



# Air Solutions

Ein System zur Unterstützung  
des Luftqualitätsmanagements

## Projektarbeit 3.Semester

in der Weiterbildung zum  
Dipl. Techniker/in HF Informatik

**Auftraggeber:**

Pascal Rusca  
Kirchweg 2  
6048 Horw  
p.rusca@hotmail.com

**Projektleiter:**

Janik Schilter  
Chilenmattli 7  
6055 Alpnach Dorf  
janik.schilter@gmail.com

**Gutachter:**

Fachlehrer Projektmanagement  
Bruno Hammer  
teko@dasycon.ch



## 1 Management Summary

Gemäss dem Bundesamt für Gesundheit (BAG) ist die Qualität der Raumluft in einem Grossteil der Schweizer Schulen ungenügend. In einer Studie des BAG war die Luftqualität in zwei Dritteln der Schweizer Schulen zu schlecht.

Anfang Februar 2020 wurde die Schweiz von der Corona-Pandemie erfasst. Seitdem wurden diverse Massnahmen und Empfehlungen vom Bundesrat beschlossen, um das Ansteckungsrisiko in geschlossenen Räumlichkeiten zu verringern. Unter anderem wird vom Bundesrat regelmässiges Stosslüften empfohlen.

Diese Projektarbeit befasst sich mit dieser Problematik und ermöglicht eine praktische Lösung, nicht nur für Schulen, sondern auch für andere Räumlichkeiten wie beispielsweise Büro- und Schulungsräume.

Air Solutions ist ein System zur Unterstützung des Luftqualitätsmanagements. Mittels Sensoren im Raum wird die Luftqualität in regelmässigen Abständen gemessen. Die anwesenden Personen können über eine LED am Sensor die derzeitige Luftqualität ablesen und allenfalls die Räumlichkeiten lüften.

Das entwickelte System überzeugt durch intuitive Handhabung, grosse Erweiterungsmöglichkeiten und geringem Kostenaufwand. Die Wünsche und Erwartungen der Projektbeteiligten wurden erfüllt.

Nach Fertigstellung der Projektarbeit waren alle Musskriterien vollumfänglich erfüllt. Die Wunschkriterien wurden fast vollumfänglich erfüllt. Die Projektbeteiligten behalten sich vor, die restlichen Wunschkriterien zu gegebener Zeit, ausserhalb der Projektarbeit zu erfüllen.

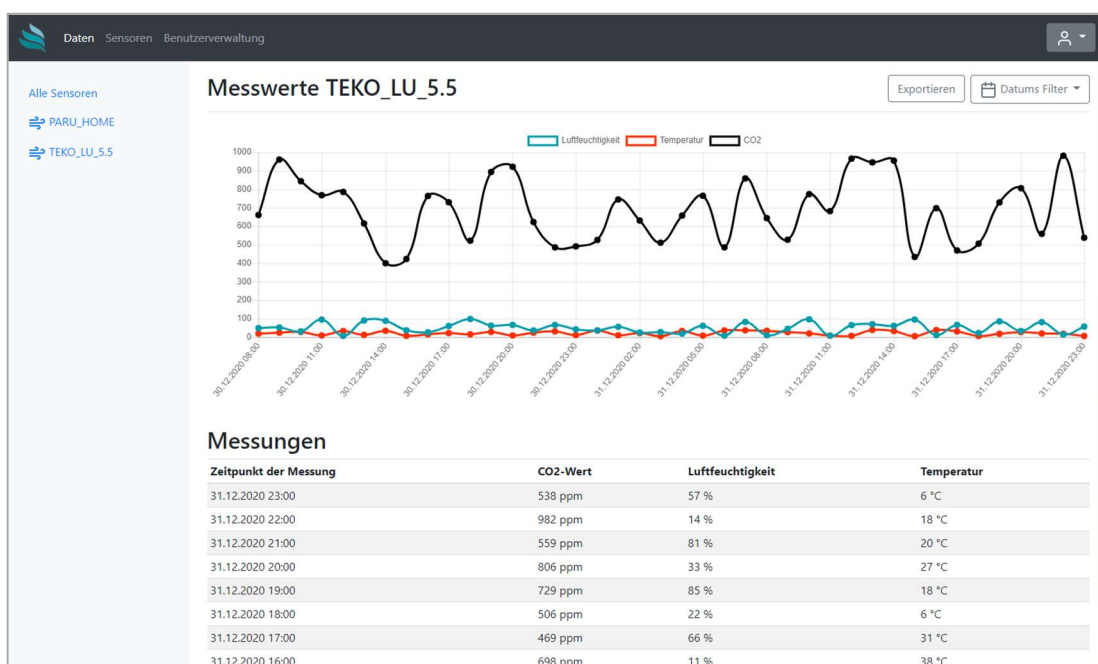


Abbildung 1 - Management Summary



## 2 Inhalt

1	Management Summary .....	1
3	Abbildungsverzeichnis .....	4
4	Einleitung .....	5
4.1	Vorwort .....	5
4.2	Erwartungen .....	5
4.3	Projektorganisation .....	5
4.4	Die Projektbeteiligten .....	6
5	Richtlinien / Aufgabenstellung .....	7
6	Analyse .....	8
6.1	Ist-Zustand an Schulen .....	8
6.2	Ist-Zustand in Schulungsräumen .....	8
6.3	Ist-Zustand in Büroräumlichkeiten .....	8
6.4	Ist-Zustand COVID-19 .....	8
6.5	Konzept .....	9
6.5.1	Funktionsübersicht .....	9
6.6	Vorstudie .....	10
6.7	Auswahl und Evaluierung .....	11
6.7.1	Sensor .....	11
6.7.2	Server .....	12
6.8	Termin- und Zeitmanagement .....	13
7	Pflichtenheft .....	14
7.1	Zielbestimmung .....	14
7.2	Produkteinsatz .....	14
7.3	Produktfunktionen .....	14
7.4	Qualitätsanforderungen .....	14
7.5	Benutzeroberfläche .....	15
7.6	Technische Produktumgebung .....	15
8	Server .....	16
8.1	Datenbank .....	16
8.1.1	Struktur .....	16
8.2	Webinterface .....	17
8.2.1	Navigationsleiste .....	17
8.2.2	Seite «Daten» .....	18
8.2.3	Seite «Sensoren» .....	19
8.2.4	Seite «Benutzerverwaltung» .....	21
8.3	API .....	22



8.3.1	Sensordaten speichern .....	22
9	Luftqualitäts-Sensor .....	23
9.1	Webinterface .....	23
9.2	Warnsignal .....	23
10	Testbericht / Auswertung .....	24
10.1	Testphase 1 .....	24
10.2	/ Testphase 2 .....	24
11	Schlussteil .....	25
11.1	Sachergebnisse .....	25
11.1.1	Ursprüngliche Aufgabenstellung und Ziele .....	25
11.1.2	Veränderungen der Aufgabenstellung / Zielsetzung .....	25
11.1.3	Erarbeitete Sachergebnisse .....	25
11.2	Projektverlauf .....	26
11.2.1	Überblick Kosten- und Zeitmanagement .....	26
11.3	Ausblick .....	28
11.3.1	Restaktivität .....	28
11.3.2	Ergänzungen und Erweiterungen .....	28
11.4	Schlusswort / Reflexion .....	29
11.4.1	Pascal Rusca .....	29
11.4.2	Janik Schilter .....	29
12	Anhang .....	30
12.1	Projektunterlagen .....	30
12.2	GIT-Repository .....	30
12.3	Quellenangaben .....	30
12.4	Sitzungsprotokolle .....	32



### 3 Abbildungsverzeichnis

Abbildung 1 - Management Summary	1
Abbildung 2 - Projektorganisation	5
Abbildung 3 – Pascal Rusca	6
Abbildung 4 - Janik Schilter	6
Abbildung 5 - Schule	8
Abbildung 6 - Schulungsräume	8
Abbildung 7 - Büros	8
Abbildung 8 - COVID19	8
Abbildung 9 - ESP-32 Dev-Board	11
Abbildung 10 - Ubuntu Logo	12
Abbildung 11 - Flask Logo	12
Abbildung 12 - Zeitplan komprimiert	13
Abbildung 13 - Struktur UML	16
Abbildung 14 - Navigationsleiste ausgeloggt	17
Abbildung 15 - Login-Popup	17
Abbildung 16 - Navigationsleiste eingeloggt	17
Abbildung 17 - Navigationsleiste Dropdown	17
Abbildung 18 - Seite Daten	18
Abbildung 19 - Messzeitraum	18
Abbildung 20 - Button Exportieren	18
Abbildung 21 - Seite Sensoren	19
Abbildung 22 - Popup Sensor hinzufügen	19
Abbildung 23 - Popup Sensor löschen	20
Abbildung 24 - Seite Benutzerverwaltung	21
Abbildung 25 - Popup Benutzer hinzufügen	21
Abbildung 26 - POST Request für Sensordaten	22
Abbildung 27 - Sensor Webinterface	23
Abbildung 28 - Zeitplan	26



## 4 Einleitung

Während der Weiterbildung zum Dipl. Techniker/in HF Informatik werden mehrere Projektarbeiten durchgeführt. Wir befinden uns zur aktuellen Zeit im 3. Semester unserer Weiterbildung und dürfen hiermit unsere zweite Projektarbeit durchführen. Ziel dieser Arbeit ist, ein Projektthema aus dem Themenbereich der Informatik zu wählen und bei der Durchführung unsere Kenntnisse in verschiedenen Bereichen der Weiterbildung zu festigen und erweitern.

Der inoffizielle Leitspruch dabei lautet: „Fehler machen – und aus Ihnen Lernen!“

### 4.1 Vorwort

Wir haben des Öfteren die Erfahrung gemacht, dass in Schulungs- und Weiterbildungsräumlichkeiten schlechte Luftqualität herrscht und häufig zu wenig gelüftet wird, um die Luftqualität zu verbessern.

Daher haben wir uns dazu entschieden, ein System zur Unterstützung des Luftqualitätsmanagement zu entwickeln. Dadurch können wir unsere Fachkenntnisse im Bereich der Elektronik, Datenverwaltung und Webdesign erweitern und vertiefen.

### 4.2 Erwartungen

Mit dieser Projektarbeit möchten wir eine Machbarkeitsstudie realisieren. Es geht darum, einen Prototyp zu entwickeln und unsere Kenntnisse zu erweitern.

### 4.3 Projektorganisation

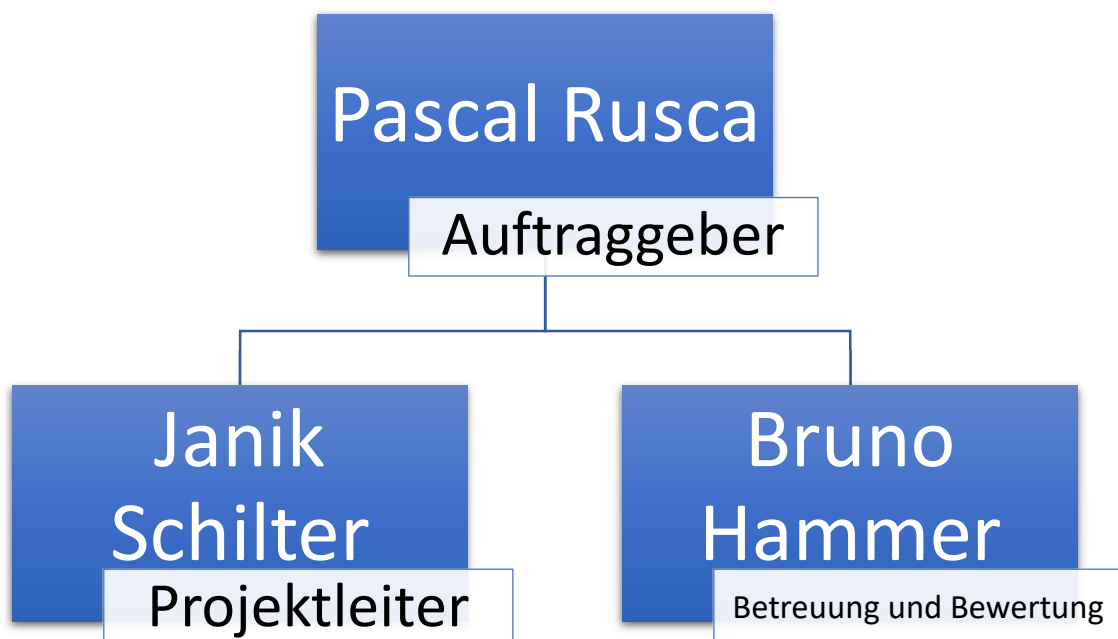


Abbildung 2 - Projektorganisation



## 4.4 Die Projektbeteiligten

### Auftraggeber

Pascal Rusca  
Kirchweg 2  
6048 Horw  
p.rusca@hotmail.com



*Abbildung 3 – Pascal Rusca*

---

### Projektleiter

Janik Schilter  
Chilenmattli 7  
6055 Alpnach Dorf  
janik.schilter@gmail.com



*Abbildung 4 - Janik Schilter*



## 5 Richtlinien / Aufgabenstellung

Untenstehend finden Sie eine Zusammenfassung der Richtlinien für die Projektarbeit  
Techniker HF Informatik 2020/2021:

**Umfang:** Die Richtzeit beträgt ca. 60 Stunden pro Studenten.

---

**Projektthema:** Themen aus einem der folgenden Gebiete:

- Software-Engineering: OOA – OOD – OOP
- Datenbank: Analyse – Design – Implementation
- IT/ICT Analyse – Design – Implementation
- IT-Sicherheit: Analyse – Design – Implementation

---

**Betreuung durch:** Bruno Hammer, Fachlehrer Projektmanagement

---

**Start Projektarbeit:** Montag, 16. November 2020

---

**Besprechungstermine:** Die Besprechungstermine je nach Bedarf auf Wunsch der Teams durchgeführt.

---

**Hilfsmittel:** Der Einsatz eines Versionsverwaltungssystem VCS ist obligatorisch.

---

**Abgabe der Arbeit:** Montag, 15. März 2021, 18:00 Uhr

---

**Präsentation:** Gruppe 1: Samstag 13. März 2021, 12:30 – 14:00 Uhr  
Gruppe 2: Samstag 27. März 2021, 12:30 – 14:00 Uhr

---

**Publikation:** Management Summary → wird im Internet publiziert  
Gesamte Arbeit → nur nach Absprache mit Betreuer

---





## 6 Analyse

### 6.1 Ist-Zustand an Schulen



Gemäss dem Bundesamt für Gesundheit (BAG) ist die Qualität der Raumluft in einem Grossteil der Schweizer Schulen unzureichend. In einer Studie des BAG war die Luftqualität in zwei Dritteln der Schweizer Schulen zu schlecht.

Abbildung 5 - Schule

### 6.2 Ist-Zustand in Schulungsräumen



Schulungsräumlichkeiten werden häufig von vielen verschiedenen Personen genutzt. Dabei entsteht das Problem, dass der vorgängige Nutzer des Schulungsraumes, am Ende seiner Schulung häufig nicht mehr lüftet und daher bereits zum Start der nächsten Schulung eine schlechte Luftqualität vorhanden ist.

Abbildung 6 - Schulungsräume

### 6.3 Ist-Zustand in Büroräumlichkeiten



Der problematischste Faktor in Büroräumlichkeiten ist, dass sich Personen häufig über längere Zeit darin aufhalten, häufig acht bis zehn Stunden am Tag. Insbesondere im Winter, oder auch im Hochsommer wird viel zu wenig gelüftet.

Abbildung 7 - Büros

### 6.4 Ist-Zustand COVID-19



Abbildung 8 -  
COVID19

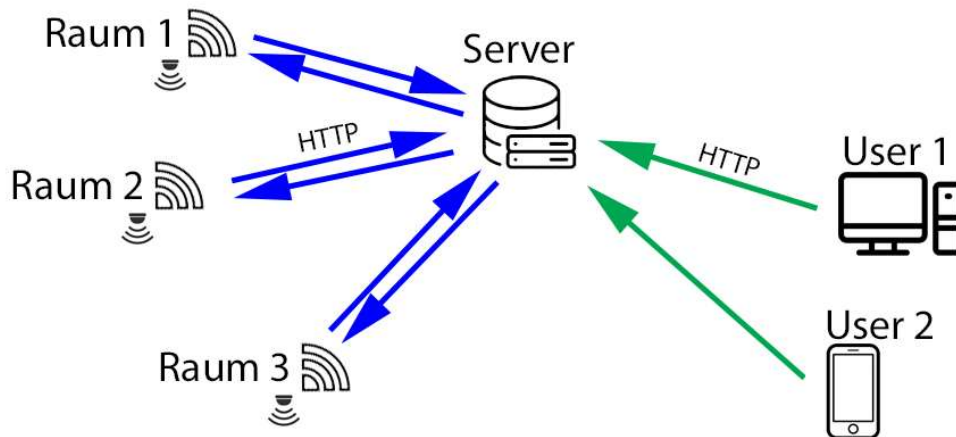
Anfang Februar 2020 wurde die Schweiz von der Corona-Pandemie erfasst. Seitdem wurden diverse Massnahmen und Empfehlungen vom Bundesrat beschlossen. Um das Ansteckungsrisiko in geschlossenen Räumlichkeiten zu verringern, wird vom Bundesrat regelmässiges und häufiges Stosslüften empfohlen.



## 6.5 Konzept

Air Solutions soll möglichst viele Räume überwachen können und die erfassten Daten an einem zentralen Ort zur Verfügung stellen. Um die Personen im Raum darauf hinzuweisen, dass die Luftqualität abnimmt, wird der Sensor mit einer LED ausgestattet, die je nach Farbe das Level der Luftqualität anzeigt.

### 6.5.1 Funktionsübersicht



Die Sensoren sollen an verschiedenen Orten montiert werden und verbinden sich per W-LAN mit dem Server. Auf diesem Weg werden die Daten auf dem Server in einer Datenbank gespeichert und können von den Usern aufgerufen werden.



## 6.6 Vorstudie

Die Luftqualität in einem geschlossenen Raum ist stark davon abhängig wie viele Personen sich darin befinden. Personen produzieren Gase, die für das menschliche Auge nicht sichtbar sind. Dies kann zu einer verringerten Denkleistung oder im schlimmsten Fall zu einer Vergiftung führen.

Die Luft besteht aus 78% Stickstoff, 21% Sauerstoff und zu 1% aus H<sub>2</sub>O, CO<sub>2</sub> und Spuren von anderen Gasen.

Sauerstoffgehalt in der Luft (%)	Auswirkungen auf den Menschen
19	Es können psychische Effekte auftreten, müssen aber nicht zwingen erkennbar sein.
15 – 19	Verschlechterung des Denkens und der Aufmerksamkeit. Erhöhter Puls und/oder Atmungsfrequenz. Reduzierte Koordinationsfähigkeit. Verringerte physikalische und intellektuelle Leistung.
12 – 15	Schlechtes Urteilsvermögen.
<10	Massive Einschränkung der Bewegungsfreiheit. Bewusstlosigkeit. Tod.

Da der Sauerstoff durch das Atmen kontinuierlich in Kohlenstoffdioxid umgewandelt wird, sinkt der Sauerstoffgehalt in einem Raum stetig.

Sauerstoff-Sensoren sind sehr teuer, jedoch gibt es noch sekundäre Indikatoren, die die Luftqualität bestimmen können. Dazu gehört auch CO<sub>2</sub>, wobei diese Sensoren ebenfalls teuer sind und langsamer reagieren.

Personen produzieren nicht nur CO<sub>2</sub> sondern auch H<sub>2</sub>, Ethanol und andere Gase in kleinen Mengen. Diese nennt man auch **Volatile Organic Compounds**, kurz VOC, was auf Deutsch übersetzt so viel bedeutet wie «Flüchtige organische Verbindung».

Diese Sensoren sind viel günstiger und reagieren schneller als Sauerstoff oder CO<sub>2</sub> Sensoren. Zusätzlich verbrauchen diese weniger Strom, womit bei einem mit Batterie betriebenen Gerät die Laufzeit verlängert werden kann.

Ebenfalls ist zu beachten, dass CO<sub>2</sub> schwerer ist als Sauerstoff und die Konzentration somit am Boden höher ist als an der Decke. Daher ist es wichtig die Messungen auf der richtigen Höhe vorzunehmen.



## 6.7 Auswahl und Evaluierung

### 6.7.1 Sensor

#### 6.7.1.1 VOC Sensor

Durch mehrere Recherchen im Internet sind wir auf viele verschiedene Sensoren gestossen, die es ermöglichen den VOC-Wert in der Luft zu messen. Wir haben uns für den Sensor «SGP30» auf einem Breakoutboard entschieden. Dieser kann per I2C angesteuert werden und liefert einen VOC Wert, wie auch einen errechneten CO<sub>2</sub> Wert, den wir für unser System verwenden werden.

#### 6.7.1.2 Steuerung

Da die Messwerte zentral ausgelesen werden müssen, werden diese an einen Server gesendet. Dies setzt eine Netzwerkverbindung voraus. Der Sensor soll möglichst frei platziert werden können, daher soll der Sensor W-LAN fähig sein. Zusätzlich muss die Steuerung über ein I2C-Interface und GPIOs verfügen, damit der ausgewählte Luftsensor verwendet und die LED-Anzeige realisiert werden kann. Die Steuerung soll möglichst leicht zu konfigurieren sein, um ein schnelles Aufsetzen des Sensors zu ermöglichen.

Verglichen haben wir zwei Geräte, die uns bereits bekannt sind. Der Minicomputer «Raspberry Pi Zero W» und der Mikrokontroller «ESP-32»

	Raspberry Pi Zero W	ESP-32
<b>Betriebssystem</b>	Raspberry Pi OS	RTOS
<b>WLAN</b>	Ja	Ja
<b>Cores</b>	1	2
<b>Speicher</b>	Abhängig von der SD-Karte	4 MB
<b>Stromverbrauch</b>	100-120 mA	80-170 mA
<b>Deep-Sleep</b>	Nein	Ja
<b>I2C</b>	Ja	Ja
<b>GPIO</b>	27	19
<b>RTC</b>	Nein	Ja



Abbildung 9 - ESP-32 Dev-Board

Wir haben uns für das ESP-32 entschieden, da es das Aufsetzen eines neuen Sensors deutlich vereinfacht. Da nur der Programmcode auf die Steuerung geladen werden muss und nicht ein ganzes Betriebssystem. Auch ist der «Deep-Sleep» praktisch, falls man den Sensor mit einer Batterie betreiben möchte.



### 6.7.1.3 LED

Für eine visuelle Referenz im Raum wird eine einfache RGB-LED verwendet.

### 6.7.1.4 Luftsensord

Um zusätzlich zum CO<sub>2</sub> Gehalt noch die Lufttemperatur und -feuchtigkeit messen zu können, haben wir uns für einen AM2320 Sensor entschieden, da wir diesen bereits vorrätig hatten. Dieser ermöglicht es uns noch mehr Werte zur aktuellen Luftqualität zu Messen und auf dem Server zu erfassen.

## 6.7.2 Server

Der Server wird auf einem bestehenden Server als virtuelle Maschine erstellt und betrieben. Der Serverstandort befindet sich in der Schweiz.

### 6.7.2.1 Betriebssystem



Abbildung 10 - Ubuntu Logo

Wir haben uns beim Betriebssystem des Servers für Ubuntu entschieden, da dieses zu einer der beliebtesten Linux-Distributionen gehört.

Dieses Betriebssystem gibt es auch in einer Server Version, bei dem kein grafisches Userinterface (GUI) dazu installiert wird. Hierbei wird hauptsächlich über die Kommandozeile administriert.

Dies spart Ressourcen, die ansonsten unnötig verschwendet werden würden.

Wir haben uns für die aktuelle Version «Ubuntu Server 20.04.01 LTS» entschieden.

### 6.7.2.2 Datenbank

Beim Datenbanksystem fiel unsere Wahl auf PostgreSQL der Version 12, da dieses System im Unterricht eingesetzt wird.

### 6.7.2.3 Web-Framework

Da unsererseits bereits Erfahrungen mit dem Web-Framework Flask vorhanden sind, wird dieses auch in diesem Projekt eingesetzt.



Abbildung 11 - Flask Logo

### 6.7.2.4 Styling

Um das Webinterface zu gestalten und dies gleichzeitig für Smartphones, Tablets und Computer bereitzustellen, wurde Bootstrap ausgewählt. Dieses Open Source Toolkit ermöglicht das schnelle Erstellen von responsiven Webseiten.



## 6.8 Termin- und Zeitmanagement

Eine komprimierte Version des Zeitplanes:

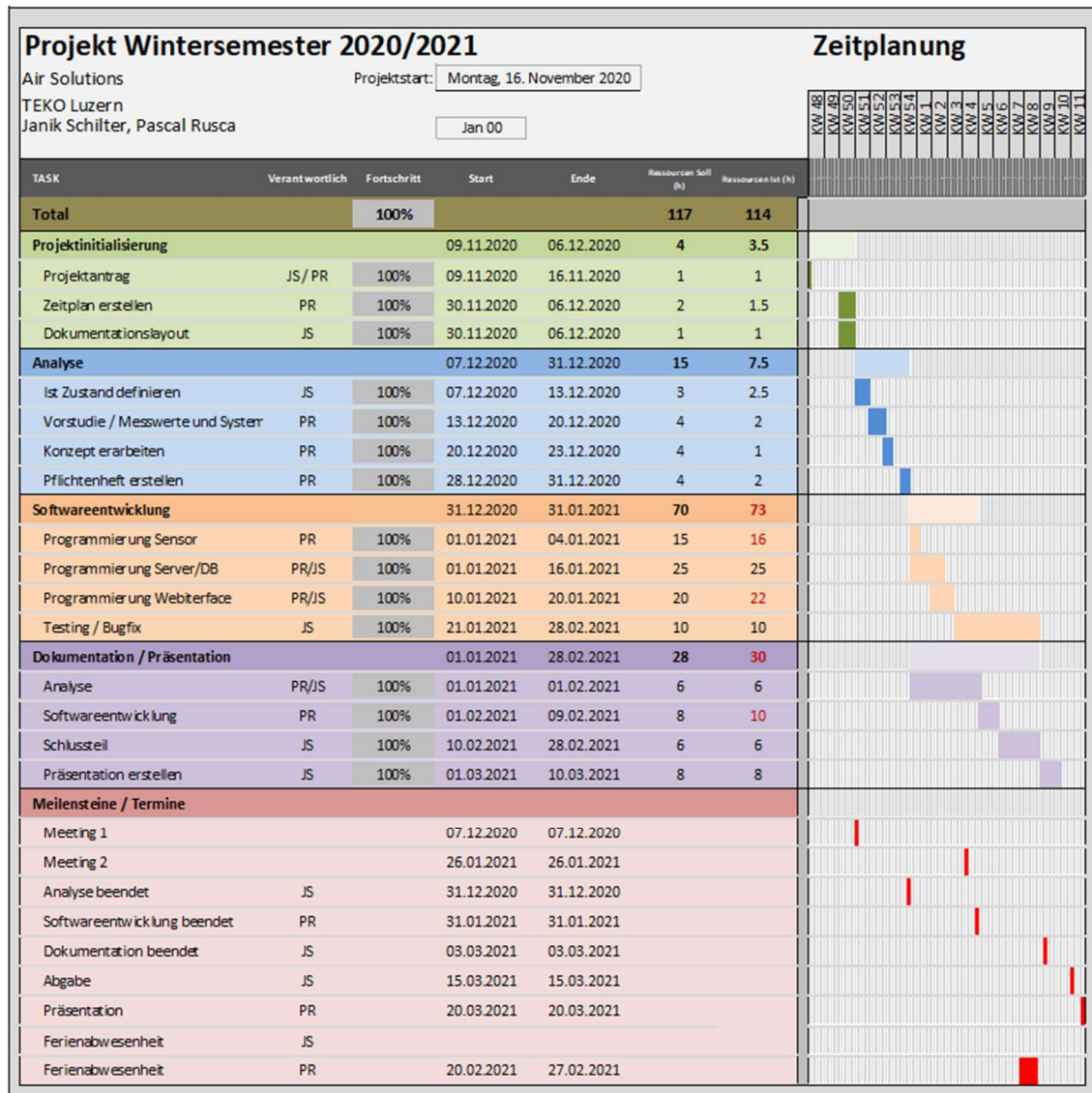


Abbildung 12 - Zeitplan komprimiert

Eine Auflistung der wichtigsten Meilensteine und Termine:

Meilensteine / Termine	Verantwortung	Start	Ende
Fertigstellung der Analyse-Phase	JS	31.12.2020	31.12.2020
Fertigstellung Softwareentwicklung	PR	31.01.2021	31.01.2021
Fertigstellung der Dokumentation	JS	03.03.2021	03.03.2021
Abgabe der Dokumentation	JS	15.03.2021	15.03.2021
Präsentation	PR	20.03.2021	20.03.2021



## 7 Pflichtenheft

---

### 7.1 Zielbestimmung

#### Musskriterien

- Anzeige eines Diagramms pro Sensor auf einer Webseite
- Alarmierung der Personen im Raum
- Sensor Management auf einer Webseite
- Kommunikation zwischen Sensor und Server via HTTP/HTTPS

#### Wunschkriterien

- Alarmierung von Bestimmten Usern via Mail und/oder Telegramm
- Usermanagement via Webseite
- Export und Download der Daten, zum Beispiel in einem CSV File
- Zusätzliche Werte (Temperatur, Luftfeuchtigkeit)

#### Abgrenzungskriterien

- In dieser Projektarbeit möchten wir einen Prototyp, respektive eine Machbarkeitsstudie realisieren. Es ist nicht Ziel, ein für den Markt fertiges Produkt zu erstellen.
- 

### 7.2 Produkteinsatz

#### Anwendungsbereiche

Das Air-Solutions System soll die Luftqualität der gemessenen Werte auf einer Webseite anzeigen und Personen darüber informieren wann es Zeit ist zu lüften oder eine Pause einzulegen.

#### Zielgruppen

Firmen, Schulen oder andere Institutionen, die die Luftqualität in geschlossenen Räumen überwachen möchten, um die Luftqualität auf einem angenehmen Level zu halten.

---

### 7.3 Produktfunktionen

#### Sensor

Der Sensor misst die Luftqualität im Raum und gibt diese an den Server weiter. Gleichzeitig informiert der Sensor die Personen im Raum mittels einer Farbigen LED über die Qualität der Luft.

#### Webinterface

Die gemessene Luftqualität soll für alle zugänglich sein, ohne sich vorher einloggen zu müssen. Für Administrations aufgaben gibt es einen mit Login gesicherten Bereich, den es ermöglicht Sensoren hinzuzufügen oder zu entfernen.

---

### 7.4 Qualitätsanforderungen

Die Luftqualität wird in regelmässigen Abständen gemessen, damit eine zeitnahe Rückmeldung über die LED möglich ist. Ausserdem entsteht durch die frequentierten Messungen ein rundes Gesamtbild in den auf dem Webinterface bereitgestellten Diagrammen.



---

## 7.5 Benutzeroberfläche

### Design

Das Design der App wird gemäss den Anforderungen und Wünschen des Kunden erstellt.

### Struktur

Die Struktur und der Aufbau der App wird gemäss den Anforderungen und Wünschen des Kunden erstellt.

---

## 7.6 Technische Produktumgebung

### Entwicklungsumgebung

- Visual Studio Code

### Serverbetriebssystem

- Ubuntu 20.04.01 LTS

### Webinterface

- Flask 1.1x
- Bootstrap v4.5





## 8 Server

### 8.1 Datenbank

Um das Entwickeln des Servers zu vereinfachen, wurde zuerst mit einer SQLite Datenbank gearbeitet. Dies ermöglicht das Verwalten des Codes zur Generierung der Datenbank über GIT, ohne das Verwenden eines Datenbank Servers.

Das FLASK Framework ermöglicht es, die Datenbankverbindung schnell zu ändern, ohne grosse Änderungen am bestehenden Code vorzunehmen.

#### 8.1.1 Struktur

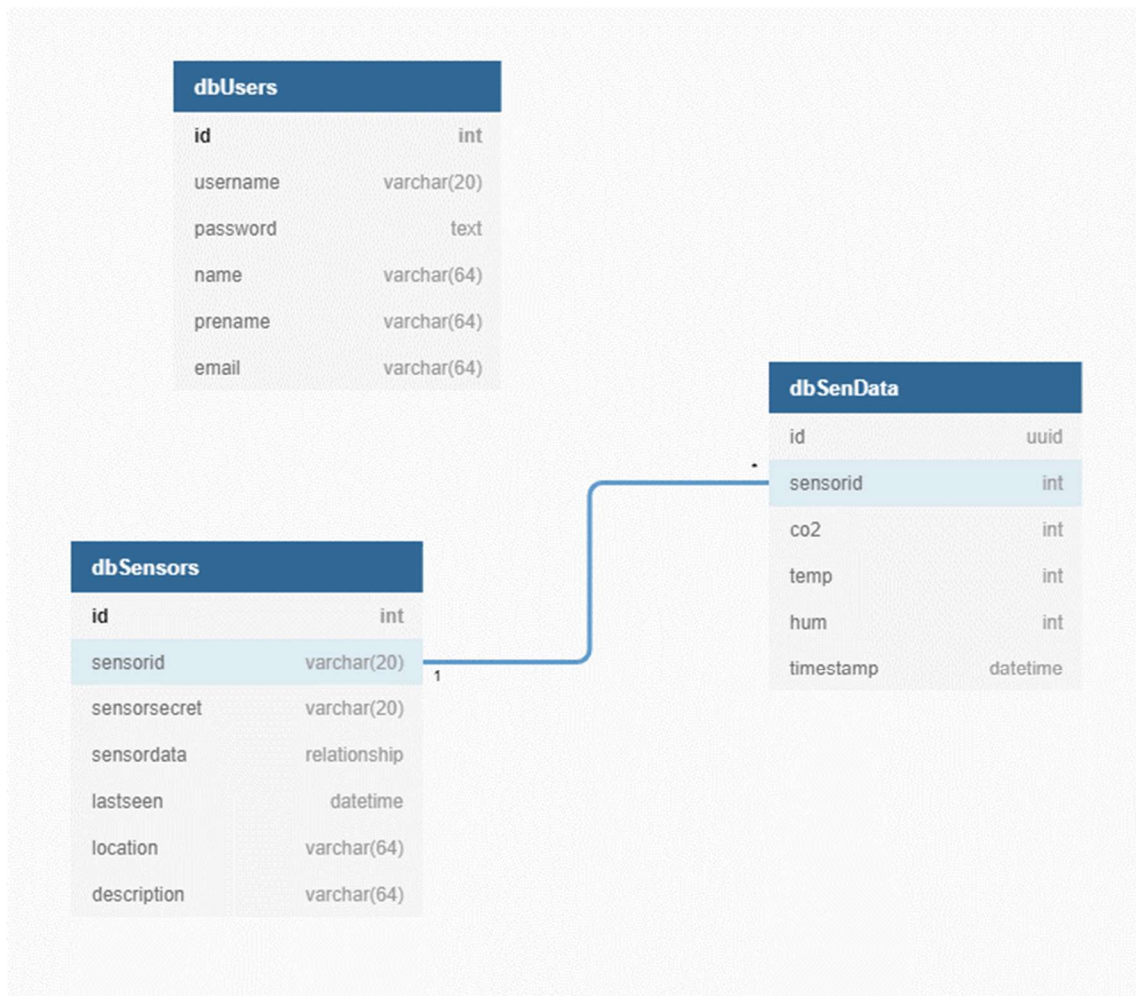


Abbildung 13 - Struktur UML



## 8.2 Webinterface

### 8.2.1 Navigationsleiste

Das Webinterface wurde möglichst einfach gehalten, um die Bedienung zu erleichtern.

Um alle Funktionen nutzen zu können, muss man sich auf dem Webinterface eingeloggt haben. Ansonsten wird man auf die Login Seite weitergeleitet.



Abbildung 14 - Navigationsleiste ausgeloggt

The image shows a login popup window titled 'Login' with a close button (X). It contains two input fields: 'Benutzername' and 'Passwort'. At the bottom right are two buttons: 'Abbrechen' (grey) and 'Login' (blue).

Um sich einzuloggen klickt man auf den Button «Login» und ruft hiermit ein Popup-Fenster auf. Hier werden nun Benutzername und Passwort angegeben.

Anschliessend klickt man auf den Button «Login» im Popup.

Abbildung 15 - Login-Popup

Sobald der Benutzer eingeloggt ist, hat er zusätzlichen Zugriff auf die Seiten «Sensoren» und «Benutzerverwaltung». Nachfolgend werden die einzelnen Seiten in der Dokumentation detaillierter beschrieben.



Abbildung 16 - Navigationsleiste eingeloggt

Über das Dropdown-Menü in der Navigationsleiste kann der Benutzer das Passwort ändern oder sich ausloggen.

Beim Ändern des Passwortes muss das bestehende Passwort eingegeben werden, ansonsten erscheint eine Fehlermeldung.

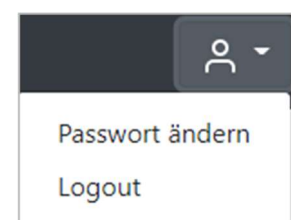


Abbildung 17 -  
Navigationsleiste  
Dropdown



### 8.2.2 Seite «Daten»

Auf der Seite «Daten» können die gemessenen Luftqualitätswerte der Sensoren eingesehen werden.

Die Messwerte werden als Liniendiagramm und als Tabelle zur Verfügung gestellt. Dabei hat der Benutzer die Möglichkeit zwischen den einzelnen Sensoren zu wechseln, oder über den Button «Alle Sensoren» alle Messwerte tabellarisch dargestellt einzusehen.

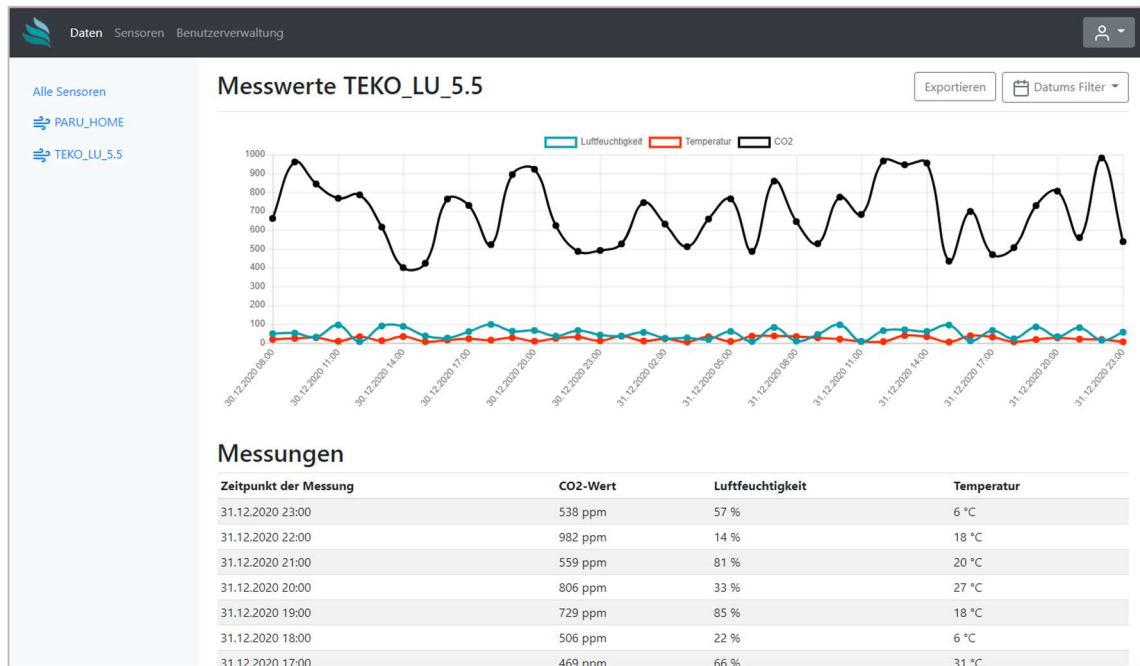


Abbildung 18 - Seite Daten

### Messzeitraum

The screenshot shows a 'Datums Filter' dropdown menu with two date options: '08.02.2021' and '15.02.2021'. Below the dates is a 'Filter anwenden' button.

Über das Dropdown-Menü «Datums Filter» kann der gewünschte Messzeitraum ausgewählt werden.

Abbildung 19 - Messzeitraum

### Sensordaten exportieren

Die Sensordaten werden als CSV Datei exportiert und können von jedem Besucher der Webseite heruntergeladen werden. Hierfür kann über den Dropdown-Button «Datums Filter» der gewünschte Zeitraum definiert werden. Somit werden nur die erfassten Messwerte innerhalb dieses Zeitraums im CSV File exportiert.

Exportieren

Abbildung 20 - Button Exportieren



### 8.2.3 Seite «Sensoren»

Sensor ID	Standort	Beschreibung	Sensor Secret	Letzter Kontakt	Bearbeiten
PARU_HOME	Horw	Dev Sensor	PW12345w!	17.02.2021 19:38	Löschen
TEKO_LU_5.5	TEKO Luzern Zimmer 5.5	An der Decke	PW12345w!	13.02.2021 12:50	Löschen

Abbildung 21 - Seite Sensoren

Auf der Seite «Sensoren» können die Sensoren verwaltet werden.

### Sensor Hinzufügen

Abbildung 22 - Popup Sensor hinzufügen

Es ist möglich über den Reiter «Sensoren» einen neuen Sensor hinzuzufügen.

Hierfür klickt man auf den Button «Sensor hinzufügen» und es öffnet sich ein Popup.

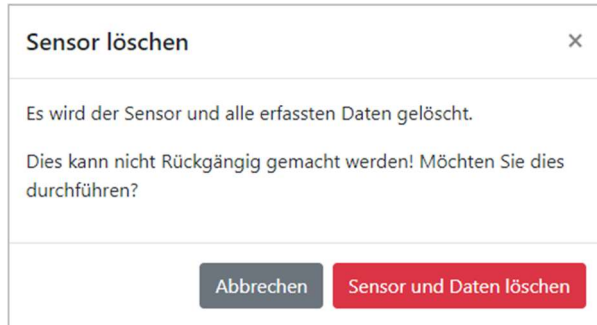
Hier können nun folgende Felder beschrieben werden:

Eingabefeld	Beschreibung
Sensor ID	Die Sensor ID ist die Bezeichnung des gewünschten Sensors.
Sensor Secret	Passwort mit dem sich der Sensor am Server anmeldet.
Standort	Beim Standort wird sinnvollerweise ein Gebäude, Stockwerk oder Raum definiert
Beschreibung	In der Beschreibung bietet es sich an, den Standort noch genauer zu beschreiben. Beispielsweise: «An der Decke montiert»

Sobald man die Informationen eingegeben hat, kann man dies mit «Sensor hinzufügen» bestätigen und der Sensor wird im System erfasst. Ab diesem Zeitpunkt ist es möglich, einen Sensor mit dem Server zu verbinden und Messdaten abzuspeichern.



## Sensor Löschen



Um einen bestehenden Sensor zu löschen, klickt man auf den Reiter «Sensoren» und wählt den Button «Löschen» beim gewünschten Sensor. Mit Hilfe eines Popups wird man nun gefragt, ob der Sensor und die dazu gehörigen Daten wirklich gelöscht werden sollen. Dies kann via Button bestätigt oder abgebrochen werden

Abbildung 23 - Popup Sensor löschen



## 8.2.4 Seite «Benutzerverwaltung»

Benutzername	Vorname	Name	E-Mail	Bearbeiten
admin				Löschen
paru	Pascal	Rusca		Löschen

Abbildung 24 - Seite Benutzerverwaltung

Auf der Seite «Benutzerverwaltung» können die Benutzer verwaltet werden.

### Benutzer hinzufügen

Benutzer hinzufügen

Benutzername

Passwort

Passwort wiederholen

Name (Optional)

Vorname (Optional)

E-Mail (Optional)

Abbrechen Benutzer hinzufügen

Über den Reiter «Benutzerverwaltung» ist es möglich, einen neuen Benutzer zu Erstellen.

Hierfür klickt man auf den Button «Benutzer hinzufügen» und ruft hiermit ein Popup auf. Hier werden nun Benutzername und Passwort definiert.

Nach Erstellung des Benutzers wird dieser im Reiter «Benutzerverwaltung» aufgeführt.

Abbildung 25 - Popup Benutzer hinzufügen

### Benutzer Löschen

Um einen bestehenden Benutzer zu löschen, klickt man auf den Reiter «Benutzerverwaltung» und wählt den Button «Löschen» beim gewünschten Benutzer. In einem Popup wird man nun gefragt, ob der Benutzer wirklich gelöscht werden soll und kann dies via Button bestätigen oder die Aktion abbrechen.

User löschen

Es wird folgender Benutzer gelöscht:

admin

Dies kann nicht Rückgängig gemacht werden! Möchten Sie dies durchführen?

Abbrechen User löschen



## 8.3 API

Um die Messdaten der Sensoren empfangen zu können, wurde eine einfache API (Application Programming Interface oder auch Programmierschnittstelle) aufgebaut, die ein einfaches Abspeichern der gemessenen Daten ermöglicht.

### 8.3.1 Sensordaten speichern

Die Daten werden per POST-Request im JSON-Format der Schnittstelle übergeben und vom Server verarbeitet.

Dazu muss der Sensor eine gültige Sensor-ID mit dem dazu passenden Sensor-Secret (Passwort) besitzen. Ansonsten werden die Daten nicht gespeichert und ein HTML Fehler-Code «403» zurückgegeben.

Die gesendeten Daten sehen wie folgt aus:

```
{
  "sensor_id": "PARU_HOME",
  "sensor_secret": "EinEmachtnochlangekeInCordonBleu!",
  "data": [
    {
      "co2": 426,
      "temp": 25,
      "hum": 40,
      "timestamp": "13.01.2021 23:00:00"
    },
    {
      "co2": 443,
      "temp": 25,
      "hum": 41,
      "timestamp": "13.01.2021 23:01:00"
    }
  ]
}
```

Abbildung 26 - POST Request für Sensordaten

API URL: <HOSTNAME>/sensors/api/savedata

Die erhaltenen Messdaten werden vom Server in der Datenbank abgespeichert. Ebenfalls wird ein Feld in der Sensor-Tabelle mit der Zeit des Abspeicherns aktualisiert, um unkompliziert prüfen zu können, wann die letzten Daten empfangen wurden oder ob der Sensor momentan offline ist oder ein Fehler mit der Netzwerkverbindung besteht.

Der Vorteil dieser Struktur besteht darin, dass der Sensor die Daten Lokal abspeichern kann und alle paar Minuten mehrere Messwerte gleichzeitig hochladen kann. Ebenfalls verringert dieser Ansatz den Netzwerkverkehr zwischen Sensor und Server.



## 9 Luftqualitäts-Sensor

### 9.1 Webinterface

Um möglichst aktuelle Werte von einem Raum zu erhalten ist es möglich, direkt per Browser den Sensor aufzurufen.

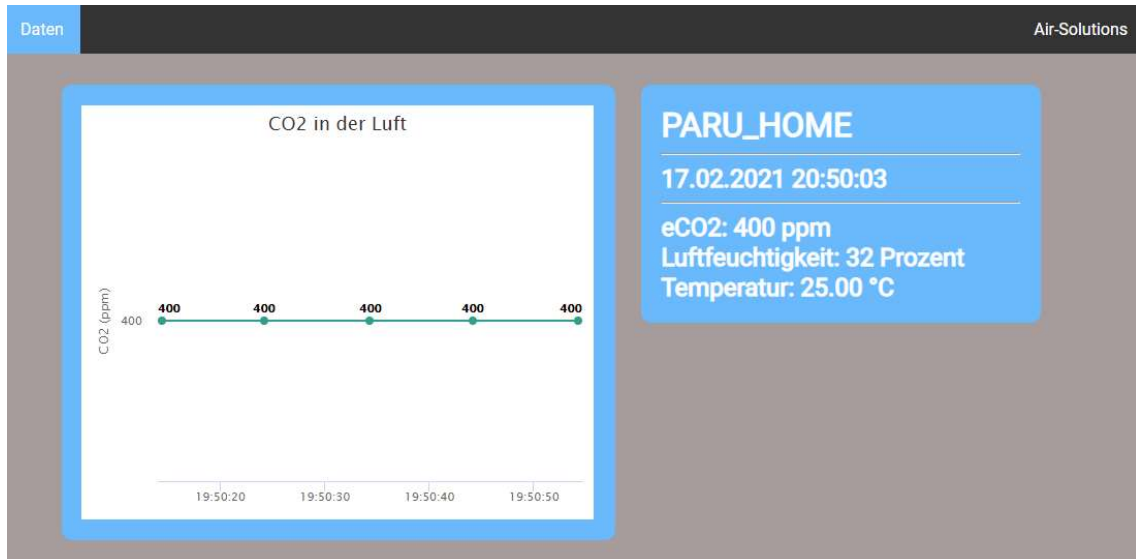


Abbildung 27 - Sensor Webinterface

Diese Webinterface ist dazu gedacht, die Funktionalität des Sensors nach der Installation oder einer Wartung sicher zu stellen.

### 9.2 Warnsignal

Damit die Personen in einem überwachten Raum nicht ständig einen Browser offen haben müssen, um die Werte im Auge zu behalten, verfügt der Sensor über eine LED die mithilfe von verschiedenen Farben den Status des CO<sub>2</sub> Gehaltes im Raum anzeigt.

Farbe	Bedeutung	CO <sub>2</sub> Wert (ppm)
Grün	Perfekte Luftqualität	Bis 499
Gelb	Gute Luftqualität	Ab 500
Orange	Lüften	Ab 1000
Rot	Schlechte Luftqualität	Ab 2000
Rot (Blinkend)	Sofort den Raum verlassen	Ab 3000

Diese Tabelle wurde mit Hilfe bestehender Richtlinien erstellt:

- ISS max. 5250 ppm (24h Durchschnitt)
- U-Boote max. 3500 – 4100 ppm (24h Durchschnitt)

Laut dem Umweltbundesamt (DE) sind CO<sub>2</sub> Werte in Schulzimmer unter 1000 ppm gesundheitlich unbedenklich. Werte von 1000 bis 2000 ppm sind auffällig und Werte über 2000 ppm sind gesundheitsgefährdend.





## 10 Testbericht / Auswertung

### 10.1 Testphase 1

Da in der ersten Testphase noch kein Sensor in Betrieb war, wurde ein Data-Generator-Skript erstellt. Dieses Skript ermöglichte die Simulation der Messdaten eines Sensors bereits vor Inbetriebnahme des ersten Sensors.

Mithilfe des Skripts konnten mehrere Bugs in der API und in der Darstellung auf dem Webinterface gefunden und behoben werden. Ebenfalls konnte die Leistung des Servers evaluiert werden und wir kamen zum Schluss, dass die gemessenen Daten gebündelt an den Server gesendet werden müssen, um die Performance zu verbessern.

### 10.2 / Testphase 2

In der zweiten Testphase wurde das fertige System mit einem Sensor in der Wohnung des Auftraggebers getestet.

Dabei wurde klar, dass der ausgewählte Sensor für diesen Einsatzzweck nur bedingt geeignet ist, da der Sensor nur einen CO<sub>2</sub>-Wert errechnet und durch andere Stoffe in der Luft getäuscht werden kann. Würde dieses Produkt in sicherheitsrelevanten Anlagen eingesetzt werden, müsste zu einem CO<sub>2</sub>-Sensor gewechselt werden. Dies kann durch eine einfache Anpassung im Sensor-Code bewerkstelligt werden.

Ebenfalls hat sich gezeigt, dass das Verwenden der integrierten RTC des ESP-32 ohne eine Absicherung mit einer Batterie suboptimal ist, da nach einem Ausfall des Elektrizitätsnetzes auch das W-LAN nicht erreichbar sein wird. Daher würde beim Aufstarten die Zeit nicht mit einem Zeitserver synchronisiert werden können. Dies führt zu Messdaten mit verfälschten Zeit- und Datumsangaben. Für einen produktiven Einsatz wäre ein batteriebetriebener Sensor besser geeignet.

Eine Speichererweiterung wäre ebenfalls eine sinnvolle Erweiterung. Damit könnte man länger Daten auf dem Sensor zwischenspeichern und der Sensor könnte eine längere Zeit ohne eine Netzwerkverbindung auskommen, ohne Messdaten zu verlieren.

Auch müsste die Implementierung des Diagramms auf dem Webinterface noch angepasst werden, damit ein leerer Zeitbereich angezeigt wird, wenn keine Messdaten vorhanden sind.



## 11 Schlussteil

### 11.1 Sachergebnisse

#### 11.1.1 Ursprüngliche Aufgabenstellung und Ziele

Das Ziel dieser Arbeit war, ein System zur Unterstützung des Luftqualitätsmanagements zu erschaffen, welches helfen soll, die Luftqualität im Raum zu verbessern.

#### Musskriterien

- Anzeige eines Diagramms pro Sensor auf einer Webseite
- Alarmierung der Personen im Raum
- Sensor Management auf einer Webseite
- Kommunikation zwischen Sensor und Server via HTTP/HTTPS

#### Wunschkriterien

- Alarmierung von Bestimmten Usern via Mail und/oder Telegramm
- Usermanagement via Webseite
- Export und Download der Daten, zum Beispiel in einem CSV File
- Zusätzliche Werte (Temperatur, Luftfeuchtigkeit)

#### Abgrenzungskriterien

- In dieser Projektarbeit möchten wir einen Prototyp, respektive eine Machbarkeitsstudie realisieren. Es ist nicht Ziel, ein für den Markt fertiges Produkt zu erstellen.

---

#### 11.1.2 Veränderungen der Aufgabenstellung / Zielsetzung

Alle Musskriterien konnten erfüllt werden. Somit mussten die Aufgabenstellung und die Zielsetzung nicht verändert werden.

Die Wunschkriterien wurden, abgesehen von folgendem Ziel, alle erreicht:

- Alarmierung von Bestimmten Usern via Mail und/oder Telegramm
- 

#### 11.1.3 Erarbeitete Sachergebnisse

Das entwickelte System zur Unterstützung des Luftqualitätsmanagements entspricht den im Pflichtenheft definierten Kriterien. Die gemessenen Daten werden als Diagramm auf der Webseite dargestellt, die Personen im Raum werden mittels LED über die derzeitige Luftqualität informiert, die Sensoren können auf der Webseite verwaltet werden und die Kommunikation zwischen Sensor und Server funktioniert via HTTP/HTTPS.

Ausserdem kann das Usermanagement via Webseite getätigt werden, die Messwerte als CSV-File exportiert werden und zusätzliche Werte wie Temperatur und Luftfeuchtigkeit gemessen und ausgewertet werden.



## 11.2 Projektverlauf

### 11.2.1 Überblick Kosten- und Zeitmanagement

#### Zeitmanagement:

Folgend ein Ausdruck aus unserem nachgeführten Zeitplan:

TASK	Verantwortlic h	Fortschritt	Start	Ende	Ressourcen Soll (h)	Ressourcen Ist (h)
<b>Total</b>		<b>100%</b>			<b>117</b>	<b>114</b>
<b>Projektinitialisierung</b>			09.11.2020	06.12.2020	<b>4</b>	<b>3.5</b>
Projektantrag	JS / PR	100%	09.11.2020	16.11.2020	1	1
Zeitplan erstellen	PR	100%	30.11.2020	06.12.2020	2	1.5
Dokumentationslayout	JS	100%	30.11.2020	06.12.2020	1	1
<b>Analyse</b>			07.12.2020	31.12.2020	<b>15</b>	<b>7.5</b>
Ist Zustand definieren	JS	100%	07.12.2020	13.12.2020	3	2.5
Vorstudie / Messwerte und Syst	PR	100%	13.12.2020	20.12.2020	4	2
Konzept erarbeiten	PR	100%	20.12.2020	23.12.2020	4	1
Pflichtenheft erstellen	PR	100%	28.12.2020	31.12.2020	4	2
<b>Softwareentwicklung</b>			31.12.2020	31.01.2021	<b>70</b>	<b>73</b>
Programmierung Sensor	PR	100%	01.01.2021	04.01.2021	15	16
Programmierung Server/DB	PR/JS	100%	01.01.2021	16.01.2021	25	25
Programmierung Webinterface	PR/JS	100%	10.01.2021	20.01.2021	20	22
Testing / Bugfix	JS	100%	21.01.2021	31.02.2021	10	10
<b>Dokumentation / Präsentation</b>			01.01.2021	28.02.2021	<b>28</b>	<b>30</b>
Analyse	PR/JS	100%	01.01.2021	01.02.2021	6	6
Softwareentwicklung	PR	100%	01.02.2021	09.02.2021	8	10
Schlussenteil	JS	100%	10.02.2021	28.02.2021	6	6
Präsentation erstellen	JS	100%	01.03.2021	10.03.2021	8	8

Abbildung 28 - Zeitplan

Wie in der [Abbildung 28 - Zeitplan](#) zu sehen ist, haben wir unsere mit 117 Stunden budgetierte Zeit fast genau erreicht. Unser effektiver Aufwand hat 114h betragen.

#### Projektinitialisierungsphase

In der Projektinitialisierungsphase benötigten wir 0.5h weniger als vorgesehen. Dies lag daran, dass wir aus unserem letzten Projekt bereits Vorlagen für die zu erstellenden Dokumente hatten.

#### Analysephase

In der Analysephase hatten wir nur 7.5h Aufwand, statt den budgetierten 15h Aufwand. Der Grund hierfür war, dass die Informationsbeschaffung sehr einfach ausfiel, da im Internet viele der benötigten Informationen einfach zu finden waren.



### Softwareentwicklungsphase

Für die Softwareentwicklung wurden vorgängig 70h budgetiert. Der effektive Aufwand belief sich auf 73h. Da wir aber in den vorherigen Projektphasen weniger Zeit benötigten als budgetiert, konnten wir diese Aufwände querkompensieren.

### Dokumentationsphase

Für die Dokumentationsphase waren 28h Aufwand vorgesehen. Der Effektivaufwand hat 30h betragen. Auch hier waren wir in der Lage mit der gewonnenen Zeit aus den ersten Phasen zu kompensieren. Der Mehraufwand in der Dokumentationsphase ist damit zu begründen, dass wir grossen Wert auf Layout und Design der Projektdokumentation gelegt haben und zum Beispiel auch ein Logo entwickelt haben.

### Start- und Enddaten

Alle Termine konnten gemäss den Vorgaben aus den «Richtlinien für die Projektarbeit Techniker HF Informatik 2020/2021» eingehalten werden.

### Kostenzusammenstellung Honorar:

Kostenpunkt	Budgetiert	Kosten	Eingehalten
Honorar Projektleiter	CHF 14'400 exkl. MwSt.	CHF 13'680 exkl. MwSt.	Ja

### Kostenzusammenstellung Material:

Produkt	Hersteller	Typ	Anzahl	Stückpreis	Preis
Breakoutboard				CHF 2.00	CHF 2.00
Mikrokontroller	Espressif	ESP-32	2	CHF 3.00	CHF 6.00
VOC-Sensor	Sensirion	SGP30	2	CHF 6.30	CHF 12.60
RGB-LED			2	CHF 5.00	CHF 10.00
Server	Self Hosted	VM	1	-	-
				<b>Total</b>	<b>CHF 20.60</b>

### Planungsqualität

Die Planung wurde zu Beginn des Projektes detailliert und mit genügend Freiraum erarbeitet, um für allfällige Abweichungen einen Puffer zu haben. Daher konnten wir uns sehr auf die geplanten Termine verlassen und diese auch ohne grössere Abweichungen einhalten.

Da viele Dokumente von der bevorstehenden Projektarbeit übernommen werden konnten, hat das Erstellen der Dokumentation weniger Zeit beansprucht als noch im Vorprojekt.



## 11.3 Ausblick

### 11.3.1 Restaktivität

Innerhalb dieser Projektarbeit sind ausser der Präsentation am Samstag 20. März 2020 keine Restaktivitäten eingeplant.

### 11.3.2 Ergänzungen und Erweiterungen

Aus unserer Sicht sind folgende Ergänzungen bzw. Erweiterungen des Projektes möglich:

- Sensor Webinterface (Sensor Einstellungen)
- Alarmierung von Bestimmten Usern via Mail und/oder Telegramm
- Darstellung der Sensorstandorte mittels Grundrissplan im Webinterface
- Verbesserte Kompatibilität mit mobilen Endgeräten (Webinterface)
- Ersatz von VOC-Sensor durch CO2-Sensor
- Spannungsversorgung der Sensoren mittels Batterie (Ausfallsicherheit)
- Speichererweiterung für den Sensor



## 11.4 Schlusswort / Reflexion

### 11.4.1 Pascal Rusca

Mithilfe dieses Projektes konnte ich meine Kenntnisse im Bereich Python Flask und das Arbeiten mit einer Datenbank verbessern. Auch gelerntes vom Unterricht hat im Projekt sehr geholfen, zum Beispiel der Werteüberlauf eines Integer. Des Weiteren konnte ich meine Fähigkeiten im Bereich des Kommentierens verbessern, um diese verständlicher zu gestalten.

Ich hatte grosse Probleme, mich auf einen Bereich zu konzentrieren und den Server methodisch zu programmieren. Dies konnte ich dann beim Programmieren des Sensors besser umsetzen und dieser konnte effizienter programmiert werden. Auch konnte ich das Zeitmanagement bei diesem Projekt besser umsetzen als beim Vorprojekt.

Dank des, bereits sehr etablierten Online Unterrichts, konnte das Projekt auch trotz der COVID-19 Pandemie ohne Probleme durchgeführt werden. Mithilfe von regelmässigen Team-Calls konnte effizient zusammengearbeitet werden. Ich war immer sehr gut über den aktuellen Stand der Arbeiten informiert und wir konnten dadurch die nächsten Schritte planen.

### 11.4.2 Janik Schilter

Meine bisherigen Kenntnisse in den Bereichen Datenbank, Webdesign und Elektronik waren sehr begrenzt. Daher war es mein Ziel, mit dieser Projektarbeit meine Kenntnisse in diesen Bereichen zu verbessern.

Zu Beginn der Projektarbeit hatte ich grössere Schwierigkeiten, da meinerseits schlichtweg zu wenig Know-How vorhanden war. Aber Mithilfe von diversen Websites, lehrreichen Gesprächen im Projektteam und mit ausgebildeten Informatikern, war es mir möglich, das nötige Wissen für die zu erledigenden Arbeiten anzueignen.

Die Zusammenarbeit im Projektteam empfand ich als sehr angenehm und unkompliziert. Durch regelmässige Projektbesprechungen im Microsoft Teams konnten wir trotz Corona-Krise eine einwandfreie Kommunikation und Zusammenarbeit gewährleisten.



## 12 Anhang

### 12.1 Projektunterlagen

Alle Projektunterlagen sind im ELAD unter folgendem Link abgelegt:

<https://elad.ch/olat/auth/RepositoryEntry/83820552/CourseNode/102665760791044>

---

### 12.2 GIT-Repository

Der Quellcode inklusive Readme-Datei ist auf dem GIT-Repository unter folgender URL abgelegt:

<https://elad.ch/gitblit/summary/L-TIN-19-T-a!3-SA!Team-06.git>

---

### 12.3 Quellenangaben

Zusammensetzung der Luft:

<https://www.grc.nasa.gov/www/k-12/airplane/airprop.html#:~:text=Air%20is%20a%20mixture%20of,from%20all%20the%20individual%20components.>

Einfluss der Luftqualität auf die Arbeitsleistung:

[https://www.seco.admin.ch/dam/seco/de/dokumente/Arbeit/Arbeitsbedingungen/Gesundheitsschutz%20am%20Arbeitsplatz/Arbeitsr%C3%A4ume%20und%20Umgebungsfaktoren/VW\\_Arbeitsleistung-und-Raumluft.pdf.download.pdf/VW\\_Arbeitsleistung-und-Raumluft\\_de.pdf](https://www.seco.admin.ch/dam/seco/de/dokumente/Arbeit/Arbeitsbedingungen/Gesundheitsschutz%20am%20Arbeitsplatz/Arbeitsr%C3%A4ume%20und%20Umgebungsfaktoren/VW_Arbeitsleistung-und-Raumluft.pdf.download.pdf/VW_Arbeitsleistung-und-Raumluft_de.pdf)

BAG-Studie:

<https://www.bag.admin.ch/bag/de/home/das-bag/aktuell/news/news-04-03-2019.html>

Erstellung Logo:

<https://www.logomaker.com/de/>



Abbildungen:

- Abbildung 4:  
<https://palmade.mon-ent-occitanie.fr/vie-de-l-etablissement/espace-numerique-de-travail-du-college-jules-palmade-106.htm>
- Abbildung 5:  
<https://www.busseltonttc.com/>
- Abbildung 6:  
<https://homedelivery.com.ng/?SA>
- Abbildung 7:  
<https://pixabay.com/ko/illustrations/covid-19-%EC%BD%94%EB%A1%9C%EB%82%98-%EB%AC%B4%EB%8A%AC-5222513/>
- Abbildung 9:  
<https://design.ubuntu.com/downloads/>
- Abbildung 10:  
<https://flask.palletsprojects.com/>





## 12.4 Sitzungsprotokolle

---

### Kickoff-Sitzung Softwareentwicklung vom 08.12.2020

---

**Teilnehmer:** Janik Schilter, Pascal Rusca

#### Themen

- Systemvoraussetzungen
- Besprechung Konzeptionelles
- Aufgabenteilung

#### Systemvoraussetzungen

- Ubuntu WSL installieren (im Microsoft Store) (Linux Subsystem für Windows)
- Vs Code auf Laptop installieren
- Im Vs Code: REST Client installieren
- Flask installieren
- <https://www.w3schools.com/>  
--> Learn Bootstrap
- [getbootstrap.com/docs/4.4/wxamples/dashboard/](https://getbootstrap.com/docs/4.4/wxamples/dashboard/)

#### Aufgabenteilung

To do jasc:

- Diagramm
- Designteil Website (Bootstrap)

To do paru:

- DB aufsetzen
- Sensoren bestellen

#### Sonstiges

- Ist-Zustand: Corona --> in Doku einfließen lassen



---

## Projektsitzung vom 26.01.2021

---

**Teilnehmer:** Janik Schilter, Pascal Rusca

### Themen

- Stand der Arbeiten
- Server, was gibt's noch zu tun
- Sensor, was gibt's noch zu tun
- Start Doku?

#### Sensor

- ☒ Code optimieren / Kommentieren
- ☒ Code aufräumen
- ☐ Gehäuse (Ideen?)
- ☐ RGB LED

#### Server (Backend)

- ☒ Data Table PK to UUID
- ☒ Diagramm letzten werte ganz rechts
- ☒ SensorId aus Tabelle entfernen (Messwerte für einzelnen Sensor)
- ☒ Sensorseite Zeit Format ändern
- ☒ Benutzerverwaltung:
  - ☒ Vorname Name (optional)
  - ☒ Mail Adresse (optional)

#### Server (Frontend)

- ☒ Kontodropdown rechtsbündig, Design Verbessern
- ☒ Datumsfilter rechtsbündig, Design Verbessern
- ☐ Windsymbol (verschwinden bei zu kleinem Display (nichts geändert))
- ☒ Scaling Diagramm
- ☐ Menu für Smartphones (weglassen)
- ☐ Farbwahl Tabellen --> Wir müssten das bootstrap.css im Projekt einfügen (weglassen)
- ☒ Kennwort hinweise im Modal

### Nächste Termine

- 07.02.2021 Anpassungen durchgeführt
- 08.02.2021 Dokumentation
- 08.02.2021 Meeting 3 19:00 Teams



---

## Projektsitzung vom 10.02.2021

---

**Teilnehmer:** Janik Schilter, Pascal Rusca

### Themen

- Stand der Arbeiten
- Server, was gibt's noch zu tun
- Sensor, was gibt's noch zu tun
- Dokumentation

### Sensor

- ☐ RGB LED
- ☐ Gehäuse

### Server (Backend)

- ☒ DB installieren

### Server (Frontend)

- ☒ Kontodropdown: Logout unten
- ☒ Deploy on Server

### Dokumentation

Fertigstellen



---

## Projektsitzung vom 03.03.2021

---

### Themen

- Stand der Dokumentation
- Präsentation

### Sensor

- ☐ Gehäuse

### ToDo Paru

- ☒ Abbildungs Quelle ergänzen
- ☒ Planungs Qualität anpassen
- ☒ Doku durchlesen + anpassen

### To do jasc:

- ☒ Abbildungs Quelle ergänzen
- ☒ 11.3.2 Ergänzung und Erweiterungen von Testing
- ☒ Doku durchlesen + anpassen

### Weitere Aufgaben:

- ☐ Alle Dokumente auf ELAD Laden
- ☐ Git transferieren auf Git von Bruno
- ☐ Start Presentation

### Nächste Besprechung:

- 08.03.2021 19:00



---

## Projektsitzung vom 08.03.2021

---

### Themen

- Stand Dokumentation
- Präsentation

### Sensor

- ☐ Gehäuse

### Präsentation: (max. 12min)

#### Folien:

1. Titelblatt PARU
2. Ablauf
3. Motivation (Ist-Zustand) JASC
4. Ziele JASC
5. Systemübersicht (Sensor, DB, Webinterface) JASC
6. Kurzübersicht Sensor PARU
7. Kurzübersicht API+DB PARU
8. Kurzübersicht Webinterface (via Browser) JASC
9. Reflexion (z.B. anderer Sensor) PARU/JASC
10. Ausblick PARU

### Weitere Aufgaben:

- ☐ Alle Dokumente auf ELAD Laden
- ☐ Git transferieren auf Git von Bruno