

Die hard- und softwaretechnische Implementierung eines CO₂-Sensors zur Messung der Raumluftqualität

Julius Caesar, Péter Egermann, Paul Görtler, Johannes Leyrer

5. Mai 2022

Inhaltsverzeichnis

1	Einleitung	4
2	CO₂-Grenzwerte und deren Auswirkungen auf den Menschen	5
2.1	CO ₂ -Grenzwerte für eine unbedenkliche Atemluft	5
2.2	Physiologische Auswirkungen eines zu hohen CO ₂ -Gehaltes in der Atemluft	6
3	Hardwaretechnische Umsetzung	8
3.1	Technische Anforderungen an die benötigte Hardware	8
3.2	Überblick über die verwendete Hardware	8
4	Softwaretechnische Umsetzung	9
4.1	Benötigte Software	9
4.2	Zusammenspiel der Softwarekomponenten	9
4.3	Aufbau und Einrichten der Software	9
4.3.1	Aufbau und Einrichten des Backends	9
4.3.2	Aufbau und Einrichten der Lese-Software	11
4.3.3	Aufbau und Einrichten des Frontends	12
5	Zusammenfassung	14
	Abbildungsverzeichnis	15
	Listings	16
	Glossar	17
	Anhang	21

Abkürzungsverzeichnis

API	Application Programming Interface	11
ASR	Technische Regeln für Arbeitsstätten	5
DGUV	Deutsche Gesetzliche Unfallversicherung	5
HIDs	Human Interface Devices	18
ppm	parts per million	5
ZNS	Zentralnervensystem	6

1 Einleitung

Im Laufe der Corona-Pandemie und der damit verbundenen Ausgangsbeschränkungen musste man sich zunehmen in Innenräumen aufhalten, um der Verbreitung des Virus entgegenzuwirken. Aus diesem Grund musste man sich zwangsweise mit der Luftqualität im Homeoffice und besonders in Büroräumen auseinandersetzen, um das Gesundheitsrisiko zu minimieren und konzentriert arbeiten zu können. Da die Luftqualität durch Menschen meist nur subjektiv wahrgenommen werden kann, benötigt man dafür eine Messstelle. Da die gesundheitlichen Auswirkungen von schlechter Raumluftqualität ausreichend erforscht sind, existieren genügend konkrete Werte zur Orientierung.

Diese Arbeit soll einen Überblick darüber geben, welche medizinischen Risiken durch eine unzureichende Luftqualität entstehen können und welche positiven Auswirkungen eine ausreichende Versorgung mit Frischluft haben kann.

Danach soll ein grober Leitfaden zur Einrichtung eines CO₂-Sensors zur Bestimmung der Luftqualität in Arbeitsräumen erstellt werden. Dabei soll über die Anforderungen an die Hardware, die benötigten Komponenten, die Einbindung in das Netzwerk und die Umsetzung der dazugehörigen Software eingegangen werden. Die Arbeit richtet sich dabei hauptsächlich an technisch versierte Leser*innen, welche bereits grundlegende Kenntnisse in den Bereichen Hard- und Software besitzen. Grundsätzlich wird versucht, auch Laien die Grundlagen und die allgemeine Herangehensweise nahe zu legen.

2 CO₂-Grenzwerte und deren Auswirkungen auf den Menschen

Um sich der Wichtigkeit einer Messung des CO₂-Gehaltes in der Atemluft bewusst zu werden, werden in diesem Kapitel die von verschiedenen Institutionen festgelegten CO₂-Grenzwerte zusammengefasst.

Weiterhin werden die verschiedenen physiologischen Risiken, die durch einen zu hohen CO₂-Gehalt in der Raumluft drohen, vorgestellt. Der CO₂-Gehalt wird in parts per million (ppm) angegeben.

2.1 CO₂-Grenzwerte für eine unbedenkliche Atemluft

Die Deutsche Gesetzliche Unfallversicherung (DGUV) veröffentlicht Technische Regeln für Arbeitsstätten (ASR), zu diesen gehört auch die ASR A3.6 „Lüftung“ [18], welche sich in Kapitel 4 mit Regelungen zur Luftqualität von Arbeitsplätzen auseinandersetzt.

Wie in Tabelle A.1 auf Seite 24 zu sehen, wird eine generelle CO₂-Konzentration von unter 1000 ppm als ungefährlich eingestuft und fordert keine weiteren Maßnahmen. Bei einer CO₂-Konzentration von 1000 bis 2000 ppm wird die Luftqualität als hygienisch auffällig beschrieben, dabei werden bereits Maßnahmen wie bessere Lüftungsmaßnahmen und eine Überprüfung des aktuellen Lüftungsverhalten empfohlen. Ab einer generellen CO₂-Konzentration von über 2000 ppm stuft die DGUV die Atemluft als hygienisch inakzeptabel ein, hierbei werden Gegenmaßnahmen wie eine verstärkte Lüftung bis zur Reduktion der Personenzahl im Raum empfohlen.

Diese Messungen sollen laut der ASR A3.6 [18] in einer Höhe von 1,50 m und einem Abstand von 1 bis 2 m von Wänden erfolgen, mit einer ausreichend großen Distanz zu Personen, damit diese die Messungen nicht direkt durch ihre Atemluft beeinflussen.

Die DIN EN 16798-1 [19] „Energetische Bewertung von Gebäuden - Lüftung von Gebäuden“ [19], regelt die CO₂-Grenzwerte strikter und feingranularer als die ASR A3.6 [18]. Wie in Tabelle B.1 auf Seite 25 wird bei Werten unter 950 ppm von einer hohen Raumluftqualität gesprochen, von 950 bis 1200 ppm gilt die Raumluftqualität als „mittlere Raumluftqualität“. CO₂-Konzentrationen von 1200 bis 1750 ppm werden als „Mäßige Raumluftqualität“ und Werte über 1750 ppm als „Niedrige Raumluftqualität“ eingestuft. Aus den beiden vorgestellten Bewertungen der CO₂-Konzentration für die Raumluft kann geschlossen werden, dass die CO₂-Konzentration generell nicht über 1000 ppm steigen

sollte, damit die Raumluft eine ausreichend hohe Qualität aufweist.

2.2 Physiologische Auswirkungen eines zu hohen CO₂-Gehaltes in der Atemluft

Eine von Kajtár et al. [3] durchgeführte Studie zeigt, dass eine höhere CO₂-Konzentration zu einer schnelleren Erschöpfung beim Lösen von Aufgaben führt. Dafür sollten zehn Probanden und Probandinnen unter verschiedenen CO₂-Konzentrationen über einen Zeitraum von jeweils zwei mal 70 Minuten einen Text auf Fehler korrigieren. Dabei wurde festgestellt, dass sich die mentale Leistungsfähigkeit im Gegensatz zu einer akzeptablen CO₂-Konzentration von 600 ppm verringert. Die Messung von physiologischen Parametern der Teilnehmenden zeigte außerdem, dass das Lösen von Denkaufgaben dadurch ebenfalls anstrengender wird. Bei einem Aufenthalt von zwei bis drei Stunden in einem geschlossenen Raum führte eine CO₂-Konzentration von 3000 ppm zu einer verringerten Konzentrationsfähigkeit.

Zhang et al. [20] kommen in einer Studie mit 15 Probanden und Probandinnen zu der Erkenntnis, dass eine erhöhte CO₂-Konzentration in der Atemluft zu einer stark verringerten Leistungsfähigkeit beim Lösen von Denkaufgaben führt. Dafür wurden die Teilnehmenden verschiedenen CO₂-Konzentrationen (1500, 3500 und 5000 ppm) ausgesetzt, während sie verschiedene Aufgaben lösen mussten. Der Abfall der Leistungsfähigkeit wurde dabei erst bei der Veränderung der CO₂-Konzentration von 1500 ppm auf 3500 ppm festgestellt. Bei einem weiteren Anstieg des Wertes auf 5000 ppm wurden keine weiteren Veränderung der mentalen Leistungsfähigkeit festgestellt.

Bei einer Studie, die von Myhrvold et al. [4] mit 548 Schüler*innen durchgeführt wurde, werden die Auswirkungen auf den Körper, speziell auf das Zentralnervensystem (ZNS), die Haut und Schleimhaut untersucht. Während bei dieser Studie die Auswirkungen auf die mentale Leistungsfähigkeit nicht signifikant auffällig waren, wirkte sich eine CO₂-Konzentration von über 1500 ppm negativ auf das körperliche Wohlbefinden aus. Dabei wurden unter anderem Symptome wie Kopfschmerzen, Müdigkeit, Konzentrationsschwäche und Hustenanfälle festgestellt.

Ein ähnliches Ergebnis erzielt auch eine von Tiesler et al. [2] durchgeführte Studie. Dafür wurde der CO₂-Gehalt der Atemluft in den Klassenräumen von 16 Klassen aus drei verschiedenen Bremer Schulen und dessen Wirkung auf die Schüler*innen untersucht. Dabei wurde festgestellt, dass sich regelmäßiges Lüften, um die CO₂-Konzentration zu senken, positiv auf das Wohlbefinden der Teilnehmenden auswirkt. Der zu erreichende Zielwert durch das erhöhte Lüftungsintervall lag dabei unter 1000 bzw. 1500 ppm, die vorher erreichten Werte nach einer Doppelstunde erreichten bis zu 4000 ppm. Die Schüler*innen berichteten in Befragungen nach der Erhöhung der Lüftungsintervalle von einer verbesser-

serten Aufmerksamkeit, welche sich wahrscheinlich auf die geringere CO₂-Konzentration in der Raumluft zurückführen lässt.

3 Hardwaretechnische Umsetzung

3.1 Technische Anforderungen an die benötigte Hardware

Um einen CO₂-Monitor betreiben zu können werden ein Computer und ein CO₂-Sensor benötigt. Die Mindestanforderungen an den Computer sind 512MB Arbeitsspeicher, ein ARM Cortex-A53 Quad-Core 1GHz 64-bit Prozessor und mindestens ein USB Typ A Anschluss. Diese Werte sind die Mindestanforderungen eines minimalen Linux Betriebssystems, welches zur Erfüllung der Softwareanforderungen benötigt wird.

Der CO₂-Sensor sollte ein Messbereich von 0 ppm bis einschließlich 2000 ppm vorweisen. Mit dieser Spannweite können die in Kapitel 2 auf Seite 5 vorgestellten Werte der Luftqualität gemessen und aufgezeichnet werden.

3.2 Überblick über die verwendete Hardware

Für den Betrieb der Programme rund um den CO₂-Monitor wird folgende Hardware verwendet:

- Raspberry Pi 3 Model B+
- TFA-Dostmann CO₂-Monitor AIRCO2NTROL MINI

Der Raspberry Pi ist ein Minicomputer auf einer Platine von der Größe einer Kreditkarte, der sich vielfältig durch zusätzliche Hardware erweitern lässt. Er wurde ursprünglich zu Lern- und Demonstrationszwecken entwickelt, eignet sich aber auch für die Realisierung verschiedener privater oder professioneller Anwendungen. Er überzeugt vor allem durch seinen vergleichsweise geringen Anschaffungspreis und die sehr große Internet-Community, die bei Fragen und Problemen bei der Lösungsfindung helfen kann.

Mit dem CO₂-Monitor AIRCO2NTROL MINI kann die Luft in Räumen, in denen sich Personen aufhalten, z.B. Schulen, Büros, öffentliche Einrichtungen, kontrolliert werden. Der Monitor zeigt auf einem Display den CO₂-Gehalt der Raumluft im Wechsel mit der Temperatur an. Eine farbige Ampelanzeige neben dem Display wechselt je nach Luftqualität von Grün über Gelb zu Rot.

4 Softwaretechnische Umsetzung

4.1 Benötigte Software

Um die Programme rund um den CO₂-Monitor in Betrieb nehmen zu können, wird folgende Software benötigt:

- Linux-Distribution mit mitgelieferter Standardsoftware
- Docker
- docker-compose

Ist die benötigte Software installiert und eingerichtet, kann mit der Implementierung der Auslese- und Verarbeitungssoftware begonnen werden. Alle in den folgenden Kapiteln genannten Softwarekomponenten sind auf GitHub unter <https://github.com/FMinister/> zu finden.

4.2 Zusammenspiel der Softwarekomponenten

Die Daten des CO₂-Sensors werden persistent in einer Datenbank gespeichert, damit die Werte abrufbar sind und auch über längere Zeiträume ausgewertet werden können. Dafür werden die CO₂- und Temperaturwerte mittels einer Software-Komponente zum Auslesen der Daten an ein Backend gesendet und gespeichert. Diese Daten können dann mittels eines Frontends angezeigt werden. Diese Verknüpfung der Software-Komponenten ist in Abb. 4.1 auf der nächsten Seite zu sehen.

4.3 Aufbau und Einrichten der Software

4.3.1 Aufbau und Einrichten des Backends

Die zentrale Stelle, an der Daten eingehen, gespeichert und abgerufen werden können, wird mittels der *CO2MonitorAPI* realisiert. Diese ist in Python geschrieben und verwendet FastAPI als Grundlage für das Bereitstellen einer API. Um das Bereitstellen der Anwendung und deren Isolierung vom Betriebssystem zu erleichtern, wird Docker als Containervirtualisierungs-Software eingesetzt.

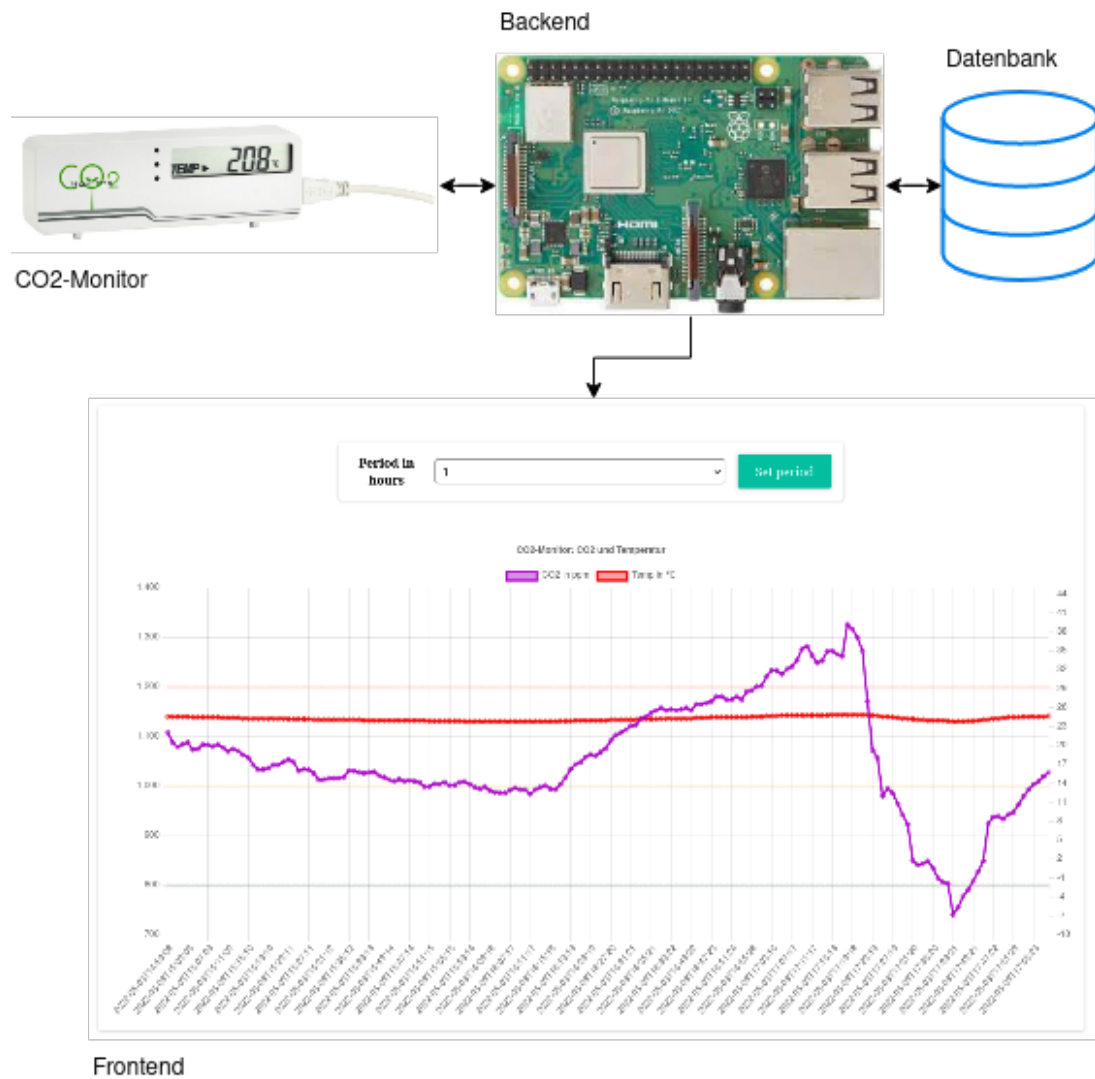


Abbildung 4.1: Verbundplan der Komponenten

Nachdem die Anwendung von GitHub bezogen wurde, müssen noch Konfigurationswerte angepasst werden. Beispielsweise muss der Port festgelegt werden, auf dem die Application Programming Interface (API) erreichbar sein soll. In der `docker-compose.yml`-Datei kann hierfür besagter Port angegeben werden, welcher dem Frontend und dem *CO2Reader* Zugang zur Backend-Logik erlaubt, um Daten abzuspeichern und abzurufen.

Durch das Öffnen eines Terminal-Fensters im Ordner der API und der Eingabe des Befehls

```
docker-compose -f docker-compose.yml up -d
```

wird die Anwendung gestartet. Der Docker-Container läuft ab jetzt im Hintergrund und wartet auf Speicher- oder Abrufbefehle. Ob die Applikation richtig funktioniert kann im Browser mittels

```
IpAdresseDesPis:angegebenerPort/api/test
```

getestet werden.

4.3.2 Aufbau und Einrichten der Lese-Software

Die Daten des CO₂-Sensors werden mittels der USB-Schnittstelle ausgelesen. Diese Lesesoftware ist in Python geschrieben und nutzt die CO2Meter-Bibliothek von Vladimir Filimonov. [1]

Dazu muss die `docker-compose.yml`-Datei des *CO2Readers* angepasst werden. Um den richtigen USB-Port in die Datei schreiben zu können, können alle angeschlossenen USB-Geräte mit dem in Listing 4.1 zu sehenden Bash-Skript angezeigt und die hidraw-Id des Geräts *Holtek Semiconductor, Inc. USB-zyTemp* ausgelesen werden. [12]

Listing 4.1: Bash-Skript zum Erkennen der hidraw-Id

```
1  #!/bin/bash
2
3  FILES=/dev/hidraw*
4  for f in $FILES
5  do
6      FILE=${f##*/}
7      DEVICE="$(cat /sys/class/hidraw/${FILE}/device/uevent | grep
8          HID_NAME | cut -d '=' -f2)"
9      printf "%s \t %s\n" $FILE "$DEVICE"
```

Ist der USB-Port bestimmt, kann der Teil vor dem Doppelpunkt der `devices`, in diesem Fall `„/dev/hidraw0“`, mit dem ausgelesenen Port ersetzt werden, zu sehen in Listing 4.2 auf der nächsten Seite.

Listing 4.2: Anpassen des USB-Ports in der docker-compose.yml

```
1  ...
2  devices:
3    - /dev/hidraw0:/dev/hidraw16
4  ...
```

Soll der *CO2Reader* auf einem anderen Gerät als die API ausgeführt werden, muss die `co2Reader.ini`-Datei angepasst werden. Diese ist in Ordner `app/` zu finden. Hier muss die IP-Adresse der API anstelle der bestehenden IP-Adresse angegeben werden. Auch kann hier der Standort, an dem sich der Sensor befindet, eingetragen werden.

Nach dem Abspeichern der Datei kann ein Terminal-Fenster im Ordner des *CO2Readers* geöffnet werden und mittels

```
docker-compose -f docker-compose.yml up -d
```

die Anwendung gestartet werden. Der Docker-Container läuft ab jetzt im Hintergrund, liest die Daten des Sensors aus und schickt diese an die angegebene IP-Adresse der API.

4.3.3 Aufbau und Einrichten des Frontends

Um die Daten ansehnlich darstellen zu können, kann ein Frontend eingebunden werden. Das hier verwendete Frontend ist mit React [9] und ChartsJs [6] umgesetzt worden. Ein Beispiel der Datenvisualisierung ist in Abb. 4.2 auf der nächsten Seite zu sehen.

Nachdem das Frontend bezogen wurde, muss die `docker-compose.yml`-Datei angepasst werden. In dieser Datei muss die IP-Adresse der `REACT_APP_API_URL` mit der IP des Netzwerkgerätes, auf dem die API läuft, ausgetauscht werden, zu sehen in Listing 4.3.

Der ausgehende Port, von dem das Frontend am Ende erreichbar ist, kann ebenfalls angepasst werden. Hierzu muss lediglich der `ports`-Abschnitt verändert werden, ebenfalls zu sehen in Listing 4.3. Hier muss der Port vor dem Doppelpunkt auf den gewünschten Port gesetzt werden.

Listing 4.3: Anpassen der API-IP und des Ports in der docker-compose.yml

```
1  ...
2  environment:
3    REACT_APP_API_URL: http://192.168.178.33:8008/api/
4  ports:
5    - 3000:3000
6  ...
```

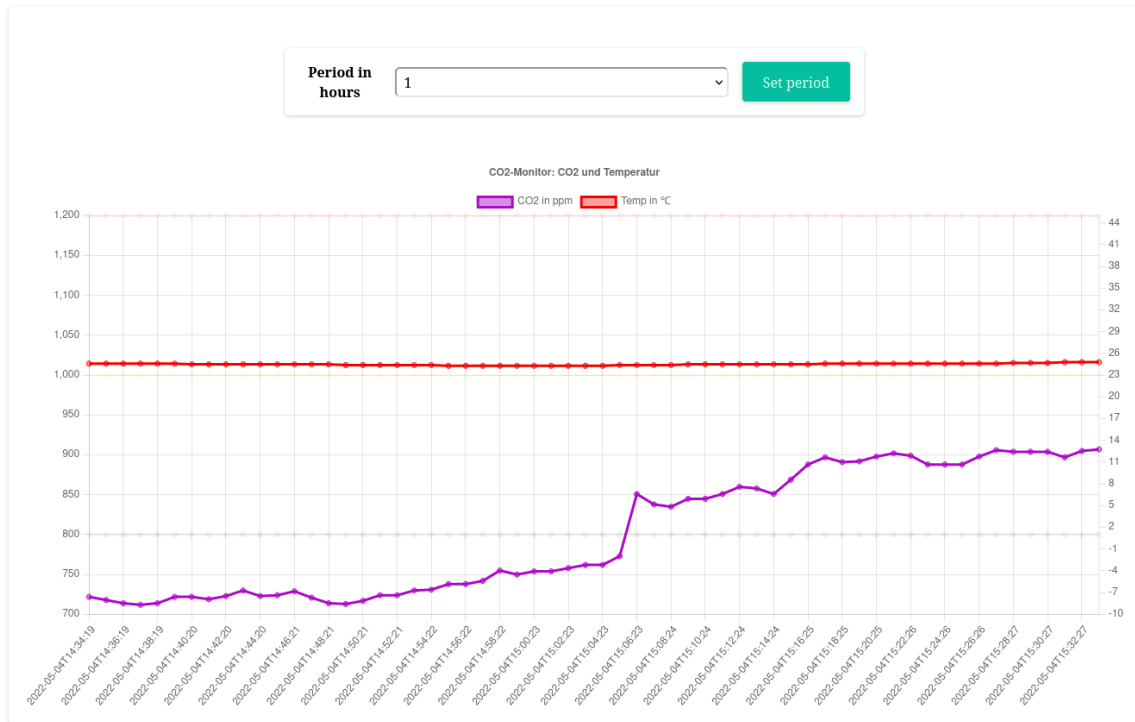


Abbildung 4.2: Datenvisualisierung mittels Frontend

Mit dem Öffnen eines Terminal-Fensters im Ordner des Frontends und mittels

```
docker-compose -f docker-compose.yml up -d
```

wird Anwendung gestartet. Der Docker-Container läuft ab jetzt im Hintergrund und kann mittels der IP-Adresse des ausführenden Gerätes sowie dem in der `docker-compose.yml`-Datei angegebenen Port aufgerufen werden.

5 Zusammenfassung

Eine zu hohe CO_2 -Konzentration in der Raumluft mindert die Konzentrationsfähigkeit, die Produktivität und das körperliche Wohlbefinden der in diesen Räumen arbeitenden Menschen. Die von der DGUV ASR A3.6 [18] und der DIN EN 16798-1 [19] festgelegten Grenzwerte für den CO_2 -Gehalt der Raumluft decken sich in etwa mit den durch verschiedene Studien belegten Grenzwerten und bilden damit eine aussagekräftige Grundlage für die Einschätzung der Raumluftqualität. Die Einrichtung eines CO_2 -Monitors mittels Raspberry-Pi und eines CO_2 -Sensors ist eine relativ kostengünstige und mit etwas Vorwissen auch einfach zu implementierende Möglichkeit, den CO_2 -Gehalt der Raumluft zu bestimmen. Diese Messungen sollten vor allem in Büros und Klassenräumen stattfinden, da dort generell viele Menschen auf engem Raum arbeiten und ihre Leistung möglichst nicht durch ein schlechtes Raumklima vermindert werden sollte.

Abbildungsverzeichnis

4.1	Verbundplan der Komponenten	10
4.2	Datenvisualisierung mittels Frontend	13

Listings

4.1	Bash-Script zum Erkennen der hidraw-Id	11
4.2	Anpassen des USB-Ports in der docker-compose.yml	12
4.3	Anpassen der API-IP und des Ports in der docker-compose.yml	12

Glossar

API ist der englische Fachbegriff für eine Programmierschnittstelle (Application Programming Interface) oder auch Anwendungsschnittstelle. Es handelt sich dabei um einen Programmteil, der genutzt wird, um anderen Anwendungen Funktionen einer Anwendung zur Verfügung zu stellen. In der Regel gibt es dafür eine detaillierte Dokumentation der einzelnen Funktionen mit den jeweiligen Parametern.[15]. 9

Backend ist der Teil eines IT-Systems, der sich mit der Datenverarbeitung im Hintergrund beschäftigt – der Data Layer. Der Begriff dient der Unterteilung bei komplexeren Softwarestrukturen. [7]. 9, 11

Containervirtualisierung ist ein Konzept, bei dem auf einem Betriebssystem mehrere Anwendungen voneinander isoliert betrieben werden können. Das ermöglicht das einfache Bereitstellen von funktionierenden und fertig konfigurierten Anwendungen, besonders im Hinblick auf den Datenaustausch der einzelnen Anwendungen untereinander. [14]. 9

Datenbank ist ein System zur elektronischen Datenverwaltung. Eine Datenbank, auch Datenbanksystem genannt, ist allgemein dafür verantwortlich, große Datenmengen effizient, widerspruchsfrei und dauerhaft zu speichern, sowie diese bereitzustellen. [16]. 9

Docker ist eine freie Softwareplattform der Docker, Inc. Das Ziel von Docker ist es, das Erstellen, Testen und Bereitstellen von Software im wesentlichen zu beschleunigen. Mit Docker können Softwarepakete, sogenannte Container, erstellt werden. Diese gewährleisten eine Trennung und Verwaltung der lokalen Computerressourcen. Die am 13.03.2013 erstmals veröffentlichte Software ist unter anderem für die Betriebssysteme Linux, Microsoft Windows und macOS verfügbar. [13] [11]. 9, 11–13

docker-compose ist ein Tool, das die Definition und gemeinsame Nutzung von Multicontainer-Anwendungen erleichtern soll. Dienste werden mittels einer YAML-Datei erstellt und mit einem einzigen Befehl können alle Container gestartet oder abgebaut werden. [11]. 9

FastAPI ist ein Web-Framework zum Entwickeln von RESTful-APIs in Python. FastAPI basiert auf Pydantic und gibt Hinweise zum Validieren, Serialisieren und Deserialisieren von Daten und zum automatischen Generieren von OpenAPI-Dokumenten. [17]. 9

Frontend bezeichnet bei einem IT-System die Presentation Layer, also den Teil eines IT-Systems, der näher am Anwender ist. [8]. 9, 11–13, 15

GitHub ist ein netzbasierter Dienst zur Versionsverwaltung für Software-Entwicklungsprojekte. Namensgebend war das Versionsverwaltungssystem Git. [10]. 9, 11

hidraw ist eine Hardware-Treiber-Schnittstelle. Der hidraw-Treiber bietet eine Rohschnittstelle zu USB- und Bluetooth-Human Interface Devices (HIDs). 11, 16

Python ist eine plattformunabhängige, multiparadigmatische, stark und dynamisch typisierte Programmiersprache. Die Programmiersprache zeichnet sich besonders durch ihre Trivialität und vielseitige Einsetzbarkeit aus. Python ist eine der meist verbreiteten Programmiersprachen weltweit. Die Weiterentwicklung der Programmiersprache wird durch die Python Software Foundation und den ursprünglichen Entwickler Guido van Rossum durchgeführt. [5]. 9, 11

Literatur

- [1] Vladimir Filimonov. *CO2meter*. Verfügbar unter: <https://github.com/vfilimonov/co2meter>. abgerufen am 05.03.2022.
- [2] Tiesler G, Schönwälder H & F Ströver. *Gesundheitsfördernde Einflüsse auf das Leistungsvermögen im schulischen Unterricht*. 2008. ISBN: 978-3.86509-916-7. abgerufen am 03.05.2022.
- [3] Lázló Kajtár & Levente Herczeg. „Influence of carbon-dioxide concentration on human well-being and intensity of mental work“. en. In: *QJ Hung. Meteorol. Serv* 116.2 (2012), S. 145–169. Verfügbar unter: https://www.researchgate.net/profile/Laszlo-Kajtar/publication/231558578_Influence_of_carbon-dioxide_concentration_on_human_well-being_and_intensity_of_mental_work/links/0912f506c5e44909eb000000/Influence-of-carbon-dioxide-concentration-on-human-well-being-and-intensity-of-mental-work.pdf. abgerufen am 03.05.2022.
- [4] AN Myhrvold, E Olsen & O Lauridsen. „Indoor environment in schools—pupils health and performance in regard to CO2 concentrations“. In: *Indoor Air* 96.4 (1996), S. 369–371. abgerufen am 03.05.2022.
- [5] N. N. *3.10.4 Documentation*. Verfügbar unter: <https://docs.python.org/3/>. abgerufen am 27.04.2022.
- [6] N. N. *Chart.js | Open source HTML5 Charts for your website*. Verfügbar unter: <https://www.chartjs.org/>. abgerufen am 27.04.2022.
- [7] N. N. *Definition Backend Erklärung Backend*. Verfügbar unter: <http://www.softselect.de/business-software-glossar/backend>. abgerufen am 05.03.2022.
- [8] N. N. *Definition Frontend Erklärung Frontend*. Verfügbar unter: <http://www.softselect.de/business-software-glossar/frontend>. abgerufen am 05.03.2022.
- [9] N. N. *Getting Started – React*. en. Facebook Inc. Verfügbar unter: <https://reactjs.org/docs/getting-started.html>. abgerufen am 27.04.2022.

- [10] N. N. *GitHub*. de. Page Version ID: 221919304. 04/2022. Verfügbar unter: <https://de.wikipedia.org/w/index.php?title=GitHub&oldid=221919304>. abgerufen am 05.05.2022.
- [11] N. N. *Install Docker Engine*. en. 04/2022. Verfügbar unter: <https://docs.docker.com/engine/install/>. abgerufen am 27.04.2022.
- [12] N. N. *Matching /dev/hidraw* devices with physical devices*. en. 10/2019. Verfügbar unter: <https://arvchristos.github.io/post/matching-dev-hidraw-devices-with-physical-devices/>. abgerufen am 04.05.2022.
- [13] N. N. *Orientation and setup*. en. Verfügbar unter: <https://docs.docker.com/get-started/>. abgerufen am 27.04.2022.
- [14] N. N. *Warum ist die Containervirtualisierung bei Anwendungen effizienter?* de-DE. 04/2021. Verfügbar unter: <https://www.hosteurope.de/blog/containervirtualisierung-von-anwendungen-als-effektive-ergaenzung-zu-virtuellen-maschinen/>. abgerufen am 05.05.2022.
- [15] N. N. *Was ist eine API? - Definition im IT-Lexikon*. de-DE. Verfügbar unter: <https://it-service.network/it-lexikon/api>. abgerufen am 05.05.2022.
- [16] N. N. *Was ist eine Datenbank?* de-DE. Verfügbar unter: <https://www.oracle.com/de/database/what-is-database/>. abgerufen am 03.05.2022.
- [17] Sebastián Ramírez. *FastAPI*. Verfügbar unter: <https://fastapi.tiangolo.com/>. abgerufen am 27.04.2022.
- [18] DGUV e. V. *ASR A3.6*. de. Text. 2018. Verfügbar unter: https://www.baua.de/DE/Angebote/Rechtstexte-und-Technische-Regeln/Regelwerk/ASR/pdf/ASR-A3-6.pdf?__blob=publicationFile&v=1. abgerufen am 03.05.2022.
- [19] DIN e. V. *DIN EN 16798-1:2021-04*. de. Text. 2017. abgerufen am 03.05.2022.
- [20] Jie Zhang u. a. „The effects of elevated carbon dioxide concentration and mental workload on task performance in an enclosed environmental chamber“. In: *Building and Environment* 178 (2020), S. 106938. ISSN: 0360-1323. DOI: <https://doi.org/10.1016/j.buildenv.2020.106938>. Verfügbar unter: <https://www.sciencedirect.com/science/article/pii/S0360132320302973>.

Anhang

Inhalt des Anhangs

Tabellen im Anhang	23
A Bewertung der CO₂ Konzentration in der Raumluft nach DGUV ASR A3.6	24
B Bewertung der CO₂ Konzentration in der Raumluft nach DIN EN 16798-1	25

Tabellen im Anhang

A.1	Bewertung der CO ₂ Konzentration in der Raumluft nach DGUV ASR A3.6	
[18]	24
B.1	Bewertung der CO ₂ Konzentration in der Raumluft nach DIN EN 16798-1	
[19]	25

A Bewertung der CO₂ Konzentration in der Raumluft nach DGUV ASR A3.6

Tabelle A.1: Bewertung der CO₂ Konzentration in der Raumluft nach DGUV ASR A3.6 [18]

CO ₂ -Konzentration in ppm	Maßnahmen
<1000	<ul style="list-style-type: none">• Keine weiteren Maßnahmen (sofern durch die Raumnutzung kein Konzentrationsanstieg über 1000 ppm zu erwarten ist)
1000-2000	<ul style="list-style-type: none">• Lüftungsverhalten überprüfen und verbessern• Lüftungsplan aufstellen (z. B. Verantwortlichkeiten festlegen)• Lüftungsmaßnahme (z. B. Außenluftvolumenstrom oder Luftwechsel erhöhen)
>2000	<ul style="list-style-type: none">• weitergehende Maßnahmen erforderlich (z. B. verstärkte Lüftung, Reduzierung der Personenzahl im Raum)

B Bewertung der CO₂ Konzentration in der Raumluft nach DIN EN 16798-1

Tabelle B.1: Bewertung der CO₂ Konzentration in der Raumluft nach DIN EN 16798-1
[19]

CO ₂ -Konzentration in ppm	Beschreibung
<950	Hohe Raumluftqualität
950-1200	Mittlere Raumluftqualität
1200-1750	Mäßige Raumluftqualität
>1750	Niedrige Raumluftqualität