

Report aus dem Projekt sensor Cube

von Ugur Turhal, Berkan Kurt & Silvan Lenzlinger

23. Januar 2022

Inhaltsverzeichnis

1	Einleitung	2
2	Aufteilung	2
3	Jigs/Werkzeuge	2
4	Sammeln der Daten	3
4.1	Erkenntnisse durch die Daten	4
5	Umsetzung	4
6	Resultat	7
7	Gewonnene Erkenntnisse	8
7.1	Nützlichkeit im Alltag	8
7.2	Das verwendete Material	9
7.3	Zeitmangement	11
7.4	Verbesserungswürdig	11
8	Danksagung	12

Abstract

Dieses Report Dokument ist im Rahmen der Vorlesung, Rechenarchitektur & vertrauenwürdiges Rechnen entstanden. Das Ziel war ein 4 x 4 x 4 Cube zu löten und diesen mit einem digitalen Luftfeuchtigkeits- & Temperatursensor auszustatten. Dieser soll anhand der Luftfeuchtigkeit und Temperatur, spezifische Muster anzeigen, welche uns verraten ob die Werte ideal sind für ein angenehmes Einschlafen. Alle Dokumente und Programme sind auffindbar unter:

🔗 <https://github.com/ugurtu/CatcProject>

1 Einleitung

Als wir im November uns als Team zusammengesetzt haben war die Projektidee sofort klar. Wir wollten mit dem Projekt etwas schaffen, welches uns im alltäglichen Leben auch über den Projektzeitraum hinaus begleiten soll. Nach reichlicher Überlegung entschieden wir uns für das Oberthema Schlaf. Wir haben den Schlaf gewählt, weil uns aufgefallen ist, dass wir alle drei unsere Einschlafphase als suboptimal beurteilen würden. Als wir uns über die Ursachen und Gründe für unseren suboptimalen Einschlafprozess ausgetauscht haben, kamen wir zum Entschluss, dass wir uns mit der optimalen Raumtemperatur nie auseinandersetzen werden. Diese ist vor allem im Winter wichtig, da die Raumtemperatur sich zu dieser Jahreszeit oftmals über der optimalen Raumtemperatur, bedingt durch das Überheizen, befindet. Im nachfolgenden Report werden wir darüber reden, wie wir ein Gadget erstellt haben, welches uns hilft, dieses Problem zu lösen.

2 Aufteilung

Ugur: Bestellung, funktionsfähiges „Circuit-Modell“ auf Tinkercad ohne DHT-22 Sensor, Programmierung des Python Codes zum Auslesen des Serial-Ports, C-Code Grundgerüst für den Cube und Jigs vorbereiten. Die Einzelnen Layers Löten. Siehe Kapitel 3. Den Anderen zeigen wie richtig gelötet wird.

Berkan: Kupferdraht richten mit Bohrmaschine und zuschneiden damit die Layers gleichmäßig aussehen. Schrumpfschläuche über die Kabel ziehen und verlöten, welche mit den Layers verbunden sind (Kathode), mit Hilfe von Ugur. Case aus Plexiglas herstellen mit Möglichkeit den Arduino in eine sichere Umgebung zu platzieren und einer Türmechanismus ähnlichen Oberfläche.

Silvan: 88 LEDs in Form bringen mit Hilfe der Jigs. Mithilfe von Berkan und Ugur alle Layers verlöten. Siehe Kapitel 6 so dass ein Cube entsteht. Siehe Kapitel 3. Code von Ugur verfeinern, Fehler erkennen und ein clean Code entwickeln und Methodennamen anpassen. Cube Nachlöten falls etwas unsauber verlötet wurde.

Alle: Bericht schreiben und den sensor Cube testen.

3 Jigs/Werkzeuge

Als Jigs werden Hilfsmittel bezeichnet, die beim Prozess des Fertigstellen das Leben vereinfachen.

- Ein Raster aus Holz, in welchem die LEDs reingestellt und verlötet wurden. Siehe Abbildung 1.
- Ein Holzbrett, mit welchem die Anode und Kathode in Form gebracht wurden. Siehe Abbildung 2
- Eine Bohrmaschine, welche den Kupferdraht gerade gewickelt hat.
- Ein Schraubstock, welches den Kupferdraht hielt.

Die folgenden Werkzeuge wurden für die Kommunikation und als Austausch verwendet.

- GitHub, sorgte für einen reibungslosen Austausch.
- WhatsApp, auf welchem die Meetings festgelegt wurden.
- Zoom, für Onlinemeetings, falls jemand nicht anwesend war.
- L^AT_EX für die Dokumentenerfassung.

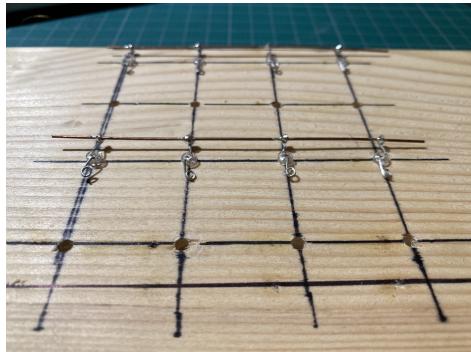


Abbildung 1: Jig für die Layers



Abbildung 2: Jig für die LEDs

4 Sammeln der Daten

Bevor das Projekt begann, wurde für die Sicherstellung des Projektes der Sensor anhand des Serialports ausgelesen. Mittels Python Programmen wurde die Raumtemperatur gesammelt und geplottet. Genaue Aufbauinformationen findet man im Dokument: **Aufbau.pdf**. Das Dokument befindet sich auf dem angegebenen GitHub Repository: <https://github.com/ugurtu/CatcProject/blob/main/Aufbau%20DHT%20Programme%20und%20Daten/Aufbau.pdf>.

Den spezifischen Python Code findet man unter: https://github.com/ugurtu/CatcProject/tree/main/SerialPort_python

Diesem Dokument aber liegen sowohl der Aufbau des Sensors und auch das Pythonprogramm bei. Wenn man Lüftet, sollte der Sensor in einer bestimmten Zeit eine Reaktion zeigen. Diese Reaktion verhält sich in Form von einem entsprechend angehenden LED Lämpchen. Dieses Testen und auslesen des Sensors diente für uns als Sicherstellung. Somit wurde uns klar, dass wir einen funktionstüchtigen Sensor besitzen, welchen wir für dieses Projekt ohne Probleme verwenden konnten.

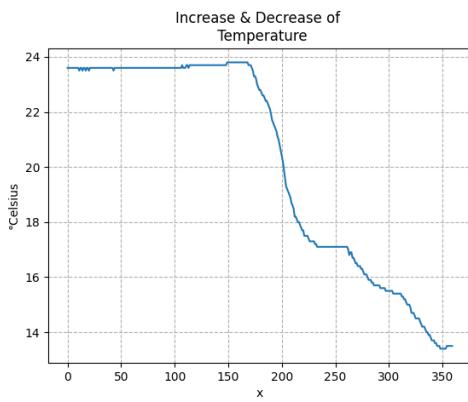


Abbildung 3: Plot des Temperatur Anstiegs und der Abfall

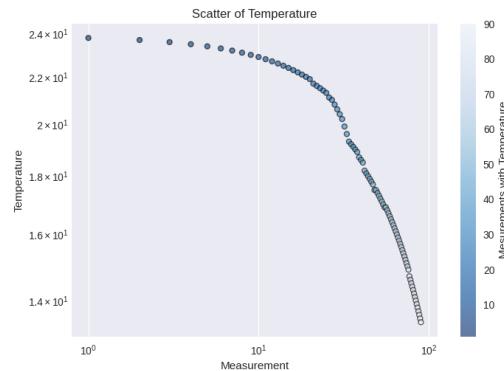


Abbildung 4: Scatter Plot des Temperatur Abfalls

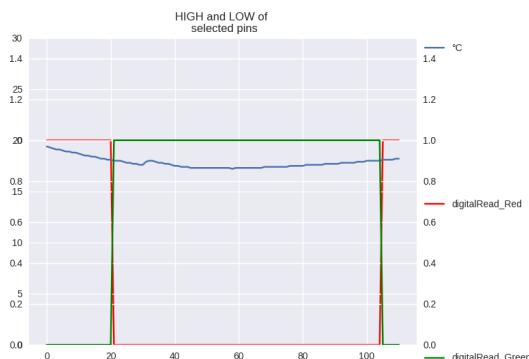


Abbildung 5: HIGH und LOW in einem ersten Test

4.1 Erkenntnisse durch die Daten

In der Abbildung 3 und 4 sieht man wie sich die Temperatur bei offenem Fenster zunehmend verringert. Somit hatten wir Gewissheit, dass der Sensor relativ schnell reagiert und für unser Projekt verwendbar ist. Die x - Achse in der Abbildung 3 ist die Zeit. Die Abbildung 4 ist Mesurement als 100 Messpunkte definiert. Alle doppelten Daten aus Abbildung 3 wurden ignoriert. In der Abbildung 5 sollen die zwei Farben, grün und rot, die Farben unserer LED's und den Zustand HIGH and LOW repräsentieren. Sobald die Temperatur unter 18° C betrug, schalteten sich die grünen LED's an (HIGH) und die roten LED's aus (LOW). Damit wurde angezeigt, dass die Bedingung für das Einschlafen optimal ist. Sobald das Fenster

wieder geschlossen und die Raumtemperatur damit wieder zugenommen hat, schalteten sich die grünen LED's aus (LOW) und die roten LED's gingen wieder an (HIGH).

5 Umsetzung

Unser Projekt hat in mehrheitlich sechs Phasen stattgefunden. Diese waren:

- (i) Einkauf
- (ii) Prototyp auf Tinkercad
- (iii) Einzelne Layers verlöten mit LEDs
- (iv) Layers zu einem Cube verlöten
- (v) Cube mit Arduino verbinden
- (vi) Case

Erster Punkt Wir haben einige Sachen bewusst doppelt gekauft. Falls etwas defekt gewesen wäre, hätten wir keine Zeitengpässe gehabt.

Zweiter Punkt Dieses folgende umgesetzte „Circuit-Modell“ wurde von der Grundstruktur so übernommen, allerdings soll angemerkt werden, dass dieses Modell mit einem Arudino UNO erstellt wurde, da kein Arduino ATmega auf Tinkercad zur Verfügung stand. Siehe: Abbildung 6

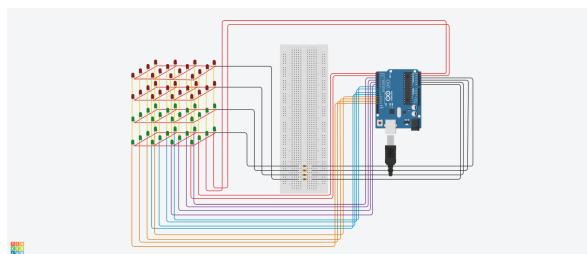


Abbildung 6: LED Cube Circuit mit einem Arduino UNO

Dritter Punkt Die Layers, welche in der Abbildung 7 zu sehen sind leiten den Strom. Es gibt vier davon.

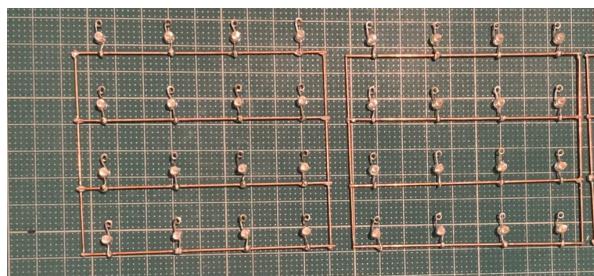


Abbildung 7: Anode mit dem Kupferdraht verlötet, zu sehen zwei Layers

Vierter Punkt Die Layers, werden durch vertikale Kupferdrähte miteinander verbunden, dies sorgt dafür dass die Kathode verbunden ist. Danach wurde der Cube mit der Europlatine verlötet. Damit jede Ebene, durch den Mikrocontroller Strom bekommt, wurden zusätzlich vier Kupferdrähte jeweils treppenförmig

mit den vier Layers verbunden. Jeder Layer ist jeweils mit einem $100\ \Omega$ Widerstand verlötet. Das Datasheet für die LEDs sieht ein Peak Forward Current von 60 mA vor. Daher wurden $100\ \Omega$ verwendet.¹

$$U = R \cdot I \quad (1)$$

$$U := 5V, I := 0.060A \quad (2)$$

$$\frac{5V}{0.060A} = 83.333\Omega \quad (3)$$

$$\approx 100\Omega \quad (4)$$

Danach wurde auf der linken Seite des sensor Cubes der DHT-22 verlötet. Als letzter Schritt wurden die Kupferdrähte der Kathode und die Pins des Sensors mit Kabeln verlötet, isoliert und mit den Stifteleisten verlötet. Siehe Abbildung 8.

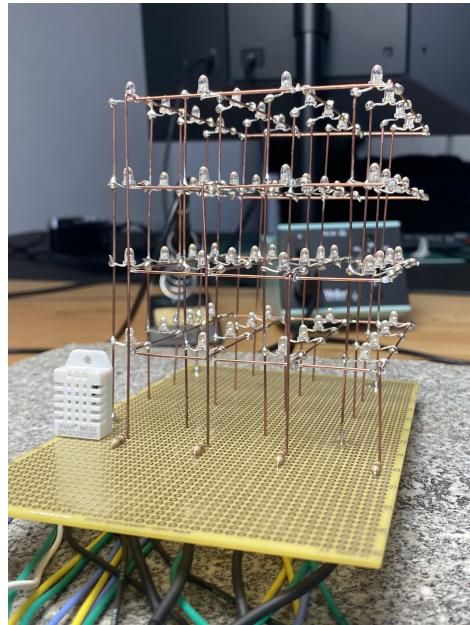


Abbildung 8: sensor Cube gelötet

¹<https://www.reichelt.com/de/en/led-3-mm-leaded-green-125-mcd-25--evl-1224-10sygc-p230786.html>

Fünfter Punkt Der 5 Schritt war die entscheidende Phase, da alles fertig war wurden die Pins verbunden. hier eine Übersicht:

1. A0 → versorgt Layer 0 mit Strom.
2. A1 → versorgt Layer 1 mit Strom.
3. A2 → versorgt Layer 2 mit Strom.
4. A3 → versorgt Layer 3 mit Strom.

DHT-22 Pins

1. DHT-22 Anode → 5V.
2. DHT-22 Kathode → GND.
3. DHT-22 Digital-Pin → 50.

Die Kathode der LEDs sind wie in Abbildung 9 aufgeteilt. Die Pins werden durch jeweils zwei For-Loops als OUTPUT initialisiert. Einmal für die Layers und einmal für die „Spalten“. Dies hat den Vorteil, dass wir nun durch dieses System jede LED einzeln ansteuern.

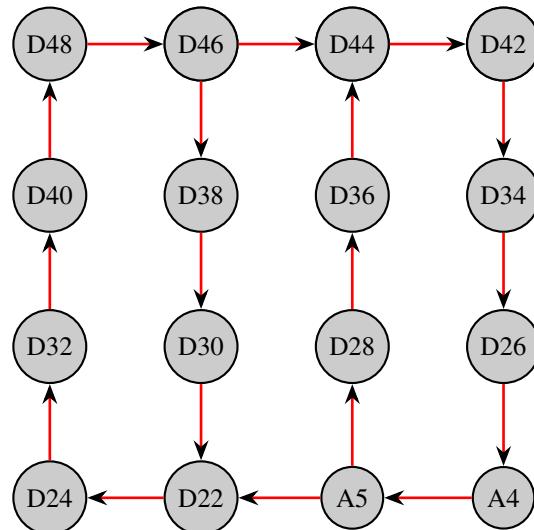


Abbildung 9: Obere Ansicht des sensor Cubes mit den Verbundenen Pins und des Stromflusses

Sechster Punkt Als der sensor Cube fertig war und wir somit wussten, dass wir das Ziel erreicht haben, haben wir uns auf die Suche nach einem passenden Case gemacht. In einem Meeting wurde die Entscheidung getroffen, ein Case aus Plexiglas herzustellen. In verschiedenen Geschäften suchte man das passende Plexiglas, nach Scharnieren und Klebstoffen. Das Plexiglas wurde mit den passenden Massen geschnitten und schlussendlich zu einem Rechteck zusammengeklebt. Anschliessend wurden in die Seite des Case, sowie in die Europlatine zwei Schaniere montiert. Dies sollte uns ermöglichen, einen Türähnlichen Mechanismus zu haben. sind die Kabel und das Arduino in Sicherheit. Ausserdem wurde ein Loch in die Seite des Case gefräst, um die Stromversorgung zu gewährleisten Ebenfalls können Reparaturen vorgenommen werden, falls ein Kabel defekt ist. Für das Ergebnis siehe Abbildung 10.

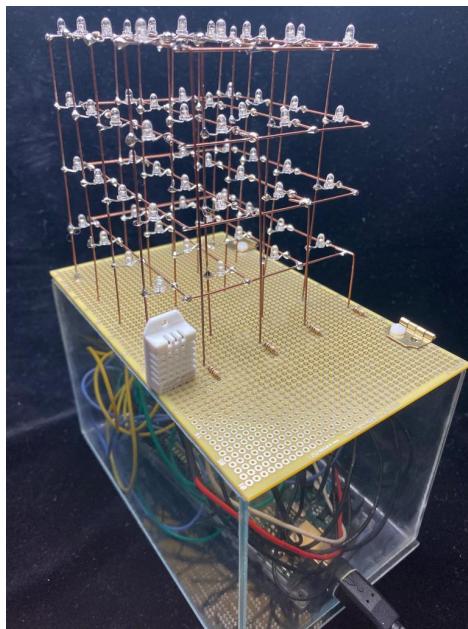


Abbildung 10: Cube mit Case

6 Resultat

Der Cube ist unserer Meinung nach genau und symmetrisch gelötet. Auch wenn es noch Raum nach oben gibt, sind wir mit dem Endprodukt sehr zufrieden. Zudem lässt das durchsichtige Case einen direkten Blick auf das Innenleben des Cubes zu. Da für die Verdrahtung sehr wenig Platz war, haben wir Schrumpfschläuche an den Enden der Kabel benutzt. Diese verhindern einen allfälligen Kurzschluss. Danach hatten wir zwischen dem Arduino und allen LEDs sowie dem Sensor eine sichere Verbindung. In Abhängigkeit von den Messdaten kann der Cube die folgenden fünf Muster wiedergeben:

- Temperatur: $< 15^\circ \text{ C}$ Raumtemperatur
 - Dynamisches/pulsierendes Feedback
- Temperatur: $\geq 15^\circ \text{ C}$ Raumtemperatur $\leq 18^\circ \text{ C}$
 - Feuchtigkeit: $< 30\%$ Luftfeuchtigkeit im Raum
 - * Dynamisches Regenmuster von unten nach oben
 - Feuchtigkeit: $\geq 30\%$ Luftfeuchtigkeit im Raum $\leq 60\%$
 - * Die zwei grünen Levels leuchten statisch
 - Feuchtigkeit: $> 60\%$ Luftfeuchtigkeit im Raum
 - * Dynamisches Regenmuster von oben nach unten
- Temperatur: $> 18^\circ \text{ C}$ Raumtemperatur
 - Dynamisches/ propellerartiges Leuchtmuster

Durch die visuelle Wiedergabe der LEDs hat man Zeitnah ein Feedback über das Raumklima. Die dynamischen Leuchtmuster lassen sich schnell interpretieren und es kann entsprechend reagiert werden. Nach Anpassungen des Raumklimas leuchtet das statische Muster in grün. Danach steht einer guten Einschlafphase „nichts“ mehr im Weg.

7 Gewonnene Erkenntnisse

Dieser Abschnitt besteht nicht nur aus einer Erkenntnis sondern aus mehreren Erkenntnissen. Deshalb sollen in den nachfolgenden Abschnitten folgende Erkenntnisse angesprochen werden:

1. Die Fähigkeit beziehungsweise die Nützlichkeit des sensor Cubes im Alltag,
2. Erkenntnisse über das verwendete Material,
3. Zeitmanagement bis zur Abgabe,
4. Verbesserungen für ein nächstes Projekt.

7.1 Nützlichkeit im Alltag

Das unser Cube im Stande ist auf unterschiedlichliche Temperaturen und eine veränderte Luftfeuchtigkeit zu reagieren ist ausser Frage. Dies wurde in einem Praxistest bei allen drei Gruppenmitgliedern zu Hause getestet. Das bei einer spezifischen Luftfeuchtigkeit & Temperatur spezifische festgelegte Muster als Feedback angezeigt werden sollen ist vorhanden.

Wir sind per E-Mail mit Schlafprofessor Prof. Dr. Björn Rasch in Kontakt getreten, um zusätzliche Informationen über die optimale Schlaftemperatur und Luftfeuchtigkeit zu erhalten. Aus einer E-Mail von 03.01.22 wurden drei Punkte deutlich.

1. Der Schlaf steht in Korrelation mit der psychologischen Verfassung des Benutzers des Kubus.
2. Die Temperatur für den optimalen Schlaf hängt von dem Microklima unter der Bettdecke ab.
3. Verschiedene Studien kommen zum Schluss, dass 30-50% oder 30-60% Luftfeuchtigkeit optimal sind.

Zu Punkt eins soll keine Stellung genommen werden, da wir über keine fundierte psychologische Ausbildung verfügen, selbst wenn das der Fall wäre, würde dies den Rahmen des Reports sprengen.

Zu Punkt zwei sind wir der gleichen Meinung, dass wenn eine Bettdecke zu warm gibt, dies sich auf das Einschlafen negativ auswirkt. Trotzdem hat der Cube das Einschlafen bei allen Gruppenmitgliedern positiv beeinflusst. Somit sehen wir einen direkten Zusammenhang zwischen dem Raumklima und der Qualität des Einschlafens. Wir vermuten, dass durch die zusätzliche Wahl einer passenden Bettdecke die Qualität des Einschlafens weiter verbessert wird.

Zu Punkt drei, dass die Luftfeuchtigkeit stimmen muss für den Schlaf/Einschlaf, vor allem in einer kalten Jahreszeit, da kalte Luft relativ trocken ist, kann unser Kubus auch evaluieren.

Fazit Die Benutzung des sensor Cubes kann hilfreich sein beim Einschlafen. Allen drei Gruppenmitgliedern hat der sensor Cube geholfen.

7.2 Das verwendete Material

Folgende Komponenten wurden verwendet:

1. Kupferdraht
2. Kabel mit einer Litze
3. Arduino UNO 328P → Arduino ATMega 2560
4. Lötzinn
5. Flussmittel
6. Schrumpfschlauch
7. Euro-Platine aus Epoxidharz
8. Plexiglas

9. Löt-Station
10. DHT-22
11. LEDs
12. Scharniere
13. Sekundenkleber
14. USB-Stecker - Typ B
15. Multimeter
16. Heissluftföhn
17. Bohrmaschine

In der Tabelle 1 auf der nächsten Seite sollen nur die Teile aufgelistet werden, welche Probleme bereitet haben. Die Komponenten die nicht in der Tabelle sind, haben **keine** Probleme verursacht.

Tabelle 1: Probleme und Lösungen

Komponente	Problem	Lösung
Kupferdraht	Der Kupferdraht, welcher bestellt wurde, war beschichtet und ohne Lötzinn war dieser nicht Leitfähig. Die Schicht musste geschmolzen werden oder abgeschliffen werden	Neuer Kupferdraht wurde erworben
Arduino	Da wir alle LEDs einzeln ansteuern wollten aber auch den Sensor verbauen mussten, waren wir gezwungen mit i ² c zu arbeiten oder mehr Pins haben	Ugur entschied sich im Sinne der Gruppe auf einen Arduino AT-Mega 2560 umzustellen
Lötzinn	Der Lötzinn welcher mitgegeben wurde, war relativ alt und ergab beim ersten Löten Zinnkugeln, welche optisch nicht ansprechend waren	Flussmittel wurde erworben
Schrumpfschlauch	Der Schrumpfschlauch der erworben wurde, hatte einen zu kleinen Durchmesser für die Verbindungskabel	Schrumpfschlauch mit 3mm, welcher auf eine grössere von 1mm schrumpfte wurde erworben
Euro-Platine aus Epoxidharz	Bis heute sind wir in der Gruppe nicht einig, ob wir die Euro-Platine anders herum hätten verwenden sollen, da der Lötzinn nicht haftete	-
Plexiglas	Wir wollten nicht nur ein Platz für unseren Arduino, sondern auch ein Case für den Cube aus Plexiglas, leider reflektierte das Material die Lichter der LEDs zu stark. Somit war nicht eindeutig erkennbar, welche LED rot oder Grün leuchtete.	Verzicht den Cube in ein Plexiglas Gehäuse zu stellen.
Löt-Station	Die Spitze des Lötkolbens, welcher zur Verfügung gestellt wurde, war oxidiert. Somit war es nicht möglich mit dieser Lötstation zu löten, entweder musste die Lötspitze gereinigt werden oder eine andere Lötstation musste her.	Ugurs Arbeitgeber verlieh eine professionelle Löt-Station aus.
DHT-22	Ein Sensor der mitgeliefert wurde, war defekt. Da wir aber aus Sicherheitsgründen zwei bestellt haben, hatten wir ein Backup	Der zusätzliche Sensor wurde verwendet.
Sekundenkleber	Die Kombination aus Sekundenkleber und Plexiglas verträgt sich nicht gut. Beim ausprobieren hat Berkan festgestellt das Spuren des Klebers sichtbar werden	Berkan hat trotz der Einschränkung, versucht so sauber wie möglich zu arbeiten
USB-Stecker - Typ B	Der USB-Stecker Typ B war relativ dick. Da wir keine Werkstatt haben, standen uns nur wenige Aufsätze zur Verfügung. Das Loch, welches für das Kabel gedacht ist, ist ein bisschen Eng.	-

7.3 Zeitmanagement

Da wir einen strikten Gantt-Chart befolgt haben, wurde alles nach Zeitplan erledigt. Wir waren mit dem sensor Cube am 31. Dezember 2021 fertig. Somit konnten wir viel testen und uns zusätzliche Gedanken darüber machen, wie wir die Luftfeuchtigkeit zusätzlich in unser Projekt integrieren können. Außerdem hatten wir genug Zeit um Ästhetische Aspekte, wie die LED Feedbacks und das Case fertigzustellen. Wir können behaupten, dass wir die Verwendung des Gantt-Charts nochmal in einem ähnlichen Projekt einbinden würden, da dieses den Zeitplan relativ gut eingeteilt hat und wir somit immer wieder überprüfen konnten, wo wir stehen.

7.4 Verbesserungswürdig

Wir kamen zum Entschluss, dass wir in einem nächsten Projekt mehr Zeit für die Planung investieren würden. Diese zusätzliche Zeit hätte uns viel Mühe und Zeit erspart. Viele Erkenntnisse kamen zu spät, beispielsweise haben wir das Kupfergestell gelötet ohne zu überlegen, wie wir es in die Euro-Platine reinbekommen sollen. Dieses Problem wäre einfacher zu lösen gewesen, wenn wir das Grundgerüst direkt in die Europlatine integriert hätten. Das gleiche Problem hatten wir auch mit unserem Case, welches sich schwerer als gedacht erwies, da wir schon ein fertigen Kubus hatten welchen wir nun mit einem Case austatten wollten. Zudem ist es auch wichtig sich vorher schon Gedanken zu machen ob uns genug Pins zur Verfügung stehen, dass hat uns zwei Tage gekostet, da die Lieferung bisschen verzögert war.

8 Danksagung

Folgenden Personen/ dem Arbeitgeber soll gedankt werden:

Frau Sarah-Lia Andrea Wotke, Sie hat Ugur eine Einführung in das Löten gegeben, er konnte sein Wissen mit Berkan und Silvan teilen. Zu dem hat Sie wichtige Tipps gegeben, wie wir die Kabel isolieren sollten.

Herr Prof. Dr. Björn Rasch an der Universität Fribourg, er hat dem Projekt eine andere Sichtweise gegeben.

Herr Ilir Ferati Welcher wichtige Tipps bezüglich des Gehäusebaus gegeben hat.

Als letztes wollen wir uns bei der **Abteilung Technik des Telebasels** bedanken. Diese hat uns das nötige Werkzeug zur Verfügung gestellt, welche den Zeitplan sichergestellt haben.