General Engineering - 6CCE3EEP

09/05/2025

Design and Integration of a Real-Time Indoor Environmental Monitoring System with Embedded Visualization using multi-sensor hardware

Ada Kanlıbaş K22052656

Supervisor: Dr. Naseem Ramli

Word count: 11837

RELEASE OF PROJECT: I **agree** to the release of my project

RELEASE OF VIDEO: I **agree** to the release of my project video

Signature:

Abstract

This project demonstrates the design and development of a portable, real-time, indoor environmental monitoring device that integrates multiple sensors for CO, CO_2, PM2.5, temperature, humidity, light, and sound. By combining Arduino-based sensor input with a mini PC running TouchDesigner for visualization, the system enables immediate and interactive feedback about indoor air quality and comfort. The solution is designed to be informing, accurate, and engaging, with reactive graphical outputs and user-friendly interfaces. Test results confirm that the system is stable, responsive, and well-suited for home or public use, offering great potential for future expansions like AI-based predictions or mobile integration.

Content

1. Introduction: Importance of Indoor Air Quality (IAQ)

"During the last two decades there has been increasing concern within the scientific community over the effects of indoor air quality on health." [1] Additionally, the development of synthetic building materials has led to a significant increase in the use of construction technologies. There is a fact that these enhancements have resulted in buildings that are more appealing and have reduced operating costs. However, the downside is that they also include the creation of indoor conditions in which pollutants are easily created and have the potential to accumulate to considerably higher concentrations than those found outside. [1]

"Indoor air quality (IAQ) without a doubt influences human health and well-being, "as individuals spend approximately 90% of their time indoors where the concentrations of some pollutants are often 2 to 5 times higher than typical outdoor concentrations." [2].Exposure to both indoor air pollutants and outdoor pollutants that enter the buildings can lead to immediate effects such as irritation of the eyes, nose, and throat, headaches, dizziness, and fatigue. Long-term exposure can be associated with poisoning(such as carbon monoxide poisoning), respiratory diseases, heart disease, and cancer. [2]

Research indicates that exposure to air pollution, whether it be for a short period of time or for a longer period of time, can have significant and long-lasting adverse effects on mental health in addition to impacts on physical health. Researchers have discovered that being exposed to high amounts of air pollution over the course of the last five years can result in a negative impact on one's quality of life, as well as an increased likelihood of experiencing depression and suicide thoughts. [3] Furthermore, it is also found that exposure to air pollution is associated with dementia, Alzheimer's disease, and Parkinson's disease. [4] In spite of the fact that these consequences might not be seen right away, being exposed to these might either directly or indirectly raise the likelihood of developing conditions of this kind in later stages of life. Understanding the hazards associated with air quality and taking measures to mitigate them becomes more important each day in order to reduce the long-term burden of illnesses caused by increased urbanisation and the growth in the levels of indoor pollutants.

Children are especially vulnerable as their brains are still developing. When exposed to poor air quality, children and adolescents are at elevated risk of bipolar disorders, schizophrenia, personality disorder, major depression, affective disorders, or suicidal ideation in adolescents and children [5]. Furthermore, a number of studies have discovered that individuals of any age who are subjected to poor indoor air quality can experience a decline in their normal day-to-day functioning. This includes both children and adults. While the performance of adults on cognitive tests may be affected [6], the abilities of children in schools may be poorer for reading comprehension and mathematics [7].

In addition "Sick Building Syndrome (SBS) refers to a situation where occupants of a building experience acute health or discomfort effects that appear to be linked directly to time spent in the building, with no specific illness or cause identified. These symptoms often improve or disappear upon leaving the building." In spite of the fact that the precise causes of SBS are not fully known, there are a number of conditions that have been determined to be connected with its occurrence. These factors include physical factors such as poor ventilation, excessive noise, high humidity, and unsuitable lighting, as well as chemical contaminants. [8] To achieve the most possible healthy indoor environment, it is necessary to take into consideration a number of aspects, including light, noise, and temperature.

Additional study has shown that these factors go beyond air pollution. In order to achieve a healthy indoor environment, it is necessary to take into consideration the thermal, visual, and auditory comfort of the space. These factors primarily have an impact on the physiological aspects of the environment. In addition, thermal, visual, and audio comfort are also considered to be components of indoor comfort, as stated by Frontzcak, who examined 891 relevant titles. He defines each of these components using different characteristics. According to the standard, the thermal environment—which is the most significant—visual conditions, and sound environment are all determined by a wide range of physical factors, such as radiant temperature, sound frequency, and light colour. [9] These factors may not be affecting any health issues directly, but the fact that it may disrupt comfort or raise SBS is enough to consider while trying to improve the health conditions of indoor spaces.

Summary of Health Impacts of Specific Environmental Factors

- **Temperature**: Maintaining an optimal indoor temperature is crucial for comfort and health. Extremes in temperature can raise the risk of conditions such as cardiovascular diseases and respiratory disorders.
- **Humidity**: Proper humidity levels (40%-60%) are essential to prevent the growth of mold and dust mites, which can trigger asthma and allergic reactions. High humidity fosters mold proliferation, leading to respiratory issues and insomnia, while low humidity can cause skin dryness and irritation. [10]
- **Carbon Dioxide (CO_2)**: Elevated CO_2 levels indicate inadequate ventilation and can cause headaches, dizziness, and cognitive impairments. Prolonged exposure to high CO_2 concentrations may lead to chronic respiratory problems. [11]
- **Carbon Monoxide (CO)**: Carbon monoxide poisoning exerts its most harmful immediate impact on organs that have elevated oxygen demands, specifically the heart and brain. Consequently, individuals diagnosed with ischaemic heart disease face an elevated risk. [12][13]
- **Particulate Matter (PM2.5 and PM10)**: Fine particulate matter has the ability to penetrate the respiratory system, leading to irritation and worsening pre-existing conditions such as asthma and bronchitis. Prolonged exposure elevates the likelihood of developing cardiovascular diseases and lung cancer. [14]

- **Light Intensity**: Sufficient illumination is essential for visual comfort and can significantly impact mood and productivity. Inadequate illumination can result in visual discomfort and may also exacerbate symptoms of depression.
- **Sound Levels**: Sound Levels: Continuous exposure to high noise levels may lead to increased stress, disruptions in sleep patterns, and potential hearing impairment. It could also cause heightened risks for individuals with mental health conditions, such as schizophrenia or autism, even potential cardiovascular impacts. [15]

These factors, along with other pollutants from household activities, industrial byproducts, and poor ventilation systems, can cause respiratory issues, allergic reactions, cardiovascular diseases, and long-term neurological disorders. Considering the increasing level of indoor pollution caused by modern construction materials, energy-efficient yet poorly ventilated structures, raising dependence on artificial heating and cooling systems, as well as pollutant distribution technologies, there exists a growing requirement for continuous, real-time assessment of indoor air quality (IAQ). This system should enable the immediate detection of hazardous conditions, allowing rapid action and preventing the risks associated with exposure to harmful environmental factors. The deployment of these monitoring tools is more essential for at-risk groups, such as children, the elderly, and those with pre-existing respiratory conditions, who are more vulnerable to the negative effects of air pollution. In addition to individual health advantages, widespread implementation of indoor air quality monitoring has the potential to boost workplace efficiency, enhance educational settings in schools, and support public health by mitigating pollution-related health issues.

2. Literature Review

2.1 Existing Indoor Environmental Monitoring Solutions

When people spend a major portion of their time indoors, where a variety of factors might have an effect on their health and well-being, monitoring the environment inside has become an ever more crucial habit. The monitoring of indoor air quality (IAQ) and other environmental factors has been the subject of a number of commercial and academic solutions to date. This section presents an overview of the solutions that are now available, observing both their strengths and drawbacks, with a particular emphasis on the affordability, real-time data availability, and user-centric characteristics of these potential solutions.

1. IQAir's AirVisual Pro

This device became popular in the IAQ monitoring market in recent years. It can monitor particulate matter (PM2.5), carbon dioxide (CO_2), temperature, and humidity while showing some good features.

The performance of the device is a critical factor. The AirVisual Pro operates on a sophisticated operating system designed for air quality monitoring, and it has the capability to store data locally for up to one year, which is one of the most extensive local data storage capacity in air quality monitors. The device illustrates impressive responsiveness, even while operating a sophisticated OS with minimal occurrences of lag. The system maintained stability throughout its operation, presumably owing to the quality components supporting this high-resolution display monitor. The software appears to be designed as a comprehensive weather station; however, it fails to provide any real weather data for forecasting purposes [17]. The relevant table: Accuracy(PM2.5) highlights for an AirVisual Pro laboratory evaluation regarding its accuracy [18]:

Accuracy ($PM_{2.5}$)

$$A\,(\%) = 100 - \frac{|\bar{X} - \bar{R}|}{\bar{R}} * 100$$

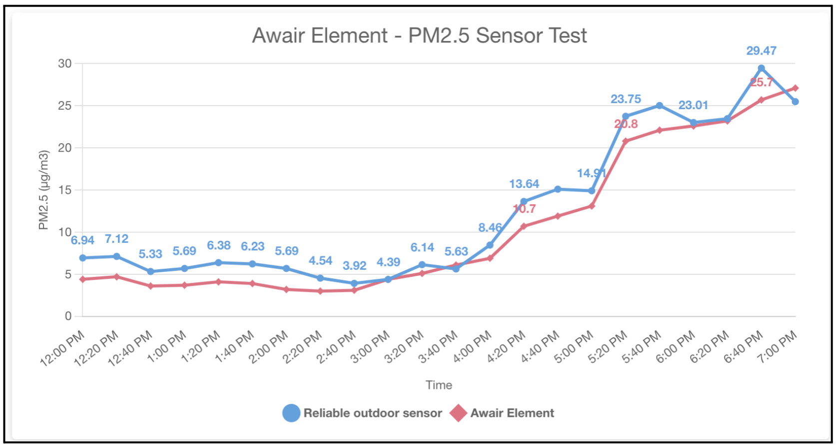
Steady state #	Sensor Mean (µg/m³)	FEM GRIMM (µg/m³)	Accuracy (%)
1	11.2	10.18	89.7
2	17.5	15.20	84.9
3	64.6	59.62	91.7
4	172.4	153.11	87.4
5	338.2	270.07	74.8

The device demonstrates a high level of accuracy, although it tends to underreport PM2.5 concentrations. Nonetheless, there exist more precise monitors available at a reduced cost. It also shows that its accuracy will be adversely affected by various environmental factors, so the device is not suitable for outdoor use. [17]

Strengths:	*Weaknesses:*
• **High-Quality Sensors** for accurate readings. • **User-Friendly Interface** with a clear display and an intuitive app for historical data.	• **Limited Parameter Monitoring** that lacks sensors for other environmental factors such as other gases or factors • **Cost** makes it less accessible for budget-conscious consumers.

2. Awair Element

The Awair Element is designed to monitor five key aspects of indoor air: temperature, humidity, CO_2, chemicals (volatile organic compounds - VOCs), and PM2.5. It provides real-time feedback and personalized recommendations through its mobile app. However, there is one general problem with VOC sensors: these sensors group all VOC sources and then determine the number of VOCs and pollution based on that. Many studies have shown that this is not the best way to measure the concentration of VOCs [19]. The test results for the PM2.5 sensor regarding its accuracy can be seen in the table: Awair Element - PM2.5 Sensor Test [20]. It shows small deviation in outdoor space even though still in the acceptable range



Strengths:	Weaknesses:
broad spectrum of air quality indicators, offering a holistic view of IAQ. **Smart Home Integration** with platforms like Google Assistant and Amazon Alexa, allowing for seamless integration into smart home ecosystems.	**Cost-Performance** is a feature where this device is left behind a bit **Data Handling** is not as optimized as it can be

3. Nest Protect

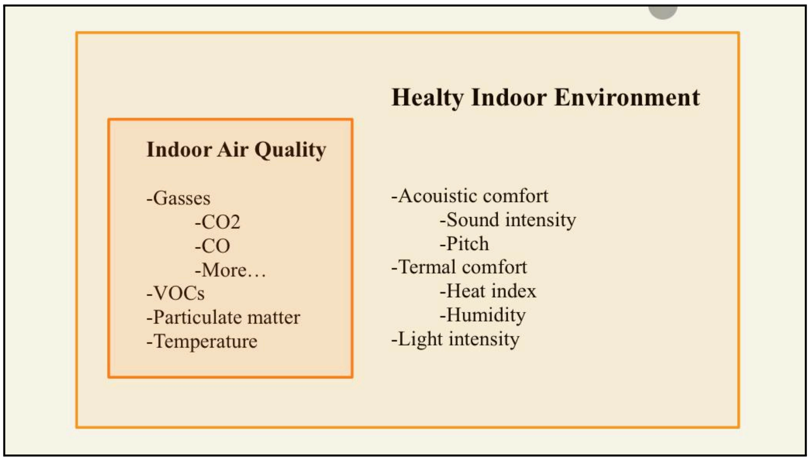
The Nest Protect serves as more than a conventional smoke detector; it integrates critical safety features with advanced technological capabilities, ensuring full safety for your residence. The Nest Protect is engineered to deliver early warnings and comprehensive information, with the objective of reducing stress levels. "It utilizes ses two different sensors to detect which type of fire is burning – a raging, fast-burning one or slow, smouldering one – as well as a humidity sensor so that it receives fewer false alerts from steam" [21]

Strengths:	Weaknesses:
Real-Time Alerts when air quality deteriorates, enabling prompt action. **Detailed focus** on the feature it is designed to monitor for	**Sensor Accurac**y is reportedly inconsistent in sensor readings over time. **Limited Environmental Scope** which only focuses on one feature, not caring other ones

2.2 Problem Statement

Although there are numerous indoor monitoring systems that offer valuable insights into the quality of the air, they all share a few common limitations. There can be an opportunity to create deeper and user-focused indoor environmental monitoring systems if these gaps are filled.

Mainly, They are not suitable for broad usage because they either focus on a particular feature (such carbon dioxide and particulate matter) or need an expensive infrastructure. When they are examined, this is something that can be identified. The main goal of current systems is to monitor air quality, but they ignore other crucial environmental factors that significantly affect general well-being, such as light, sound, and thermal comfort. The following figure illustrates the key distinction between indoor air quality (IAQ) and a healthy indoor environment.



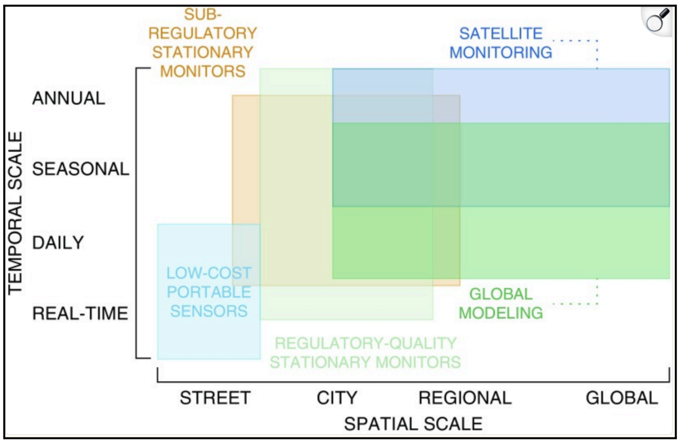
Indoor Air Quality (IAQ) focuses strictly on airborne pollutants such as Carbon Dioxide (CO_2), Particulate Matter (PM2.5, PM10), Volatile Organic Compounds (VOCs).... On the other hand, Healthy Indoor Environment (HIE) expands beyond IAQ and includes mental comfort factors, which contribute to long-term well-being and productivity, such as Noise Levels (acoustic pollution), Light Quality (brightness, flicker rate, color rendering), Thermal Comfort (temperature & humidity control) The diagram emphasizes that IAQ is only a subset of a truly healthy indoor environment, which encompasses both immediate and long-term comfort. Despite advancements in indoor environmental monitoring, existing solutions fail to provide a holistic view of indoor well-being. The primary technical side of the shortcomings include:

- **Narrow Focus on Airborne Pollutants**
  Most monitors concentrate on air-related parameters, neglecting many other environmental factors such as noise pollution, other gasses, light pollution,... which also influence occupant health and comfort. For example IQAir's AirVisual Pro tracks only air quality metrics (e.g., PM2.5, CO_2, VOCs) while neglecting other indoor comfort factors such as PM1.0 which is the most important one, lighting, sound levels, and thermal comfort.
- **Expensive and Complex Infrastructure**
  Traditional IAQ monitors relied on stationary, high-cost equipment that limited accessibility and adoption [22]. While recent portable, real-time IAQ monitors have emerged, they still lack multi-sensor integration to provide a complete indoor environment assessment. Even though this shows the ability to support the demand, High-quality IAQ monitors still come with a substantial price tag, limiting their accessibility to a broader audience.
- **Lack of Immediate, Actionable Feedback**
  Most existing systems do not offer real-time alerts for users to take immediate corrective action. In addition, Many lack integration with visual or interactive elements to effectively communicate health risks.

- **Limited Consumer Engagement & Accessibility**
  Many solutions cater to industrial applications or specialized markets, making them unaffordable for everyday users. This results in No comprehensive solution that provides a user-friendly interface, real-time visualization, and multi-sensory tracking with actionable suggestions.

A gap that needs to be bridged exists not only on the technical side, but on the other perspectives as well where the market, trends and the demand is regarded. "Historically, monitoring indoor air quality (IAQ) has been challenged by the inadequacy of outdoor pollution measurement technologies, due to differences in pollutant sources, concentrations, and airflow patterns."[23] As a result, indoor air quality concerns were often **overlooked** in favor of outdoor pollution studies. until the 1970s, the health effects of indoor air pollution received relatively little attention from the scientific community, with most public concern directed at outdoor pollution rather than the indoor spaces where people spend most of their time [24].

However "the latest developments in sensor technology are currently oppening the way "lower-cost, easy-to-use, portable air pollution monitors (sensors) that provide high-time resolution data in near real-time," making comprehensive IAQ monitoring increasingly feasible"[22] and widespread than ever before. This shift has created new opportunities to track multiple environmental factors, providing users with efficient insights into their current indoor conditions. As a result of the increasing feasibility of complicated monitoring technologies, enjoying the benefits of a healthier indoor environment became possible for regular users as well .



Graph of use cases of monitoring technologies

This graphic is an indicator of how air quality monitoring across different time and space settings can change regarding choice of the device. When affordable portable sensors are observed, it is found that they give real time and location focused data, unlike satellite systems with larger time spans. This allows for the gap between personal indoor health management and large-scale environment monitoring to be closed. This will ensure a more accessible and sympathetic approach to environmental well-being.These advances have made building air quality monitoring easier, meeting the increased need for up-to-date environmental information.

Not only the technical side, the historical side, or the cost, but also the market and the demand points out a big gap. The health technology sector has experienced remarkable growth in recent years, reflecting a significant surge in consumer interest and demand for health-monitoring devices. By 2025, the global market for digital health is projected to reach approximately $197.88 billion, with an expected annual growth rate of 6.88% from 2025 to 2029. Similarly, it is expected that the medical technology market will demonstrate a steady annual growth rate of 5.25 percent from 2025 to 2029. [25] with wearable devices like the Apple Watch and the WHOOP strap gaining popularity. The WHOOP strap, for instance, is designed to inform users to maximise their health and performance by providing personalised recovery scores and exertion recommendations based on continuous monitoring. There is a growing consumer interest in devices that offer real-time health data, which suggests that there is a market opportunity for comprehensive indoor environmental monitoring systems that integrate multiple sensors and provide user-friendly visualisation tools. This trend of tracking your health with more detailed data indicates that consumers are becoming more interested in these types of devices.

This project proposes the creation of a cheap, multi-sensor, real-time monitoring system that incorporates numerous environmental indicators into a visualisation platform (Touchdesigner) which is is a node-based visual programming software used for real-time interactive multimedia content creation, combining data input, logic, and visual output in one dynamic interface. The goal of this project is to meet the observed market gap. This solution intends to provide users with quick insights about their interior environment by using developments in sensor technology and aligning itself with current trends in health technology. This will enable users to make educated decisions that will improve their health and well-being.

This is the perfect time for such a device since there is a growing need for real-time health monitoring, and there is also the capacity to be able to fulfil this demand in a more affordable, accurate, and efficient manner than has ever been possible previously. Not only does it provide benefits that are practical, but it also fulfills the preferences of consumers for health solutions that are engaging and interactive. In addition, this technique fulfills the functional needs of complete environmental monitoring.

2.3 Objectives and Methodology

- **Construct a hardware network for the gathering of environmental data:** Implement an integrated setup with multiple sensors connected to two Arduinos for robust data acquisition.
- **Implement a data processing system to analyze and filter sensor data:** Develop Python scripts that preprocess and smooth raw sensor data through moving averages and other filtering techniques.
- **Develop a real-time visualisation dashboard utilising TouchDesigner software:** Build an interactive, node-based dashboard that displays sensor readings and visualizes environmental trends in real time.
- **Establish a notification mechanism for extreme environmental conditions:** Create threshold-based alerts that trigger visual warnings when sensor values exceed predefined safe levels.

- **Support easy interaction with the data for the user:** Design a user-friendly GUI that provides both detailed numerical data and intuitive visual feedback for quick decision-making.
- **Integrate all components into a portable device:** Assemble the hardware components inside a vintage radio box, ensuring neat cable management and secure mounting for portability and ease of use.
- **Validate system accuracy through calibration and testing:** Conduct systematic calibration using references and perform comprehensive testing to verify data accuracy and system reliability.

2.4 Sensor Technologies & Justification

After consideration, each sensor was selected with care for the project. Size, measurement range, and Arduino compatibility were considered mainly, voltage and quantity of wires needed being the second thought. After careful consideration of these parameters, the following sensors were chosen. To reduce the number of sensors in the technology network, the hardware network includes many sensors that may provide a range of data. These multi reading sensors simplify testing and calibration, simplify hardware network, and make it easier to attach an Arduino IDE to a data visualisation program (Touchdesigner). **Each sensor's purpose**:

- **DHT22 for Temperature, Humidity and Heat Index:** Impacts thermal comfort, productivity, and health. Important for preventing mold growth, maintaining comfort.
- **SEN0159 for CO_2:** High levels indicate poor ventilation, can cause drowsiness, cognitive decline.
- **SEN0132 for CO:** reads carbon monoxide levels which is hazardous at even low levels
- **PMS5003 for Particulate Matter:** Tracks PM1.0, PM2.5 and PM10, which are linked to respiratory diseases.
- **BH1750 for Light Intensity:** Ensures optimal lighting for workspaces, reducing eye strain.
- **Basic Microphone for Sound:** Identifies noise pollution, which affects stress and concentration.

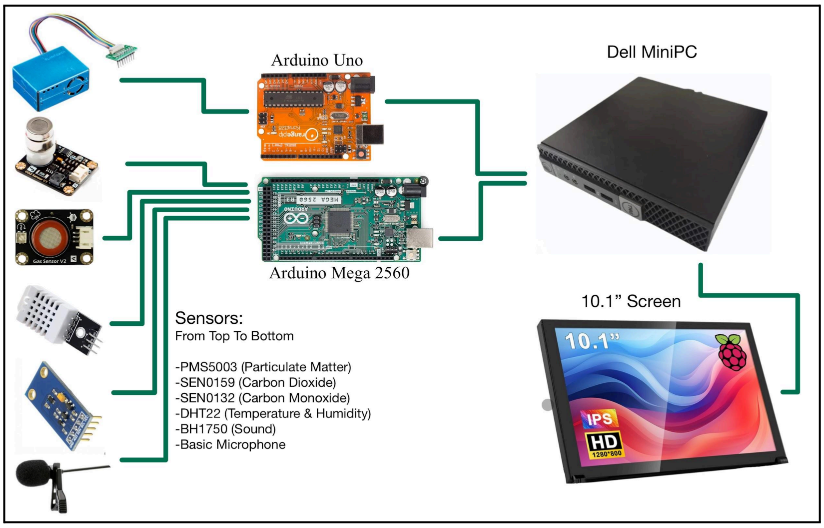
2.5 Threshold Values for Each Detectable Feature:

Feature	Ranges	Risk-Comfortable Level
Particulate matter (10 micron) (µg/m^3)[31]	<70 70-170 >170	N/A Meidum High
Particulate matter (2.5 micron) (µg/m^3)[32]	<15 15-45 >45	N/A Medium High
Particulate matter (1 micron) (µg/m^3)[33]	<50 50-150 >160	N/A Medium High
Sound (Db) [34]	<50 50-80 >80	N/A Medium High
Light (Lux) [35]	<200 200-500 500-1000 >1000	High Medium N/A High

Feature	Ranges	Risk-Comfortable Level
Temperature (C°) [26]	< 16 17–21 21-26 27-29 >30	High Medium N/A Medium High
Humidity (%) [27]	<25 25-30 30-65 >65	High Medium N/A High
Heat Index (C°) [28]	<26 27-36 >36	N/A Medium High
CO2 (ppm) [29]	<600 600-1000 >1000	N/A Medium High
CO (ppm)	0-20 20-50 >50	N/A Medium High

3. System Design & Production

3.1 Hardware Architecture



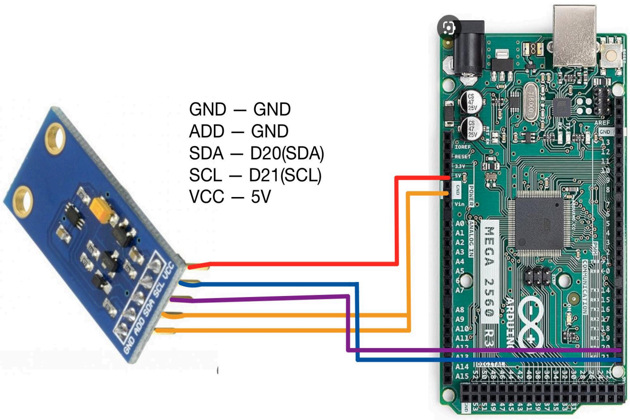
The fully operational system uses six environmental sensors: the BH1750 for light measurement, the DHT22 for temperature and humidity, the PMS5003 for particulate matter detection, the SEN0159 for CO_2 levels, the SEN0132 for carbon monoxide, and a microphone for measuring sound levels. A small PC, which acts as the main processing and visualising device, has data from these sensors sent via two Arduinos.

While a separate Arduino Uno is used just for the PMS5003 and to power the SEN0159, an Arduino Mega gathers data from the BH1750, DHT22, and SEN0132, SEN0159 to maximise data flow. This separation became essential because of communication problems probably related to timing conflicts or sensor interference trying to run the PMS5003 on the same board. Furthermore, SEN0159's accuracy and dependability are much improved when it is linked to an independent power source as it needs high voltage and constant current.

The gadget runs on its own; the little PC is a basic component of its design, hence no outside computer is required. Unlike conventional installations where Arduinos provide raw data to a PC only for short-term applications, this method is dynamically adaptable and allows ongoing data processing and display. Real-time calibration changes using external data sources helps to improve accuracy.

This system has immense upgrade potential thanks to its Arduino Mega for expansion, a secondary Arduino Uno for dedicated tasks, and an onboard PC for advanced processing. Additional sensors, improved algorithms, and future software updates can be integrated seamlessly, making it a highly adaptable, long-term solution for real-time indoor environmental monitoring. The sensor diagrams, assembly process, testings and calibrations can be found below for each specific sensor.

1. BH1750 (Light Sensor) - Testing & Calibration



The BH1750 is a digital ambient light sensor that communicates via the I²C protocol, making it easy to integrate with microcontrollers and PCs. Assembly begins by wiring the VCC and GND pins to the power supply (5V), while the SCL (clock) and SDA (data) lines are connected to the corresponding I²C pins on the microcontroller (e.g., an Arduino or ESP32). An additional 10kΩ pull-up resistor may be used but not necessary on the SCL and SDA lines to ensure signal stability.

This sensor continuously measures illuminance (lux) and transmits the data digitally to the microcontroller, which then relays it to the PC via USB serial communication. The sensor updates readings in real-time with a fast response time, making it ideal for monitoring changes in indoor light conditions. One of its key advantages is automatic gain control, which allows for high precision measurements even in low-light conditions.

Testing:

- Basic Test: Placed the sensor in different lighting conditions (e.g., direct sunlight, shade, no light) and observed if the readings change accordingly.
- Comparative Test: Used a smartphone lux meter app (like Lux Meter on Android or iOS) and compared its readings with the BH1750 output.

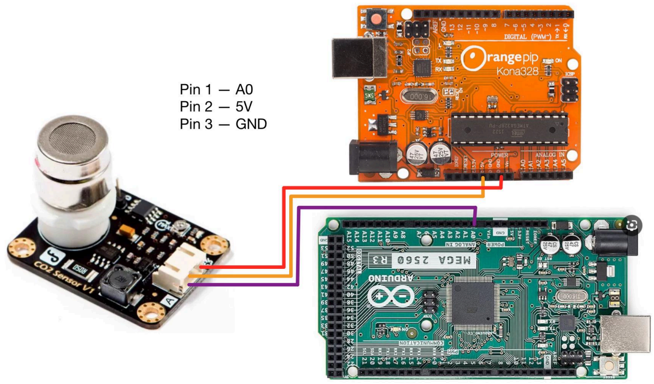
Calibration: If the values were consistently higher or lower than expected, indicating systematic error, an offset correction in your code is applied.

Results: The sensor passed both basic and comparative tests with error less than 7% error, therefore no calibration nor correction code is applied. The comparative test's results are below:

Device/Trial #	1	2	3	4	5
BH1750	0 Lux	14 Lux	139 Lux	265 Lux	1275 Lux
Lux-meter App	0 Lux	15 Lux	136 Lux	259 Lux	1289 Lux
ERROR %	0%	6.6%	2.1%	2.3%	1.08%

2. SEN0159 (CO_2 Sensor) - Testing & Calibration

The SEN0159 CO_2 sensor is a digital sensor that works with the UART (Universal Asynchronous Receiver-Transmitter) system. This means that it can send directly in digital form, without having to go through an ADC conversion. To put it together, you need to connect its VCC pin to 5V, GND pin to ground, and TX/RX (send/receive) pins to the microcontroller's UART port.



The sensor constantly measures the amount of CO2 in the air in parts per million (ppm) and sends the information over a serial link once it is turned on. This information is read by the microcontroller and sent to the PC so that it can be analysed. The sensor has temperature correction built in, which makes it more accurate when the temperature changes. To keep its accuracy over time, it also needs to be calibrated on a regular basis using fresh air from outside.

Testing: Placed the sensor in a well-ventilated area (outdoors) to check if it stabilizes near 400 ppm CO_2 (ambient air).

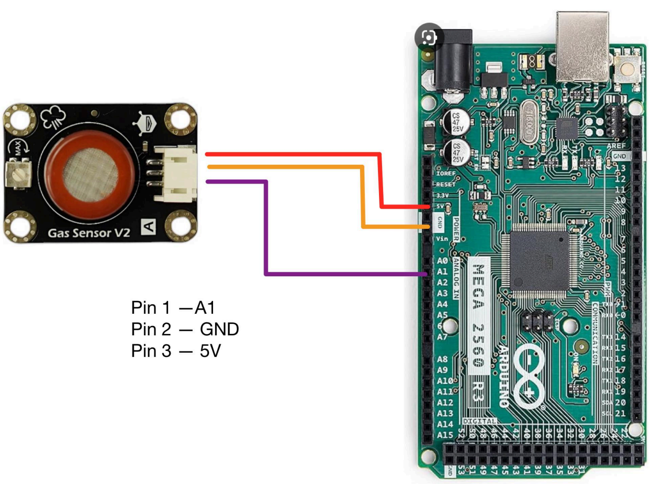
Calibration: Left the sensor outdoors for at least 2 hours to stabilize. an offset adjustment is applied in your code if it does not read ~400 ppm.

Results: After leaving the sensor outside for a decent amount of time, it is realized that the zero_point_voltage of the sensor is not in parallel with the data sheet (probably due to built-in potentiometer's settings), therefore a series of readings with their voltage values was read. From these values, both with basic mathematical calculations and troubleshooting, manual calibration of the Zero-point-voltage variable of the sensor is done. This helped with the curve structure of the data analyzation for co2, giving more stable and accurate readings.

Voltage and ppm values for SEN0159 after left in outside for 3 hours

2.52V	1795 ppm
2.53V	1707 ppm
2.54V	1707 ppm
2.53V	1655 ppm

3. SEN0132 (CO Sensor) - Testing & Calibration



The SEN0132 CO sensor is an analog gas sensor that requires proper calibration before use. It is wired by connecting its VCC to 5V, GND to ground, and the AO (analog output) pin to an analog input on the microcontroller. Since it provides an analog voltage proportional to CO concentration, the microcontroller must convert this signal into a digital value using an ADC (Analog-to-Digital Converter).

The microcontroller periodically reads the voltage output and sends the CO concentration data to the PC over a USB serial connection. The sensor requires a preheating time of at least 24 hours for optimal accuracy and should be placed in a well-ventilated area to prevent sensor saturation. An additional 10kΩ pull-up resistor may be used but not necessary on the SCL and SDA lines to ensure signal stability.

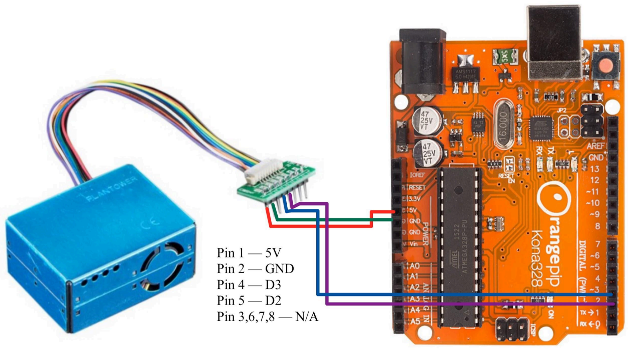
Testing:
- **Baseline Test:** Checked if it reads 0 ppm in fresh air.
- **Controlled Exposure:** Lighted a matchstick or candle and let it burn near the sensor for a few seconds before extinguishing. This should have produced a small amount of CO which is enough to be detected by the sensor.

Calibration: It is placed in fresh air for 2 hours and the baseline is adjusted if it does not read 0 ppm.

Results: The sensor passed controlled exposure, where a spike was seen when candle was burning, However after zero calibration in fresh air, there was a systematic offset in the reading which resulted in a correction code.

Device/Trial #	1	2	3	4	5
BH1750	15 ppm	20 ppm	16 ppm	18 ppm	16 ppm
Reference	0 ppm	0 ppm	0 ppm	0 ppm	0 ppm

4. PMS5003 (Particulate Matter Sensor) - Testing & Calibration



The PMS5003, a digital laser-based particulate matter sensor is capable of measuring PM1.0, PM2.5, and PM10 concentrations. A laser beam reflects light off of airborne particles to determine their size and concentration after a fan-driven device sucks air into the chamber. A microcontroller is attached to the circuit's VCC (5V), GND, TX, and RX pins. The TX and RX pins are connected to the microprocessor's UART port since it employs UART serial communication.

The monitor transmits data packets containing real-time PM concentration values as soon as it is turned on. The microcontroller reads it and forwards it to the PC for additional analysis. The PMS5003 must be positioned appropriately for readings to be reliable. To prevent particle counts from being thrown off, it should be placed in a location with good ventilation but away from direct wind sources like fans.

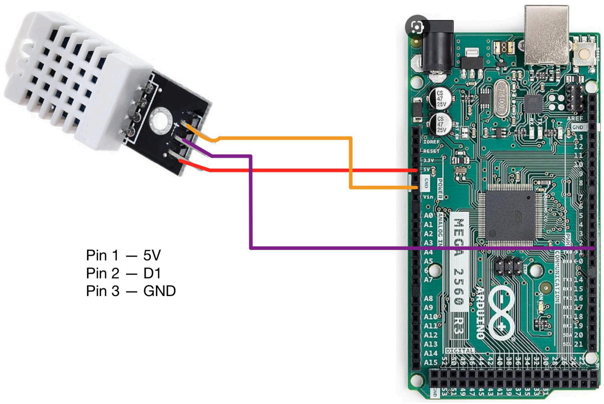
Testing:
- **Baseline Test:** Ran the sensor in clean indoor air and recorded PM2.5/PM10 values.
- **Controlled Test:** Burn a piece of paper or incense nearby and see if PM values increase significantly.

Calibration: If the values were consistently higher or lower than expected, indicating systematic error, an offset correction in your code is applied.

Results: The sensor passed both basic and comparative tests, with visible spikes in the presence of a burning paper and relatively low values in the fresh air. Therefore no calibration nor correction code is applied.

5. DHT22 (Temperature & Humidity Sensor) - Testing & Calibration

A single-wire digital interface is used by the DHT22, a digital temperature and humidity sensor. VCC, GND, Data, and NC (not connected) are its four pins. In order to assemble, VCC must be connected to 5V, GND to ground, and Data to a microcontroller digital input pin.



Temperature and humidity measurements from the DHT22 sensor are transmitted in 40-bit packets, which the microcontroller decodes and transmits to the PC over USB serial connection. For the majority of real-time monitoring applications, the sensor updates data around every two seconds. To avoid inaccurate readings, it should be maintained away from dampness and direct airflow.

Testing:
- **Basic Test:** Held the sensor inside my hands to check if the temperature rose gradually. also Placed it near an ice pack or in a cooler environment and saw if it reacted.
- **Compare Against a Reference:** Used a cheap digital thermometer/hygrometer as a reference. Before that, placed both in the same environment for 10-15 minutes and compared readings.
- **Humidity Check:** Placed it inside a sealed bag with a damp paper towel and checked if humidity readings increased to 90-100%.

Calibration: If it consistently read too high or too low, a software correction factor was applied

Results: The sensor passed both basic and comparative tests in the beginning for each factor it reads. However, when the device is fully assembled, the internal heat created by the device started affecting the temperature reading. Therefore, a subtractive correction code is implemented after calculating the difference in temperature between the inside and outside of the device. Since one error was a systematic one while the other error was only caused after PC heats up, one correction is done at the beginning while the other one after 10 mins

Condition/Trial #	1	2	3	4	5
Outside with PC	38	37.6	38.5	36.2	37
Outside without PC	23.4	22.8	23.6	22.3	23
Actual Outside	15C°	15C°	15C°	15C°	15C°

6. Audio Microphone

A simple commercial USB microphone handles the audio sensing for the reactive system. It links directly to the small PC via USB, bypassing the Arduino network completely. This was made so that real-time audio input could be of better quality and not be limited by the way microcontrollers can handle analogue sound.

The PC sees the microphone as a normal audio input device, so data can be used right away for video effects or ambient analysis. The data is fed directly into the visualisation software (TouchDesigner). The microphone works right out of the box with consistent sensitivity and stability, so it's a simple but useful addition to the multi-sensor system. It doesn't need any extra calibration or tuning.

3.2 Data Processing and Handling Pipeline

For Arduino Mega (COM4) [36] [37] [38] [39]

The Arduino Mega is an important part of the system's multi-sensor data collection network because it is the main microprocessor that collects data from four sensors: the DHT22 (temperature and humidity), the SEN0159 (CO_2), the SEN0132 (CO), and the BH1750 (light intensity). The Mega was picked over more common boards like the Uno because it has more I/O ports, better memory management, and more serial ports. All of these features are necessary for reliable connection with many sensors running at the same time.

With its extra pin ports, the current setup allows for a lot of room for future growth or sensor expansion, and it only uses about 4% of program storage space and 5% of dynamic memory. This makes the platform for testing and calibration very flexible, and it promises smooth real-time data capture that doesn't put too much strain on the system.

```
#include "DHT.h"
#include <BH1750.h>
#include <Wire.h>
#define DHTPIN 2
#define DHTTYPE DHT22
DHT dht(DHTPIN, DHTTYPE);
BH1750 lightMeter;


const float V_REF = 5.0;  // Reference voltage
const int ADC_RES = 1023; // ADC resolution

const float V_CLEAN_AIR = 0.4;  // Voltage in clean air (from datasheet) for CO
const float SENSITIVITY = 0.03;  // Sensor sensitivity (mV per ppm) for CO

// Calibration values for CO2 from datasheet
const float DC_GAIN = 8.5;  // Amplifier Gain
const float ZERO_POINT_X = 2.602; //  Voltage at 400 ppm CO2
const float ZERO_POINT_VOLTAGE = 0.3305; // Voltage at 400 ppm CO2
const float MAX_POINT_VOLTAGE = 0.265;  // Voltage at 10,000 ppm CO2
const float REACTION_VOLTAGE = 0.059;   // 0.059 Voltage drop from 400 ppm to 1000 ppm
const int READ_SAMPLE_INTERVAL = 50; //how many samples you are going to take in normal operation
const int READ_SAMPLE_TIMES = 10; // time interval(in milisecond) between each samples in normal operation
float CO2Curve[3] = {ZERO_POINT_X, ZERO_POINT_VOLTAGE, (REACTION_VOLTAGE / (2.602- 4))};
```

This first part of the Arduino Mega script sets up the core libraries and constants needed for the system's operation. Libraries specifically for DHT22 (temperature & humidity) and BH1750 (light intensity) sensors are needed, and the sensors are initialized with the appropriate pin.

After the sensor setup, a range of constants are defined; such as reference voltage, ADC resolution, and calibration values for the SEN0132 (CO) and SEN0159 (CO_2) sensors. These values are mostly copied from the related data sheets but tuned for the system's unique operating environment and behavior. This foundational section ensures that each sensor's data is accurately interpreted once reading and processing begin followingly.

```
void setup() {
  pinMode(22, INPUT); //for audio sensor
  Serial.begin(9600);
  Serial.println(F("STARTING"));
  Wire.begin();  //lightsensor
  lightMeter.begin();  //lightsensor
  dht.begin();
}

void loop() {
delay(2000); // Wait a few seconds between measurements.
  float h = dht.readHumidity(); // Reading takes about 250 milliseconds!
  float t = dht.readTemperature();  // Read temperature as Celsius (the default)
  float hic = dht.computeHeatIndex(t, h, false); // Compute heat index in Celsius (isFahreheit = false)
  float SVCO = analogRead(1);  // Read CO sensor
  float SVCO2 = MGRead(0);
  float lux = lightMeter.readLightLevel(); //read from light sensor
```

This code sets the microphone input pin and starts serial connection at 9600 baud. It also activates DHT22-light sensor communication. This instructs the board on how to communicate with its sensors.

To prevent the serial buffer from filling too quickly, loop() begins with a two-second wait with a delay function. After that, it adjusts humidity, temperature, heat index, and light level using the right methods. Reading the CO sensor uses AnalogRead(1), and reading the CO2 sensor uses MGRead(). All data collected thus far is recorded in variables which ensures that all devices capture data simultaneously.

```
  // Check if any reads failed and exit early (to try again).
if (isnan(h) || isnan(t)) {
  Serial.println(F("Failed to read from DHT sensor!"));
  return;
}
 if (lux < 0) {
  Serial.println(F("Failed to read from light sensor!"));
  return;
}








if (SVCO2 <= 0 || SVCO2 > V_REF) {
  Serial.println(F("Failed to read valid CO2 sensor voltage!"));
  return;
}
int percentage;
percentage = MGGetPercentage(SVCO2, CO2Curve);
if (percentage == -1 || percentage < 400) {
percentage = 400;  // Clamp to minimum realistic value
}


float voltage2 = (SVCO / ADC_RES) * V_REF;  //Convert to voltage
if (voltage2 <= 0 || voltage2 > V_REF) {
  Serial.println(F("Failed to read valid CO sensor voltage!"));
  return;
}
float CO = (((voltage2 - V_CLEAN_AIR) / SENSITIVITY)); //voltage to ppm using datasheet formula
```

This part of the loop continues with validity checks to ensure that no sensor returned a faulty or unreadable value. It uses basic conditionals to catch failed readings from the DHT22 or BH1750 sensors (e.g., isnan() or lux < 0) and returns early with a warning message if any issues are detected.

For the CO_2 sensor (SEN0159), the system checks whether the voltage is within a valid range. If it passes, the sensor's analog voltage is run through a MGGetPercentage() function and clamped to a minimum realistic value to eliminate any possible noise. Similarly, for the **CO sensor (SEN0132)**, the raw analog value is converted to voltage, and then that voltage is translated into **ppm of CO** using a formula derived from the sensor's datasheet.

```
  Serial.print(h);
  Serial.print(F(" "));
  Serial.print(correctedTemp);
  Serial.print(F(" "));
  Serial.print(hic);
  Serial.print(F(" "));
  Serial.print(percentage);
  Serial.print(F(" "));
  Serial.print(CO);
  Serial.print(F(" "));
  Serial.println(lux);

}
```

```
float MGRead(int mg_pin) {
  int i;
  float v = 0;

  for (i = 0; i < READ_SAMPLE_TIMES; i++) {
    v += analogRead(mg_pin);
    delay(READ_SAMPLE_INTERVAL);
  }
  v = (v / READ_SAMPLE_TIMES) * 5 / 1024 ;
  return v;
}

int MGGetPercentage(float volts, float *pcurve) {
  volts = volts / DC_GAIN;
  if (volts > ZERO_POINT_VOLTAGE || volts < MAX_POINT_VOLTAGE ) {
    return-1;
  } else {
    return pow(10, (volts- pcurve[1]) / pcurve[2] + pcurve[0]);
    volts = 0; }
}
```

This section effectively translates raw sensor readings into clean, scientifically meaningful values, with built-in checks to catch and skip faulty data to maintain system reliability.

After the sensor values are checked and processed, they are sent out one at a time using Serial.print() in a structured way. There are several numbers sent to the serial port that can be read by a PC that is attached. After the loop, two custom functions are written to deal with CO_2:

- MGRead(int mg_pin) takes analogue readings from the CO2 sensor, averages them to make things more stable, and then turns the raw analogue data into voltage.
- MGGetPercentage(float volts, float* pcurve) uses the voltage and a logarithmic conversion code based on the sensor's datasheet curve to figure out the CO2 concentration in parts per million (ppm). It also has range checks to make sure that the result is still valid.

This part finishes the data collection loop by making sure that every sensor value is handled and sent out in a clear, correct way so that it can be interpreted later. The Arduino Mega has now finished its job in the system; it is now a strong sensor data pipe that leads to the small PC.

For Arduino Uno (COM3) [40]

The Arduino Uno that is linked to COM4 is only being used to check the PMS5003 sensor. This choice was made because the PMS5003 was so unstable when attached to other sensors on the Mega board. This was probably due to timing issues in serial communication caused by other sensors stopping read processes. The method makes sure that readings of PM1.0, PM2.5, and PM10 values are always correct.

The Uno only uses 17% of active memory and 12% of program storage space, so it is a light and efficient choice for this one-task use. The code is kept simple to keep communication and extra costs low. This makes sure that this important air quality sensor always works correctly and doesn't mess up the hardware network as a whole.

```
#include <SoftwareSerial.h>

SoftwareSerial pmsSerial(2, 3);  // RX, TX

void setup() {
    Serial.begin(115200);
    pmsSerial.begin(9600);
    delay(1000);
    wakeUpPMS5003();  // Ensure the sensor is awake
    sendActiveModeCommand();  // Force Active Mode
}

void loop() {
    if (pmsSerial.available() >= 32 && pmsSerial.read() == 0x42 && pmsSerial.read() == 0x4D) {
        uint8_t buffer[30];
        for (int i = 0; i < 30; i++) buffer[i] = pmsSerial.read();

        Serial.print((buffer[8] << 8) | buffer[9]);
        Serial.print(F(" "));
        Serial.print((buffer[10] << 8) | buffer[11]);
        Serial.print(F(" "));
        Serial.println((buffer[12] << 8) | buffer[13]);
    }
    delay(1000);
}

/* Wake up PMS5003 Sensor */
void wakeUpPMS5003() {
    byte wakeCommand[] = {0x42, 0x4D, 0xE4, 0x00, 0x01, 0x01, 0x74};
    pmsSerial.write(wakeCommand, 7);
    Serial.println("Wake-up command sent.");
}

/* Set PMS5003 to Active Mode */
void sendActiveModeCommand() {
    byte command[] = {0x42, 0x4D, 0xE1, 0x00, 0x01, 0x01, 0x71};
    pmsSerial.write(command, 7);
    Serial.println("Active mode command sent.");
}
```

The code begins by including the SoftwareSerial library. After, new serial connection is created on pins 2 (RX) and 3 (TX) for communicating with the PMS5003 sensor. In the setup() function, both serial interfaces are initialized; 115200 baud for PC, and 9600 baud for the PMS5003. After a short delay, the code sends two commands to the sensor using custom functions from data sheet:

- wakeUPPMS5003() – ensures the sensor is active
- sendActiveModeCommand() – sets the sensor to continuous measurement mode

In the loop(), the program waits for **32 bytes of data**, then checks for a valid PMS5003 frame header (0x42 0x4D). If correct, it stores the incoming bytes in a buffer and prints the relevant indexes:

- buffer[8] & buffer[9] for PM1.0
- buffer[10] & buffer[11] for PM2.5
- buffer[12] & buffer[13] for PM10

These values are extracted in raw form (using bit shifting) and sent to the PC. The frame structure and byte positions are taken directly from the sensor datasheet to ensure proper alignment with PMS5003's communication. The entire code is optimized for simplicity and reliability, focusing only on this one sensor. With dedicated bandwidth, the Uno handles PMS5003 readings consistently and without delays.

Filtering Noisy Data and Data Smoothing

To handle data noise, missing values, and fluctuations, this system uses TouchDesigner’s node-based environment, which enables real-time data manipulation and visual processing. TouchDesigner provides a powerful set of built-in tools for smoothing sensor outputs, applying moving averages, and managing irregular or missing data; all without the need for complex coding.

The system creates a clear and modular pipeline, where each transformation (e.g., smoothing, clamping, thresholding) is visually traceable and easily adjustable. This visual approach not only simplifies tuning and calibration but also ensures that noisy or unstable readings are stabilized before reaching the user interface.

In the first step of the normalization process, a 10-sample moving average is calculated using the trail and analyse nodes. By doing this, momentary variations in the data are removed, making it more consistent for both analytical and visual display.

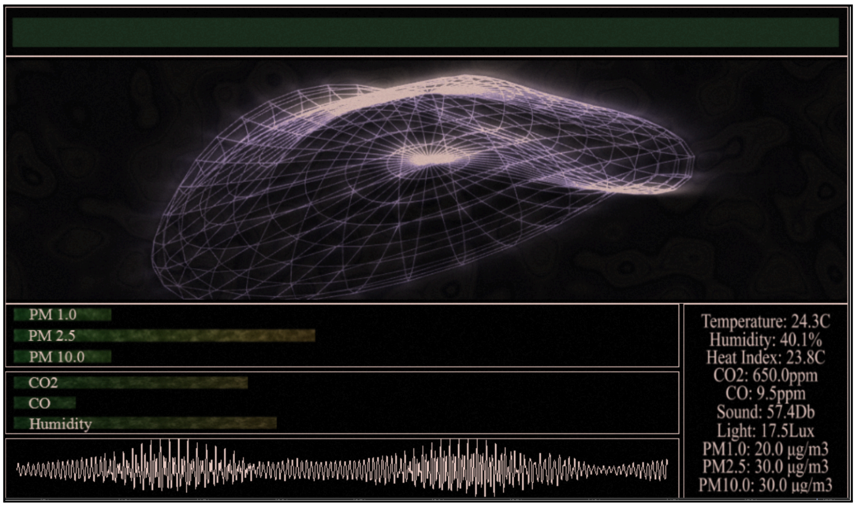
Following, the data is normalised to guarantee that it is shown in a logical and significant way. This is accomplished by a specified constant minimum and maximum range. If Temporary adjustments in the minimum and maximum values are done depending on trail values, it can cause the results to differ since they are not recalculated in real time. Therefore, These constant levels were chosen in order to strike a balance between accepted scientific norms and useful applicability in real-world situations. Different upper (hazardous) and lower (non-reactive) limits were set for each sensor based on the analysis done in the threshold section, as shown in Figure X10.

With math and logic nodes, any number below the predetermined minimum threshold is shown as a percentage of 0%, while values over the maximum threshold are given in a percentage of 100%. This helps the system to minimise misunderstanding caused by excessive or uncommon data points by giving consumers a clear and understandable visual reaction.

The microphone input, lastly, can instantly record the sound level thanks to the direct USB interface with Windows and TouchDesigner, completely avoiding the Arduino. The signal is smoothed using a moving average, and power is converted to decibels using the standard 20Log(X) method for better clarity.

Quick and simple tweaking is made possible by this node-based method, which also makes it possible to add new features later on without having to completely redo the logic structure.

3.3 GUI Development in TouchDesigner



Just like normalization and data handling, TouchDesigner is also used to make the device's graphical user interface (GUI) where it has two levels of input to fit the needs of different types of users:

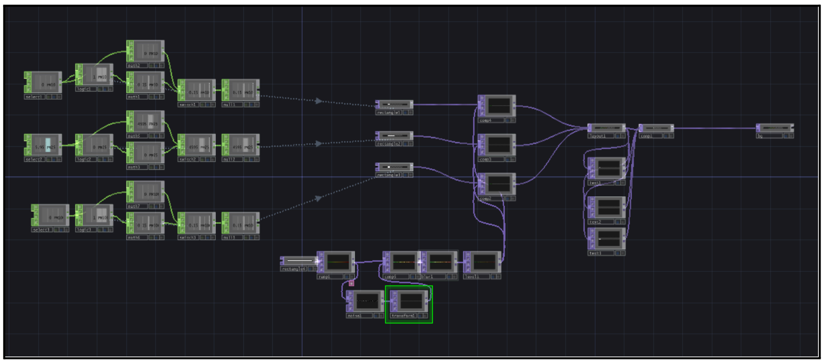
- **Detailed Mode (Scientific Panel):** Panel of Researchers (Comprehensive Mode): This part shows accurate sensor data, like temperature values, PM2.5 levels in µg/m³, and CO_2 concentrations in ppm, along with the bar signs that go with them. This item is meant for people who want to know a lot about their surroundings. This mode helps users make smart, proactive choices by making it clear which sensor values are higher and how much they vary from normal levels.
- **Reactive Mode (Visual Feedback):** This layer turns data into symptoms that can be seen on a centre 3D object when the Reactive Mode is on (Visual Feedback). When the surroundings changes, the item changes its colour, intensity, or movement based on data from sensors. For instance, it might show that colours fade at lower temperatures or that brightness changes based on how much light is present. For customers who would rather not be involved as much, this keeps them from being overwhelmed with numbers and gives them a quick, easy-to-understand answer.

The two-sided design strikes a good mix between easy-to-use features and informing content, meeting the needs of regular users while still giving users who need more in-depth information what they need. A third part, called "alert states," only uses big, noticeable signs to let people know about extreme and urgent conditions.

Detailed Mode

The part of the system screen with the most information is this detailed mode. Real-time sensor readings are shown in scientific measures like ppm, °C, µg/m^3, dB, Lux, and %RH. This gives users a clear picture of their current home environment. This mode is great for people who want to get a clear picture of the air quality, comfort, and safety levels.

Along with basic numbers, six important environmental factors are shown on a horizontal bar graph: humidity, CO_2, CO, PM1.0, PM2.5, and PM10. Each bar is a different colour, ranging from green to red, which shows how bad the situation is or how close it is to being dangerous based on set levels.



This is the full network system behind the detailed mode's bar graph visualization in TouchDesigner. On the left-hand side, each of the six key environmental features has its own processing pipeline. These pipelines use a combination of nodes to convert percentages into graphical widths which also implements clamping to minimum. The right-hand section merges the processed data streams and sends them into the final visual output chain. This is done via building each bar using rectangle and ramp nodes, where the rectangle length is driven by the sensor's percentage value, and the color shifts from green to red based on danger. Each bar is layered with a label showing the feature name, creating a clean, readable, and responsive visual indicator for each sensor.

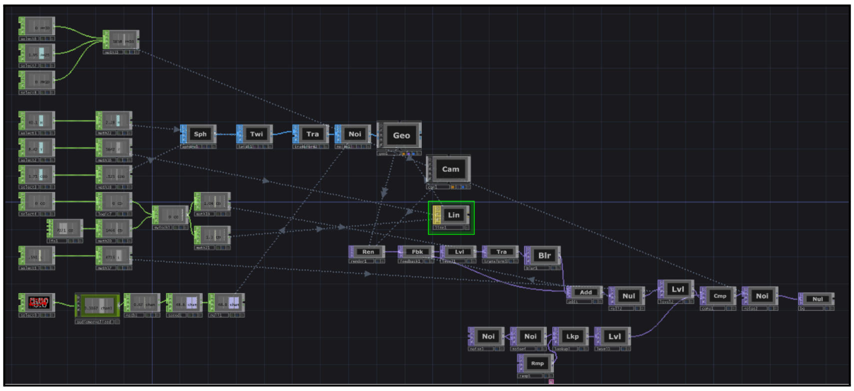
This audio trail system is minimal but powerful. It captures 20,000 sound samples, converts them into a series of visual lines, and instances them frame by frame, effectively plotting the real-time waveform of the environment's sound levels.

Sensor	Related Visual Effect	How the Effect Varies	Justification
Temperature	Color (Gradient)	Low - Blue Normal - Soft White High - Red	Physiologically, red is linked to heat while blue to cold - intuitive
Sound	Vibration	Low - Still Normal - Minimal High - Shaking	Visualizing physical sound vibration makes noise understandable while links to motion
Light	Brightness	Low - Dim Normal - Neutral High - Glowy	Mimics how human eye adapts to light
Particulate Matter	Texture Grain	Low - Smooth Normal - Light Static High - Heavy distortion	Converts invisible pollution to visible digital perception
Humidity	Bending	Low - Straight Normal - Arc High - Melted	Humidity is linked to the melting motion, like bending for realism
CO	Flickering Alert	Low - N/A Normal - Slow High - Intense	CO_2 impacts breathing therefore linking it with the disk struggling to "breathe".
CO2	Vertical Size	Low - Deflation Normal - Flat High - Inflation	CO is silent and deadly. urgent red flashes mimic danger signals

Real-Time Mode

The Real-Time Mode offers an immediate, effortless glance at environmental conditions by translating sensor data into data-driven visual effects on a 3D object. Instead of displaying numbers, this section uses intuitive graphical cues, such as color shifts or motion changes to reflect whether conditions are normal or drifting toward harmful levels. Each visual effect is deliberately chosen to be psychologically relatable. For example, a warm color for high CO_2 levels or pulsating light for rising sound. This allows users to understand the environment within milliseconds, without needing to read any values.

For deeper understanding, a written appendix is also shown on the back side of the visualization box which has the information from each visual effect's meaning to information about calibrating the device after power on. Below is a table to show each of the factor's related effect on the 3D object, with its given justification:



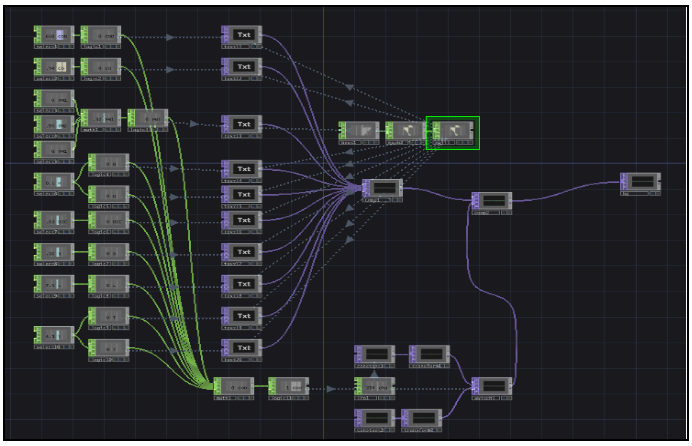
One of the two main elements that create this sub-network is the rendering pipeline, which generates and animates the primary object based on live sensor data. On the other hand, the other part is the after-effect logic, which adds distortion or visual warnings as responses to abnormal levels. The above screenshot provides a breakdown of the specifics of each of these stages. Left hand side green and blue nodes are where all sensor inputs affect the core rendering. The object's structure, color, or size is modified based on environmental values. This gives the user a real-time view. The rest, the purple nodes dealing with post-processing enhancements like glow, blur, or noise are layered based on thresholds. This emphasizes abnormal readings in a visually engaging way, reinforcing warnings.

3.4 Alert System & Notifications

In addition to visual feedback and detailed mappings, the system has a dynamic alert and notification mechanism that gives users alerts of any significant changes in the interior environment. When environmental factors exceed pre-defined safe or healthy limits, these alarms serve clear warnings. Instead of relying solely on unprocessed data or subtle visual cues, this layer gives priority to instant and focused attention on the most important parameters.

There are upper and lower bounds for each monitored feature. When a value is higher than a safe range, a visual alert is triggered inside the interface. In order to enable timely and well-informed response, the system not only detects a problem but also identifies the problematic characteristic . The threshold for each factor and the associated message are shown in the following table:

Condition	Message
CO (Carbon Monoxide) > 50 ppm	Dangerous CO! Ventilate immediately!
CO_2 > 1000 ppm	Stale air! Open windows!
PM > 80%	Air quality poor! Use purifier!
Heat Index > 103°F (39.4°C)	Extreme heat! Hydrate & cool!
Humidity < 25% Humidity > 70%	Dry air! Use humidifier! High humidity! Dehumidify now!
Noise > 85 dB	Too loud! Use ear protection!
Light too bright (>1000 lux)	Reduce brightness! Avoid glare!
Too Hot (>35°C) Too Cold (<15°C)	Reduce Temperature! Too Hot! İncreaseTemperature! Too Cold!



Each incoming sensor value is compared against predefined safe thresholds using logic nodes. When a value falls outside the acceptable range, its corresponding logic node is triggered, activating a warning message. These messages are displayed clearly for the user, indicating which variable is in alert. All logic values are also summed together, and if any of them indicate danger (making the summation not 0), the background changes color and becomes blinking red from pure green. This ensures the system gives both individual alerts and a general environment health status at a glance.

3.5 Debugging and TouchDesigner Enhancements

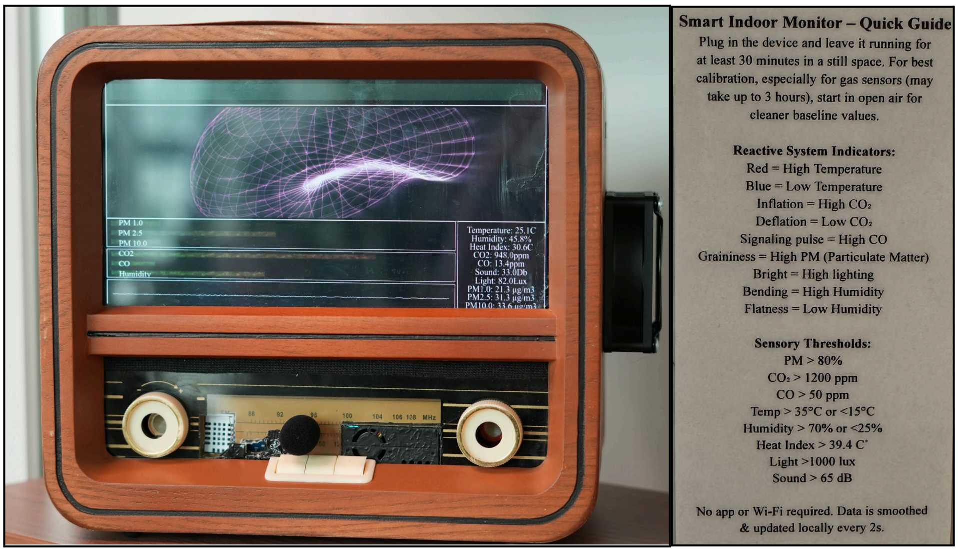
The development process was made a lot harder when multiple sensor sources were added to an Arduino setup at the same time. Due to problems being found with the PMS5003 particulate matter which was probably its communication being interrupted, likely due to the blocking nature of some sensors or timing conflicts. To fix this problem, the PMS5003 was linked to an extra Arduino Uno through a different COM port, thus, none of the apps will have any problems running at the same time. In addition the power source of the CO2 sensor was a hustle, since it was the one that needs the most power and consistent high current. This problem was also resolved by being attached to extra arduino uno for power while still using arduino mega for communication

On the touchdesigner side, Simple structures were used to make complex systems. The node nature allows easier improvements over time and maintenance as time goes on. The raw sensor data could not be used directly because there was too much noise. So, trail and analysis nodes were added to improve the visualisations . After that, the mathematical and logical nodes indicated dangers that were represented by numbers. This let users see their data in real time through the interface's reactive patterns. To make it easier for the node network to grow, it has been split into three modes: detailed mode, real-time mode, and alert systems. Because of this, growth and scalability becomes possible where displays can be changed or made bigger in the future. Now that these changes have been made, the system has gone from being a good test to a well-designed tool that anyone can use.

4. Production & Testing

4.2 Production and Assembly

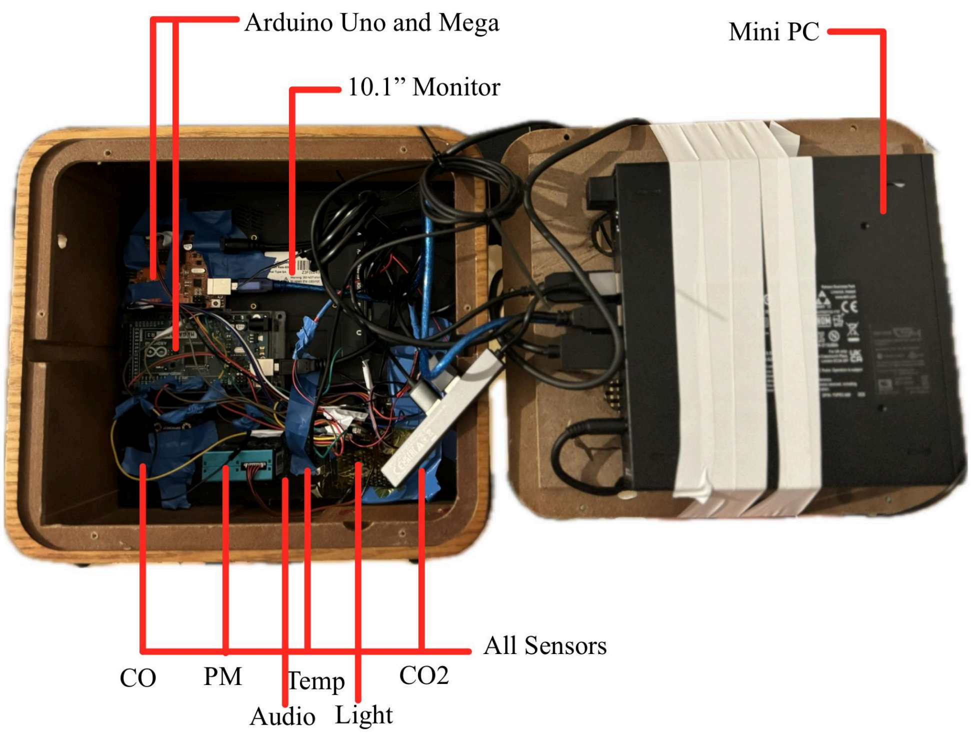
The whole system is housed in a 25×25×18 cm wooden old radio box that was chosen for its good looks and portability initially. This choice fits with the nostalgic draw of analogue devices and keeps the project's user-friendly shape, making it perfect for use in the home. The outside shell, which includes the buttons, dials, frequency markers, and quick use guide, has been preserved in its original form, making it look like an old radio.



The structure inside is set up very precisely. The original speaker compartment was a rectangle opening, which was changed with a 10.1" display that fits perfectly. The vintage interface of knobs and slides was carefully drilled just below the screen to make room for weather sensors without messing up the retro look. Some sensors can be seen barely, while others are hidden behind the panel and read data through holes or gaps.

The system comes in a box with a small PC, two Arduino boards (Mega and Uno), five environment sensors, a microphone, a USB hub, and cables. Two cooling fans and thermal insulation support keeping inside and DHT22 sensor cool. The small PC that runs the main software is placed on the back wall so that it faces inward. It is held in place with hot silicon, tape, and glue to make sure it stays in place. The two Arduinos are put behind the monitor so that they are close to the sensors.

Initially, the internal heat that will be produced by all the components inside was nat considered a big ussie. However in the first tests, the internal heat became so much that the DHT22 temperature sensor started giving way off readings. Therefore, additional two cooling fans were positioned on each side wall, effectively creating a ventilation tunnel to address early-stage overheating, which reached up to 80°C without airflow. After fan installation, temperature stabilized at a safe operational range.



Zip ties keep all the wires in order so they don't get tangled up or cause confusion. The gadget only has one power cord, which makes it easier to use and more convenient. There is a small button on the side of the system that can be used to turn it on or off without any other interaction.

The system is designed to be open for improvements, even though it is small. Using screws, the back panel can be taken off easily, giving full access to the internal parts for repairs, tuning, or possible hardware improvements. Instead of just a microcontroller, a small PC is built in. This makes the device very flexible for real-time calibration using external datasets and cloud processing processes, which is something that most embedded systems can't do on their own. The dual-Arduino architecture adds safety and handles the communication loads from the sensors.

4.2 Testing & Calibration

Before integration into the final system, each sensor was already individually tested with a benchmark and calibrated on a breadboard setup using basic sketches and correction codes provided by the manufacturer libraries **(Section 3.1 Hardware Architecture)**. The first tests showed that the sensors worked within the accuracy ranges listed on their datasheets and for those which weren't, appropriate correction codes were implemented. For example, the DHT22 sensor worked within ±2% of the reading. Therefore, there wasn't any need to perform additional tuning. Instead, the focus was on making sure that the whole network of sensors, microcontrollers, and software worked together correctly.

This project utilizes both analogue and digital devices, two different microcontrollers, and a PC-based data visualisation layer with a connection to a monitor. The most important thing to test was how well these parts would work together in the real world.

The following tests were selected based on:

- Their relevance to actual home or indoor conditions (CO_2 buildup in small rooms, background noise)
- The ability to validate system's general **responsiveness**, **accuracy**, and **resilience**
- Feedback from **potential users**, emphasizing the importance of **clarity** and **trust in data**

Summary of Conducted Tests:

Test ID	Test Name	Purpose	Method	Expected Results
T1	System Cohesion Test	Ensure consistent performance of all sensors and components	Full system ran for 3 hours. Observed for data freezing or crashes.	All sensors remained active and consistent. No lag or failure.
T2	Extrememium Scenario Test	Simulate buildup of particulate matter in an enclosed space	Device placed in closed room for 20 mins with a lighting candle. Noted values before, during, and	PM increased from to ~500 ppm, dropped back to normal ppm post-ventilation.
T3	Sensitivity Test	Evaluate audio reactivity	Room observed during silence, podcast, conversation, claps.	Real-time fluctuations tracked clearly. Claps triggered rapid spikes.
T4	User Feedback Evaluation	Test ease of use and understanding	4 people viewed and interpreted visual data. Asked for confusion points and improvement ideas.	3/4 found it clear. 1 suggested labeling light and CO values better.
T5	Accuracy Verification	To further guarantee sensor correctness with the full system even though each sensor's benchmarking and calibration was done before	Compare temperature, CO_2, humidity, and PM readings to reference meters or expected indoor values.	Sensor readings fall within ±5–10% of reference values.
T6	Portability & Room Change	To ensure the system works consistently in different locations.	Move device between rooms (bedroom, kitchen, hall). Watch if data adapts quickly and remains reliable.	Sensor values adapt to each room's environment within ~10 seconds.
T7	Responsiveness (Reaction Speed)	To evaluate how quickly the system reflects environmental changes.	Turn lights on/off, blow on sensors, open windows. Measure delay until value change is seen on-screen.	Light/audio: <1s; others up to 10 seconds due to moving average calculations
T8	System Load on Mini PC	To measure how heavy the software is on the mini PC running TouchDesigner.	Use Task Manager to monitor CPU and memory use while system is live.	CPU usage under 40% during full use; no overheating or lags.
T9	Data Smoothing Test	To check if noise and jitter in sensor data is properly filtered for readable output.	Leave system idle in a still environment. Observe if values flicker or stay steady.	Data remains stable due to smoothing filters; no heavy jitter or flickering.

5. Results & Analysis

The collected data was analyzed in terms of accuracy, responsiveness, system performance, and user interaction quality. While sensors were already pre-tested and known to deliver calibrated outputs, these results should verify that the system is usable, portable, and dependable under actual usage scenarios.

R1 – System Stability & Integration (T1)

For more than three hours, the system operated without any freezing, sensor disconnections, or PC-side slowness. Without any delays or conflicts, the Arduino Mega and Uno transmitted sensor data, and TouchDesigner updated instantly. Additionally, no heat is accumulating within the gadget thanks to two fans that were installed subsequently. This confirms stable integration and validates the coherence of the entire hardware-software stack.

R2 – PPM Detection & Air Quality Responsiveness (T2)

Poor ventilation was clearly detected in the enclosed room test, as µg/m3 values increased from around 6 ppm to over 900 ppm in less than 20 seconds. Levels plummeted shortly after the door opened. This allows use in HVAC or ventilation study and validates the sensor's response in the actual world. When the same test is applied to other sensors, similar results can be seen with up to 2 meters of detection range for small emitters. However, the detection range is not negligible in this system since it is assumed that a given indoor space is homogen in air quality.

R3 – Audio Sensor Reactivity (T3)

The microphone successfully detected different types of sounds such as silence, speech, and clapping with <0.5s latency. Peak values rose sharply with sudden noise, confirming excellent real-time responsiveness. No false triggering or sensitivity issues were noted.

R4 – User Interface Usability (T4)

4 participants were asked to use and interpret the system for 5 minutes:

- 3 were able to understand the visuals (color-coded values, bar graphs, etc.) with no prior training.
- 1 suggested clearer labels and color coding for light intensity and sound indicators
- 3/4 could understand particulate matter labellings.
- 1 needed assistance interpreting sound data.
- 2 requested a clearer visual distinction between safe and risky ranges (e.g., color bands or emojis).

The feedback will help fine-tune future interface upgrades, proving that even non-experts can understand the system after brief usage.

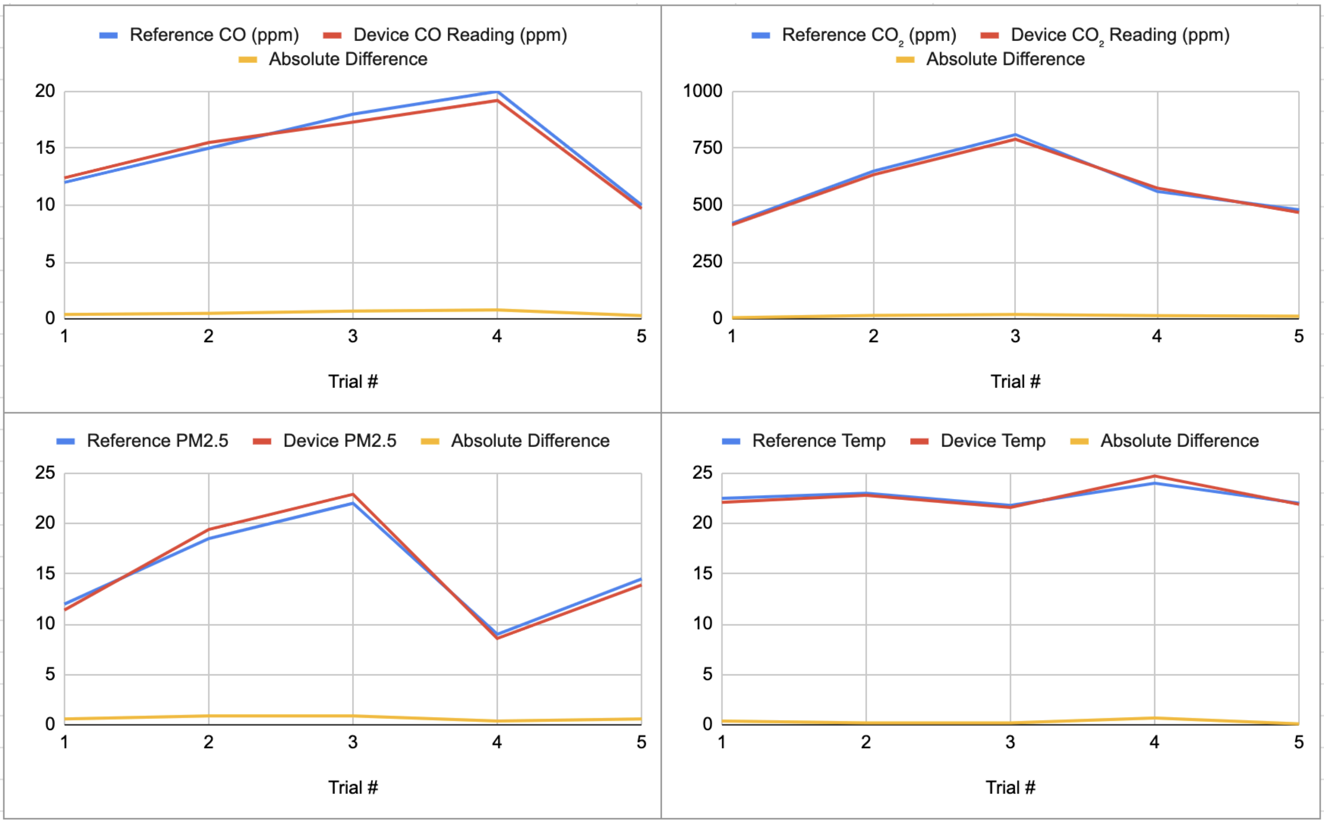
R5 – Sensor Accuracy Against Reference Devices (T5)

After the implementation of correction codes to the sensors using benchmark values, each sensor became calibrated and stable. However some fluctuations and errors may still occur after the full system assemble. Thus the accuracy of the sensors are re-tested using specific methods and benchmarks. Such features like PM2.5 values were compared to readings from an open-air government station (approximated).

The deviations were calculated using:

$$Error\ (\%) = \frac{|Reference - Measured|}{Reference} \cdot 100$$

Across multiple readings, the results was:



When the graphs are observed, it can be seen that the absolute deviation is as close as 0. Therefore, these results are within acceptable tolerances, confirming the system's reliable environmental sensing. All readings are within acceptable ranges for residential monitoring.

R6 – Portability Across Rooms (T6)

The system was moved to three rooms (bedroom, kitchen, hallway). Sensor values have adapted correctly to each room's environment and remained consistent.

Based on whether the gas sensors (CO_2 and CO) have cooled down, the system takes either 1 min or 30 mins to stabilize the readings. Since the gas sensors can only read well when they are heated up (which takes time) cooled-down sensors take much more time to re-stabilize. Even Though, the system can be said to be demonstrating practical portability.

R7 – Environmental Change Responsiveness (T7)

In the section 3.2, subsection "Filtering Noisy Data and Data Smoothing", it is dictated that moving averages are used for better data-pipeline. Since the moving average system implemented uses the last 10 samples, the response to any sudden spike or change can take 2 to 10 seconds based on the spike's duration and amplitude. However, some features like sound Db and the voice trail and the effects on the object are immediate, since no handling is done for the audio data that can cause a delay. Even though, for daily use cases in most of the scenarios, the average system latency is sufficiently fast.

R8 – CPU Load and Mini PC Performance (T8)

Using Windows Task Manager, the CPU usage while TouchDesigner ran all visuals and logic:

- Idle: 12–18% CPU
- Full visualization load: 30–35% CPU

No frame drops or overheating occurred. The PC handled continuous operation smoothly with plenty of headroom.

R9 – Data Smoothing & Visual Clarity (T9)

Raw values were piped through moving average filters and real-time lag compensation nodes in TouchDesigner. Therefore, the output data showed no jitter or sudden spikes during static conditions. This filtering made the interface pleasant to read and accurate in capturing meaningful changes, while ignoring noise from minor fluctuations.

6. Technical Comparison to Existing Commercial Devices

To understand the positioning and performance of this system in a better way, a comparison was made between the SEM and two leading commercial indoor environmental monitors: IQAir's AirVisual Pro and Awair Element. This comparison highlights how the system solves key limitations of existing solutions, particularly in terms of sensor variety, real-time customization, and offline operability. Unlike closed, cloud-reliant devices that primarily focus on air quality alone, this system offers a more holistic approach to indoor environment monitoring by including additional factors such as sound and light, both of which impact comfort and mental well-being. Its TouchDesigner-based visualization engine allows for dynamic, and fully customizable feedback without the need for external connectivity or fixed dashboards. Furthermore, the device's open-source, modular architecture and reliance on affordable, widely available hardware make it both scalable and suitable for experimental, educational, and even artistic or psychological applications—filling a unique niche in the current market landscape.

1. Sensor Coverage and Data Breadth

Device	CO_2	PM (1.0 / 2.5 / 10)	Temperature	Humidity	CO	Light and Sound
This Project	✅	✅	✅	✅	✅	✅
IQAir AirVisual Pro	✅	✅ (PM2.5 only)	✅	✅	❌	❌
Awair Element	✅	✅ (PM2.5 only)	✅	✅	❌	❌

2. Real-Time Data Handling and Visualization

Device	Real-Time Feedback	Custom Visualization	On-Device Processing
This Project	✅ (Live updates)	✅ (TouchDesigner GUI)	✅ (via embedded mini PC)
IQAir AirVisual Pro	✅ (Mobile/Web)	❌ (Fixed dashboard)	❌ (Cloud-based)
Awair Element	✅ (App feedback)	❌ (Limited control)	❌ (Cloud-based)

5. Target Use Cases

Use Case	This Project	AirVisual Pro	Awair Element
Real-time health feedback in events	✅	❌	❌
Educational use & customization	✅	❌	❌
Home wellness & alerting	✅	✅	✅
Factory/lab air monitoring	✅	✅	❌
Mood/comfort-based visual installations	✅	❌	❌

7. Risk Management, Ethics & GDPR

7.1 Risk Matrix

Risk	Impact	Likelihood	Severity	Mitigation Strategy
Sensor failure (hardware defect)	High	Medium	High	Use proven, widely-tested components; purchase backups; test sensors in isolation before integration.
Inaccurate sensor data / drift	Medium	High	High	Calibrate each sensor during testing phase; allow periodic recalibration; compare to public datasets for correction.
Overheating due to enclosed design	High	Medium	High	Ventilation tunnel added with dual side-mounted fans; thermal insulation materials used; components monitored during testing.
Arduino or PC communication	Medium	Medium	Medium	Divide sensor duties between two Arduinos to reduce serial conflict; test serial outputs continuously; include restart routines if needed.
GUI lag / failure in real-time feedback	Medium	Low	Medium	Use a powerful mini-PC; TouchDesigner file is optimized for low resource usage.
Power loss or unplugging	Medium	Medium	Medium	Only one cable is required; system reboots safely without long-term logging loss.
Incorrect threshold setup	Medium	Medium	High	Thresholds reviewed using reliable literature; easily changeable in code/TouchDesigner for future refinements.
Data misinterpretation by	Low	Medium	Medium	Real-time mode uses simple visual feedback; detailed view includes units and benchmark guidance.
Internal cable loosening /	Medium	Low	Medium	Zip ties used for cable management; stable mounting with hot glue, tape, and silicon; easy access via back panel.
Software bug / crash	Medium	Medium	High	TouchDesigner and Arduino scripts developed incrementally with frequent live tests; test-driven approach adopted.

7.2 Ethical Considerations

One of the most important ethical strengths of this project is the complete absence of personal data collection. The system only reads environmental parameters and does not track, identify, or log any user behavior or identity. No cameras, biometric tools, or login systems are used in any part of the system. This removes potential risks involving surveillance, privacy violations, or psychological profiling that are often associated with modern sensor-based devices. The feedback generated by the system is general and environment-focused, ensuring that it can be safely used in homes, public spaces, or shared environments without any ethical concern tied to personal data tracking. Even though there is a 10 sample tracking feature in order to get the moving average of the data, it is only recorded and deleted inside that node due to the nature of touchdesigner. Additionally, the device runs 100% local with no connection to outside.

On a more extended level, SDG Target 3.9 is to significantly reduce fatalities and illnesses from hazardous chemicals and air, water, and soil pollution and contamination by 2030 [14]. Although Indoor air pollution is overlooked, VOCs, CO, CO_2, PM2.5 & PM10, and biological pollutants can aggravate asthma, lung cancer, and cardiovascular disease. Therefore, IAQ enhancement is essential to disease prevention and public health promotion, supporting SDG 3's preventative healthcare approach. Advanced monitoring, ventilation, and air purification can detect indoor air pollutants early, lowering healthcare expenses for indoor pollution problems. Excellent air quality in schools, workplaces, hospitals, and households. Large-scale adoption of such monitoring systems can influence policy-making and urban planning to prioritise health-focused building designs, boosting global sustainable development and public health.

7.3 GDPR Compliance

GDPR cannot be violated by this project since it does not collect or use any personal data. The tiny PC inside the gadget processes all the sensor data locally. Not a single third-party APIs, or cloud-based services are used to store, track, or analyse data. When the info is ready to be used, only the GUI does so. No logs for the past are saved. This is fully in line with GDPR because it doesn't deal with personally identifiable information (PII) and all data stays on the local device without the user's permission or GDPR rights.

7.4 Potential Societal Impact

The versatility and modular nature of this system open doors to a wide range of societal applications beyond traditional indoor air monitoring. Its visual layer can be redesigned and recontextualized for diverse environments which can vary from clubs, to calm and focused interfaces for libraries, study halls, or healthcare spaces. With the recent trend of decentralized health management, the help this device can do can be a life saving one in extreme cases. Its scalability; from personal desks and bedrooms to offices, classrooms, and public spaces, means it has the potential to bridge the gap between environmental health and real-time human-centered experience, promoting well-being through smarter, more interactive environments.

7.5 Trade Offs and Benefit Analysis

Although the system is created to be both mobile and functional, some trade offs were made throughout the engineering process to optimize the device in needed regards. Main trade off is between data stability and response delay. Due to the systems data handling process which is done via moving averages, increasing the sample size increases smoothness but comes with an extra delay.

Another trade off was during the optimization of airflow and preserving aesthetic space in the original radio box. One of the cooling fans for the system were placed outside for better emission of the air , however this greatly diminishes its aesthetic appeal. This gadget is helpful and adaptable for future advancements since performance, modularity, and user control were carefully balanced throughout the design process.

8. Conclusion & Future Work

In this project, the concept and engineering of a smart environmental monitor system that is both user-centric and accurate is shown in depth. The device increases awareness of the users via an interface about the environment's health conditions. The visuals, which are driven by a lot of data including CO2, PM2.5, temperature, ... is fully customisable and scalable due to the embedded mini PC that runs touchdesigner continuously. This enables a range of applications, from immersive public projects to health focused applications in any space. By encouraging people to take tiny but meaningful steps towards improved indoor well-being, this approach not only improves indoor air quality monitoring but also fosters awareness and interaction.

8.1 Challenges & Limitations

Throughout the development process, several challenges were encountered. Certain sensors, such as the PMS5003 for particulate matter and the SEN0132 for CO, exhibited sensitivity to electrical noise, necessitating the use of separate microcontrollers to ensure reliable communication. Although these sensors come factory-calibrated, there remains the potential for sensor drift over time, which could introduce inaccuracies without periodic recalibration or validation against known standards. Additionally, while the mini PC managed data processing tasks effectively, implementing smoothing filters like moving averages introduced a slight delay—approximately 10 seconds—in data responsiveness. Design considerations also had to accommodate physical constraints; for instance, incorporating external fans to enhance airflow led to minor compromises in the device's form factor. Furthermore, the current system is optimized for single-room monitoring and does not yet support multi-room or wireless communication, limiting its applicability in larger or more complex environments.

8.2 Future Enhancements

There are a number of ways to improve the system's functionality in the future. mainly, by incorporating artificial intelligence-based predictive models, the system may be able to learn from past sensor data and predict trends in the indoor environment. For example, it may be able to predict when the quality of the air will deteriorate while cooking or modify lighting and noise levels to improve sleeping conditions. The system's scalability and applicability for settings including homes, businesses, and schools would be improved by expanding it to enable multi-room monitoring through the installation of wireless sensor nodes. Creating a mobile application interface would also enable users to engage with the system outside of the physical display, view real-time data remotely, and get notifications.

By providing smart, responsive, and user-focused solutions for indoor environmental management, these developments would transform the system from a useful monitoring tool into a customized environmental assistant.

9. References

[1] A. P. Jones, “Indoor air quality and health,” *Atmospheric Environment*, vol. 33, no. 28, pp. 4535–4564, Dec. 1999, doi: https://doi.org/10.1016/s1352-2310(99)00272-1.
[2] “US EPA,” *US EPA*. https://epa.gov
[3] J. Shin, J. Y. Park, and J. Choi, “Long-term exposure to ambient air pollutants and mental health status: A nationwide population-based cross-sectional study,” *PLOS ONE*, vol. 13, no. 4, p. e0195607, Apr. 2018, doi: https://doi.org/10.1371/journal.pone.0195607.
[4] L. Shi *et al.*, “Long-term effects of PM2·5 on neurological disorders in the American Medicare population: a longitudinal cohort study,” *The Lancet Planetary Health*, vol. 0, no. 0, Oct. 2020, doi: https://doi.org/10.1016/S2542-5196(20)30227-8.
[5] A. Khan *et al.*, “Environmental pollution is associated with increased risk of psychiatric disorders in the US and Denmark,” *PLOS Biology*, vol. 17, no. 8, p. e3000353, Aug. 2019, doi: https://doi.org/10.1371/journal.pbio.3000353.
[6] J. G. Cedeño Laurent *et al.*, “Associations between acute exposures to PM2.5 and carbon dioxide indoors and cognitive function in office workers: a multicountry longitudinal prospective observational study,” *Environmental Research Letters*, vol. 16, no. 9, p. 094047, Sep. 2021, doi: https://doi.org/10.1088/1748-9326/ac1bd8.
[7] R. J. Shaughnessy, U. Haverinen-Shaughnessy, A. Nevalainen, and D. Moschandreas, “A preliminary study on the association between ventilation rates in classrooms and student performance,” *Indoor Air*, vol. 16, no. 6, pp. 465–468, Dec. 2006, doi: https://doi.org/10.1111/j.1600-0668.2006.00440.x.
[8] “US EPA,” *US EPA*, 2021. https://19january2021snapshot.epa.gov. (accessed Apr. 10, 2025).
[9] M. Frontczak and P. Wargocki, “Literature survey on how different factors influence human comfort in indoor environments,” *Building and Environment*, vol. 46, no. 4, pp. 922–937, Apr. 2011, doi: https://doi.org/10.1016/j.buildenv.2010.10.021.
[10] P. Wolkoff, “Indoor air humidity, air quality, and health – An overview,” *International Journal of Hygiene and Environmental Health*, vol. 221, no. 3, pp. 376–390, Apr. 2018, doi: https://doi.org/10.1016/j.ijheh.2018.01.015.
[11] A. P. Jones, “Indoor air quality and health,” *Atmospheric Environment*, vol. 33, no. 28, pp. 4535–4564, Dec. 1999, doi: https://doi.org/10.1016/s1352-2310(99)00272-1.
[12] A. P. Jones, “Indoor air quality and health,” *Atmospheric Environment*, vol. 33, no. 28, pp. 4535–4564, Dec. 1999, doi: https://doi.org/10.1016/s1352-2310(99)00272-1.
[13] U.S. EPA. Air Quality Criteria for Carbon Monoxide (Final Report, 1991). U.S. Environmental Protection Agency, Washington, DC, EPA/600/8-90/045F, 1991.
[14] World, “Household air pollution,” *Who.int*, Oct. 16, 2024. https://www.who.int/news-room/fact-sheets/detail/household-air-pollution-and-health?utm_source=chatgpt.com
[15] I. van Kamp and H. Davies, “Noise and health in vulnerable groups: A review,” *Noise and Health*, vol. 15, no. 64, p. 153, 2013, doi: https://doi.org/10.4103/1463-1741.112361.
[16] “Targets of Sustainable Development Goal 3,” *Who.int*, 2021. https://www.who.int/europe/about-us/our-work/sustainable-development-goals/targets-of-sustainable-development-goal-3?utm_source=chatgpt.com
[17] “IQAir AirVisual Pro Review - The Most Famous Air Quality Monitor for a Reason?,” *Breathe Safe Air*, Mar. 29, 2024. https://breathesafeair.com/airvisual-pro-review/
[18]“Evaluation-Summary.”https://www.aqmd.gov/docs/default-source/aq-spec/summary/iqair-airvisual-pro-v1-1683---summary-report.pdf?sfvrsn=18
[19] T. Salthammer, “TVOC - Revisited,” *Environment International*, vol. 167, p. 107440, Sep. 2022, doi: https://doi.org/10.1016/j.envint.2022.107440.
[20] https://www.facebook.com/airpurifierfirst, “Awair Element Air Quality Monitor - Hands-on Review (+Sensor Test),” *Air Purifier First*, Mar. 25, 2024. https://www.airpurifierfirst.com/review/awair-element/ (accessed Apr. 10, 2025).
[21] A. McNamara, “Something in the air: five indoor air quality monitors tested,” *BBC Science Focus Magazine*, Apr. 30, 2019.

https://www.sciencefocus.com/future-technology/something-in-the-air-four-indoor-air-quality-monitors-tested (accessed Apr. 10, 2025).
[22] E. G. Snyder *et al.*, “The Changing Paradigm of Air Pollution Monitoring,” *Environmental Science & Technology*, vol. 47, no. 20, pp. 11369–11377, Oct. 2013, doi: https://doi.org/10.1021/es4022602.
[23] A. P. Jones, “Indoor air quality and health,” *Atmospheric Environment*, vol. 33, no. 28, pp. 4535–4564, Dec. 1999, doi: https://doi.org/10.1016/s1352-2310(99)00272-1.
[24] K. R. Cromar *et al.*, “Air Pollution Monitoring for Health Research and Patient Care. An Official American Thoracic Society Workshop Report,” *Annals of the American Thoracic Society*, vol. 16, no. 10, pp. 1207–1214, Oct. 2019, doi: https://doi.org/10.1513/AnnalsATS.201906-477ST.
[25] “Digital Health - Worldwide | Statista Market Forecast,” *Statista*.
https://www.statista.com/outlook/hmo/digital-health/worldwide#revenue
[26] Public Health England, “Minimum home temperature thresholds for health in winter - A systematic literature review,” Oct. 2014. Available:
https://assets.publishing.service.gov.uk/media/5c5986f8ed915d045f3778a9/Min_temp_threshold_for_homes_in_winter.pdf
[27] “What are the regulations on indoor humidity?,” *www.condair.co.uk*.
https://www.condair.co.uk/industry-guidelines-and-regulations-on-indoor-humidity
[28] neil, “Understanding the Heat Index: What It Is, How It’s Calculated, and...,” *Crondall Weather*, Sep. 04, 2024.
https://crondallweather.co.uk/understanding-the-heat-index-what-it-is-how-its-calculated-and-its-importance-for-your-health/#google_vignette (accessed Apr. 10, 2025).
[29] “High CO2 Levels Indoors Will Surprise You,” *CO2 Meter*, Dec. 07, 2023.
https://www.co2meter.com/en-uk/blogs/news/high-carbon-dioxide-co2-levels-indoors
[30] “Carbon Monoxide Levels Chart,” *CO2 Meter*, Aug. 24, 2023.
https://www.co2meter.com/en-uk/blogs/news/carbon-monoxide-levels-chart
[31] D. Smith, “PM10: How Do Coarse Particles (Particulate Matter) Affect Air Quality?,” *learn.kaiterra.com*, Jan. 23, 2020. https://learn.kaiterra.com/en/air-academy/pm10-particulate-matter-pollutes-air-quality
[32] indoor air hygiene institute, “PM2.5 Explained,” *Indoor Air Hygiene Institute*, Apr. 06, 2021.
https://www.indoorairhygiene.org/pm2-5-explained/
[33] Jobaer Ahmed Saju, Quazi Hamidul Bari, Kazi, and Vladimir Strezov, “Measurement of ambient particulate matter (PM1.0, PM2.5 and PM10) in Khulna City of Bangladesh and their implications for human health,” *Environmental systems research*, vol. 12, no. 1, Dec. 2023, doi: https://doi.org/10.1186/s40068-023-00327-2.
[34] “Workplace noise : Tips for better office acoustics,” *BASWA*. https://www.baswa.com/en/workplace-noise/
[35] - Ledrise Led Professional, “Illuminating the Home: A Comprehensive Guide to Lighting and Recommended Light Levels,” *Ledrise*, Apr. 02, 2025.
https://www.ledrise.eu/blog/home-lighting-guide-lr/?srsltid=AfmBOopLPIz5Qi4JREtAls-m4aME50O0WVKlTTjL2i2GdGc9OxxG46oE (accessed Apr. 10, 2025).
- Engineering ToolBox, “Illuminance - Recommended Light Level,” *Engineeringtoolbox.com*, 2004.
https://www.engineeringtoolbox.com/light-level-rooms-d_708.html
[36] SEN0132 Official Data Sheet
[37] SEN0132 Official Data Sheet
[38] DHT22 Official Data Sheet
[39] BH1750 Official Data Sheet
[40] PMS5003 Official Data Sheet