es bei der Konstantstromquelle zu einem Spannungsabfall von ungefähr 2V kommt, kann man bei einem 27V-Netzteil in der Realität nur 25V verwenden.

Laut Datenblatt benötigt eine LED eine Spannung von 3,2 – 3,6V und einen Strom von 700mA. Wenn man mit der maximalen Spannung von 3,6 V rechnet, (Spannung ist in Abhängigkeit von der Temperatur der LED. D.h.: Desto wärmer die LEDs werden, desto höher steigt die benötigte Spannung an) kommt man bei 6 Stück auf insgesamt 21,6 V. Werden die LEDs mit zu geringer Spannung versorgt, kann es passieren, dass sich die Spektren der jeweiligen Leuchten verschieben. Dies wäre unvorteilhaft, da die LEDs möglicherweise keinen bzw. einen abgeschwächten Einfluss auf das Wachstum der Pflanzen haben. Da die Spannung der LEDs aber von Serie zu Serie variiert und somit von der Produktionsreihe abhängt, wäre es auch möglich, dass diese nur 3,2V hoch ist. Bei 6 LEDs wäre die Gesamtspannung nur mehr 19,6V und ein kostengünstigeres 24V-Netzteil würde ausreichen.

Lösungsansätze:

Um dieses Konzept mit so wenig Zeit- und Geldaufwand wie möglich realisieren, wurden probehalber sechs Stück einzelne LEDs aneinander gelötet und für Tests verwendet. Der Nachteil dieser Lösung ist jedoch die Hitzeabführung, welche praktisch nicht gegeben ist. In diesem Fall löten sich die LEDs von selbst aus und man muss sie wieder verbinden, bevor man weiter testen kann.



Abbildung 118: Lösung 1 Konzept 1

Deshalb wurde mit gekauften, an einem Sockel platzierten LEDs, ein neues Konzept entwickelt. Diesmal bestehend aus 12 LEDs, wobei zwei Mal sechs Stück in Serie geschaltet sind. Durch diese Lösung kann man viel länger testen, da die Sockel mit Aluminiumsubstrat ausgestattet sind und Wärme gut ableiten. Nun besteht das Problem mit den ausgelötzten LEDs nichtmehr und man kann die Platine jederzeit testen, auch ohne die fix verbauten LEDs in der Vertical-Farming Einheit.



Abbildung 119: Lösung 2 Konzept 1

2.3.3.2 Beleuchtungsplatine

Schaltplan:

Der Schaltplan ist so aufgebaut wie unsere Testplatine. Jedoch sind in dem Konzept zwei Mal 6 LEDs für die Beleuchtung einer Etage inkludiert.

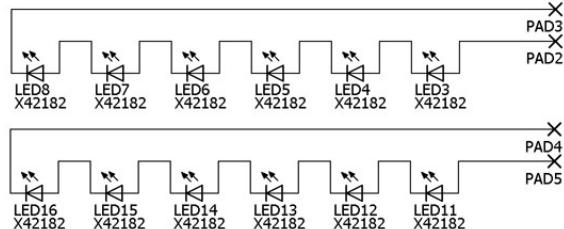


Abbildung 120: Schaltplan Konzept 2

Platinenlayout:

Bei der ersten Version der Beleuchtungsplatine befindet sich jeweils eine LED genau über jeder Pflanze. Ein Vorteil besteht darin, dass jede Pflanze die gleiche Menge an Licht erhält und somit das Wachstum gleichmäßig verteilt stattfindet. Dennoch hat diese Platine ein Abmaß von ca. 400mm x 400m und ist somit nicht überall fertigbar. Weiters ist diese Lösung wenig innovativ und eher kostenintensiv, was dazu führt, dass für unser Endprodukt das folgende Konzept verwendet wird.

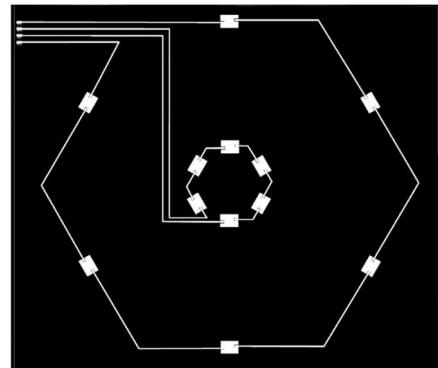


Abbildung 121: Layout Konzept 2

2.3.3.3 Beleuchtungsplatte

Hierfür kann man den obigen Schaltplan, sowie das Layout verwenden. Bei dem Konzept sind ebenfalls die LEDs zentriert oberhalb der Pflanze angebracht, jedoch ist die gesamte Platte nicht aus einer Platte mit Aluminiumsubstrat, sondern besteht lediglich aus einer zugeschnittenen Aluminiumplatte. Auf dieser sind die High Power LEDs mittels doppelseitigen Wärmeleitpads angebracht. Die Pads wurden zugeschnitten geliefert und haben die Eigenschaft, dass sie hohen Temperaturen standhalten und diese auch anhand der Aluminiumplatte wie in Abbildung 122 zu sehen ableiten können.

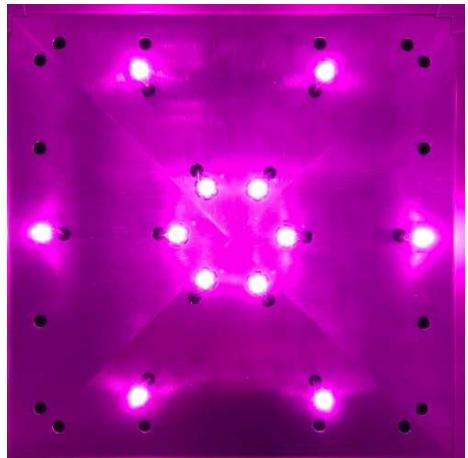


Abbildung 122: Konzept 3

2.3.3.4 LEDs mit Sockel verbinden

Um die LEDs anbringen zu können, müssen diese an die dafür vorhergesehenen Sockel gelötet werden. Normalerweise ist das kein allzu großes Problem, doch die LEDs halten nicht mal der Temperatur bis zum Schmelzpunkt des Lots stand. Wie man in Abbildung 123 erkennen kann, ist der Versuch die Diode im Lötofen zu löten gescheitert. Da die Wärme der Diode abgeleitet werden muss, ist auch das Verbinden mit Wärmeleitpaste nicht vom Vorteil, da die Paste größere Verluste an einer so kleinen Fläche hat. Die obere Schutzschicht löst sich teilweise vom restlichen Kunststoff und legt die Diode ungeschützt frei. Die Funktion ist zwar noch gegeben, da aber ein langlebiges Produkt gebaut werden soll, müssen wir eine andere Lösung finden.



Abbildung 123: LED Lötofen

Lösungsansatz:

Da man diese Dioden in Österreich nur selten zu kaufen bekommt, wird auf fertig gelötete Star High Power LEDs wie in Abbildung 124 zurückgegriffen, welche schlussendlich wieder von China stammen. Diese haben jedoch Pads, womit man die Kabel gleich am Sockel anlöten kann. Außerdem ist die ganze Rückseite aus Aluminium, welches die Wärme perfekt ableiten kann. Die fertig gelöteten LEDs sind anfangs aber nicht bestellt worden, da man zuvor das Konzept 2 umsetzen wollte.



Abbildung 124: Power LED

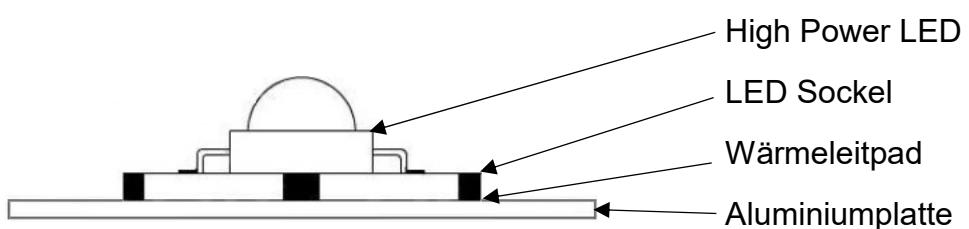


Abbildung 125: Schematik Power LED

2.3.4 Berechnung LED Kühlung

Pro Etage werden 12 LEDs (mit ca. $12 \times 3\text{W} = 36\text{W}$... gerundet ca. 40W) verwendet. Diese sollen mit einer Aluplatte: ($500 \times 500 \times 2\text{mm} \rightarrow 2500\text{cm}^2$) gekühlt sein. Zum Berechnen braucht den Wärmewiderstand von Aluminium, welcher jedoch davor zu messen ist.

$$R_{th,min} = \frac{\Delta T}{P}$$

Der Wärmewiderstand muss weniger als $0,66\text{K}/^\circ\text{C}$ betragen, damit die 20°C Differenz eingehalten werden kann. Da es für eine Aluminiumplatte keinen genauen Wärmewiderstand zur Berechnung gibt, muss dieser ausgemessen werden. Dazu wird im 15 Minuten Abstand die Raumtemperatur notiert und anschließend an eine Funktion angenähert. Zur besseren Veranschaulichung der gemessenen, sowie berechneten Werte dient ein Diagramm.

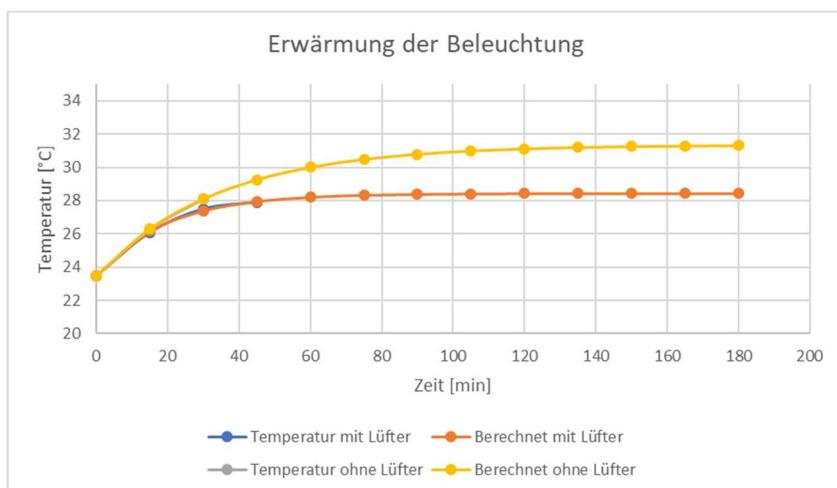


Abbildung 126: Erwärmungsdiagramm

In der Abbildung 126 ist der Temperaturanstieg ohne und mit Lüfter eingezeichnet. Anfangs wurde kein Lüfter im System berücksichtigt. Da die Pflanzen jedoch am besten bei einer Temperatur von ca. 28°C - 30°C wachsen können, muss eine Lösung zur Reduktion der Temperatur auf der Beleuchtungsplatte gefunden werden. Da nicht viel Platz für Kühlkörper besteht, wird ein Lüfter genau mittig oberhalb von der Beleuchtungsplatte platziert. Dieser reduziert die Erwärmung des Systems auf $28,5^\circ\text{C}$ anstatt $31,3^\circ\text{C}$ bei einer Umgebungstemperatur von $23,5^\circ\text{C}$.

2.3.5 Platinenfertigung

Die Platine wird belichtet, geätzt, gewaschen und anschließend gebohrt. Es sind die Leiterbahnen mit einem Messer angepasst, da nicht alles vollständig weggeätzt wird. Nach dem freikratzen wird die Platine mit Lötlack beschichtet, weshalb sie nun gegen Oxidation geschützt ist. Schlussendlich wird sichergestellt, dass die Platine keine Fehler, wie beispielsweise Verbindungen zwischen den Bahnen hat, obwohl keine sein sollen. Nachdem nun immer noch Fehler im System sind, werden diese Verbindungen erneut weggekratzt.

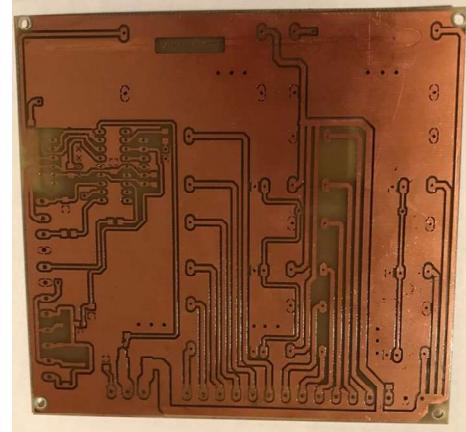


Abbildung 127: geätzte Platine

Nun ist die Platine bereit bestückt zu werden. Dafür werden die oben erwähnten vorgesehenen Komponenten verwendet. Zusätzlich zu diesen wird noch eine Status LED verbaut, welche signalisiert, ob die Platine betrieben wird, oder nicht. Da bei dieser Version der Platine ein Teil des Levelshifters fehlt, muss ein Spannungsteiler eingebaut werden. Dieser wird, wie oben beschrieben, mit zwei Widerständen nachgebessert, um den Pegel zu senken. Die verbesserte Version der Platine ist in der Abbildung 128 zu sehen.

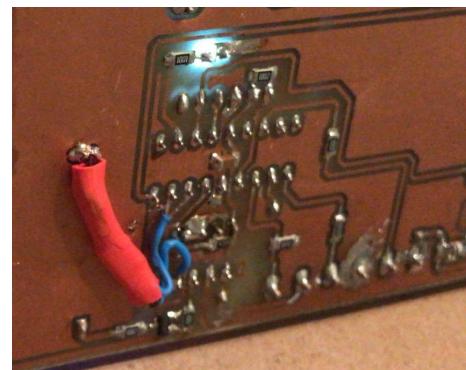


Abbildung 128: Platinenverbesserung

Nachdem die Schaltung getestet wird und die Kommunikation zwischen Bluetooth-Modul und PIC mit einem Oszilloskop gemessen wird, wird der Pegel tatsächlich auf 3,3V gezogen. Das heißt, die Schaltung funktioniert nun. Nach einem Test der Pumpensteuerung, sowie dem Auslesen der Sensorwerte wird festgestellt, dass der PIC mit dem Bluetooth Modul kommunizieren kann. Hierzu wird ein Bitmuster, wie man in Abbildung 129 an den PIC gesendet.



Abbildung 129: Test 1

Nun kann das Platinenlayout auf die neue Version geändert und anschließend bestellt werden. Dabei wird darauf geachtet, dass das Logo auf die Platine gedruckt wird.

Dafür muss das Vertical-Farming Logo als Bitmap Datei abgespeichert und anschließend monochrom gespeichert werden. Nun hat das Logo jedoch nur 2 Blätter, da das eine zu hell ist und somit nicht als schwarz abgespeichert werden kann.

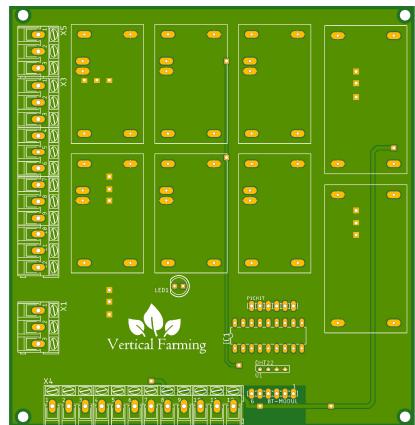


Abbildung 130: Platine Vorderseite

Da das Emblem sowieso auf beiden Seiten gedruckt wird, wird das zweiblättrige gespiegelt, und das gespiegelte Blatt somit übernommen. Auf der anderen Seite folgt dasselbe. Nun kann die Datei in Eagle eingefügt werden und als Beschriftungslayer konvertiert werden. Schon bei der Vorschau der Platinenansichten kann das Logo von beiden Seiten betrachten werden.

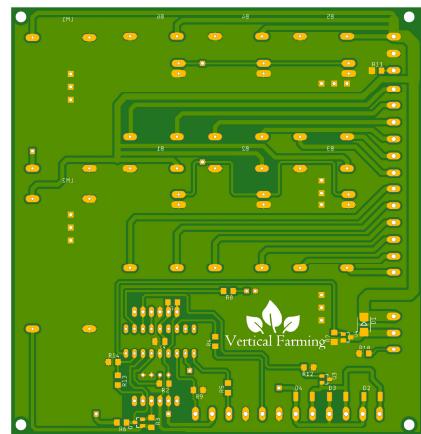


Abbildung 131: Platine Rückseite

Auf der Homepage von Beta-Layout bekommt man eine 3D PDF Datei zugesandt und kann sich die Platine von jeder Seite ansehen, wie in Abbildung 132. Die genormten Bauteile sind gleich integriert. Da die Pinanordnung für die Konstantstromquellen, sowie die Step-Down Regler nirgends so zu finden sind, muss ein neues Package erstellt werden, welches hier nicht in 3D dargestellt werden kann. Außerdem sind die Anschlüsse vom Temperatur- und Feuchtigkeitssensor ebenfalls nicht integriert.

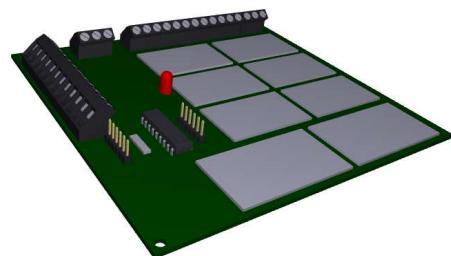


Abbildung 132: Platine 3D

2.3.6 Vorteile Schaltregler im Gegensatz zu Spannungsregler

Anfangs sollte zwar ein linearer Spannungsregler eingesetzt werden, hier ist jedoch ein Beispiel, wieso der Schaltregler besser für unser System ist. Am Eingang liegen 10V, sowie 1A an. Ausgangsseitig befinden sich die geregelten 5V.



Abbildung 133: Reglervergleich

Schaltregler: $\eta=0,95$

$U_e = 10V$

$I_e = 1A$

$U_a = 5V$

$I_a = 2A$

$$P_e = U_e \cdot I_e = 10W$$

$$P_a = U_a \cdot I_a \cdot \eta = 9,5W$$

$$P_{tot} = (P_e - P_a) \cdot (1 - \eta) = 0,5W$$

Die Berechnung wird unter Berücksichtigung des angenommenen Wirkungsgrades η durchgeführt. Mit einer Verlustleistung von 0,5W ist diese Schaltung sehr verlustarm, deshalb hat man sich für den Schaltregler entschieden.

Spannungsregler: (linear)

$U_e = 10V$

$U_a = 5V$

$I_a = 1A$

$$P_{tot} = (U_e - U_a) \cdot I_a = 5W$$

$$P_a = U_a \cdot I_a = 5W$$

Der Wirkungsgrad ist nur bei 50%, somit ist eine Schaltung mit dem verwendeten Beispiel unbrauchbar!

Da so wenige Verluste wie möglich im System sind, entscheidet man sich für die oben genannten Step-Down Regler. Dabei handelt es sich nicht um einen Linearregler, wodurch eine annähernd verlustfreie Regelung der Spannung möglich ist. Außerdem wird dadurch viel Arbeit gespart, da man diese im Internet problemlos bestellen kann.

Würde man das Modul aus einzelnen Bauteilen selbst bauen, muss man mit einem deutlich höheren Preis rechnen. Außerdem ist es einfacher, ein fertiges Modul mittels Pins ins System integrieren, als die einzelnen Bauteile im Schaltplan einzufügen, im Layout zu berücksichtigen und anschließend zu löten.

2.3.7 Berechnung der gesamten Leistung des Gerätes:

$$P = U * I$$

Mit dieser Formel kann man die Leistung ausrechnen.

Grobe Berechnung (nur Verbraucher lt. Blockschaltbild):

Power LEDs+ Konstantstromquellen: 6x 27V, 0,7A

Pumpe: 12V, 1,8A

rote Status LED: 2,5V, 0,02A

grüne Status LED: 2,8V, 0,02A

BT-Modul: 5V, 0,04A

Lüfter: 5V, 0,17A

Leistung LEDs inkl. Konstantstromquellen: 113,4W

Leistung Pumpe: 21,6W

Leistung Lüfter 2,55W

Leistung BT-Modul 0,2W

Leistung Status LEDs: 0,11W

Gesamtleistung (ohne Verluste und nur lt. Blockschaltbild): 137,86W

Mit Hilfe der oben berechneten Leistung kann man den benötigten Strom berechnen.

Berechnung des gesamten Stromes des Gerätes:

$$I = \frac{P}{U} = \frac{137,86W}{27V} = 5,2A$$

I= 5,2A, ohne Berücksichtigung der Verluste!

Gewählt: Mean Well RSP-200-27: 27V, 202.5W

2.3.8 Verkabelung

2.3.8.1 Beleuchtungsplatte Verkabelung

Die Verkabelung wird über der Beleuchtungsplatte geführt. Dort befindet sich der Lüfter, der Temperatursensor, sowie die Anschlüsse für die Power LEDs. Diese werden miteinander verbunden und es wird markiert, welche Leitung welche Komponente steuert. Die Leitungen werden innerhalb der Vertical-Farming Einheit verlegt und mit Klebesockeln befestigt. Dabei wird darauf geachtet, dass genügend Kabellänge für etwaige Veränderungen vorhanden ist. Im fertig montierten Zustand wird die Verkabelung durch eine Nirosta Platte bedeckt. Vier Schlitze sind vorhanden, mit welchen der Lüfter das System mit Frischluft versorgen kann. Es wird darauf geachtet, dass die Verkabelung auch von der Unterseite nicht sichtbar ist, weshalb die überschüssigen Leitungen in Schleifen im Zwischenraum verlegt sind.

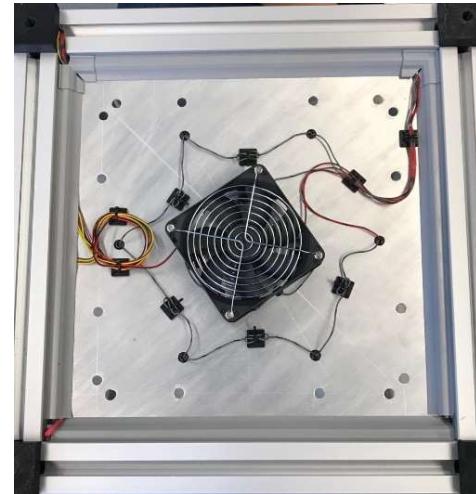
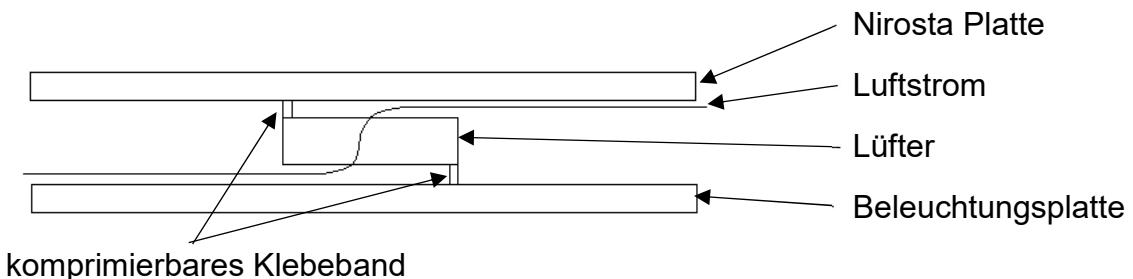


Abbildung 134: Rückseite Beleuchtungsplatte

Außerdem wird der Lüfter so montiert, dass ein Luftstrom vorhanden ist und dieser die warme Luft ableiten kann. Zur Verwirklichung dieser Luftpumulation wird ein spezielles Klebeband, welches sich komprimierbar ist und somit abdichtet, verwendet.



2.3.8.2 Temperatur-/Feuchtigkeitssensorverkabelung

Der DHT22 Sensor befindet sich zwischen Beleuchtungsplatte und Aluminiumprofil, wodurch nicht nur die Temperatur im Anpflanzbereich, sondern auch oberhalb gemessen wird. Die vier Leitungen sind innerhalb der Dichtung der Beleuchtungsplatte angebracht. Ist der Sensor defekt oder liefert unrealistische Werte, kann er aus dem Zwischenraum hinausgezogen und ausgetauscht werden, da er nur gesteckt montiert ist. Die Kabel sind jedoch angelötet und führen über Dichtung und Profil bis in die untere Korpuseinheit.

2.3.8.3 Status LED Verkabelung

Die Status LED ist an der Platte in der Türe der unteren Korpuseinheit angebracht. Dadurch ist diese gut sichtbar, wenn die Vertical-Farming Einheit auf einer Erhöhung in der Küche steht. Die LEDs sind als Duo-LED wie in Abbildung 135 umgesetzt, welche drei Anschlüsse besitzt. Ein Anschluss für die Versorgung der roten LED, einer für die Grüne und der Dritte für die Masse. Es können auch beide gleichzeitig leuchten, um eine neue Farbe zu generieren. Dies ist in diesem Anwendungsfall jedoch nicht nötig.



Abbildung 135: Duo-LED

Da die LED allein nicht montiert werden kann, wird eine Fassung wie in Abbildung 136 eingesetzt. Diese reflektiert das Licht und lässt sich mit einem Gewinde in der Platte fixieren.



Abbildung 136: LED Fassung

2.3.8.4 Netzteil Verkabelung

Da das Projekt auch eine Stromversorgung braucht, muss ein Netzteil nach der oben genannten Leistung und Spannung gewählt werden. Dabei wird ein Mean Well RSP-200-27 Netzteil gewählt, denn dieses deckt alle Anforderungen. Das Netzteil wird mittels doppelseitiger Klettbänder im System montiert.

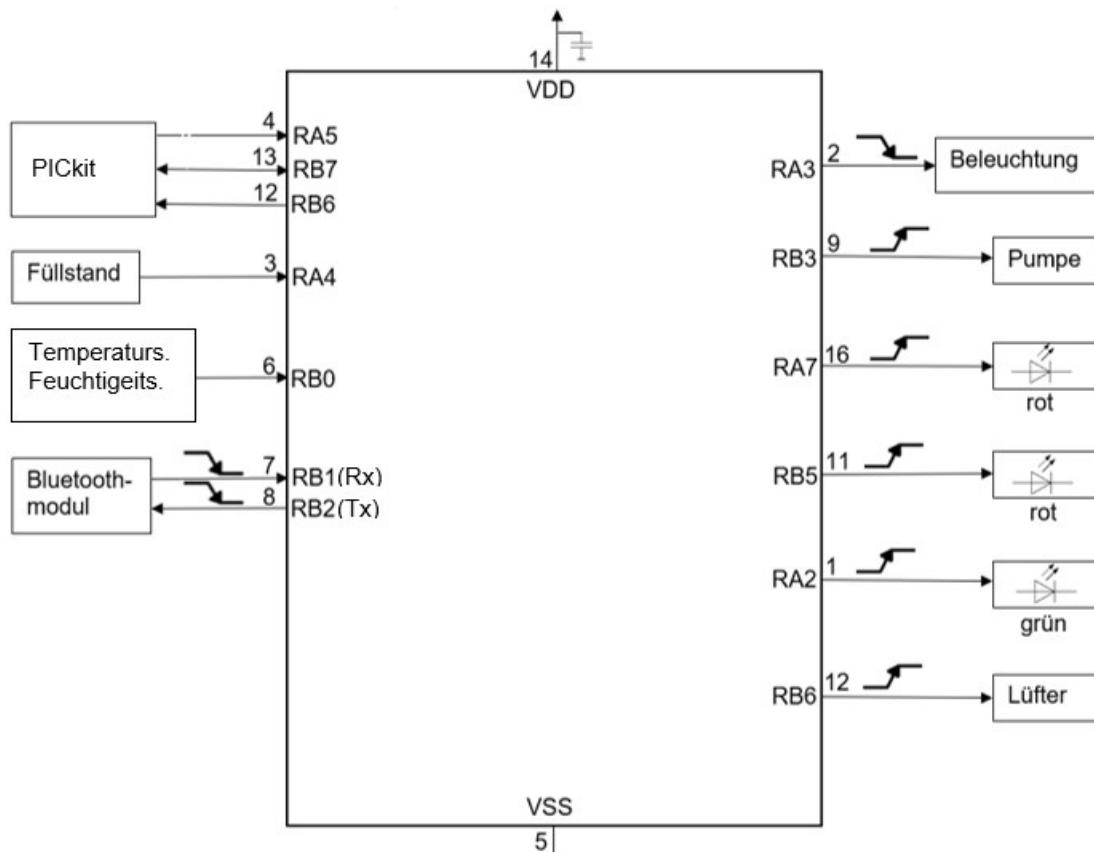
Die Netzversorgung wird über eine Kaltgerätebuchse, die in der Rückwand eingesetzt ist, realisiert. Die gewählte Buchse hat eine Sicherung integriert, welche das Projekt zusätzlich absichert. Weiters befindet sich auf der Buchse ein Hauptschalter, mit welchem man das gesamte System ein- bzw. ausschalten kann. Außerdem kann man einen Kaltgerätestecker im Normalfall nicht falsch anschließen. Damit ist sichergestellt, dass der Schutzleiter immer mit dem Gehäuse verbunden wird.

2.4 Programmierung

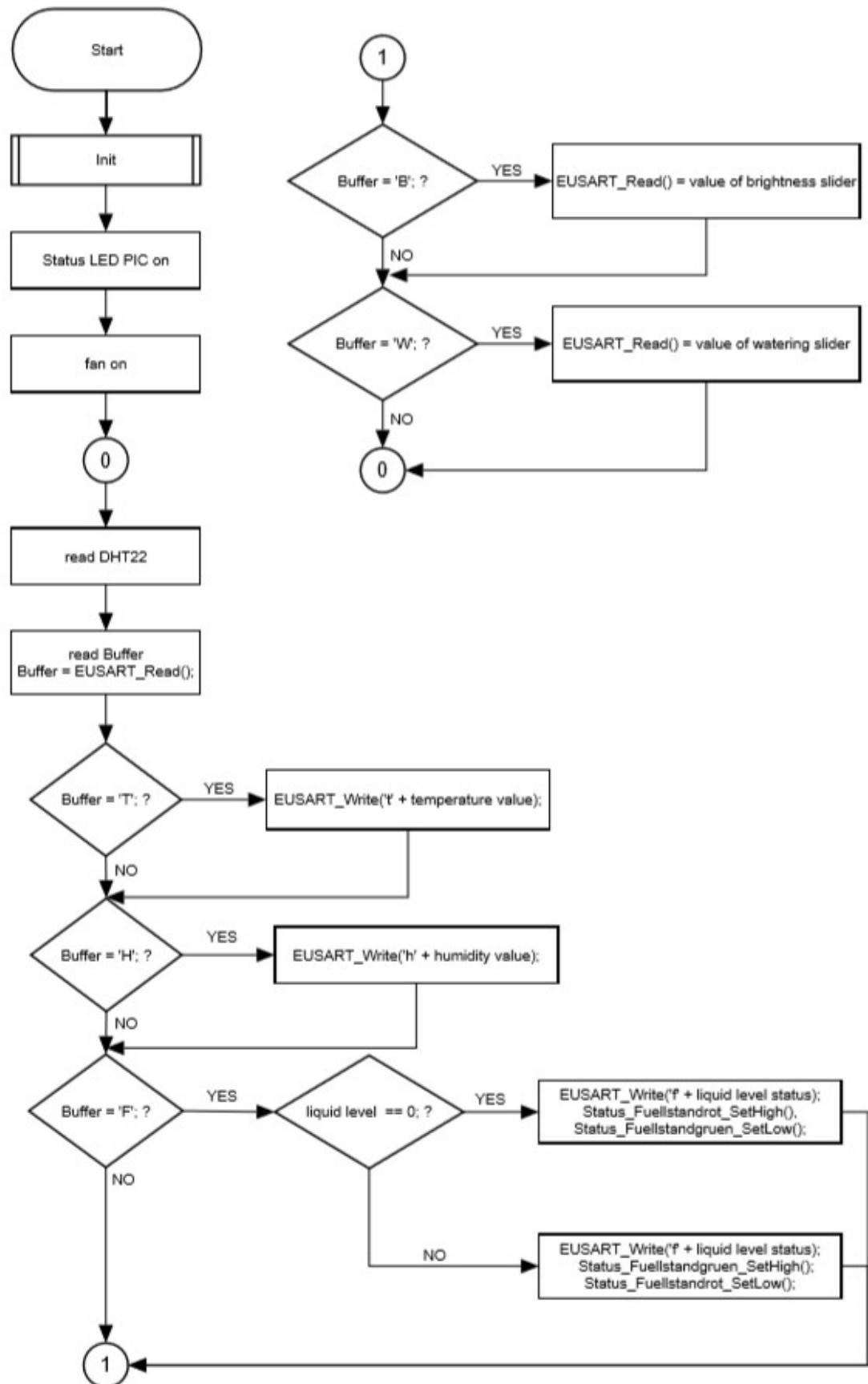
2.4.1 Blockschaltbild

Das Blockschaltbild dient im Prinzip dazu, ein mehr oder weniger komplexes technisches System grafisch darzustellen. Diese schematische Form beinhaltet mehrere Bauteile, welche einerseits Eingänge und andererseits Ausgänge sind. Jene Bauteile werden gleichzeitig mit dem korrekten PIN-Anschluss (im diesem Fall ein Mikrocontroller) versehen. Der Pin für die Versorgungsspannung wird mit VDD bezeichnet während jener Pin der gegen Ground, also Masse geschaltet wird, die Bezeichnung VSS trägt. Bei dem Versorgungspin ist darauf zu achten, dass dieser mit einem Entkoppelkondensator versehen ist. Dieser Kondensator ist dafür zuständig, dass der Mikrocontroller trotz Spannungsschwankungen noch versorgt wird. Wäre kein Entkoppelkondensator vorhanden und der PIC würde für eine kurze Zeit keinen Strom bekommen, würde sich dieser zurücksetzen.

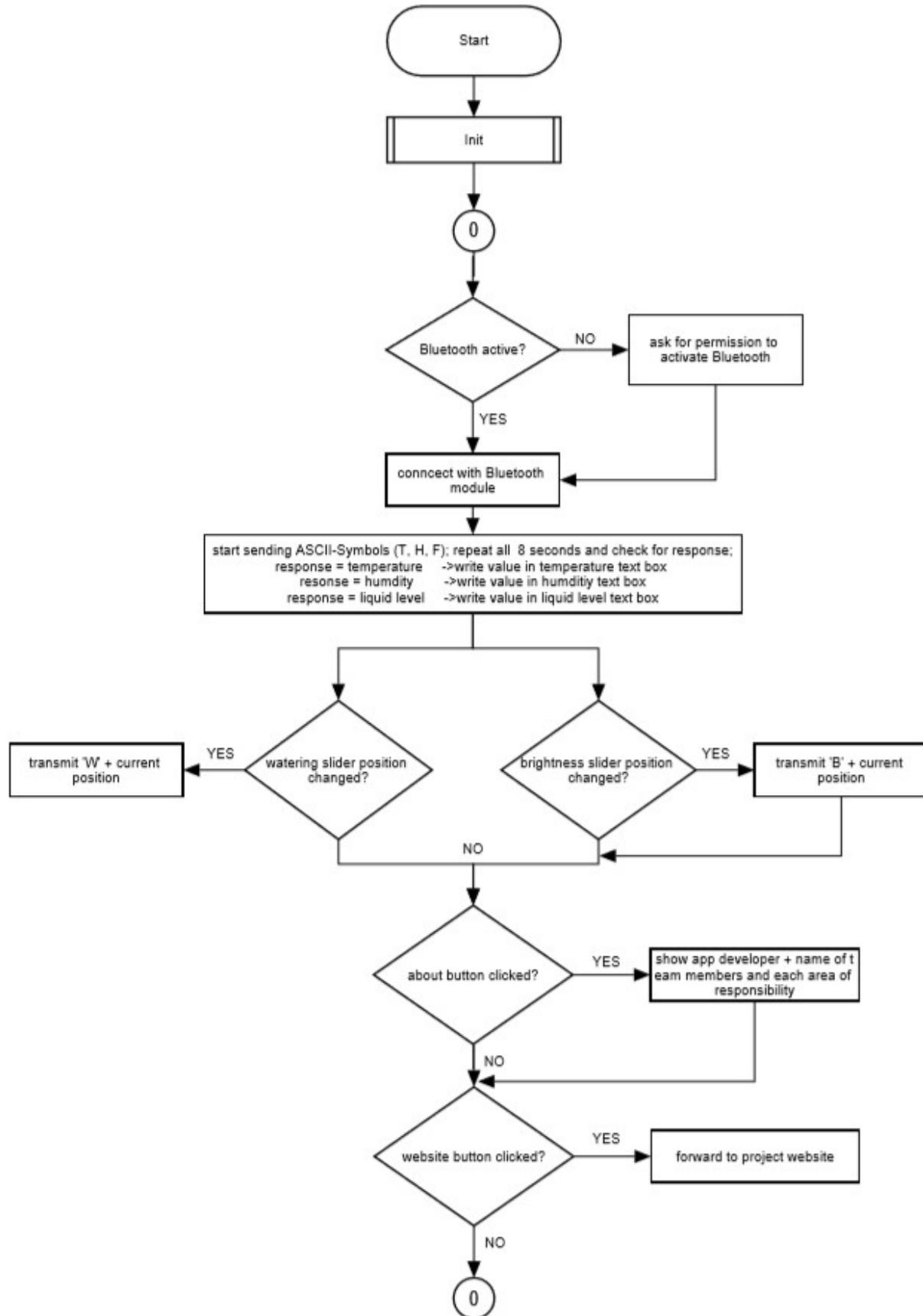
Auf der linken Seite des Blockschaltbildes werden jene Teilnehmer platziert, welche ein Signal an den Mikrocontroller senden, die sogenannten Inputs. Auf der anderen Seite befinden sich alle Bauteile, die ein Signal vom Mikrocontroller zugesendet bekommen und gleichzeitig auch ihren Zustand ändern, das sind die Outputs. Ein Beispiel für einen Output wäre eine LED, welche nach dem Erhalt eines Signals vom PIC zu leuchten beginnt.



2.4.2 Flussdiagramm – Mikrocontroller



2.4.3 Flussdiagramm - Android App



2.4.4 Android App

2.4.4.1 Erstes Layout der App und Funktionalität (MIT App Inventor)



Abbildung 137: Layout nach Starten der App

Abbildung 137 zeigt das Standard Layout nach dem Öffnen bzw. Starten der App. Der Schriftzug „Landwirtschaft für Jedermann“ zeigt sofort, dass diese Applikation ein System im Bereich der Landwirtschaft steuert. Um mit dem Mikrocontroller zu kommunizieren wird eine Bluetooth-Verbindung verwendet. Zuerst muss das Smartphone mit dem dazugehörigen Bluetoothmodul gekoppelt werden. Wird nun die App gestartet, so verbindet sich diese automatisch mit dem Modul, da die Mac-Adresse im Programmcode festgelegt wurde. Ist man verbunden, so erscheint für eine kurze Zeit die Meldung „Connected“. Mit Hilfe des Links „www.vertical-farming.at“ gelangt man durch einen Klick auf die Website der Diplomarbeit Vertical Farming. Im oberen grauen Balken, ist der Name der App zu erkennen. Weiters ist in der linken oberen Ecke, direkt neben dem Wort „Menü“, ein Taster platziert. Ist der Taster nicht gedrückt, befindet sich dieses Symbol „>“ darin. Wird der Taster dann aber gedrückt, so ändert das Symbol die Richtung „<<“.

Der vorher erwähnte Druckknopf fährt nach der Betätigung ein Drop-Down-Menü aus. Dieses Menü beinhaltet vier verschiedene Taster. Zur Auswahl stehen die Taster Bewässerung, Beleuchtung, Sensoren und About. Wählt man „Bewässerung“, so kommt man auf die Schaltfläche zur Steuerung der Wasserpumpe. Diese wird stufenlos mit einem Regler gesteuert werden. Unter dem Reiter „Beleuchtung“ findet man die Steuerung der High-Power-LEDs. Diese LEDs werden ebenfalls stufenlos mit einem Regler heller bzw. dunkler gestellt werden. Mit dem Taster „Sensor“ gelangt man auf eine Oberfläche, auf welcher die Temperatur, die Feuchtigkeit und der Füllstand des Wasserreservoirs abgelesen werden. Der letzte Taster „About“ zeigt nach Betätigung einige Informationen an.



Abbildung 138: Layout nach Öffnen des Drop-Down-Menüs

2 Umsetzung



Abbildung 139: Layout der Auswahlmöglichkeit "About"

Abbildung 139 zeigt die Oberfläche, nachdem der vierte Taster, der „About-Taster“, ausgewählt wurde. Einerseits kann der Entwickler der App herausgelesen werden und andererseits kann das Team, welches das Diplomarbeitsprojekt „Vertical Farming“ verwirklicht hat, erkannt werden. Neben den verschiedenen Rollen jedes einzelnen Teammitglieds sind auch die dazugehörigen Zuständigkeitsbereiche beschrieben.

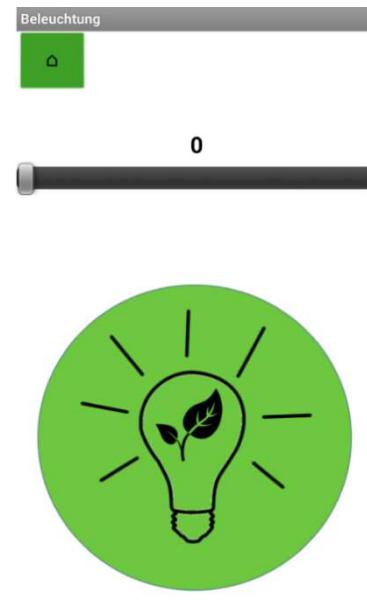
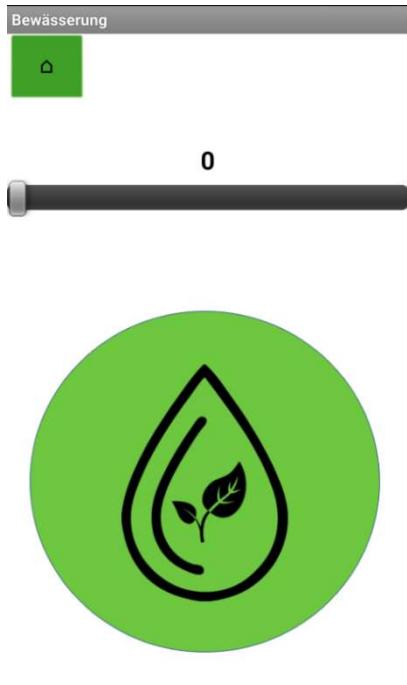


Abbildung 140: Layout der Auswahlmöglichkeit "Beleuchtung"

Auf Darstellung 140 ist die Benutzeroberfläche zur Steuerung der Beleuchtung zu sehen. In der linken oberen Ecke ist ein Taster zu erkennen, welcher zurück zum Startbildschirm führt. Darunter befindet sich ein Schieberegler mit dazugehörigem Textfeld. Dieses Textfeld zeigt den aktuellen Wert an, bei welchem sich der Regler momentan befindet. Um die Oberfläche so zu gestalten, dass definitiv erkannt wird das es sich hierbei um die Steuerung der Beleuchtung handelt, ist ein großes Symbol unter dem Regler platziert worden.



Das Layout für die Bewässerung ist auf Abbildung 141 zu sehen. Diese Illustration ist Abbildung 140 sehr ähnlich. Ein grundlegender Unterschied lässt sich bei dem Symbol, welches sich direkt unter dem Regler befindet, erkennen. Dieses vereinfacht das Herleiten, auf welcher Steuerungsoberfläche man sich momentan befindet, erheblich. Das Textfeld oberhalb des Reglers übermittelt je nach Stellung des Reglers den aktuellen Wert. Der Vorteil gegenüber eines Tasters hierbei ist, dass sich die Wasserpumpe stufenlos einstellen lässt.

Abbildung 141: Layout der Auswahlmöglichkeit "Bewässerung"

Abbildung 142 zeigt die Benutzeroberfläche, auf welchem die Werte der Temperatur, Feuchtigkeit und die des Füllstandes abgelesen werden. Die Temperatur wird in °C ausgelesen und hilft dabei, ein optimales Klima für das Wachstum der Pflanzen zu gewährleisten. Die Feuchtigkeit wird in % angegeben und trägt ebenfalls dazu bei, die optimalen Bedingungen für die Entwicklung verschiedenster Pflanzen sicherzustellen. Dieses Projekt verwendet einen Füllstandsensor mit der Funktionsweise des Schwimmerprinzips (dieses Prinzip wird auf der nächsten Seite genau erklärt). Somit zeigt das letzte Textfeld nur an, ob im Wasserreservoir genug Flüssigkeit ist damit die Wasserpumpe keine Luft ansaugt und der Benutzer rechtzeitig erfährt, wann der Behälter nachzufüllen ist.



Abbildung 142: Layout der Auswahlmöglichkeit "Sensoren"

Funktionsweise des Schwimmerprinzips:

Der Sensor (Schwimmer) weist einen sehr geringen Wert von Dichte auf und kann aufgrund dieser Eigenschaft auf der Wasseroberfläche schwimmen. Ist genug Flüssigkeit im Behälter, so schließt sich der Kontakt im Sensor und dieser gibt entsprechende Messwerte an den Benutzer weiter. Wenn der Füllstand unterhalb des Sensors liegt, dann ist der Kontakt offen und dem entsprechend sind auch die Werte anders. Je nach dem an welcher Stelle der Füllstandssensor angebracht ist, schließt oder öffnet sich der Kontakt. Es lässt sich also grob sagen, dass die Auslösung des Sensors davon abhängt, auf welcher Höhe er in einem Behältnis angebracht wird.

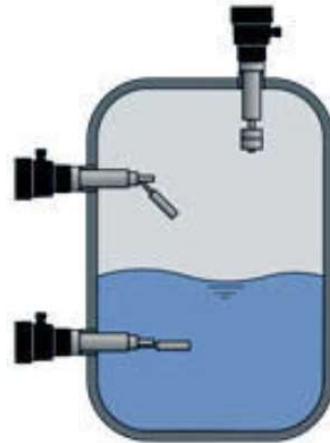


Abbildung 143: Verbilligung der Messung des Füllstandes

2.4.4.2 Verbessertes Layout und angepasste Funktionalität der Applikation

Abbildung 144 zeigt ein verändertes Layout, welches ungefähr dieselbe Funktionalität aufweist, wie die des vorherigen Layouts. Es sind dennoch einige

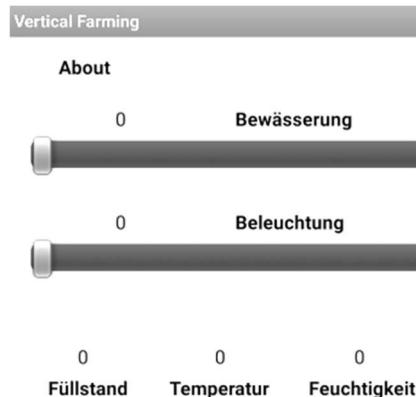
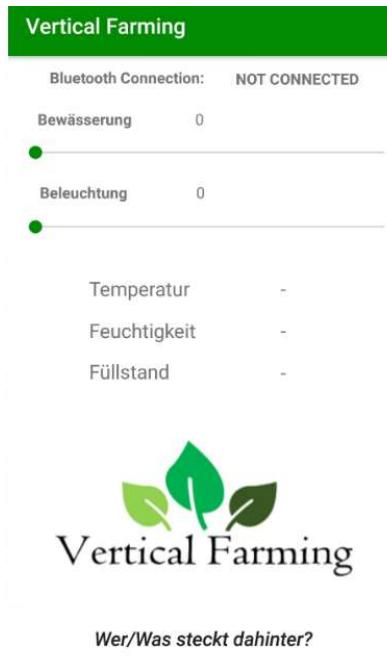


Abbildung 144: Layout der gesamten App

Unterschiede zu erkennen. Einerseits ist das Drop-Down-Menü entfernt und durch einen einzelnen Taster ersetzt worden. Dieser Taster trägt den Namen „About“ und führt einen zu einer Oberfläche, auf welcher die Teammitglieder des Projekts sowie die verschiedenen Rollen jedes Mitglieds und der App-Entwickler aufgeführt sind. Jener Bildschirm ist bereits auf Abbildung 139 zu sehen. Ein weiterer Unterschied ist bei den Schiebereglern zu erkennen. Diese wurden nun auf eine einzige Seite, statt auf zwei verschiedene, platziert. Ebenso wurden die Textfelder für die Ausgabe der Messwerte der Sensoren auf dieselbe Seite angelegt. Abschließend ist noch zu erkennen, dass der Link, mit welchem man zuvor auf die Website des Projektes gelang, verschwunden ist. Dies ist immer noch möglich, allerdings muss dazu nur auf das Logo, welches sich ganz unten in der App befindet, tippen. Die Kommunikation wird mit Bluetooth gewährleistet und erfolgt automatisch nach dem Starten der Applikation.

2.4.4.3 Endgültiges Layout der App (Android Studio)



In Abbildung 145 ist ein Layout zu erkennen, welches dem vorigen vom Aufbau her ziemlich ähnelt. Dieses Applayout zeichnet sich jedoch mit den hochwertigeren Bausteinen und den hohen Individualisierungsmöglichkeiten aus. Einerseits ist es in dieser App möglich, den Verbindungsstatus zwischen Mikrocontroller und der App dauerhaft im Blick zu haben. Die dazugehörigen Textfelder sind unterhalb des Appnamens zu finden. Andererseits gibt es eine große Auswahl an Farben, sodass die App je nach Geschmack individualisiert werden kann. Da sich diese Diplomarbeit im allgemeinen mit Pflanzen beschäftigt wurde, ein passender Grünton gewählt. Bei näherer Betrachtung des Bildes auf der linken Seite ist zu erkennen, dass der Grünton auch mit der Farbe der Schieberegler übereinstimmt.

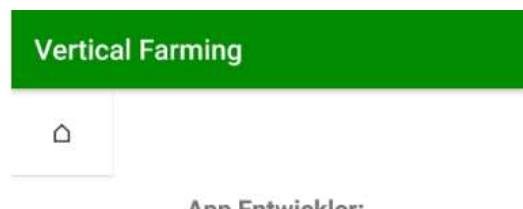
Abbildung 145: Endgültiges App-Layout

Mit zwei Reglern kann die Beleuchtung sowie die Bewässerung stufenlos geregelt werden. Zusätzlich wird in einer Textbox oberhalb des Reglers die aktuelle Position angezeigt. Dabei wurde allerdings ein Minimum von 0 und ein Maximum von 100 definiert.

Unter den Reglern befinden sich sechs Textfelder. Jeweils eines für die Schriftzüge Temperatur, Feuchtigkeit und Füllstand und zusätzlich drei jeweils rechts daneben, wobei diese Textfelder die aktuellen Temperatur-, Feuchtigkeits- bzw. Füllstandswerte anzeigen.

Das Logo der Diplomarbeit stellt in dieser App einen Button dar, welcher nach Betätigung auf die Projektwebsite weiterleitet.

Der Schriftzug „Wer/Was steckt dahinter“ ist ebenfalls ein Button. Mit dem einzigen Unterschied, dass nach Betätigung eine Oberfläche eingeblendet wird, auf welcher der App Entwickler und die Mitglieder des Diplomarbeitsteams zu sehen sind. Siehe Abbildung 146.



Team hinter dem Projekt

- PL Philipp Gasser → Projektmanagement
- PL Stv. Stefan Stetina → Konstruktion
- PM Ulrich Obetshauser → Elektronik
- PM Noah Gruber → Programmierung

Abbildung 146: Wer/Was steckt dahinter?

2.4.4.4 Entwicklung der App mit App Inventor

2.4.4.4.1 Allgemeines über „App Inventor“

Bei diesem Programm handelt es sich um eine spezielle Entwicklungsumgebung um Apps mit verschiedensten Funktionen für Android-Nutzer zu gestalten. Aufgrund der einfachen bzw. benutzerfreundlichen Gestaltung dieser Umgebung, kann jeder Laie in kürzester Zeit eine App entwickeln und diese anschließend sogar im Google Play Store veröffentlichen.

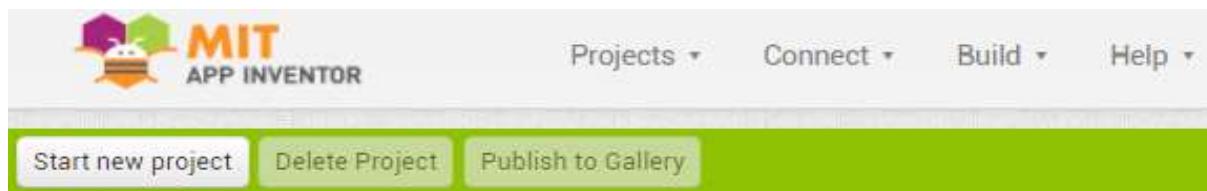


Abbildung 147: Benutzeroberfläche

Es besteht die Möglichkeit mehrere Projekte zu starten und diese auch gleichzeitig direkt auf der Website zu speichern. Unter dem Reiter „Start new project“ kann man ein neues Projekt anfangen und dieses mit einem passenden Namen versehen. Sind bereits mehrere Konzepte abgespeichert, so ist es relativ simpel zwischen den einzelnen Entwürfen zu wechseln, indem man auf den Taster „Projects“ tippt. Nun öffnet sich ein Drop-Down-Menü und unter „MyProjects“ sind alle anderen Projekte zu finden.

Zusätzlich ist es dem Benutzer möglich, wenn dieser einmal nicht weiter weiß, unter dem Taster „Help“ nachzusehen, ob es Lösungsvorschläge bzw. Ansätze für dessen Probleme gibt.

MIT App Inventor eignet sich besonders für Anfänger. Also Personen, welche zum ersten Mal eine App programmieren. Diese Eigenschaft ist darauf zurückzuführen, dass jeder einzelne Baustein, der in der Entwicklungsumgebung zur Verfügung steht, mit dazugehörigen Programmblöcken verbunden ist. Das bedeutet, dass der Anwender sich lediglich überlegen muss, wie die einzelnen Teile zusammengefügt werden müssen, damit dessen Applikation so funktioniert wie es gewünscht ist.



Start new project

Abbildung 148: Oberfläche des Drop-Down-Menüs "Projects"

2.4.4.4.2 Funktion der App

2.4.4.4.2.1 Verbindung mit Bluetooth-Modul

Damit eine Verbindung zwischen dem Mikrocontroller und der App besteht wird ein Bluetoothmodul verwendet. Die App verbindet sich automatisch mit dem Modul, wenn diese ausgeführt wird. Um dies zu gewährleisten muss zunächst die MacAdresse des Bluetoothmoduls eruiert werden. Ist das erledigt, so wird zunächst eine Variable, in diesem Fall heißt die Variable „MacAdress“, definiert. Danach wird diese auf die zuvor herausgefundene Adresse festgelegt. Siehe Abbildung 149.

```
initialize global [MacAdress] to [98:D3:34:90:4C:8E]
```

Abbildung 149: Initialisieren einer Variable

Auf Abbildung 150 sind jene Programmblöcke zu erkennen, welche die zuvor beschriebene Funktion möglich machen. Dieser Code sagt aus, dass wenn die App gestartet wird, soll die Auswahl der verfügbaren Bluetooth Geräte auf die vorher definierte MacAdresse festgelegt werden. Weiters soll eine Meldung mit dem Inhalt „Connected“ ausgegeben werden, wenn sich das Bluetoothmodul ordnungsgemäß mit der App verbunden hat.

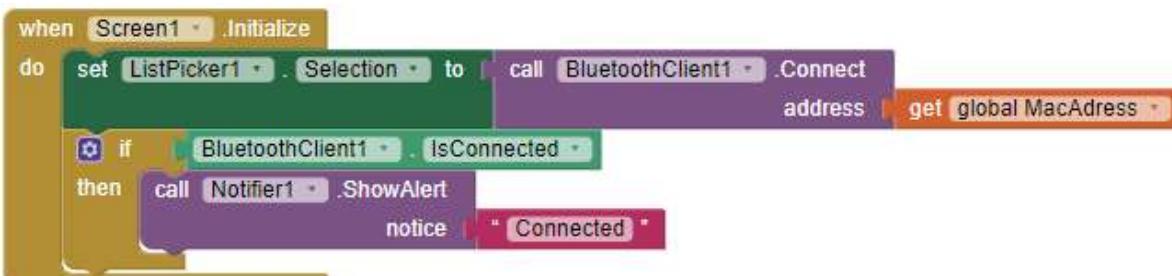


Abbildung 150: Automatisches Verbinden mit Bluetooth

2.4.4.4.2.2 Einlesen der Sensordaten

Das Diplomarbeitsprojekt spezialisiert sich auf das Auslesen von Temperatur-, Feuchtigkeits- und Füllstandwerten. Da es sich bei diesen Werten um ungefähr dieselbe Programmierung handelt, wird nur einer dieser Messwerte als Beispiel erklärt. Anfangs wird wieder eine Variable erstellt und ist mit dem Namen „Temperatur“ festgelegt. Siehe Abbildung 151.

```
initialize global [temperatur] to [0]
```

Abbildung 151: Variable "temperatur"

In der App ist außerdem ein Timer mit einer zuvor definierten Zeit eingebaut. Dieser Timer wird verwendet, damit die Abfrage der Messwerte in einem festgelegten Intervall erfolgt. Die Stoppuhr startet immer dann, wenn die App geöffnet wird. Anschließend wird abgefragt ob die Bluetoothverbindung aktiv ist. Wenn die Verbindung vorhanden ist, dann wird das Ascii-Zeichen „T“ gesendet. Das T dient dazu, dem Mikrocontroller zusagen, dass dieser jetzt die Messerte vom Temperatursensor weiterleiten soll. Danach folgt eine weitere if-Abfrage, welche kontrolliert ob die empfangenen Werte größer null sind. Trifft dies zu, so werden nun die Messwerte in die Variable „temperatur“ geschrieben und in dem Textfeld „Label6“ ausgegeben. Die passenden Programmblöcke sind auf Abbildung 152 zu sehen.

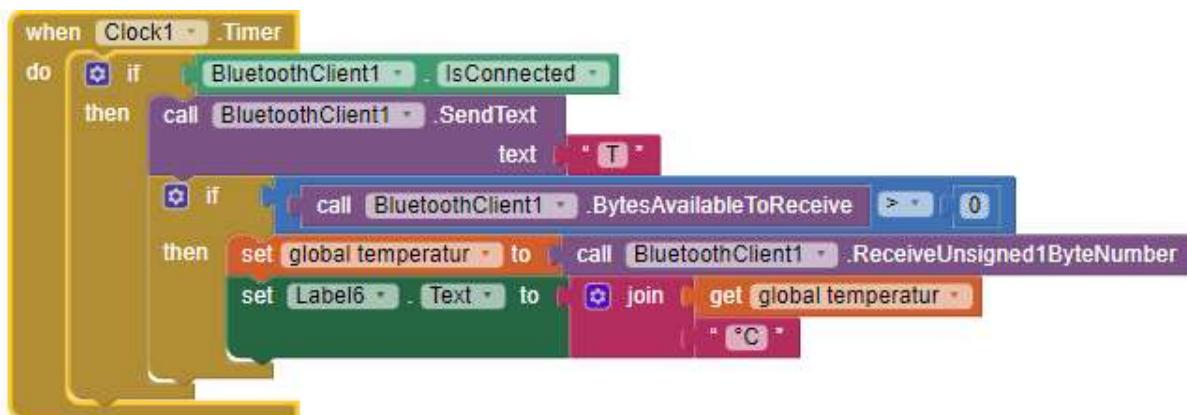


Abbildung 152: Einlesen der Messwerte

2.4.4.4.2.3 Ein- und Ausblenden von Bausteinen

Abbildung 153 zeigt, wie das Ein- und Ausblenden von gewünschten Bausteinen funktioniert. Diese Funktion wird in diesem Projekt verwendet, um in einer Steueroberfläche mehrere haben zu können. Die Programmierung ist einfach zu erklären. Wird der Taster „Button“ gedrückt, dann startet das Programm eine if-Abfrage.

In dieser Abfrage wird am Anfang verglichen ob der Text „△“ im Taster steht. Wenn ja, dann werden die definierten Bausteine Aus- oder Eingeblendet und der Schriftzug im Taster wird in „About“ geändert. Die Funktion dieses Programmteils hängt davon ab, auf welcher Oberfläche in der App man sich gerade befindet.

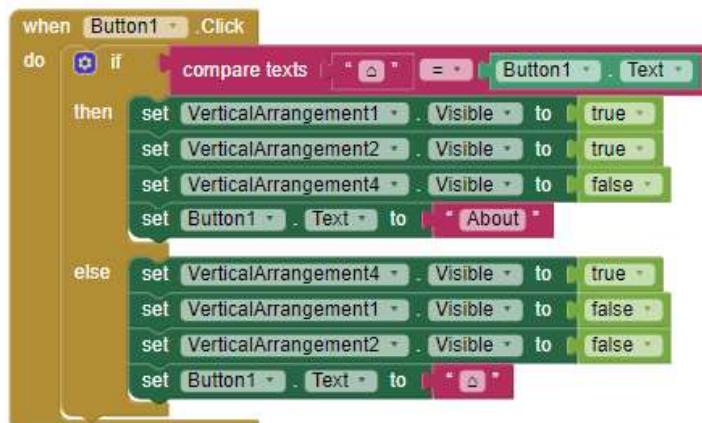


Abbildung 153: Ein- und Ausblenden von Bausteinen

2.4.4.4.2.4 Website-Button

Hinter dem Logo des Projekts „Vertical Farming“ versteckt sich ein normal Taster. Damit man durch ein Antippen des Taster auf die Website gelangt werden jene Blöcke verwendet, welche sich auf Abbildung 154 befinden.



Abbildung 154: Weiterleitung auf Homepage

2.4.4.4.2.5 Regelung der Bewässerung bzw. Beleuchtung

In der App wird ein Schieberegler dazu verwendet, die Wasserpumpe stufenlos zu steuern. Um zu wissen auf welcher Position sich der Regler gerade befindet, zeigt ein eingebautes Textfeld den aktuellen Wert in Dezimalzahlen an. Für den Schieberegler wird ein Minimum von 0 und ein Maximum von 100 fixiert. Damit es überhaupt möglich ist die Pumpe zu steuern und dem Mikrocontroller zu sagen, dass es sich um die Wasserpumpe handelt, wird erneut mit Ascii-Zeichen gearbeitet. Das bedeutet somit, wenn der Regler bewegt wird, egal in welche Richtung, sendet die App ein W und den aktuellen Wert der Stelle, auf welcher sich der Schieberegler befindet, an den PIC. Die Funktionsblöcke dafür sind auf Abbildung 155 zu sehen.

Bei der Beleuchtung handelt es sich um dieselben Funktionsblöcke, allerdings mit dem Unterschied, dass ein B nach der Bewegung des Reglers statt einem W gesendet wird.

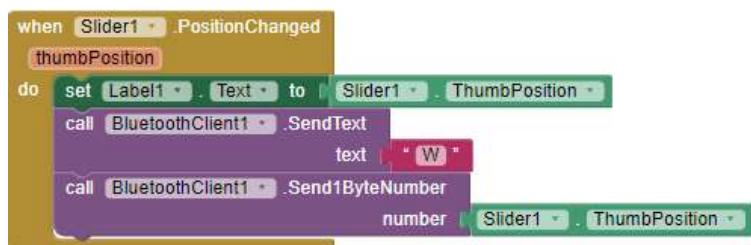


Abbildung 155: Steuerung der Bewässerung

2.4.4.5 Entwicklung der App mit Android Studio

2.4.4.5.1 Allgemeines über das Programm Android Studio



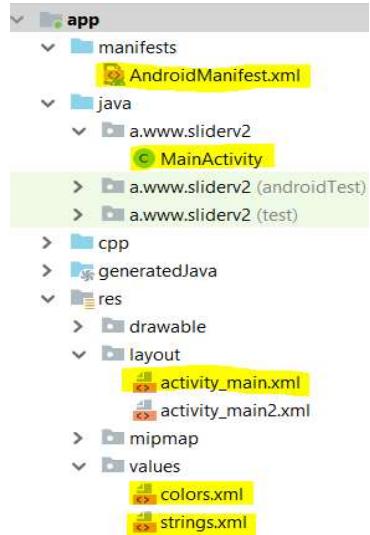
Abbildung 156: Schriftzug des Programms

Dieses Programm wird hauptsächlich dazu verwendet, um Android Apps zu entwickeln. Im Vergleich zu der Entwicklungsumgebung von App Inventor ist die von Android Studio um einiges komplexer. Für das Erstellen des Layouts sind jedoch auch fertige Bausteine verfügbar, die per Drag & Drop an die passende Stelle gezogen werden können. Ein wesentlicher Unterschied zwischen den zwei Programmen, MIT App Inventor und Android Studio, ist die Programmierung. Während in App Inventor fertige Programmblöcke zur Verfügung stehen, muss in Android Studio jede einzelne Programmzeile eigenhändig geschrieben werden. Die zu verwendende Programmiersprache ist Java. Außerdem gilt zu erwähnen, dass Android Studio dem Entwickler beim Personalisieren der App kaum Grenzen setzt. Von der Namensgebung bis hin zur farblichen Gestaltung der App oder der einzelnen Bausteine ist alles möglich.

Zusammenfassend kann gesagt werden, dass es mit diesem Programm möglich ist, hochwertigere Apps zu generieren und das ein gewisses Maß an Ambition gefordert ist, um auch am Ende eine funktionierende App zu erhalten.

2.4.4.5.2 Funktion der App

2.4.4.5.2.1 Allgemeines zu Android Studio



Wenn das Programm gestartet wurde muss zunächst ein neues leeres Projekt erstellt werden. Der Name der neuen Datei sollte möglichst funktionsbezogen vergeben werden. Ist der neue Entwurf geladen, so sind im Funktionsbaum, Abbildung 157, einige Unterordner zu erkennen. Alle benötigten Dateien wurden auf Illustration 156 mit der Farbe Gelb markiert. Unter der ersten Markierung ist die Datei „AndroidManifest.xml“ herauszulesen. In diesem Teil des gesamten Programmcodes werden alle Berechtigungen festgelegt, welche die App vom Smartphone benötigt um ordnungsgemäß arbeiten zu können.

Abbildung 157: Funktionsbaum

Die nächste Markierung ist auf dem Dateinamen „MainActivity“ zu finden. Dies ist der Hauptprogrammteil der Applikation. Darin befindet sich jener Code, welcher die eigentliche Funktion der App fixiert. Damit das Programmieren starten kann, müssen zunächst einige „Bibliotheken“ importiert `import android.widget.TextView;` werden. Als Beispiel dafür, wird in Abbildung 158 eine Zeile dargestellt, mit welcher ein „widget“ für ein Textfeld importiert wird. Nach dem Einfügen aller benötigten „Bibliotheken“ folgt das Hauptprogramm. Das Hauptprogramm inkludiert alle Funktionen der App, dass bedeutet alles was die App später machen soll muss innerhalb dieser Hauptfunktion stehen. Ist dies nicht der Fall, reagiert Android Studio mit einer Fehlermeldung und die App kann nicht ausgeführt werden.

In der Applikation für das Projekt gibt es einige wichtige Funktionen. Dazu zählen, das Herstellen einer Bluetooth-Verbindung zum Mikrocontroller, das Steuern der High-Power-LEDs mit einem Schieberegler, das Steuern der Wasserpumpe ebenfalls mit einem Schieberegler und das Auslesen von Sensordaten. Die zu messenden Werte der Sensoren sind die Temperatur in °C, die Feuchtigkeit in % und den Füllstand. Der Füllstand wird in dem Sinne nicht ausgelesen, der Zustand des Sensors wird lediglich vom Mikrocontroller abgefragt und dementsprechend wird eine Nachricht an die App übermittelt („Okay“ oder „Leer“).

2.4.4.5.2.2 Verbindung mit Bluetooth-Modul

```
<uses-permission android:name="android.permission.BLUETOOTH" />
<uses-permission android:name="android.permission.BLUETOOTH_ADMIN" />
```

Abbildung 159: Berechtigung für Bluetooth

Da dieses Projekt darauf ausgelegt ist, die Beleuchtung und die Bewässerung mittels einer Bluetooth-App zu steuern und die Sensordaten auszulesen, muss die Berechtigung für das Aktivieren von Bluetooth auf dem Smartphone definiert werden. Die dazugehörigen Programmzeilen gehen aus Abbildung 159 hervor.

Außerdem wird sofort nach dem Start der App eine Verbindung mit dem Bluetooth-Modul aufgebaut. Dies funktioniert somit automatisch. Für das Umsetzen dieser automatischen Verbindung, muss lediglich die Mac-Adresse des Moduls eruiert werden. Abbildung 160 zeigt jenen Programmteil, in welchem die Mac-Adresse für die Verbindungsherstellung festgelegt wird.

```
public String DEV_MAC="98:D3:34:90:4C:8E";
```

Abbildung 160: Mac-Adresse Bluetooth-Modul

2.4.4.5.2.3 Regelung der Bewässerung bzw. Beleuchtung

```
int min = 0, max = 100, current = 0;

seekBar2.setOnSeekBarChangeListener(new SeekBar.OnSeekBarChangeListener() {
    @Override
    public void onProgressChanged(SeekBar seekBar, int progress, boolean fromUser)
    {
        current = progress + min;
        textView2.setText("" + current);
        mConnectedThread.write(input: "W" + current + ",");
    }
})
```

Abbildung 161: Code => Regelung der Bewässerung

Aus Abbildung 161 geht der Programmcode für die Steuerung der Bewässerung hervor. Es wird mit einem Regler gearbeitet, welche in Android Studio den Namen „seekbar“ trägt. Zu Beginn wird ein Minimum von 0 und ein Maximum von 100 definiert. Die Position des Reglers ist also eine Zahl zwischen 0 und 100. Dieser Wert wird in einem Textfeld (textView2) mit dem Befehl „.setText“ dargestellt. Mit der Zeile „mConnectedThread.write“ wird das Ascii-Zeichen ‘W’, der aktuelle Wert des Reglers und ein ‘,’ an den Mikrocontroller gesendet. Der Beistrich wird zusätzlich mit gesendet, damit der Mikrocontroller erkennt wo das Ende des empfangenen Datenstrangs ist.

Der Programmblock für die Steuerung der Beleuchtung ähnelt jenem für die Steuerung der Bewässerung sehr, allerdings mit dem einen Unterschied, dass statt dem ‘W’ das Ascii-Zeichen ‘B’ übermittelt wird.

2.4.4.5.2.4 Einlesen der Sensordaten

```

private boolean started;
private Handler timerHandler = new Handler();
private Runnable runnable = new Runnable()
{
    @Override
    public void run()
    {
        mConnectedThread.write( input: "T" );
        mConnectedThread.write( input: "H" );
        mConnectedThread.write( input: "F" );

        if (started)
        {
            start();
        }
    }
};

public void start()
{
    started = true;
    timerHandler.postDelayed(this.runnable, delayMillis: 8000);
}
    
```

Abbildung 162: Abfrage von Sensordaten

Damit die entsprechenden Sensordaten überhaupt in ein dazugehöriges Textfeld eingetragen werden können, muss von der App abgefragt werden ob diese Daten am PIC zur „Abholung“ bereit sind. Dies wird mit einfachen Ascii-Zeichen gehandhabt. Für die Daten der Temperatur wird ein ‘T‘ von der App weggeschickt, für die Messdaten der Feuchtigkeit wird ein ‘H‘ verschickt und für die Füllstanddaten wird ein ‘F‘ versendet. Das ganze wiederholt sich alle acht Sekunden, damit die Messwerte aktualisiert werden. Der dazugehörige Code geht aus Abbildung 162 hervor.

```

if (recString.length() < 5)
{
    return;
}
switch(recString.charAt(0))
{
    case 't':
    {
        Temperatur.setText(recString.substring(1,5) + "°C");
        break;
    }
}
    
```

Abbildung 163: Einlesen von Datenstrang; Bsp.: Temperatur

In Illustration 163 ist zu sehen, dass mit einer if-Anweisung gearbeitet wird um den empfangenen Datenstrang zu analysieren. Wenn der Datenstrang kleiner als 5 Byte ist, dann führt das Programm die if-Anweisung solange durch bis der Datenstrang mindestens 5 Byte lang ist. Danach wird mit einer switch-case-Anweisung weitergearbeitet. Diese ist in Verwendung, weil der Code einerseits übersichtlicher und andererseits einfacher gestaltet werden kann. Die switch-case-Anweisung funktioniert folgendermaßen. Wenn das 0te Byte ein kleines t ist, dann sollen anschließend Byte 1 bis Byte 5 plus die Einheit „°C“ in das Textfeld „Temperatur“ eingetragen werden. Die Ausgabe der Messdaten von der Feuchtigkeit erfolgt genauso wie beim Auslesen der Daten von der Temperatur.

Lediglich beim Füllstand gibt es einen Unterschied. Und zwar muss beim Füllstand noch eine zusätzliche Funktion eingebaut werden. Diese Funktion besteht daraus, dass der Regler der Bewässerung auf 0 gesetzt werden soll wenn der Füllstand Zustand „LEER“ auftritt. Der Programmcode dafür ist in Abbildung 164 zu erkennen.

```
case 'f':
{
    Fuellstand.setText(recString.substring(1,5));
    if (recString.substring(1,5).equals("LEER"))
        seekBar2.setProgress(0);
    break;
}
```

Abbildung 164: Einlesen von Datenstrang; Füllstand

2.4.4.5.2.5 Website-Button

```
public void website(View view)
{
    Intent websitebtnIntent = new Intent(Intent.ACTION_VIEW, Uri.parse("http://www.vertical-farming.at"));
    startActivity(websitebtnIntent);
}
```

Abbildung 165: Programmcode Website-Button

Der Website-Button wird verwendet, um eine Möglichkeit zu schaffen innerhalb der App mit einem Klick auf die Projektwebsite zu kommen. Hierfür wird ein ganz normaler Button verwendet, welcher aber als Hintergrund das Logo von der Diplomarbeit zeigt. Auf Abbildung 165 ist zu erkennen, dass lediglich zwei Programmzeilen dazu benötigt werden. In diesen Programmzeilen ist eine Aktion definiert, welche nach dem betätigen des Website-Button ausgeführt wird.

2.4.4.5.2.6 About-Button

```
public void btnaboutClicked(View view)
{
    Temperatur.setVisibility(View.GONE);
    Feuchtigkeit.setVisibility(View.GONE);
    btnhome.setVisibility(View.VISIBLE);
    appdeveloper.setVisibility(View.VISIBLE);
}
```

Abbildung 166: Ein- und Ausblenden von Bausteinen

Der About-Button ist ein extra, welches dem Benutzer zu einer Oberfläche weiterleitet, auf welcher der App Entwickler, die Mitglieder des Projektteams und deren Aufgaben aufgelistet sind.

Abbildung 166 zeigt ein Beispiel wie einzelne Bauteile ein- und ausgeblendet werden. In der App wird diese Funktion verwendet, damit sich die Textfelder von der Temperatur- oder der Feuchtigkeitsanzeige nicht mit jenen von den Namen der Projektmitglieder überschneiden und die Bluetooth-Verbindung nach betätigen des Tasters bestehen bleibt. Es handelt sich hierbei sozusagen um eine Ersatzlösung für eine zusätzliche externe Oberfläche.

Mit dem Befehl „set.Visibility“ wird die Sichtbarkeit eines Bausteins beeinflusst. In der Klammer steht der gewünschte Zustand. Zur Auswahl stehen VISIBLE für sichtbar, INVISIBLE für unsichtbar und GONE für das komplette verschwinden eines Bausteins.

2.4.5 Mikrocontroller (PIC16F1827)

2.4.5.1 MPLAB X IDE

Diese Anwendung wird dazu verwendet um den Mikrocontroller den Projektanforderungen entsprechend zu programmieren. Es handelt sich hierbei um ein kostenloses Programm, welches von der Firma „Microchip“ entwickelt wurde. Damit mit diesem Programm gearbeitet werden kann, muss ein gewisses Maß an Verständnis für die Programmiersprache C gegeben sein. Zusätzlich ist es wichtig auch ein PIC-Kit zu haben, da es sich dabei um ein Werkzeug handelt, welches benötigt wird um ein fertig programmiertes Programm auf den Mikrocontroller zu übertragen und auszuprobieren.



Abbildung 167: Oberfläche beim Booten des Programms

2.4.5.1.1 Der MPLAB Code Configurator

„MPLAB Code Configurator“ ist ein hilfreiches Tool, welches das Definieren von Ein- und Ausgängen erleichtert und übersichtlich gestaltet. Das Definieren der Ein- und Ausgänge ist insofern wichtig, weil beim Start des Programms alle Ausgänge auf null vorbereitet werden müssen damit auf der Hardwareseite nichts passieren kann. Außerdem können bereits im Code Configurator entsprechende Namen und Funktionen für die jeweiligen Pins vergeben werden. Abbildung 168 zeigt beispielsweise ein paar Pinbelegungen mit verschiedenen Namen und Funktionen, so wie sie in dieser Diplomarbeit auch verwendet werden.

Pin Na...▲	Module	Function	Custom ...	Start High	Analog	Output	WPU	OD
RA2	Pin Module	GPIO	Status_F1	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>		
RA3	CCP3	CCP3		<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>		
RA4	Pin Module	GPIO	Fuellstan	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		
RA7	Pin Module	GPIO	Status_F1	<input type="checkbox"/>		<input checked="" type="checkbox"/>		
RB0	Pin Module	GPIO	IO_RB0	<input type="checkbox"/>		<input checked="" type="checkbox"/>	<input type="checkbox"/>	

Abbildung 168: Definition einiger Ein- und Ausgänge

2.4.5.2 Implementieren des Feuchtigkeit-, Temperatur- und Füllstandssensors

Zur Messung der Feuchtigkeit und der Temperatur in einer Etage wird ein DHT22-Sensor verwendet. Die Entscheidung fiel trotz komplizierterer Initialisierung im Programmcode auf diesen Sensor, weil damit zwei Werte gleichzeitig gemessen werden können. Die Temperatur soll später auf der App in „°C“ und die Feuchtigkeit in „%“ ausgegeben werden. Zunächst muss der Sensor ordnungsgemäß initialisiert werden. Hierfür wurde im Internet nach einem passenden Programmcode gesucht, da das Entwickeln einer eigenen Initialisierung für diesen Sensor zu zeitintensiv gewesen wäre. Der dazu gehörende Programmteil ist im Anhang zu finden. Nach der korrekten Initialisierung des Sensors ist dieser bereits dazu in der Lage zu messen. Da die Werte über eine Bluetooth-Schnittstelle auf die App gesendet werden, müssen die zuvor gemessenen Temperatur- und Feuchtigkeitswerte allerdings noch in ein passendes Register vom Mikrocontroller geschrieben werden. Hierbei handelt es sich um das „Eusart-Register“.

Abbildung 169: Definieren der Variablen

Abbildung 169 zeigt jene

Variablen, welche zum Auslesen der Messdaten festgelegt wurden. Es ist zu erkennen, dass die Variablen als „char“ bzw. „character“ Variablen definiert wurden. Der Datentyp „character“ kann verschiedene Werte in einem Hexadezimal-Bereich von 00 bis FF annehmen, wobei ein Zeichen eines Speicherbereichs aus je 8 Bit besteht.

```

while (1)
{
    increaseTimer();
    readDHT22();
    if(eusartRxCount > 0)
    {
        buffer = EUSART_Read();
        switch(buffer)
        {
            case 'T':
                {
                    EUSART_Write('t');
                    EUSART_Write(Temperature[7]);
                    EUSART_Write(Temperature[8]);
                    EUSART_Write(Temperature[9]);
                    EUSART_Write(Temperature[10]);
                    break;
                }
        }
    }
}

```

Abbildung 170: Messwerte in EUSART-Register schreiben

Als praktisches Beispiel wird der Programmteil für das einschreiben der Sensordaten von der Temperatur in das „Eusart-Register“ auf Abbildung 170 dargestellt.

Die Deklaration der Variablen für den Füllstand ist nicht unbedingt nötig, da sich der Wert, welcher an die App übermittelt wird, nicht ändert. Es wird lediglich das Wort „OKAY“ gesendet wenn der Füllstand im grünen Bereich liegt und wenn der Wasserspiegel zu gering ist wird das Wort „LEER“ gesendet. Somit könnten an der passenden Stelle des Programms in das „Eusart-Register“ einfach die Buchstaben des jeweiligen Wortes fixiert werden.

2.4.5.3 Bluetooth Kommunikation zwischen Mikrocontroller und App

Für die Kommunikation zwischen der Android App und dem Mikrocontroller wird das EUSART-Register verwendet. In dieses Register werden alle zu übermittelnden Daten hineingeschrieben. Das beinhaltet, von PIC zu App, die Messdaten der Sensoren und von App zu PIC, der Positions Wert von den Reglern.

Als Übertragungsmedium ist ein HC-06 Bluetooth-Modul in Verwendung. An den Einstellungen des Moduls wurde die Baudrate und der Name geändert. Die Baudrate ist auf 9600 festgelegt und der Name des Moduls auf „Vertical Farming“. Damit die Kommunikation auch gewährleistet ist, muss sichergestellt werden, dass die Baudrate des Bluetooth-Moduls mit der des Mikrocontrollers übereinstimmt.

2.4.5.4 Bewässerung und Beleuchtung der Einheit

Die Bewässerung und die Beleuchtung einer Anpflanzeinheit wurde mit einer PWM verwirklicht. Bei der Pulswidtemodulation handelt es sich um eine Modulationsart, bei welcher ein digitales Signal so moduliert wird, dass man einen analogen Wert erhält. Das Signal wechselt zwischen einer logischen „0“ und einer logischen „1“ . Die Periodendauer der Zustände ist unterschiedlich und hängt vom „duty cycle“, dem Auslastungsgrad, ab. Der „duty cycle“ berechnet sich aus der Einschaltzeit „ T_{on} “ und der Periodendauer „ T “. Für die Periodendauer wird eine Frequenz benötigt.

Die Formel für den Auslastungsgrad lautet: $Duty\ Cycle = \frac{T_{on}}{T} * 100$

Die Formel für die Periodendauer lautet: $f = \frac{1}{T} \rightarrow T = \frac{1}{f}$

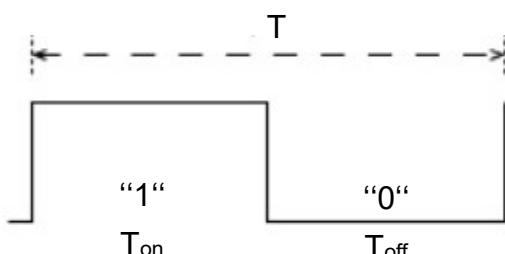


Abbildung 171: Signalzustand "1" und "0"

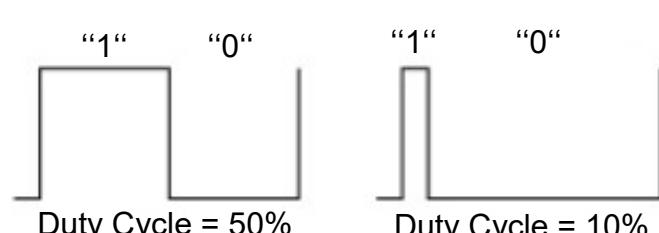


Abbildung 172: Auswirkung des Duty Cycles

Bild 171 illustriert die zwei verschiedenen Zustände T_{on} und T_{off} während Bild 172 zeigt, wie sich der Auslastungsgrad auf das Signal auswirkt.

Im Programmcode musste nun eine Lösung gefunden werden, damit sich die Bewässerung und auch die Beleuchtung mit der Android App steuern lassen. Hierfür wurde ganz einfach auf Ascii-Zeichen zurückgegriffen. Für das Steuern der Wasserpumpe wurde das Ascii-Zeichen „W“ ausgewählt und für das Steuern der High-Power-LEDs wurde ein „B“ festgelegt. Das „W“ wurde aufgrund des englischen Wortes „watering“ gewählt und „B“ aufgrund des englischen Wortes „brightness“.

Um hierfür einen passenden Programmteil zu schreiben könnte einerseits mit mehreren If-Abfragen und andererseits mit einer switch-case-Anweisung gearbeitet werden. Im Laufe dieses Projekts fiel die Entscheidung allerdings auf die switch-case-Anweisung, weil sich das Programm dann viel übersichtlicher gestalten lässt.

```

case 'B':
{
    NOP();
    unsigned char buffer1[] = {0, 0, 0, 0};
    for (unsigned char i = 0; i < 4; i++)
    {
        buffer1[i] = EUSART_Read();
        if (buffer1[i] == ',')
        {
            switch (i)
            {
                case 1: //0-9
                {
                    x = buffer1[0] - 0x30; //bsp.: B5 received
                    break;
                }
                case 2: //10-99
                {
                    x = 10 * (buffer1[0] - 0x30); //10er Stelle
                    x += buffer1[1] - 0x30; //1er Stelle
                    break; //bsp.: B10 received
                }
                case 3: //100-255
                {
                    x = 100 * (buffer1[0] - 0x30); //100er Stelle
                    x += 10 * (buffer1[1] - 0x30); //10er Stelle
                    x += buffer1[2] - 0x30; //1er Stelle
                    break; //bsp.: B100 received
                }
            }
            break;
        }
    }

    x = 399 - (x - 0) * (399 - 0) / (100 - 0) + 0; //convert % to value between 0 - 399; active low
    PWM3_LoadDutyValue(x); //brightness
    break;
}

```

Abbildung 173: case 'B' brightness

Illustration 173 zeigt den Programmcode für die Beleuchtungssteuerung. Bei der Konfiguration des entsprechenden Pins für die PWM muss darauf geachtet werden, dass die PWM auf 100% festgelegt ist, weil die in dieser Diplomarbeit verwendeten LEDs „active low“, also aktiv wenn eine 0 anliegt, sind. Aus der Konfiguration ist außerdem herauszulesen, dass sich bei 100% Pulsweitenmodulation ein Zahlenwert von 399 ergibt.

Im Code wird zunächst abgefragt ob der Mikrocontroller das Ascii-Zeichen „B“ von der App erhalten hat. Ist dies der Fall, dann springt der Mikrocontroller in die Funktion hinein und führt diese aus.

In der Funktion selbst ist eine Variable „x“ definiert, welche über die Bluetooth-Schnittstelle das Zeichen und einen Wert einliest. Der Wert, welcher mit dem Zeichen „B“ mitgeschickt wird, hängt von der momentanen Stellung des Schiebereglers in der Applikation ab. Weil die App nur einzelne Zeichen schickt, das heißt nur das ASCII-Zeichen „B“ dann die Zahl „1“ und dann die Zahl „0“. Dies ist ein Beispiel für den Regler der Beleuchtung, wenn dessen Position den Wert 10 hat.

Somit muss dem PIC mitgeteilt werden, wann der empfangene String endet. Dazu wird zusätzlich ein Beistrich von der App mitgeschickt. Im Code wurde es dann so gehandhabt, dass der Beistrich das Ende des Strings darstellt. Wurde der Beistrich erkannt, so wird eine weitere switch-case-Anweisung durchgeführt. In dieser Anweisung wird in „case 1“ eine einstellige Zahl, in „case 2“ eine zweistellige Zahl und in „case 3“ eine dreistellige Zahl berechnet. Beispiele dazu sind jeweils neben den dazugehörigen Programmzeilen, als Kommentare in Abbildung 171 zu sehen.

Unter der switch-case-Anweisung in „case B“ befindet sich eine weitere Berechnung, die einen %-Wert in eine Zahl umrechnet. Dies ist notwendig, da dem PIC ein normaler Zahlenwert übermittelt wird und eine PWM normalerweise in % ausgelesen wird. Der gesamte Programmcode dieses Projekts beinhaltet mehr als nur den „case“ für die Beleuchtung. Es gibt noch einen „case“ für die Bewässerung und jeweils einen für die Daten von der Feuchtigkeitsmessung, der Temperaturmessung und der Füllstandmessung.

Der Programmcode für die Steuerung der Bewässerung unterscheidet sich nur sehr gering von jenem auf Abbildung 173. Einerseits kommt statt der Variable „x“ die Variable „y“ vor und andererseits wird bei der Umrechnung von einem %-Wert in einen Zahlenwert nicht davor mit „399 - ...“ subtrahiert. Dies ist auf den Zustand zurückzuführen, dass die Wasserpumpe im Gegensatz zu den High-Power-LEDs nicht „active low“ ist.

2.4.5.5 Status LEDs mit dazugehöriger Statusdefinition (inkl. Status LED PIC)

Die LEDs zeigen in diesem Projekt den Status des Füllstandes im Wasserreservoir an. Das heißt es gibt eine grüne LED, welche symbolisiert, dass der Wasserspiegel noch im grünen Bereich liegt und ein rote LED, welche dem Benutzer mitteilt, dass das Wasserbecken neu befüllt werden muss. Zusätzlich symbolisiert eine weitere LED den aktiven Zustand des Mikrocontrollers. Auf Abbildung 174 ist der Programmteil illustriert, welcher für das Leuchten bzw. das nicht Leuchten der LEDs zuständig ist. Am Anfang befindet sich eine If-Abfrage, mit welcher verglichen wird ob der Zustand des Füllstandssensors „1“ ist oder nicht. Trifft es zu, dass der momentane Zustand des Sensors „1“ ist dann soll die grüne LED auf „High“ gesetzt bzw. eingeschaltet werden. Ist jedoch das Gegenteil der Fall, also der aktuelle Zustand des Füllstandssensors ist „0“, dann wird die rote LED aktiviert.

```

if(Fuellstand_GetValue() == 1)
{
    Status_Fuellstandgruen_SetHigh();           //level of fluid okay. Green LED power on.
    Status_Fuellstandrot_SetLow();
}
else if(Fuellstand_GetValue() == 0)
{
    Status_Fuellstandrot_SetHigh();            //level of fluid not okay, Red LED power on.
    Status_Fuellstandgruen_SetLow();
}
    
```

Abbildung 174: Programmcode für Statusleds

Auf Darstellung 175 ist zusehen, dass es lediglich eine Zeile für die Ansteuerung der Status LED des PICs gibt. Diese Zeile befindet sich direkt nach der Initialisierung des Systems.

```

void main(void)
{
    // initialize the device
    SYSTEM_Initialize();
    Status_PIC_SetHigh();
}
    
```

Abbildung 175: Status LED für PIC

2.4.5.6 Lüfter

Da die High-Power-LEDs, welche für die Beleuchtung verwendet werden, in einer überschaubaren Zeit relativ warm werden, wird eine Kühlung der LEDs benötigt. Gekühlt werden die LEDs mit der Edelstahlplatte auf welcher sie verbaut sind, aber zusätzlich auch noch mit einem Lüfter der ebenfalls auf der selben Platte verbaut ist. Der Lüfter wurde so programmiert, dass dieser aktiviert wird, wenn das komplette System startet. Also direkt nach dem Initialisieren des Systems. Die Idee dahinter ist, dass nicht nur die LEDs gekühlt werden sondern, dass auch in der Anpflanzeinheit ein kleiner Luftzug entsteht.

```
void main(void)
{
    // initialize the device
    SYSTEM_Initialize();
    Status_PIC_SetHigh();
    Luefter_SetHigh();
```

Abbildung 176: Aktivieren des Lüfters

3 Lessons Learned

3.1 Philipp Gasser

Wie bin ich an die Aufgabe herangegangen?

Bereits in den Sommerferien war es mir wichtig die Projektplanung abzuschließen. Der Hintergedanke hierbei bestand darin, dass das Projektteam pünktlich zu Schulbeginn mit der Umsetzung und somit mit der Fertigung beginnen konnte. Außerdem konnten Fragen wie „Welches Konzept setzen wir jetzt um?“ vorab geklärt werden. Dies war essenziell, um Kommunikationsprobleme zu vermeiden.

Zudem habe ich alle Termine etwas früher angesetzt, um genügend Puffer bei möglichen Problemen des Projektablaufes zu haben. Jedoch hätte ich die jeweiligen Puffer etwas kleiner einplanen sollen, da sich jede kleinste Verzögerung sofort im SOLL-IST Vergleich des Meilensteinplans widerspiegelte.

Letztlich war es mir außerordentlich wichtig mit Kooperationspartnern zusammenzuarbeiten, um einerseits unser Know-How im Bereich des „Vertical Farmings“ zu erweitern und andererseits Rabatte, Sach.- oder Geldmittel zu bekommen. Dies hat zwar sehr viel Zeit in Anspruch genommen, doch hat sich rückblickend definitiv gelohnt.

Was sagt mir das Ergebnis?

Das Ergebnis der Diplomarbeit ist genauso, wie ich es mir vorgestellt hatte. Bei jeglicher Art von Gefährdung des Projektverlaufes wurde ein Management Summary verfasst, welches die Probleme im Projekt erläutert und die dazugehörigen Gegenmaßnahmen. Darüber hinaus gab es auch regelmäßige Jour Fixes, in welchen der Projektstatus besprochen wurden und die nächsten Schritte geplant wurden. Ich bin stolz, dass wir die erste abgenommene Diplomarbeit der Abteilung „Mechatronik“ des Schuljahres 2018/18 waren. Durch diese „Auszeichnung“ erlaube ich es mir zu sagen, dass die Arbeit dieser Diplomarbeit sinngemäß erfüllt wurde.

Was verlief gut, was schlecht und was würde ich nächstes Mal anders machen?

Das einzige Kriterium, welches verbessert werden könnte, ist die Kommunikation und die Arbeitsbereitschaft des Teams. Oftmals kam es zu Missverständnissen, welche man mit einer einfachen Verständnisfrage, wie „Ist dir wirklich alles klar?“ vermeiden hätte können. Bei der Arbeitsmotivation am Anfang habe ich jedoch versucht das betroffene Projektmitglied zu motivieren, wenn auch nicht immer mit Erfolg.

3.2 Stefan Stetina

Was verlief wirklich gut?

Sowohl die Zusammenarbeit mit diversen Firmen, die teilweise auch ein Sponsoring lieferten als auch die mechanische Planung und der Aufbau des Projektes Vertical Farming verliefen annähernd ohne Probleme. Die gemeinsame Planung mit dem Projektteam der verschiedenen Konzepte sowie die Planung und der Aufbau des finalen Konzepts wurde schon in den Sommerferien umgesetzt, wodurch wir ausreichend Zeit für den mechanischen Aufbau zur Verfügung hatten.

Was hat nicht funktioniert?

Bei der Entwicklung des Drehsystems hatten wir anfangs die Schwierigkeit, das bei der Planung die Radien zu klein angenommen wurden, doch nach weiterer Zusammenarbeit und Beratung mit dem Lehrpersonal konnte eine zweite Version erstellt werden, wodurch das System anschließend erfolgreich umgesetzt werden konnte.

Teilweise gab es im Team intern diverse Verständigungsprobleme, wie beispielsweise die Bewässerung, welche direkt die Mechanik und Programmierung miteinander verknüpft. Hierbei bestand das Problem, dass es anfänglich geplant war, die Bewässerung von oben in das System einzuführen und zwei verschiedene Kreisläufe für Zu- und Abfluss in den Profilen zu führen. Da allerdings relativ rasch klar war, dass keine weiteren Einheiten gebaut werden, entschieden wir, die Bewässerung von unten in das System einzuführen, wodurch Zu- und Abfluss durch denselben Schlauch geführt werden.

Optional wurden ein bis zwei weitere Einheiten geplant, jedoch erfolgte die Umsetzung dieser aus Zeit und Budgetmangel nicht.

Was habe ich gelernt?

Ich habe bei der Umsetzung des Projekts „Vertical Farming“ gelernt, wie wichtig eine gute Projektplanung ist, da sie direkt mit der Umsetzung zusammenhängt. Weiters war es spannend als Schüler in Kooperation mit Firmen zu arbeiten und in weiterer Folge ein Sponsoring von diesen zu erhalten. Beim Aufbau des Projekts war es interessant allein nach seinen eigenen Vorgaben mit diversen Maschinen zu arbeiten und somit den Aufbau voran zu treiben.

3.3 Ulrich Obetzhauser

Wie verlief die Diplomarbeit?

Im Grunde genommen war die ganze Diplomarbeit zeitlich gut eingeteilt. Von der Suche der Sponsoren über restliche Finanzierung wurde im Detail alles berücksichtigt und auch eingehalten. Die Elektronik jedoch, sollte recht rasch umgesetzt werden, was sich aber zeitlich überhaupt nicht realisieren gelassen hat. Als Gründe für die mangelnde Zeit war natürlich das ebenso mangelnde Wissen und den schulischen Stress, den jeder von der Diplomarbeitsgruppe nebenbei ausgesetzt war.

Außerdem gab es innerhalb der Gruppe teilweise Kommunikationsschwierigkeiten, welche ebenso Zeit und Nerven gekostet haben. Doch nach paar Tagen waren diese Unklarheiten geklärt und das Team wieder in Einklang. Außerdem wollten wir anfangs mehrere Anpflanzeinheiten bauen, jedoch nach dem Einkauf der Teile der Ersten war klar, dass das wohl die einzige Etage unseres Projektes bleiben wird.

Schwierigkeiten, welche aus dem Weg geschafft wurden, waren auch kaputte Konstantstromquellen, welche sich erst kurz vor Ende des Projektes als defekt erwiesen haben. Da die Platine schon mit dem Footprint der gekauften Stromquelle bestellt wurde, mussten wir diese noch einmal bestellen. Doch hierbei musste auf die schnellste Bestellung zurückgegriffen werden, da die Zeit der Abnahme immer näher rückte. Schlussendlich konnten wir die neuen Konstantstromquellen innerhalb kurzer Zeit beziehen, testen und das Projekt abnehmen lassen.

Was habe ich gelernt?

Bei dem Projekt „Vertical Farming“ habe ich viel Neues gelernt, beispielsweise auch bei dem Aufbau und der Herangehensweise von Schaltungen. Dank den Lehrern, von welchen ich tatkräftig unterstützt wurde, konnte ich mir eben neues Wissen aneignen, durch welches das Projekt erst realisierbar geworden wurde. Außerdem habe ich gelernt, dass die Testphase und das Ausprobieren, sowie die Bestellung von neuen Komponenten viel mehr Zeit als gedacht benötigen.

3.4 Noah Gruber

Wie bin ich an die Aufgabe herangegangen?

Zu Beginn habe ich mich allgemein mit verschiedenen Programmiersprachen auseinandergesetzt um deren Syntax zu verstehen. Das half einerseits beim Programmieren des Microcontrollers und anderseits auch beim entwickeln der Android-App. Die Codierung des Mikrocontrollers erforderte sehr viel Fachwissen. Dabei half nicht immer das world wide web, somit habe ich entweder bei meinem individuellen Betreuer nachgefragt oder bei einigen Kollegen, welche mehr Erfahrung und Know-How in diesem Bereich haben.

Die App hingegen konnte ich im Großen und Ganzen eigenständig entwickeln. Ich musste mich lediglich für eine Entwicklungsumgebung entscheiden. Angefangen habe ich mit „MIT App inventor“. Nach mehreren Tagen stellte ich aber fest, dass mir diese Umgebung zu simple ist und die App auch nicht wirklich hochwertig aussieht, also entschied ich mich für Android Studio.

Beim Lösen der Problemstellung war ich allerdings etwas schleißig. Ich hatte keinen exakten Plan wann, wie etwas erledigt sein sollte. Das spiegelte sich dann in der Umsetzung wieder, welche mir gegen Ende etwas Stress bereitete.

Was sagt mir das Ergebnis?

Das Endergebnis ist dann aber sehr positiv ausgefallen. Nach Fertigstellung der Diplomarbeit kann ich sagen, dass ich mit meiner Arbeit sehr zufrieden bin. Trotzdem muss ich auch erwähnen, dass mir aufgefallen ist wie wichtig es ist, die einzelnen Schritte, welche einem zum Ziel bringen, so gut wie möglich zu planen um während der Durchführung nicht in Stress zu geraten.

Was verlief gut, was schlecht und was würde ich nächstes Mal anders machen?

Zusammenfassend kann gesagt werden, dass einiges gut aber auch ein paar Sachen nicht so gut gelaufen sind. Auf der einen Seite ist das Ausarbeiten von Lösungen ziemlich gut abgelaufen und ich wusste genau wo ich nachforschen muss um mir fehlendes Wissen anzueignen. Zudem ist der Verlauf des Projektes auch nur positiv zu bewerten. Denn wir haben das Projekt termingerecht abgeschlossen und ich bin stolz sagen zu dürfen, dass mein Team und ich zu den ersten der vielen Diplomarbeitsgruppen gehören, welche bei der Abnahme erfolgreich waren. Auf der anderen Seite muss ich sagen, dass die Kommunikation innerhalb des Teams etwas schlecht war. Dort gab es Missverständnisse und hier ein paar Ausfälligkeit, dass sollte eigentlich nicht vorkommen. Die Kommunikation in einer Diplomarbeitsgruppe ist essentiell für eine gute Zusammenarbeit und das werde ich mir in meinem nächsten Projekt zu Herzen nehmen und verbessern.

Anhang A

Agenda/Protokoll

Projekttitle: <Vertical Farming>
Auftraggeber: <Martin Meschik / Unternehmen>
Auftragnehmer: <Philipp Gasser / Unternehmen>
Schuljahr: <2018/19> Klasse: <5BM>

VERSION	DATUM	AUTORIN/AUTOR	ÄNDERUNG
v1.0	26.08.2018	<Philipp Gasser>	<Beschreibung der Änderung>

Thema/Ziel des Meetings:		Status des Projektes		
Datum:	26.08.2018	Dauer (Uhrzeit):	2h (17:00)	Ort: Stockerau
Teilnehmer ¹		Position / Funktion		Aufgabenbereich
Philipp Gasser		PL		Projektmanagement, Konstruktion
Stefan Stetina		PL Stv.		Konstruktion, Mechanik

Verteiler ¹
Siehe obige Anwesenheit

Tagesordnungspunkte							
#	Zeit	Dauer	Tagesordnungspunkt	Vor. ²	Erg ³	Ver.	Deadline
01	17:00	30'	Einleitung und Aufnahme des Projekt-Status				

Aufnahme vergangener Beschlüsse (Konzepte)	B/D	F	PG	-
--	-----	---	----	---

02	17:30	90'	Besprechung				
Konzept 2: Hydroponik - Konstruktion: Robotunits oder 4-Kant-Profile - Elektronik: „ansteuerbare Durchlasspumpe“				-	B/D	A	-
Aufgabenliste							
#	TO DO				Ver.	Deadline	
01	Stahlunternehmen kontaktieren				PG	29.08.2018	
02	Unternehmen im Bereich „ansteuerbare Durchlasspumpen“ kontaktieren				PG	29.08.2018	
03	Stückliste für Robotunits Profile (Profile inkl. Verbindungselemente)				SS	29.08.2018	

1) Personenangaben ohne Titel, 2) Vorbereitung: B - Bericht (mündlich), H - Handout, P - Präsentation, D - Dokument (digital), M (Multimedia)
3) Ergebnis: A - Aktion, B - Beschluss, E - Empfehlung, I - Information, O - Offener Punkt, F - Erledigt (fertig)
Abnahme des Protokolls erfolgt automatisch, wenn binnen 48 Stunden kein Einspruch erfolgt.

Dokumentenverzeichnis			
#	Titel	Dateiname	Ablageort
01	Protokoll des Meetings v. 1.0.	p_protokoll_20180826.pdf	https://htl3r.sharepoint.com/sites/Vertical-Farming2/Freigegebene%20Dokumente/Forms/AllItems.aspx?id=%2Fsites%2FVertical-Farming2%FFreigegebene%20Dokumente%2F1_Projekmanagement%2F1%2E4_Protokolle%20%26%20Templates%2F1%2E4%2E1_Protokolle
02			

Philipp Gasser-----
Stefan Stetina

1) Personenangaben ohne Titel, 2) Vorbereitung: B - Bericht (mündlich), H - Handout, P - Präsentation, D - Dokument (digital), M (Multimedia)
 3) Ergebnis: A - Aktion, B - Beschluss, E - Empfehlung, I - Information, O - Offener Punkt, F - Erledigt (fertig)
 Abnahme des Protokolls erfolgt automatisch, wenn binnen 48 Stunden kein Einspruch erfolgt.

Agenda/Protokoll

Projekttitel: <Vertical Farming>
Auftraggeber: <Martin Meschik / Unternehmen>
Auftragnehmer: <Philipp Gasser / Unternehmen>
Schuljahr: <2018/19> Klasse: <5BM>

VERSION	DATUM	AUTORIN/AUTOR	ÄNDERUNG
v1.1	28.08.2018	<Philipp Gasser>	<Beschreibung der Änderung>

Thema/Ziel des Meetings:		Status des Projektes		
Datum:	28.08.2018	Dauer (Uhrzeit):	4h (13:00)	Ort: Stockerau
Teilnehmer ¹		Position / Funktion		Aufgabenbereich
Philipp Gasser		PL		Projektmanagement, Konstruktion
Stefan Stetina		PL Stv.		Konstruktion, Mechanik
Noah Gruber		PM		Software, Firmware
Ulrich Obetzauser		PM		Elektronik

Verteiler ¹
Siehe obige Anwesenheit

Tagesordnungspunkte						
#	Zeit	Dauer	Tagesordnungspunkt	Vor. ²	Erg ³	Ver.

01	13:00	30'	Einleitung und Aufnahme weitere TO's				
			Aufnahme weiterer Tagesordnungspunkte	-	-	PG	-
			Überprüfung des letzten Protokolls	-	-	PG	-

02	13:30	60'	Besprechung über vergangene TO's				
			Konzept 2: Hydroponik - <p_protokoll_20180826> Aktion 02: Robotunits/Bosch oder 4-Kant - <p_protokoll_20180826> Aktion 02: ansteuerbare Durchlasspumpe Link: https://www.conrad.at/de/niedervolt-durchlaufpumpe-barwig-02-1200-lh-12-v-537454.html?WT.mc_id=sea_9_Shopping&WT.srch=1&gclid=Cj0KCQjwn4ncBRCaARIsAFD5-gU8ZZyggNZyrX1sVb-8XheQWOkV8KjTwV4e_QuesefTbqFqcErWWFAaAgwEEALw_wcB&insert=U3	- B/D B	- O F	- SS PG	- 29.08 29.08

1) Personenangaben ohne Titel, 2) Vorbereitung: B - Bericht (mündlich), H - Handout, P - Präsentation, D - Dokument (digital), M (Multimedia)
3) Ergebnis: A - Aktion, B - Beschluss, E - Empfehlung, I - Information, O - Offener Punkt, F - Erledigt (fertig)

Abnahme des Protokolls erfolgt automatisch, wenn binnen 48 Stunden kein Einspruch erfolgt.

03	14:30	150'	Besprechung				
Projektmanagement:				A A	PG PG	03.09 03.09	
	- In Kontakt mit Herz As Media treten + Termin vor Ort						
Elektronik:				A A A	UO UO UO	03.09 03.09 03.09	
	- Sensoren recherchieren - Platinenlayout erstellen (Eagle) - Kompatibilität der ansteuerbaren Durchlasspumpe						
Informatik:				A A	NG NG	03.09 03.09	
	- Blockschaltbild (Händisch/Programm) - Flussdiagramm (Händisch/Programm)						

Aufgabenliste			
#	TO DO	Ver.	Deadline
01	Termin zum Besuch bei Herz As Media festlegen mit Hr. Kave Atefie	PG	03.09.2018
02	Stückliste für Robotunits/Bosch Profile (Profile inkl. Verbindungselemente)	SS	29.08.2018
03	Blockschaltbild erstellen (händisch oder mit geeignetem Programm)	NG	03.09.2018
04	Flussdiagramm (händisch oder mit geeignetem Programm)	NG	03.09.2018
05	Sensoren ermitteln bzw. recherchieren (Liste inkl. Datenblätter)	UO	03.09.2018
06	Platinenlayout erstellen (Eagle)	UO	03.09.2018
07	Kompatibilität einer geeigneten Durchlasspumpe	UO	03.09.2018

Dokumentenverzeichnis			
#	Titel	Dateiname	Ablageort
01	Protokoll des Meetings v. 1.0.	p_protokoll_20180826.pdf	https://htl3r.sharepoint.com/sites/Vertical-Farming2/Freigegebene%20Dokumente/Forms/AllItems.aspx?id=%2Fsites%2FVertical-Farming2%2FFreigegebene%20Dokumente%2F1_Projekmanagement%2F1%2E4_Protokolle%20%26%20Templates%2F1%2E4%2E1_Protokolle
02	Protokoll des Meetings v. 1.1.	p_protokoll_20180828.pdf	Siehe obiger Ablageort

Philipp Gasser

Stefan Stetina

Noah Gruber

Ulrich Obetshauser

1) Personenangaben ohne Titel, 2) Vorbereitung: B - Bericht (mündlich), H - Handout, P - Präsentation, D - Dokument (digital), M (Multimedia)
 3) Ergebnis: A - Aktion, B - Beschluss, E - Empfehlung, I - Information, O - Offener Punkt, F - Erledigt (fertig)
 Abnahme des Protokolls erfolgt automatisch, wenn binnen 48 Stunden kein Einspruch erfolgt.

Agenda/Protokoll

Projekttitel: <Vertical Farming>
 Auftraggeber: <Martin Meschik / Unternehmen>
 Auftragnehmer: <Philipp Gasser / Unternehmen>
 Schuljahr: <2018/19> Klasse: <5BM>

VERSION	DATUM	AUTORIN/AUTOR	ÄNDERUNG
v1.2	30.08.2013	<Philipp Gasser>	<Beschreibung der Änderung>

Thema/Ziel des Meetings:		Jour Fixe		
Datum:	25.09.2018	Dauer (Uhrzeit):	08:55	Ort: Labor MSK
Teilnehmer ¹		Position / Funktion		Aufgabenbereich
Martin Meschik		PHB		Elektronik, Informatik
Philipp Gasser		PL		Projektmanagement, Konstruktion
Ulrich Obetshauser		PM1		Elektronik

Verteiler¹

Siehe obige Anwesenheit

Tagesordnungspunkte						
#	Zeit	Dauer	Tagesordnungspunkt	Vor. ²	Erg ³	Ver.
01	08:55	30'	Besprechung			
Elektronik: Hauptplatine						
- Schaltplan ändern						
- Von +24V auf +12V:				B	A	UO
- Schaltregler (Verlustleistung ca. 0,5W) statt Spannungsregler (36 W)				B	I	UO
- +12V vernachlässigbar, wenn Pumpe mit PWM (25%, $P=U^2/R$)				B	I	UO
Bei dieser Lösung Sicherungspflicht!						-
- Übersichtlichkeit des Schaltplans				B	A	UO
- Eingänge links, Ausgänge rechts				B	A	UO
- Schaltregler: links nach rechts: +24V, +12V, +5V, +3,3V				B	A	UO
- Masse Symbole verwenden, unnötige Leitungen vermeiden				B	A	UO

1) Personenangaben ohne Titel, 2) Vorbereitung: B - Bericht (mündlich), H - Handout, P - Präsentation, D - Dokument (digital), M (Multimedia)

3) Ergebnis: A - Aktion, B - Beschluss, E - Empfehlung, I - Information, O - Offener Punkt, F - Erledigt (fertig)

Abnahme des Protokolls erfolgt automatisch, wenn binnen 48 Stunden kein Einspruch erfolgt.

Aufgabenliste			
#	TO DO	Ver.	Deadline
01	Schaltplan: Schaltregler oder PWM statt Spannungsregler	UO	02.09.2018
02	Schaltplan: Eingänge links, Ausgänge rechts	UO	02.09.2018
03	Schaltplan: Schalt-/Spannungsregler links nach rechts: +24V, +12V, +5V, +3,3V	UO	02.09.2018
04	Schaltplan: Masse Symbole verwenden, unnötige Leitungen vermeiden	UO	02.09.2018
05	Protokoll über Besprechung schreiben und auf SharePoint hochladen	PG	25.09.2018

Dokumentenverzeichnis

#	Titel	Dateiname	Ablageort
01	Protokoll des Meetings v. 1.2.	p_protokoll_20180925.pdf	https://htl3r.sharepoint.com/sites/Vertical-Farming2/Freigegebene%20Dokumente/Forms/AllItems.aspx?id=%2Fsites%2FVertical-Farming2%FFreigegebene%20Dokumente%2F1_Projekmanagement%2F1%2E4_Protokolle%20%26%20Templates%2F1%2E4%2E1_Protokolle

1) Personenangaben ohne Titel, 2) Vorbereitung: B - Bericht (mündlich), H - Handout, P - Präsentation, D - Dokument (digital), M (Multimedia)
 3) Ergebnis: A - Aktion, B - Beschluss, E - Empfehlung, I - Information, O - Offener Punkt, F - Erledigt (fertig)
 Abnahme des Protokolls erfolgt automatisch, wenn binnen 48 Stunden kein Einspruch erfolgt.

Agenda/Protokoll

Projekttitel: <Vertical Farming>
 Auftraggeber: <Martin Meschik / Unternehmen>
 Auftragnehmer: <Philipp Gasser / Unternehmen>
 Schuljahr: <2018/19> Klasse: <5BM>

VERSION	DATUM	AUTORIN/AUTOR	ÄNDERUNG
v1.3	30.08.2013	<Philipp Gasser>	<Beschreibung der Änderung>

Thema/Ziel des Meetings:		Jour Fixe		
Datum:	02.10.2018	Dauer (Uhrzeit):	10:30	Ort: Labor MSK, Labor CEK
Teilnehmer ¹		Position / Funktion		Aufgabenbereich
Martin Meschik		PHB		Elektronik, Informatik
Ewald Cekan		PNB		Projektmanagement
Philipp Gasser		PL		Projektmanagement, Konstruktion
Stefan Stetina		PL Stv.		Mechanik & Konstruktion
Noah Gruber		PM1		Programmierung
Ulrich Obetzauser		PM2		Elektronik

Verteiler ¹
Siehe obige Anwesenheit

Tagesordnungspunkte						
#	Zeit	Dauer	Tagesordnungspunkt	Vor. ²	Erg ³	Ver.
01	10:30	30'	Besprechung			
Elektronik: Hauptplatine						
- MSK:						
- Gesamtes System:						
- Schaltregler (Wirkungsgrad 95%) statt Spannungsregler (50%)				B	I	UO
- Besprechung des gesamten Stroms des Systems ($I = 4,33A$)				B	I	UO
- Temperaturfühler im Lötofen nicht an Aluminium				B	I	UO
Elektronik: „Beleuchtungsplatine“						
- CEK:						
- Berechnung ob Kühlung der LED's passt:						
- 1W pro 10cm ² (12 LED's in Verwendung, 83 LED's möglich)				B	A	UO

1) Personenangaben ohne Titel, 2) Vorbereitung: B - Bericht (mündlich), H - Handout, P - Präsentation, D - Dokument (digital), M (Multimedia)
 3) Ergebnis: A - Aktion, B - Beschluss, E - Empfehlung, I - Information, O - Offener Punkt, F - Erledigt (fertig)
 Abnahme des Protokolls erfolgt automatisch, wenn binnen 48 Stunden kein Einspruch erfolgt.

Aufgabenliste			
#	TO DO	Ver.	Deadline
01	„Beleuchtungsplatine“: Wärmeabfuhr der LED's auf ALU-Platte OK?	UO	12.10.2018
02	Protokoll über Besprechung schreiben und auf SharePoint hochladen	PG	12.10.2018
Dokumentenverzeichnis			
#	Titel	Dateiname	Ablageort
01	Protokoll des Meetings v. 1.3.	p_protokoll_20181002.pdf	https://htl3r.sharepoint.com/sites/Vertical-Farming2/Freigegebene%20Dokumente/Forms/AllItems.aspx?id=%2Fsites%2FVertical-Farming2%2FFreigegebene%20Dokumente%2F1_Projekmanagement%2F1%2E4_Protokolle%20%26%20Templates%2F1%2E4%2E1_Protolle

1) Personenangaben ohne Titel, 2) Vorbereitung: B - Bericht (mündlich), H - Handout, P - Präsentation, D - Dokument (digital), M (Multimedia)
 3) Ergebnis: A - Aktion, B - Beschluss, E - Empfehlung, I - Information, O - Offener Punkt, F - Erledigt (fertig)
Abnahme des Protokolls erfolgt automatisch, wenn binnen 48 Stunden kein Einspruch erfolgt.

Agenda/Protokoll

Projekttitel: <Vertical Farming>
 Auftraggeber: <Martin Meschik / Unternehmen>
 Auftragnehmer: <Philipp Gasser / Unternehmen>
 Schuljahr: <2018/19> Klasse: <5BM>

VERSION	DATUM	AUTORIN/AUTOR	ÄNDERUNG
v1.5	30.08.2013	<Philipp Gasser>	<Beschreibung der Änderung>

Thema/Ziel des Meetings:		Jour Fixe		
Datum:	16.10.2018	Dauer (Uhrzeit):	10:30	Ort: Labor MSK, Labor CEK
Teilnehmer ¹		Position / Funktion		Aufgabenbereich
Martin Meschik		PHB		Elektronik, Informatik
Ewald Cekan		PNB		Projektmanagement
Philipp Gasser		PL		Projektmanagement, Konstruktion
Stefan Stetina		PL Stv.		Mechanik & Konstruktion
Noah Gruber		PM1		Programmierung
Ulrich Obetzhauser		PM2		Elektronik

Verteiler ¹
Siehe obige Anwesenheit

Tagesordnungspunkte						
#	Zeit	Dauer	Tagesordnungspunkt	Vor. ²	Erg ³	Ver.
01	10:30	30'	Besprechung			
Elektronik: Hauptplatine				D	A	UO
- MSK:				D	A	UO
- Besprechen der ERC Fehler (Gegenmaßnahmen)				D	A	23.10.
- Besprechen der Fehler im Schaltplan				D	A	23.10.
- Konstantstromquelle einzelne Pins				D	A	23.10.
- Schaltregler einzelne Pins				D	A	23.10.
- Footprints mit korrekten Maßen erstellen				D	A	23.10.
Projektmanagement: Management Summary				D	F	PG
- CEK:						-
- Kein Handlungsbedarf, Summary alles OK						

1) Personenangaben ohne Titel, 2) Vorbereitung: B - Bericht (mündlich), H - Handout, P - Präsentation, D - Dokument (digital), M (Multimedia)
 3) Ergebnis: A - Aktion, B - Beschluss, E - Empfehlung, I - Information, O - Offener Punkt, F - Erledigt (fertig)

Abnahme des Protokolls erfolgt automatisch, wenn binnen 48 Stunden kein Einspruch erfolgt.

Aufgabenliste			
#	TO DO	Ver.	Deadline
01	Schaltplan/Platinenlayout: Konstantstromquelle einzelne Pins	UO	23.10.2018
02	Schaltplan/Platinenlayout: Schaltregler einzelne Pins	UO	23.10.2018
03	Schaltplan/Platinenlayout: Footprints mit korrekten Maßen erstellen	UO	23.10.2018
04	Protokoll über Besprechung schreiben und auf SharePoint hochladen	PG	23.10.2018

Dokumentenverzeichnis			
#	Titel	Dateiname	Ablageort
01	Protokoll des Meetings v. 1.5.	p_protokoll_20181016.pdf	https://htl3r.sharepoint.com/sites/Vertical-Farming2/Freigegebene%20Dokumente/Forms/AllItems.aspx?id=%2Fsites%2FVertical-Farming2%2FFreigegebene%20Dokumente%2F1_Projekmanagement%2F1%2E4_Protokolle%20%26%20Templates%2F1%2E4%2E1_Prokollo

1) Personenangaben ohne Titel, 2) Vorbereitung: B - Bericht (mündlich), H - Handout, P - Präsentation, D - Dokument (digital), M (Multimedia)
 3) Ergebnis: A - Aktion, B - Beschluss, E - Empfehlung, I - Information, O - Offener Punkt, F - Erledigt (fertig)
 Abnahme des Protokolls erfolgt automatisch, wenn binnen 48 Stunden kein Einspruch erfolgt.

Agenda/Protokoll

Projekttitle: <Vertical Farming>
Auftraggeber: <Martin Meschik / Unternehmen>
Auftragnehmer: <Philipp Gasser / Unternehmen>
Schuljahr: <2018/19> Klasse: <5BM>

VERSION	DATUM	AUTORIN/AUTOR	ÄNDERUNG
v1.6	30.08.2013	<Philipp Gasser>	<Beschreibung der Änderung>

Thema/Ziel des Meetings:		Jour Fixe			
Datum:	06.11.2018	Dauer (Uhrzeit):	30min (14:45)	Ort:	Raum 60
Teilnehmer ¹	Position / Funktion			Aufgabenbereich	
Martin Meschik	PHB			Elektronik, Informatik	
Ewald Cekan	PNB			Projektmanagement	
Philipp Gasser	PL			Projektmanagement, Konstruktion	
Stefan Stetina	PL Stv.			Mechanik & Konstruktion	
Noah Gruber	PM1			Programmierung	
Ulrich Obetzhauser	PM2			Elektronik	

Verteiler ¹
Siehe obige Anwesenheit

Tagesordnungspunkte						
#	Zeit	Dauer	Tagesordnungspunkt	Vor. ²	Erg ³	Ver.
01	10:30	30'	Besprechung			
Schaltplan inkl. Platinenlayout: - MSK: - Schaltplan wurde abgenommen - Blockschaltbild wurde abgenommen				D H	F F	UO NG
Plakat für TOFT - CEK + MSK: - Aufgabenstellung weg - Pro Überschrift jeweils 1 Satz - Bilder weg				D D D	A A A	PG PG PG

1) Personenangaben ohne Titel, 2) Vorbereitung: B - Bericht (mündlich), H - Handout, P - Präsentation, D - Dokument (digital), M (Multimedia)
3) Ergebnis: A - Aktion, B - Beschluss, E - Empfehlung, I - Information, O - Offener Punkt, F - Erledigt (fertig)

Abnahme des Protokolls erfolgt automatisch, wenn binnen 48 Stunden kein Einspruch erfolgt.

Aufgabenliste			
#	TO DO	Ver.	Deadline
01	Plakat für TOFT überarbeiten	PG	07.11.2018
02	Protokoll über Besprechung schreiben und auf SharePoint hochladen	PG	07.11.2018
Dokumentenverzeichnis			
#	Titel	Dateiname	Ablageort
01	Protokoll des Meetings v. 1.6.	p_protokoll_20181106.pdf	https://htl3r.sharepoint.com/sites/Vertical-Farming2/Freigegebene%20Dokumente/Forms/AllItems.aspx?id=%2Fsites%2FVertical-Farming2%FFreigegebene%20Dokumente%2F1_Projekmanagement%2F1%2E4_Protokolle%20%26%20Templates%2F1%2E4%2E1_Protokolle

1) Personenangaben ohne Titel, 2) Vorbereitung: B - Bericht (mündlich), H - Handout, P - Präsentation, D - Dokument (digital), M (Multimedia)
 3) Ergebnis: A - Aktion, B - Beschluss, E - Empfehlung, I - Information, O - Offener Punkt, F - Erledigt (fertig)
 Abnahme des Protokolls erfolgt automatisch, wenn binnen 48 Stunden kein Einspruch erfolgt.

Agenda/Protokoll

Projekttitel: <Vertical Farming>
 Auftraggeber: <Martin Meschik / Unternehmen>
 Auftragnehmer: <Philipp Gasser / Unternehmen>
 Schuljahr: <2018/19> Klasse: <5BM>

VERSION	DATUM	AUTORIN/AUTOR	ÄNDERUNG
v1.7	30.08.2013	<Philipp Gasser>	<Beschreibung der Änderung>

Thema/Ziel des Meetings:		Jour Fixe		
Datum:	04.12.2018	Dauer (Uhrzeit):	30min (15:10)	Ort: Labor MSK
Teilnehmer ¹		Position / Funktion		Aufgabenbereich
Martin Meschik		PHB		Elektronik, Informatik
Ewald Cekan		PNB		Projektmanagement
Philipp Gasser		PL		Projektmanagement, Konstruktion
Stefan Stetina		PL Stv.		Mechanik & Konstruktion
Noah Gruber		PM1		Programmierung

Verteiler ¹
Siehe obige Anwesenheit

Tagesordnungspunkte						
#	Zeit	Dauer	Tagesordnungspunkt	Vor. ²	Erg ³	Ver.

01	15:10	30'	Besprechung				
Elektronik - Platine:				M	A	UO	-
- MSK: - Mit Oszilloskop Verbindung zwischen HC-06 (BT) und PIC testen (PIC ausstecken und Oszilloskop direkt an Pin anhängen)							
Mechanik - Wasserbehälter + Drehmechanismus:				B	A	SS	-
- CEK + MSK: - Drehmechanismus fertigen zu lassen = OK - Kunststoff und passenden Kleber, ggf. Silikon suchen				B	A	SS	-
Projektmanagement - Berechnung Instandhaltungskosten:				D	A	PG	-
- CEK: - Leistung anpassen und Kosten von Substraten etc. miteinbinden							

1) Personennamen ohne Titel, 2) Vorbereitung: B - Bericht (mündlich), H - Handout, P - Präsentation, D - Dokument (digital), M (Multimedia)
 3) Ergebnis: A - Aktion, B - Beschluss, E - Empfehlung, I - Information, O - Offener Punkt, F - Erledigt (fertig)
 Abnahme des Protokolls erfolgt automatisch, wenn binnen 48 Stunden kein Einspruch erfolgt.

Aufgabenliste			
#	TO DO	Ver.	Deadline
01	Instandhaltungskosten: Leistung anpassen und Kosten von Substrat einbinden	PG	18.12.2018
02	Protokoll über Besprechung schreiben und auf SharePoint hochladen	PG	05.12.2018
03	Drehmechanismus: Mail mit CAD-Daten an Bilek + Schüll GmbH	PG	05.12.2018
04	Wasserbehälter: Kunststoff und passenden Kleber, ggf. Silikon suchen	SS	18.12.2018
05	Platine: Verbindung mit Oszilloskop zwischen Bluetooth-Modul und PIC testen	UO	18.12.2018

Dokumentenverzeichnis

#	Titel	Dateiname	Ablageort
01	Protokoll des Meetings v. 1.7.	p_protokoll_20181204.pdf	https://htl3r.sharepoint.com/sites/Vertical-Farming2/Freigegebene%20Dokumente/Forms/AllItems.aspx?id=%2Fsites%2FVertical-Farming2%FFreigegebene%20Dokumente%2F1_Projekmanagement%2F1%2E4_Protokolle%20%26%20Templates%2F1%2E4%2E1_Protokolle

1) Personenangaben ohne Titel, 2) Vorbereitung: B - Bericht (mündlich), H - Handout, P - Präsentation, D - Dokument (digital), M (Multimedia)
 3) Ergebnis: A - Aktion, B - Beschluss, E - Empfehlung, I - Information, O - Offener Punkt, F - Erledigt (fertig)
Abnahme des Protokolls erfolgt automatisch, wenn binnen 48 Stunden kein Einspruch erfolgt.

Agenda/Protokoll

Projekttitel: <Vertical Farming>
 Auftraggeber: <Martin Meschik / Unternehmen>
 Auftragnehmer: <Philipp Gasser / Unternehmen>
 Schuljahr: <2018/19> Klasse: <5BM>

VERSION	DATUM	AUTORIN/AUTOR	ÄNDERUNG
v1.8	30.08.2013	<Philipp Gasser>	<Beschreibung der Änderung>

Thema/Ziel des Meetings:		Jour Fixe			
Datum:		Dauer (Uhrzeit):		30min (15:30)	
Teilnehmer ¹		Position / Funktion			Aufgabenbereich
Martin Meschik		PHB			Elektronik, Informatik
Ewald Cekan		PNB			Projektmanagement
Philipp Gasser		PL			Projektmanagement, Konstruktion
Stefan Stetina		PL Stv.			Mechanik & Konstruktion
Noah Gruber		PM1			Programmierung
Ulrich Obetshauser		PM2			Elektronik

Verteiler ¹
Siehe obige Anwesenheit

Tagesordnungspunkte							
#	Zeit	Dauer	Tagesordnungspunkt	Vor. ²	Erg ³	Ver.	Deadline
01	15:30	30'	Besprechung				
Elektronik - Platine:				M	I	UO	-
- CEK: - Platine kontrolliert							
Mechanik - Wasserbehälter + Drehmechanismus:				M	A	SS	-
- CEK: - Wasserbehälter mit z.B.: Polokalrohr - Drehmechanismus überarbeiten a. Modell aus Karton od. Papier fertigen und testen b. Erst wenn es funktioniert -> Oberplatte neu fertigen				M	A	SS	-
				M	A	SS	-
				M	A	SS	-

1) Personenangaben ohne Titel, 2) Vorbereitung: B - Bericht (mündlich), H - Handout, P - Präsentation, D - Dokument (digital), M (Multimedia)
 3) Ergebnis: A - Aktion, B - Beschluss, E - Empfehlung, I - Information, O - Offener Punkt, F - Erledigt (fertig)

Abnahme des Protokolls erfolgt automatisch, wenn binnen 48 Stunden kein Einspruch erfolgt.

Vertical Farming | Projektmanagement

Projektmanagement - Management Summary: - CEK + MSK: - Next Time: „mangelhaft“ od. „unzureichendes Know-How“ weglassen - Angeführte Arbeiten müssen (nicht sollten) abgeschlossen werden	H H	F A	PG PG	-
Programmierung: - MSK: - Variablen als Global statt Lokal definieren	M	I	NG	-

Aufgabenliste			
#	TO DO	Ver.	Deadline
01	Protokoll über Besprechung schreiben und auf SharePoint hochladen	PG	20.12.2018
02	Drehmechanismus: Funktionsweise mithilfe von Karton-Modell testen	SS	18.12.2018
03	Wasserbehälter: Pumpe in Polokalrohr	SS	18.12.2018
04	Siehe Management Summary „DA_Management_Summary_18.12.“	PG UO SS NG	08.01.2018

Dokumentenverzeichnis			
#	Titel	Dateiname	Ablageort
01	Protokoll des Meetings v. 1.8.	p_protokoll_20181219.pdf	https://htl3r.sharepoint.com/sites/Vertical-Farming2/Freigegebene%20Dokumente/Forms/AllItems.aspx?id=%2Fsites%2FVertical-Farming2%2FFreigegebene%20Dokumente%2F1_Projekmanagement%2F1%2E4_Protokolle%20%26%20Templates%2F1%2E4%2E1_Protokolle

1) Personennamen ohne Titel, 2) Vorbereitung: B - Bericht (mündlich), H - Handout, P - Präsentation, D - Dokument (digital), M (Multimedia)
 3) Ergebnis: A - Aktion, B - Beschluss, E - Empfehlung, I - Information, O - Offener Punkt, F - Erledigt (fertig)
Abnahme des Protokolls erfolgt automatisch, wenn binnen 48 Stunden kein Einspruch erfolgt.

Agenda/Protokoll

Projekttitel: <Vertical Farming>
 Auftraggeber: <Martin Meschik / Unternehmen>
 Auftragnehmer: <Philipp Gasser / Unternehmen>
 Schuljahr: <2018/19> Klasse: <5BM>

VERSION	DATUM	AUTORIN/AUTOR	ÄNDERUNG
v1.9	30.08.2013	<Philipp Gasser>	<Beschreibung der Änderung>

Thema/Ziel des Meetings:		Jour Fixe			
Datum:	15.01.2019	Dauer (Uhrzeit):	15min (14:30)	Ort:	Raum [60]
Teilnehmer ¹	Position / Funktion			Aufgabenbereich	
Martin Meschik	PHB			Elektronik, Informatik	
Ewald Cekan	PNB			Projektmanagement	
Philipp Gasser	PL			Projektmanagement, Konstruktion	
Stefan Stetina	PL Stv.			Mechanik & Konstruktion	
Noah Gruber	PM1			Programmierung	
Ulrich Obetzhauser	PM2			Elektronik	

Verteiler ¹
Siehe obige Anwesenheit

Tagesordnungspunkte					
#	Zeit	Dauer	Tagesordnungspunkt	Vor. ²	Erg ³

01	14:30	15'	Besprechung				
			Drehmechanismus - V2.0 Rotationsplatte:				
- CEK + MSK:			- Kartonmodell ok -> bei Kooperationspartner fertigen lassen	M	A	SS	-
- MSK:			- Creo Illustrate Video von Drehmechanismus bei V1.0 belassen	M	A	PG	-

1) Personenangaben ohne Titel, 2) Vorbereitung: B - Bericht (mündlich), H - Handout, P - Präsentation, D - Dokument (digital), M (Multimedia)
 3) Ergebnis: A - Aktion, B - Beschluss, E - Empfehlung, I - Information, O - Offener Punkt, F - Erledigt (fertig)
 Abnahme des Protokolls erfolgt automatisch, wenn binnen 48 Stunden kein Einspruch erfolgt.

Aufgabenliste			
#	TO DO	Ver.	Deadline
01	Protokoll über Besprechung schreiben und auf SharePoint hochladen	PG	16.01.2019
02	Drehmechanismus: *.dwg-Datei an Bilek + Schüll	PG	22.01.2019
Dokumentenverzeichnis			
#	Titel	Dateiname	Ablageort
01	Protokoll des Meetings v. 1.9.	p_protokoll_20190115.pdf	https://htl3r.sharepoint.com/sites/Vertical-Farming2/Freigegebene%20Dokumente/Forms/AllItems.aspx?id=%2Fsites%2FVertical-Farming2%2FFreigegebene%20Dokumente%2F1_Projekmanagement%2F1%2E4_Protokolle%20%26%20Templates%2F1%2E4%2E1_Protokolle

1) Personenangaben ohne Titel, 2) Vorbereitung: B - Bericht (mündlich), H - Handout, P - Präsentation, D - Dokument (digital), M (Multimedia)
 3) Ergebnis: A - Aktion, B - Beschluss, E - Empfehlung, I - Information, O - Offener Punkt, F - Erledigt (fertig)
 Abnahme des Protokolls erfolgt automatisch, wenn binnen 48 Stunden kein Einspruch erfolgt.

Agenda/Protokoll

Projekttitel: <Vertical Farming>
 Auftraggeber: <Martin Meschik / Unternehmen>
 Auftragnehmer: <Philipp Gasser / Unternehmen>
 Schuljahr: <2018/19> Klasse: <5BM>

VERSION	DATUM	AUTORIN/AUTOR	ÄNDERUNG
v1.10	30.08.2013	<Philipp Gasser>	<Beschreibung der Änderung>

Thema/Ziel des Meetings:		Jour Fixe			
Datum:	22.01.2019	Dauer (Uhrzeit):		15min (14:55)	Ort: Raum [60]
Teilnehmer ¹		Position / Funktion			Aufgabenbereich
Martin Meschik		PHB			Elektronik, Informatik
Ewald Cekan		PNB			Projektmanagement
Philipp Gasser		PL			Projektmanagement, Konstruktion
Stefan Stetina		PL Stv.			Mechanik & Konstruktion
Noah Gruber		PM1			Programmierung
Ulrich Obetzhauser		PM2			Elektronik

Verteiler¹

Siehe obige Anwesenheit

Tagesordnungspunkte

#	Zeit	Dauer	Tagesordnungspunkt	Vor. ²	Erg ³	Ver.	Deadline
01	14:55	15'	Besprechung				
Elektronik - Platine:				M	A	UO	-
- MSK: - Spannungsteiler im Schaltplan vergessen							
Drehmechanismus - V2.0 Rotationsplatte:				P	F	SS	-
- CEK + MSK: - Gefertigte V2.0 Rotationsplatte funktioniert -> OK							

1) Personenangaben ohne Titel, 2) Vorbereitung; B - Bericht (mündlich), H - Handout, P - Präsentation, D - Dokument (digital), M (Multimedia)
 3) Ergebnis: A - Aktion, B - Beschluss, E - Empfehlung, I - Information, O - Offener Punkt, F - Erledigt (fertig)

Abnahme des Protokolls erfolgt automatisch, wenn binnen 48 Stunden kein Einspruch erfolgt.

Vertical Farming | Projektmanagement

Projektmanagement - Vorbereitung TOFT:				
- CEK:	B B	A A	PG PG	- -
- Holzgeflecht von Kooperationspartner „Herbeus Greens“ abholen				
- Säzlinge statt bei Sponsor bei „Dehner“ kaufen				

Aufgabenliste			
#	TO DO	Ver.	Deadline
01	Protokoll über Besprechung schreiben und auf SharePoint hochladen	PG	23.01.2019
02	Vorbereitung TOFT: „Herbeus Greens“ kontaktieren und Holzgeflecht abholen	PG	23.01.2019
03	Vorbereitung TOFT: Säzlinge bei „Dehner“ kaufen	PG	25.01.2019
04	Elektronik: Spannungsteiler auf Platine hinzufügen	UO	29.01.2019

Dokumentenverzeichnis

#	Titel	Dateiname	Ablageort
01	Protokoll des Meetings v. 1.10	p_protokoll_20190122.pdf	https://htl3r.sharepoint.com/sites/Vertical-Farming2/Freigegebene%20Dokumente/Forms/AllItems.aspx?id=%2Fsites%2FVertical-Farming2%2FFreigegebene%20Dokumente%2F1_Projekmanagement%2F1%2E4_Protokolle%20%26%20Templates%2F1%2E4%2E1_Protokolle

1) Personenangaben ohne Titel, 2) Vorbereitung: B - Bericht (mündlich), H - Handout, P - Präsentation, D - Dokument (digital), M (Multimedia)
 3) Ergebnis: A - Aktion, B - Beschluss, E - Empfehlung, I - Information, O - Offener Punkt, F - Erledigt (fertig)
 Abnahme des Protokolls erfolgt automatisch, wenn binnen 48 Stunden kein Einspruch erfolgt.

Agenda/Protokoll

Projekttitle: <Vertical Farming>
Auftraggeber: <Martin Meschik / Unternehmen>
Auftragnehmer: <Philipp Gasser / Unternehmen>
Schuljahr: <2018/19> Klasse: <5BM>

VERSION	DATUM	AUTORIN/AUTOR	ÄNDERUNG
v1.11	30.08.2013	<Philipp Gasser>	<Beschreibung der Änderung>

Thema/Ziel des Meetings:		Jour Fixe		
Datum:	12.02.2019	Dauer (Uhrzeit):	30min (14:45)	Ort: Labor Mikroprozessor
Teilnehmer ¹		Position / Funktion		Aufgabenbereich
Martin Meschik		PHB		Elektronik, Informatik
Ewald Cekan		PNB		Projektmanagement
Philipp Gasser		PL		Projektmanagement, Konstruktion
Stefan Stetina		PL Stv.		Mechanik & Konstruktion
Noah Gruber		PM1		Programmierung
Ulrich Obetzhauser		PM2		Elektronik

Verteiler ¹
Siehe obige Anwesenheit

Tagesordnungspunkte						
#	Zeit	Dauer	Tagesordnungspunkt	Vor. ²	Erg ³	Ver.

01	14:45	30'	Besprechung				
Diplomarbeitsbuch:				B	B	ALL	-
- CEK: - 20 Seiten Text, ohne Bilder - Erste Version bis 01.03.2019 - Unterpunkte nur bei jeweils 2 Überschriften (1 = überflüssig!)				B	B	ALL	-
Programmierung:				B	B	ALL	-
- MSK: - DHT22 Sensor: Auslesen der Werte (EUSART-Register)				D	A	NG	-

1) Personenangaben ohne Titel, 2) Vorbereitung: B - Bericht (mündlich), H - Handout, P - Präsentation, D - Dokument (digital), M (Multimedia)
3) Ergebnis: A - Aktion, B - Beschluss, E - Empfehlung, I - Information, O - Offener Punkt, F - Erledigt (fertig)

Abnahme des Protokolls erfolgt automatisch, wenn binnen 48 Stunden kein Einspruch erfolgt.

Vertical Farming | Projektmanagement

Projektmanagement - Diplomarbeitsbuch:				
- CEK:				
- Bei Projektmanagement Bilder erlaubt (Arbeit berücksichtigt)	D	B	PG	-
- Aufbau mit Theorie inkl. Beispiel = OK	D	F	PG	-
- Meilensteinplan: Erklären warum Verzug (wenn vorhanden)	B	B	PG	-
- Risikoanalyse: Grafik auch quadratisch gestalten	D	A	PG	-
- Wasserfall-Methode: Dunkelgrün auf andere Farbe ändern	D	A	PG	-

Aufgabenliste			
#	TO DO	Ver.	Deadline
01	Protokoll über Besprechung schreiben und auf SharePoint hochladen	PG	13.02.2019
02	Programmcode: EUSART-Register überarbeiten	NG	20.02.2019
03	Erste Version des Diplomarbeitsbuches schreiben	ALL	01.03.2019

Dokumentenverzeichnis			
#	Titel	Dateiname	Ablageort
01	Protokoll des Meetings v. 1.11	p_protokoll_20190212.pdf	https://htl3r.sharepoint.com/sites/Vertical-Farming2/Freigegebene%20Dokumente/Forms/AllItems.aspx?id=%2Fsites%2FVertical-Farming2%2FFreigegebene%20Dokumente%2F1_Projekmanagement%2F1%2E4_Protokolle%20%26%20Templates%2F1%2E4%2E1_Protokolle

1) Personenangaben ohne Titel, 2) Vorbereitung: B - Bericht (mündlich), H - Handout, P - Präsentation, D - Dokument (digital), M (Multimedia)
 3) Ergebnis: A - Aktion, B - Beschluss, E - Empfehlung, I - Information, O - Offener Punkt, F - Erledigt (fertig)
Abnahme des Protokolls erfolgt automatisch, wenn binnen 48 Stunden kein Einspruch erfolgt.

Agenda/Protokoll

Projekttitel: <Vertical Farming>
 Auftraggeber: <Martin Meschik / Unternehmen>
 Auftragnehmer: <Philipp Gasser / Unternehmen>
 Schuljahr: <2018/19> Klasse: <5BM>

VERSION	DATUM	AUTORIN/AUTOR	ÄNDERUNG
v1.12	30.08.2013	<Philipp Gasser>	<Beschreibung der Änderung>

Thema/Ziel des Meetings:		Jour Fixe			
Datum:	05.03.2019	Dauer (Uhrzeit):		15min (13:45)	Ort: Labor Mikroprozessor
Teilnehmer ¹		Position / Funktion			Aufgabenbereich
Martin Meschik		PHB			Elektronik, Informatik
Ewald Cekan		PNB			Projektmanagement
Philipp Gasser		PL			Projektmanagement, Konstruktion
Stefan Stetina		PL Stv.			Mechanik & Konstruktion
Noah Gruber		PM1			Programmierung
Ulrich Obetzhauser		PM2			Elektronik

Verteiler¹

Siehe obige Anwesenheit

Tagesordnungspunkte

#	Zeit	Dauer	Tagesordnungspunkt	Vor. ²	Erg ³	Ver.	Deadline
01	13:45	15'	Besprechung				
			Diplomarbeitsbuch: - CEK: - Struktur OK	B	F	PG	-
			Mechanik: - CEK: - Aufpassen bei Wörtern: Konstruieren und fertigen - Wenn System funktioniert => Fertigung und Konstruktion = OK	B D	B A	PGSS PGSS	-

1) Personenangaben ohne Titel, 2) Vorbereitung: B - Bericht (mündlich), H - Handout, P - Präsentation, D - Dokument (digital), M (Multimedia)
 3) Ergebnis: A - Aktion, B - Beschluss, E - Empfehlung, I - Information, O - Offener Punkt, F - Erledigt (fertig)

Abnahme des Protokolls erfolgt automatisch, wenn binnen 48 Stunden kein Einspruch erfolgt.

Aufgabenliste			
#	TO DO	Ver.	Deadline
01	Protokoll über Besprechung schreiben und auf SharePoint hochladen	PG	06.03.2019
Dokumentenverzeichnis			
#	Titel	Dateiname	Ablageort
01	Protokoll des Meetings v. 1.12	p_protokoll_20190305.pdf	https://htl3r.sharepoint.com/sites/Vertical-Farming2/Freigegebene%20Dokumente/Forms/AllItems.aspx?id=%2Fsites%2FVertical-Farming2%2FFreigegebene%20Dokumente%2F1_Projekmanagement%2F1%2E4_Protokolle%20%26%20Templates%2F1%2E4%2E1_Protokolle

1) Personangaben ohne Titel, 2) Vorbereitung: B - Bericht (mündlich), H - Handout, P - Präsentation, D - Dokument (digital), M (Multimedia)
 3) Ergebnis: A - Aktion, B - Beschluss, E - Empfehlung, I - Information, O - Offener Punkt, F - Erledigt (fertig)
 Abnahme des Protokolls erfolgt automatisch, wenn binnen 48 Stunden kein Einspruch erfolgt.

Agenda/Protokoll

Projekttitle: <Vertical Farming>
Auftraggeber: <Martin Meschik / Unternehmen>
Auftragnehmer: <Philipp Gasser / Unternehmen>
Schuljahr: <2018/19> Klasse: <5BM>

VERSION	DATUM	AUTORIN/AUTOR	ÄNDERUNG
v1.13	30.08.2013	<Philipp Gasser>	<Beschreibung der Änderung>

Thema/Ziel des Meetings:		Jour Fixe		
Datum:	12.03.2019	Dauer (Uhrzeit):	15min (13:55)	Ort: Labor Mikroprozessor
Teilnehmer ¹		Position / Funktion		Aufgabenbereich
Martin Meschik		PHB		Elektronik, Informatik
Ewald Cekan		PNB		Projektmanagement
Philipp Gasser		PL		Projektmanagement, Konstruktion
Stefan Stetina		PL Stv.		Mechanik & Konstruktion
Noah Gruber		PM1		Programmierung
Ulrich Obetshauser		PM2		Elektronik

Verteiler ¹
Siehe obige Anwesenheit

Tagesordnungspunkte							
#	Zeit	Dauer	Tagesordnungspunkt	Vor. ²	Erg ³	Ver.	Deadline
01	13:55	15'	Besprechung				
Werbefideo: - CEK + MSK: - Passend gewählt; kurz und bündig				M	F	PG	-
Temperaturverhalten in Anpflanzbehälter: - CEK + MSK: - Messwerte in äquidistanten Zeitabständen nehmen - Lüfter in bestimmten Winkel montieren für besseren Luftzyklus				B	B	UO	-
				B	B	UO	-

1) Personenangaben ohne Titel, 2) Vorbereitung: B - Bericht (mündlich), H - Handout, P - Präsentation, D - Dokument (digital), M (Multimedia)
3) Ergebnis: A - Aktion, B - Beschluss, E - Empfehlung, I - Information, O - Offener Punkt, F - Erledigt (fertig)

Abnahme des Protokolls erfolgt automatisch, wenn binnen 48 Stunden kein Einspruch erfolgt.

Aufgabenliste			
#	TO DO	Ver.	Deadline
01	Protokoll über Besprechung schreiben und auf SharePoint hochladen	PG	13.03.2019
02	Messwerte nehmen und Temperaturanstieg analysieren + interpretieren	UO	19.03.2019
03	Lüfter über LEDs mit Winkel einbauen, um besseren Luftzyklus zu erhalten	UO	19.03.2019

Dokumentenverzeichnis			
#	Titel	Dateiname	Ablageort
01	Protokoll des Meetings v. 1.13	p_protokoll_20190312.pdf	https://htl3r.sharepoint.com/sites/Vertical-Farming2/Freigegebene%20Dokumente/Forms/AllItems.aspx?id=%2Fsites%2FVertical-Farming2%2FFreigegebene%20Dokumente%2F1_Projekmanagement%2F1%2E4_Protokolle%20%26%20Templates%2F1%2E4%2E1_Protokolle

1) Personangaben ohne Titel, 2) Vorbereitung: B - Bericht (mündlich), H - Handout, P - Präsentation, D - Dokument (digital), M (Multimedia)
 3) Ergebnis: A - Aktion, B - Beschluss, E - Empfehlung, I - Information, O - Offener Punkt, F - Erledigt (fertig)
 Abnahme des Protokolls erfolgt automatisch, wenn binnen 48 Stunden kein Einspruch erfolgt.



Projekte und Projektmanagement

Höhere Technische Bundeslehranstalt Wien 3, Rennweg 89b, A -1030 Wien

Projekttitel:	VERTICAL FARMING		
Management Summary		Version:	0.2
Schuljahr:	2018/19	Klasse:	5BM
AuftraggeberIn:	Martin MESCHIK	ProjektleiterIn:	Philipp GASSER

Version	Datum	AutorIn	Änderungen
0.2	18.12.2018	Philipp GASSER	Erstellen der Vorlage

Projektstatus

	Projektmanagement: Derzeit dürfen wir uns über ein Budget von insgesamt 1400,- freuen. Der Großteil der Materialen sind bereits geliefert worden. Nach Absprache mit Herrn Obetzhauser bestellen wird das Netzteil erst, wenn die Platine voll funktionstüchtig ist.
	Mechanik: Die Konstruktion ist größtenteils fertig gestellt. Das Wasserreservoir muss noch gefertigt bzw. die Pumpe + diverse Kabel müssen noch angebracht werden.
	Elektronik: Die Platine ist bereits geätzt und bestückt worden. Die Status-Led wurde zudem erfolgreich angesteuert. Das Testen der Verbindung von Bluetooth-Modul zu PIC erfolgt sehr baldig.
	Programmierung: Das Blockschaltbild wurde bereits erstellt. Die Android-App ist zudem auch schon fertig. Alleinig die Programmierung des Micro-Controller ist derzeit, durch nicht ausreichendes Know-How, mangelhaft.

Handlungsbedarf seitens der Betreuer

Probleme im Projekt

Momentan müssen wir das Wasserreservoir fertigen, die Anpflanzwanne dem Schlauch der Pumpe anpassen, die Platine testen und den Programmcode schreiben bzw. überarbeiten. Ich, als PL, sehe im Moment einen Handlungsbedarf. Trotz einigen Puffern kam es zu einem „Nichteinhalten“ der Meilensteine.

Next Steps

Liste aller künftigen Schritte, um Projekt-Verzug zu minimieren

#	Milestone	Deadline	Ver.
1	Gesamte Einheit zusammengebaut	08.01.2019	STE
2	Gesamte Einheit verkabelt	08.01.2019	OBE
3	Programmode erstellt	08.01.2019	GRU
4	Umsetzungsphase abgeschlossen	08.01.2019	ALLE
5	Testphase abgeschlossen	18.01.2019	ALLE

FOLGENDE DINGE SIND IN DEN FERIEN ZU ERLEDIGEN:

1. **Gesamte Einheit zusammengebaut** [Name + Deadline: siehe oben!]
 - a. *Anpflanzbecken an Durchmesser des Schlauches anpassen*
 - Anpflanzbecken mit Bohrungen versehen
 - ggf. PG-Verschraubungen kaufen montieren
 - b. *Dreh-Mechanismus bearbeiten*
 - Feilen der Kanten, wo Samenbehälter hängen bleiben.
2. **Gesamte Einheit verkabelt** [Name + Deadline: siehe oben!]
 - a. *Komponenten mit Platine verkabeln*
 - High Power Leds mit Platine verbinden
 - ggf. Kabelmanagement
 - b. *Netzteil testen*
 - Netzteil bestellen und mit gesamtem System testen
3. **Programmcode erstellt** [Name + Deadline: siehe oben!]
 - a. *MPLAB: Programmcode zur Ansteuerung des Systems*
 - Beleuchtung mit PWM über App steuern
 - Bewässerung (Pumpe) mit PWM über App steuern
 - Temperatur-/Feuchtigkeitssensor via Android-App ausgeben
 - Füllstand mittels 0/1 Zustand via Android-App ausgeben
 - b. *MPLAB: Status LEDS?*
 - Besprechen mit Betreuern (MUSS-Ziel steht in Antrag)
 - c. *MPLAB: Vollautomatische Bewässerung?*
 - Wenn ausreichend Zeit vorhanden! (Optionales-Ziel steht in Antrag)

Milestones

Liste aller Meilensteine des Projekts zur Nachverfolgung

#	Milestone	Termine		Ver.	Abnahme
		SOLL	IST		
1	Projekt grob geplant	14.09	14.09	GAS	✓
2	Projektplanung abgeschlossen	21.09	19.09	GAS	✓
3	Gesamte Einheit konstruiert	28.09	06.11	GAS	(✓)
4	Platinenlayouts erstellt	28.09	06.11	OBE	✓
5	Blockschaltbild	28.09	06.11	GRU	✓
6	Tech. Planung abgeschlossen	28.09	06.11	ALLE	✓
7	Zwischenpräsentation abgehalten	04.12	04.12	ALLE	✓
8	Gesamte Einheit zusammengebaut	23.11		STE	
9	Gesamte Einheit verkabelt	07.12		OBE	
10	Programmcode erstellt	07.12		GRU	
11	Android-App erstellt	07.12	04.12	GRU	✓
12	Umsetzungsphase abgeschlossen	07.12		ALLE	
13	Testphase abgeschlossen	18.01		ALLE	
14	Abschlusspräsentation abgehalten	01.03		ALLE	
15	Projekt beendet/abgenommen	01.04		ALLE	



Projekte und Projektmanagement

Höhere Technische Bundeslehranstalt Wien 3, Rennweg 89b, A -1030 Wien

Projekttitle: VERTICAL FARMING			
Management Summary		Version:	0.1
Schuljahr:	2018/19	Klasse:	5BM
AuftraggeberIn:	Martin MESCHIK	ProjektleiterIn:	Philipp GASSER

Version	Datum	AutorIn	Änderungen
0.1	23.10.2018	Philipp GASSER	Erstellen der Vorlage

Projektstatus

	Projektmanagement: Ein Großteil der Materialien im Bereich der Mechanik und Elektronik wurden bereits zugestellt. Die Website ist Up-To-Date und Kooperationsvereinbarungen bzw. Honorarnoten wurden an die dazugehörigen Unternehmen gesendet. Derzeit wurden uns neben Rabatten und Materialien, auch finanzielle Mittel i.d. Höhe von €1300,-- brutto bestätigt. Einige Materialien bzw. €300,-- brutto haben wir erhalten.
	Mechanik: Das Grundgerüst steht. Die Kanten des Anpflanzbecken werden gerade verschweißt. Die Konstruktionszeichnungen für den Deckel des Anpflanzbeckens inkl. Drehmechanismus sind derzeit noch fehlerhaft. Das Modellieren der Konstruktion ist aufgrund von spontanen Änderungen derzeit nicht mehr aktuell.
	Elektronik: Hier ändern wir gerade die letzten Bauteile im Schaltplan. Beim Platinenlayout fehlen die Layer, welche zur Verbindung der Bauteile benötigt werden.
	Programmierung: Das Ansteuern einer Led mithilfe eines Potentiometers über PWM hat funktioniert. Zudem ist das Layout der Android-App fertig. Das Blockschaltbild und das Flussdiagramm werden derzeit überarbeitet.

Handlungsbedarf seitens der Betreuer

Kein Handlungsbedarf.

Probleme im Projekt

Momentan müssen wir das Platinenlayout inkl. Schaltplan, das dazugehörige Blockschaltbild und das dazugehörige Flussdiagramm überarbeiten. Ich, als PL, sehe im Moment keinen Handlungsbedarf. Da ein Puffer eingeplant wurde, hat diese Verzögerung der Meilensteine keine grobe Auswirkung auf das Projekt.

Milestones

Liste aller Meilensteine des Projekts zur Nachverfolgung

#	Milestone	Termine		Ver.	Abnahme
		SOLL	IST		
1	Projekt grob geplant	14.09	14.09	GAS	✓
2	Projektplanung abgeschlossen	21.09	19.09	GAS	✓
3	Gesamte Einheit konstruiert	28.09	-	GAS	
4	Platinenlayouts erstellt	28.09	-	OBE	
5	Blockschaltbild	28.09	-	GRU	
6	Tech. Planung abgeschlossen	28.09	-	ALLE	



Mechatronik

Vertical Farming

Vertikales Anbauen von Obst & Gemüse



Platzsparend und erweiterbar!

Vertical Farming bringt die Natur und die Nachhaltigkeit in die eigenen 4-Wände, wodurch es jeder Person möglich ist, einen individuellen Obst- oder Gemüsegarten anzulegen.

Allseits einsatzbereit!

Da das System auch an sonnenarmen oder dunklen Orten eingesetzt werden soll, sind Leuchtdioden über den jeweiligen Anbauflächen montiert, welche die Photosynthese der Pflanzen unterstützen.

Einfache Kontrolle per App!

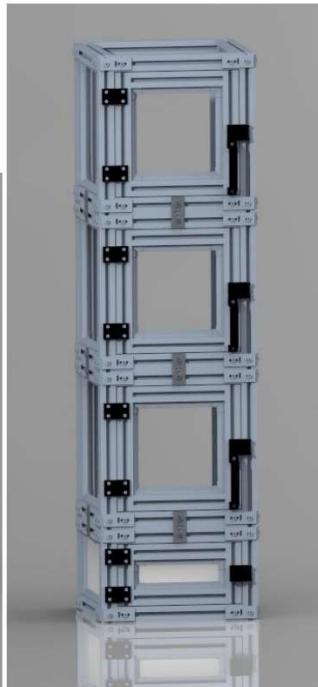
In unserer Android-App hat man Einsicht auf die Messwerte des Systems und kann die Bewässerung als auch die Beleuchtung steuern.

Betreuung:

- > Dr. Ewald Cekan
- > DI Martin Meschik
- > Mark Sauerwein

Weitere Informationen finden Sie unter:
www.vertical-farming.at

INFO	
Schuljahr	2018/19
Projektleitung	Philipp Gasser
Rolle Stv. PL	Stefan Stetina
Rolle PMA	Noah Gruber
Rolle PMA	Ulrich Obetzhauser
Gesamtstunden	600 Stunden



Abnahmeprotokoll

Projekttitle: <Vertical Farming>
Auftraggeber: <Vorname Nachname / Unternehmen>
Auftragnehmer: <Philipp Gasser / Unternehmen>
Schuljahr: <2018/19> Klasse: <5BM>

VERSION	DATUM	AUTORIN/AUTOR	ÄNDERUNG
V1.0	26.8.2018	Vorname Nachname	<Beschreibung der Änderung z.B. Erstellung ...>

- 1 Grundlage der Abnahme 2
- 2 Abnahmekriterien..... 2

1 Grundlage der Abnahme

Die gegenständliche Abnahme basiert auf dem Diplomarbeitsantrag vom 28.09.2018, Dokumentenname „da_antrag_gasser“ und der folgenden Change Requests: *Keine vorhanden!*

2 Abnahmekriterien

Muss-Ziele

Ziel	Zielerreichung
Ziel a) Projektplanung und Projektcontrolling Der Erfolg des Projekts oder Produkts hängt auch von der Planung und dem Controlling ab. Als Planungsmethode für das OSP und PSP wird das Programm „NCH Software“ eingesetzt. Mithilfe der Online-Cloud „Microsoft: SharePoint“ wird das Projekt sowohl koordiniert als auch dokumentiert. Zudem finden Meetings statt, um den IST-Zustand mit dem SOLL-Zustand zu vergleichen und um gegebenenfalls Gegenmaßnahmen zu unternehmen.	<input checked="" type="checkbox"/> erfüllt <input type="checkbox"/> teilweise erfüllt <input type="checkbox"/> nicht erfüllt
Anmerkung:	
Ziel b) Repräsentation des Projekts nach außen Das Projekt soll mithilfe einer Internetseite vertreten werden. Diese ist nötig, um auf Rückmeldungen und Verbesserungsvorschläge von Interessenten reagieren zu können. Weiters kann diese von möglichen Sponsoren besichtigt werden, um benötigte Informationen abrufen zu können.	<input checked="" type="checkbox"/> erfüllt <input type="checkbox"/> teilweise erfüllt <input type="checkbox"/> nicht erfüllt
Anmerkung:	
Ziel c) Gesamte Konstruktion modellieren inkl. Thingmark Um die Konstruktion Interessenten als Simulation zeigen zu können, muss die Einheit (Grundgerüst + Wasserreservoirs + Anpflanzbehälter) mithilfe des Programmes „CREO Parametric 4.0“ modelliert werden. Weiters ist es nötig ein Thingmark zu erstellen, mit welchen man sich das Vertical Farming System mit Hilfe der App „Vuforia View“ an jedem Ort visuell darstellen lassen kann.	<input checked="" type="checkbox"/> erfüllt <input type="checkbox"/> teilweise erfüllt <input type="checkbox"/> nicht erfüllt
Anmerkung:	
Ziel d) Sponsoring Um unser Projekt einerseits leichter finanzieren und andererseits hochwertige Komponenten verwenden zu können, ist es nötig Sponsoren zu suchen. Unser Ziel ist es, dass uns Sponsoren mindestens 25% der voraussichtlichen Kosten übernehmen.	<input checked="" type="checkbox"/> erfüllt <input type="checkbox"/> teilweise erfüllt <input type="checkbox"/> nicht erfüllt
Anmerkung:	

Ziel	Zielerreichung
Ziel e) Konstruktion fertigen Die Konstruktion besteht aus 4 Kant Profilen (Robotunits), welche das Fundament für die gesamte Konstruktion ist. Das Gerüst erhält zusätzlich ein aus welchem bei Bedarf, Wasser zur Anpflanzeinheit gepumpt wird. Die Pumpe, die Platinen, das Wasserreservoir sowie die gesamte Elektronik sollen im Basisbehälter Platz finden. Anmerkung:	<input checked="" type="checkbox"/> erfüllt <input type="checkbox"/> teilweise erfüllt <input type="checkbox"/> nicht erfüllt
Ziel f) 1 Anpflanzeinheit fertigen Die Anpflanzeinheit besteht aus einem Wasserreservoir (Edelstahl), aus einem Rahmen (Aluminium) und der LED Beleuchtung für die Pflanzen. Das Wasserreservoir erhält einen Deckel mit integriertem Drehmechanismus aus Kunststoff. Weiters werden kleine Anpflanzbehälter gefertigt, die in den Deckel des Wasserreservoirs integriert werden. Auf die Oberseite der gesamten Einheit kommt weiters eine Abdeckplatte, welche mit Lüftungsschlitzten versehen ist. Anmerkung:	<input checked="" type="checkbox"/> erfüllt <input type="checkbox"/> teilweise erfüllt <input type="checkbox"/> nicht erfüllt
Ziel g) Beleuchtungsplatine montieren Die Beleuchtungsplatine soll über der Anpflanzfläche Platz finden. Anmerkung:	<input checked="" type="checkbox"/> erfüllt <input type="checkbox"/> teilweise erfüllt <input type="checkbox"/> nicht erfüllt
Ziel h) Status LEDs im Programmcode festlegen Die Farbe der LED's symbolisiert einen spezifischen Zustand des Systems (Die grüne LED weist darauf hin, dass das Wasserreservoir ausreichend gefüllt ist. Die gelbe LED weist darauf hin, dass der Wasserspiegel unter dem grünen Bereich liegt. Die rote LED weist darauf hin, dass der Wasserbehälter vollkommen leer ist.) Anmerkung:	<input checked="" type="checkbox"/> erfüllt <input type="checkbox"/> teilweise erfüllt <input type="checkbox"/> nicht erfüllt
Ziel i) Bewässerung und Beleuchtung mit Hilfe der Android App steuern Die Bewässerung und die Beleuchtung werden über eine Bluetooth-App angesteuert. Anmerkung:	<input checked="" type="checkbox"/> erfüllt <input type="checkbox"/> teilweise erfüllt <input type="checkbox"/> nicht erfüllt
Ziel j) Implementieren von Feuchtigkeits-, Temperatur- und Füllstandsensor im Programmcode Die Feuchtigkeits- und Temperatursensoren messen die Feuchtigkeit und die Temperatur in einer Anpflanzeinheit. Der Füllstandsensor misst den Wasserspiegel im Wasserreservoir. Anmerkung:	<input checked="" type="checkbox"/> erfüllt <input type="checkbox"/> teilweise erfüllt <input type="checkbox"/> nicht erfüllt

Ziel	Zielerreichung
Ziel k) Ansteuerung des PIC über Bluetooth-Schnittstelle Die Bluetooth-Schnittstelle am Mikroprozessor dient dazu, sich via Bluetooth mit der Vertical-Farming-Einheit zu verbinden und diese auch zu steuern.	<input checked="" type="checkbox"/> erfüllt <input type="checkbox"/> teilweise erfüllt <input type="checkbox"/> nicht erfüllt
Anmerkung:	
Ziel l) Android App programmieren Mit Hilfe der Android-App werden Feuchtigkeit, Temperatur und Füllstand in einer Anpflanzeinheit ausgelesen. Außerdem werden Bewässerung und Beleuchtung ferngesteuert.	<input checked="" type="checkbox"/> erfüllt <input type="checkbox"/> teilweise erfüllt <input type="checkbox"/> nicht erfüllt
Anmerkung:	
Ziel m) Erstellen der Platinenlayouts Die Platinenlayouts für die Hauptplatine und Beleuchtungsplatten werden erstellt.	<input checked="" type="checkbox"/> erfüllt <input type="checkbox"/> teilweise erfüllt <input type="checkbox"/> nicht erfüllt
Anmerkung:	
Ziel n) Fertigung der Haupt- und Beleuchtungsplatten Die Platten werden gefertigt, bestückt und getestet.	<input checked="" type="checkbox"/> erfüllt <input type="checkbox"/> teilweise erfüllt <input type="checkbox"/> nicht erfüllt
Anmerkung:	
Ziel o) Verkabelung der gesamten Einheit Die elektronischen Bauteile werden in die Konstruktion eingebaut und vor Inbetriebnahme, sowie während des Betriebs getestet.	<input checked="" type="checkbox"/> erfüllt <input type="checkbox"/> teilweise erfüllt <input type="checkbox"/> nicht erfüllt
Anmerkung:	
Ziel p) Recherche der benötigten Sensoren Geeignete Sensoren und Bauteile werden gewählt.	<input checked="" type="checkbox"/> erfüllt <input type="checkbox"/> teilweise erfüllt <input type="checkbox"/> nicht erfüllt
Anmerkung:	

Optionale-Ziele:

Ziel	Zielerreichung
Ziel a) Bewerben des Projekts nach außen Während der Diplomarbeit sollte das Projekt auf verschiedenen Plattformen, wie zum Beispiel Facebook, Instagram und Twitter vertreten werden. Dies wird benötigt, um auf Rückmeldungen und Verbesserungsvorschläge von Interessenten reagieren zu können.	<input checked="" type="checkbox"/> erfüllt <input type="checkbox"/> teilweise erfüllt <input type="checkbox"/> nicht erfüllt
Anmerkung:	
Ziel b) Werbevideo Es wird ein Werbevideo erstellt, welches unsere Projektidee repräsentiert. Die Mindestlänge wurde auf 30 Sekunden festgelegt.	<input checked="" type="checkbox"/> erfüllt <input type="checkbox"/> teilweise erfüllt <input type="checkbox"/> nicht erfüllt
Anmerkung:	
Ziel c) Folder und Visitenkarten Es werden Folder erstellt damit Interessenten Informationen über unser Projekt erhalten. Weiters werden Visitenkarten für den Projektleiter erstellt, um unseren Sponsoren bzw. Interessenten Kontakt-Informationen hinterlassen zu können.	<input checked="" type="checkbox"/> erfüllt <input type="checkbox"/> teilweise erfüllt <input type="checkbox"/> nicht erfüllt
Anmerkung:	
Ziel d) Weitere Anpflanzeinheit fertigen Optional sollen weitere Anpflanzeinheiten gefertigt werden. Geplant sind maximal 3-4 Einheiten damit das ernten noch bequem möglich ist.	<input type="checkbox"/> erfüllt <input type="checkbox"/> teilweise erfüllt <input checked="" type="checkbox"/> nicht erfüllt
Anmerkung:	
Ziel e) Vollautomatische Bewässerung Optional wird eine Vertical-Farming-Einheit vollautomatisch bewässert.	<input type="checkbox"/> erfüllt <input type="checkbox"/> teilweise erfüllt <input checked="" type="checkbox"/> nicht erfüllt
Anmerkung:	
Ziel f) Manuelle Steuerung Eine manuelle Steuerung wird mittels Taster und Status LEDs verwirklicht.	<input type="checkbox"/> erfüllt <input checked="" type="checkbox"/> teilweise erfüllt <input type="checkbox"/> nicht erfüllt
Anmerkung:	

Hiermit bestätigen die Unterzeichnenden, dass die beauftragte Diplomarbeit „Vertical Farming“ vom Projektteam an den Auftraggeber übergeben wurde.

Ort und Datum der Unterzeichnung

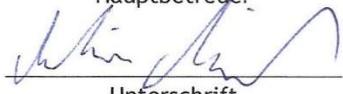
Wien, 18.3.2019

Philipp Gasser, PL



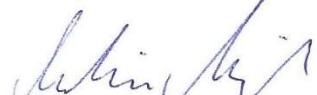
Unterschrift

Martin Meschik,
Hauptbetreuer



Unterschrift

Optional VN NN, PA



Unterschrift

Anhang B

	Tol. Maß	Größtmaß	Kleinstdmaß
--	----------	----------	-------------

A-A
1:1

Front View Dimensions:

- Central Hole: $\varnothing 11$
- Total Height: 25
- Side Slots: 2

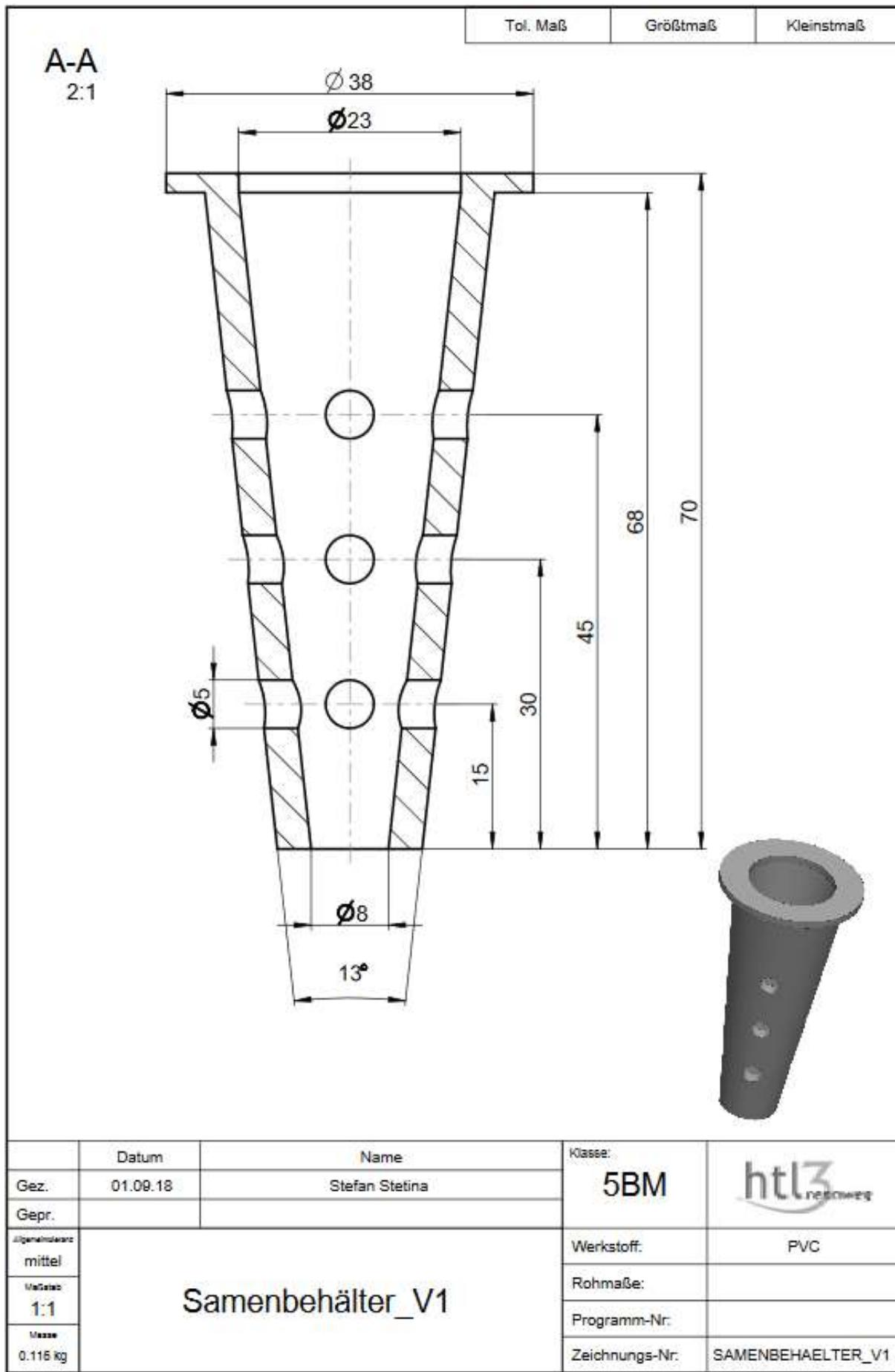
Top View Dimensions:

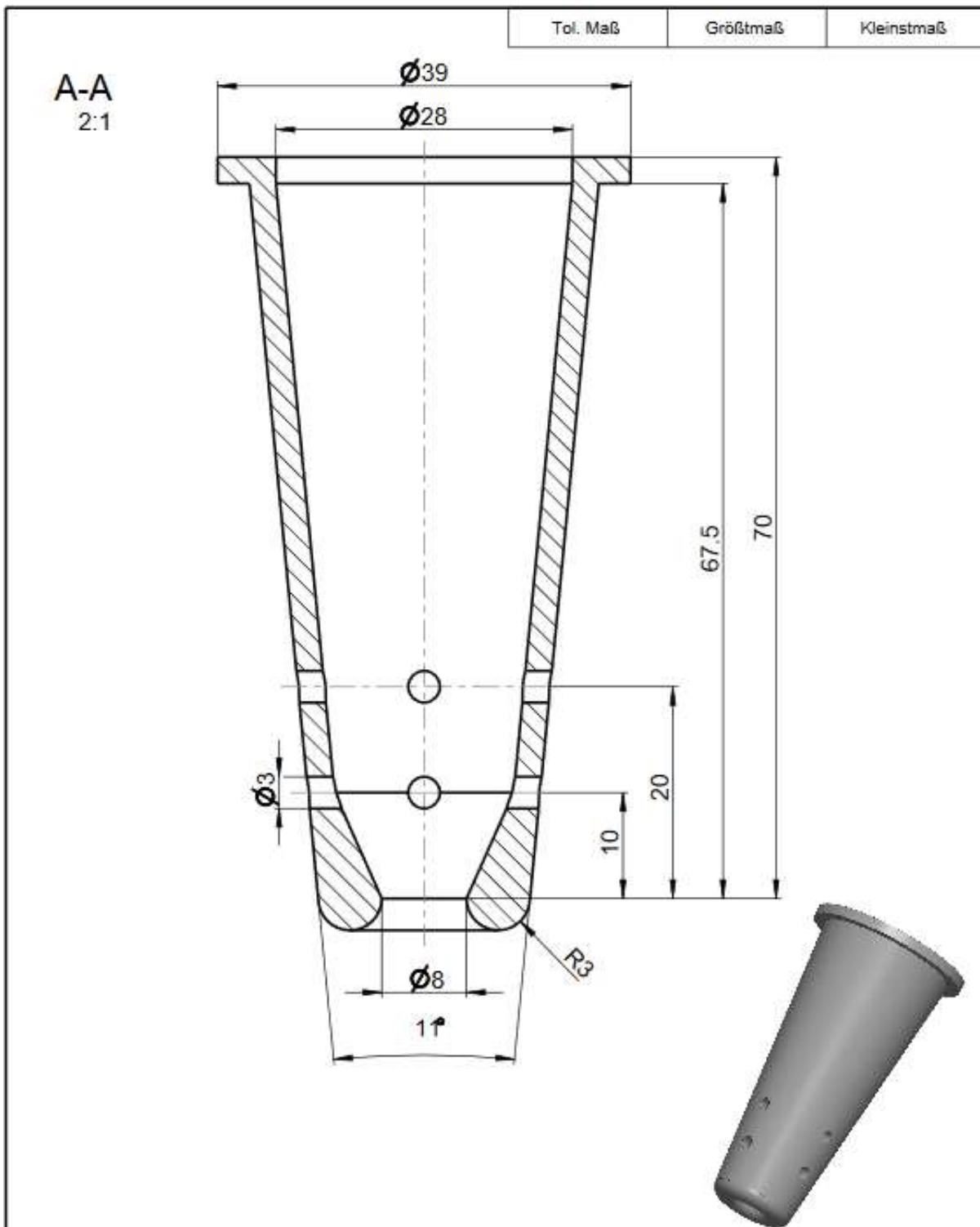
- Central Circle: 1
- Corner Slots: 4
- Total Width: 50
- Total Height: 25

Overall Dimensions:

- Total Length: 50
- Total Width: 25
- Total Height: 25

Part Number: ABSTANDSHALTER



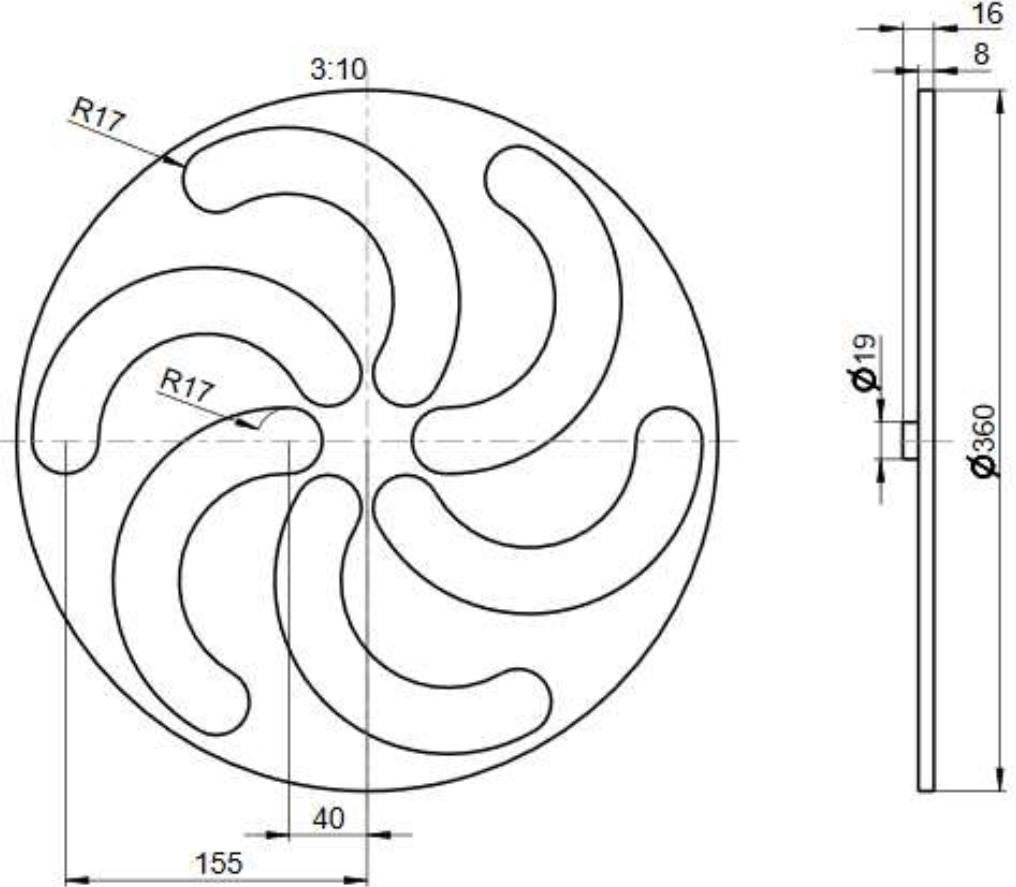


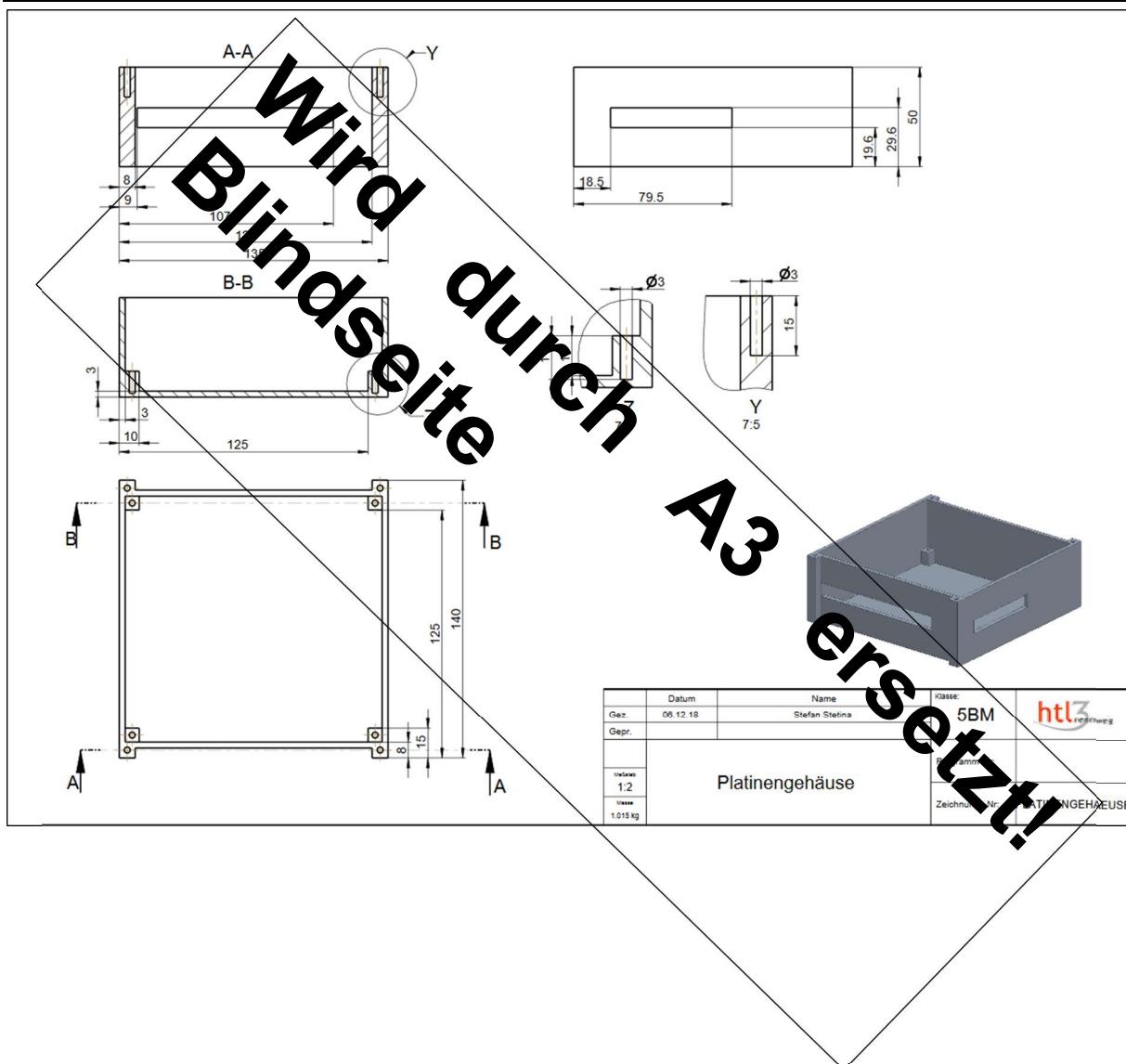
	Datum	Name	Klasse:	
Gez.	25.09.18	Stefan Stetina	5BM	
Gepr.				
Allgemeindaten				
mittel			Werkstoff:	PVC
Maßstab:			Rohmaße:	
2:1			Programm-Nr.:	
Masse			Zeichnungs-Nr.:	SAMENBEHAELTER
0.000 kg				
Samenbehälter				

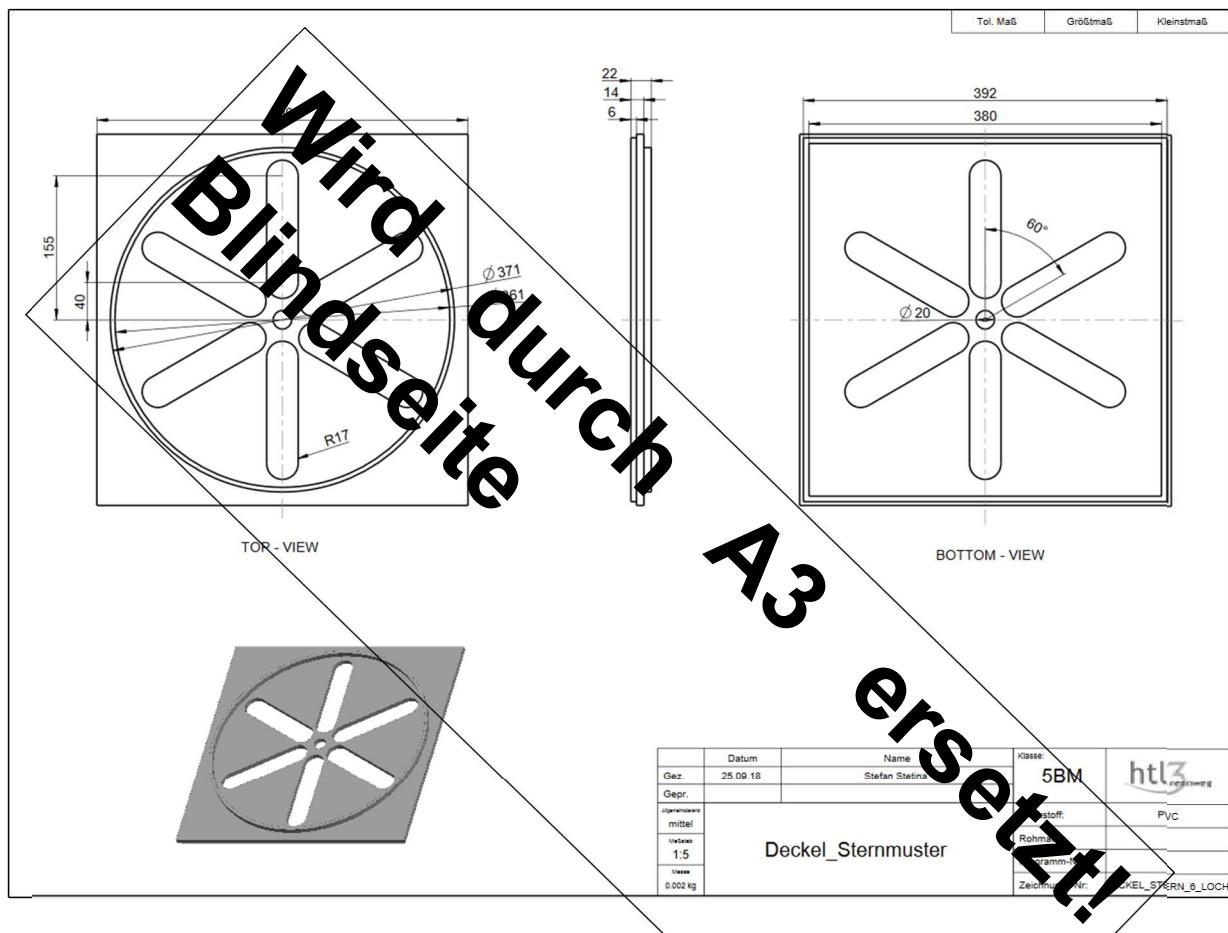
	Tol. Maß	Größmaß	Kleinstmaß

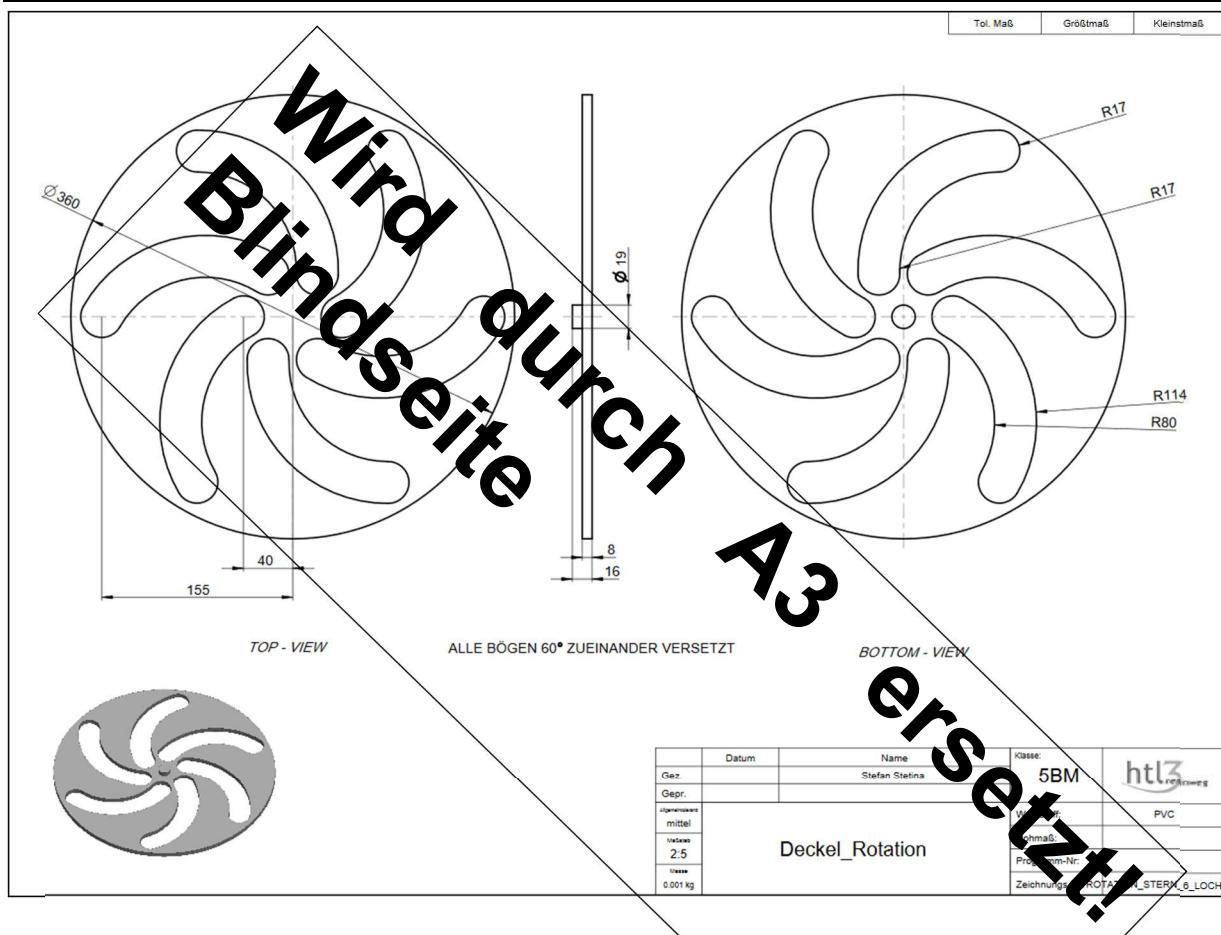
alle Bögen 60° zueinander versetzt

	Datum	Name	Klasse:
Gez.	18.09.18	Stefan Stetina	5BM
Gepr.			
Abmessungen: mittel			Werkstoff:
Maßstab 1:3			PVC
Kasse 0.001 kg	Deckel_Rotation		Rohmaße:
			Programm-Nr.:
			Zeichnungs-Nr.: ROTATIONSTEIL

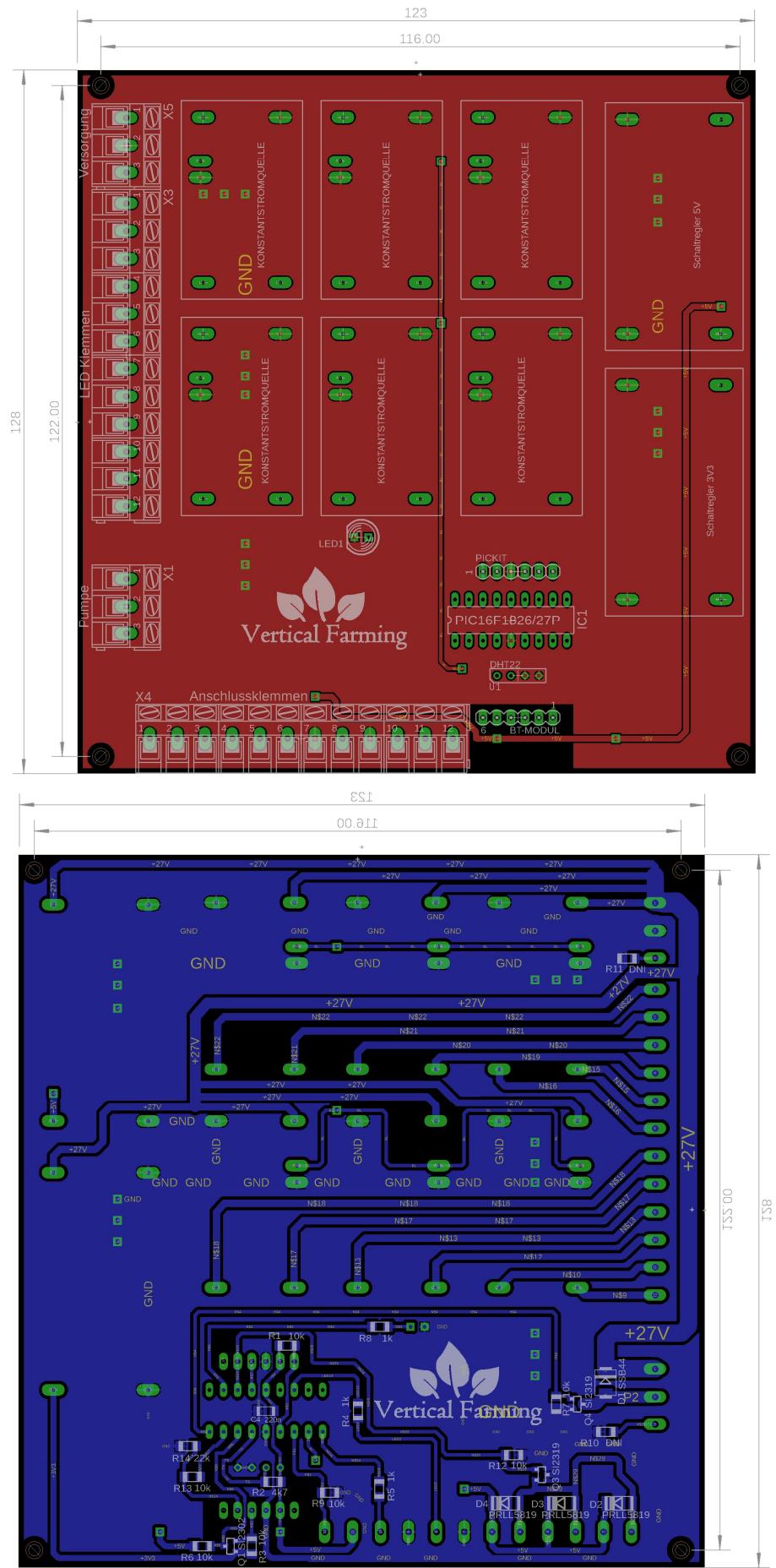
	Tol. Maß	Größtmaß	Kleinstmaß
			
alle Bögen 60 ° zueinander versetzt			
			
Gez.	Datum	Name	Klasse:
Gez.	25.09.18	Stefan Stetina	5BM
Gepr.			
Abgussmaßanz. mittel		Werkstoff:	PVC
Maßstab: 1:10		Rohmaße:	
Masse 0.001 kg		Programm-Nr.:	
		Zeichnungs-Nr.:	ROTATION_STERN_6.LOC
Deckel_Rotation			

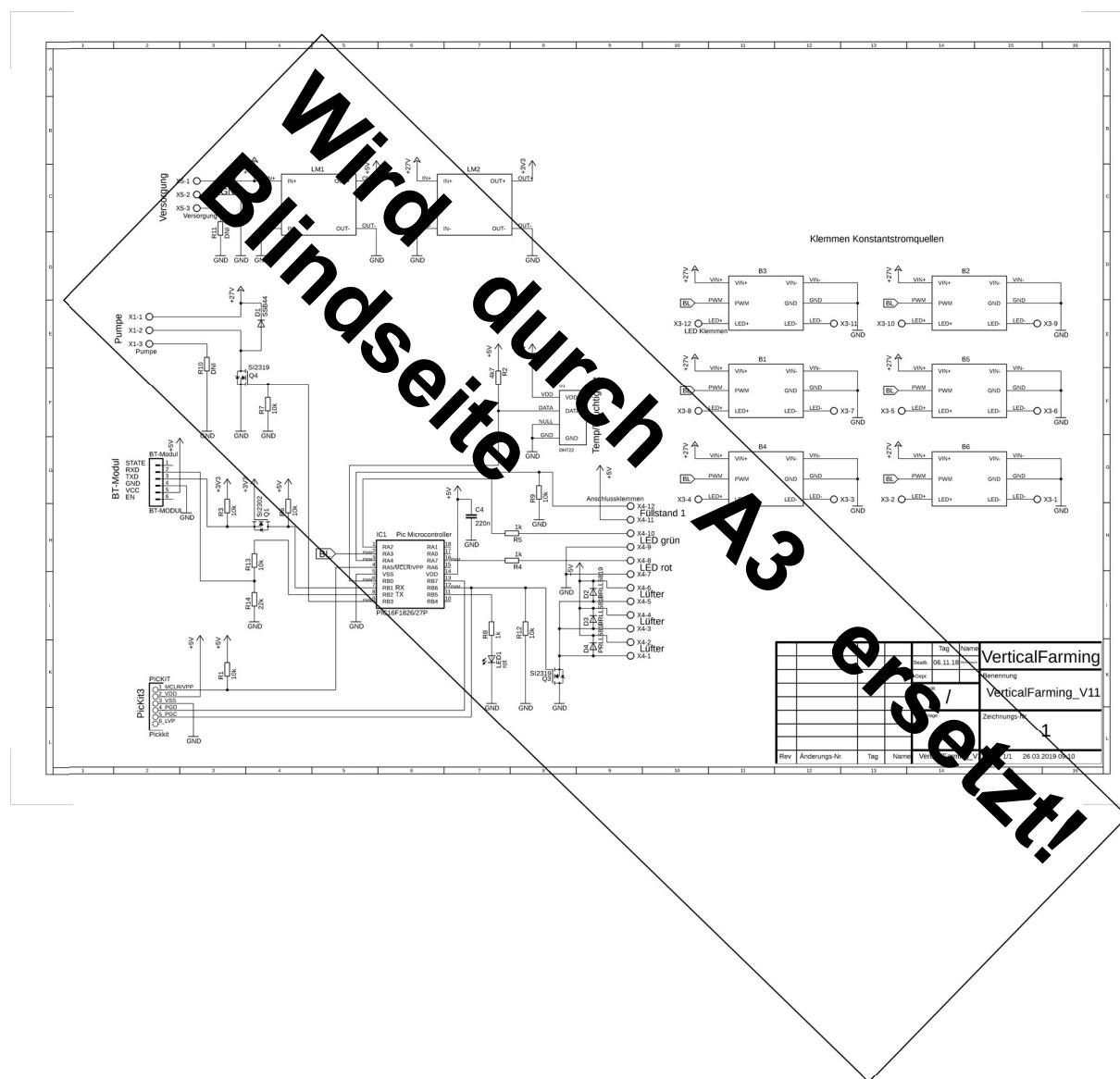






Anhang C





Anhang D

Programmcode PIC 16F1827:

```
/*
```

Generated Main Source File

Company:

Microchip Technology Inc.

File Name:

main.c

Summary:

This is the main file generated using PIC10 / PIC12 / PIC16 / PIC18 MCUs

Description:

This header file provides implementations for driver APIs for all modules selected in the GUI.

Generation Information :

Product Revision : PIC10 / PIC12 / PIC16 / PIC18 MCUs - 1.65.2

Device : PIC16F1827

Driver Version : 2.00

```
*/
```

```
/*
```

(c) 2018 Microchip Technology Inc. and its subsidiaries.

Subject to your compliance with these terms, you may use Microchip software and any

derivatives exclusively with Microchip products. It is your responsibility to comply with third party

license terms applicable to your use of third party software (including open source software) that

may accompany Microchip software.

THIS SOFTWARE IS SUPPLIED BY MICROCHIP "AS IS". NO WARRANTIES, WHETHER

EXPRESS, IMPLIED OR STATUTORY, APPLY TO THIS SOFTWARE, INCLUDING ANY

IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY, AND FITNESS

FOR A PARTICULAR PURPOSE.

IN NO EVENT WILL MICROCHIP BE LIABLE FOR ANY INDIRECT, SPECIAL, PUNITIVE,

INCIDENTAL OR CONSEQUENTIAL LOSS, DAMAGE, COST OR EXPENSE OF ANY KIND

WHATSOEVER RELATED TO THE SOFTWARE, HOWEVER CAUSED, EVEN IF MICROCHIP

HAS BEEN ADVISED OF THE POSSIBILITY OR THE DAMAGES ARE FORESEEABLE. TO

THE FULLEST EXTENT ALLOWED BY LAW, MICROCHIP'S TOTAL LIABILITY ON ALL

CLAIMS IN ANY WAY RELATED TO THIS SOFTWARE WILL NOT EXCEED THE AMOUNT

OF FEES, IF ANY, THAT YOU HAVE PAID DIRECTLY TO MICROCHIP FOR THIS

SOFTWARE.

*/

```
#define DHT22_PIN    RB0
#define DHT22_PIN_DIR TRISB0

#include "mcc_generated_files/mcc.h"
unsigned int x;
unsigned int y;
unsigned char buffer;
unsigned long timer;
unsigned long millis;
/*
Main application
*/
//variables declaration
char Temperature[] = "Temp = 00.0 C ";
char Humidity[] = "RH = 00.0 % ";
char Fuellstandok[] = "OKAY      ";
char Fuellstandnichtok[] = "LEER      ";
unsigned char T_Byte1, T_Byte2, RH_Byte1, RH_Byte2, CheckSum ;
unsigned int Temp, RH;
//unsigned char TempBL, TempTenthBL, HumidityBL, HumidityTenthBL;

// send start signal to the sensor
void Start_Signal(void) {
    DHT22_PIN_DIR = 0; // configure DHT22_PIN as output
    DHT22_PIN = 0; // clear DHT22_PIN output (logic 0)
```

```
_delay_ms(25);      // wait 25 ms
DHT22_PIN = 1;      // set DHT22_PIN output (logic 1)

_delay_us(30);      // wait 30 us
DHT22_PIN_DIR = 1;  // configure DHT22_PIN as input
}

// Check sensor response
_bit Check_Response() {
    TMR1H = 0;          // reset Timer1
    TMR1L = 0;
    TMR1ON = 1;         // enable Timer1 module

    while(!DHT22_PIN && TMR1L < 100); // wait until DHT22_PIN becomes
high (checking of 80µs low time response)

    if(TMR1L > 99)        // if response time > 99µS ==> Response error
        return 0;           // return 0 (Device has a problem with response)

    else
    {
        TMR1H = 0;          // reset Timer1
        TMR1L = 0;

        while(DHT22_PIN && TMR1L < 100); // wait until DHT22_PIN becomes low
(checking of 80µs high time response)

        if(TMR1L > 99)        // if response time > 99µS ==> Response error
            return 0;           // return 0 (Device has a problem with response)

        else
            return 1;           // return 1 (response OK)
    }
}

// Data read function
_bit Read_Data(unsigned char* dht_data)
{
    *dht_data = 0;

    for(char i = 0; i < 8; i++)
    {
        TMR1H = 0;          // reset Timer1
```

```

TMR1L = 0;

while(!DHT22_PIN)    // wait until DHT22_PIN becomes high
if(TMR1L > 100) {   // if low time > 100 ==> Time out error (Normally it
takes 50µs)
    return 1;
}

TMR1H = 0;           // reset Timer1
TMR1L = 0;

while(DHT22_PIN)     // wait until DHT22_PIN becomes low
if(TMR1L > 100) {   // if high time > 100 ==> Time out error (Normally it
takes 26-28µs for 0 and 70µs for 1)
    return 1;         // return 1 (timeout error)
}

if(TMR1L > 50)       // if high time > 50 ==> Sensor sent 1
*dht_data |= (1 << (7 - i)); // set bit (7 - i)
}

return 0;             // return 0 (data read OK)
}
void readDHT22(void)
{
    Start_Signal(); // send start signal to the sensor

    if(Check_Response()) // check if there is a response from sensor (If OK
start reading humidity and temperature data)
    {
        // read (and save) data from the DHT22 sensor and check time out errors
        if(Read_Data(&RH_Byte1)      ||      Read_Data(&RH_Byte2)      ||
Read_Data(&T_Byte1) || Read_Data(&T_Byte2) || Read_Data(&CheckSum))
        {
        }

        else // if there is no time out error
        {
            if(CheckSum == ((RH_Byte1 + RH_Byte2 + T_Byte1 + T_Byte2) & 0xFF))
            { // if there is no checksum error
                RH = RH_Byte1;
                RH = (RH << 8) | RH_Byte2;
                Temp = T_Byte1;
            }
        }
    }
}

```

```

Temp = (Temp << 8) | T_Byte2;

if (Temp > 0X8000)
{
    // if temperature < 0
    Temperature[6] = '-';
    Temp = Temp & 0X7FFF;
}
else // otherwise (temperature > 0)
{
    Temperature[6] = ' ';
    Temperature[7] = (Temp / 100) % 10 + '0';
    Temperature[8] = (Temp / 10) % 10 + '0';
    Temperature[10] = Temp % 10 + '0';
    Temperature[11] = 223; // put degree symbol (°)

if(RH == 1000) // if the relative humidity = 100.0 %
    Humidity[6] = 1 + '0'; // put 1 of hundreds
else
{
    Humidity[6] = ' '; // put space ''
    Humidity[7] = (RH / 100) % 10 + '0';
    Humidity[8] = (RH / 10) % 10 + '0';
    Humidity[10] = RH % 10 + '0';
}

void main(void)
{
    // initialize the device
    SYSTEM_Initialize();
    Status_PIC_SetHigh(); //PIC power on = LED power on.
    Luefter_SetHigh(); //activate fan
    // When using interrupts, you need to set the Global and Peripheral
    Interrupt Enable bits
    // Use the following macros to:

    // Enable the Global Interrupts
    INTERRUPT_GlobalInterruptEnable();

    // Enable the Peripheral Interrupts
    INTERRUPT_PeripheralInterruptEnable();
}

```

```
//while(1)
//{
//while(EUSART_is_rx_ready() == 0);
//EUSART_Write(EUSART_Read());}

// Disable the Global Interrupts
// INTERRUPT_GlobalInterruptDisable();

// Disable the Peripheral Interrupts
// INTERRUPT_PeripheralInterruptDisable();

while (1)
{
    readDHT22();
    if(eusartRxCount > 0)
    {
        buffer = EUSART_Read();
        switch(buffer)
        {

            case 'T':
            {
                EUSART_Write('t');
                EUSART_Write(Temperature[7]);
                EUSART_Write(Temperature[8]);
                EUSART_Write(Temperature[9]);
                EUSART_Write(Temperature[10]);
                break;
            }
            case 'H':
            {
                EUSART_Write('h');
                EUSART_Write(Humidity[7]);
                EUSART_Write(Humidity[8]);
                EUSART_Write(Humidity[9]);
                EUSART_Write(Humidity[10]);
                break;
            }
            case 'F':
            {
                if (Fuellstand_GetValue() == 0)
                {
```

```

EUSART_Write('f');
EUSART_Write(Fuellstandnichtok[0]);
EUSART_Write(Fuellstandnichtok[1]);
EUSART_Write(Fuellstandnichtok[2]);
EUSART_Write(Fuellstandnichtok[3]);
break;
}
else if (Fuellstand_GetValue() == 1)
{
    EUSART_Write('f');
    EUSART_Write(Fuellstandok[0]);
    EUSART_Write(Fuellstandok[1]);
    EUSART_Write(Fuellstandok[2]);
    EUSART_Write(Fuellstandok[3]);
    break;
}
}

case 'B':
{
    NOP(); //no operation
    unsigned char buffer1[] = {0, 0, 0, 0};
    for (unsigned char i = 0; i < 4; i++) //Startwert der
Zählervariable setzen; Durchlauf-Bedingung (wird geprüft); -
    {
        // Operation auf Zählervariable
ausführen;
        buffer1[i] = EUSART_Read();
        if (buffer1[i] == ':')
        {
            switch (i)
            {
                case 1: //0-9
                {
                    x = buffer1[0] - 0x30; //bsp.: B5 received
                    break;
                }
                case 2: //10-99
                {
                    x = 10 * (buffer1[0] - 0x30); //10er Stelle
                    x += buffer1[1] - 0x30; //1er Stelle
                    break; //bsp.: B10 received
                }
            }
        }
    }
}

```

```

        case 3:          //100-255
    {
        x = 100 * (buffer1[0] - 0x30); //100er Stelle
        x += 10 * (buffer1[1] - 0x30); //10er Stelle
        x += buffer1[2] - 0x30;      //1er Stelle
        break;                  //bsp.: B100 received
    }
}
break;
}

}
x = 399 - (x - 0) * (399 - 0) / (100 - 0) + 0;    //convert % to value
between 0 - 399; active low
    PWM3_LoadDutyValue(x);           //brightness
    break;
}
case 'W':
{
    NOP();                      //no operation
    unsigned char buffer2[] = {0, 0, 0, 0};
    for (unsigned char i = 0; i < 4; i++)
    {
        buffer2[i] = EUSART_Read();
        if (buffer2[i] == ':')
        {
            switch (i)
            {
                case 1:          //0-9
                {
                    y = buffer2[0] - 0x30;    //bsp.: W5 received
                    break;
                }
                case 2:          //10-99
                {
                    y = 10 * (buffer2[0] - 0x30); //10er Stelle
                    y += buffer2[1] - 0x30;     //1er Stelle
                    break;                  //bsp.: W10 received
                }
                case 3:          //100-255
                {
                    y = 100 * (buffer2[0] - 0x30); //100er Stelle
                    y += 10 * (buffer2[1] - 0x30); //10er Stelle
                }
            }
        }
    }
}

```

```

        y += buffer2[2] - 0x30;      //1er Stelle
        break;                      //bsp.: W100 received
    }
}
break;
}

y = (y - 0) * (88 - 0) / (100 - 0) + 0;      //convert % to value
between 0 - 399
    EPWM1_LoadDutyValue(y);          //maximaler Wert bei
der Pumpe ist 88
    break;
}
default:
{
}

}
}

if (Fuellstand_GetValue() == 1)
{
    Status_Fuellstandgruen_SetHigh();           //level of fluid okay.
Green LED power on.
    Status_Fuellstandrot_SetLow();
}
else if (Fuellstand_GetValue() == 0)
{
    Status_Fuellstandrot_SetHigh();             //level of fluid not okay,
Red LED power on.
    Status_Fuellstandgruen_SetLow();
}
}

/***
End of File
*/

```

Programmcode Android-App:

```
//MAINACTIVITY FILE

package a.www.sliderv2;

import android.app.Service;
import android.net.Uri;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.widget.Button;
import android.widget.SeekBar;
import android.widget.TextView;
import android.bluetooth.BluetoothAdapter;
import android.bluetooth.BluetoothDevice;
import android.bluetooth.BluetoothSocket;
import android.content.Intent;
import android.os.Handler;
import android.os.Message;
import android.view.View;
import android.widget.Toast;

import java.io.IOException;
import java.io.InputStream;
import java.io.OutputStream;
import java.util.Set;
import java.util.UUID;

public class MainActivity extends AppCompatActivity
{
    // Used to load the 'native-lib' library on application startup.
    static
    {
        System.loadLibrary("native-lib");
    }

    public TextView HelloField;
    public TextView Temperatur;
    public TextView Feuchtigkeit;
    public TextView Fuellstand;
    public TextView ConnectionStatus;
    TextView textView2;
    TextView textView3;
    SeekBar seekBar2;
    SeekBar seekBar3;
    Button btnabout;
    Button btnhome;
    Button websitebtn;
    TextView appdeveloper;
    TextView nogru;
    TextView projectteam;
    TextView pg;
    TextView ss;
    TextView uo;
    TextView ng;
    TextView projectmanagement;
    TextView konstruktion;
    TextView elektronik;
    TextView programmierung;
    TextView Bewässerung;
```

```

TextView Beleuchtung;
TextView TextView;
TextView TextView4;
TextView TextView5;

int min = 0, max = 100, current = 0;

//BT
public BluetoothAdapter BtAdapter=null;
public static UUID MY_UUID;
public String DEV_MAC="98:D3:34:90:4C:8E";
public ConnectThread DevConnect;
private ConnectedThread mConnectedThread;
private char TermC='\n';

//Handler
protected static final int SUCCESS_CONNECT = 0;
protected static final int MESSAGE_READ = 1;
public String recString="";

@Override
protected void onCreate(Bundle savedInstanceState)
{
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_main);

    start();
    HelloField=(TextView) findViewById(R.id.HalloID);
//ausblenden
    Temperatur=(TextView) findViewById(R.id.temperatur);
//ausblenden
    Feuchtigkeit=(TextView) findViewById(R.id.feuchtigkeit);
//ausblenden
    Fuellstand=(TextView) findViewById(R.id.fuellstand);
//ausblenden
    ConnectionStatus=(TextView) findViewById(R.id.ConnStat);
//ausblenden
    ConnectionStatus.setText("not
                                Connected");
//ausblenden
    textView2      =      (TextView)      findViewById(R.id.textView2);
//ausblenden
    seekBar2       =      (SeekBar)      findViewById(R.id.seekBar2);
//ausblenden
    textView3      =      (TextView)      findViewById(R.id.textView3);
//ausblenden
    seekBar3       =      (SeekBar)      findViewById(R.id.seekBar3);
//ausblenden
    Bewässerung     =      (TextView)      findViewById(R.id.Bewässerung);
//ausblenden
    Beleuchtung     =      (TextView)      findViewById(R.id.Beleuchtung);
//ausblenden
    TextView = (TextView) findViewById(R.id.textView);
    TextView4 = (TextView) findViewById(R.id.textView4);
    TextView5 = (TextView) findViewById(R.id.textView5);

    btnabout = (Button) findViewById(R.id.btnabout);
    btnhome = (Button) findViewById(R.id.btnhome);
    websitebtn = (Button) findViewById(R.id.websitebtn);

```

```

appdeveloper = (TextView) findViewById(R.id.appdeveloper);
nogru = (TextView) findViewById(R.id.nogru);
projectteam = (TextView) findViewById(R.id.projectteam);
pg = (TextView) findViewById(R.id.pg);
ss = (TextView) findViewById(R.id.ss);
uo = (TextView) findViewById(R.id.uo);
ng = (TextView) findViewById(R.id.ng);
projectmanagement = (TextView) findViewById(R.id.projectmanagement);
konstruktion = (TextView) findViewById(R.id.konstruktion);
elektronik = (TextView) findViewById(R.id.elektronik);
programmierung = (TextView) findViewById(R.id.programmierung);

seekBar2.setMax(max - min);
seekBar2.setProgress(current - min);
textView2.setText(" " + current);
seekBar3.setMax(max - min);
seekBar3.setProgress(current - min);
textView3.setText(" " + current);

//BT
BtAdapter = BluetoothAdapter.getDefaultAdapter();
if (BtAdapter == null) {
    Toast.makeText(getApplicationContext(), "No bluetooth detected!", Toast.LENGTH_SHORT).show();
    finish();
} {
    if (!BtAdapter.isEnabled()) {
        turnOnBT();
    }
    try {
        int count=0;
        do {
            Thread.sleep(1000);
            count=count+1;
        }while (!connect2devMAC(DEV_MAC)&&count<10);
    } catch (InterruptedException ex) {
    }
}
seekBar2.setOnSeekBarChangeListener(new SeekBar.OnSeekBarChangeListener() {
    @Override
    public void onProgressChanged(SeekBar seekBar, int progress,
boolean fromUser)
    {
        current = progress + min;
        textView2.setText(" " + current);
        mConnectedThread.write("W" + current + ",");
    }

    @Override
    public void onStartTrackingTouch(SeekBar seekBar) {

    }

    @Override
    public void onStopTrackingTouch(SeekBar seekBar) {

    }
});

```

```

seekBar3.setOnSeekBarChangeListener(new
SeekBar.OnSeekBarChangeListener()
{
    @Override
    public void onProgressChanged(SeekBar seekBar, int progress,
boolean fromUser) {
        current = progress + min;
        textView3.setText("" + current);
        mConnectedThread.write("B" + current + ",");
    }

    @Override
    public void onStartTrackingTouch(SeekBar seekBar) {

    }

    @Override
    public void onStopTrackingTouch(SeekBar seekBar) {

    }
});
// Example of a call to a native method
//TextView tv = (TextView) findViewById(R.id.sample_text);
//tv.setText(stringFromJNI());
}

public void btnaboutClicked(View view)
{
    HelloField.setVisibility(View.GONE);
    Temperatur.setVisibility(View.GONE);
    Feuchtigkeit.setVisibility(View.GONE);
    Fuellstand.setVisibility(View.GONE);
    ConnectionStatus.setVisibility(View.GONE);
    textView2.setVisibility(View.GONE);
    textView3.setVisibility(View.GONE);
    seekBar2.setVisibility(View.GONE);
    seekBar3.setVisibility(View.GONE);
    Bewässerung.setVisibility(View.GONE);
    Beleuchtung.setVisibility(View.GONE);
    btnabout.setVisibility(View.GONE);
    websitebtn.setVisibility(View.GONE);
    TextView.setVisibility(View.GONE);
    TextView4.setVisibility(View.GONE);
    TextView5.setVisibility(View.GONE);

    btnhome.setVisibility(View.VISIBLE);
    appdeveloper.setVisibility(View.VISIBLE);
    nogru.setVisibility(View.VISIBLE);
    projectteam.setVisibility(View.VISIBLE);
    pg.setVisibility(View.VISIBLE);
    ss.setVisibility(View.VISIBLE);
    uo.setVisibility(View.VISIBLE);
    ng.setVisibility(View.VISIBLE);
    projectmanagement.setVisibility(View.VISIBLE);
    konstruktion.setVisibility(View.VISIBLE);
    elektronik.setVisibility(View.VISIBLE);
    programmierung.setVisibility(View.VISIBLE);
}

public void btnhomeClicked(View view)
{
}

```

```
btnhome.setVisibility(View.GONE);
appdeveloper.setVisibility(View.GONE);
nogru.setVisibility(View.GONE);
projectteam.setVisibility(View.GONE);
pg.setVisibility(View.GONE);
ss.setVisibility(View.GONE);
uo.setVisibility(View.GONE);
ng.setVisibility(View.GONE);
projectmanagement.setVisibility(View.GONE);
konstruktion.setVisibility(View.GONE);
elektronik.setVisibility(View.GONE);
programmierung.setVisibility(View.GONE);

TextView.setVisibility(View.VISIBLE);
TextView4.setVisibility(View.VISIBLE);
TextView5.setVisibility(View.VISIBLE);
websitebtn.setVisibility(View.VISIBLE);
HelloField.setVisibility(View.VISIBLE);
Temperatur.setVisibility(View.VISIBLE);
Feuchtigkeit.setVisibility(View.VISIBLE);
Fuellstand.setVisibility(View.VISIBLE);
ConnectionStatus.setVisibility(View.VISIBLE);
textView2.setVisibility(View.VISIBLE);
textView3.setVisibility(View.VISIBLE);
seekBar2.setVisibility(View.VISIBLE);
seekBar3.setVisibility(View.VISIBLE);
Bewässerung.setVisibility(View.VISIBLE);
Beleuchtung.setVisibility(View.VISIBLE);
btnabout.setVisibility(View.VISIBLE);
}

public void website(View view)
{
    Intent websitebtnIntent = new
Intent(Intent.ACTION_VIEW,Uri.parse("http://www.vertical-farming.at"));
    startActivity(websitebtnIntent);
}
private boolean started;
private Handler timerHandler = new Handler();
private Runnable runnable = new Runnable()
{
    @Override
    public void run()
    {
        mConnectedThread.write("T");
        mConnectedThread.write("H");
        mConnectedThread.write("F");

        if (started)
        {
            start();
        }
    }
};

public void start()
{
    started = true;
```

```

        timerHandler.postDelayed(this.runnable, 8000);
    }

public void stop()
{
    started = false;
    timerHandler.removeCallbacks(runnable);
}

private void turnOnBT() {
    Intent intent = new Intent(BluetoothAdapter.ACTION_REQUEST_ENABLE);
    startActivityForResult(intent, 1);
}

//connect with BT-MAC
public boolean connect2devMAC(String btMAC) {
    String deviceMAC = btMAC;
    BluetoothDevice result = null;
    boolean retval=false;

    Set<BluetoothDevice> deviceArr = BtAdapter.getBondedDevices();
    if (deviceArr != null) {
        for (BluetoothDevice device : deviceArr) {
            if (deviceMAC.equals(device.getAddress())) {
                result = device;
                break;
            }
        }
    }
    if (result!=null) {
        DevConnect = new ConnectThread(result);
        DevConnect.start();
        retval=true;
    }
    return retval;
}

// Handler to communicate with BT
private Handler mHandler = new Handler() {
    @Override
    public void handleMessage(Message msg) {
        //super.handleMessage(msg);
        switch (msg.what) {
            case SUCCESS_CONNECT:
                mConnectedThread = new
ConnectedThread((BluetoothSocket) msg.obj);
                Toast.makeText(getApplicationContext(),
"CONNECTED", Toast.LENGTH_SHORT).show();
                ConnectionStatus.setText("Connected");
                BtAdapter.cancelDiscovery();
                //String s = "-2w?" + TermC;;
                //mConnectedThread.write(s);
                mConnectedThread.start(); //wenn auskommentiert, kommt er in die Funktion in der er die Werte ausliest nicht hinein
                break;
            case MESSAGE_READ:
                byte[] readBuff = (byte[]) msg.obj;
                String string = new String(readBuff);
                Integer iend=string.indexOf('\0');
                Integer inl=string.indexOf('\n');

```

```

        //ConnectionStatus.setText(string);
        recString=recString+string.substring(0,iend);
        //if (recString.indexOf('\n')<0)
        //  break;
        //String temp=recString;
        if (recString.length() < 5)
        {
            return;
        }
        switch(recString.charAt(0))
        {
            case 't':
            {
                Temperatur.setText(recString.substring(1,5)
+ "°C");
                break;
            }

            case 'h':
            {

                Feuchtigkeit.setText(recString.substring(1,5) + "%");
                break;
            }

            case 'f':
            {

                Fuellstand.setText(recString.substring(1,5));
                if
                (recString.substring(1,5).equals("LEER"))
                    seekBar2.setProgress(0);
                break;
            }
            recString = "";
            default:
                //Toast.makeText(getApplicationContext(),
                "RECEIVING", Toast.LENGTH_SHORT).show();
                //break;
            }

        }
    };

// add'l private classes -----
-----
//create new class for connect thread
private class ConnectThread extends Thread {

    private BluetoothSocket mmSocket=null;
    private BluetoothDevice mmDevice=null;

    public ConnectThread(BluetoothDevice device) {
        // Use a temporary object that is later assigned to mmSocket,
        // because mmSocket is final
        BluetoothSocket tmp = null;
        mmDevice = (BluetoothDevice)device;

        // Get a BluetoothSocket to connect with the given
        BluetoothDevice

```

```

try {
    // MY_UUID is the app's UUID string, also used by the
server code
    MY_UUID=mmDevice.getUuids()[0].getUuid();
    tmp =
mmDevice.createInsecureRfcommSocketToServiceRecord(MY_UUID);
} catch (IOException e) {
    e.printStackTrace();
}
mmSocket = tmp;
//mmSocket.wait();
}

public void run() {
    // Cancel discovery because it will slow down the connection
BtAdapter.cancelDiscovery();

try {
    // Connect the device through the socket. This will block
    // until it succeeds or throws an exception
    mmSocket.connect();
} catch (IOException connectException) {
    // Unable to connect; close the socket and get out
    try {
        mmSocket.close();
    } catch (IOException e) {
    }
    return;
}

mHandler.obtainMessage(SUCCESS_CONNECT,mmSocket).sendToTarget();
}

< /**
 * Will cancel an in-progress connection, and close the socket
 */
public void cancel() {
    try {
        mmSocket.close();
    } catch (IOException e) {
    }
}
}

//create new class for connected thread
private class ConnectedThread extends Thread {
    private final InputStream mmInStream;
    private final OutputStream mmOutStream;

    //creation of the connect thread
public ConnectedThread(BluetoothSocket socket) {
    //btSocket=socket;
    InputStream tmpIn = null;
    OutputStream tmpOut = null;

    try {
        //Create I/O streams for connection
        tmpIn = socket.getInputStream();
        tmpOut = socket.getOutputStream();
    } catch (IOException e) {
}

```

```

        }

        mmInStream = tmpIn;
        mmOutStream = tmpOut;
    }

    public void run() {
        byte[] buffer = new byte[256];
        int bytes;

        // Keep looping to listen for received messages
        while (true) {
            try {

                if (mmInStream.available() > 0) {
                    buffer = new byte[1024];
                    bytes      =          mmInStream.read(buffer);
//read bytes from input buffer
                    String readMessage = new String(buffer, 0,
bytes);

                    //String myString = new String(buffer);
                    //HelloField.setText(bytes);

                    // Send the obtained bytes to the UI Activity
via handler
                    mHandler.obtainMessage(MESSAGE_READ, bytes, -1,
buffer).sendToTarget();
                }
            } catch (IOException e) {
                break;
            }
        }
    }

    //write method
    public void write(String input) {
        byte[] msgBuffer = input.getBytes();                      //converts
entered String into bytes
        try {
            mmOutStream.write(msgBuffer);                         //write bytes
over BT connection via outstream
            mmOutStream.flush();
        } catch (IOException e) {
            //if you cannot write, close the application
            Toast.makeText(getApplicationContext(), "Connection Failure",
Toast.LENGTH_LONG).show();
        }
    }

    /**
     * A native method that is implemented by the 'native-lib' native
library,
     * which is packaged with this application.
     */
// public native String stringFromJNI();

}

```

```
//ANROIDMANIFEST FILE

<?xml version="1.0" encoding="utf-8" ?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="a.www.sliderv2">

    <uses-permission android:name="android.permission.BLUETOOTH" />
    <uses-permission android:name="android.permission.BLUETOOTH_ADMIN" />

    <application
        android:allowBackup="true"
        android:icon="@mipmap/ic_launcher"
        android:label="@string/app_name"
        android:roundIcon="@mipmap/ic_launcher"
        android:supportsRtl="true"
        android:theme="@style/AppTheme">
        <activity android:name=".MainActivity">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />

                <category android:name="android.intent.category.LAUNCHER"
/>

            </intent-filter>
        </activity>
    </application>

</manifest>
```

```
//ACTIVITY_MAIN (FOR LAYOUT)

<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:background="#ffffffff"
    tools:context=".MainActivity">

    <TextView
        android:id="@+id/textView5"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_below="@+id/textView4"
        android:layout_alignStart="@+id/textView4"
        android:layout_alignLeft="@+id/textView4"
        android:layout_alignEnd="@+id/seekBar3"
        android:layout_alignRight="@+id/seekBar3"
        android:layout_marginStart="0dp"
        android:layout_marginLeft="0dp"
        android:layout_marginTop="13dp"
        android:layout_marginEnd="192dp"
        android:layout_marginRight="192dp"
        android:text="Füllstand"
        android:textSize="18sp"
        android:visibility="visible" />

    <TextView
        android:id="@+id/textView3"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_alignStart="@+id/seekBar3"
        android:layout_alignLeft="@+id/seekBar3"
        android:layout_alignTop="@+id/Beleuchtung"
        android:layout_alignEnd="@+id/seekBar3"
        android:layout_alignRight="@+id/seekBar3"
        android:layout_marginStart="163dp"
        android:layout_marginLeft="163dp"
        android:layout_marginTop="0dp"
        android:layout_marginEnd="162dp"
        android:layout_marginRight="162dp"
        android:text="Value2"
        android:visibility="visible"
        tools:layout_editor_absoluteX="188dp"
        tools:layout_editor_absoluteY="159dp" />

    <TextView
        android:id="@+id/Beleuchtung"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_below="@+id/seekBar2"
        android:layout_alignStart="@+id/seekBar2"
        android:layout_alignLeft="@+id/seekBar2"
        android:layout_alignEnd="@+id/seekBar3"
        android:layout_alignRight="@+id/seekBar3"
        android:layout_marginStart="20dp"
        android:layout_marginLeft="20dp"
        android:layout_marginTop="18dp"
```

```
        android:layout_marginEnd="247dp"
        android:layout_marginRight="247dp"
        android:text="Beleuchtung"
        android:textStyle="bold"
        android:visibility="visible"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintHorizontal_bias="0.253"
        app:layout_constraintLeft_toLeftOf="parent"
        app:layout_constraintRight_toRightOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.323" />

<TextView
    android:id="@+id/Bewässerung"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_alignEnd="@+id/seekBar2"
    android:layout_alignRight="@+id/seekBar2"
    android:layout_alignParentTop="true"
    android:layout_marginTop="48dp"
    android:layout_marginEnd="247dp"
    android:layout_marginRight="247dp"
    android:text="Bewässerung"
    android:textStyle="bold"
    android:visibility="visible"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintHorizontal_bias="0.253"
    app:layout_constraintLeft_toLeftOf="parent"
    app:layout_constraintRight_toRightOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintVertical_bias="0.113" />

<SeekBar
    android:id="@+id/seekBar2"
    android:layout_width="368dp"
    android:layout_height="wrap_content"
    android:layout_below="@+id/textView2"
    android:layout_alignParentStart="true"
    android:layout_marginStart="9dp"
    android:layout_marginTop="12dp"
    android:visibility="visible"
    tools:layout_editor_absoluteX="8dp"
    tools:layout_editor_absoluteY="108dp" />

<SeekBar
    android:id="@+id/seekBar3"
    android:layout_width="368dp"
    android:layout_height="wrap_content"
    android:layout_alignParentStart="true"
    android:layout_alignParentTop="true"
    android:layout_marginStart="9dp"
    android:layout_marginTop="147dp"
    android:visibility="visible"
    tools:layout_editor_absoluteX="8dp"
    tools:layout_editor_absoluteY="206dp" />

<TextView
    android:id="@+id/textView2"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_alignStart="@+id/seekBar3"
```

```
        android:layout_alignLeft="@+id/seekBar3"
        android:layout_alignTop="@+id/Bewässerung"
        android:layout_alignEnd="@+id/seekBar3"
        android:layout_alignRight="@+id/seekBar3"
        android:layout_marginStart="162dp"
        android:layout_marginLeft="162dp"
        android:layout_marginEnd="163dp"
        android:layout_marginRight="163dp"
        android:text="Value1"
        android:visibility="visible"
        tools:layout_editor_absoluteX="188dp"
        tools:layout_editor_absoluteY="56dp" />

<TextView
    android:id="@+id/HalloID"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_alignParentStart="true"
    android:layout_alignParentTop="true"
    android:layout_marginStart="38dp"
    android:layout_marginTop="10dp"
    android:text="Bluetooth Connection:"
    android:textStyle="bold"
    android:visibility="visible" />

<TextView
    android:id="@+id/ConnStat"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_alignStart="@+id/HalloID"
    android:layout_alignLeft="@+id/HalloID"
    android:layout_alignTop="@+id/HalloID"
    android:layout_marginStart="170dp"
    android:layout_marginLeft="170dp"
    android:layout_marginTop="1dp"
    android:text="TextView"
    android:textAllCaps="true"
    android:textStyle="bold"
    android:visibility="visible" />

<Button
    android:id="@+id/websitebtn"
    android:layout_width="372dp"
    android:layout_height="149dp"
    android:layout_below="@+id/seekBar3"
    android:layout_alignStart="@+id/seekBar3"
    android:layout_alignLeft="@+id/seekBar3"
    android:layout_alignEnd="@+id/seekBar3"
    android:layout_alignRight="@+id/seekBar3"
    android:layout_alignParentBottom="true"
    android:layout_marginStart="2dp"
    android:layout_marginLeft="2dp"
    android:layout_marginTop="150dp"
    android:layout_marginEnd="-6dp"
    android:layout_marginRight="-6dp"
    android:layout_marginBottom="47dp"
    android:background="@drawable/logodiplov4"
    android:onClick="website"
    android:visibility="visible" />
```

```
<TextView
    android:id="@+id/textView"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_below="@+id/seekBar3"
    android:layout_alignEnd="@+id/seekBar3"
    android:layout_alignRight="@+id/seekBar3"
    android:layout_marginTop="35dp"
    android:layout_marginEnd="194dp"
    android:layout_marginRight="194dp"
    android:text="Temperatur"
    android:textSize="18sp"
    android:visibility="visible" />

<TextView
    android:id="@+id/textView4"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_below="@+id/textView"
    android:layout_alignStart="@+id/textView"
    android:layout_alignLeft="@+id/textView"
    android:layout_marginStart="0dp"
    android:layout_marginLeft="0dp"
    android:layout_marginTop="14dp"
    android:text="Feuchtigkeit"
    android:textSize="18sp"
    android:visibility="visible" />

<TextView
    android:id="@+id/temperatur"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_alignStart="@+id/ConnStat"
    android:layout_alignBottom="@+id/textView"
    android:layout_marginStart="41dp"
    android:layout_marginBottom="0dp"
    android:text="-"
    android:textSize="18sp"
    android:visibility="visible" />

<TextView
    android:id="@+id/feuchtigkeit"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_below="@+id/temperatur"
    android:layout_alignStart="@+id/temperatur"
    android:layout_alignLeft="@+id/temperatur"
    android:layout_alignBottom="@+id/textView"
    android:layout_marginStart="0dp"
    android:layout_marginLeft="0dp"
    android:layout_marginTop="14dp"
    android:layout_marginBottom="-39dp"
    android:text="-"
    android:textSize="18sp"
    android:visibility="visible" />

<TextView
    android:id="@+id/fuellstand"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_alignStart="@+id/ConnStat"
```

```
        android:layout_alignTop="@+id/textView5"
        android:layout_alignBottom="@+id/textView"
        android:layout_marginStart="41dp"
        android:layout_marginTop="1dp"
        android:layout_marginBottom="-77dp"
        android:text="-"
        android:textSize="18sp"
        android:visibility="visible" />

<TextView
    android:id="@+id/appdeveloper"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_alignParentStart="true"
    android:layout_alignParentLeft="true"
    android:layout_alignParentTop="true"
    android:layout_marginStart="123dp"
    android:layout_marginLeft="123dp"
    android:layout_marginTop="72dp"
    android:text="App Entwickler:"
    android:textSize="18sp"
    android:textStyle="bold"
    android:visibility="gone" />

<TextView
    android:id="@+id/nogru"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_below="@+id/appdeveloper"
    android:layout_alignParentStart="true"
    android:layout_marginStart="135dp"
    android:layout_marginTop="0dp"
    android:text="• Noah Gruber"
    android:visibility="gone" />

<TextView
    android:id="@+id/projectteam"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_alignParentStart="true"
    android:layout_alignParentTop="true"
    android:layout_marginStart="78dp"
    android:layout_marginTop="164dp"
    android:text="Team hinter dem Projekt"
    android:textSize="18sp"
    android:textStyle="bold"
    android:visibility="gone" />

<TextView
    android:id="@+id/pg"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_below="@+id/projectteam"
    android:layout_alignParentStart="true"
    android:layout_alignParentLeft="true"
    android:layout_marginStart="25dp"
    android:layout_marginLeft="25dp"
    android:layout_marginTop="22dp"
    android:text="• PL Philipp Gasser"
    android:visibility="gone" />
```

```
<TextView
    android:id="@+id/ss"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_below="@+id/pg"
    android:layout_alignStart="@+id/pg"
    android:layout_marginStart="0dp"
    android:layout_marginTop="10dp"
    android:text="• PL Stv. Stefan Stetina"
    android:visibility="gone" />

<TextView
    android:id="@+id/uo"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_below="@+id/ss"
    android:layout_alignStart="@+id/pg"
    android:layout_marginStart="-1dp"
    android:layout_marginTop="15dp"
    android:text="• PM Ulrich Obetzhauser"
    android:visibility="gone" />

<TextView
    android:id="@+id/ng"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_below="@+id/uo"
    android:layout_alignStart="@+id/pg"
    android:layout_marginStart="-1dp"
    android:layout_marginTop="12dp"
    android:text="• PM Noah Gruber"
    android:visibility="gone" />

<TextView
    android:id="@+id/projectmanagement"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_alignTop="@+id/pg"
    android:layout_alignParentEnd="true"
    android:layout_alignParentRight="true"
    android:layout_marginTop="0dp"
    android:layout_marginEnd="30dp"
    android:layout_marginRight="30dp"
    android:text="→ Projektmanagment"
    android:textStyle="italic"
    android:visibility="gone" />

<TextView
    android:id="@+id/konstruktion"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_alignStart="@+id/projectmanagement"
    android:layout_alignTop="@+id/ss"
    android:layout_marginStart="1dp"
    android:layout_marginTop="0dp"
    android:text="→ Konstruktion"
    android:textStyle="italic"
    android:visibility="gone" />
```

```
<TextView
    android:id="@+id/elektronik"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_below="@+id/ss"
    android:layout_alignStart="@+id/projectmanagement"
    android:layout_marginStart="0dp"
    android:layout_marginTop="15dp"
    android:text="→ Elektronik"
    android:textStyle="italic"
    android:visibility="gone" />

<TextView
    android:id="@+id/programmierung"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_alignStart="@+id/projectmanagement"
    android:layout_alignTop="@+id/ng"
    android:layout_marginStart="0dp"
    android:layout_marginTop="0dp"
    android:text="→ Programmierung"
    android:textStyle="italic"
    android:visibility="gone" />

<Button
    android:id="@+id/btnhome"
    android:layout_width="71dp"
    android:layout_height="61dp"
    android:layout_alignParentStart="true"
    android:layout_alignParentTop="true"
    android:background="@android:color/white"
    android:onClick="btnhomeClicked"
    android:text="△"
    android:textSize="18sp"
    android:visibility="gone" />

<Button
    android:id="@+id/btnabout"
    android:layout_width="383dp"
    android:layout_height="56dp"
    android:layout_alignStart="@+id/websitebtn"
    android:layout_alignLeft="@+id/websitebtn"
    android:layout_alignEnd="@+id/websitebtn"
    android:layout_alignRight="@+id/websitebtn"
    android:layout_alignParentBottom="true"
    android:layout_marginStart="-11dp"
    android:layout_marginLeft="-11dp"
    android:layout_marginEnd="9dp"
    android:layout_marginRight="9dp"
    android:layout_marginBottom="0dp"
    android:background="@android:color/white"
    android:onClick="btnaboutClicked"
    android:text="Wer/Was steckt dahinter?"
    android:textAllCaps="false"
    android:textSize="18sp"
    android:textStyle="italic"
    android:visibility="visible" />

</RelativeLayout>
```


Literaturverzeichnis

Bücher:

Nicht vorhanden!

Dissertationen:

Nicht vorhanden!

Abbildungen:

Nicht vorhanden!

Internet:

2.1.2 Projektmanagement Methode

<https://www.pinuts.de/blog/webstrategie/projektmanagement-wasserfall-gegen-scrum>

05.01.2019, 17:49

https://www.computerwoche.de/a/die-richtige-projektmanagement-methode-findest_3545546 05.01.2019, 18:53

2.1.3 Projektorganisation

<https://slideplayer.org/slide/5457130/> 07.01.2019, 19:53

2.1.4 Projektumfeldanalyse

<https://www.projektmagazin.de/glossarterm/projektumfeldanalyse>

08.01.2019, 18:36

2.1.5 Risikoanalyse

<https://projekte-leicht-gemacht.de/blog/pm-in-der-praxis/warum-risikomanagement-wichtig-ist/> 08.01.2019, 21:11

<https://projekte-leicht-gemacht.de/blog/pm-methoden-erklaert/strategien-zum-umgang-mit-risiken/> 08.01.2019, 21:43

2.1.6 Objekstrukturplan

<http://projektnachwuchs.de/projektstrukturplan-und-objektstrukturplan/>

11.01.2019, 16:32

2.1.7 Projektstrukturplan

<http://projektnachwuchs.de/projektstrukturplan-und-objektstrukturplan/>

11.01.2019, 18:13

<https://www.projektmanagementhandbuch.de/handbuch/projektplanung/projektstrukturplan/> 11.01.2019, 19:57

2.1.8 Meilensteinplan

<https://dieprojektmanager.com/meilensteinplan/> 13.01.2019, 19:46

Quellenverzeichnis

Abbildungen:

Abbildung 6: RobotUnits Profile

<https://www.robotunits.com/de/Webshop/Profiltechnik/shopitem/proilstange-50-x-50-mm-laenge-bis-6050-mm/141/> 05.01.2019, 16:58

Abbildung 7: 4-Kant Hohlprofile

<https://www.amazon.de/Quadratrohr-Stahlrohr-Hohlprofil-Vierkantrohr-30x30x3/dp/B01N3OZTGC> 05.01.2019, 16:39

Abbildung 40: Vertical Farming-Einheit auf B2B.Partcommunity

Abbildung 41: Additional Functions B2B.Partcommunity

Abbildung 42: Virtual Reality B2B.Partcommunity

Abbildung 43: Model Position B2B.Partcommunity

Abbildung 44: Model Shading B2B.Partcommunity

Abbildung 45: Shortcuts B2B.Partcommuniy

Abbildung 46: Environment B2B.Partcommunity

<http://www.vertical-farming.at/> 13.01.2019, 17:48

Abbildung 48: Aufbau des Folders

<https://www.saxoprint.de/blog/wickelfalz/> 17.01.2019, 18:35

Abbildung 94: Microcontroller

<https://www.microchip.com/wwwproducts/en/PIC16F1827> 12.02.2019, 13:33

Abbildung 96: Schottkydiode

https://upload.wikimedia.org/wikipedia/commons/8/8e/Diode-Schottky-EN_A-K.svg
12.02.2019, 14:03

Abbildung 97: Schaltregler

https://www.amazon.de/dp/B01GJ0SC2C/ref=asc_df_B01GJ0SC2C58464329/?tag=googshopde-21&creative=22434&creativeASIN=B01GJ0SC2C&linkCode=df0&hvadid=309088177091&hvpos=1o1&hvnetw=g&hvrand=13730461827298759729&hvpone=&hvptwo=&hvqmt=&hvdev=c&hvdvcmdl=&hvlocint=&hvlocphy=9062726&hvtargid=pla-305123447649&th=1&psc=1&tag=&ref=&adgrpid=65229878087&hvpone=&hvptwo=&hvadid=309088177091&hvpos=1o1&hvnetw=g&hvrand=13730461827298759729&hvqmt=&hvdev=c&hvdvcmdl=&hvlocint=&hvlocphy=9062726&hvtargid=pla-305123447649 12.02.2019, 14:05

Abbildung 99: Konstantstromquelle

https://www.amazon.de/MagiDeal-Treiber-700mA-Dimen-Konstantstrom/dp/B071FTF6L8/ref=mp_s_a_1_21?__mk_de_DE=%C3%85M%C3%85Z%C3%95%C3%91&qid=1536738589&sr=8-21&pi=AC_SX236_SY340_QL65&keywords=konstantstromquelle+700ma&dpPl=1&dplID=51ivASRRc8L&ref=plSrch#productDescription_secondary_view_div_1536777832810 12.02.2019, 14:15

Abbildung 101: Füllstandssensor

https://www.amazon.de/Schwimmerschalter-F%C3%BClstandsschalter-Pegelschalter-Niveauschalter-Sensor/dp/B06Y2139TX/ref=pd_rhf_dp_p_img_3?encoding=UTF8&psc=1&refRID=E7ED6MAB4PTRYZ3DVDF3 12.02.2019, 14:17

Abbildung 103: Status LED

https://eshop.hbc-radiomatic.com/out/pictures/generated/product/1/250_200_85/009-10-00025.jpg 12.02.2019, 14:21

Abbildung 105: DHT22 Sensor

https://hackster.imgix.net/uploads/attachments/332099/image_kxdoftWKju.png?auto=compress%2Cformat&w=1280&h=960&fit=max 12.02.2019, 14:35

Abbildung 107: Lüfter

https://www.amazon.de/ELUTENG-USB-120mm-Geh%C3%A4use%C3%BCber-Metallgrillschutz/dp/B06XCTHQG3/ref=sr_1_4?ie=UTF8&qid=1549977145&sr=8-4&keywords=l%C3%BCfter+5v+120 12.02.2019, 15:38

Abbildung 109: Pumpe

<https://www.conrad.at/de/barwig-0444-niedervolt-tauchpumpe-600-lh-6-m-539090.html> 12.02.2019, 15:38

Abbildung 111: Power LED, Abbildung 124: Power LED

https://ae01.alicdn.com/kf/HTB1FcU0KpXXXXOXVXXq6xFXXXo/3w-led-grow-light-chip-cob-full-spectrum-380-840nm-3w-DIY-led-grow-light-chip.jpg_220x220.jpg
12.02.2019, 15:42

Abbildung 112: Spektrum Power LED

<https://static.lightstrade.com/201607/201607081119594275.jpg> 12.02.2019, 15:51

Abbildung 113: HC-06

<https://images-na.ssl-images-amazon.com/images/I/51uc0d%2BukcL.jpg>
12.02.2019, 16:05

Abbildung 115: Netzteil

<https://www.amazon.de/Schaltnetzteil-Netzteil-200W-MeanWell-RSP-200-27/dp/B00UANKALQ> 17.02.2019, 18:16

Abbildung 130: Platine Vorderseite

<https://de.beta-layout.com/leiterplatten pcb-pool-konfigurator/> 19.02.2019, 13:28

Abbildung 131: Platine Rückseite

<https://de.beta-layout.com/leiterplatten pcb-pool-konfigurator/> 19.02.2019, 13:33

Abbildung 132: Platine 3D

<https://de.beta-layout.com/leiterplatten pcb-pool-konfigurator/> 19.02.2019, 13:38

Abbildung 135: Duo-LED

https://asset.conrad.com/media10/isa/160267/c1/-de/185000_BB_02_FB/vossloh-schwabe-cqx-95-led-mehrfarbig-rot-gruen-rund-5-mm-90-mcd-70-mcd-60-20-ma-2-v-21-v.jpg?x=520&y=520 21.02.2019, 18:32

Abbildung 136: LED Fassung

https://asset.conrad.com/media10/isa/160267/c1/-de/185957_BB_00_FB/signal-construct-smz1089-led-fassung-metall-passend-fuer-led-5-mm-schraubbefestigung.jpg?x=520&y=520 21.02.2019, 18:47

Abbildung 143: Verbildlichung der Messung des Füllstandes

<https://www.buerkert.at/de/Service-Support/Support/Glossar/Fuellstandsmessung>
09.03.2019, 16:54

Abbildung 156: Aufnahme vom booten des Programms „Android Studio“

Abbildung 167: Aufnahme vom booten des Programms „MPLAB X IDE“

Abbildung 171: Signalzustand „1“ und „0“

<https://www.slideshare.net/corradosantoro/pulse-width-modulation-signal-generation-with-mcus> 12.03.2019, 14:32

Abbildung 172: Auswirkung des Duty Cycles

<https://www.slideshare.net/corradosantoro/pulse-width-modulation-signal-generation-with-mcus> 12.03.2019, 14:32