POLLUTION MONITORING SYSTEM FOR ENHANCING SUSTAINABILITY OF ENVIRONMENT USING IOT

ECD416 Project Phase II: Report Submitted by

ATHUL K. (Reg. No. KNR20EC029)

BEVIN JOSE V. (Reg. No. KNR20EC030)

MEGHNA T. (Reg. No. IDK20EC031)

SAMRUDH KISHSAN P. M. (Reg. No. KNR20EC073)

Towards the partial fulfillment of the requirement for the award of B. Tech. degree in

Electronics and Communication Engineering

Department of Electronics and Communication Engineering

Government College of Engineering Kannur

Parassinikkadavu (P. O.), Kannur - 670 563

May 2024

GOVERNMENT COLLEGE OF ENGINEERING KANNUR

DEPARTMENT OF
ELECTRONICS AND COMMUNICATION ENGINEERING

CERTIFICATE

Certified that this is a bonafide report of the Final year Project Phase II work done by **ATHUL K. (Reg. No. KNR20EC029), BEVIN JOSE V.(Reg. No. KNR20EC030)**, **MEGHNA T.(Reg.No.IDK20EC031)** and **SAMRUDH KISHSAN P. M. (Reg. No. KNR20EC073)** on the topic **POLLUTION MONITORING SYSTEM FOR ENHANCING SUSTAINABILITY OF ENVIRONMENT USING IOT** , towards the partial fulfillment of the requirements for the award of the Degree of Bachelor of Technology in Electronics and Communication Engineering under APJ Abdul Kalam Technological University during the academic year 2023-2024.

Prof. Nikhil kumar P. P.	**Prof Rohit P.**	**Dr. A. Ranjith Ram**
Project Supervisor	Project Co-ordinator	Professor & HoD
Department of ECE	Department of ECE	Department of ECE

Head of Department
Dept. of Electronics & Comm
Govt. College of Engine
Kannur - 670 563

GOVERNMENT COLLEGE OF ENGINEERING
Dept. of Electronics & Communication Engg.
KANNUR-670 563

DECLARATION

We, the undersigned, hereby solemnly declare that this project report titled **POLLUTION MONITORING SYSTEM FOR ENHANCING SUSTAINABILITY OF ENVIRONMENT USING IOT**, submitted for the partial fulfillment of the requirements for the award of the degree of Bachelor of Technology in Electronics and Communication Engineering from APJ Abdul Kalam Technological University is a bonafide record of our own work carried out under the supervision of **Prof. Nishil kumar P. P.**.

Wherever we have used materials (data, theoretical analysis, and text) from other sources, we have adequately and accurately cited the original sources.

We also declare that this work has not been submitted to any other institution in this University or any other University.

Athul K.

Bevin Jose V.

Meghna T.

Samrudh kishsan p m

ACKNOWLEDGMENTS

We would like to take this opportunity to extend our sincere thanks to people who helped us to make this project possible. This work will be incomplete without mentioning all the people who helped us to make it real.

Firstly, we would like to thank our project guide, Prof Nishil Kumar P.P. Associate Professor in Electronics and Communication Engineering , whose expertise, insights, and unwavering commitment to our academic growth have been instrumental in shap ing the content and direction of this project. Their mentorship and encouragement have significantly enhanced my understanding of the subject matter.

We would also like to thank Prof Rohit P. ,Our project coordinator, for their role in facilitating this project and providing administrative support. Their dedication to the smooth organization of the project has been instrumental in ensuring its success.

We also sincerely thank our parents, friends and all our well-wishers for their help and support without which we couldn't have completed our work smoothly. Above all, we thank almighty for his blessings without which completion of seminar wouldn't be possible.

ABSTRACT

In this project we have developed a sophisticated environmental monitoring setup utilizing an ESP32 micro-controller as its core. The system integrates an array of sensors including the MQ135 sensor for monitoring air quality and the KY038 sound sensor for assessing sound pollution levels. Additionally, it incorporates an MQ7 sensor to detect carbon monoxide concentrations, supplemented by a suction pump for air extraction in the event of heightened CO levels. Further enhancing its capabilities, the system includes a PMS 7003 Particulate Matter Sensor to measure various particulate concentrations in the atmosphere.

Moreover, the integration of an ESP32 CAM enables real-time visualization of the monitored area, providing users with live updates via the ”Airify” Android application. This application serves as a centralized platform for users to access, analyze, and visualize environmental data, empowering them to make informed decisions regarding their surroundings. Furthermore, the system operates sustainably by harnessing solar energy, thereby reducing its environmental footprint and ensuring long-term energy efficiency. Overall, this project offers a comprehensive solution for environmental monitoring, combining advanced sensor technology with user-friendly visualization tools in a sustainable framework.

TABLE OF CONTENTS

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

In the development of our project ,we have designed an advanced environmental monitoring system that seamlessly integrates various sensor to provide a comprehensive assessment of atmospheric air pollution and sound pollution.

At its core, the ESP32 microcontroller acts as the main control unit, managing the operation of multiple sensors. Among these sensors, the MQ135 is particularly notable for its ability to monitor air quality, detecting harmful gases like ammonia and carbon dioxide. This real-time data empowers users to promptly assess and manage air quality concerns.

In addition to air quality monitoring, the project utilizes the KY038 sound sender sensor to monitor sound pollution levels. By measuring ambient noise intensity, the system identifies areas with excessive noise pollution, enabling targeted interventions to mitigate its impact on the environment and public health.

Furthermore, the inclusion of an MQ7 sensor enhances safety by monitoring carbon monoxide (CO) levels. CO is a hazardous gas, and elevated levels can pose serious health risks. To address this, a suction pump is integrated with the MQ7 sensor. If CO levels surpass predefined thresholds, the pump activates to extract air, reducing CO levels and ensuring safety.

Additionally, the project incorporates a PMS 7003 sensor to assess particulate matter concentration in the atmosphere. Particulate matter, composed of fine particles suspended in the air, can have adverse effects on human health and the environment. By monitoring particulate matter levels, the system enables comprehensive assessment of air quality and appropriate pollution mitigation measures.

To facilitate data visualization and access, an ESP32 Cam module is integrated, enabling live visualization of the monitored area via the ”Airifi” Android application. This app acts as a centralized platform where users can access sensor data, visualize the environment, and locate the monitoring system. This heightened situational awareness empowers users to make informed decisions regarding their surroundings.

Moreover, the project operates sustainably by harnessing solar energy as its primary power source. A 15W solar panel powers the system, reducing reliance on conventional energy sources and promoting energy efficiency and environmental sustainability.This project signifies a significant advancement in environmental monitoring technology. By leveraging the ESP32 microcontroller, advanced sensors, and solar energy, it creates a robust system for monitoring air quality, sound pollution, carbon monoxide levels, and particulate matter concentration. Through the ”Airifi” Android application, users can access real-time data, visualize the environment, and make informed decisions to foster a healthier and more sustainable environment..

CHAPTER 2

LITERATURE REVIEW

Environmental monitoring System utilize the MQ135 sensor for precise air quality assessment and the KY038 sensor for effective sound pollution monitoring. The integration of these sensors under the control of an ESP 32 microcontroller has garnered attention, showcasing a robust and cost-effective approach. [1] Furthermore, the incorporation of the ESP8266 WiFi module to transmit data to IoT platforms, particularly Blynk, is recognized for its real-time monitoring capabilities. This synthesis of sensors, microcontrollers, and wireless connectivity underscores a trend towards comprehensive and accessible environmental monitoring solutions, contributing significantly to research in the field.

Integration of ESP32 microcontroller with PMS 7003 sensor facilitates accurate and real-time measurement of particulate matter in the atmosphere. Studies demonstrate the effectiveness of the PMS 7003 sensor and the versatility of the ESP32 microcontroller for PM monitoring. Integrated systems offer scalable solutions with applications in environmental research and public health, while calibration methods ensure measurement accuracy. Overall, this integration holds promise for advancing air quality monitoring efforts and informing evidence-based decision-making for environmental management.

The system utilize the integration of the ESP32 microcontroller with the MQ7 sensor for air pollution monitoring, augmented by a suction pump to address high carbon monoxide levels. Studies highlight the MQ7 sensor's sensitivity in detecting carbon monoxide concentrations swiftly, while the ESP32 microcontroller's versatility facilitates real-time data collection and transmission. Integrating a suction pump into these systems helps mitigate spikes in carbon monoxide pollution. This approach shows promise for efficient air pollution monitoring, with potential implications for public health and environmental management. Motion detection systems with PIR sensors and ESP32 CAM modules have gained prominence for their applications in security and surveillance. [2] PIR sensors, known for their ability to detect infrared radiation emitted by moving objects, serve as reliable triggers for capturing images using ESP32 CAMs. The integration of these components offers a dynamic solution for unauthorized intrusion detection. Researches highlights the efficiency and responsiveness of PIR sensors in identifying motion, coupled with the image capturing capabilities of ESP32 CAMs, providing a comprehensive security framework.This synthesis of sensor technologies signifies a prevailing inclination toward sophisticated and efficient motion-based surveillance systems, contributing significantly to advancements in the realm of intelligent security applications.

The integration of renewable energy, particularly solar cells, to charge ESP-32 based systems and sensors . This trend aligns with the growing demand for sustainable solutions in electronics, emphasizing the effectiveness of solar cells in generating continuous power. [3] The synergy between ESP-32 microcontroller and renewable energy not only aids environmental conservation but also provides a reliable, autonomous power source for remote applications. Studies leads to optimization methods, energy storage, and overall efficiency of solar-powered charging systems, contributing to the advancement of energy efficient and environmentally conscious electronic solutions.

CHAPTER 3

PROPOSED SYSTEM

3.1 Problem Definition

The problem addressed by this system is the lack of a comprehensive environmental monitoring solution capable of monitoring multiple pollutants simultaneously and providing real-time data visualization. Prior to the implementation of this system, existing solutions typically focused on monitoring individual pollutants or lacked integration with visual data. This fragmented approach limited the ability to obtain a holistic understanding of environmental conditions. Additionally, the absence of a sustainable power source posed challenges in ensuring continuous operation. By integrating the ESP32 microcontroller with sensors for monitoring air quality, sound pollution, carbon monoxide levels, and particulate matter, along with a suction pump for CO mitigation and a solar power source, this system offers a unified solution. The system provides real-time data visualization through an Android application named "Airify," empowering users to access comprehensive environmental data, visualize the environment, and monitor the system's location. Thus, this project addresses the need for a holistic, integrated, and sustainable environmental monitoring solution.

3.2 Theory

The project integrates various sensors with the ESP32 microcontroller to create a sophisticated environmental monitoring system, targeting air pollution concerns comprehensively. At its core, the ESP32 micro-controller serves as the central processing unit, orchestrating the operation of multiple sensors and facilitating data transmission to the user interface.

The MQ135 sensor is instrumental in monitoring air quality by detecting the presence of harmful gases such as ammonia, benzene, and carbon dioxide. This sensor operates based on the principle of gas sensing through a chemical reaction, where the change in resistance due to gas exposure is measured and correlated with gas concentration. This real-time data provides crucial insights into the atmospheric composition, enabling users to assess overall air quality in their surroundings and take necessary measures to mitigate pollution.

In addition to air quality monitoring, the KY038 sound sensor tracks sound pollution levels by measuring the intensity of ambient noise. The sensor functions by converting sound waves into electrical signals, which are then processed to quantify noise levels. This capability allows the system to identify areas with excessive noise pollution, enabling targeted interventions to mitigate its impact on the environment and public health.

Moreover, the project incorporates an MQ7 sensor to enhance safety by monitoring carbon monoxide (CO) levels. CO is a highly toxic gas that can pose serious health risks at elevated concentrations. The MQ7 sensor operates based on the principle of electrochemical detection, where the change in conductivity due to CO exposure is measured. To address potential CO hazards, a suction pump is integrated with the MQ7 sensor. If CO levels exceed predefined thresholds, the pump activates to extract air and reduce CO levels, ensuring the safety of individuals in the monitored area.

Additionally, the system incorporates a PMS 7003 particulate matter sensor to assess various particulate matter concentrations in the atmosphere. Particulate matter, consisting of fine particles suspended in the air, can have detrimental effects on human health and the environment. The PMS 7003 sensor operates based on light scattering principles, where particles passing through a laser beam scatter light, and the scattered light intensity is measured to determine particle concentration. By monitoring particulate matter levels, the system enables users to assess air quality more comprehensively and take appropriate measures to mitigate pollution.

To facilitate data visualization and accessibility, the system integrates an ESP32 Cam module, enabling live visualization of the monitored area. This module captures images and streams live video of the surroundings, allowing users to remotely access real-time data through the ”Airify” Android application. Additionally, users can view the system’s location, enhancing situational awareness and enabling targeted interventions in areas with poor air quality or high pollution levels.

Furthermore, the system operates sustainably by harnessing solar energy as its primary power source. With a 15W solar panel, the system reduces its reliance on conventional energy sources, contributing to energy efficiency and environmental sustainability. This integration of renewable energy sources aligns with global efforts to promote sustainable development and mitigate the impacts of climate change.

This project represents a significant advancement in environmental monitoring technology, leveraging the capabilities of the ESP32 microcontroller and advanced sensors to create a comprehensive system for monitoring air quality, sound pollution, carbon monoxide levels, and particulate matter concentration. By providing real-time data visualization and accessibility through the ”Airify” Android application, the system empowers users to make informed decisions regarding their surroundings, ultimately contributing to a healthier and more sustainable environment

3.3 Design

The hardware design of the project involves integrating multiple sensors and components with the ESP32 micro-controller to create a comprehensive environmental monitoring system. Here’s a detailed overview of the hardware components and their connections:

- ESP32 Microcontroller: The ESP32 serves as the central processing unit and is responsible for controlling all sensor operations and data transmission. It communicates with each sensor via GPIO pins and interfaces with the ESP32 CAM module for live visualization.
- MQ135 Air Quality Sensor: The MQ135 sensor is connected to the ESP32 microcontroller through analog pins. It measures the concentration of various harmful gases such as ammonia, benzene, and carbon dioxide in the air.
- KY038 Sound Sensor: The KY038 sensor detects sound pollution levels by measuring the intensity of ambient noise. It is interfaced with the ESP32 microcontroller via analog pins to capture sound data.
- MQ7 Carbon Monoxide Sensor: The MQ7 sensor monitors carbon monoxide (CO) levels in the air. It is connected to the ESP32 microcontroller through analog pins. A suction pump is associated with the MQ7 sensor to extract air if CO levels are high, enhancing safety.
- PMS 7003 Particulate Matter Sensor: The PMS 7003 sensor measures various particulate matter concentrations in the atmosphere. It communicates with the ESP32 microcontroller via digital pins to provide real-time data on air quality.
- ESP32 Cam Module: The ESP32 Cam module is connected to the ESP32 microcontroller via GPIO pins. It captures images and streams live video of the monitored area for visualization purposes in the ”Airify” Android application.
- Solar Panel: A 15W solar panel is utilized as the primary power source for the entire system. It charges a battery or capacitor to ensure continuous operation, reducing reliance on conventional energy sources and promoting sustainability.
- Power Management Circuitry: Power management circuitry is incorporated to regulate the voltage and current supplied by the solar panel and manage power distribution among the components.

By integrating these hardware components and ensuring proper connectivity, the system enables comprehensive monitoring of air quality, sound pollution, carbon monoxide levels, and particulate matter concentration. The live visualization of data through the "Airify" Android application empowers users to make informed decisions regarding their environment and enhances overall environmental awareness

3.3.1 Hardware Design

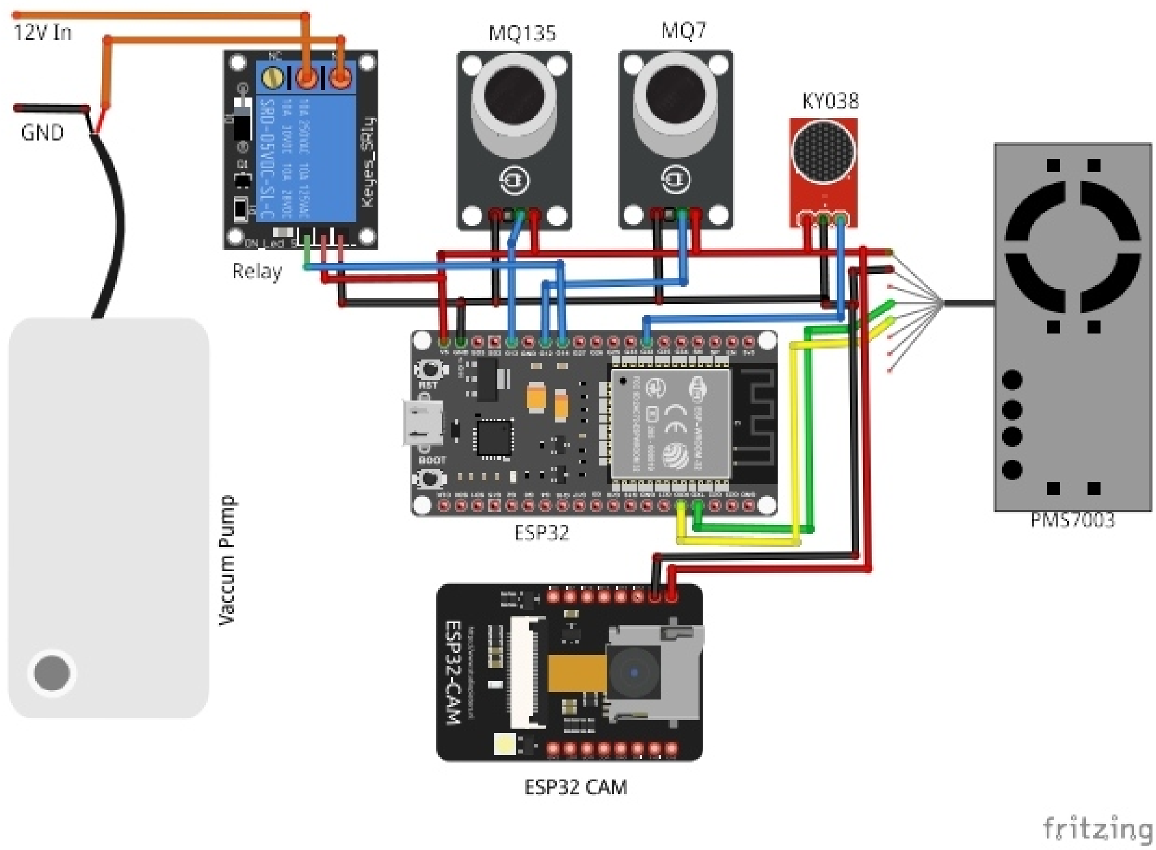

Fig. 3.1. Hardware design of the pollution monitoring system for enhancing sustainability of environment using IoT

3.3.2 ESP32 Microcontroller

The ESP32 is a powerful and cost-effective IoT platform by Espressif Systems, offering a dual-core processor, integrated Wi-Fi and Bluetooth, numerous GPIO pins, and low power consumption. It facilitates multitasking, supports various interfaces, and is suitable for energy-efficient applications. Programming is commonly done in

C++ using Arduino IDE, PlatformIO, or ESP-IDF.

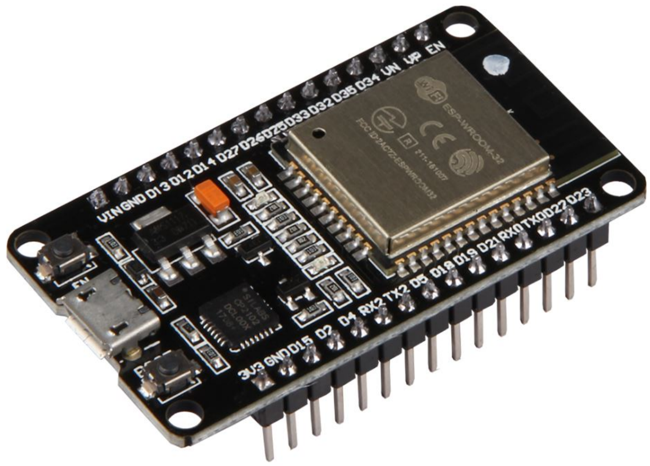

Fig. 3.2. ESP32 Microcontroller

Features

- Processor: ESP32 (24 MHz dual core)
- Flash memory: 4 MB
- Built-in microSD card connector
- PSRAM (pseudo-static random access memory): 8 MB
- Built-in Wi-Fi, Bluetooth, USB to serial converter (CP2104 or CH9102F)
- Built-in Li-ion/Li-Po battery charging circuit: TP4054 chip

Fig 3.2 shows ESP 32 microcontroller.

3.3.3 ESP32-CAM

ESP32-CAM is a low-cost ESP32-based development board with onboard camera, small in size. It is an ideal solution for IoT application, prototypes constructions and DIY projects. The board integrates Wi-Fi, traditional Bluetooth and low power BLE, with 2 high performance 32-bit LX6 CPUs. It adopts 7-stage pipeline

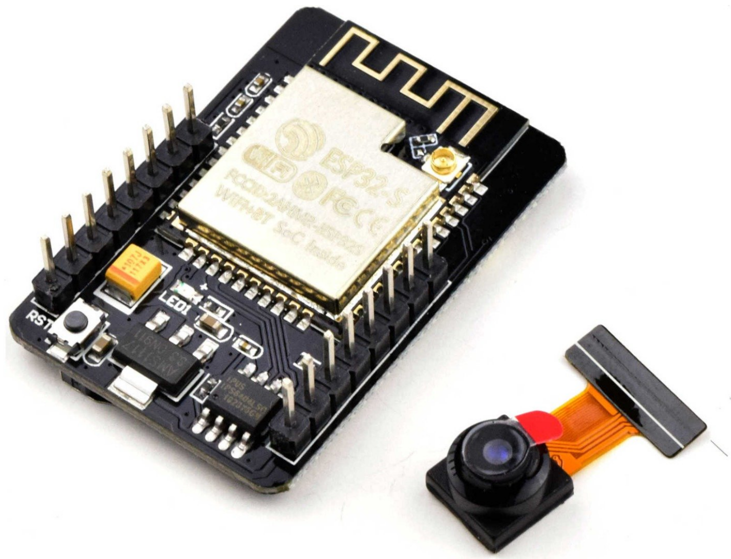

Fig. 3.3. ESP32-CAM

architecture, on-chip sensor, Hall sensor, temperature sensor and so on, and its main frequency adjustment ranges from 80MHz to 240MHz. Fully compliant with Wi-Fi 802.11 and Bluetooth 4.2 standards, it can be used as a master mode to build an independent network controller, or as a slave to other host MCUs to add networking capabilities to existing devices. It is suitable for home smart devices, industrial wireless control, wireless monitoring, QR wireless identification, wireless positioning system signals and other IoT applications. Fig 3.3 shows ESP 32 CAM.

3.3.4 MQ-135 Sensor

The MQ-135 is a gas sensor it used for detecting or sensing harmful gases in the atmosphere. It has wide detecting scope. It gives fast response and also it it high sensitivity sensor. It is simple and long life device. They are used in air quality control equipment for building offices are suitable for detecting of NH3, alcohol, benzene, smoke CO2 etc. Fig 3.4 shows MQ135 Gas sensor.

Feature

- Wide detecting scope
- Fast response and High sensitivity

Fig. 3.4. MQ-135 Gas Sensor

- Stable and long life
- Operating Voltage is +5V
- Detect/Measure NH3, NOx, alcohol, Benzene, smoke,
- CO2, etc.
- Analog output voltage: 0V to 5V

3.3.5 KY038 Sensor

The KY-038 sound sensor is a module designed for detecting sound levels in its environment. It is often used in projects involving sound-activated systems or noise monitoring applications. The sensor utilizes a small microphone that captures variations in air pressure caused by sound waves. The output signal is then processed by the built-in amplifier, making it suitable for interfacing with microcontrollers or other electronic devices. The KY-038 is widely used in hobbyist electronics and prototyping due to its simplicity and ease of use. Fig 3.5 shows KY038 Sound Sensor.

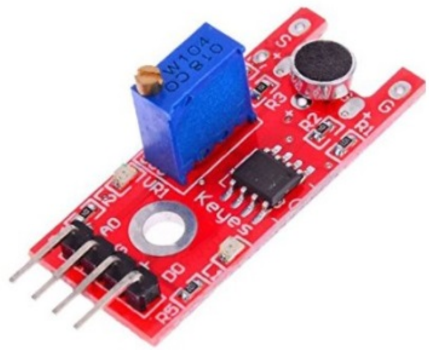

Fig. 3.5. KY038 Sound sensor

Feature:

- Adjustable sensitivity for different sound levels.
- Typically operates at 3.3V to 5V DC.
- Analog output signal proportional to the sound level. Built-in amplifier to enhance the sensor's sensitivity.
- Easily interfaces with microcontroller through analog input pins.
- Detection Range varies based on environmental factors, capable of detecting sound in a moderate range.
- Some variants may include an LED indicator to signal sound detection.
- Commonly used in projects related to sound-activated lighting, alarms, and monitoring systems.
- Compact size, making it suitable for integration into various electronic projects.

3.3.6 MQ7 Sensor

The MQ7 sensor is a crucial component of the environmental monitoring system, tasked with detecting carbon monoxide (CO) levels in the air. This sensor operates on the principle of chemiresistive detection, where changes in CO concentration cause variations in electrical resistance. The MQ7 sensor consists of a tin dioxide (SnO_2) sensing element heated to a specific temperature, typically around 200-400°C. When exposed to CO, the sensing element undergoes a change in conductivity, resulting in a measurable output signal. The sensor's sensitivity to CO makes it an essential tool for identifying potentially hazardous environments and enabling timely interventions to ensure safety. In this project, the MQ7 sensor is integrated with a suction pump mechanism to extract air if CO levels exceed predefined thresholds, further enhancing safety measures. This proactive approach underscores the sensor's importance in safeguarding individuals and communities against the harmful effects of carbon monoxide exposure.

Fig. 3.6. MQ7 Sensor

3.3.7 PMS7003 Sensor

The PMS7003 sensor is a highly sensitive particulate matter (PM) sensor designed to measure various particulate matter concentrations in the atmosphere. It utilizes laser scattering technology to detect particles suspended in the air and pro-

Fig. 3.7. PMS7003 Sensor

vides real-time data on PM1.0, PM2.5, and PM10 concentrations. The sensor consists of a light-emitting diode (LED) that emits laser light and a photodetector that measures the intensity of scattered light. By analyzing the intensity of scattered light, the sensor can determine the concentration of particles in different size ranges. The PMS7003 sensor is compact, lightweight, and offers high accuracy, making it suitable for applications such as air quality monitoring, pollution detection, and environmental research. In this project, the PMS7003 sensor is integrated with the ESP32 microcontroller to provide comprehensive monitoring of particulate matter levels, enabling users to assess air quality and take appropriate measures to mitigate pollution.

3.3.8 Solar Battery Controller

A solar battery controller is a vital component of solar power systems, tasked with managing the charging process of batteries from solar panels. Its primary function is to regulate the voltage and current supplied by the solar panels to ensure efficient and safe charging of batteries.

By preventing overcharging, the controller helps prolong the lifespan of batteries and maintains their optimal performance. Additionally, it offers protection against reverse current flow, over-discharge, and short circuits, safeguarding both the batteries and the entire solar power system. With features like temperature compen-

Fig. 3.8. Solar Battery Controller

sation and maximum power point tracking (MPPT), modern solar battery controllers further enhance the efficiency and reliability of solar energy systems, making them essential for off-grid and hybrid applications.

3.3.9 Solar Cell

A 12V, 15W solar cell refers to a photovoltaic panel that generates electrical power at a voltage of 12 volts and a maximum power output of 15 watts under optimal conditions. This type of solar cell is commonly used in various applications, such as charging batteries, powering small electronic devices, or contributing to off-grid solar systems. Fig 3.9 shows 12 V 15 W solar cell.

- **Feature**
- voltage output:12V
- Power output:15watt
- Material:The solar cell is likely composed of semiconductor materials, often silicon-based, which are efficient in converting sunlight into electrical energy.
- Size and Dimensions:54.8 L*34.8 W*2cm H
- Efficiency:The efficiency of solar cells can vary, but generally in the range of 15 to 20 percentage

Fig. 3.9. Solar Cell

- Weather Resistance: Solar cells are designed to withstand outdoor conditions, including exposure to sunlight, rain, and other environmental factors.

3.3.10 Relay Module

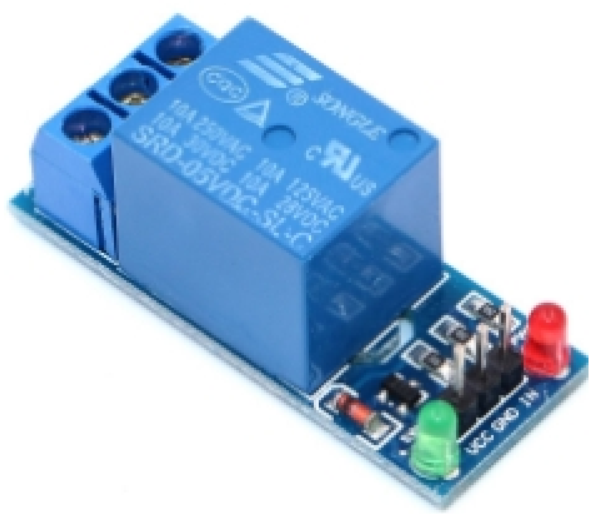

Fig. 3.10. Relay Module

A relay module is an electronic device that enables low-voltage microcontrollers, such as the ESP32, to control higher-voltage circuits. It features a relay, an electromechanical switch, along with interfacing circuitry. When energized by the microcontroller, the relay's switch contacts change state, allowing or interrupting the flow of current in the controlled circuit. Used in various applications like automation and IoT projects, relay modules provide isolation between the control signal and the high-voltage circuit, ensuring safety and protecting the microcontroller. They often include features such as status indicators and flyback diodes for voltage spike suppression.

3.3.11 Suction Pump

A 12V suction pump with an RS555 motor is commonly used for various air suction applications due to its efficiency and reliability. Here are some clear details about this setup:

Fig. 3.11. Suction Pump

- Power Supply: The pump operates using a 12V power supply, making it suitable for a wide range of applications, including automotive, industrial, and consumer electronics.

- RS555 Motor: The RS555 motor is a brushed DC motor known for its durability and torque. It provides the necessary power to drive the suction pump, creating suction to draw air into the system.
- Airflow: The pump generates airflow by creating a pressure differential between its inlet and outlet. This airflow is used to suction air from the surrounding environment into the system.
- Compact Design: These pumps are often compact in size, making them suitable for installations where space is limited. Their small form factor allows for versatile placement in various devices and systems.
- Efficiency: The RS555 motor is known for its efficiency, consuming relatively low power while delivering sufficient suction force. This makes the setup cost-effective to operate over extended periods.
- Applications: Common applications of 12V suction pumps with RS555 motors include air sampling devices, vacuum sealers, pneumatic systems, and air filtration systems.
- Customization: Depending on the specific application requirements, these pumps can be customized with different suction capacities, materials, and mounting options.

Overall, a 12V suction pump with an RS555 motor provides a reliable and efficient solution for air suction applications, offering versatility, durability, and cost-effectiveness.

3.4 Block Diagram

The system works as follows:

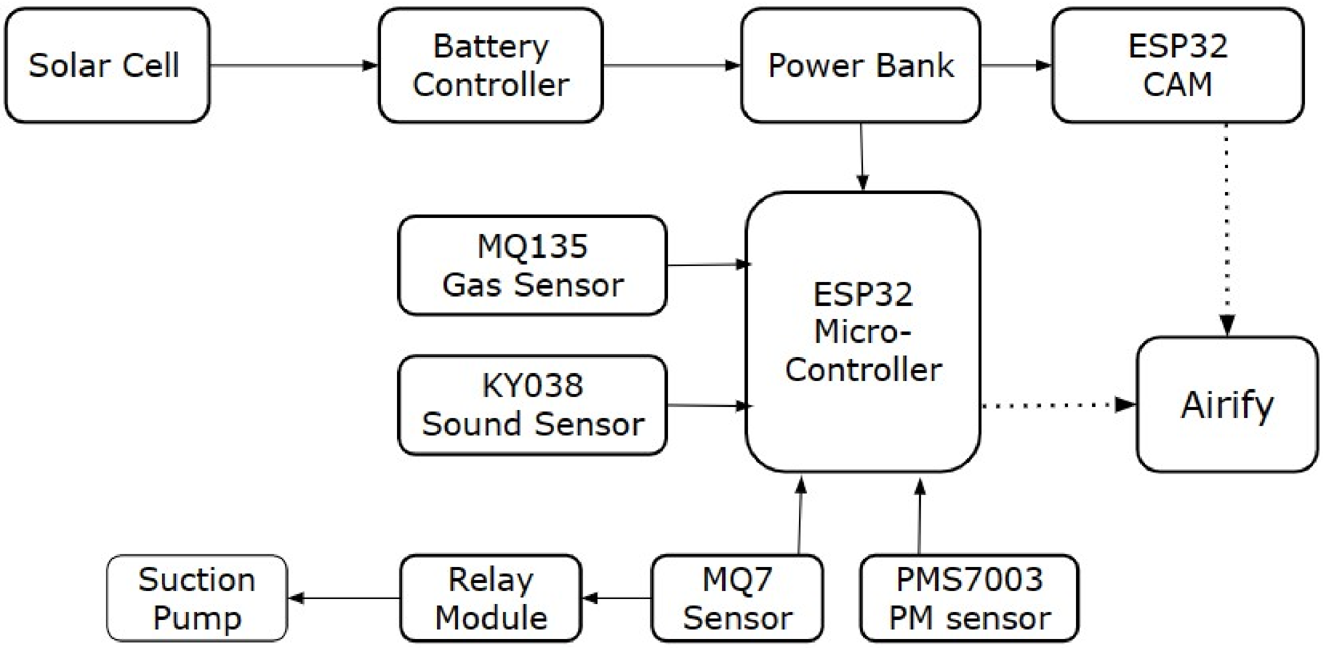

Fig. 3.12. Block diagram of pollution monitoring system for enhancing sustainability of evironment using IoT

- The solar cell converts sunlight into electricity to power the system.
- The solar battery controller converts the solar cell's voltage to a lower voltage that is compatible with the other components.
- The Battery stores excess energy generated by the solar cell for use when sunlight is not available.
- The ESP32 microcontroller reads data from the MQ135 gas sensor, KY038 sound sensor, MQ7 gas sensor,PMS7003 Particulate Matter sensor
- The ESP32 microcontroller displays the air quality and sound levels ,Carbon Monoxide level,Particulate matter level on the Airify android app .
- If the MQ7 detects Carbon Monoxide gases beyond a particular level it will activate the suction pump using a relay module to lower its concentration in that particular area.
- The ESP32 CAM helps in live visulaization of that particular area and this is displayed through Airify android app. Fig 3.11 shows Block Diagram of the

system.

3.5 Software design

3.5.1 App development

An android app is developed using kodular for real time monitoring of air pollution and other features like camera, location, pump activation/deactivation. Kodular is a drag-and-drop app development platform that allows users to create Android apps without needing to write code. Here's a brief description of Kodular and the steps involved in app development using it:

Kodular provides a visual interface where users can design and develop Android applications by assembling components and configuring their behavior using a block-based programming language similar to Scratch. It offers a range of pre-built components and features, enabling users to create a variety of apps, from simple utilities to more complex applications.

Steps involved in app development using Kodular:

1. Sign Up/Login: Visit the Kodular website and sign up for an account or log in if you already have one.

2. Create a New Project: After logging in, create a new project by clicking on the "Create project" button or similar option.

3. Design Interface: Use Kodular's drag-and-drop interface to design the user interface of your app. Add components such as buttons, text boxes, labels, images, etc., to create the desired layout.

4. Configure Components: Once you've added components, configure their properties through the properties panel. Set properties such as text, color, size, visibility, etc., according to your app's requirements.

5. Add Functionality: Use Kodular's blocks editor to add functionality to your

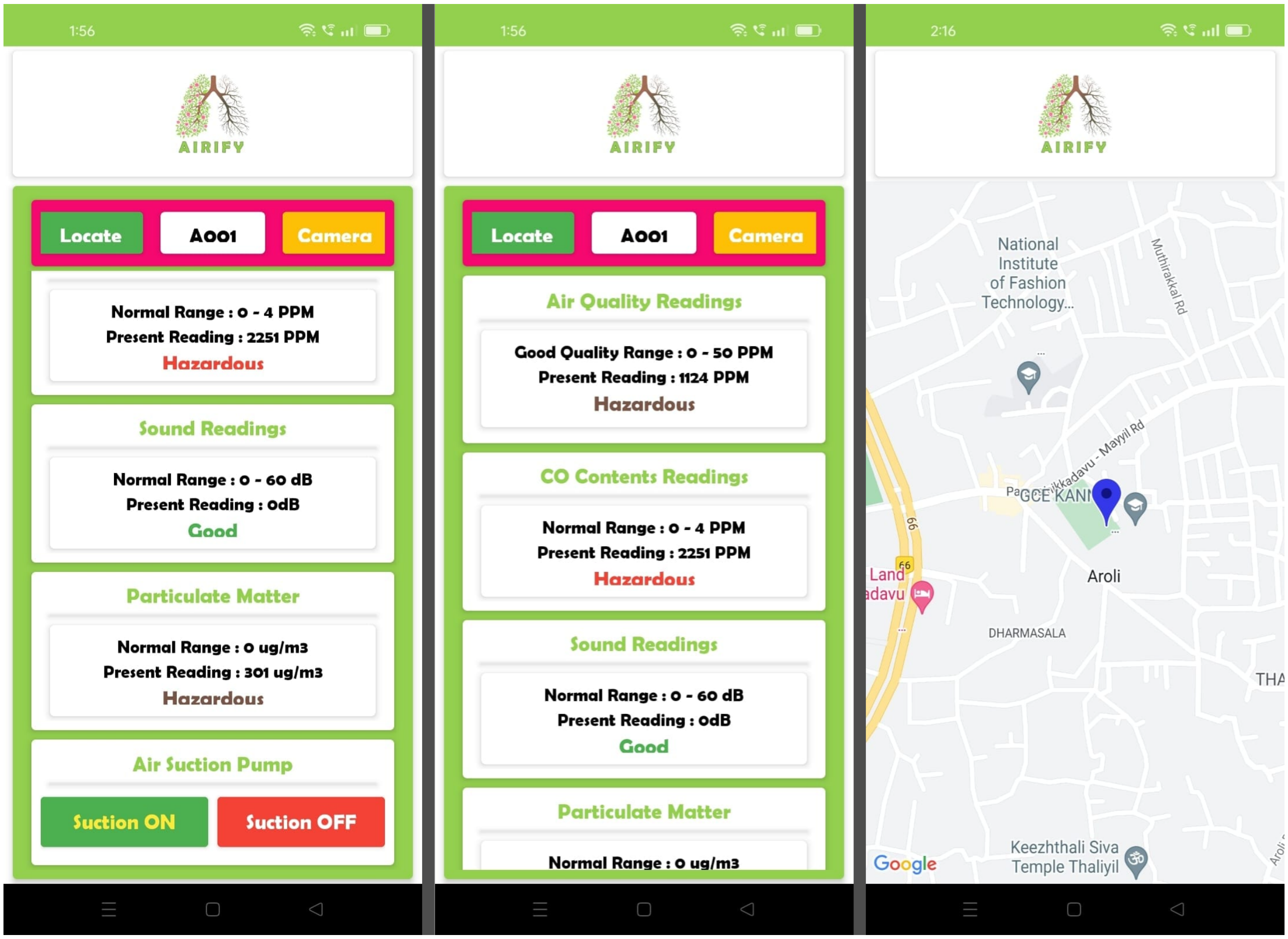

Fig. 3.13. App User Interface

app. Drag and connect blocks to define the behavior of components, handle user interactions, and implement app logic. Kodular provides a wide range of blocks for various purposes, including user input, data handling, device features, etc.

6. Test Your App: Use Kodular's built-in companion app or QR code to test your app directly on your Android device. Make sure to test all functionalities and interactions to ensure your app works as expected.

7. Refine and Iterate: Iterate on your app design and functionality based on testing feedback and user experience. Refine the user interface, optimize performance, and fix any bugs or issues.

3.5.2 Firebase

Firebase is a comprehensive platform developed by Google that provides various services for building mobile and web applications. One of its key features is the Firebase Realtime Database, which is a cloud-hosted database that allows developers to store and synchronize data between clients in real-time.

Firebase can be utilized in an app for air pollution monitoring, specifically using an ESP32 microcontroller as follows:

3.5.2.1 Firebase Realtime Database

- Data Storage: Realtime Database can store the air pollution data collected by the ESP32 microcontroller. This data can include parameters such as air quality index (AQI), pollutant concentrations (e.g., PM2.5, PM10, CO, etc.),sound intensity and location information.
- Real-time Updates: The Firebase Realtime Database enables real-time synchronization of data between the ESP32 microcontroller and the mobile app. As the ESP32 collects air pollution data, it can push updates to the Firebase database, and the mobile app can receive these updates in real-time, providing users with live data.
- Data Access: The Firebase SDKs (Software Development Kits) for various platforms (Android, iOS, web) allow easy integration with the database. The mobile app can use Firebase SDK to read and write data to the database, as well as listen for real-time updates.

3.5.2.2 Integration with ESP32 Microcontroller

- ESP32 Data Collection: The ESP32 microcontroller can be equipped with sensors to measure air quality parameters such as PM2.5, PM10, CO. These sensors can periodically collect data from the environment.

- Firebase API Integration: The ESP32 can be programmed to use Firebase APIs to push the collected air pollution data to the Firebase Realtime Database. This requires connecting the ESP32 to the internet, either through Wi-Fi or other means.

- Real-time Data Transmission: Using Firebase's real-time capabilities, the ESP32 can push data updates to the Firebase database as soon as new data is collected. This ensures that the app receives live updates on air pollution levels.

3.5.3 Arduino IDE

3.5.3.1 Firebase integration

1. Set Up Firebase Project: Create a new Firebase project in the Firebase console. Configure Firebase Realtime Database or Firestore as per your requirements.

2. Install Firebase Arduino Library: In the Arduino IDE, navigate to Sketch ->Include Library ->Manage Libraries. Search for "Firebase" and install the Firebase Arduino library. This library allows you to interact with Firebase Realtime Database or Firestore from your ESP32.

3. ESP32 Data Transmission to Firebase: Write Arduino code to connect your ESP32 to Firebase. Utilize the Firebase Arduino library to send sensor data to Firebase's Realtime Database or Firestore. Ensure that your ESP32 is connected to the internet (e.g., using Wi-Fi) and properly configured to communicate with Firebase.

4. Testing and Debugging: Test the integration thoroughly to ensure that sensor data is being transmitted from the ESP32 to Firebase correctly. Debug any issues that arise during testing, such as connectivity problems or data formatting errors.

5. User Authentication: If you implemented Firebase Authentication, handle user authentication in your Arduino code if necessary. Authenticate your ESP32

device with Firebase using appropriate authentication methods (e.g., OAuth, service account credentials).

3.5.3.2 Sensors integration

- KY-038 Sound Sensor: To set up the KY-038 Sound Sensor, you do not need a specific library for basic functionality. Begin by defining the analog pin (soundSensorPin) to which the KY-038 is connected within your Arduino sketch, such as const int soundSensorPin = A0;. Use the analogRead() function to retrieve the raw analog value from the sensor, storing the result in a variable (int soundValue = analogRead(soundSensorPin);). Optionally, if you wish to convert the analog value to decibel (dB), you can apply calibration data using the map() function. For example, you can convert the analog value (soundValue) ranging from 0 to 1023 to a corresponding dB value ranging from 30 to 120 dB, adjusting the minimum and maximum dB values as needed for calibration (float dBValue = map(soundValue, 0, 1023, 30, 120);). Customize the pin assignment (A0 in this case) based on your specific hardware configuration, and incorporate these steps into your code to capture and process sound sensor data effectively.
- MQ-135 Gas Sensor: To integrate the MQ-135 Gas Sensor with an Arduino or ESP32 using the "MQ135" library, first ensure that you have installed the library via the Arduino Library Manager. Once the library is installed, include it in your Arduino sketch using #include <MQ135.h>. In your setup, create an instance of the MQ135 class and specify the analog pin (e.g., A1) to which the MQ-135 sensor is connected. For example, MQ135 mq135(A1); initializes an instance named mq135 for the sensor connected to analog pin A1. Within your loop() function or at the appropriate location, use analogRead() to obtain the raw analog value from the MQ-135 sensor: int mq135Value = analogRead(A1);. Then, use the getPPM() method of the MQ135 class to convert the analog reading to parts per million (PPM) representing the gas concentration: float ppmValue = mq135.getPPM(mq135Value);. Customize the analog pin assignment (A1) to

match the pin to which your MQ-135 sensor is connected.

- MQ-7 Gas Sensor: For the MQ-7 Gas Sensor, no specific library is required for basic analog reading. To set up the sensor, begin by defining the analog (mq7AnalogPin) and digital (mq7DigitalPin) pins connected to the MQ-7 within your Arduino sketch, such as const int mq7AnalogPin = A2; and const int mq7DigitalPin = 13;. Utilize analogRead(mq7AnalogPin) to retrieve the analog value from the MQ-7 sensor, and use digitalRead(mq7DigitalPin) to obtain the digital status, such as the heating status, from the MQ-7 sensor. Customize these pin assignments based on your specific hardware setup, and integrate these functions into your code to effectively capture and process sensor data.

- PMS7003 Particulate Matter Sensor: To set up the PMS7003 Particulate Matter Sensor, begin by including the ”SoftwareSerial” library in your Arduino sketch to enable serial communication. Define a SoftwareSerial instance named pmsSerial with pin assignments for RX (receiving) and TX (transmitting), where pmsSerial(16, -1) indicates that only the RX pin (pin 16) is used for communication. In the setup() function, initialize serial communication with both the built-in serial (Serial.begin(9600)) for debugging output and the pmsSerial instance (pmsSerial.begin(9600)) for communication with the PMS7003 sensor. In the loop() function, continuously check if data is available from the PMS7003 sensor (pmsSerial.available() ¿ 0). If data is present, read and print the received data character by character (Serial.print(char(pmsSerial.read()))) until all available data is processed. After printing, insert a new line (Serial.println()) for readability. Use delay(1000) or adjust the delay as needed to control the frequency of data retrieval from the sensor. Customize the pin assignments and baud rate (9600 in this example) based on your specific hardware configuration and sensor setup requirements

3.5.3.3 ESP32 CAM Integration

1. Install ESP32 Board Support: Open the Arduino IDE and go to File >Preferences. In the ”Additional Boards Manager URLs” field, add the following URL: https://dl.espressif.com/dl/package_esp32_index.json Click ”OK” to close the Preferences window. Now, go to Tools >Board >Boards Manager. Search for ”esp32” and install the ”esp32” board package by Espressif Systems.

2. Select Board and Port: After installing the ESP32 board package, select the appropriate ESP32 board variant (e.g., ”ESP32 Wrover Module”) from Tools >Board. Choose the correct port for your ESP32 board under Tools >Port.

3. Install Camera Libraries: To work with the ESP32-CAM module, you’ll need to install the necessary libraries. Look for libraries that support the OV2640 camera on the ESP32 platform.

4. Upload Example Sketch: Utilize example sketches provided by the libraries to set up a basic camera streaming project. Look for examples that demonstrate capturing images or streaming video over HTTP.

5. Access Camera Stream: Once the example sketch is uploaded to your ESP32 board, monitor the Serial Monitor output to obtain the IP address of your board on the local network. Open a web browser and enter the IP address to access the camera stream from the ESP32-CAM module.

3.6 Results

1. The project’s output data, encompassing readings from the ESP32 microcontroller, MQ135 gas sensor, KY038 sound sensor, MQ7 gas sensor, and PMS7003 particulate matter sensor, is efficiently stored and managed within the Firebase database. This centralized repository not only aggregates the diverse environmental data but also includes controls for the suction pump, offering users the capability to remotely manage air quality interventions. Moreover, Firebase

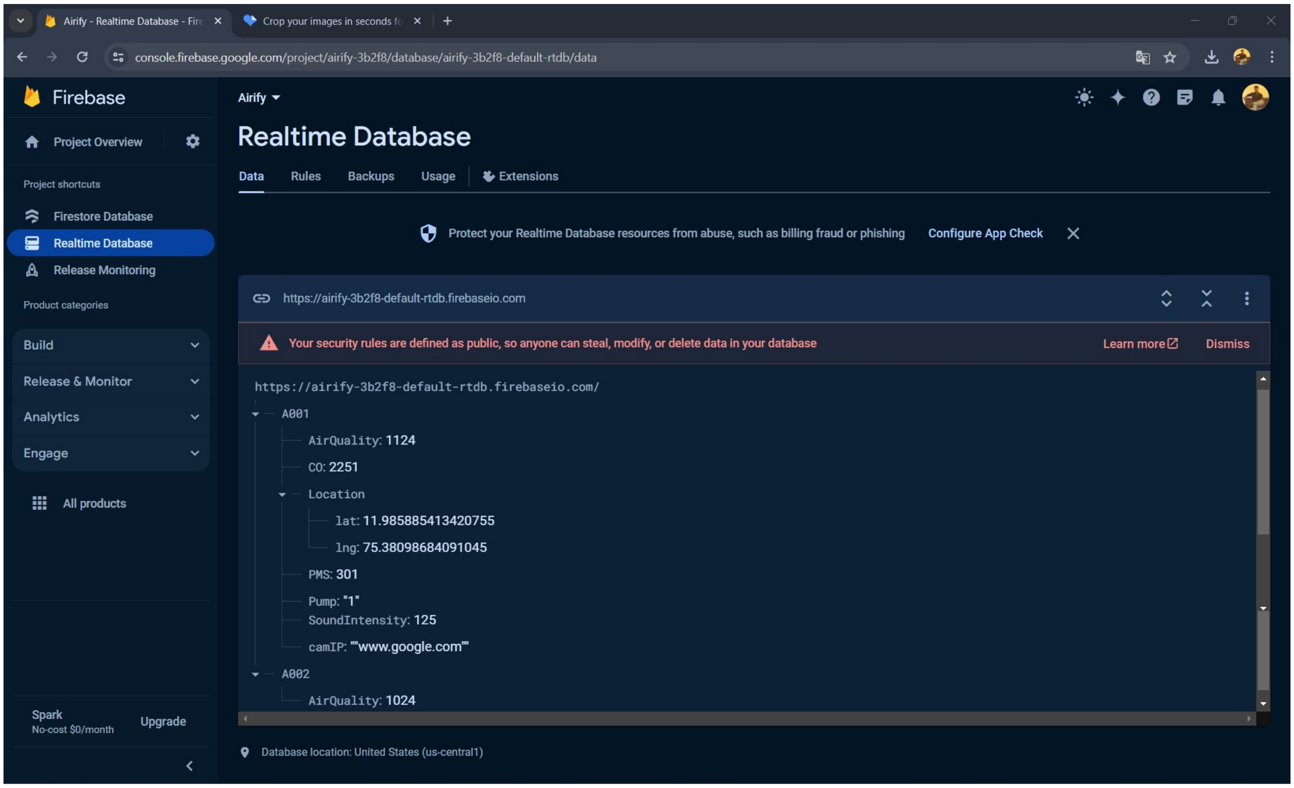

Fig. 3.14. Firebase Output

stores the longitude and latitude coordinates of the system's installation location, providing users with precise geospatial information for context-aware analysis. Through seamless Firebase integration, users gain access to real-time environmental insights, empowering them to make informed decisions regarding pollution mitigation strategies. By enabling remote access to environmental data and control functionalities, the Firebase platform facilitates proactive environmental monitoring, supporting sustainable practices and fostering a healthier ecosystem.

2. The Airify Android app is a comprehensive tool designed to empower users with accurate environmental insights. By leveraging the ESP32 microcontroller along with various sensors like the MQ135 gas sensor, KY038 sound sensor, MQ7 gas sensor, and PMS7003 particulate matter sensor, it provides calibrated data on multiple fronts.

   Users gain access to real-time readings on crucial parameters such as air quality, sound pollution levels, carbon monoxide concentrations, and particulate matter concentration directly through the app interface. This real-time data allows users to stay informed about their immediate surroundings, empowering them to make informed decisions about their environment. Users can control the suction pump for air quality improvements, providing an active means to mitigate pollution levels. Additionally, precise longitude and latitude coordinates of the monitoring system's location are displayed, enhancing the utility of the data for geographic analysis and tracking.

One of the standout features of the Airify app is its integration with the ESP32 Cam module, enabling users to visualize live video streams. This feature enhances situational awareness and make informed decisions based on real-time visual data.

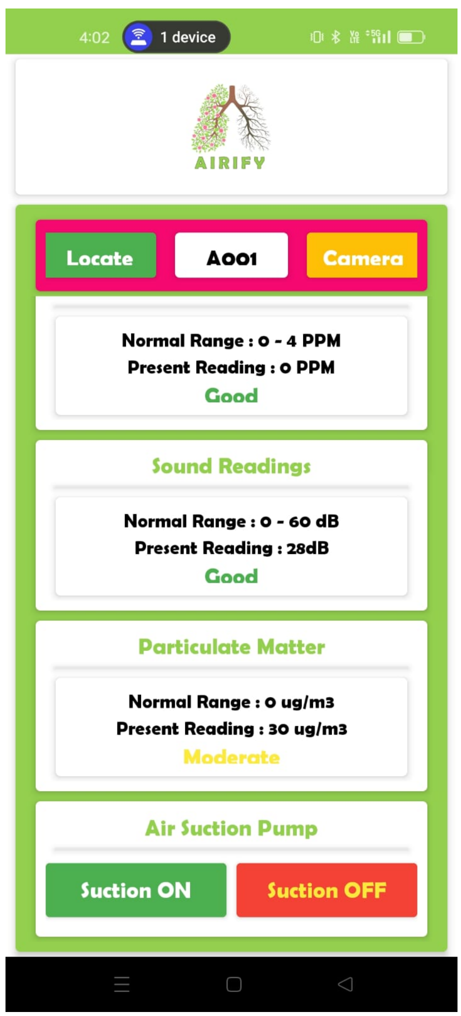

Fig. 3.15. Airify App

CHAPTER 4

THE FINAL PRODUCT

Fig. 4.1. Final product

- ESP32 Microcontroller: The ESP32 microcontroller serves as the central processing unit, responsible for interfacing with and controlling the various sensors and components of the system. It offers built-in Wi-Fi and Bluetooth connectivity, making it suitable for IoT applications.

- MQ135 Air Quality Sensor:The MQ135 sensor detects a wide range of gases, including ammonia, benzene, and carbon dioxide, allowing for comprehensive air quality monitoring. It provides analog output proportional to the concentration of gases detected.

- KY038 Sound Sensor:The KY038 sensor measures ambient noise levels, enabling the detection of sound pollution in the environment. It typically outputs a digital signal based on the intensity of sound detected.

- MQ7 Carbon Monoxide Sensor: The MQ7 sensor specializes in detecting carbon monoxide (CO) levels in the air. It provides analog output, which varies based on the concentration of CO present.

- Suction Pump:The suction pump is associated with the MQ7 sensor and is activated when carbon monoxide levels exceed predefined thresholds. It helps mitigate high CO levels by extracting air from the environment.

- PMS7003 Particulate Matter Sensor:The PMS7003 sensor measures the concentration of particulate matter in the atmosphere, including fine particles that can be harmful to health. It typically provides digital output with particle count and size information.

- ESP32 Cam:The ESP32 Cam module captures live video of the monitored area, providing visual feedback to users. It interfaces with the ESP32 microcontroller and transmits video data for remote viewing via the ”Airifi” Android application.

- ”Airifi” Android Application:The ”Airifi” application serves as the user interface for accessing data collected by the sensors and viewing the live camera feed. It allows users to visualize environmental conditions, track air quality parameters, and monitor the system’s location.

CHAPTER 5

CONCLUSION AND FUTURE SCOPE

5.1 Conclusion

This pollution monitoring system marks a significant milestone in environmental monitoring, showcasing a sophisticated integration of the ESP32 microcontroller with a diverse suite of sensors for comprehensive environmental analysis. The project harnesses the capabilities of the MQ135 sensor to monitor air quality by detecting pollutants such as ammonia, benzene, and carbon dioxide, providing users with real-time insights into atmospheric composition. Simultaneously, the KY038 sound sender sensor tracks sound pollution levels, enabling the identification of areas with excessive noise pollution and facilitating targeted interventions to mitigate its impact.

- Moreover, the integration of the MQ7 sensor addresses carbon monoxide (CO) levels, a critical safety concern. With the ability to detect CO, a highly toxic gas, the system incorporates a suction pump to extract air and reduce CO levels in the event of elevated concentrations, thereby enhancing safety measures and safeguarding individuals' health.
- In parallel, the PMS7003 particulate matter sensor offers insights into various particulate matter concentrations in the atmosphere, contributing to a more comprehensive assessment of air quality. This multi-faceted approach ensures that the system provides holistic and accurate data on environmental parameters.
- Furthermore, the inclusion of the ESP32 Cam module enables live visualization of the monitored area, offering users real-time situational awareness. By

streaming live video footage, the system enhances visibility and allows for immediate response to environmental changes or anomalies.

- Facilitating seamless access to data and control functionalities, the Android application ”Airifi” serves as a user-friendly interface. Users can effortlessly access real-time data, visualize environmental conditions, and pinpoint the system’s location, empowering them to make informed decisions about their surroundings.

- Crucially, the project’s reliance on solar energy for power underscores its commitment to sustainability and environmental stewardship. By utilizing renewable energy sources, the system reduces its carbon footprint and promotes energy efficiency, aligning with global efforts to combat climate change and foster sustainable development.This project offers a scalable and adaptable solution for environmental monitoring, empowering users to make informed decisions and take proactive measures to promote a healthier and safer environment for all.

5.2 Future Scope

- Data Analysis and Predictive Modeling: Implement advanced data analysis techniques and machine learning algorithms to analyze the collected data more effectively. This could involve predicting air quality trends, identifying patterns, and generating insights to support decision-making processes.

- Integration with Smart Devices: Integrate the environmental monitoring system with smart home devices or IoT platforms to enable automated responses based on environmental conditions. For example, automatically adjusting ventilation systems or air purifiers in response to changes in air quality parameters.

- Crowdsourced Data Collection: Allow users to contribute data from their own monitoring devices to create a larger network of environmental data. This crowdsourced approach could provide more comprehensive coverage and insights into air quality and pollution levels across different locations.

- Enhanced User Interface:Continuously improve the user interface of the ”Airifi” Android application to enhance user experience and accessibility. Incorporate features such as customizable dashboards, alerts for critical environmental conditions, and historical data visualization.

- Remote Control and Automation: Enable remote control and automation of the suction pump and other system components through the ”Airifi” app. This would allow users to take immediate action in response to high carbon monoxide levels or other environmental hazards.

- Integration with Public Health Systems: Collaborate with public health authorities to integrate the environmental monitoring data into public health systems. This could support initiatives related to public health awareness, policy-making, and urban planning.

- Expansion to Other Environmental Parameters: Explore the addition of sensors to monitor other environmental parameters such as temperature, humidity, or UV radiation. This would provide a more comprehensive understanding of the overall environmental conditions.

- Community Engagement and Education: Develop educational materials and outreach programs to raise awareness about environmental issues and the importance of monitoring air quality. Encourage community involvement in environmental stewardship efforts.

REFERENCES

[1] A. Sonawane, *"Iot based air and sound pollution monitoring system,"* International Journal for Modern Trends in Science and Technology, vol. 7, july 2021.

[2] M. JAYASREE, *"Esp32- cam motion detector with photo capture using pir sensor,"* IJARIIE, vol. 8, April 2022.

[3] P. SHEKAR, *"Iot based solar energy monitoring,"* IJNIET, vol. 8, june 2022. 35

[4] S. Kirešová and M. Guzan, "Measurement of particulate matter: Principles and options of measurement at present," Department of Theoretical and Industrial Electrical Engineering, Faculty of Electrical Engineering and Informatics, Technical University of Košice, Košice, Slovak Republic.

[5] D. Wang, D. P. Agrawal, W. Toruksa, C. Chaiwatpongsakorn, M. Lu, and T. C. Keener, "Monitoring ambient air quality with carbon monoxide sensor-based wireless network," *Communications of the ACM*, vol. 53, no. 5, pp. xx-xx, May 2010.

APPENDIX

DATA SHEETS

- KY 038 Sound sensor : https://pdf1.alldatasheet.com/datasheetpdf/view/1138845/ETC2/KY-038.html
- PMS7003 PM sensor : https://download.kamami.pl/p564008-PMS700320series20data
- MQ135 gas sensor : https://www.alldatasheet.com/datasheet-pdf/pdf/1307647/WINSEN/MQ135.html
- MQ7 sensor : https://www.alldatasheet.com/view.jsp?Searchword=Mq7
- ESP 32 : https://www.alldatasheet.com/datasheet-pdf/pdf/1148023/ESPRESSIF/ESP32.html
- ESP32 CAM: https://www.handsontec.com/dataspecs/module/ESP32-CAM.pdf
- Solar Charge Controller: https://shopdelta.eu/solar-charge-controller-scc-30a-pwm-lcdl2p17419
- Relay Module: https://components101.com/switches/5v-single-channel-relay-module-pinout-features-applications-working-datasheet