

ENDBERICHT PROJEKT 1684 IoT AIRCLEAN

INHALT

1	Einleitung	4
1.1	Was ist IoT AirClean?	5
1.2	Zielgruppe.....	6
1.2.1	Zielgruppe Bildungswesen	6
1.2.2	Zielgruppe Unternehmen.....	6
1.3	Problem.....	8
1.4	Lösung	9
2	Meilensteine	10
2.1	Meilenstein 1 - Projektstart.....	10
2.2	Meilenstein 2 – Erster Sensorknoten	12
2.3	Meilenstein 3 – Standalone Gerät.....	14
2.4	Meilenstein 4 – Anbindung zentraler Server	17
2.5	Meilenstein 5 – Feldtest bei Privatperson.....	19
2.6	Meilenstein 6 – Feldtest in Schulungsraum.....	20
2.7	Meilenstein 7 – Fehleranalysen	22
2.8	Meilenstein 8 – Entwicklung der Basisstation	24
2.9	Meilenstein 9 – Mehrere SensorBoxen	26
2.10	Meilenstein 10 – Projektabschluss.....	27
3	Arbeitspakete.....	28
3.1	Projektmanagement	28
3.2	Recherche	29
3.3	Evaluierung.....	31

3.3.1	Sensoren	31
3.3.2	Hardware	31
3.3.3	Gehäuse	32
3.3.4	Programmiersprachen.....	34
3.4	Softwareentwicklung	36
3.4.1	Datenkommunikation	36
3.4.2	Darstellung der Daten.....	36
3.4.3	Die Systemarchitektur	37
3.5	Datenverwaltung.....	38
3.6	Tests.....	40
3.7	Bugfixing	41
4	Zusammenfassung	43
4.1	Projektergebnisse	43
4.2	Geplante Weiterentwicklung	44
4.3	Persönliche Meinung	45
5	Anhang	47

ABBILDUNGSVERZEICHNIS

Abbildung 1: Paperprototyp SensorBox	13
Abbildung 2: Erster Prototyp einer mobilen Teststation.....	14
Abbildung 3: Excel Auswertung der CSV Sensor Aufzeichnungen.....	15
Abbildung 4: Push-Benachrichtigung	18
Abbildung 5: Schulungsraum WIFI Steiermark Graz	20
Abbildung 6: Mobile Teststation mit Mobile Internet	20
Abbildung 7: Mobile Teststation und die SensorBox	21
Abbildung 8: Raspberry Pi mit Touchscreen und Web Interface	22
Abbildung 9: Mobile App für Serversuche	23
Abbildung 10: 3D Druck - Server mit Display	24
Abbildung 11: 3D Druck - IoT AirClean Cube.....	25
Abbildung 12: 3D Druck - 1. Entwurf - viel zu klein	32
Abbildung 13: 3D Druck - 1. Entwurf - Detailansicht	33
Abbildung 14: 3D Druck - weiterer Entwurf - größer jedoch immer noch zu klein.....	33
Abbildung 15: 3D Druck - 1.SensorBox - kompakte Größe und 1. SensorBox	34
Abbildung 16: 3D Druck - finale Version der SensorBox	40
Abbildung 17: Usability Erweiterung einer RGB LED der SensorBox.....	41
Abbildung 18: Aufmerksamkeitskurve	46

1 EINLEITUNG

Dieser Endbericht bietet eine Projektübersicht zum **Netidee Projekt 1684 IoT AirClean**. Wir beantworten unter anderem die Fragen „Was geschah in diesem Projekt?“, „Was wurde entwickelt?“ und „Was wurde aus der Idee aus dem Antrag?“.

Wir geben einen **Gesamtüberblick** beginnend vom Projektstart, über die ersten Evaluierungen unserer Rechercheergebnisse, bis hin zu der Entwicklung einer Datenaufzeichnung, Datenauswertung und Datenvisualisierung von zwei Sensoren zur Messung von Temperatur, Luftfeuchtigkeit und CO2 Gehalt der Raumluft. Tests, Usability-Evaluierungen und Bugfixes folgten nach der Entwicklung einer Anwendung bestehend aus Hardware und Software.

In den Beschreibungen der **Meilensteine** fassen wir Informationen der wichtigen Erkenntnisse und Ergebnisse zum Erreichen dieser Meilensteine zusammen. Diese spannen einen roten Faden durch das gesamte Projekt.

In den Kapiteln der einzelnen **Arbeitspakete** gehen wir näher ins Detail, um Projektergebnisse und Tätigkeiten näher zu präsentieren.

Der Projektendbericht endet mit einer **Zusammenfassung** der Projektergebnisse und einer Übersicht zu den diversen Onlinequellen, welche unsere Projektergebnisse frei verfügbar laut Lizenzvereinbarungen bereitstellen. Wir geben einen Ausblick auf die weitere geplante Entwicklung des Projekts und geben ein persönliches Statement zum Projekt ab.

Zusätzliche Dokumente dienen zur Erweiterung des Endberichts und zur Dokumentation des Projekts. Es wird im Endbericht auf die jeweiligen Dokumente als **Anhang** verwiesen und sind im Kapitel Anhang gelistet.

1.1 WAS IST IoT AIRCLEAN?

IoT AirClean ist ein vollständiges System (bestehend aus Hardware sowie der zugehörigen Software) zur Messung und Analyse der Raumluft. Eine SensorBox führt eine fortlaufende CO₂-, Temperatur- sowie Luftfeuchtheitsmessung des Raums durch. Diese SensorBox sendet die Aufzeichnung an einen lokalen Server. Die gesammelten Daten können über eine Webanwendung mittels grafischer Oberfläche ausgewertet werden.

Ein Unternehmen oder eine Bildungseinrichtung kann beispielsweise einen Server im Gebäude betreiben und pro Raum, Büro oder Klasse eine Sensorbox aufstellen.

Die grafische Oberfläche unterstützt Sie dabei, Ihren Arbeitsalltag bzw. Unterricht so zu gestalten, dass Sie in einer Umgebung mit idealen Bedingungen arbeiten beziehungsweise lernen können. IoT AirClean informiert Sie dabei über geeignete Zeitpunkte zum Lüften und ist die **ERINNERUNG** zum regelmäßigen und gezielten **LÜFTEN**.

Angst um Ihre Daten? Ihre Daten bleiben stets bei Ihnen! IoT AirClean benötigt nicht zwingend eine Internetverbindung. Eine Anbindung an einen über das Internet erreichbaren IoT AirClean Server bietet weitere Vorteile, das System kann jedoch auch nur innerhalb Ihres eigenen Netzwerk betrieben werden.

IoT AirClean

- führt **Raumklima Messungen** durch.
- **analysiert** dabei insbesondere den **CO₂-Gehalt** der Luft.
- **visualisiert** den zeitlichen Verlauf des Raumklimas.
- **erinnert** sie daran, zum passenden Zeitpunkt **zu lüften**.

Die wichtigste Funktion auf den Punkt gebracht:

Der Benutzer wird über passende Lüftungszeitpunkte informiert.

1.2 ZIELGRUPPE

IoT AirClean hat zwei Zielgruppen, Bildungswesen und Unternehmen. Beide Zielgruppen benötigen möglichst optimale Umgebungen, um Arbeit und Lernen effizient zu gestalten. IoT AirClean unterstützt diese Zielgruppen in der Analyse und Überwachung des aktuellen Raumklimas.

1.2.1 ZIELGRUPPE BILDUNGSWESEN

Durch die Datenanalyse im Unterricht, in Seminaren, in Workshops, etc. ist es möglich in der Unterrichtsgestaltung auf die jeweilige Lernumgebung noch genauer einzugehen. Eindrücke wie zunehmende Wärme, stickige Luft (die oftmals nicht wahrgenommen wird), weniger aktive Teilnehmer, und vieles mehr werden oftmals selbst von geübten Trainern ignoriert.

Durch die Datenerfassung von CO₂, Temperatur und Luftfeuchtigkeit kann ein schneller Einblick in das mögliche Potential zur Verbesserung und Optimierung gewonnen werden. Zusätzlich kann sogar ein Überblick über noch nicht ausgeschöpfte Möglichkeiten einer Lernumgebung gegeben werden. IoT AirClean kann zusätzlich zur Unterrichtsanalyse – speziell zur zeitlichen Analyse der Lernumgebung – herangezogen werden.

Lehrbeauftragte können gezielter in notwendige Pausen einlenken, um die Teilnehmer wieder mit einer optimalen Lernumgebung zu versorgen. Eine kurze 5 Minuten Pause und gezieltes Lüften kann bereits den Raum für die nächsten 30-45 Minuten mit Sauerstoff versorgen. 5 Minuten können über innerhalb der nächsten Stunde entscheiden, ob die Teilnehmer sich in einer möglichst optimalen Lernumgebung befinden, oder ob sie bereits durch die Lernumgebung das mögliche Potential, dem Unterricht zu folgen, nicht mehr ausschöpfen können.

1.2.2 ZIELGRUPPE UNTERNEHMEN

Besprechungen und Büros können mit IoT AirClean unterstützt werden. Eine „schlechte“ Arbeitsumgebung kann unterschiedlich hervorgerufen werden:

- zu warm / zu kalt, abhängig von einer ungeeigneten Temperatur
- zu stickig / schlecht belüftet, was auf einen zu hohen CO₂-Gehalt schließen lässt
- sowie weitere Faktoren, die wir bisher noch nicht erfassen können

Durch die Datenerfassung von IoT AirClean können die Personen im Unternehmen sensibilisiert werden, bestimmte Abläufe im Arbeitsalltag zu integrieren. Das regelmäßige Lüften wird oft vernachlässigt. IoT AirClean unterstützt AnwenderInnen dabei, derartige Routinen in den Arbeitsalltag zu integrieren. Die regelmäßigen Benachrichtigungen beim Erreichen von definierten CO₂ Schwellenwerten signalisieren AnwenderInnen die Notwendigkeit, den Raum durch gezieltes Lüften wieder mit Sauerstoff zu versorgen. Diese wichtigen Kleinpausen in der Arbeit „zwingen“ AnwenderInnen sich dabei zu bewegen und fördert die ebenso vernachlässigte Bewegung während des Arbeitsalltags.

Ein besseres Wohlbefinden in der Arbeit fördert auch die Motivation der Mitarbeit. Viele positive Nebeneffekte werden durch die Hauptfunktionen von IoT AirClean hervorgerufen.

1.3 PROBLEM

IoT AirClean konzentriert sich auf ein uns allen bekanntes Problem: Es bekämpft einer Person. Wir neigen dazu bestimmte Tätigkeiten im Alltag zu ignorieren beziehungsweise nehmen die Notwendigkeit bestimmter Handlungen nicht bewusst wahr. IoT AirClean beschäftigt sich mit der Problemlösung die Raumluft durch gezielteres Lüften zu verbessern. Dabei bedient es sich nicht eines ausgeklügelten Filter- oder Lüftungssystems, sondern nützt die Menschen selbst zur Lüftung.

Versetzen Sie sich in folgende Situationen:

- Schulklasse, alle Fenster sind geschlossen, und der Unterricht läuft bereits 37 Minuten.
- Büroalltag, Sie sind seit über 30 Minuten in ihre Arbeit vertieft.
- Seminar oder Workshop mit mehreren Teilnehmern, der Vortrag läuft bereits seit 53 Minuten.

Eine dieser Situationen haben Sie bestimmt schon einmal erlebt.

Nun stellen sich weiteres vor:

- Schulklasse, Sie verlassen die Klasse und kommen nach 5 Minuten wieder.
- Büroalltag, Sie arbeiten noch längere Zeit weiter, müssen zu einem Kollegen und kommen nach 10 Minuten wieder.
- Seminar oder Workshop, Sie erhalten einen Anruf, verlassen den Raum und kommen nach dem Telefonat wieder.

Überlegen Sie sich was Sie in solchen Situationen meistens erlebt haben.

Wir lösen auf: In allen drei Situationen erleben Sie oftmals ein AHA! Erlebnis der besonderen Art. Schlechte Luft in diversen Ausprägungen. Warum haben Sie das nicht vorher bemerkt. Die Verschlechterung der aktuellen Raumluft erfolgt nicht plötzlich, so wie Sie es im Moment des Wiederkehrens empfinden, sondern auf „längere“ Zeit. Doch leider ist diese „längere“ Zeit in Wirklichkeit trotzdem sehr gering. Wann wäre nun ein guter Zeitpunkt um die Raumluft mit einem einfachen Lüften wiederaufzufrischen?

1.4 LÖSUNG

Schlechte Raumlufth ist das Ergebnis aus verschiedenen Faktoren, sodass es keine allgemein gültige Regel zum Lüften geben kann. Stündliches Lüften in Umgebungen mit mehreren Personen ist sicher kein Nachteil. Ebenso ist stündliches Lüften in Arbeitsumgebungen sowie Lernumgebungen besser als ein ganzer Tag ohne Bewegung und ohne frischer Luft. IoT AirClean löst mittels IoT (Internet of Things) Sensoren und Benachrichtigungen sowie Analysemöglichkeiten dieses individuelle Problem.

Regelmäßige Abläufe in unserem Alltag übernehmen wir erst zu einer routinierten Handlung, wenn wir uns diese durch regelmäßige Wiederholungen über einen längeren Zeitraum auch tatsächlich selbst aneignen. Sie können sich auch einen Wecker stellen und jede halbe Stunde zum Fenster gehen um den Raum zu lüften. Das fördert die Bewegung im Büroalltag und unterstützt eine bessere Raumlufth. Der Wecker wird jedoch weder Ihre Umgebung noch die Anzahl der Personen im Raum berücksichtigen.

IoT AirClean ist nicht nur eine simple Erinnerung für „Lüfte jede Stunde“ sondern analysiert die aktuellen Gegebenheiten, benachrichtigt AnwenderInnen und gibt Informationen über einen möglichen optimalen Zeitpunkt zum Lüften. Damit wird das Lüften an Sie und Ihre Umgebung angepasst.

2 MEILENSTEINE

2.1 MEILENSTEIN 1 - PROJEKTSTART

Der erfolgreiche **Projektstart** des eingereichten Projekts war als Meilenstein 1 definiert. Der vorgelegte Vertrag wurde durchgesehen, unterschrieben und der **Internet Privatstiftung Austria** (IPA) übermittelt. Der dazugehörige Projektplan wurde erstellt, und von der IPA abgenommen.

Die **Projektwebsite**¹ wurde zum Projektstart in Betrieb genommen, um diesem Projekt eine fortlaufende Präsenz im Internet zur Verfügung zu stellen. Die Website wurde während des Projektes laufend befüllt sowie im Design überarbeitet. Im Rahmen des Projekts wurde das Hosting für 2 Jahre berücksichtigt, sodass nach offiziellen Projektende die Website weiterhin verfügbar bleibt.

Durch die positive Projektentwicklung wird **mea IT Services** die Projektwebsite voraussichtlich auch nach diesen 2 Jahren weiter finanzieren und am Projekt weiterarbeiten. Die Weitergabe der **Projektergebnisse** erfolgt in erster Linie über die Website. Dies geschieht durch Bereitstellung der **Dokumentation** und zielgruppengerechter **Anleitungen** zur **Anwendung und Entwicklung**. Zusätzlich ist entwickelte **Source Code** auf einem GitHub Repository² verfügbar. Das GitHub Repository wird von bereits installierten Umgebungen als Quelle für Updates verwendet. GitHub Repositories sind kostenlos und stehen nach Projektende weiterhin zur Verfügung.

Die Lizenz und der Ort der öffentlichen Bereitstellung wurden in Rücksprache mit IPA geklärt. Die Lizenz war in der Einreichphase restriktiver gesetzt und wurde nach Projektzusage umgeändert. Das Projekt wird nun unter der Lizenz **GNU GPLv3**³ sowie **CC-BY Sharealike-3.0 AT**⁴ entwickelt und weitergegeben.

¹ <http://www.iot-airclean.at>

² <https://github.com/michaelulm/iotairclean>

³ <https://www.gnu.org/licenses/gpl-3.0.de.html>

⁴ <https://creativecommons.org/licenses/by-sa/3.0/at/>

Damit dürfen Personen den Source Code für **private sowie kommerzielle Zwecke teilen und weiterbearbeiten**. Das darf uneingeschränkt gemacht werden, sofern folgende Bedingungen eingehalten werden: Die **Namensnennung** – also Urheberangaben – muss **in angemessener Art und Weise** erfolgen, sowie die **daraus entstandenen Entwicklungen müssen unter gleichen Bedingungen weitergegeben werden**. Dadurch bleibt der Kern und die Weiterentwicklungen des Projekt IoT AirClean **Open Source**.

Durch den 1. Meilenstein ist es der Online Community möglich, dem Projekt zu folgen, sowie Ideen und Anregungen über GitHub oder über die Website zu kommunizieren.

2.2 MEILENSTEIN 2 – ERSTER SENSORKNOTEN

Der **erste Sensorknoten** (Gerät mit Datenaufzeichnung) wurde laut Projektplanung zeitgerecht in Betrieb genommen. Die Entwicklung einer einfachen Aufzeichnung von aktuellen Sensorwerten war für das Gesamtprojekt ein wichtiger Schritt.

Erste **Tests** konnten ermitteln, wie sich die **IoT Sensorhardware** verhält und es wurde abgeklärt welche möglichen Einschränkungen in der weiteren Entwicklung zu erwarten waren. Vergleiche mit herkömmlichen Geräten, wie beispielsweise Raumthermometern, wurden vorgenommen, um die Messwerte zu verifizieren. Durch das Arbeitspaket **Recherche und Evaluierung** wurde abgeklärt welche Sensoren verwendet werden.

Bereits im Rahmen dieses Meilensteins konnten erste Bugs beseitigt werden. Durch Anpassung von diversen Konfigurationen konnten die **Messergebnisse der Sensoren verbessert** werden.

Des Weiteren wurde das **Aufzeichnungsintervall** analysiert. Die Frage: „Wie genau, beziehungsweise wie ausführlich ist eine Erfassung der notwendigen Sensordaten und eine ausführliche Speicherung in der Datenbank überhaupt notwendig“ konnte durch die intensive Evaluierung der Hardware sowie der Anwender geklärt werden.

Werte im Sekundenbereich sind für die Anwendung nicht notwendig, da der Anwender nicht jede Sekunde auf mögliche Änderungen reagieren kann bzw. überhaupt reagieren möchte. Das sekundliche Abtastungsintervall wurde auf das Intervall von 30 Sekunden für die Aufzeichnung ausgedehnt.

Wir konnten bei der Implementierung eines einzelnen Sensorknotens unsere **Kenntnisse über Arduino und IoT Devices** ausbauen. Die Hardwarenahe Konzeption eines Systemdesigns mit geeigneten Bauteilen bis hin zu der der Assemblierung der Test- und Entwicklungsgeräte brachte dabei die Notwendigkeit über die Grenzen der uns bekannten Softwareentwicklung hinauszudenken.

Bisher waren wir lediglich mit „fertigen Systemen“ (Hardware) und der Entwicklung passender Softwaresysteme vertraut. Nun mussten wir das Gesamtsystem konzipieren, evaluieren, assemblieren, entwickeln und erproben.

Der Zusammenbau und eine **platzsparende Architektur** war eine Herausforderung in der Gestaltung eines benutzerfreundlichen Geräts. Zum Zeitpunkt des 2. Meilensteins befanden wir uns in der Phase eines „Hardware Paperprototyps“. Wir haben mit Boxen und der IoT Hardware versucht, ein geeignetes Format für die finale Version zu finden.



Abbildung 1: Paperprototyp SensorBox

Abbildung 1 zeigt einen Paper Prototyp der ersten SensorBox. Die Idee einer kompakten Form entstand relativ früh, da wir anhand des ersten Sensorknotens erkannten, dass wir die Elektronikbauteile nicht nur an einem Ort nutzen werden, sondern in Entwicklung und Tests an verschiedenen Orten einsetzen werden müssen.

2.3 MEILENSTEIN 3 – STANDALONE GERÄT

Meilenstein 3 sah die **Implementierung eines Standalone Geräts** (Datenaufzeichnung kann über Computer als CSV ausgelesen werden) vor.

In der Planung haben wir uns auf Teilbereiche beziehungsweise funktionierende Prototypen konzentriert. Dieser Ansatz war gut, da wir schnelle Erfolge in der Implementierung der einzelnen Meilensteine erzielen konnten, dadurch mussten wir bereits geplante Aufgaben neu bedenken.

Durch den Einsatz eines **funktionierenden Prototyps**, der die **Aufzeichnung sowie Speicherung als CSV** beherrscht, und den dadurch immer wiederkehrenden Aufwand der Datenanalyse wurde uns bewusst, dass eine **Zentralisierung der Datenhaltung ehestmöglich** erfolgen musste. CSV ist aufgrund des damit verbundenen Aufwandes auf längere Sicht für immer wiederkehrende Datenauswertungen eher unbrauchbar.



Abbildung 2: Erster Prototyp einer mobilen Teststation

Abbildung 2 zeigt einen der ersten Einsätze einer mobilen Teststation. Dieser **Sensorknoten** konnte Daten als CSV-Datei erfassen, welche in Folge mittels Excel ausgewertet wurden. Die **ersten Aufzeichnungen**, Verläufe der Daten und Interpretationen dienten als weitere Grundlage der folgenden Software Entwicklung. Im Anhang sind die ersten Auswertungen verfügbar.

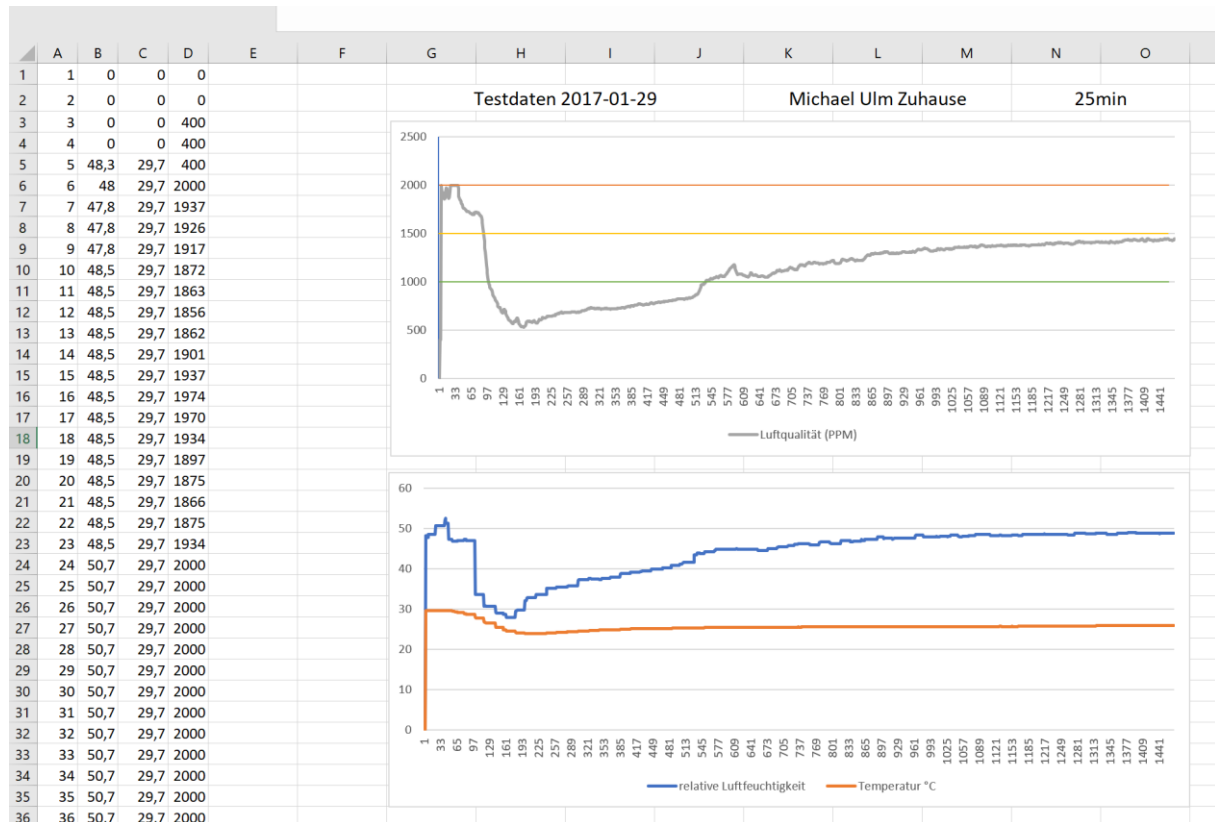


Abbildung 3: Excel Auswertung der CSV Sensor Aufzeichnungen

Die CSV Aufzeichnung und Auswertung (CSV Daten wurden in Excel importiert, Darstellung der Datenauswertung mittels Diagramm siehe Abbildung 3) war ein wertvoller Zwischenschritt um zu sehen, dass eine fortlaufende Datenaufzeichnung möglich ist. Durch diese Analysemöglichkeiten war es äußerst umständlich brauchbare Visualisierungen der Daten zu erzeugen. **Unterschiedliche Datenmengen und Zeitintervalle** lieferten uns Anfangs **unterschiedliche Graphen**, die wir erst wieder in ein einheitliches Format bringen mussten, um diese vergleichen zu können. (durch die Tests entstanden unterschiedliche Zeitintervalle, die mittels Excel kompliziert zu vergleichen waren, z.B. 1h30m oder 2h:14min),

Durch ein Vorziehen der Aufgaben

- **Datenkommunikation** mit zentraler Station
- **Datenbank** Implementierung zum vereinfachten Auslesen der Daten

konnten wir den ursprünglich nächsten Schritt der

- **Datenanalyse**

effizienter gestalten, womit wir nach dem 3. Meilenstein einige Vorarbeit für den 4. Meilenstein erledigen konnten.

Nachträglich betrachtet haben wir uns in der Gesamtplanung auf Teilbereiche beziehungsweise einen Prototyp-orientierten Ansatz konzentriert, um möglichst bei jedem Meilenstein eine lauffähige Version entwickelt zu haben. Dies hat in Summe gut funktioniert, jedoch hätte die Aufteilung der dazugehörigen Arbeitspakete Potenzial zur Optimierung geboten. So sind die Tätigkeiten der einzelnen Arbeitspakete teilweise verschwommen. Ein mögliches besseres oder auch effizienteres Vorgehen wäre in Form einer sauberen Definition von „UseCase“ / Anwendungsfällen möglich gewesen. Wir haben uns zu sehr auf die Vorgaben der Projektplanung konzentriert. Einzelne Arbeitspakete würden wir aus heutiger Sicht weiter unterteilen und darin die einzelnen Anwendungsfälle verpacken.

2.4 MEILENSTEIN 4 – ANBINDUNG ZENTRALER SERVER

Meilenstein 4 bestand aus der Anbindung des funktionierenden Stands eines Standalone Geräts an einen im Internet öffentlich zugänglichen Server. Es ist nun möglich, dass die Daten von der lokalen Infrastruktur an einen weiteren Server übertragen werden.

Der bestehende **Arduino Sensorknoten** – von uns mittlerweile **SensorBox** genannt – kommuniziert mittels **XBee** zu einem Raspberry Pi, der als lokaler Server dient. Dieses Bundle für sogenannte **offline Messungen** kann beispielsweise in einem Haushalt, einem Büro, etc. betrieben werden. Die Infrastruktur ist so vorbereitet, dass **mehrere SensorBoxen** mit einem Server betrieben werden können.

Die **Datenanalyse** wurde weiterentwickelt, sodass wir einerseits das **Lüften im Raum erkennen** können. Zusätzlich können wir unter bestimmten Umständen erkennen, dass der Raum leer ist und somit **keine Personen anwesend** sind. Wir haben diese beiden erkannten Zustände in der Anwendung als „**Lüften**“ bzw. „**Abwesend**“ gekennzeichnet.

Durch die unterschiedlichen An- und Abstiege können diese zwei Ereignisse voneinander unterscheiden und mittels Graphen visualisiert werden. Das Logging dieser Ereignisse ist nicht implementiert, da wir es als sinnvoller erachten, wenn diese **kritischen Daten nur durch implementierte Algorithmen zur Verfügung stehen**.

Wir sehen diese Entscheidung als sinnvolle Maßnahme in Bezug auf den Datenschutz von Personen. Dadurch leidet in einem zulässigen Ausmaß die Geschwindigkeit der Anwendung. Uns ist es wichtig, dass diese kritischen Daten nicht direkt in der Datenbank zum Abruf vorliegen. Ein Speichern in der Datenbank sehen wir als möglich jedoch nicht notwendig. Die Daten werden möglichst schlank gehalten und weitere Daten zur Analyse der genannten Ereignisse werden bei Bedarf aus den Rohdaten errechnet.

Durch die Anbindung an einen zentralen Server können wir auch von dort die Möglichkeit für Push-Benachrichtigungen schaffen. Die dazu notwendigen Codes werden ebenso über GitHub zur Verfügung gestellt. Zur einfachen Nutzung steht aktuell der Webservice des Projekts zur Verfügung.



Abbildung 4: Push-Benachrichtigung

Zu Testzwecken integrierten wir einen uns bekannten Push-Benachrichtigungs-Dienst, Pushover⁵, der bis zu einer bestimmten Anzahl an Benachrichtigungen kostenlos genutzt werden kann (siehe Abbildung 4). Eine einfache Integration von Pushover war mittels API Call möglich, sodass wir dazu keine eigene App entwickeln mussten. Erst im späteren Projektverlauf haben wir einen Android App Prototyp für das Auffinden des IoT AirClean Servers entwickelt. Daher denken wir im weiteren Ausbau der Applikation die Integration der Push Benachrichtigungen durch Google Firebase in die nun bestehende App an.

Die Tests mit den Benachrichtigungen haben gezeigt, dass Anwender auf die Kombination der Wachstumsratenberechnung und der daraus resultierenden Benachrichtigungen reagieren. Die Signalisierung durch diese kurze Nachricht ermöglicht den Anwendern, selbst einen geeigneten Zeitpunkt zu wählen. Das ist in Unterrichtssituationen von Vorteil, da sehr gezielt eine Pause eingeleitet werden kann und somit ein notwendiges Lüften des Raums durchgeführt wird. Dies geschieht ohne zu offensichtliche und störende Unterbrechung des normalen Unterrichtsverlaufs.

⁵ <https://pushover.net/>

2.5 MEILENSTEIN 5 – FELDTTEST BEI PRIVATPERSON

Durch die vorhergegangenen Implementierungen konnten wir Feldtests bei Privatpersonen durchführen und dazugehörigen Meinungen einholen. Wir hatten die Gelegenheit den **Usability Experten** Elmar Krainz⁶ mit einer Teststation auszustatten. Der Usability Test von Elmar Krainz wird als zusätzliches Dokument zur Verfügung gestellt.

Die Evaluierung von Herrn Krainz wurde mit der Methode Heuristic Evaluation (nach Nielsen⁷) mit den 10 Nielsen Heuristiken⁸ durchgeführt.

Die Ergebnisse waren durchwachsen, jedoch mit sehr positiven Gesamtergebnis. Kleinere Designfehler bzw. Erklärungen konnten noch zeitgerecht eingepflegt werden, um die Nutzung der Weboberfläche noch intuitiver zu gestalten. Das getestete **Interface wird als einheitlich, klar und fehlerfrei beschrieben**. Nach dem Anschließen und dem Zugriff auf die Website war die **weitere Bedienung selbsterklärend**.

Das Feedback zur Farbsehschwäche (z.B. grün-rot bei der Darstellung der Messwerte) wird in der Weiterentwicklung berücksichtigt. Durch die interaktive Komponente (Mouse Over Effekt und Darstellung der zusätzlichen Daten) ist aktuell kein dringender Handlungsbedarf notwendig.

Die Teststationen befanden sich in privaten Wohnungen sowie im Büro der mea IT Services. Die Teststation im Büro dient als Demonstration der Gesamtanwendung. In unserer Entwicklungszeit haben wir eine Teststation so ausgestattet, dass wir diese mobil einsetzen können. Dies wurde in den Feldtest in Schulungsräumen mehrfach genutzt.

⁶ <https://www.fh-joanneum.at/hochschule/person/elmar-krainz/>

⁷ <https://www.nngroup.com/articles/how-to-conduct-a-heuristic-evaluation/>

⁸ <https://www.nngroup.com/articles/ten-usability-heuristics/>

2.6 MEILENSTEIN 6 – FELDTTEST IN SCHULUNGSRaum

Durch den Feldtest in einem Schulungsraum konnte der 6. Meilenstein erreicht werden.



Abbildung 5: Schulungsraum WIFI Steiermark Graz

In einem **Feldtest während des Abend Unterrichts** am WIFI Steiermark Graz konnte IoT AirClean in einer **Unterrichtssituation** erfolgreich getestet werden. Als Testumgebung diente der Computerraum wie in Abbildung 6 dargestellt.



Abbildung 6: Mobile Teststation mit Mobile Internet

Die Teststation wurde unauffällig im Raum positioniert. Server und Mobile Internetanbindung wurden für den Einsatz im Unterricht vorab getestet. Die SensorBox selbst wurde in kurzer Distanz zum Server aufgestellt um mögliche Störeinflüsse zu minimieren (siehe Abbildung 7).



Abbildung 7: Mobile Teststation und die SensorBox

Die Feldtests wurden über drei Abende durchgeführt um mögliche Fehlmessungen auszuschließen. Die **Tests** brachten ähnliche Ergebnisse hervor und **zeigten einen korrekten Betrieb der Anwendung**. Es wurden gezielte Pausen und Lüftungen durchgeführt. Aus den Tests ging hervor, dass eine Push-Benachrichtigung ein gutes Instrument zur Orientierung sein kann. Eine Verbesserung aus den Tests fügten wir den SensorBox noch hinzu, um einen weiteren Indikator der aktuellen Raumluft zu besitzen.

Dieses Tests werden als Anhang dem Endbericht hinzugefügt.

2.7 MEILENSTEIN 7 – FEHLERANALYSEN

Durch die erkannten Fehler und deren Behebung haben wir mit Meilenstein 7 einen weiteren Projekterfolg verzeichnet. In den ersten Projektmonaten wurden die „Basisfunktionalitäten“ entwickelt und der **Feinschliff** wurde durch die Arbeitspakete Tests und Bugfixing durchgeführt.

Die Feldtests ergaben, dass das **UI Design** trotz kleinerer **intuitiv sowie klar verständlich** war. Adaptierungen im Layout ermöglichten eine Behebung dieser kleineren Fehler und eine **übersichtliche Darstellung der Gesamtinhalte**. Die Unterstützung eines Raspberry Pi Bildschirms (siehe Abbildung 8) und ein Responsive Design passen die Webinhalte bestmöglich an die jeweiligen Bildschirmgrößen an.

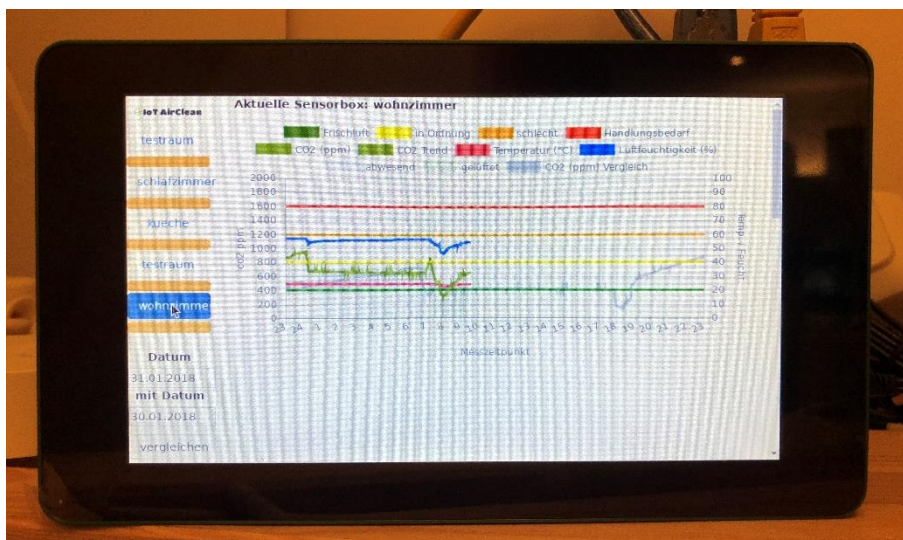


Abbildung 8: Raspberry Pi mit Touchscreen und Web Interface

Im Anwendungsbereich konnten kaum Fehler gefunden werden. Hierbei handelte es sich um Schwierigkeiten, mit denen ein wenig IT versierter Anwender zu kämpfen hat. Die Adressierung des Raspberry Pi kann sich für eine ungeübte Person als schwierig erweisen. Um diese Situation zu verbessern haben wir einen Prototyp einer **Android App** (siehe Abbildung 9) entwickelt, die das Netzwerk nach IoT AirClean durchsucht und den Anwender direkt auf die Weboberfläche weiterleitet.

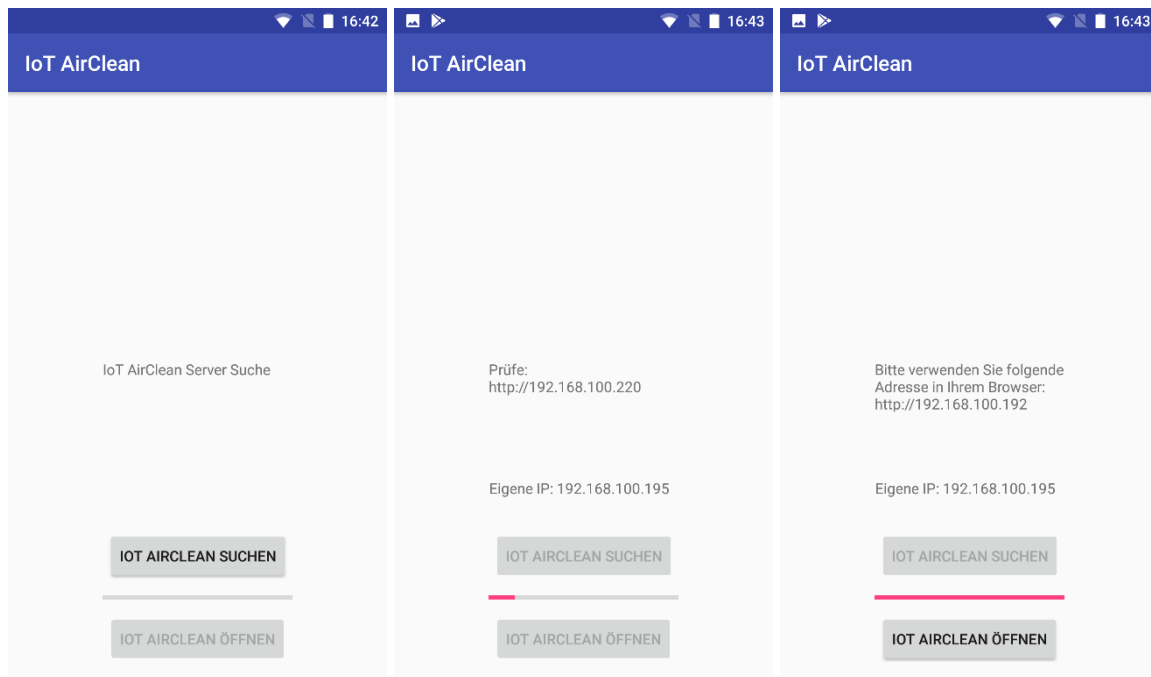


Abbildung 9: Mobile App für Serversuche

Die letzte Integration der **Visualisierung der aktuellen CO2 Werte** verzeichnen wir auch als wichtigen Bestandteil zur Erfolgsmessung der Fehleranalyse. Wir erkannten, dass im Betrieb für eine Unterrichtssituation oder allgemein in der Verwendung in einer Arbeitsumgebung der Mensch sehr gut auf **visuelle Reize** reagiert. Daher integrierten wir eine RGB LED System und übernahmen die Farbgebung aus der Weboberfläche. Diese ist bereits nach dem Ampelsystem entworfen worden.

2.8 MEILENSTEIN 8 – ENTWICKLUNG DER BASISSTATION

Parallel zu Meilenstein zu Meilenstein 7 – Fehleranalysen wurde auf Meilenstein 8 mit Entwicklung einer Basisstation hingearbeitet.

Die Basisstation, im Projekt mittlerweile als **Server** deklariert, kann einerseits als **Raspberry Pi** „alleine“ oder kombiniert mit einem 7“ **Touchscreen** verwendet werden. Die Verwendung mit einem Touchscreen hat den Vorteil, dass Anwender ohne weiteres Endgerät das Produkt IoT AirClean verwenden können. Abbildung 8 zeigt den IoT AirClean Server mit der responsiven Oberfläche. Abbildung 10 zeigt den **3D Druck**. Das „unhübsche“ blaue USB Kabel wurde in weiterer Folge mit einem gewinkelten USB Kabel ausgetauscht.



Abbildung 10: 3D Druck - Server mit Display

Die Anzeige am Raspberry Pi mit Touchscreen wird dank **Responsive Design** der Größe angepasst und benötigt keiner gesonderten Entwicklung. Ein **Automatischer Start** des Chrome Browsers macht den IoT AirClean Server zum **POI-System**. Hintergrundgedanke

dieser Integration ist eine mögliche fixe Montage in einem Büro oder in Unterrichtsklassen, Seminarräumen oder anderen Plätzen, wo eine schnelle Analyse ohne weiteres Endgerät eine Rolle spielen könnte. Weitere Adaptierung als reines Client System wäre notwendig, da der Server nur einmal in einer Infrastruktur benötigt werden würde. Ansonsten müsste pro Raum ein Server und mindestens eine SensorBox betrieben werden.

Unsere weiteren Überlegungen einer Basisstation führten zur letzten Anpassung eines **einzelnen Gehäuses**. Durch unsere Erfahrungen im mobilen Betrieb erkannten wir, dass mehr als ein Gerät für unterwegs relativ kompliziert im Setup werden könnte. So haben wir als **Ausgangsdesign** einen **Würfel** verwendet.

Das Cube Gehäuse für IoT AirClean bietet genügend Platz für die Server- sowie SensorBox Komponenten. Eine gesonderte Systemarchitektur sehen wir hier nicht vor, sondern fügen beide Geräte in ein Gehäuse zusammen, um weiterhin mehrere SensorBoxen, wie in Kapitel 2.9 beschrieben, zu unterstützen.



Abbildung 11: 3D Druck - IoT AirClean Cube

Der IoT AirClean Cube ist für unterwegs handlicher als zwei Geräte und eine alternative zur vorherigen Gehäuse Lösung mit dem 7“ Touchscreen.

2.9 MEILENSTEIN 9 – MEHRERE SENSORBOXEN

Nach den Tests einer Basisstation (Server) und mehreren Sensorknoten (SensorBoxen) konnten wir den Meilenstein 9 erreichen. In unserer Testumgebung hatten wir bis zu 4 SensorBoxen zur gleichen Zeit in Betrieb.

Die Auswahl aus mehreren SensorBoxen erfolgt über die implementierten Schaltflächen der Weboberfläche. Die Schaltflächen leuchten zusätzlich auf, wenn eine SensorBox aktiv verbunden ist. So kann ein Anwender schnell erkennen ob die Verbindungen erfolgreich waren. Die Daten werden im Betrieb live aktualisiert.

Die Tests mit **mehreren SensorBoxen** erfolgten auch über **mehrere Stockwerke**, um eine möglichst reale Situation nachzustellen. (beispielsweise ein Unternehmen welches in einem Gebäude über mehrere Stockwerke verteilt ist). Die SensorBoxen wurden in verschiedenen Räumen positioniert. Wir erkannten beim Test eine durchaus stabilere Kommunikation als mit einer einzelnen SensorBox. XBee kommuniziert über die einzelnen Module und bei Abbruch einer direkten Kommunikation kann das Signal über ein anderes Modul geleitet werden. Diese bessere Abdeckung ist ein positiver Nebeneffekt. Somit sind Störungseinflüsse weniger relevant als beim Betrieb mit nur einer SensorBox.

2.10 MEILENSTEIN 10 – PROJEKTABSCHLUSS

Beim Meilenstein 10 sind alle Meilensteine erreicht und die finalen Schritte zum Projektabschluss sind gesetzt.

Die Dokumentation sowie der Endbericht wurden finalisiert und übermittelt. Das Projekt befindet sich noch in der fortlaufenden Bewerbung und Verbreitung in der Online Community. In der Nachprojektphase wollen wir zusätzlich die Online Community stärken und weitere Tester dazu ermutigen, das Produkt sich im Live Betrieb anzusehen.

3 ARBEITSPAKETE

3.1 PROJEKTMANAGEMENT

Wir betreiben fortlaufende **Dokumentation** und **Veröffentlichung** der bisherigen Ergebnisse auf

- GitHub,
- Facebook,
- Twitter,
- Instagram
- und auf dem Blog auf unserer Website

In der Projektlaufzeit sind Blogeinträge sowie Social Media Interaktionen vorgesehen. Wir haben versucht die Intervalle von ca. 3-4 Wochen nicht zu überschreiten. Erst gegen Projektendphase hatten wir etwas Verzug zu unseren Blogeinträgen, da wir den starken Fokus auf die Finalisierung des Projekts gelegt haben.

Es ist eine Vernetzung mit öffentlichen Einrichtungen und Schulungsanbietern zum Test des Projekts beziehungsweise zur weiteren Verbreitung des Projekts geplant. Unter den Einrichtungen befinden sich unter anderem die FH JOANNEUM und das WIFI Steiermark.

Projektcontrolling wurde fortlaufend durchgeführt, um die Zeit nicht aus den Augen zu verlieren. Das Projekt wurde durch äußere Einflüsse zu Projektende in der Zeitplanung etwas gestört, sodass wir noch zusätzlich den Jänner 2018 für Dokumentation, Bugfixing und letzte Finalisierungen benötigten.

3.2 RECHERCHE

Unsere Recherchetätigkeiten konzentrieren sich vor allem im Bereich des Projektbeginns. **Architekturentscheidungen und Technologieentscheidungen** – für Software und Hardware – wurden ebenso größtenteils vor der Entwicklungsarbeit getroffen.

Details wurden während der Softwareentwicklung beziehungsweise nach Anschaffung der Hardware analysiert. Es gab mehrere aussagekräftige Meinungen der Community, jedoch gab es in Bezug auf die CO₂ Messung unterschiedliche Erfahrungswerte sowie Umsetzungsvorschläge.

Der ausgewählte Sensor **DHT22** für Temperatur und Luftfeuchtigkeit⁹ ist bei Anwendungen zur Bestimmung der Raumtemperatur bzw. der Luftfeuchtigkeit der gängigste Sensor. Somit waren auch viele Quellen zur Bedienung des Sensors vorhanden.

Der zweite Sensor für die Ermittlung der CO₂ Werte war im Vergleich zum DHT22 deutlich vielseitiger. Hauptunterschiede lagen in der Genauigkeit und dem damit verbundenen Preis. Wir haben uns für den **MH-Z16** CO₂ Sensor¹⁰ entschieden. Dieser bietet eine sehr hohe Genauigkeit sowie einen moderaten Preis.

Eine gesamte Bauteilliste (inkl. Sensoren) ist auf der Projektwebsite¹¹ als **Bauanleitung** sowie als Anhang als PDF verfügbar.

Wir haben uns für eine **NoSQL Datenbank** entschieden, um das Datenmodell während der Entwicklungszeit oder bei Weiterentwicklungen flexibler erweitern zu können.

Des Weiteren haben uns für eine **Client-Server-Architektur** entschieden, um mehrere Sensorknoten – nun als SensorBox im Projekt bezeichnet – betreiben zu können. Der lokale Server soll den Vorteil schaffen, die Daten nicht auf einen im Internet verfügbaren Server übertragen zu müssen. Die Gesamtanwendung kann in einem eigenen Netzwerk von der

⁹ <http://www.mikrocontroller-elektronik.de/dht22-am2302-luftfeuchte-und-temperatursensor/>

¹⁰ <http://sandboxelectronics.com/?product=mh-z16-ndir-co2-sensor-with-i2cuart-5v3-3v-interface-for-arduinoraspeberry-pi>

¹¹ <http://www.iot-airclean.at/bauanleitung/>

Öffentlichkeit abgeschottet verwendet werden. Es besteht jedoch auch die Möglichkeit, die Daten an einen zentralen Server weiterzuleiten, um z.B. die Daten auch extern zugänglich zu machen.

In Summe konnten die notwendigen Recherchearbeiten bereits in einer sehr frühen Projektphase nahezu vollständig abgeschlossen werden. Eine detailliertere Evaluierung möglicher Systemarchitekturen wäre im Nachhinein empfehlenswert gewesen. Bei der Durchführung ähnlicher Projekte werden wir dies in Zukunft stärker berücksichtigen und auch bei der Planung einzelner Meilensteine ebenso miteinbeziehen.

3.3 EVALUIERUNG

Die Evaluierungen wurden in kleineren Schritten durchgeführt um mögliche Fehlerquellen frühzeitig zu reduzieren. Durch die Recherche waren uns bereits viele Komponenten bekannt und wir konnten diese in einem ersten Schritt auf die gewünschte Funktion evaluieren.

In den nachfolgenden Kapiteln gehen wir auf die wichtigen Bereiche der durchgeführten Evaluierungen ein.

3.3.1 SENSOREN

Die Sensoren wurden einzeln getestet und möglichst mit verfügbaren Mitteln auf deren Korrektheit geprüft.

Temperatur und **Luftfeuchtigkeit** konnten durch eine vorhandene Wetterstation auf Genauigkeit geprüft werden, bevor eine größere Anzahl für weitere Teststation angekauft wurde.

Betreffend **CO2** Sensor haben wir uns auf subjektive Messkriterien verlassen. Dabei haben wir die Funktion im Frühjahr sowie in den Sommermonaten gut testen können. Die „frische“ Luft außerhalb der Stadt bot eine geeignete Möglichkeit, den CO2 Sensor auch in einem Langzeittest zu erproben. Die notwendige Genauigkeit der beiden Hauptsensoren für das Projekt ist sehr gut.

3.3.2 HARDWARE

Die weiteren Hardware Bauteile beschränken sich im weiteren Projekt auf

- Arduino
- Raspberry Pi
- Touchscreen
- XBee
- RGB LED

Die ausgewählten Hardware (Bau)teile sind in vielen Onlineshops oder lokalen Händlern verfügbar. Eine Bauanleitung haben wir als Anhang dem Endbericht hinzugefügt.

3.3.3 GEHÄUSE

Bereits bei der Auswahl eines Raspberry Bundles haben wir mit dem Standard Gehäuse gearbeitet. Durch den Einsatz des externen XBee Moduls haben wir weitere Gehäuse evaluiert bzw. haben für den unseren Einsatzbereich keine passenden Gehäuse für Raspberry + XBee Modul bzw. Arduino mit XBee Modul, DHT22 sowie CO2 Sensor gefunden.

Wir haben uns dazu entschlossen selbst Gehäuse zu entwerfen und mittels **3D Druck** Verfahren anzufertigen. Durch die mögliche Individualisierung der einzelnen Gehäuse konnten wir bestehende Modelle der Online Community adaptieren und für unsere Bedürfnisse anpassen.

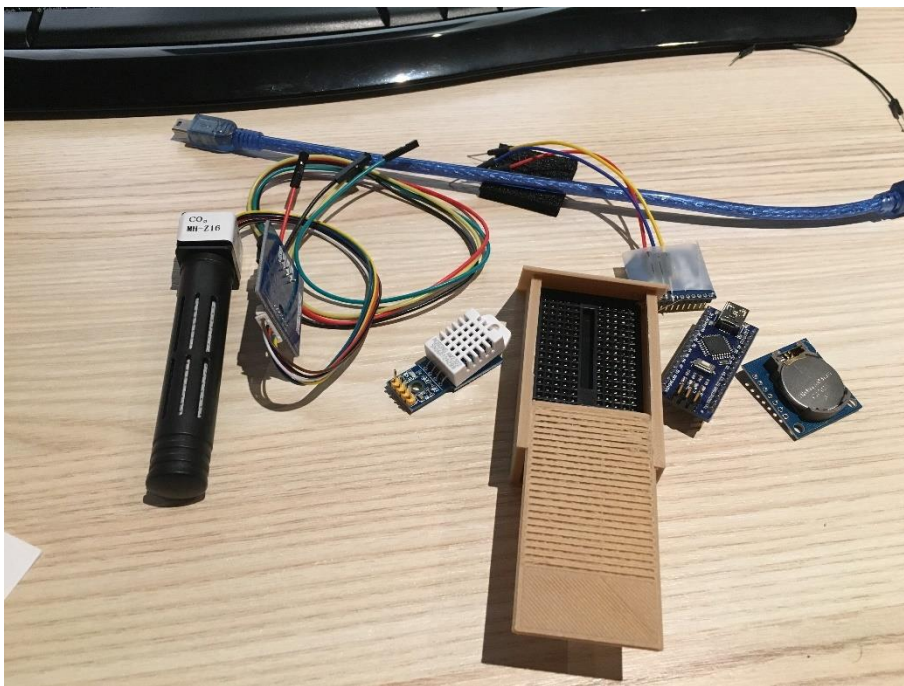


Abbildung 12: 3D Druck - 1. Entwurf - viel zu klein

Neben der Hardware- und Softwareentwicklung mussten wir unsere Vorstellungskraft und Fähigkeiten für **3D Entwürfe, Größen und Maßstäbe** erweitern. Die Plattform Thingiverse lieferte uns erste Anregungen für mögliche Gehäuse-, Boxen- oder Aufbewahrungsdesigns. Abbildung 12 und Abbildung 13 zeigen unseren ersten Versuch ein SEHR kompaktes Gehäuse zu drucken. Uns fehlte jegliche konkrete Vorstellung von Größe bzw. notwendiger Größe. Das war bedingt durch die Assemblierung der Bauteile. In eine Box musste nicht nur Sensor und Arduino, sondern auch Funkmodule sowie doch recht zahlreiche Verkabelungen

untergebracht werden. So mussten wir uns durch mehrfache Versuche, wie z.B. in Abbildung 14, an eine erste brauchbare Version herantasten.



Abbildung 13: 3D Druck - 1. Entwurf - Detailsicht

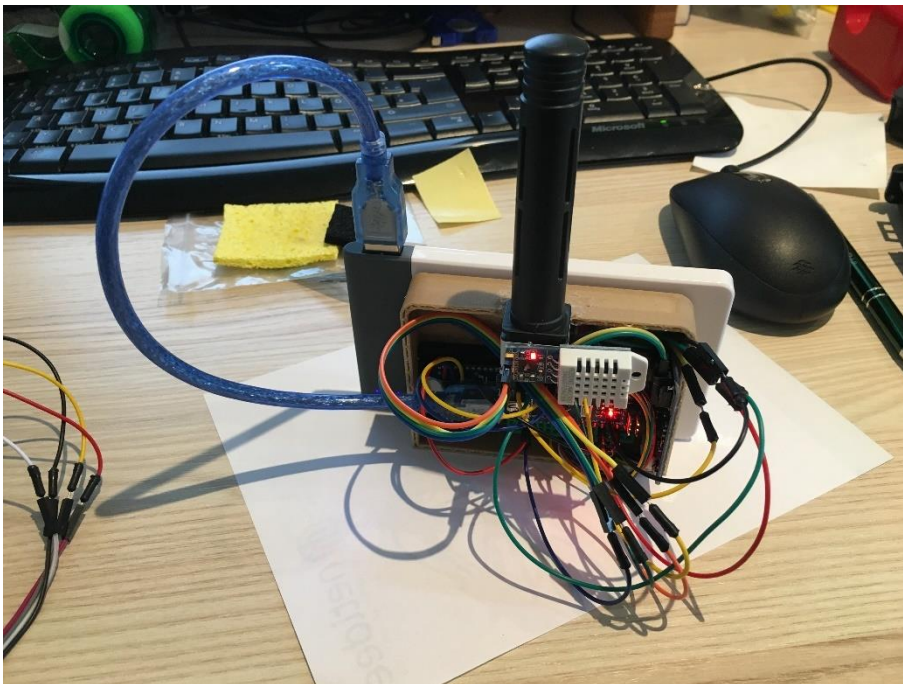


Abbildung 14: 3D Druck - weiterer Entwurf - größer jedoch immer noch zu klein



Abbildung 15: 3D Druck - 1.SensorBox - kompakte Größe und 1. SensorBox

Abbildung 15 stellt die **erste Version der SensorBox** dar. Diese Box war nun der 3D Grundentwurf, an den wir in den nächsten Monaten weitere Anpassungen nach unseren Bedürfnissen durchgeführt haben. Dieses Gehäuse hatte eine kompakte Größe und konnte in nur engsten Raum 2x Sensoren, 1x Arduino, 1x XBee Modul sowie die Verkabelungen verstauen.

Neben dieser SensorBox sind zwei weitere Gehäuse entstanden.

3.3.4 PROGRAMMIERSPRACHEN

Je nach Anwendungsgebiet haben wir in diesem Projekt einen Mix aus verschiedenen Programmiersprachen ausgewählt und angewandt.

Umgebung	Programmiersprache
Arduino (SensorBox)	C/C++
Raspberry (Server)	Python
Weboberfläche (GUI)	PHP, JavaScript

Unabhängig von den Programmiersprachen und Umgebungen haben wir die jeweiligen Komponenten, Erweiterungen oder Bibliotheken, die zum Einsatz kamen, vorerst evaluiert und danach in die Anwendung integriert. Wir haben uns bei unseren Tests an Beispielimplementierungen der Online Community orientiert und einen möglichen Einsatz für das Projekt bewertet. Nach der Auswahl wurden Tests durchgeführt, um festzustellen, ob die Beispiele auch für den gedachten Einsatz geeignet sind. Durch dieses Vorgehen konnten wir die Applikation stetig weiter ausbauen.

Die Evaluierung ganzer Frameworks haben wir nur auf inhaltlicher Ebene vorgenommen. Aufgrund der Projektlaufzeit und der verfügbaren Umsetzungszeit haben wir eine technische Evaluierung im Detail nicht durchgeführt. Im Nachhinein wäre unter bestimmten Umständen eine solche Detailanalyse eines potentiellen Frameworks nützlich gewesen. Zum damaligen Projektzeitpunkt haben wir die Entscheidung dagegen getroffen. Eine schlanke Architektur ist uns wichtiger als eine überdimensionierte Umsetzung. Wir konnten diese Entscheidung weiterhin gut im Projekt durchziehen und die funktionierende Version unserer Entwicklung bestärkt diese Entscheidung.

Uns bekannte Technologien und Projektentscheidungen haben sich im Projekt als eine sehr gute Wahl erwiesen.

- Websocket zur vereinfachten Kommunikation
- Python sowie PHP zur Umsetzung der spezifischen Programmteile
- Trennung von Server und SensorBox

3.4 SOFTWAREENTWICKLUNG

Die Softwareentwicklung hat sich im ersten Schritt auf die Infrastrukturthemen konzentriert, damit wir auf einfachere Art und Weise auf die aktuellen Testdaten zugreifen können.

3.4.1 DATENKOMMUNIKATION

Die **Datenkommunikation zwischen Arduino und Raspberry Pi** wird mittels XBee Funktechnologie durchgeführt. Die Sensorerfassung erfolgt mittels Arduino und verbundenen Sensoren. Die Daten werden per XBee auf den Raspberry Pi übertragen und dort gespeichert sowie weiterverarbeitet. Die Datenspeicherung wird näher im Kapitel 3.5 Datenverwaltung beschrieben.

Die nächsten Entwicklungen richteten sich auf die **Datenauswertung** sowie die **Interaktion** mit dem Benutzer. Dabei wurde eine Weboberfläche zur Datenanalyse entwickelt. Eine Steuerung ist durch die Weboberfläche nicht möglich und auch nicht notwendig. Die einzelnen SensorBoxen können durch die Konfiguration an den SensorBoxen einer bestehenden Infrastruktur hinzugefügt werden.

3.4.2 DARSTELLUNG DER DATEN

Die **Weboberfläche** baut sich durch die vorhandenen Daten ohne weitere Konfiguration selbständig auf und fügt neue Sensorknoten nach einem Neuladen selbstständig hinzu. Anwender können durch die intuitiv nutzbare Oberfläche Tagesdaten (von 0 bis 24 Uhr) auslesen und mit anderen Tagen (ebenso 0 bis 24 Uhr) vergleichen. Dem Anwender steht eine **Bedienungsanleitung**¹² zur Verfügung. Die selbe Bedienungsanleitung ist im Anhang als PDF verfügbar.

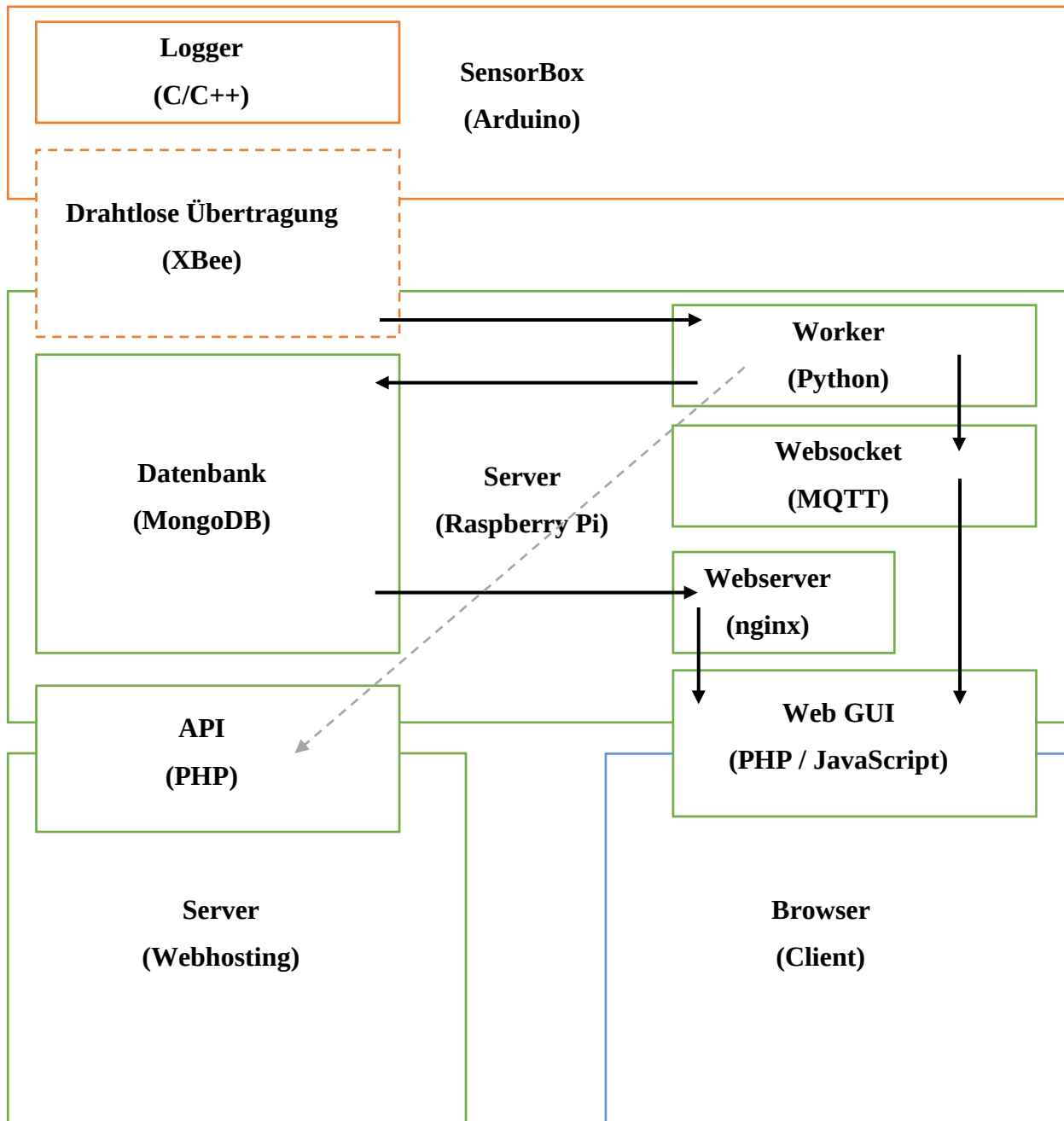
Die Datenauswertung erfolgt am Raspberry Pi. Hierbei wird auf Basis der aktuellen Sensordaten eine Wachstumsrate¹³ berechnet, eine Vorschau vorbereitet und in der Weboberfläche zur Live Auswertung dargestellt. Diese Datenauswertung löst in weiterer Folge auch Push-Benachrichtigungen (die aktuell als App Demo vorbereitet wurden) aus. Aus den Feldtests ergab die Berechnung der Wachstumsrate eine sehr gute Möglichkeit, geeignete Zeitpunkte zum Lüften aufgrund der aktuellen Gegebenheiten zu berechnen.

¹² <http://www.iot-airclean.at/bedienungsanleitung/>

¹³ <https://de.wikihow.com/Berechnung-einer-Wachstumsrate>

Unsere Implementierung setzt dabei auf Berücksichtigung der letzten Minuten und liefert selbst nach mehrfachen Lüften gute Ergebnisse.

3.4.3 DIE SYSTEMARCHITEKTUR



3.5 DATENVERWALTUNG

Wir setzen in unserem Projekt auf eine **NoSQL Datenbank**, um lokal mit sehr großen Datenmengen arbeiten zu können. Bisher hat sich dieses Konzept als gut erwiesen. Es gab bisher nur eine Einschränkung, welche nicht im Rahmen des Konzeptes, sondern der verwendeten Hardware (Raspberry) liegt. Bei größer werdenden Datenmengen (zu hohe Messintervalle, sprich jede Sekunde) war die Folge, dass die Datenbank auf dem IoT Device nicht mehr laden konnte. Daran möchten wir noch arbeiten und mögliche Lösungsansätze integrieren.

Die Datenmenge in unserem Testszenario stieg rapide an, da wir bis zu 4 SensorBoxen mit nur einer Basisstation in Betrieb hatten. Weitere SensorBoxen sind durchaus möglich. Der Betrieb mit den mehrfachen SensorBoxen hat uns gezeigt, dass die Verarbeitung der Daten – bis auf die wachsende Datenmenge – kein Problem ist. Durch die Reduzierung des Messintervalls von 1 Sekunde auf 30 Sekunden war für uns der weitere Testbetrieb kein Problem.

In weiterer Folge wäre es notwendig, die Daten nach Ablauf definierten bestimmten Zeiten zu minimieren, um die wachsende Datenmenge in Griff zu bekommen. Dabei könnten beispielsweise nach 30 Tage die Daten von 30 Sekunden auf 1 Minute, nach weiteren 90 Tagen auf 5 Minuten und nach 180 Tagen auf 10 Minuten minimiert werden.

Zurückblickend auf die Evaluierung des Datenbanksystems war zwar NoSQL eine gute Wahl aufgrund der vorliegenden Kriterien. Wir hätten jedoch Raspberry sowie Webhosting stärker berücksichtigen sollen, da der Standard aktuell weiterhin in einem RDBMS liegt und meistens MySQL im Einsatz ist. Auch unsere Kenntnisse zu NoSQL waren zu Projektbeginn nicht so stark ausgeprägt wie unsere langjährige Erfahrung mit MySQL.

In der Projektlaufzeit haben wir auch die aktuellsten Updates sowie Versionen berücksichtigt, jedoch war das mit einem Umbau des Datenbankzugriffs verbunden. Einen solchen Unterschied auf den Zugriff auf eine Datenbank haben wir in den mehr als 15 Jahren PHP/MySQL Entwicklungserfahrung bisher nicht erlebt. Dieser hat bei MongoDB etwas Adaptierungszeit gekostet.

Eine gute Definition des Datenmodells hat auch in diesem Projekt gezeigt, dass mehr Zeit in der Planung mindestens die Zeit in der Umsetzung wieder einbringt. Wir haben uns das Datenmodell erst bei der Umsetzung der Datenverwaltung zurechtgelegt. In einer früheren Projektphase wie Planung, Recherche oder Evaluierung wäre die Umsetzung etwas effizienter erfolgt.

3.6 TESTS

Eine Teststation beim Entwickler sowie weitere **Teststationen** wurden in Betrieb genommen. Die Tests fanden in Schulungsräumen sowie bei Privatpersonen statt.

Durch die Tests konnten kleinere **Probleme im Gehäusedesign** entdeckt werden. Die SensorBox Variante wie in Abbildung 15 musste adaptiert werden. Der Grund waren Ungenauigkeiten im längeren Betrieb. Durch das kompakte Design bot die Box wenig Platz für einen ausreichenden Luftaustausch, sodass die Luft in der Box erwärmt und somit falsche Resultate der Raumluft lieferte.



Abbildung 16: 3D Druck - finale Version der SensorBox

Die SensorBox wurde daraufhin (siehe Abbildung 16) optimiert.

- DHT22 Sensor (weißes Teil) ragt nun nach außen
- Zusätzliche Lüftungsschlitze
- CO2 Sensor kann vollständig ausgezogen werden (dies war durch die zusätzlichen Lüftungsschlitze nicht mehr notwendig, jedoch besteht weiterhin die Möglichkeit)

Die konzipierten bzw. adaptierten Gehäuse wurden wieder auf der **3D Modellplattform Thingiverse**¹⁴ zur Verfügung gestellt. Erste Reaktionen der Community waren bereits nach kurzer Zeit sehr positiv.

¹⁴ <https://www.thingiverse.com/>

3.7 BUGFIXING

Einige Fehler konnten in den ersten Demoimplementierungen bereits behoben werden. Weitere sind geplant und erfolgen nach intensiven Testeinsatz.



Abbildung 17: Usability Erweiterung einer RGB LED der SensorBox

Im Bugfixing griffen wir Erfahrungen der Feldtests zurück und integrierten eine **RGB LED zur Signalisierung der aktuellen Raumluft**. So kann eine SensorBox auch ohne Basisstation autark betrieben werden. Es werden aktuell am Arduino keine Daten gespeichert, jedoch kann das farbige Leuchten der LED – welches an eine Ampel angelehnt ist – bereits zur ersten Orientierung des aktuellen Status der Raumluft verwendet werden.

Rückmeldungen und Ergebnisse einzelner interner Tests wurden ebenso berücksichtigt. So ergaben sich offensichtliche Verbesserungen bzw. notwendige Erweiterungen, an die zuerst nicht gedacht wurden. Bereits erwähnt wurde die **Android App**, die den IoT AirClean Server im selben Netzwerk ausfindig macht. Anpassungen der Weboberfläche brachten noch

weitere **Usability Faktoren** und einen besseren Erfolg in der Bedienung bzw. dem Auswerten der Sensordaten. Das Umschalten zu den einzelnen SensorBoxen wurde bereits intern stark vereinfacht.

Bestimmte Themen, die uns bereits in der Entwicklung aufgefallen sind, haben wir versucht so intuitiv wie möglich zu implementieren. Das Resultat ist eine Anwendung, die in der Bedienung sehr einfach ist, jedoch für das Grundsetup und möglichen Laufzeitfehlern etwas mehr als normale PC Anwenderkenntnisse erfordert. So unterstützen wir den normalen Anwender nun auch mit einem RGB LED an der SensorBox sowie einer Visualisierung des Verbindungsstatus in der Weboberfläche. Somit kann schnell geprüft werden, ob Daten empfangen und verarbeitet werden können.

4 ZUSAMMENFASSUNG

4.1 PROJEKTERGEBNISSE

IoT AirClean zeigt das **Zusammenspiel aus Sensoren und Software**. In einer möglichst praktikablen und nützlichen Anwendung wird **Internet of Things** – Das Internet der Dinge – dargestellt. Die **Aufzeichnung** der Sensordaten wird langfristig gespeichert sowie durch eine **intuitive Darstellung** visualisiert. Somit ist eine **Analyse** der Umgebung ohne ein umfangreiches Vorwissen möglich. Dadurch wird ein scheinbar einfaches Thema, das Lüften von Räumen, technisch unterstützt.

Das Zusammenspiel zwischen entwickelter Software und der IoT Hardware wird durch

- **Bauanleitung**¹⁵,
- **Installationsanleitung**¹⁶,
- **Bedienungsanleitung**¹⁷ sowie
- umfangreichen **Fragenkatalog**¹⁸

für interessierte Personen – potentielle Anwender – möglichst verständlich und zielgruppengerecht erklärt, und steht auf der Projektwebsite frei zur Verfügung.

Eine kompakte **Übersicht für Entwickler**¹⁹ bietet einen geeigneten Einstieg zu den wichtigen Onlinequellen des Projekts

- **GitHub** Repository²⁰
- Thingiverse **3D Druck** Objekte²¹

Durch Bereitstellung und der ausgewählten Open Source Lizenzen von Source Code und weiteren Projektergebnissen ist eine freie Weiterentwicklung für interessierte Entwickler möglich.

¹⁵ <http://www.iot-airclean.at/bauanleitung/>

¹⁶ <http://www.iot-airclean.at/installationsanleitung/>

¹⁷ <http://www.iot-airclean.at/bedienungsanleitung/>

¹⁸ <http://www.iot-airclean.at/faq-haeufig-gestellte-fragen/>

¹⁹ <http://www.iot-airclean.at/entwicklerdokumentation/>

²⁰ <https://github.com/michaelulm/iotairclean>

²¹ https://www.thingiverse.com/Michael_ulm/collections/iot-airclean-powered-by-netideeat-and-mea-it-services

4.2 GEPLANTE WEITERENTWICKLUNG

Das Projekt IoT AirClean wurde mit einer **modularen Systemarchitektur** entwickelt. Diese flexible Gesamtsystemarchitektur ermöglicht es nach Projektende das Projekt ebenso modular weiter zu entwickeln.

Im Kapitel 3.4 Softwareentwicklung wurde bereits das Systemkonzept beschrieben. Damit haben wir Vorbereitungen getroffen um z.B. Daten an weitere Systeme zu senden beziehungsweise eine Anbindung einer mobilen Applikation durchzuführen.

Die Feldtests haben noch Punkte aufgezeigt, die im weiteren Ausbau als Erweiterung dienen können. Diese weiteren Punkte konnten wir in der Projektumsetzung nicht mehr integrieren.

- Anfänger / Expert Modus (Oberfläche)
- Mobile App vs. bisherige Weboberfläche (Datenvisualisierung direkt in einer App)
- Designdetails der Weboberfläche (Überarbeitung der Farbgebung bzw. Designs)
- Kompakteres (Hardware) Design für mobile Einsätze (Systemkomponenten repositionieren bzw. unter Umständen austauschen.
- Lösch-, Minimierungs- bzw. Komprimierungs-Mechanismen um wachsende Datenmengen in den Griff zu bekommen
- Autarke Datenaufzeichnung der SensorBox bis eine Basisstation verfügbar ist (Datenpuffer im Gerät)

Early Adopter konnten von diesem Projekt überzeugt werden, und sollen weiter betreut werden um das Projekt weiter zu verbreiten. Die weitere Nutzung im Unterricht und weitere Präsentation in geeigneten Einrichtungen sollen in der Nachprojektphase erfolgen.

Viele Teilbereiche mussten in diesem Bereich miteinander vereint werden. Das vorliegende Wissen wollen wir in einem zukünftigen Ausbauschnitt in ein Unterrichtsdesign integrieren. Das Projekt kann mit Adaptierungen in weitere Anwendungsfälle abgewandelt werden, und bietet ein durchgängiges Beispiel für viele Lerninhalte, die in Informatikklassen relevant sind.

4.3 PERSÖNLICHE MEINUNG

Das Projekt IoT AirClean ist aus einem Gedanken – eine Lern- oder Arbeitsumgebung und die damit verbundene Raumlufte zu optimieren – eingereicht worden. Durch das Förderprogramm der Netidee wurde aus dem Gedanken ein Produkt mit Open Source Charakter und weiteren Ausbaumöglichkeiten.

Unser Unternehmen wird das Produkt sowie die Referenzimplementierung weiterentwickeln. Das entstandene interdisziplinäre Software- und Hardwareprodukt dient uns als gelungene Umsetzung und Demonstration eines Proof-of-Concept.

Netidee bot eine geeignete Möglichkeit mit einer Idee zu starten und zu entwickeln, um eine gute **funktionsfähige Version der Öffentlichkeit zur Verfügung** zu stellen. Die Zielgruppe der Bildungseinrichtungen und Unternehmen werden stärker von diesem Projekt profitieren, da auch deren Dienstleistungen davon profitieren werden.

Durch die Einschränkung der Hardware auf wenige Hersteller entstehen für einen IoT Server und eine IoT SensorBox Hardwarekosten, die unter Umständen eine Einstiegshürde für Unternehmen oder Bildungsrichtungen darstellen könnten. Die Zielgruppe der Privatinteressierten, welche aktuell im Smarthome Bereich noch nicht so stark ausgeprägt sind, könnte damit komplett wegfallen.

Das Projekt wird Anwender (im Privatbereich) von Arduino, Raspberry und weiteren IoT nahen Bereichen gut ansprechen, welche bereits einen Großteil der Hardware besitzen. „Nerds“ bzw. „Hobby-Bastler“ werden aus „alten Projekten“ diese neue Idee umsetzen können, um deren „alte Hardware“, insbesondere den weitläufigen DHT22 Sensor für Temperatur und Luftfeuchtigkeit wieder neu einzusetzen.

Ein paar Worte aus didaktischer Sicht

Der Einflussfaktor Raumlufte wird von vielen Personen sehr unterschätzt. Das „Zusammenfassen / Reduzieren von Pausen“ wird zwar von Teilnehmern oftmals gern gesehen, doch laut Aufzeichnungen und Vergleich der Aufmerksamkeit der Teilnehmer ist es kontraproduktiv für Unterrichtssituationen. Die ähnlichen Resultate können in

Arbeitsumgebungen beobachtet werden. Meiner Meinung nach stellt das Projekt IoT AirClean die Lern- und Arbeitsgestaltung in ein neues bzw. alt bekanntes Licht, dass Pausen als wichtiger didaktischer Baustein gesehen und auch den Teilnehmern erklärt werden soll. Den Teilnehmern ist die Aufmerksamkeitskurve ein Begriff, doch wird diese selten in Verbindung mit der Raumluft gesehen.

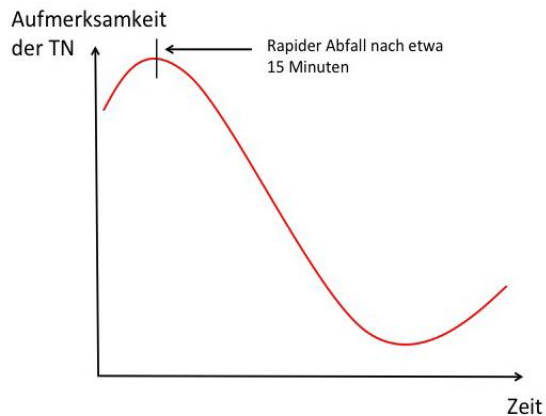


Abbildung 18: Aufmerksamkeitskurve²²

Unter Berücksichtigung der aktuellen Raumluft könnten Aufmerksamkeitskurven weniger steil abfallen und sich positiv auf viele Lebensbereiche auswirken. Um genauere Aussagen dazu treffen zu können, müssten weitere Studien recherchiert bzw. gegebenenfalls auch durchgeführt werden.

Weiteres Potential von IoT AirClean

Neben den bereits genannten Zielgruppen in der Anwendung, sehen wir auch großes Potential in der Gestaltung von Seminaren, Workshops und Bereitstellung von Konzepten für Informatikunterricht. Die umgesetzte Modularität des Projekts erlaubt es uns viele neue Bereiche, abseits des Projekts, mithilfe des Projekts zu beschreiben, darzustellen bzw. zu demonstrieren.

Der Start 2017 und die kommende Weiterentwicklung von IoT AirClean trifft den aktuellen Trend und Zeitgeist der Technik und besitzt Potential für die Zukunft. Somit werden wir dieses Projekt weiterhin verfolgen und unseren Vorstellungen sowie Rückmeldungen aus der Community, Testern, Interessierten und Anwendern weiterentwickeln.

²² <https://www2.tuhh.de/zll/leistungen/aktivierung/>

5 ANHANG

CSV Logs

analyse_2017*.xlsx

data_log_2017*.xlsx

Projektendergebnisse

Die Nummerierung orientiert sich am Projektcontrolling Dokument (Nr. Projektergebnis)

prj1684_Call11_03_Bauanleitung.pdf²³

prj1684_Call11_03_Bedienungsanleitung.pdf²⁴

prj1684_Call11_04_veröffentlichungsfähiger_Einseiter.pdf²⁵

Testdokumentation

Privatpersonen

prj1684_Call11_Testdokumentation_20170819.pdf

prj1684_Call11_Testdokumentation_20170930.pdf

prj1684_Call11_Testdokumentation_20171015.pdf

prj1684_Call11_Testdokumentation_20171126.pdf

Schulungen

prj1684_Call11_Testdokumentation_20170925.pdf

prj1684_Call11_Testdokumentation_20171002.pdf

prj1684_Call11_Testdokumentation_20171009.pdf

Usability Evaluierung

prj1684_Call11_Usability_Evaluierung_20180131.docx

Veröffentlichungsfähiger Einseiter

prj1684_Call11_04_veröffentlichungsfähiger_Einseiter.pdf²⁶

²³ Lizenz GNU GPLv3 sowie CC-BY Sharealike-3.0 AT

²⁴ Lizenz GNU GPLv3 sowie CC-BY Sharealike-3.0 AT

²⁵ Lizenz GNU GPLv3 sowie CC-BY Sharealike-3.0 AT

²⁶ Lizenz GNU GPLv3 sowie CC-BY Sharealike-3.0 AT