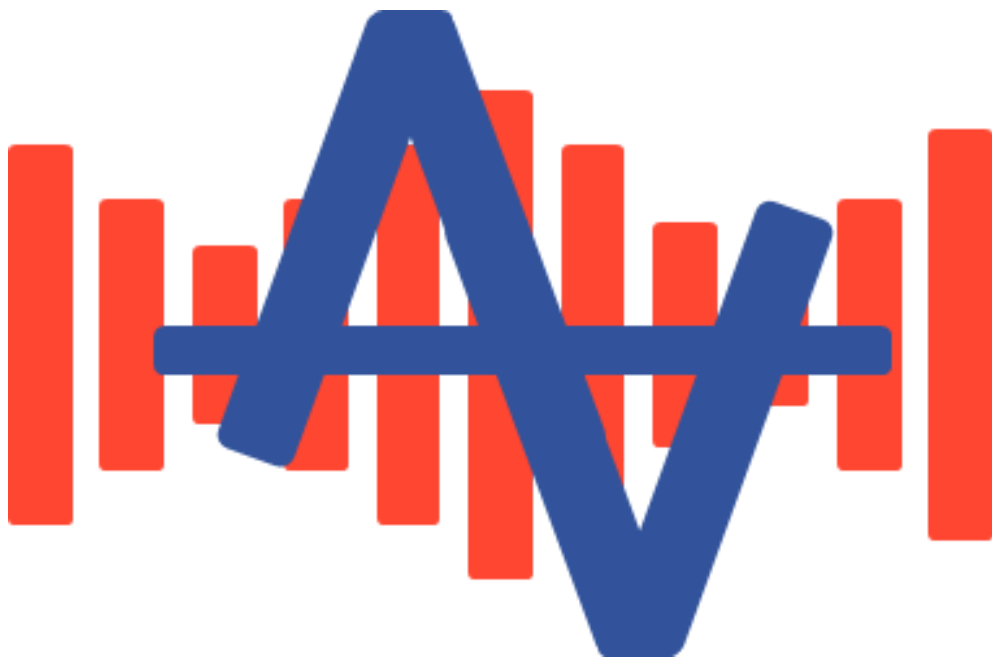


# Anwenderdokumentation

Version 1.0

**Team Audio Architects**



hochschule mannheim

abbvie





# Inhaltsverzeichnis

<b>1. Verteiler</b>	<b>3</b>
<b>2. Änderungsübersicht</b>	<b>4</b>
<b>3. Systemkontext</b>	<b>5</b>
<b>4. Ressourcen</b>	<b>7</b>
<b>5. Installation</b>	<b>8</b>
5.1 Bereitstellung der Software	8
5.2 Installation & Starten der Anwendung	8
5.2.1 Erstellung der Datenbank	8
5.2.2 Installation der Messgeräte	10
1.OG	10
2.OG	11
1.OG ohne Bereich	14
2.OG ohne Bereich	14
5.2.3 Installation der Webanwendung	15
5.3 Packages	17
5.3.1 Ausleseskript	17
5.3.2 Backend der Website	17
<b>6. Produkthandbuch</b>	<b>18</b>
6.1.1 Übersicht	18
6.1.2 Heute	19
6.1.3 Kalenderwoche	20
6.1.4 Monat	20
6.1.5 Download	21
6.1.6 Warnsystem	22
<b>7. Qualitätssicherung</b>	<b>23</b>



# 1. Verteiler

Rolle	Name	E-Mail	Institution
Stakeholder	Barbara Leimbach	Barbara.Leimbach@abbvie.com	AbbVie Deutschland GmbH & Co. KG
	Alfred Stefan	Alfred.Stefan@abbvie.com	AbbVie Deutschland GmbH & Co. KG
Betreuer	Prof. Dr. Stefan Kraus	S.Kraus@hs-mannheim.de	HS Mannheim
	Prof. Dr. Markus Gumbel	M.Gumbel@hs-mannheim.de	HS Mannheim
Team Audio Architects		Audio.Architects@outlook.de	
	Justin Bühler	2023506@stud.hs-mannheim.de	HS Mannheim
	Damian Johnson	2025027@stud.hs-mannheim.de	HS Mannheim
	Luca Koopmann	2025212@stud.hs-mannheim.de	HS Mannheim
	Nico Ohler	2024457@stud.hs-mannheim.de	HS Mannheim
Tutorin	Izem Putak	1922720@stud.hs-mannheim.de	HS Mannheim



## 2. Änderungsübersicht

Version	Datum	Bemerkung
1.0	27.06.2023	Erstfassung



### **3. Systemkontext**

Mitarbeiter der AbbVie Deutschland GmbH & Co. KG beziehen in der Woche des 01.05.2023 ein neues Bürogebäude. Während sie zuvor in kleineren Gruppen oder einzeln in abgegrenzten Büroräumen arbeiteten, wurden die Arbeitsräume des neuen Gebäudes als Open-Space-Büros konzipiert.

Open-Space-Büros tragen nicht nur zu effektiverer und flexiblerer Raumnutzung, den aktuellen Bedürfnissen eines Mitarbeiters angepassten Arbeitsplatzwahl bei, sondern können auch den Informationsfluss innerhalb des Büros verbessern. Aufgrund der räumlichen Unbegrenztheit wird ein besserer und auch spontaner Austausch zwischen Mitarbeitern ermöglicht, was generell zu einer verbesserten Kommunikation führen kann.

Um jedoch all die erwähnten Vorteile effizient ausschöpfen zu können, sollten die Mitarbeiter vor allem für das Thema Lautstärke sensibilisiert werden. Innerhalb von Open-Space-Büros kann immer wieder eine hohe Schallbelastung entstehen, welche konzentrierte Arbeit sowohl stören als auch unmöglich machen kann. Um dies zu verhindern, sollen das von uns zu erstellende Konzept und die dazugehörige Software die Schallbelastung im Raum messen und bei Überschreitung die Mitarbeiter in den entsprechenden Bereichen darauf hinweisen. Wir möchten mit unserem Endprodukt ein gutes, auf AbbVie zugeschnittenes Gesamtprodukt erstellen, welches unterstützend dazu beiträgt, einen produktiven Arbeitsalltag zu gewährleisten.

Die Mitarbeiter von AbbVie sollen in der Lage sein, schnell und unkompliziert die Schallbelastung innerhalb ihres Arbeitsbereiches



einsehen zu können und benachrichtigt werden, falls diese bestimmte Grenzwerte überschreitet. Zudem soll der Arbeitgeber die Möglichkeit haben, die gemessenen Werte über den Zeitraum von mindestens einem halben Jahr abrufen und Überschreitungen der Lautstärke untersuchen und analysieren zu können.



## 4. Ressourcen

Die für die Projektrealisierung benötigten Ressourcen befinden sich in der Materialansicht eines GitHub-Repositories und sind für alle Entwickler zugänglich. So kann jedes Teammitglied seine Dateien den anderen Mitgliedern zur Verfügung stellen und so die Versionskontrolle sichergestellt werden.

Als personelle Ressourcen sind 4 Softwareentwickler zu nennen. Das Team wird von zwei Dozenten und einer Tutorin der Hochschule Mannheim betreut: Prof. Dr. Stefan Kraus und Prof. Dr. Markus Gumbel und Izem Putak.

Für das Backend werden Python und das Python Framework Flask verwendet. Dieses ist angebunden an eine MySQL-Datenbank. Bei der Kommunikation fungiert MySQL Connector/Python als Schnittstelle zwischen Datenbank und Backend.

Für das Frontend werden HTML 5, CSS Level 3, JavaScript 2022 verwendet sowie Chart.js zur Erstellung der Graphen.

Für Mockup-, Wireframe- und Persona-Erstellung wird Figma genutzt.



## 5. Installation

### 5.1 Bereitstellung der Software

Der komplette Software-Code wird in Form des Git-Repository (<https://github.com/buehlerjustin/MEP-A-A>) bereitgestellt. Das Ausleseskript der Raspberry Pi's sowie die Dokumente (Product Backlog, Produkthandbuch & Installationsanleitung, Testkonzept) wurden jeweils in einem separaten Ordner hinterlegt. Es wird empfohlen, das Git-Repository zur weiteren Bearbeitung zu klonen. Die Verteilung ist wie folgt:

- Code der Website, Branch: master
- Ausleseskript\*, Branch: master, Ordner: Soundmeter\_skripts
- Dokumente, Branch: master, Ordner: Dokumentation

\*Das Ausleseskript und dessen Installationsanweisungen sind spezifisch auf das Messgerät SNDWAY SW-525B zugeschnitten, sollten sie andere Geräte zur Lautstärkemessung verwenden, muss das Ausleseskript unter Umständen angepasst oder ganzheitlich neu geschrieben werden.

### 5.2 Installation & Starten der Anwendung

#### 5.2.1 Erstellung der Datenbank

Richten Sie einen MySQL-Datenbankserver ein. Unter Umständen müssen Sie diesen bei der Einrichtung an Ihre Zeitzone anpassen.

Erstellen sie zunächst auf Ihrem MySQL-Datenbankserver mit folgender Query:



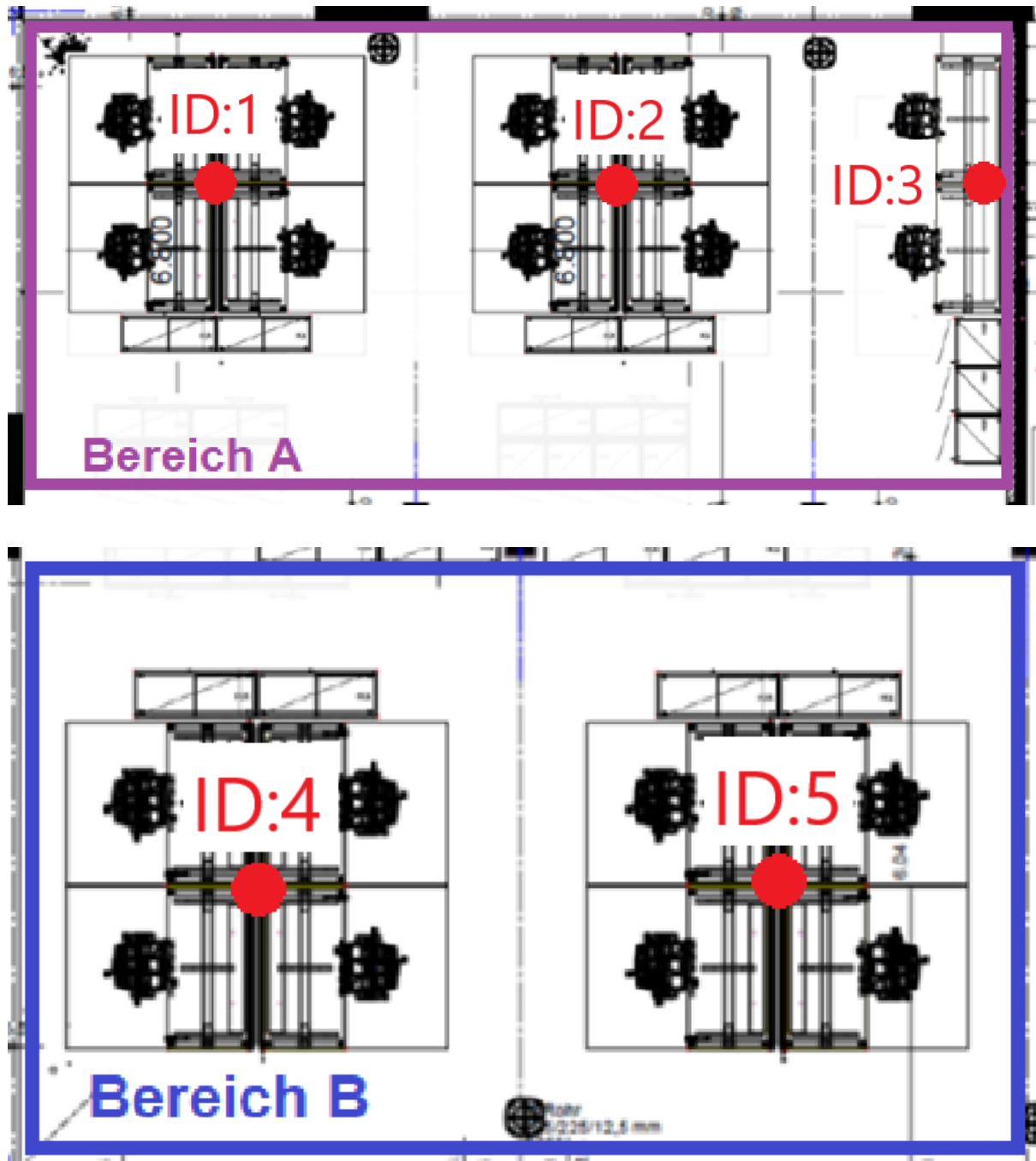


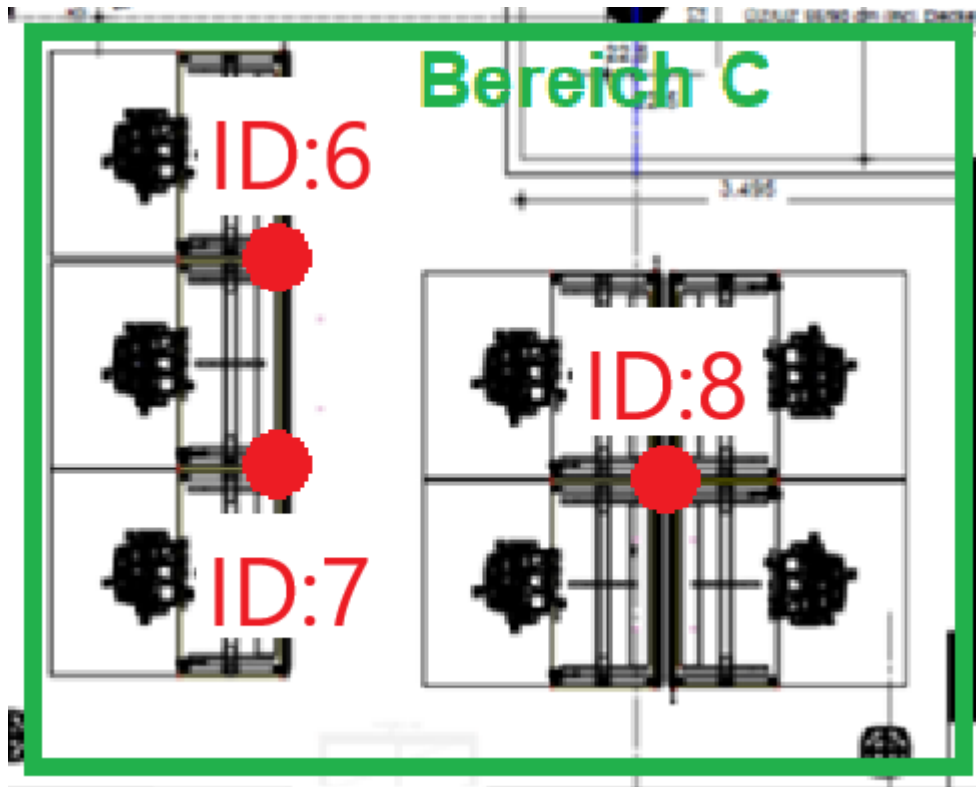
Anforderungen v1.3 - Team Audio Architects

```
CREATE DATABASE calmvie;
USE calmvie;
CREATE TABLE port (
  id INT NOT NULL PRIMARY KEY,
  designation VARCHAR(50),
  mac_address VARCHAR(50)
);
CREATE TABLE soundmeter (
  id INT NOT NULL PRIMARY KEY,
  port_id INT NOT NULL,
  area CHAR(50) NOT NULL,
  soundlimit INT NOT NULL,
  FOREIGN KEY (port_id) REFERENCES port(id)
);
CREATE TABLE measurement (
  time DATETIME NOT NULL,
  soundmeter_id INT NOT NULL,
  value INT,
  PRIMARY KEY (time, soundmeter_id),
  FOREIGN KEY (soundmeter_id) REFERENCES soundmeter(id)
);
CREATE TABLE notification (
  time DATETIME NOT NULL,
  area CHAR(50) NOT NULL,
  PRIMARY KEY (time, area)
);
CREATE TABLE role (
  id INT NOT NULL PRIMARY KEY,
  name VARCHAR(30)
);
CREATE TABLE user (
  username VARCHAR(30) NOT NULL PRIMARY KEY,
  role_id INT NOT NULL,
  password VARCHAR(30) NOT NULL,
  FOREIGN KEY (role_id) REFERENCES role(id)
);
```

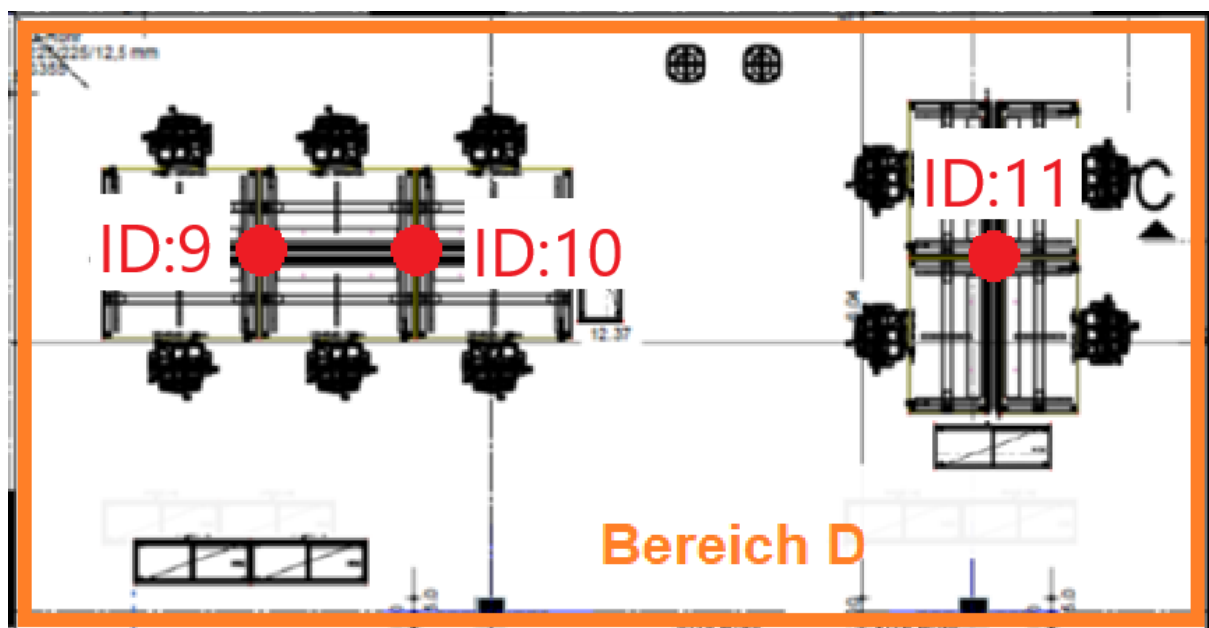
## 5.2.2 Installation der Messgeräte

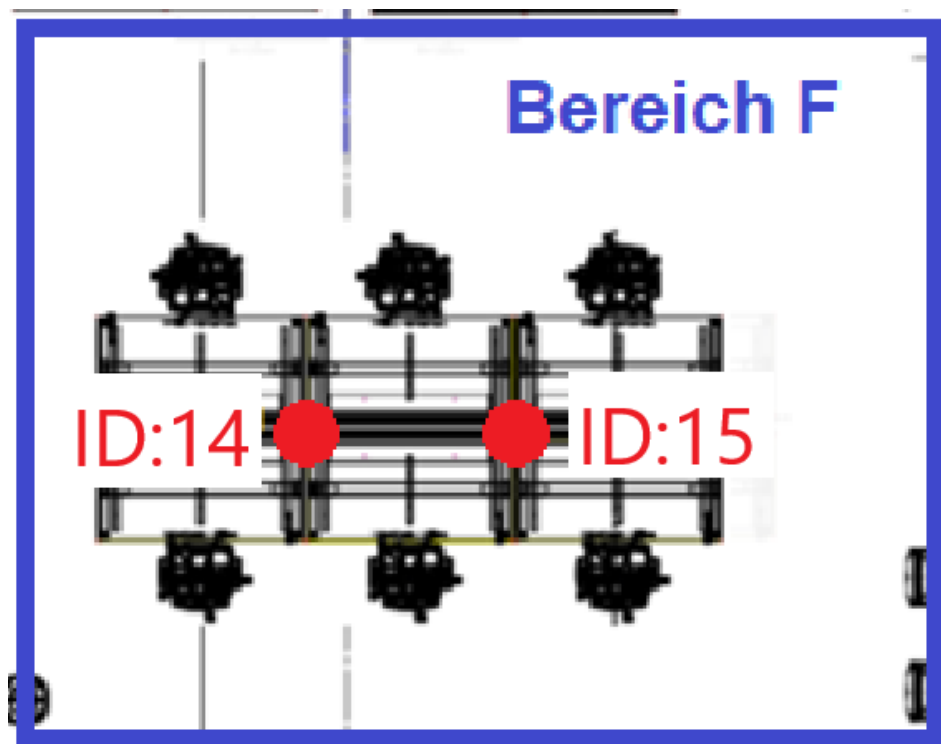
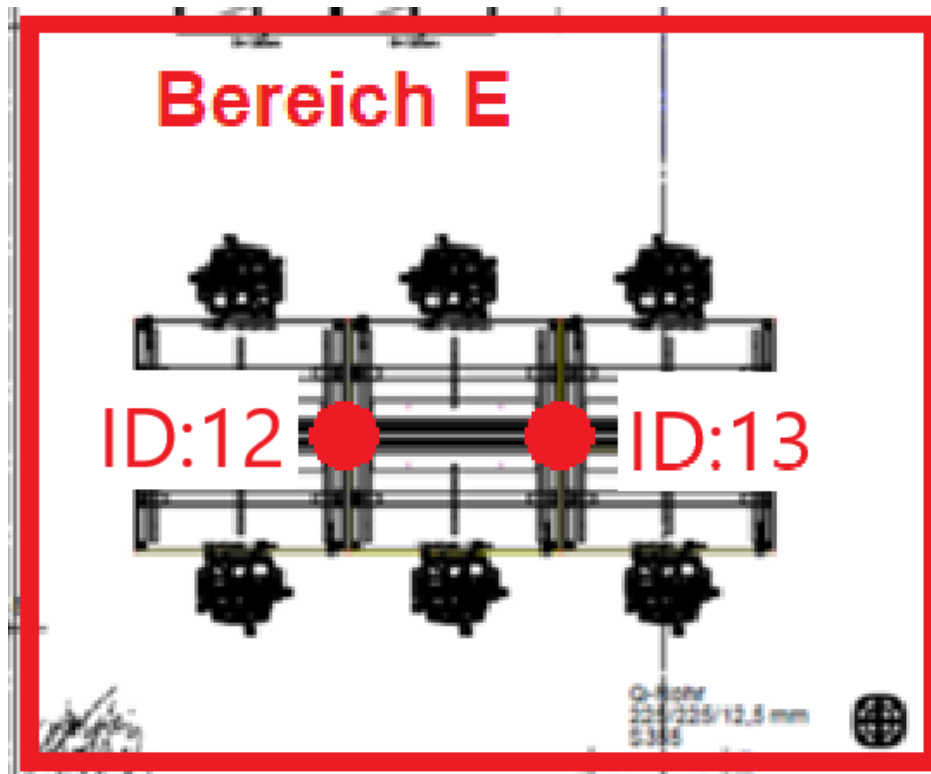
1.OG

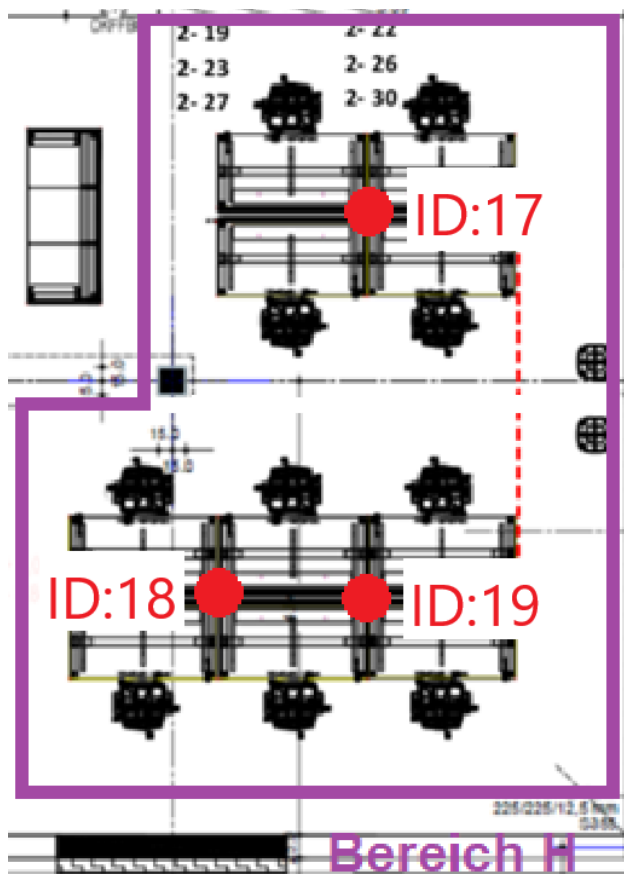
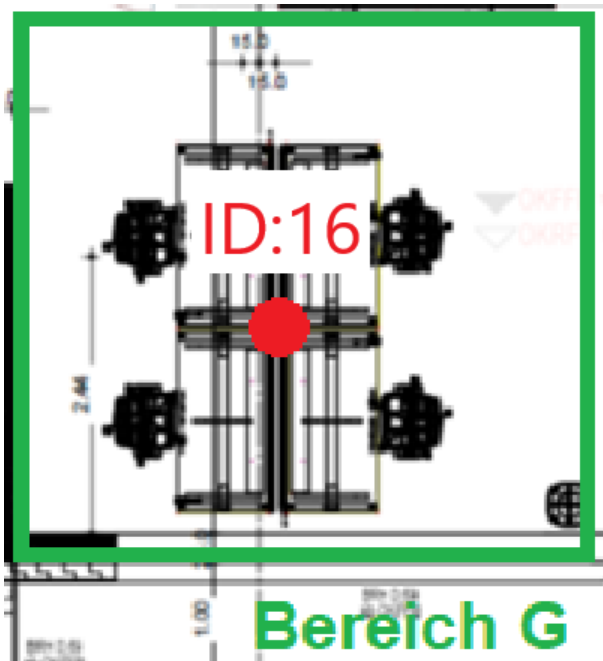




2.OG

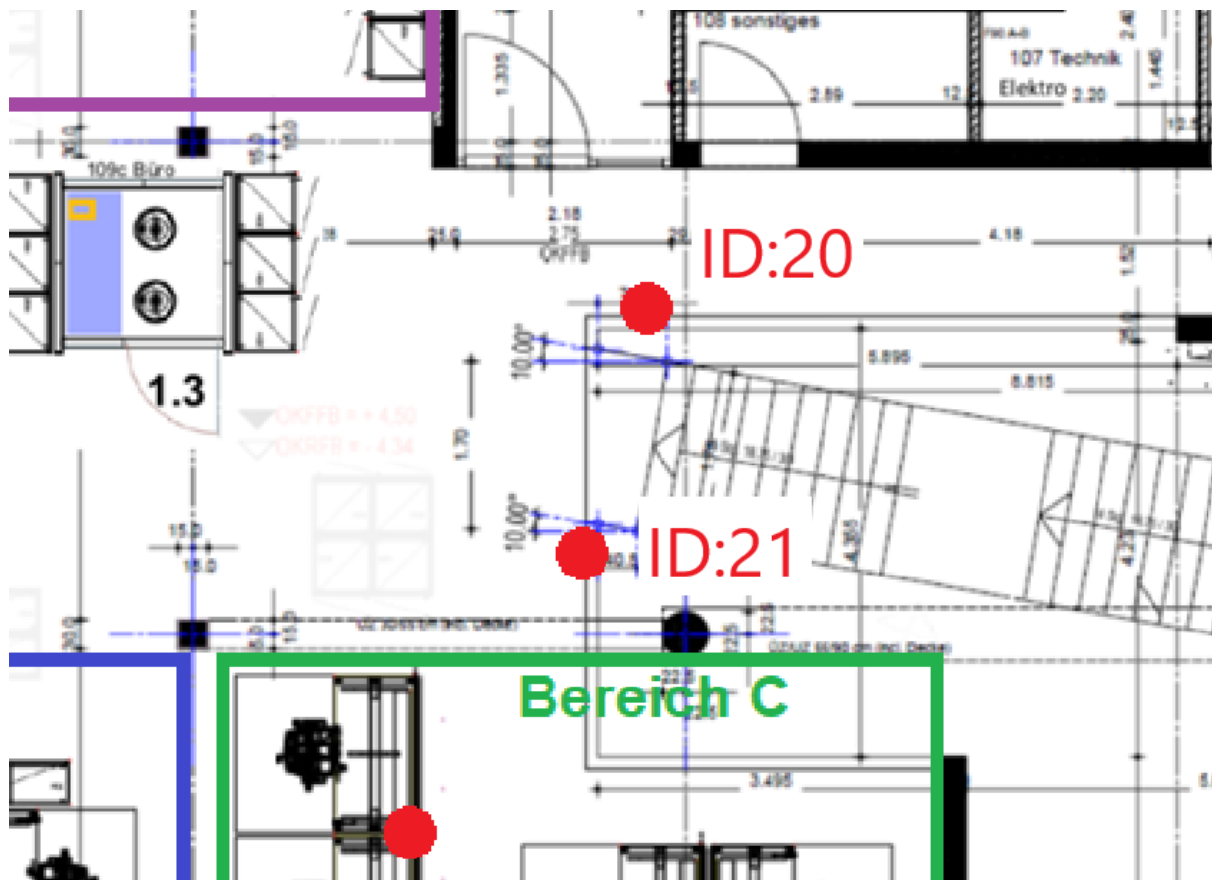




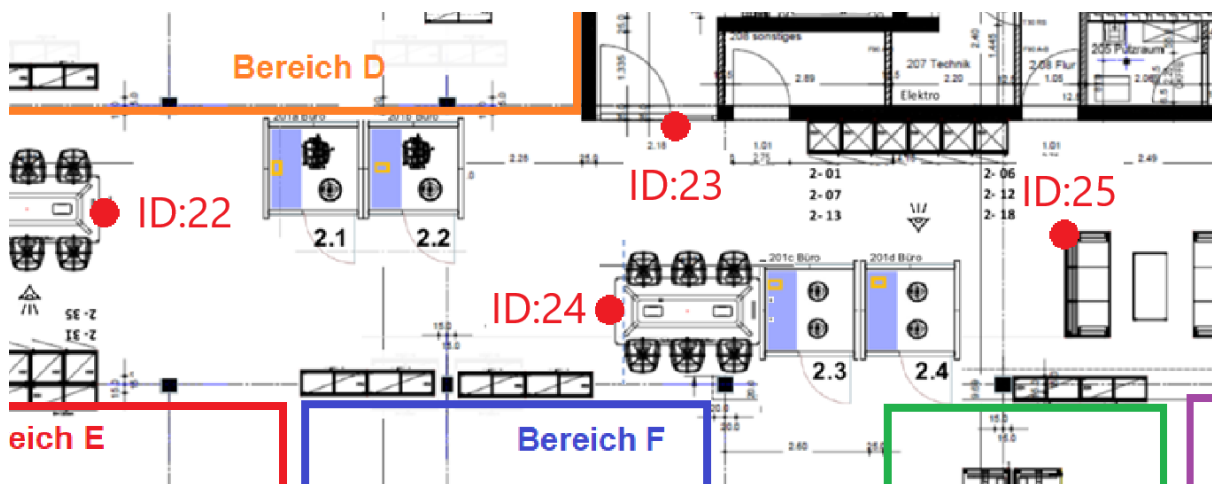




## 1.OG ohne Bereich



## 2.OG ohne Bereich





Installieren Sie Python auf jedem Raspberry Pi, für das Projekt wurde mit Python Version 3.11.3 gearbeitet. Richten Sie auf jedem Raspberry Pi das Ausleseskript ein, indem Sie dieses herunterladen, abspeichern und mit Ihrer präferierten IDE öffnen. Erstellen Sie im Ordner in welchem Sie das Ausleseskript gespeichert abgespeichert haben ein virtual environment durch:

```
python3 -m venv /path/to/new/virtual/environment
```

Führen Sie nun innerhalb des Ordners folgenden Befehl zur Installation der benötigten Packages aus.

```
pip install mysql-connector-python pyusb
```

Setzen Sie nun im Skript unter 'config' die Daten, sodass eine Verbindung mit Ihrem MySQL-Datenbankserver hergestellt werden kann. Zudem setzen Sie die Variable 'idsoundmeter' entsprechend der ID an welcher Stelle Sie den Raspberry Pi im Gebäude installieren möchten (siehe Abbildungen oben).

Befestigen Sie an den Stellen, welche in den obigen Abbildungen durch einen roten Punkt gekennzeichnet sind, die Messgeräte. Verbinden Sie jedes Messgerät per Kabelanschluss mit dem Raspberry Pi, dessen 'idsoundmeter' der Stelle entspricht.

Verbinden Sie nun die Raspberry Pi mit dem Netzwerk des Datenbankservers. Dies kann falls verfügbar über das WLAN-Modul des Raspberry Pi's erfolgen oder per USB-Verbindung (empfohlen). Starten Sie nun auf jedem Raspberry Pi das Ausleseskript.

Nachdem Sie alle Messgeräte installiert haben fügen Sie diese per MySQL-Query der Datenbank hinzu:



```
INSERT INTO port (id, designation, mac_address) VALUES (1, 'Rasp_Pi', 'Mac-Address');
INSERT INTO soundmeter (id, port_id, area, soundlimit)
VALUES (1, 1, 'A', 50), (2, 1, 'A', 50), (3, 1, 'A', 50), (4, 1, 'B', 50), (5, 1, 'B', 50), (6, 1, 'C', 55),
(7, 1, 'C', 55), (8, 1, 'C', 55), (9, 1, 'D', 50), (10, 1, 'D', 50), (11, 1, 'D', 50), (12, 1, 'E', 50),
(13, 1, 'E', 50), (14, 1, 'F', 55), (15, 1, 'F', 55), (16, 1, 'G', 55), (17, 1, 'H', 55), (18, 1, 'H', 55),
(19, 1, 'H', 55), (20, 1, '1NA', 0), (21, 1, '1NA', 0), (22, 1, '2NA', 0), (23, 1, '2NA', 0),
(24, 1, '2NA', 0), (25, 1, '2NA', 0);
```

### 5.2.3 Installation der Webanwendung

Richten Sie den Code der Website auf einem Webserver ein, indem Sie den Code aus dem Git-Repository herunterladen, auf Ihren Webserver hochladen und dort an geeigneter Stelle speichern. Installieren Sie auf dem Webserver Python, für das Projekt wurde Python Version 3.11.3 verwendet. Öffnen Sie den Ordner, in welchem Sie das Projekt abgespeichert haben, mit Ihrer präferierten IDE. Erstellen Sie im Ordner in welchem Sie den Code gespeichert abgespeichert haben ein virtual environment durch:

```
python3 -m venv /path/to/new/virtual/environment
```

Führen Sie nun innerhalb des Ordners folgenden Befehl zur Installation der benötigten Packages aus.

```
pip install mysql-connector-python Flask openpyxl APScheduler
Flask-Testing
```

Setzen Sie die folgenden Variablen in der Datei 'app.py':

- **domainname:** setzen Sie diese Variable je nachdem wie Sie die Domain nennen auf welcher die Website auf Ihrem Webserver gehostet wird (z.B. <https://calmvie>)
- **mysql-config:** Setzen Sie nun unter 'mysql-config' die Daten, sodass eine Verbindung mit Ihrem MySQL-Datenbankserver hergestellt werden kann.





- **recipient\_email:** Setzen Sie die Variable entsprechend Ihrer gewünschten Empfänger-Email.
- **sender\_email:** Setzen Sie die Variable entsprechend der Mail von welcher Sie Warnungen versenden möchten
- **username:** Benutzername zum Login in die Sender-Email
- **password:** Passwort zum Login in die Sender-Email
- **smtp\_server:** Setzen Sie die Variable entsprechende des SMTP-Servers des Hosts Ihrer Sender-Email (bei einer Outlook-Mail z.B. smtp.office365.com)
- (optional) **subject** und **body:** Passen Sie den Inhalt der Warnungsmail Ihren Vorstellungen an.

Sie können daraufhin die Anwendung über Ihre IDE starten und die Website unter Ihrer gesetzten Domain aufrufen.



## 5.3 Packages

### 5.3.1 Ausleseskript

- **pip** (Version 23.1.2)
- **protobuf** (Version 3.20.3)
- **setuptools** (Version 65.5.0)
- **mysql-connector-python** (Version 8.0.33)
- **pyusb** (Version 1.2.1)

### 5.3.2 Backend der Website

- **APScheduler** (Version 3.10.1)
- **blinker** (Version 1.6.2)
- **click** (Version 8.1.3)
- **colorama** (Version 0.4.6)
- **et-xmlfile** (Version 1.1.0)
- **Flask** (Version 2.3.2)
- **Flask-Testing** (Version 0.8.1)
- **itsdangerous** (Version 2.1.2)
- **Jinja2** (Version 3.1.2)
- **MarkupSafe** (Version 2.1.3)
- **mysql-connector-python** (Version 8.0.33)
- **openpyxl** (Version 3.1.2)
- **pip** (Version 23.1.2)
- **protobuf** (Version 3.20.3)
- **pytz** (Version 2023.3)
- **setuptools** (Version 65.5.0)
- **six** (Version 1.16.0)
- **tzdata** (Version 2023.3)
- **tzlocal** (Version 5.0.1)
- **Werkzeug** (Version 2.3.6)

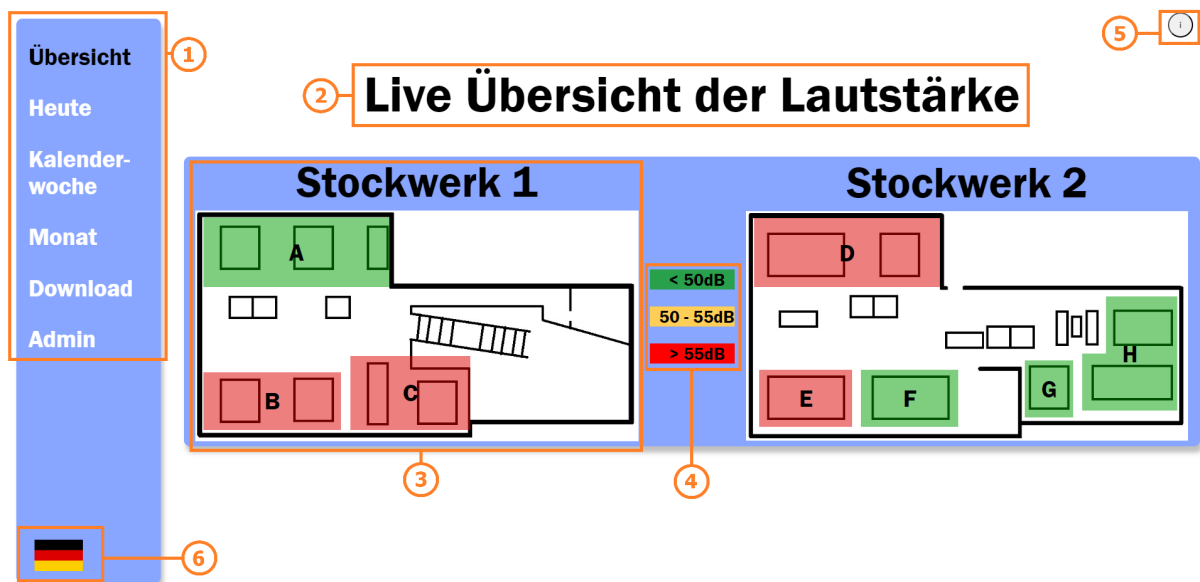


## 6. Produkthandbuch

Diese Webanwendung ermöglicht es den Mitarbeitern von AbbVie sowohl den aktuellen Lautstärkepegel als auch den Lautstärkepegel in vergangenen Zeiträumen im Gebäude 56 zu betrachten. Zudem sendet die Anwendung Lautstärkewarnungen an die Mitarbeiter sollte der Lautstärkepegel in einem Bereich über längere Zeit zu laut sein und zu einer Beeinträchtigung des Arbeitsalltags führen.

Die Funktionalitäten der Anwendung werden in folgenden Texten und Abbildungen beschrieben.

### 6.1.1 Übersicht



Bei Aufruf der Website wird dem User zuerst die Seite Übersicht angezeigt.

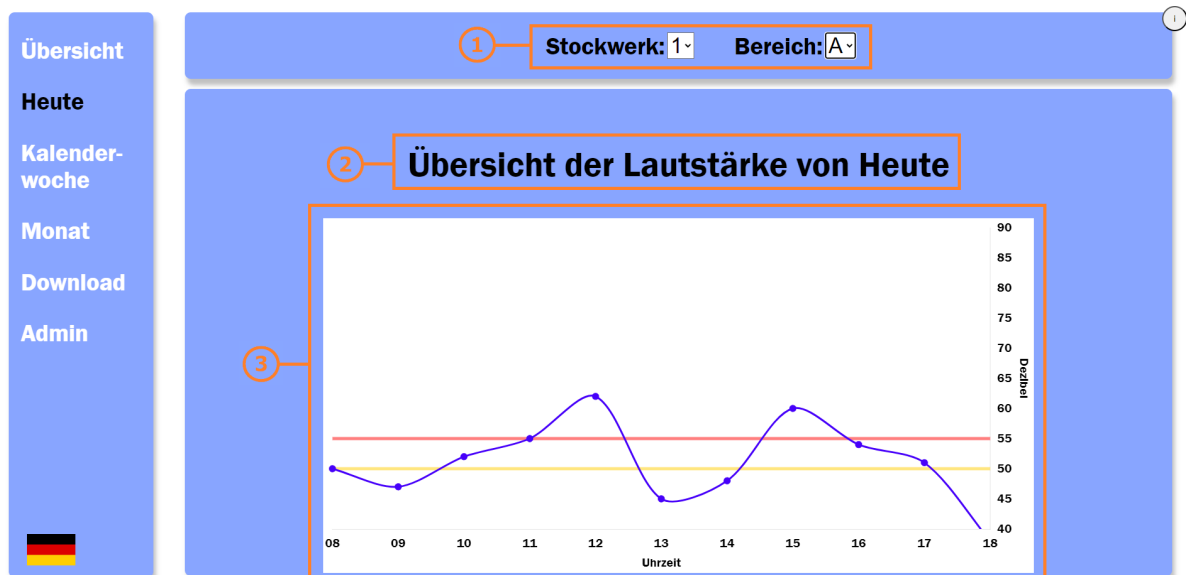
- ① Auf allen Seiten der Website, wie auch der Übersichtsseite ist das Menü zu sehen, mit welchem der User durch die verschiedenen Seiten wechseln kann.
- ② Die Seite Übersicht bietet einen schematischen Überblick der aktuellen Lautstärke im ersten und zweiten Obergeschoss des Gebäudes



56 auf dem Firmengelände von AbbVie am Standort Ludwigshafen.

③ Die Anwendung teilt die Stockwerke wie hier schematisch dargestellt in einzelne Bereiche ein. In der Anwendung färben sich die Bereiche, je nach dort vorherrschender Lautstärke, ein. Die Ansicht wird in fünf sekundigen Intervallen, anhand der in dieser Zeit gemessenen Werte, aktualisiert. Zudem kann der User durch klicken auf einen der Bereich direkt auf die [Heute](#) Seite des gewählten Bereichs wechseln. ④ Die Bedeutung der Einfärbung lässt sich über die Legende einsehen. Sollte der User Fragen kann er sich eine Erklärung der Seite über den Info-Button ⑤ anzeigen lassen. Will der Nutzer die Sprache der Anwendung ändern, kann er dies über die Flagge ⑥ bewerkstelligen. Es wird eine Auswahl zwischen Deutsch und Englisch bereitgestellt.

### 6.1.2 Heute

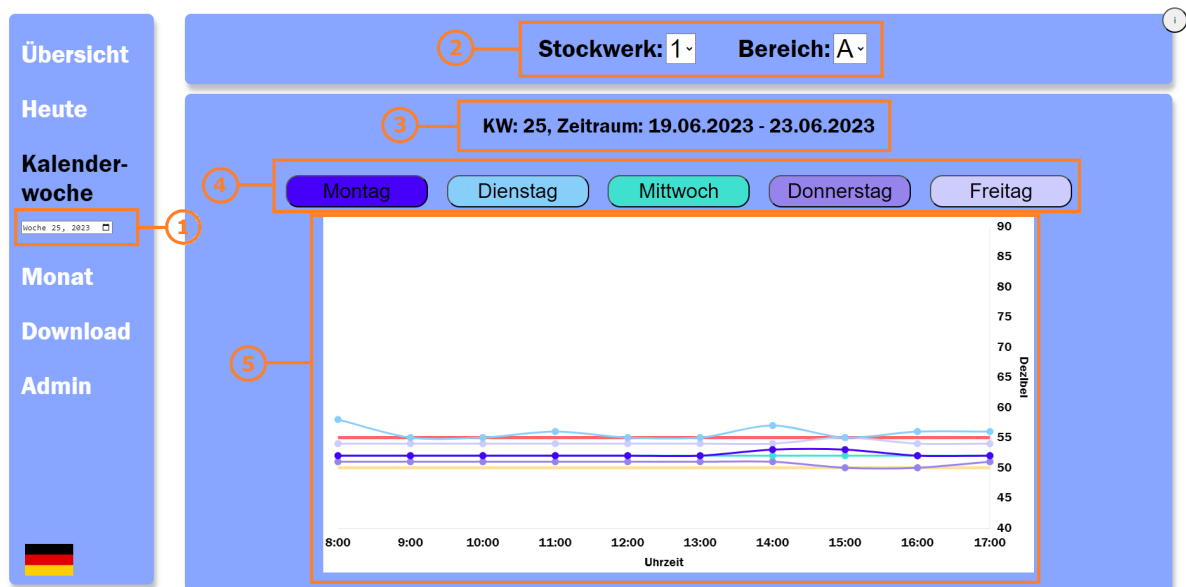


Sollte der User über das Menü auf die Heute Seite zugegriffen haben muss er zunächst Stockwerk und Bereich ① auswählen für welchen er den



Verlaufsgraphen der Lautstärke des heutigen Tages einsehen möchte. Ist der User durch Klick auf einen Bereich auf der Seite [Übersicht](#) auf die Heute Seite gelangt, ist dieser Bereich schon voreingestellt. <sup>②</sup> Auf dieser Seite kann der User den Verlaufsgraphen der Lautstärke des aktuellen Tages einsehen. Der Graph <sup>③</sup> wird von 40 bis 90 Dezibel angezeigt und passt sich automatisch den bereits gemessenen Werten an. Der Durchschnitt der Lautstärke jeder vollen Stunde wird als ein Punkt im Graphen dargestellt.

### 6.1.3 Kalenderwoche

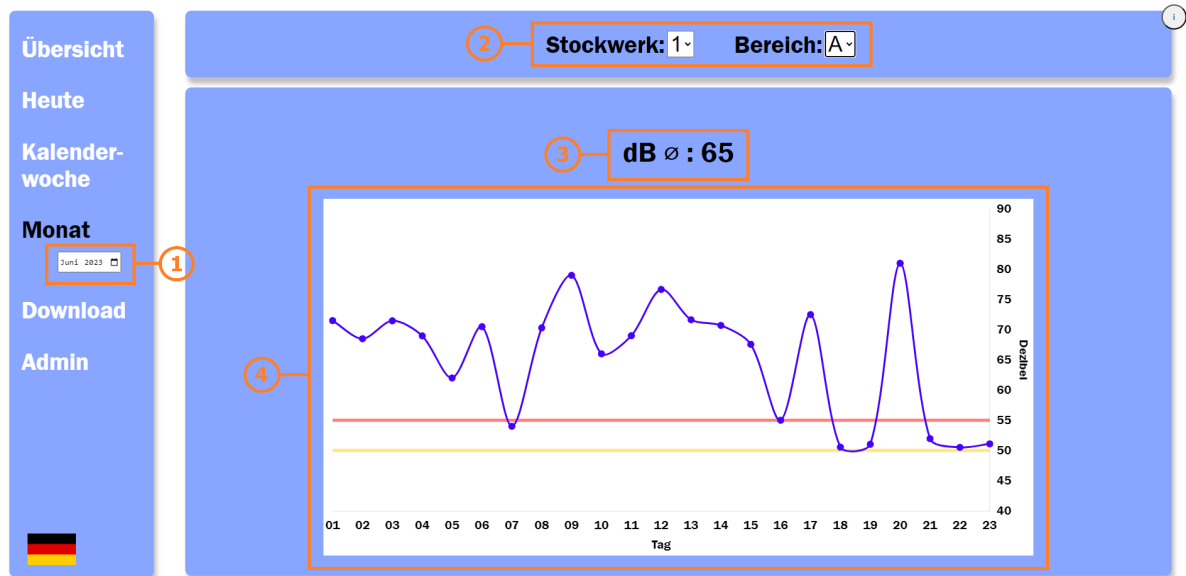


Nachdem der User die Seite Kalenderwoche ausgewählt hat, sollte er zunächst die Kalenderwoche wählen <sup>①</sup> für welche er Daten einsehen möchte. Daraufhin muss er das Stockwerk und den gewünschten Bereich auswählen <sup>②</sup>. Dem User wird angezeigt welche Kalenderwoche er ausgewählt hat <sup>③</sup>. Über die Anzeige der einzelnen Wochentage <sup>④</sup> kann der User die einzelnen Graphen der Wochentage ein-/ oder



ausblenden. Dem User werden die Verlaufsgraphen der Wochentage <sup>5</sup> der ausgewählten Woche separat angezeigt. Jeder Eintrag im Graphen entspricht dem Durchschnittswert alle in einer Stunde gemessenen Werte in eine Bereich.

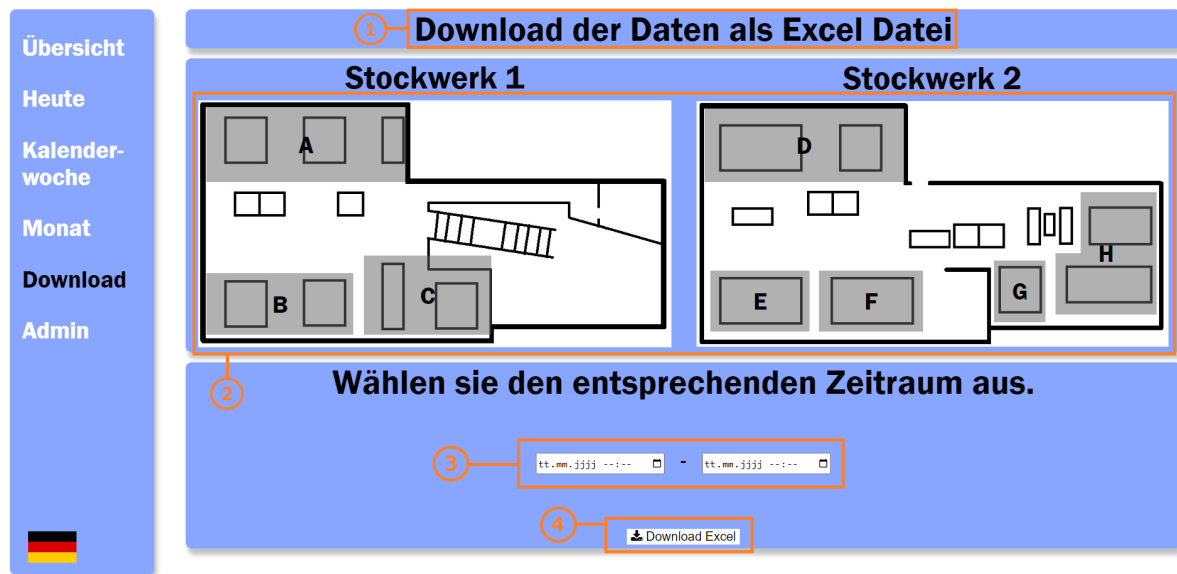
### 6.1.4 Monat



Nachdem der User die Monat Seite ausgewählt hat muss er zunächst den gewünschten Monat auswählen <sup>1</sup> für welchen er die Daten einsehen möchte. Danach muss er den Stockwerk und den Bereich auswählen <sup>2</sup> für welchen der die Daten einsehen möchte. Dem User wird der Durchschnitt der Lautstärke <sup>3</sup> aller gemessener Werte des Monats angezeigt und zusätzlich <sup>4</sup> der Verlaufsgraph der Lautstärke über den gesamten Monat. Ein Eintrag im Graphen entspricht dem Durchschnitt aller an diesem Tag gemessenen Werte.



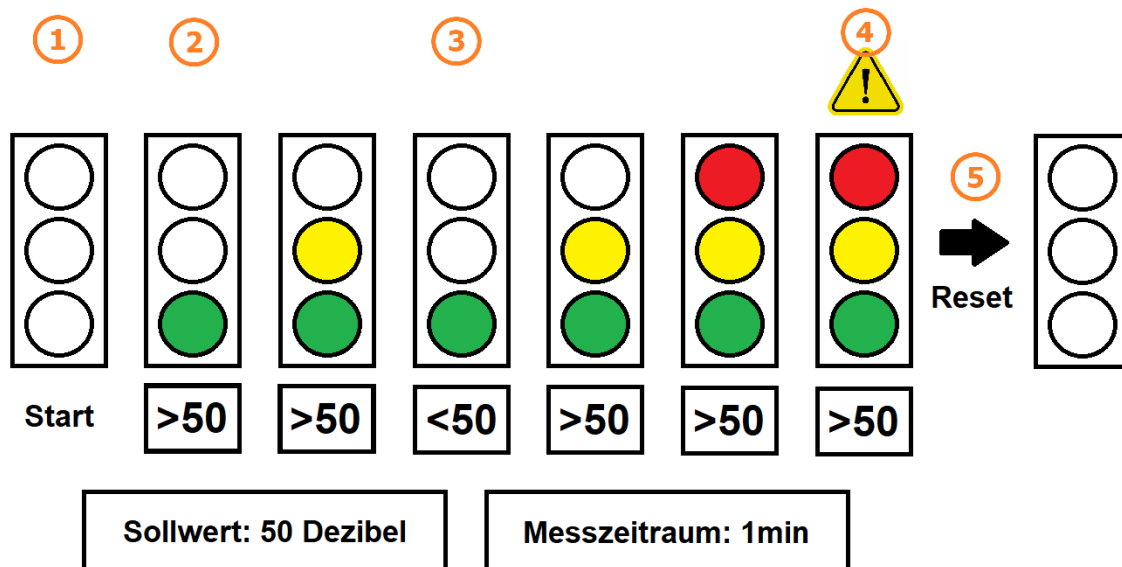
## 6.1.5 Download



Die Download Seite ermöglicht es dem User (1) die Messdaten von gewünschten Bereichen in einem vom User gewählten Zeitraum als Excel-Datei herunterzuladen. In der Übersichtsgrafik der Bereiche (2) kann der Nutzer beliebig viele Bereiche, für welche er die Messwerte als Excel-Datei herunterladen möchte, auswählen. Daraufhin kann er einen beliebigen Zeitraum wählen (3) für welchen er die Daten herunterladen möchte. Nachdem der User sowohl einen oder mehrere Bereiche und einen Zeitraum ausgewählt hat kann er mithilfe des Download-Buttons (4) die Excel-Datei herunterladen.



## 6.1.6 Warnsystem



Im Hintergrund der Webanwendung läuft das Warnsystem, welches für jeden Bereich spezifisch die Lautstärkeentwicklung untersucht. Jedem Bereich wird ein spezifischer Sollwert der Lautstärke zugewiesen, welcher auf Dauer nicht überschritten werden sollte. Das Warnsystem erfragt alle 60 Sekunden die in dieser Zeit gemessenen Werte für alle Bereiche und berechnet für jeden Bereich den Durchschnitt dieser Messwerte. Sollte der errechnete Durchschnittswert über dem Sollwert des Bereichs liegen, erhöht sich die Warnstufe des Bereichs um eine Stufe **2**, liegt der Durchschnittswert unter dem Sollwert, sinkt die Warnstufe um eine Stufe **3** bzw. bleibt auf der niedrigstmöglichen Stufe **1**. Sollte die Warnstufe eines Bereichs nach mehrfachen Überschreitungen des Sollwerts auf Stufe 4 ansteigen **4** wird eine E-Mail an die Mitarbeiter gesendet, welche auf die erhöhte Lautstärke des Bereichs hinweist. Nach Erreichen der Stufe 4 des Warnsystems wird der betreffende Bereich wieder auf die niedrigste





Stufe zurückgesetzt <sup>5</sup>. Pro Bereich können in der Stunde maximal zwei Warnungen versendet werden.

## 7. Qualitätssicherung

Im Frontend haben wir vor allem mit Hilfe von manuellen Tests und Selenium die einzelnen Seiten und Elemente untersucht und diese anhand der Anforderungen im Product Backlog und der erstellten Designvorlagen überprüft.

Um die zuverlässige und fehlerfreie Funktionalität des Backends sicherzustellen, orientieren wir uns eng an unserem Testkonzept (zu finden im Git-Repository im Ordner 'Dokumente'). Dabei orientieren wir uns an den einzelnen Anforderungen und testen diese sowie die zugehörigen Methoden isoliert mit Hilfe von Unit-Tests. Dabei simulieren wir eine Anfrage des Frontends an das Backend und prüfen die Antwort des Backends auf Plausibilität. Dabei decken wir alle Methoden ab, die von uns programmiert wurden. Wir können trotz dessen keine hundertprozentig fehlerfreie App garantieren.. Klassen innerhalb importierter Packages wurden nicht von uns getestet.