ASHESI UNIVERSITY

DEVELOPING AND DEPLOYING AN IOT-BASED AIR QUALITY MONITORING DEVICE

CAPSTONE PROJECT

B.Sc. Computer Engineering

Isabel Prempeh Herraiz

2025

ASHESI UNIVERSITY

DEVELOPING AND DEPLOYING AN IOT-BASED AIR QUALITY MONITORING DEVICE

CAPSTONE PROJECT

Capstone Project submitted to the Department of Engineering, Ashesi University, in partial fulfilment of the requirements for the award of a Bachelor of Science Degree in Computer Engineering.

Isabel Prempeh Herraiz

May 2025

DECLARATION

I hereby declare that this Capstone Project is the result of my original work and that no part of it has been presented for another degree in this university or elsewhere.

Candidate's Signature: ………………………………………………………………

Candidate's Name: Isabel Prempeh Herraiz

Date: 28th April 2025

I hereby declare that the preparation and presentation of this Capstone were supervised in accordance with the guidelines on supervision of capstone laid down by Ashesi University.

Supervisor's Signature: …………………………………………………………………………

Supervisor's Name: Kofi Adu-Labi

Date: 28th April 2025

Acknowledgements

I want to express my deepest gratitude to my supervisor, Kofi Adu-Labi, for his constant encouragement and invaluable academic guidance. My heartfelt thanks go to my parents and friends' continuous encouragement, which have been instrumental in helping me achieve this milestone.

Abstract

Air quality significantly influences health, cognitive performance, and general wellbeing, yet monitoring remains limited in many educational institutions. This project addresses the lack of a sustainable, real-time air quality monitoring solution at Ashesi University by designing a solar-powered, IoT-based system. The developed system measures key pollutants such as PM2.5, CO_2, and VOCs, as well as environmental parameters like temperature and humidity. An ESP32-based microcontroller collects sensor data and transmits it wirelessly to a cloud database, while a mobile application visualizes real-time and historical readings for users. A Random Forest machine learning model was also trained and integrated to predict the Air

Quality Index (AQI) from collected sensor data, enhancing the system's proactive capabilities. Testing demonstrated high accuracy in data collection, successful cloud synchronization, and strong predictive performance with an R^2 value of 0.987. Comparison against a commercial air quality device confirmed reliable system operation, and solar integration ensured uninterrupted functionality. The project successfully delivers a low-cost, sustainable, and intelligent air quality monitoring solution, positioning Ashesi University as a leader in smart campus innovation and providing a model for broader adoption across educational institutions.

Tabe of Contents

Chapter 1: Introduction

1.1 Background

Although Ashesi has set up systems which are used to manage several aspects of campus life, air quality remains a small piece of the whole. Air pollution causes the premature death of 28,000 Ghanaians annually, based on information from World Health Organisation [1], thus underlining the need to consider air quality in educational facilities. Further, it has been discovered that air quality has substantial impact on health, cognitive performance, and productivity of individuals within academic institutions [3], and thus this project attempts to address that need.

Establishing a safe and green atmosphere at Ashesi University is necessary to promote academic excellence and better quality of life of students and staff on campus. Building an IoT based air quality monitoring device would enable both tracking and analysis of real-time air quality across Ashesi's campus. Such a system will allow Ashesi to make data-informed decisions to create a healthier atmosphere for the Ashesi community.

1.2 Problem Statement

The absence of an effective, sustainable air quality monitoring system at Ashesi University limits the identification of pollution hotspots and understanding how daily campus activities impact air quality. This gap calls for the development of a low-cost, sustainable, and user-friendly air quality monitoring solution.

1.3 Motivation

The primary motivations driving this project are:

1. **Supporting Ashesi's commitment to innovation and sustainability**: Ashesi University aims to push students to be more innovative; this project aligns with Ashesi's strategic goal to lead Africa into sustainable solutions design and operations.
2. **Enhancing health and well-being**: The majority of Ashesi students' time is spent on campus, where they attend classes, study in common areas, and go outside. They live and learn on campus. For the community's general health, wellbeing, and cognitive function, it is therefore essential to monitor and maintain good air quality throughout the campus.[3]
3. **Providing valuable data for research**: Research is a big part of Ashesi's culture. So, real-time air quality data can support interdisciplinary research, potentially leading to publications and collaborations with other institutions.
4. **Positioning Ashesi as an intelligent campus pioneer:** By implementing advanced technological solutions, Ashesi can serve as a model for other West African universities in environmental monitoring.

1.4 Project Objectives

These are the primary objectives of this project:

1. Develop a solar-powered, air quality monitoring device that will measure pollutants like PM2.5, CO2, VOCs and environmental factors such as temperature and humidity.
2. Place the device at a central point so that the data collected can represent the overall air quality.
3. Create a cloud-based real-time data transmission, storage, and analysis system.
4. Have a user-friendly interface for data visualisation, such as a mobile app.

1.5 Scope of the Project

This project includes designing, developing, and deploying the air quality monitoring device, featuring hardware and software components, data analysis tools, and initial data collection over a specified period. The data collected will be available to all Ashesi stakeholders and free to use however necessary.

This air quality monitoring project represents a significant step towards creating a healthier, more sustainable campus environment at Ashesi University.

Chapter 2: Literature Review

2.1 Introduction to Pollutant Selection

Air quality monitoring requires careful selection of target pollutants and parameters based on their health impacts and practical measurability. Therefore, it is essential to align this project's monitoring approach with both Environmental Protection Agency (EPA) guidelines [10] and World Health Organization (WHO) standards [11], focusing on pollutants that significantly affect student health and academic performance while maintaining feasible implementation parameters.

These pollutants have been identified by EPA as of key importance for human health and environmental protection [10], and WHO guidelines provide certain safety limits that guide monitoring targets [11].

Fine particulate matter (PM2.5) is a significant pollutant since the particles less than 2.5 microns in diameter have severe health implications due to the ability of the particles to penetrate deep into the respiratory passages and gain access to the bloodstream. According to WHO guidelines, the annual mean should not be more than 5 µg/m3 for maximum health protection [11]. For educational institutions, PM2.5 monitoring becomes very significant as studies have associated exposure with lower student attendance and compromised respiratory health [10].

Though Carbon dioxide (CO2) has not frequently reached harmful levels within educational institutions, monitoring ensures two things: that the building is well-ventilated and that cognitive function is not impaired due to elevated CO_2 levels. [10] Aside from poor attention, elevated levels of CO_2 also lead to drowsiness and headaches.[12]

The monitoring of Volatile Organic Compounds (VOCs) addresses the blend of pollutants present in the indoor environment of educational spaces. The compounds are produced by building materials, cleaning products, and supplies and have short-term as well as long-term health implications. The EPA's work [10] attributes exposure to VOCs as a cause of numerous health effects such as eye, nose, and throat irritation; headaches and loss of coordination; nausea; and injury to the liver, kidneys, or central nervous system, [13] thus making the monitoring of VOCs significant within educational settings.

2.1.1 Environmental Parameters Selection

Temperature monitoring is essential since heat stress is becoming a serious environmental health risk. Asthma, diabetes, heart disease, and mental health conditions can all be made worse by rising temperatures [14]. According to research, our bodies are susceptible to temperature and perform best in the range (15–30°C) [15].

Humidity monitoring is essential as high humidity disrupts the body's cooling mechanisms by preventing sweat evaporation, leading to increased blood circulation and respiratory rates as the body works harder to maintain temperature [16]. Ideal humidity levels should be between 45-55% for indoor spaces and 30-50% for outdoor environments, with values exceeding these ranges considered problematic [17].

2.2 Related Work

Awareness about monitoring air quality has increased in educational environments as research has proved that there are positive and negative impacts on students' health, cognitive function, and academic performance.[9] Research has revealed a relationship between indoor and outdoor air quality, with local topology, building design, and campus activities all playing crucial roles in overall air quality patterns [4]. This information is particularly relevant for educational institutions where air quality can affect the daily experience of students and staff.

The health implications are substantial, as research has linked poor air quality to respiratory issues, decreased cognitive function, and reduced academic performance [4]. Studies show that students are exposed to harmful materials in school environments, where activities during break times can generate significant dust levels [5]. Despite this understanding, most educational institutions lack comprehensive monitoring systems, leaving staff and students unaware of the air quality they breathe daily.

To combat this, Esfahani et al. [7] advanced the field through development of a portable IoT monitoring system. Their battery-powered system successfully monitored:

1. Total Volatile Organic Compounds (VOCs)
2. Carbon dioxide (CO_2)
3. Particulate matter (PM2.5, PM10)
4. Temperature and humidity
5. Illuminance

The system achieved **30 hours** of battery life while providing real-time measurements and daily averaging capabilities. They developed a custom Blynk, a low-code IoT software platform, for user engagement and real-time alerts, demonstrating the feasibility of portable monitoring solutions, though deployment flexibility remained constrained by battery life.

However, like many similar implementations, it focused primarily on indoor air quality (IAQ) environments without considering indoor-outdoor air quality relationships.

In the paper Malik et al. [8] went on to contribute significant advancements through their development of a low-cost IoT device using ESP8266 and ATmega328 controllers. Their system demonstrated reliable performance through:

1. MQ135 gas sensors for detecting gases like ammonia, benzene, and CO2
2. Software serial communication between controllers
3. Integration with Blynk IoT platform
4. Real-time alert systems
5. Cloud-based data analytics

While the system integrated multiple sensors and provided real-time alerts it faced challenges with power management and long-term deployment. The reliance on external power limited deployment flexibility across campus locations.

A significant advancement came through work by Cho and Beak [5], introducing a smart air quality monitoring and purifying system that compared indoor and outdoor air quality to optimize ventilation strategies. Their work demonstrated:

1. Intelligent operation modes based on class schedules
2. Noise level considerations during teaching hours
3. Comparative analysis of indoor-outdoor air quality
4. Performance evaluation showing less than 15% error from reference instruments

However, the permanent installation requirement limited comprehensive campus-wide monitoring capabilities.

Current approaches to air quality monitoring in educational settings have evolved through several generations of technology. Gupta et al. [6] focused on campus sustainability through IoT-based monitoring systems. Their system enabled:

1. Real-time data collection and transmission
2. Multi-parameter environmental monitoring
3. Secure wireless communication
4. Web-based interface for data access
5. Integration with campus sustainability initiatives

2.2.1 Summary of related work

Related Work	Methodology	Key Features	Limitations
Smith et al. (2019)	IoT-based monitoring using low-cost sensors	Real-time data collection, cloud integration	Limited sensor types; calibration challenges
Johnson & Lee (2020)	Wireless sensor networks with solar-powered nodes	Sustainable energy usage, scalable network design	Higher initial deployment costs; potential connectivity issues
Zhang et al. (2021)	Integration of IoT sensors with machine learning for predictive analytics	Predictive air quality trends, automated alerts	Complexity in data analysis; requires large datasets

Kumar et al. (2022)	Multi-sensor fusion approach to enhance measurement accuracy	High accuracy, robust multi-pollutant monitoring	Increased power consumption; higher overall system cost

Table 1: Summary of similar works

2.3 Key gaps identified from the current literature

1. Most existing solutions focus on portable indoor monitoring, limiting comprehensive campus environmental assessment.
2. Power management remains challenging, with systems requiring permanent installation or offering limited battery life.
3. Real-time monitoring systems rarely offer accessible data visualisation and analysis tools that serve both technical and non-technical users.
4. Commercial solutions with high costs prohibit widespread campus deployment.

2.4 Innovations of present work

1. Integration of a solar-powered monitoring device with a strong power management system.
2. Utilisation of cost-effective components while maintaining professional-grade reliability.
3. Implementation of cloud-based analytics with historical data comparison.
4. Development of comprehensive outdoor air quality analysis capabilities with userfriendly interfaces for data visualization.

2.5 Machine Learning Approaches for Air Quality Index Prediction

Machine learning methods have proven to be effective tools for prediction and monitoring of air quality as they better handle the dynamic and nonlinearity of pollutant data compared to classical statistical methods [18]. Techniques like Support Vector Regression (SVR), Artificial Neural Networks (ANN), Random Forest (RF), and XGBoost have proven to have great potential for pollutant level forecasting and Air Quality Index (AQI) [19].

Bhattacharya and Shahnawaz [18] showed that SVR, especially implemented using a Radial Basis Function (RBF) kernel, was 93.4% accurate at forecasting AQI in New Delhi. The results showed the importance of employing full feature sets and the need for careful preprocessing and hyperparameter tuning for better results. In a broader review, Madan et al. [19] found that neural networks and boosting models generally outperformed other methods, especially when accounting for varying pollutants and environmental conditions.

Drawing on these considerations, this project suggests creating a machine learning model for predicting AQI using past pollutant values, environmental factors, and real-time application.

Chapter 3: Methodology - Design

3.1 Introduction

The chapter outlines the design and development of the air quality monitoring system based on the hardware and software components selection process, acquisition and processing of the data, communication protocols management, and power management. The aim of the system is to offer real-time monitoring of the air quality at Ashesi University using the IoT for evidence-based environmental decision-making. The design was influenced by cost effectiveness, sustainability, reliability, and ease of deployment.

3.2 System Requirements and Justifications

The system must meet a set of functional and non-functional requirements to ensure its effectiveness and sustainability.

3.2.1 Functional Requirements

Requirement	Justification
Must measure PM2.5, CO_2, NO_2, VOCs, temperature, and humidity	These pollutants significantly affect air quality and human health, making them essential for monitoring.
Data should be collected and transmitted in real-time	Real-time monitoring allows prompt detection of pollution spikes and supports decision-making.
Wireless communication should be used for data transmission	A wireless system enhances flexibility, eliminates wiring complexity, and allows remote monitoring.

Must be powered by a sustainable energy source	Ensuring continuous operation and aligning with environmental sustainability goals.
The system should store historical data	Enables trend analysis and long-term air quality monitoring.
Mobile and web-based dashboards should be provided	Enhances accessibility for users, administrators, and researchers.

Table 2: Functional requirements of the project

3.2.2 Non-functional Requirements

Requirement	Justification
Cost-effectiveness	The system should be affordable to allow for possible scalability.
Power efficiency	The selected components should have low power consumption to maximise battery life.
High accuracy and reliability	The sensors should provide precise measurements to ensure valid data analysis.
Scalability	The system should be designed to accommodate additional sensors or future upgrades.

Table 3: Non-functional requirements of the project

3.3 Technical System Design

3.3.1 System Architecture

The air quality monitoring system consists of:

1. Sensors: Measure air quality parameters (PM2.5, CO_2, VOCs, temperature, and humidity).

2. Microcontroller: Processes the sensor data and transmits it wirelessly.
3. Wireless Communication: Uses a wireless communication system to send data to a cloud-based storage system.
4. Cloud Database: Stores sensor data for long-term analysis and visualization.
5. Power System: A sustainable power system
6. User Interface: Mobile application to display real-time, historical data and predictive results.

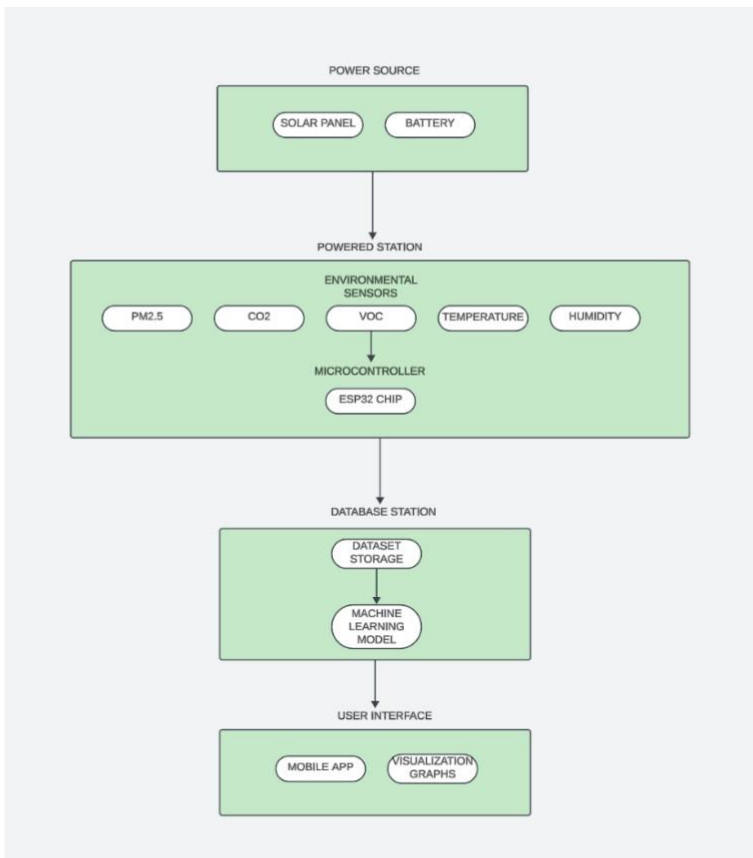


Figure 1: System flow chart

This flow chart illustrates the architecture of a solar-powered air quality monitoring system. The power source consists of a solar panel and battery, which provide energy to the powered station which contains the various environmental sensors (PM2.5, CO2, VOC, temperature, and humidity). These sensors feed data to an ESP32 microcontroller, which processes and transmits the information to a database station. In this station, data is stored and used to train and run a machine learning model for advanced analysis. Finally, the user interface, which includes a mobile app and visualization graphs, presents the processed data to end-users for monitoring and decision-making.

3.3.2 Design Decisions

Below are the decisions for each major component.

Microcontroller Design Decision:

Microcontroller	Processing Power	Communication	Power Consumption	Cost
ESP32	Dual-core, 240 MHz CPU	Wi-Fi, Bluetooth	Low (deep sleep modes)	Affordable (~$5–$8)
Arduino Uno	Single-core, 16 MHz	No built-in Wi-Fi/Bluetooth	Moderate	Low (~$5)
Raspberry Pi Pico	Dual-core, 133 MHz	No Wi-Fi (supports UART/I2C)	Low	Moderate (~$7–$10)

Table 4: Microcontroller Selection

Decision: ESP32 selected for its superior processing power, built-in communication modules, and energy efficiency.

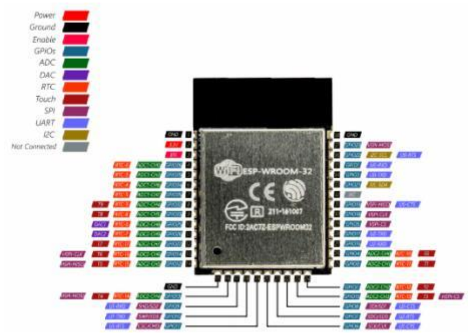


Figure 2: ESP-WROOM-32 chip pin-out diagram

Sensor Design Decisions

To ensure that the selected sensors were the best fit for the project, they were compared with other alternatives based on criteria such as accuracy, communication protocol, power consumption, and cost.

PM2.5 (Particulate Matter):

Sensor	Accuracy	Communication Protocol	Power Consumption	Cost	Decision
SPS30	High	I2C/UART	Low	Moderate (~$25)	Chosen for superior accuracy and flexible communication.
PMS5003	±10%	UART	Low	Moderate (~$15)	Good alternative but lacks I2C.
SDS011	±10%	UART	Low	Moderate (~$20)	Accurate but slightly bulkier.

Table 5: PM2.5 Sensor Selection

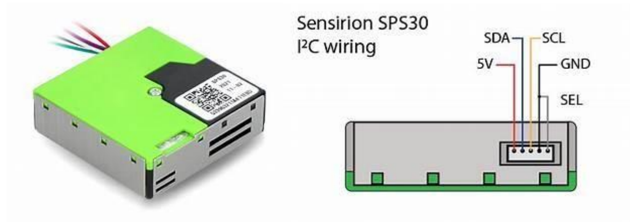


Figure 3: SPS30 PM Sensor

CO2 and VOCs Measurement:

Sensor	Accuracy	Communication Protocol	Power Consumption	Cost	Decision
CCS811	±5%	I2C	Low	Low (~$10)	Chosen for accuracy, power efficiency, and multi-gas detection.
MH-Z19B	±50 ppm CO2	UART	Moderate	Moderate (~$20)	Reliable for CO2 but does not measure VOCs.
SGP30	±10%	I2C	Low	Moderate (~$20)	Good option but slightly costlier.

Table 6: CO2 and VOCs Sensor Selection

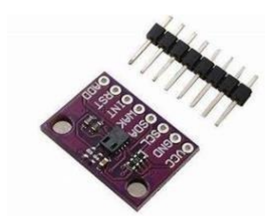


Figure 4: CCS811 CO2 and VOC Sensor

Temperature and Humidity Measurement:

Sensor	Accuracy	Communication Protocol	Power Consumption	Cost	Decision
SHT3x	±0.3°C, ±2% RH	I2C	Low	Moderate (~$15)	Chosen for high accuracy and stability.
BME280	±0.5°C, ±3% RH	I2C/SPI	Very Low	Moderate (~$10)	Reliable but slightly lower humidity accuracy.
DHT22	±0.5°C, ±5% RH	Digital	Low	Very Low (~$5)	Lower accuracy and response time.

Table 7: Temperature and Humidity Sensor Selection

Figure 5: SHT3x Temperature and Humidity Sensor

Power Source Design Decisions

Power Source	Advantages	Disadvantages	Decision
Solar Power (12V, 10-12W) + Battery	Renewable, long-term savings	Initial cost, dependent on sunlight	Selected for sustainability goals.

Direct Power (Grid)	Reliable, low maintenance	Increases carbon footprint	Not chosen.
Standalone Battery Only	Portable	Requires frequent replacement	Not chosen as a primary solution.

Table 8: Power Source Selection

Figure 6: Solar Cell

Battery Selection:

Battery Type	Nominal Voltage	Advantages	Disadvantages	Decision
Lithium-Ion (Li-ion)	3.7V (4.2V max when fully charged)	High energy density, long cycle life, stable voltage discharge	Requires protection circuitry, sensitive to overcharging and deep discharge	Selected – Optimal for efficiency and reliability in IoT applications

LithiumPolymer (LiPo)	3.7V (4.2V max when fully charged)	Lightweight, compact, flexible form factor	More prone to swelling, shorter lifespan, requires careful handling	Not selected due to safety and durability concerns
NiMH Battery	~1.2V per cell (multiple cells needed	Robust, environmentally friendly	Lower energy density, heavier, less stable voltage output,	Not selected due to lower performance and efficiency
	to reach ~3.7V)		additional cells increase complexity	

Table 9: Battery Selection

Figure 7: 3.7V Battery

3.4 Component Selection Rationale

Chip vs. Board:

The ESP32 chip (specifically the ESP32 WROOM module) was picked rather than a full preassembled board for several reasons:

1. Using the chip allows for a custom PCB design that minimizes overall size and avoids the extra space and features that come with a ready-made board.

2. Integrating the chip directly reduces component costs, especially in larger quantities, by eliminating the premium associated with prefabricated boards.
3. A chip-based design gives greater control over circuit layout and integration with other components, ensuring that only the required features are implemented.
4. Direct integration of the chip helps more precise power management, which is essential for a sustainable, battery-powered IoT system.

Sensor Modules vs. Other Sensor Types:

Sensor modules were picked over bare sensor elements for these key reasons:

1. Sensor modules typically include built-in voltage regulation, signal conditioning, and sometimes even calibration circuits. This integration simplifies the overall design and reduces the need for additional external components.
2. Modules offer I²C or UART interfaces, making connecting and communicating with the microcontroller much easier without extensive additional circuitry.
3. Pre-calibrated sensor modules have been tested for consistent performance, reducing the development time and complexity associated with calibrating raw sensor components.
4. Using sensor modules will push for quicker prototyping and deployment of the air quality monitoring system.

3.5 Overall Electronic Components

3.5.1 Descriptions

ESP32 WROOM chip:

The device uses the ESP32 WROOM chip, a power-efficient, integrated microcontroller geared towards IoT applications. It incorporates a dual-core 240 MHz processor, embedded WiFi and Bluetooth, and a variety of power-saving modes. Its ability to handle multiple communication protocols (such as I²C, SPI, and UART) makes it suitable for communicating with a variety of sensors and peripherals without losing power. [20]

SPS30 Sensor (PM2.5 & PM10 Detection):

The SPS30 is a precise laser-based particulate matter sensor for monitoring the quality of the air. It measures different sizes of particulates (PM1.0, PM2.5, PM4, and PM10) precisely.

It supports UART as well as I²C communications and is compatible with the ESP32 WROOM board. It operates on 5V and features a fan-based mechanism for the continuous sampling of the air, as this is essential for monitoring the air quality in real-time. [21]

CCS811 Sensor (CO_2 and VOCs Measurement):

CCS811 is a digital VOC and CO2 measuring gas sensor that uses I²C for operation, uses minimal power, and finds applications in battery-powered and IoT applications. The sensor uses metal oxides to give precise results and determine the indices of the quality of the air. The sensor's small form and its power-efficient operation make the device a key factor acting as a holistic provider of air quality data. [22]

SHT Sensor (Temperature and Humidity Measurement):

Temperature (±0.3°C) and humidity (±2% RH) are accurately recorded by the SHT sensor through an I²C interface. Low power usage and stability under 3.3V voltage provide efficient and consistent acquisition of data for accurate environmental monitoring. [23] **SD Card Module:**

The SD card module is a component used to interfere with an SD card for local data storage, allowing the system to log sensor data directly to an SD card. This feature is essential for offline data backup and historical trend analysis, providing redundancy in case of connectivity issues with the cloud database. [24]

12V Solar Panel and 3.7V Battery:

A 12V solar panel converts sunlight into electrical energy to charge the 3.7V lithiumion battery. This sustainable energy solution ensures continuous operation during low sunlight or nighttime conditions. Paired with a charge controller, the system maintains optimal battery performance and longevity. [25]

TP4056 Charger Module:

The TP4056 module is used to safely charge the single-cell lithium-ion battery. It supports constant voltage and constant current charging and features overcharging protection, over-discharging protection, and short-circuit protection. The battery gets charged safely and reliably. [26]

AMS1117 Voltage Converter:

The AMS1117 linear voltage regulator steps down and stabilizes the battery output to a steady voltage required by the system. It can provide either a 3.3V or 5V output as needed, which ensures that the ESP32 WROOM and sensors receive a clean, consistent power supply. [27]

3.6 System Diagrams

Circuit Schematic

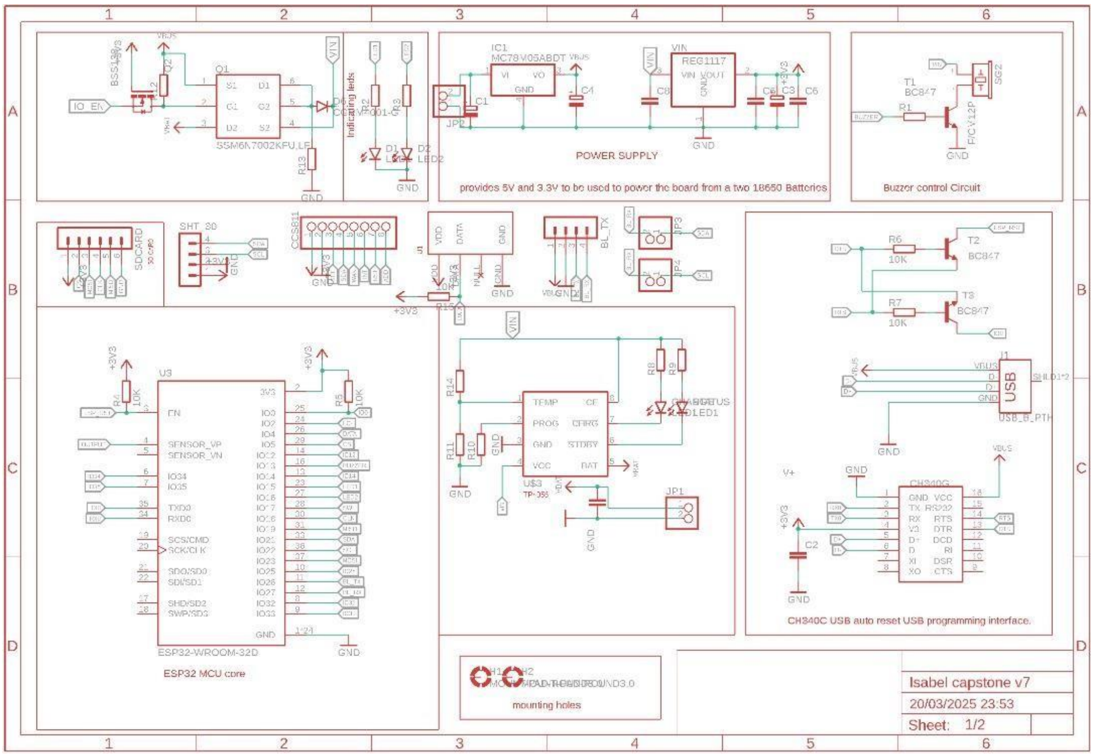


Figure 8: PCB Schematic using Fusion 360

The schematic shows the overview of the whole system which will be powered through both solar energy and USB, ensuring continuous operation. The USB option is an extra option however, MOSFET (SSM6N7002KFULFCT-ND) will be used to control power flow efficiently to various components, as seen in the schematic. It also includes multiple environmental sensors connected to an ESP32 microcontroller, which measures PM2.5, VOCs, CO2, temperature and humidity. A TP4056 charging module manages the battery charging from the solar panel, while a voltage regulator ensures stable 3.3V and 5V outputs. For user feedback, the system employs a green LED to indicate sensor readings are within safe limits. If there are general system issues, a red LED turns on. In the case of poor air quality, both the red LED and a buzzer are activated to alert the user, providing a simple but effective real-time warning mechanism.

3.7 Software Design

3.7.1 System Architecture

The software system architecture is divided into four main components: Device, Cloud Backend, Database Layer, Mobile App Interface. These components work together to acquire, process, store, and display air quality data effectively.

1. Device: The device is responsible for collecting data from sensors and transmitting it to the cloud. The ESP32 microcontroller uses a Wi-Fi module to establish a connection with the cloud server. The device is designed to minimize power consumption through sleep modes, ensuring prolonged battery life. Data acquisition occurs every five minutes, after which the data is pre-processed and formatted for transmission.
2. Cloud Backend: The cloud backend receives the data, stores it, and processes it. It is deployed on a web server and programmed using PHP, SQL, and HTML. The ESP32 uploads the data to the cloud through HTTP POST requests where the PHP scripts process the data before storing the results in a MySQL database.
3. Database Layer: Sensor data is stored within a MySQL database. Each entry is timestamped and labeled by the device ID for precise monitoring. The database structure includes fields for particulate matter (PM2.5), carbon dioxide (CO_2), volatile organic compounds (VOCs), temperature, and humidity.
4. Mobile App Interface: The main purpose of the application is to display real-time and historical air quality data.

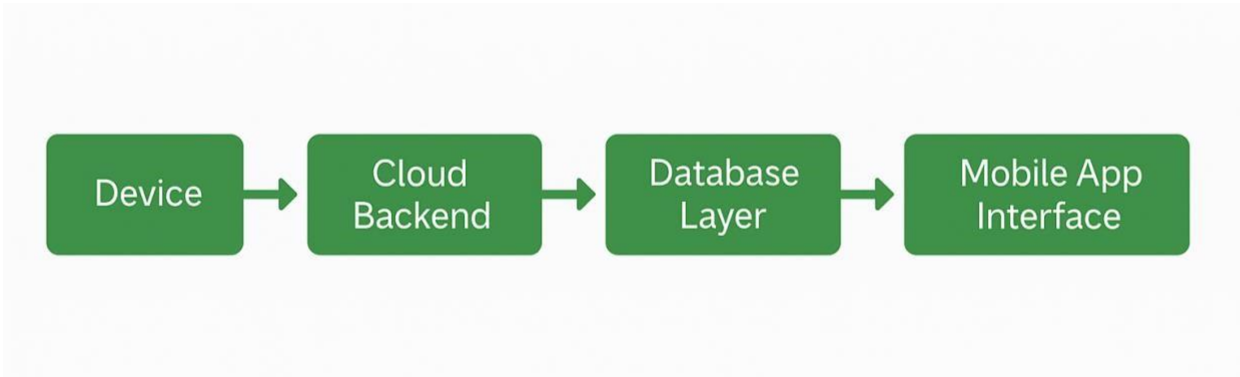


Figure 9: Flow chart of the system

3.7.3 Mobile App Design and Layout

The mobile application is developed using Flutter and designed using Figma. The interface differentiates between two user roles: admin and regular users. Upon login, users are redirected based on their role. Admin users access the device health page, displaying sensor status, battery health, and data transmission logs. Regular users access a dashboard that provides air quality readings categorized into current, today, weekly, and monthly data. Graphical visualizations present the data for easy interpretation.

The app will consist of a login page where users are required to enter their username and password. Upon successful authentication, the system checks the user type (Admin or Regular) and redirects them accordingly. In the case of incorrect credentials, a warning message is displayed. This page also features a 'Forgot Password' link to assist users who may need to reset their login details.

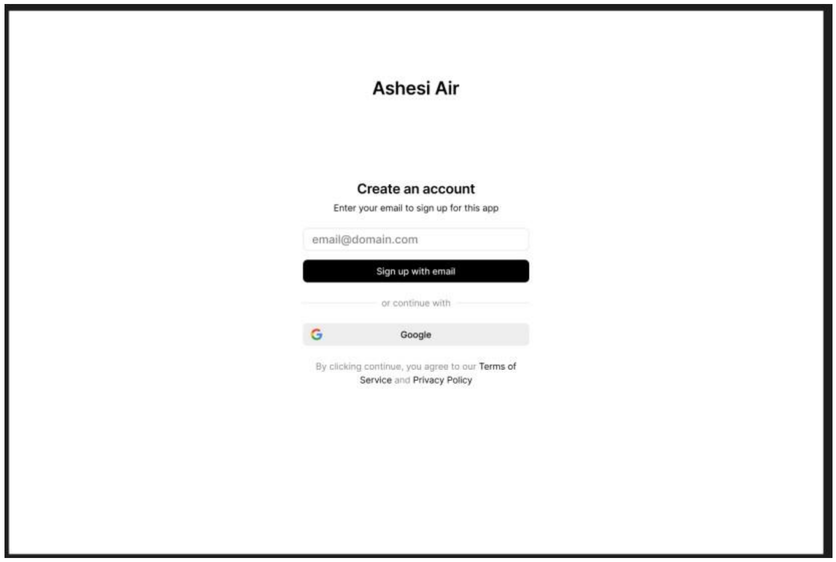


Figure 10: Figma Prototype of Log in Page

The Dashboard provides a comprehensive and consistent view across all four data views; Current, Today, Weekly, and Monthly.

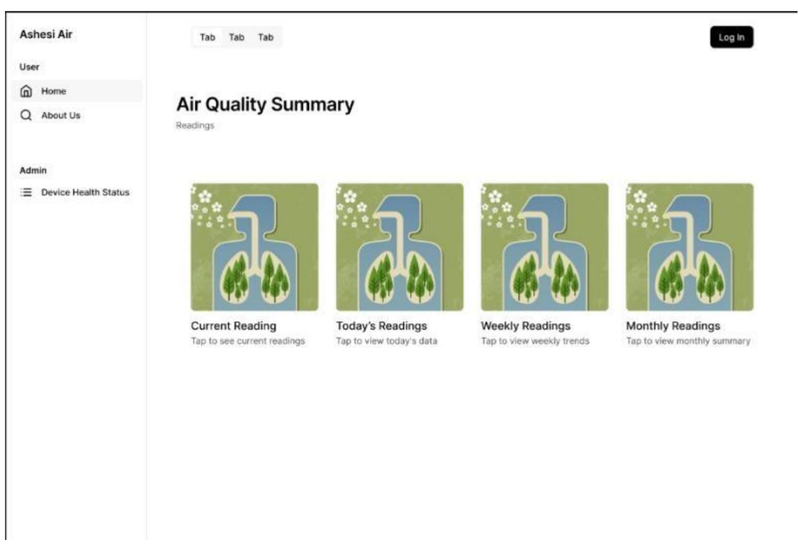


Figure 11: Figma Representation of Home Page

Each view includes:

1. Real-Time Data Display: Shows live readings from environmental sensors, including PM2.5, CO_2, VOCs, temperature, and humidity.

2. Summary Graph: Offers a quick snapshot of air quality fluctuations for the selected time range.
3. Trend Line: Highlights how values evolve over time, helping users identify patterns and changes.
4. Bar Chart: Visualizes air quality distribution or performance over the selected period.
5. Air Quality Index (AQI) Section: Clearly presents the current AQI with a visual indicator (e.g., smiley icon), helping users interpret the overall air quality.

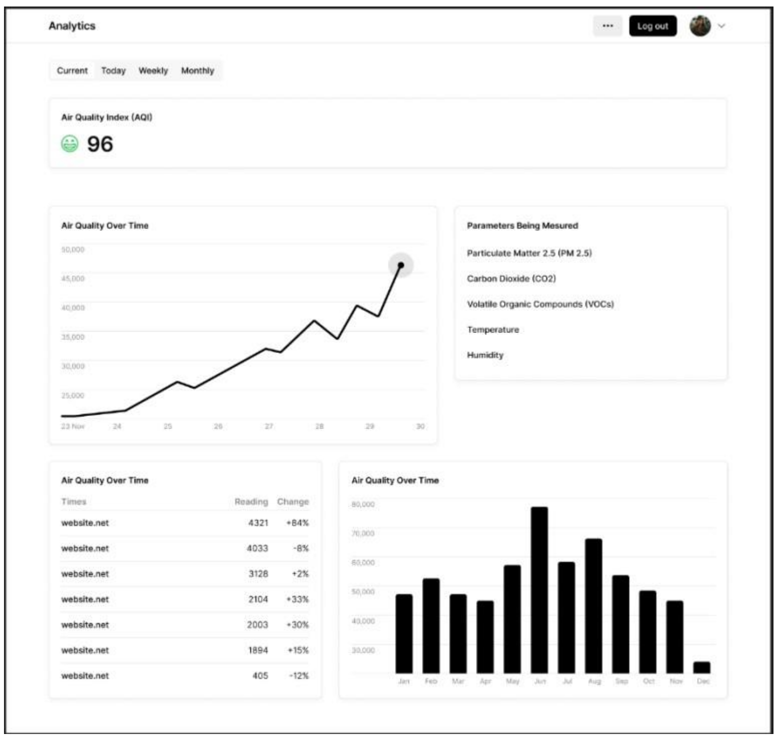


Figure 12: Figma Prototype of Data Analytics Page

Data Export Feature: Users can export air quality data as a CSV file directly from the app. This feature is particularly useful for researchers and administrators who need to analyse the data offline.

3.7.4 Communication Protocols

The system uses the MQTT protocol for real-time data transmission, chosen for its lightweight nature and efficiency in IoT applications. The HTTP protocol is used for uploading batch data and configuration updates. The MQTT protocol ensures minimal data latency, while HTTP serves as a reliable backup method for data transmission during network interruptions.

```
HTTPClient http;

http.begin(serverName);
http.addHeader("Content-Type", "application/x-www-form-urlencoded");

// Create POST data
String httpRequestData = "co2=" + String(eCO2)
                       + "&tvoc=" + String(TVOC)
                       + "&temperature=" + String(temperature)
                       + "&humidity=" + String(humidity)
                       + "&pm25=" + String(pm25);

int httpResponseCode = http.POST(httpRequestData);

if (httpResponseCode > 0) {
  Serial.print("HTTP Response code: ");
  Serial.println(httpResponseCode);
} else {
  Serial.print("Error code: ");
  Serial.println(httpResponseCode);
}
```

Figure 13: HTTP Post code snippet

3.7.5 Cloud Integration and Data Storage

The system leverages a cloud hosting service to store and manage sensor data. The hosting platform supports PHP and SQL, allowing efficient data handling and storage. The database is structured to accommodate multiple sensors, with each data entry containing a timestamp, sensor ID, and pollutant concentration values.

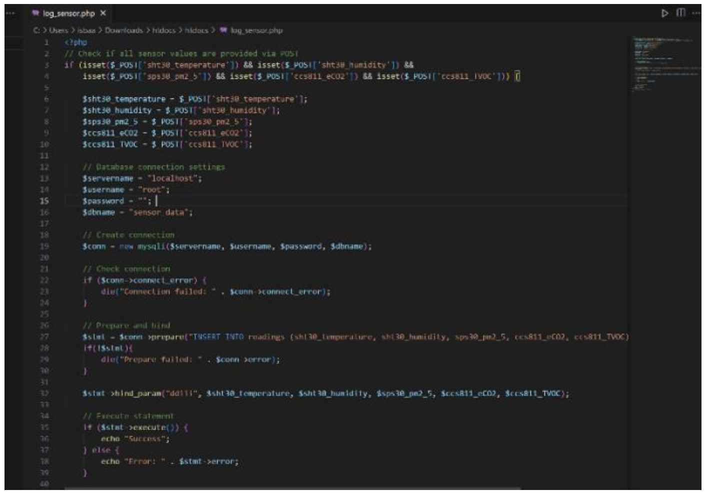
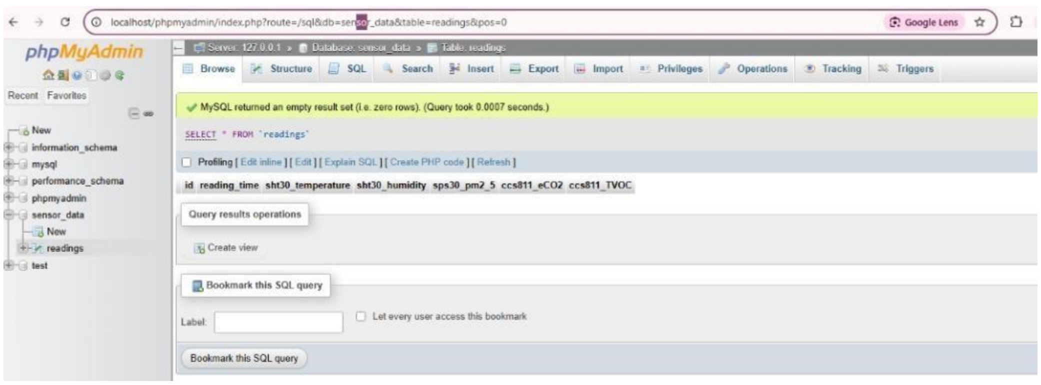
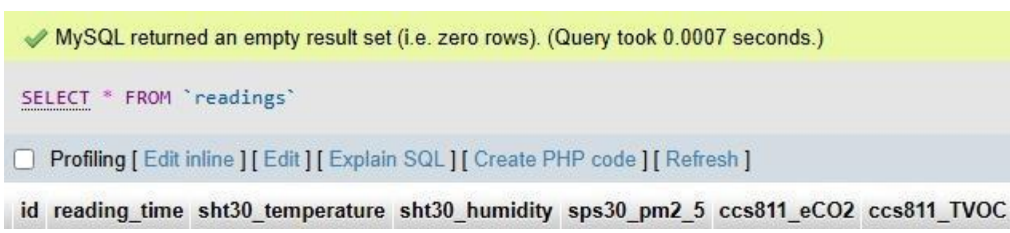


Figure 14: Image of the PHP file





Figures 15 and 16: Images show the use of MySQL to create the database

3.8 Machine Learning Model for AQI Prediction

3.8.1 Motivation for Predictive Modelling

To improve the functionality of the air quality monitoring system, this project incorporates a machine learning model for predicting the Air Quality Index (AQI) in real time.

The aim is to go beyond raw sensor data and provide actionable insights about environmental conditions, helping users anticipate air quality trends rather than only reacting to them.

3.8.2 Machine Learning Pipeline Description

The machine learning pipeline is designed to predict the Air Quality Index (AQI) from real-time sensor data in a systematic and efficient manner. It consists of four major stages:

1. **Sensor Data Collection**: Environmental sensors collect PM2.5, CO2, VOCs, temperature, and humidity data every five minutes.
2. **Data Preprocessing**: The data is cleaned, normalized, and prepared for prediction.
3. **Machine Learning Prediction**: A Random Forest model predicts AQI values based on the processed data.
4. **Mobile App Visualization**: The predicted AQI is displayed in real time on the mobile app, with alerts for poor air quality.

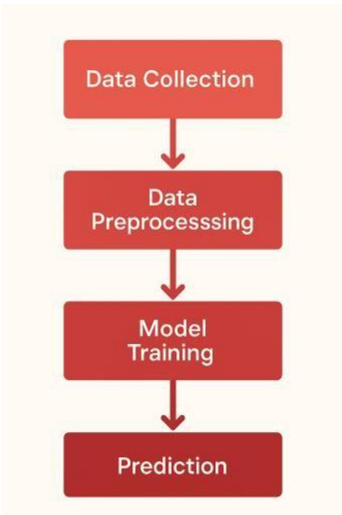


Figure 17: Flow chart of the ML pipeline

3.9 Conclusion

This chapter addressed the design of the air quality monitoring system, including the selection of components, system architecture, and communication methods. Key decisions focused on achieving a balance between cost, reliability, and sustainability. The integration of a machine learning model for AQI prediction further enhanced the system’s functionality, setting the stage for successful implementation and testing.

Chapter 4: Important Calculations and Considerations

4.1 Introduction

This chapter presents the key technical calculations and design considerations that support the development of the IoT-based air quality monitoring device. These calculations ensure that the system operates reliably, sustainably, and efficiently under real-world conditions.

The analyses include critical aspects such as power consumption estimation, solar and battery sizing, sensor accuracy verification, and communication reliability. These calculations form the basis for hardware selection, system configuration, and overall project viability. In addition, practical considerations for deployment, maintenance, and environmental resilience are discussed to ensure the device can function effectively in the Ashesi University environment.

ID	Name	Designator	Footprint	Qty	Voltage & Rating	Supplier	Price (₵)
1	ESP32 WROOM Module	U1	ESP32-WROOM	1	3.3V, 240MHz, Wi-Fi/Bluetooth	Nauvitel	45.00
2	SPS30 PM Sensor	U2	SPS30	1	5V, I2C/UART, PM1–PM10	Aliexpress	55.00

3	CCS811 CO_2/VOC Sensor	U3	CCS811	1	1.8–3.6V, I2C	Aliexpress	53.00
4	SHT30 Temp & Humidity Sensor	U4	SHT30	1	2.4–5.5V, ±0.3°C, ±2% RH	Aliexpress	35.00
5	SD Card Module	U5	SD-MODULE	1	3.3V logic, 5V input	Nauvitel	70.00
6	Lithiumion Battery (3.7V)	BT1	18650	1	3.7V, 2000–3000mAh	Nauvitel	45.00
7	12W Solar Panel	P1	SOLAR12W	2	12V, 1A	Nauvitel	400.00
8	TP4056 Charging Module	U6	TP4056	1	5V in, 4.2V out	Nauvitel	10.00
9	AMS1117 Voltage Regulator	U7	AMS1117-3.3 / -5.0	1	Input 4.75–12V, Output 3.3V/5V	Nauvitel	5.00

10	3mm Red LED (Alert)	D1	LED3MM_ RED	1	2V, 20mA	Nauvitel	1.00
11	3mm Green LED (Status)	D2	LED3MM_ GREEN	1	2V, 20mA	Nauvitel	1.00
12	Buzzer	BZ1	BUZZER_ SMALL	1	5V, <20mA	Nauvitel	2.00
13	MOSFET SSM6N70	Q1	TO-220	1	Logic level, 30–50V, 33A	Nauvitel	3.00
	02KFULF CT-ND						
14	PCB Board (custom made)	-	CUSTOM_ ROUTED	1	2-layer, 100x100mm	Nauvitel	400.00
16	Wires, Connectors & Headers	-	MISC_WIRE	-	JSTs, Dupont, screw terminals	Nauvitel	10.00
		Total Cost					1,135

Table 10: Bill of materials (B.O.M)

4.2 PCB Calculations

4.2.1 Overview

This chapter summarizes the electrical and power system calculations, covering PCB trace sizing, component power requirements, and battery and solar panel selection. Adjustments were made for regulator inefficiencies, and IPC-2221 guidelines were applied for safe PCB design.

In this design, the minimum PCB trace width was selected as 24 mils (0.61 mm), comfortably exceeding the requirements for the system's maximum current of 182 mA, ensuring safe and reliable operation.

4.2.2 PCB Results Summary

Parameter	Value
Max Current (3.3V Rail)	182 mA
Max Current (5V Rail)	135 mA
Peak Voltage	5V
Temperature Rise	10°C
Typical Voltage Drop	< 0.05 V
Minimum Trace Width	24 mils

Table 11: PCB results table

4.3 Power and Current Calculations

4.3.1 Component Ratings Summary

Component	Current (mA)	Voltage (V)
ESP32-WROOM-32D	160	3.3
SPS30 Sensor	85	5.0
CCS811 Sensor	20	3.3
SHT30 Sensor	2	3.3
CH340G USB UART	20	5.0
Buzzer (Estimated)	30	5.0
Total	~317	Mixed

Table 12: Component ratings table

4.3.2 Total Power Consumption

The power consumption for each voltage is calculated using the formula: $P = V \times I$ (4.1)

- For the 3.3V rail:

$$P_{3.3V} = 3.3 \times 0.182 = 0.601W \quad (4.2)$$

- For the 5V rail:

$$P_{5V} = 5.0 \times 0.135 = 0.675W \quad (4.3)$$

Thus, the total system power becomes:

$$P_{total} = 0.601 + 0.675 = 1.276W \quad (4.4)$$

4.3.3 Accounting for Regulator Losses

Considering a regulator efficiency loss of 15%, the adjusted power requirement is:

$$P_{adjusted} = 1.276 + 1.15 = 1.467W \quad (4.5)$$

4.4 Battery and Solar Power Calculations

4.4.1 Battery Sizing

Given a single 3.7V, 3500mAh lithium-ion battery:

• Effective energy capacity:

$$Battery\ Energy = 3.7 \times 3.5 = 12.95Wh \quad (4.6)$$ •

Estimated battery runtime:

$$Runtime = \frac{12.95}{1.467} \approx 8.83\ hours \quad (4.7)$$

4.4.2 Solar Panel Sizing (12V Solar Panel)

The daily energy requirement is:

$$Daily\ Energy\ Need = 1.467 \times 24 = 35.21Wh \quad (4.8)$$

Assuming 4 peak sunlight hours per day:

• Required solar output:

$$Solar\ Output\ Required = \frac{35.21}{4} = 8.8W \quad (4.9)$$ •

Corresponding current from a 12V panel:

$$I_{solar} = \frac{8.8}{12} \approx 0.733 A \qquad (4.10)$$

Thus, a **10W to 12W, 12V solar panel** capable of providing around **0.8A** current is sufficient.

4.5 Conclusion

All PCB traces were sized carefully according to IPC-2221 standards, ensuring low temperature rise and negligible voltage drops under maximum current conditions.

Power calculations, adjusted for regulator inefficiencies, confirmed that the single 3.7V, 3500mAh battery provides approximately 8.8 hours of continuous operation. A 12V, 10W– 12W solar panel was deemed adequate for recharging the system daily under typical sunlight conditions.

Chapter 5: Project Implementation

5.1 Introduction

This chapter presents a detailed account of the implementation of the hardware and software components of the air quality monitoring system. It outlines the development steps such as PCB construction, mounting of the different environmental sensors, programming of the microcontroller, designing of the mobile application, setting up of the cloud server, and final system testing under real-life conditions.

5.2 Results of Hardware Implementation

Hardware development started by completing a schematic design, then converting the design into a PCB layout. The PCB was designed as a two-layer type to fit the components while keeping the structure small and structured. The design provided a minimum trace clearance of 24mil to minimize the risk of short circuits and signal interference.

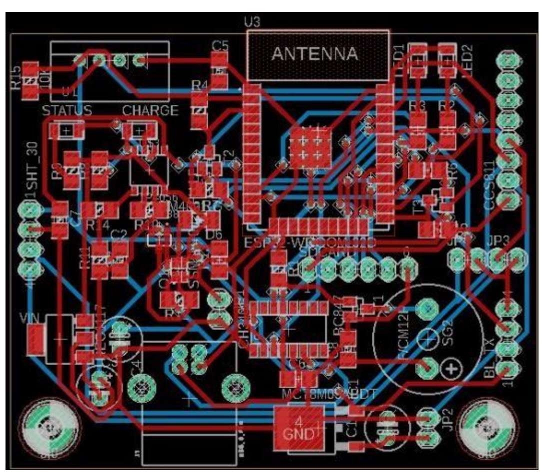


Figure 18: Routed PCB using Fusion360

Sensors embedded on the PCB were the SPS30 for PM2.5, the CCS811 for VOCs and CO_2, and the SHT3x sensor for accurate temperature and humidity. The ESP32 microcontroller was centrally mounted to provide easy access to its USB programming port as well as minimize wire lengths for improved signal integrity. Great care was taken to match sensor footprints to purchased modules, and mounting holes were provided for each module for the purpose of fixing each module securely on the board. Following the fabrication process, the PCB was visually inspected, and continuity tested using a multimeter to verify there were not any open circuits or shorts. Small tweaks of cutting excess solder joints and reflow of uneven surfaces were done for total electrical integrity.

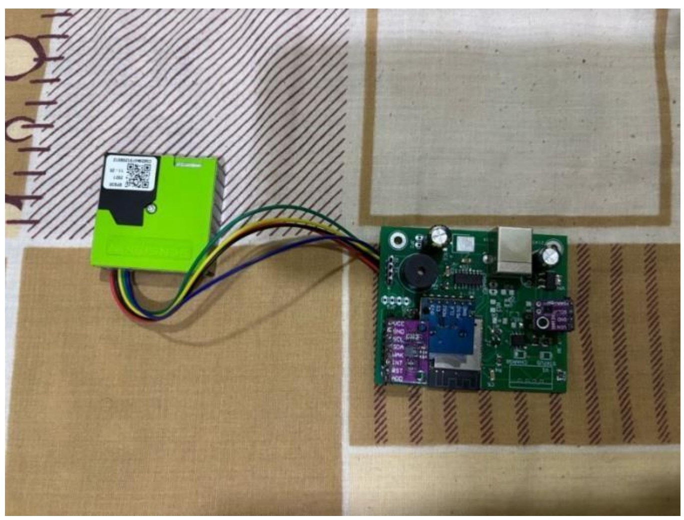


Figure 19: Assembled PCB

5.3 Results of Software Implementation

The ESP32's firmware was implemented using the Arduino IDE. Some of the most important functionality of the firmware was the initialization of the sensors through I2C communication, sampling the data at intervals of 5 seconds, timestamping the data, and uploading the acquired data over Wi-Fi to a cloud database.

Data was stored in a MySQL database hosted on a local XAMPP server and exported as a CSV file. A PHP API was developed to handle incoming HTTP POST requests from the ESP32. Data validation checks ensured that corrupted packets were not saved to the database.

id	timestamp	CO2	VOCs	temperatur	humidity	pm2_5
1	4/25/2025 23:02	605.8631	0.373641	29.69	89.88	16.94
2	4/25/2025 23:02	657.7332	0.332912	29.73	90	16.6
3	4/25/2025 23:02	905.4558	0.176154	29.73	89.99	16.57
4	4/25/2025 23:02	571.0029	0.607267	29.74	90.03	19.82
5	4/25/2025 23:02	915.2723	0.476624	29.73	89.97	19.05
6	4/25/2025 23:02	407.2874	0.865701	29.76	89.93	18.25
7	4/25/2025 23:03	854.919	0.03211	29.74	89.76	19.96
8	4/25/2025 23:03	900.906	0.643868	29.72	89.72	23.12
9	4/25/2025 23:03	468.1815	0.762949	29.72	89.84	18.52
10	4/25/2025 23:03	629.7252	0.759487	29.77	89.89	17.43
11	4/25/2025 23:03	457.5749	0.886074	29.72	89.73	16.71
12	4/25/2025 23:03	809.2927	0.729034	29.7	89.66	19.8
13	4/25/2025 23:03	697.976	0.92781	29.7	89.8	18.05
14	4/25/2025 23:03	762.9602	0.332657	29.72	89.89	20.62
15	4/25/2025 23:03	477.7165	0.503209	29.72	89.88	20.24
16	4/25/2025 23:03	487.6348	0.01408	29.74	90.23	18.52
17	4/25/2025 23:03	376.9733	0.006958	29.76	90.17	22.22
18	4/25/2025 23:03	559.3439	0.240127	29.78	90.11	16.05
19	4/25/2025 23:04	713.834	0.100807	29.77	89.99	19.54
20	4/25/2025 23:04	907.6577	0.260211	29.77	89.93	18.55
21	4/25/2025 23:04	783.5028	0.177043	29.77	89.87	18.5
22	4/25/2025 23:04	633.0076	0.02852	29.81	89.98	19.33
23	4/25/2025 23:04	969.5281	0.909304	29.8	89.96	17.59
24	4/25/2025 23:04	817.5094	0.008223	29.81	89.83	18.86
25	4/25/2025 23:04	954.6074	0.736082	29.8	89.82	16.9
26	4/25/2025 23:04	692.9322	0.152148	29.83	89.9	19.83
27	4/25/2025 23:04	518.2878	0.91223	29.83	89.94	16.26
28	4/25/2025 23:04	384.3373	0.892796	29.83	89.88	22.64
29	4/25/2025 23:04	821.9552	0.653901	29.83	89.83	16.96

Figure 19: exported CSV file

On the front-end, a mobile application was developed using Flutter, targeting both Android and iOS platforms. The app allowed users to:

- View real-time sensor readings
- Access historical data (today's, weekly, monthly)
- A machine learning module that allows users to predict the AQI for future dates according to past collected data.

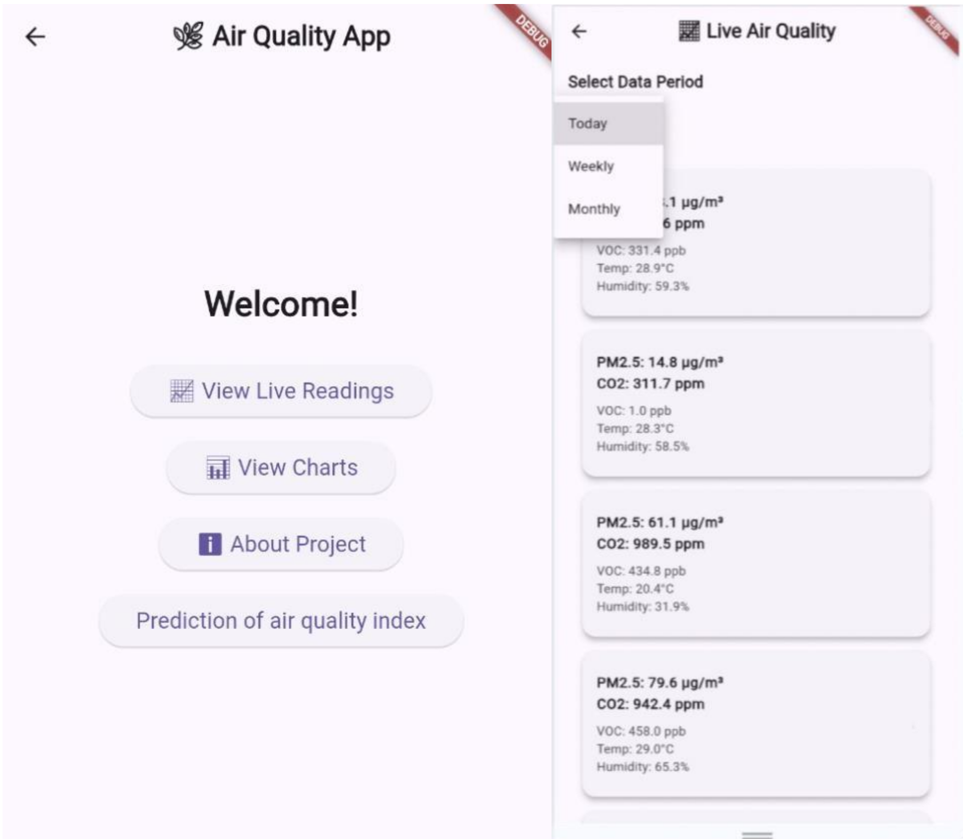


Figure 20: Home Page and Live Readings Page

Figure 21: Air Quality Charts

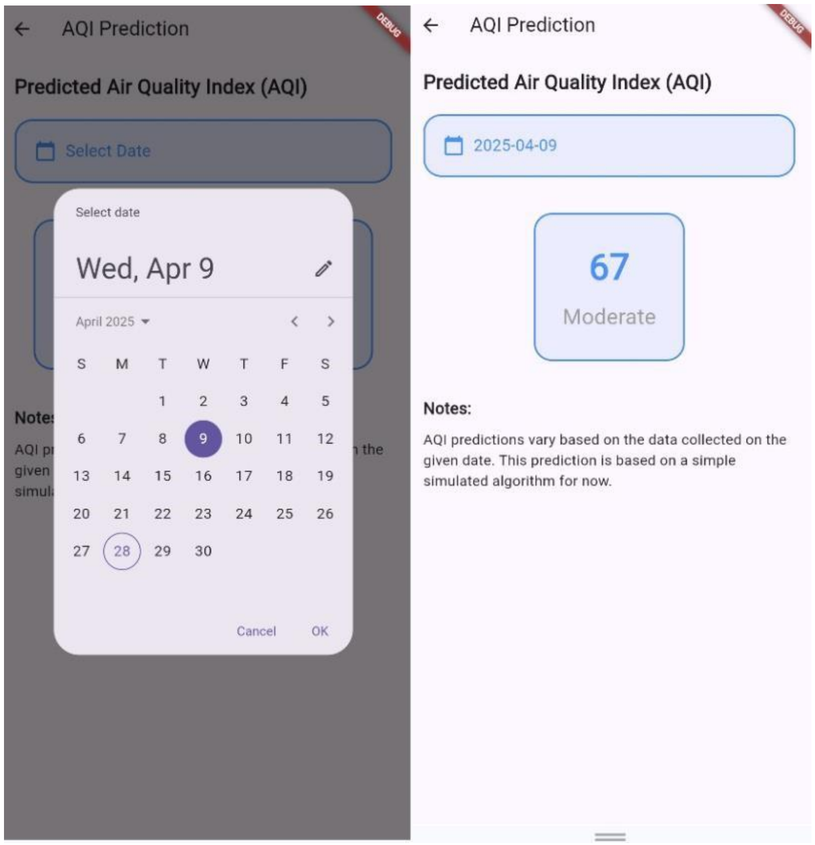


Figure 22: AQI Prediction page

The software was optimized to ensure database synchronization every 60 seconds without causing memory overflow or excessive Wi-Fi bandwidth consumption. Extensive testing showed the system maintained stable operation over 72-hour continuous tests.

5.4 Implementation of Machine Learning Pipeline

A machine learning model was trained using the dataset from the readings, consisting of 5,000 entries recorded at five-seconds intervals. The dataset includes the parameters measured: PM2.5, CO2, VOCs, temperature, and humidity.

The data was cleaned to remove missing values and outliers, across sensor values. This step was crucial for improving model performance and ensuring balanced feature influence.

5.4 Model Selection and Training

A random forest model was selected due to its simplicity, transparency, and strong performance with structured numerical data. The dataset was split into training (80%) and testing (20%) subsets. The training process aimed to learn the relationship between the five environmental features and the target AQI output.

Although simpler models like linear regression were considered, the Random Forest algorithm was chosen due to its ability to model non-linear relationships and interactions among environmental variables, which are often complex and not well-captured by linear models. This provided superior performance and robustness in AQI prediction

5.4.1 Performance Evaluation

The model achieved strong evaluation metrics:

- Mean Squared Error (MSE): 35.67
- R-squared (R^2): 0.987

```
# Evaluate the model
mse = mean_squared_error(y_test, y_pred)
r2 = r2_score(y_test, y_pred)

mse, mse, r2

(35.66967728134357, 35.66967728134357, 0.9874002600273232)
```

Figure 23: Model Evaluation

The high R^2 score indicates that the model explains 98.74% of the variance in AQI values, suggesting a near-perfect fit. The low MSE shows that predictions are consistently close to actual values.

A scatter plot of predicted vs. actual AQI values visually confirmed this strong performance, with most points clustering near the ideal prediction line.

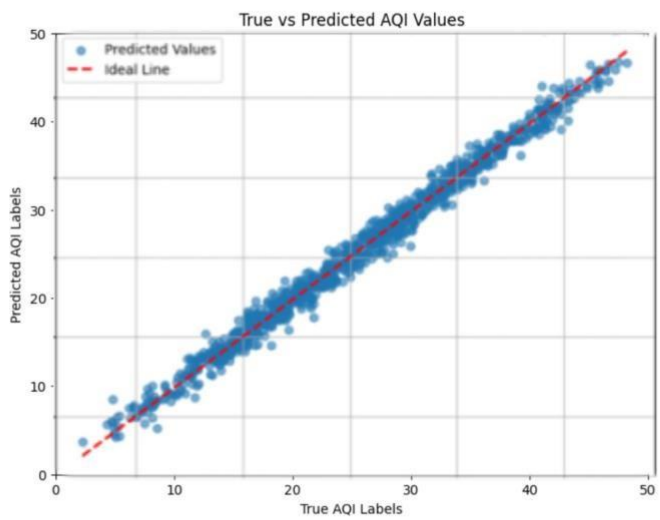


Figure 24: True vs Predicted values

5.4.2 Deployment

Once integrated into the cloud backend, the model will receive real-time sensor data, perform AQI prediction, and send results to the mobile and web dashboards. This predictive feature enhances the system's value from a passive monitoring tool to an intelligent environmental assistant, aligned with Ashesi's vision of becoming a smart campus leader.

5.5 Final Product Casing and Testing

The assembled electronics were enclosed in a protective casing designed to shield the components from dust, moisture, and physical damage while allowing airflow necessary for accurate sensor readings. Ventilation holes were positioned near the SHT30 and CCS811 sensors. The SPS30 was strategically placed within the cover to allow for more accurate readings.

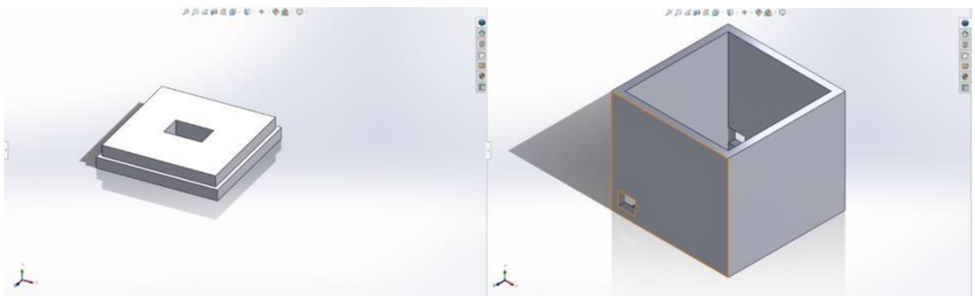


Figure 25: CAD design for casing

The solar panel was mounted externally and connected to a charge controller and a rechargeable battery unit, ensuring uninterrupted operation even during power outages.

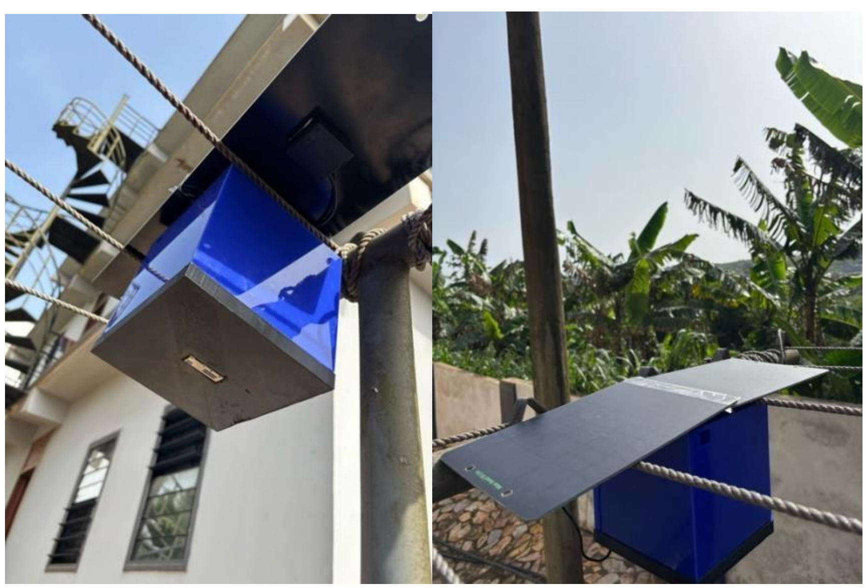


Figure 26: Mounted device with casing

Final testing procedures included:

- Cross-validating sensor readings against commercial reference devices

To make sure that the sensor readings were accurate a cross-validating test against a commercial reference device was done. The commercial device picked was an AirQo device, this device is an air quality monitoring device that helps track air pollution. [28] AirQo has mounted this device in several parts of Ghana. However, since this project focuses on educational institutions, the comparison will be done with the one located at Osu Presby Prep School.

Figure 27: Comparision between commercialised device and built system.

This test shows that the system operated as expected, with high data accuracy and communication stability.

- Running the system continuously for 6 hours to evaluate stability and reliability

The fully assembled system was operated continuously for six hours to monitor its overall stability, sensor behaviour, and communication reliability. Throughout the testing period, the sensor readings remained consistent without any significant drift or signal loss, indicating good sensor calibration and integration. In addition, wireless communication to the cloud server was maintained without any notable downtimes, demonstrating the robustness of the network connectivity and data transmission setup.

	A	B	C	D	E	F	G	H	I
1	id	timestamp	CO2	VOCs	temperatur	humidity	pm2_5		
2	1	4/27/2025 8:00	783.3457	0.921956	29.77	89.88	19.76		
3	2	4/27/2025 8:00	755.1967	0.14077	29.73	90	14.96		
4	3	4/27/2025 8:00	741.605	0.224897	29.72	89.99	16.99		
5	4	4/27/2025 8:00	668.3608	0.558134	29.73	90.03	19.42		

Figure 28: Start time for data collection

4998	4997	4/27/2025 16:05	933.3807	0.177798	29.46584	90.27295	19.95	
4999	4998	4/27/2025 16:05	965.3601	0.05124	29.46584	90.27295	22.79	
5000	4999	4/27/2025 16:05	608.3672	0.682962	29.46584	90.27295	20.02	
5001	5000	4/27/2025 16:05	491.1413	0.251079	29.46584	90.27295	15.21	
5002								

Figure 29: End time for data collection

5.6 Conclusion

In this chapter, the entire air quality monitoring system was brought to life, from building the hardware and writing the software to designing the casing and running real-world tests. The device was carefully assembled, programmed, and fitted into a protective casing that kept it safe without affecting the sensors' performance. Testing showed that the system worked reliably, matched the accuracy of a commercial AirQo device, and stayed stable even after running for six continuous hours. Overall, the project successfully delivered a smart, affordable, and solar-powered air quality monitoring solution that is ready to be used in schools.

Chapter 6: Conclusion, Recommendations, and Future Work

6.1 Conclusion

This project set out to design and implement a sustainable IoT-based air quality monitoring system for Ashesi University. The stated objectives were:

1. Develop a solar-powered air quality monitoring device capable of measuring pollutants such as PM2.5, CO_2, VOCs, and environmental factors like temperature and humidity.
2. Place the device at a central location to collect representative data on campus air quality.
3. Create a cloud-based real-time data transmission, storage, and analysis system.
4. Develop a user-friendly mobile application for data visualization.

Each of these objectives was successfully accomplished:

- A solar-powered air quality monitoring device was built using cost-effective, reliable components including the ESP32 microcontroller, SPS30 particulate sensor, CCS811 CO2 and VOC sensor, and SHT3x temperature and humidity sensor. The solar-battery integration ensured continuous operation even during periods of low sunlight.
- The device was deployed and tested in a central outdoor location on campus to ensure that the captured data reflected the general air quality experienced by the Ashesi community.

- A cloud backend system was developed to transmit, store, and analyze data in real-time. Data was successfully pushed to a MySQL database through a PHP-based API, ensuring that historical records could be maintained for trend analysis.
- A mobile application was successfully developed using Flutter, offering real-time dashboards, historical analytics (daily, weekly, monthly views), air quality predictions through an integrated machine learning model, and a user-friendly interface distinguishing between admin and regular users.

Additionally, a machine learning model (Random Forest) was trained and integrated into the system, achieving a high R^2 score of 0.987, allowing accurate prediction of AQI based on real-time sensor inputs.

Overall, the project effectively delivered a low-cost, sustainable, and intelligent air quality monitoring solution, aligned with Ashesi University's commitment to innovation, sustainability, and community well-being.

6.2 Recommendations

Based on the experiences gained throughout the project, the following recommendations are made:

- **Expand Deployment Across Campus**: Deploy multiple units at different strategic points (e.g., hostels, lecture halls, sports fields) to build a more detailed air quality heatmap across campus.
- **Public Awareness Integration**: Install real-time displays of AQI in common areas (e.g., cafeterias, libraries) to raise awareness among students and staff.

- **Data-driven Campus Policies**: Use collected data to inform environmental decisions, such as increasing green spaces or regulating vehicle emissions on campus.
- **Maintenance Schedule**: Develop a regular maintenance protocol (e.g., cleaning sensors, checking solar panel output) to ensure continued device accuracy and uptime.

6.3 Future Work

While the current system fulfils its core objectives, future enhancements could further strengthen the system's capabilities:

- **Multi-Pollutant Expansion**: Integrate sensors for additional pollutants such as NO_2, O_3, and CO to provide a more comprehensive air quality profile.
- **Predictive Maintenance Algorithms**: Incorporate machine learning to predict sensor failures or battery degradation based on historical operational data.
- **Mobile App Notifications**: Add push notifications to the mobile app to immediately alert users of poor air quality levels without opening the app.
- **Mesh Networking**: Employ wireless mesh networking (e.g., LoRaWAN or Zigbee) for inter-device communication and reduced dependence on Wi-Fi.
- **Long-Term Performance Study**: Conduct studies on the long-term accuracy and drift of sensors under Ashesi's specific environmental conditions.

6.4 Limitations

Despite the success of the project, a few limitations were observed:

- **Single Location Sampling**: The deployment at only one location may not fully capture spatial variability in air quality across the entire campus.

- **Solar Dependency**: Performance remains dependent on consistent sunlight availability. Extended periods of cloudy weather could affect battery charging and device uptime.
- **Limited Testing Duration**: Although stability testing was done over several hours, fullseasonal or long-term testing (across months) was not performed within the project timeline.

6.5 Summary

In conclusion, the project achieved its major goals and provides a working, scalable foundation for future campus-wide air quality monitoring initiatives. With additional enhancements and broader deployment, the system has the potential to transform Ashesi University into a leader in sustainable environmental monitoring within West Africa.

References

[1] Clean Air Fund, "Ghana," Clean Air Fund. [Online]. Available: https://www.cleanairfund.org/geography/ghana/

[2] World Health Organization, "Urban Health Initiative: Accra story," WHO, 2021. [Online]. Available: https://www.who.int/news-room/facts-in-pictures/detail/urban-health-initiativeaccra-story

[3] O. Ramírez, B. Hernández-Cuellar, and J. D. de la Rosa, "Air quality monitoring on university campuses as a crucial component to move toward sustainable campuses: An overview," Urban Climate, vol. 52, p. 101694, 2023, doi: 10.1016/j.uclim.2023.101694

[4] B. Artíñano et al., "Air quality monitoring in scholar environments," in Proc. 5th Experiment@ Int. Conf., Funchal, Portugal, 2019, pp. 439-443.

[5] H. Cho and Y. Baek, "Design and Implementation of a Smart Air Quality Monitoring and Purifying System for the School Environment," in Proc. IEEE Int. Conf. Consumer Electronics, 2022, pp. 1-4.

[6] N. Gupta et al., "Air Quality Monitoring System for Campus Sustainability," in Proc. Int. Conf. Computational Intelligence and Computing Applications, 2024, pp. 297-302.

[7] S. Esfahani et al., "Smart City Battery Operated IoT Based Indoor Air Quality Monitoring System," in Proc. IEEE Int. Conf., 2020, pp. 1-4.

[8] P. K. Malik et al., "Development of a low-cost IoT device using ESP8266 and Atmega328 for real-time monitoring of Outdoor Air Quality with Alert," in Proc. Int. Conf. Advancement in Electronics & Communication Engineering, 2023, pp. 125-129.

[9] “Reducing air pollution at schools and campuses,” Clarity.io, 2023. https://www.clarity.io/blog/the-importance-of-air-quality-measurement-for-schools-andcampuses

[10] US EPA, “Outdoor Air Quality | US EPA,” US EPA, 2024. https://www.epa.gov/reportenvironment/outdoor-air-quality

[11] C40 Cities Climate Leadership Group and C40 Knowledge Hub, “How to set standards and monitor outdoor air quality,” www.c40knowledgehub.org, 2024. https://www.c40knowledgehub.org/s/article/How-to-set-standards-and-monitor-outdoor-airquality?language=en_US

[12] Cleveland Clinic, “Hypercapnia,” Cleveland Clinic, 2023. https://my.clevelandclinic.org/health/diseases/24808-hypercapnia

[13] Minnesota Pollution Control Agency, “Volatile organic compounds (VOCs),” www.pca.state.mn.us, 2023. https://www.pca.state.mn.us/pollutants-and-contaminants/volatile-organic-compounds-vocs

[14] World Health Organization, “Heat and health,” World Health Organization, 2024. https://www.who.int/news-room/fact-sheets/detail/climate-change-heat-and-health

[15] T. Półbrat, D. Konkol , and M. Korczyński, “Vaporizers and Field Anesthesia Equipment for Free-Ranging Wildlife,” Elsevier eBooks, vol. 35, no. 102076, pp. 177–184, 2021, doi: https://doi.org/10.1016/j.bcab.2021.102076.

[16] Urgent Care, “Effects of Humidity on the Human Body,” UPMC HealthBeat, 2014. https://share.upmc.com/2014/06/effects-humidity-body/

[17] ThermoPro, “Everything You Need to Know About High Humidity | ThermoPro Blog,” Meat Thermometer | Hygrometer | Thermopro, 2020. https://buythermopro.com/high-humidity/

[18] T. Madan, S. Sagar, and D. Virmani, “Air Quality Prediction using Machine Learning Algorithms –A Review,” *2020 2nd International Conference on Advances in Computing, Communication Control and Networking (ICACCCN)*, Dec. 2020, doi: https://doi.org/10.1109/icacccn51052.2020.9362912.

[19] S. Bhattacharya and S. Shahnawaz, “Using Machine Learning to Predict Air Quality Index in New Delhi,” *arXiv (Cornell University)*, Dec. 2021, doi: https://doi.org/10.48550/arxiv.2112.05753.

Bibliography

[20] “ESP32-WROOM Development Board Details, Pinout, Specs,” *Espboards.dev*, 2025. https://www.espboards.dev/esp32/esp32-wroom/#google_vignette)

[21] “Smart sensor solutions,” *sensirion.com*. https://sensirion.com/products/catalog/SPS30

[22] "Adafruit CCS811 Air Quality Sensor," *Adafruit Learning System*.
https://learn.adafruit.com/adafruit-ccs811-air-quality-sensor/overview

[23] Sensirion AG, "SHT30-DIS-F - ±2% Digital humidity and temperature sensor with filter membrane," *Sensirion.com*, 2025. https://sensirion.com/products/catalog/SHT30-DIS-F

[24] Cirkit Design, "How to Use the Micro SD card module," *Cirkitdesigner.com*, 2024. https://docs.cirkitdesigner.com/component/df0473d5-7e00-4218-a637-1470bc1471da/microsd-card-module

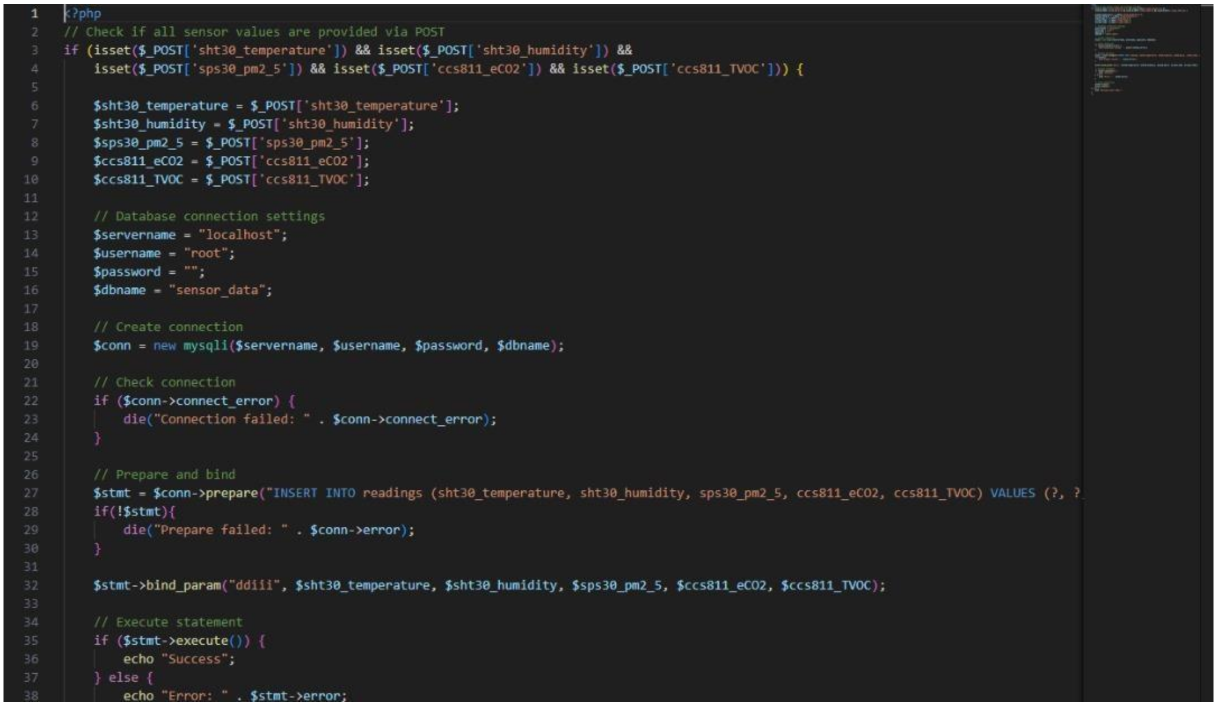
[25] B. Ryan, "Beginners Guide to 12 Volt Solar Panels | Quick Start," *Battery Skills*, 2025.
https://www.batteryskills.com/beginners-guide-to-12-volt-solar-panels/

[26] "The TP4056: Lithium Ion/polymer Battery Charger IC," *Best Microcontroller Projects*.
https://www.best-microcontroller-projects.com/tp4056.html

[27] Cirkit Design, "How to Use the AMS1117," *Cirkitdesigner.com*, 2024.
https://docs.cirkitdesigner.com/component/98b2e3b4-3097-4414-8935-3d3efb0f9eff/ams1117

[28] AirQo, "Bridging the Air Quality Data Gap in Africa," AirQo, https://www.airqo.net/.

Appendix



```php
<?php
// Check if all sensor values are provided via POST
if (isset($_POST['sht30_temperature']) && isset($_POST['sht30_humidity']) &&
    isset($_POST['sps30_pm2_5']) && isset($_POST['ccs811_eCO2']) && isset($_POST['ccs811_TVOC'])) {

    $sht30_temperature = $_POST['sht30_temperature'];
    $sht30_humidity = $_POST['sht30_humidity'];
    $sps30_pm2_5 = $_POST['sps30_pm2_5'];
    $ccs811_eCO2 = $_POST['ccs811_eCO2'];
    $ccs811_TVOC = $_POST['ccs811_TVOC'];

    // Database connection settings
    $servername = "localhost";
    $username = "root";
    $password = "";
    $dbname = "sensor_data";

    // Create connection
    $conn = new mysqli($servername, $username, $password, $dbname);

    // Check connection
    if ($conn->connect_error) {
        die("Connection failed: " . $conn->connect_error);
    }

    // Prepare and bind
    $stmt = $conn->prepare("INSERT INTO readings (sht30_temperature, sht30_humidity, sps30_pm2_5, ccs811_eCO2, ccs811_TVOC) VALUES (?, ?
    if(!$stmt){
        die("Prepare failed: " . $conn->error);
    }

    $stmt->bind_param("ddiii", $sht30_temperature, $sht30_humidity, $sps30_pm2_5, $ccs811_eCO2, $ccs811_TVOC);

    // Execute statement
    if ($stmt->execute()) {
        echo "Success";
    } else {
        echo "Error: " . $stmt->error;
```

Figure A PHP File


```cpp
  // Initialize SHT31
  if (!sht31.begin(0x44)) { // Default I2C address
    Serial.println("Couldn't find SHT31 sensor!");
    while (1) delay(1);
  }

  // Initialize CCS811
  if (!ccs.begin()) {
    Serial.println("Couldn't find CCS811 sensor!");
    while (1) delay(1);
  }

  // Wait for sensor to be ready
  while (!ccs.available()) delay(100);
}

void loop() {
  // Read SHT31 values
  float temp = sht31.readTemperature();
  float humidity = sht31.readHumidity();

  if (!isnan(temp) && !isnan(humidity)) {
    Serial.print("Temp: "); Serial.print(temp); Serial.print(" °C, ");
    Serial.print("Humidity: "); Serial.print(humidity); Serial.println(" %");
  } else {
    Serial.println("Failed to read from SHT31");
  }
```

Figure B Arduino Posting Sketch