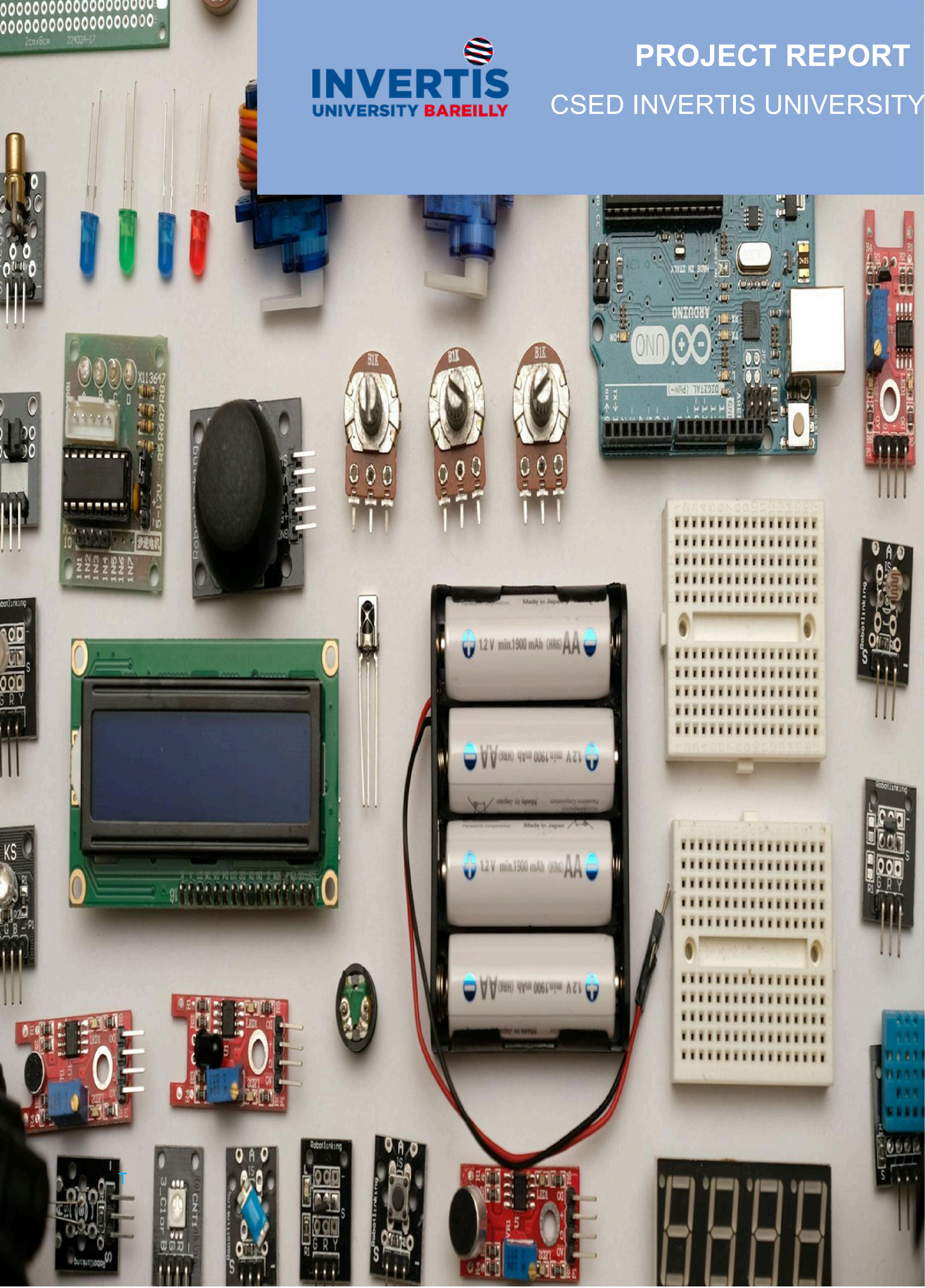
INVERTIS
UNIVERSITY BAREILLY

PROJECT REPORT

CSED INVERTIS UNIVERSITY



CENTER FOR SKILL AND ENTREPRENEURSHIP DEVELOPMENT

(CSED)

INDUSTRIAL INTERNET OF THINGS

PROGRAM CODE: IIOT-2

PROGRAM NAME:

FINDUSTRIAL COMMUNICATION PROTOCOLS AND CONNECTIVITY

PROJECT NAME: SMART INDOOR FARMING MONITORING

MENTOR NAME: MR. RAHUL CHAPLE

Team Code: 2AICCT2

Team Lead: Aditya Patel

Team Members:

Dev Saraswat

Ankush Singh

Akshay Yadav

Nitesh Kumar Gangwar

Shruti Yadav

Divyanshi

Satvik Verma

Shourya Singh

Lokesh Patwal

Anurag Bharti

Arshpreet Kaur

Pranjal Bisht

Prateek Pal

Shreya Rathore

INDEX

1. Introduction (05)
2. Need of Project (07)
3. Problem Statement (09)
4. Objective of Project (10)
5. Methodology (11)
6. Literature Review (13)
7. Explanation of Project (Working) (15)
8. Block Diagram (17)
9. Flowchart (18)
10. Circuit Diagram (19)
11. Program (20)
12. Result & Discussions (25)
13. Photographs of Project (27)
14. Future Scope (28)
15. Conclusion (30)
16. References (32)

1.INTRODUCTION

As the world faces growing challenges like climate change, urbanization, and food insecurity, traditional farming methods are no longer enough to keep up with demand. That's where technology steps in. This project focuses on creating a Smart Indoor Environment Farming Monitoring System using Industrial Internet of Things (IIoT) technology—an approach that blends agriculture with smart, connected devices to make indoor farming more efficient and reliable.

Indoor farming, also known as controlled-environment agriculture (CEA), lets us grow crops in enclosed spaces like greenhouses or vertical farms. By controlling factors like temperature, humidity, soil moisture, light, and CO_2, farmers can grow healthier plants year-round. But manually managing all these variables can be time-consuming and error-prone. Our solution is to automate this process using IIoT sensors and devices that monitor the environment in real time, send data to a central system, and even trigger automated actions—like turning on a fan or a humidifier when needed.

The core hardware components include are :

DHT11(for measuring temperature and humidity,)

SGP30: a sensor that monitors indoor air quality, including CO_2 and total volatile organic compounds (TVOCs)

ESP32: a powerful microcontroller with built-in Wi-Fi and Bluetooth, which collects sensor data and transmits it to a cloud or local server for real-time monitoring and control.

Together, these components allow the system to automatically track environmental conditions, send alerts or trigger actions (like alerting alarms when temperature or humidity execeeds threshold parameters),Be accessed remotely through a mobile or web interface, Log data for analysis and better decision-making.

We are mainly focusing on monitoring the cultivation of mushroom as it is a cultivated in a manual or human controlled and monitored environment. We monitor it throughout the process for increasing our output and make it edible because it become poisonous if we cultivate in external environment.

This project addresses the limitations of manual monitoring systems and introduces a smart, automated, and reliable method that empowers farmers to grow mushrooms efficiently, even in compact or urban indoor setups.

2.Need of Project

The global demand for food is increasing rapidly, while agricultural land and natural resources are becoming more limited. Traditional farming methods are heavily dependent on external environmental factors, which are becoming increasingly unpredictable due to climate change. Indoor farming offers a promising solution, but maintaining the ideal conditions for plant growth manually is difficult, time-consuming, and inefficient.

Mushroom cultivation is highly sensitive to environmental conditions such as temperature, humidity, CO_2 levels, and air quality. Even small changes in these parameters can lead to poor yield, contamination, or complete crop failure. Traditionally, farmers and cultivators have relied on manual methods to monitor these conditions, which are often inaccurate, time-consuming, and inefficient.

In today's fast-paced world, there is a strong need for automation and real-time monitoring to:

1. Ensure consistent growing conditions for mushrooms.
2. Reduce human error and labour dependency.
3. Improve yield and crop quality by maintaining optimal environmental parameters.
4. Get instant alerts when conditions go beyond safe limits.
5. Collect historical data to analyze patterns and optimize future cultivation cycles.

By using a Smart Indoor Environment Monitoring System with IIoT, we can provide:

Accurate, real-time data using sensors like DHT11 (temperature & humidity) and SGP30 (CO_2 & air quality).

Cloud-based remote monitoring via platforms like Thingsboard. Scalability and customization for different crops or room sizes. A sustainable solution that supports precision farming and resource optimization.

3.Problem Statement

Traditional indoor farming relies heavily on manual monitoring and adjustment of environmental conditions such as temperature, humidity, and air quality. This approach is often inefficient, labour-intensive, and prone to human error, which can lead to inconsistent crop growth, resource wastage, and increased operational costs.

There is currently a lack of affordable, automated systems that can continuously monitor and manage these key environmental parameters in real-time. Farmers and growers need a reliable solution that not only provides accurate data but also allows for remote access and control. This project aims to solve this problem by developing a Smart Indoor Environment Farming Monitoring System using IIoT hardware components such as the DHT11 (temperature and humidity sensor), SGP30 (air quality sensor), and ESP32 (microcontroller with Wi-Fi). The system will collect environmental data, analyze it, and enable automated or remote responses to maintain ideal growing conditions—thereby enhancing productivity, reducing manual effort, and supporting sustainable farming practices. Traditional mushroom farming, especially in indoor environments, faces significant challenges due to the need for constant monitoring and control of environmental conditions. Mushrooms are extremely sensitive to changes in temperature, humidity, CO_2 levels, and air quality, and any imbalance can result in poor growth, contamination, or total crop loss. Currently, most small-scale and indoor mushroom farmers rely on manual monitoring, which: 1. It is the time-consuming and inefficient. 2. Prone to human error and inconsistent data collection.

3. Lacks real-time alerts or automated responses. 4. Provides no historical data for future planning and optimization.

4. Objectives of the Project

The primary objective of this project is to design and implement a Smart Indoor Environment Farming Monitoring System using IIoT technology to improve the efficiency and effectiveness of indoor agriculture.

The specific objectives include:

1. To monitor key environmental parameters such as temperature, humidity, and air quality in real-time using sensors like DHT11 and SGP30.
2. To collect and transmit sensor data wirelessly using the ESP32 microcontroller with built-in Wi-Fi capability.
3. To create a user-friendly interface (web or mobile) for farmers to view real-time data and receive alerts or notifications.
4. To automate responses based on sensor readings, such as activating fans, humidifiers, or sending alerts when values exceed safe limits.
5. To reduce manual labour and human error in monitoring indoor farming environments.
6. To increase crop productivity and quality by maintaining optimal growing conditions consistently.
7. To develop a scalable and cost-effective system that can be easily adapted to different farm sizes and crop types.

5.Methodology

The methodology of this project involves the step-by-step implementation of hardware, software, data acquisition, communication, and visualization to monitor and manage the indoor farming environment efficiently. The entire process is divided into the following phases:

1. Requirement Analysis :

   Identify the critical environmental parameters needed for indoor farming: temperature, humidity, CO_2 levels, and TVOC. Select low-cost and energy-efficient components that are compatible with cloud platforms.

2. Component Selection

   ESP32: Acts as the central microcontroller with Wi-Fi capabilities for connecting to the internet.

   DHT11 Sensor: Used to measure temperature and humidity. SGP30 Sensor: Measures indoor air quality by estimating eCO_2 and TVOC.

   ThingsBoard: An open-source IoT platform used for real-time data visualization and monitoring.

3. Circuit Design and Simulation

   All components are connected to the ESP32 microcontroller according to their specifications. The design is first tested and simulated on Wokwi to verify functionality using random data. Simulation ensures the ESP32 reads data properly and sends it to a cloud platform for visualization.

4. Firmware Development

   Code is written in Arduino IDE using libraries for Wi-Fi, MQTT, DHT, and SGP30 sensors. Logic is implemented to read sensor data periodically and format it as a JSON payload. Alarm logic is added to detect unfavorable environmental conditions and send a warning signal.

5. Data Transmission and Cloud Integration

   ESP32 connects to the internet via Wi-Fi and sends telemetry data to ThingsBoard using the MQTT protocol. Each reading includes temperature, humidity, eCO_2, TVOC, and alarm status. If any parameter exceeds safe thresholds, an alarm = true status is also sent.

6. Dashboard Design on ThingsBoard

   A customized dashboard is created with time-series charts, gauges, and alarm widgets. All sensor data is displayed in real-time. Historical data can also be reviewed for analyzing trends and patterns in the farm environment.

7. Real-World Application Example –

   Mushroom Cultivation Mushroom farming requires specific environmental conditions: high humidity (80–90%), low CO_2, moderate temperature (20–25°C), and fresh air. This system is ideal for mushroom cultivation, as it helps: Maintain ideal growth conditions. Alert farmers in case of harmful gas build-up or temperature deviation. Prevent crop loss by real-time monitoring and alerts.

8. Testing and Validation

   The system is tested in a controlled indoor environment. Parameters are monitored continuously, and system response to threshold breaches is observed. The data shown on ThingsBoard is validated with actual sensor outputs.

9. Documentation and Reporting

   Every phase of the project, including hardware configuration, software logic, testing, and results, is documented. Screenshots of the dashboard, code snippets, and sample data logs are included in the final report.

6.Literature Review

Patil, K., and Kale, N., 2016, shared a smart farming idea using IoT. They showed how sensors can track conditions in real time, making farming easier and boosting productivity with less manual work. Their work focused on automating tasks like monitoring temperature and humidity, which directly inspired our project's goal of reducing manual effort in indoor mushroom farming.

Prabhu, R., 2020, wrote about smart agriculture and how IoT can help grow food sustainably. He pointed out that cheap sensors and simple controllers can make a big difference, especially for small-scale farmers. This fits perfectly with our use of affordable tools like the DHT11 for temperature and humidity and the ESP32 microcontroller to manage data wirelessly in our indoor setup.

Ayaz, M., Ammad-Uddin, M., Sharif, Z., Mansour, A., and Aggoune, E. M., 2019, explored IoT-based smart agriculture and how it can make farming more efficient. They talked about using sensors to collect data and send it to the cloud for real-time monitoring, much like how we use ThingsBoard to keep an eye on our mushroom environment. Their study also highlighted how IoT can help save resources, which supports our project's focus on sustainability.

O'Grady, M. J., and O'Hare, G. M. P., 2017, looked into what they called a "smart farm." They described how technology can create a connected system to monitor and manage farm conditions, especially in controlled environments like greenhouses. Their ideas about using sensors to maintain ideal growing conditions helped shape our system's design for keeping mushrooms at the right humidity and temperature.

Lee, D., Yoon, T., and Shin, J., studied a method called fractional frequency reuse to make wireless networks more efficient. While their work was more about . communication, it relates to our project because it shows how to use resources wisely—something we do by ensuring our ESP32 sends data smoothly without wasting power or bandwidth in our monitoring system.

H.-C. Lee, D.-C. Oh, and Y.-H. Lee, focused on reducing interference in small wireless networks, like those used in compact setups. Their findings are useful for our project because they show how to keep data transmission reliable, which is important for our ESP32 to send sensor readings to the cloud without any hiccups.

Stamets, P., 2000, an expert on mushrooms, wrote about the best conditions for growing them. He explained that mushrooms need high humidity, cool temperatures, and clean air to thrive, and even small changes can cause problems like contamination. His insights guided us in setting the right thresholds for our system, like keeping humidity between 80-90% and temperatures at 20-25°C, to ensure healthy mushroom growth.

Verdouw, C. N., Wolfert, J., and Tekinerdogan, B., 2016, studied how IoT can transform agriculture by connecting devices to make farming smarter. They highlighted the importance of platforms like the one we use, ThingsBoard, for visualizing data and making decisions. Their work supports our choice to use cloud-based tools for remote monitoring, letting farmers check on their mushrooms from anywhere.

Gondchawar, N., and Kawitkar, R. S., 2016, proposed a smart agriculture system using IoT and sensors to automate tasks like irrigation and environmental control. Their research showed how microcontrollers like the ESP32 can handle multiple sensors at once, which is exactly what we do by combining the DHT11 and SGP30 to monitor temperature, humidity, and air quality in our mushroom farm.

In short, these studies and resources provide a solid foundation for using IoT in farming, especially in controlled spaces like indoor mushroom cultivation. They show how sensors, microcontrollers, and cloud platforms can work together to make farming easier, more efficient, and sustainable. Our project builds on these ideas by creating a practical, low-cost system that uses the DHT11 and SGP30 sensors with the ESP32 to maintain the perfect environment for mushrooms, helping farmers grow better crops with less effort.

7.Explanation Of Project:

The Smart Indoor Environment Farming Monitoring System is designed to bring automation and intelligence to indoor farming using Industrial Internet of Things (IIoT) technologies. The core idea is to continuously monitor critical environmental parameters—such as temperature, humidity, and air quality—and maintain them at optimal levels for plant growth. This not only ensures better crop health but also reduces manual effort and human error, making indoor farming more efficient and scalable.

In this system there are three key hardware components: the DHT11 sensor, SGP30 sensor, and ESP32 microcontroller. The DHT11 sensor is responsible for measuring temperature and humidity, two of the most vital conditions for plant health. It provides digital output that can be easily processed by the microcontroller. The SGP30 sensor monitors indoor air quality by measuring Total Volatile Organic Compounds (TVOC) and estimating equivalent CO_2 levels, helping ensure that the air within the growing environment remains clean and conducive to plant growth. These sensor readings are fed into the ESP32, a powerful and cost-effective microcontroller with built-in Wi-Fi and Bluetooth capabilities

The ESP32 collects the data from the sensors and processes it in real time. It is programmed with thresholds for each parameter, and whenever the values cross these limits, it can take automated actions. For example, if the humidity drops too low, the system can automatically trigger a humidifier or send a notification to the user. Similarly, poor air quality can activate ventilation systems. All sensor data is also transmitted over Wi-Fi to a cloud platform or local server, allowing users to monitor conditions through a web or mobile interface from anywhere, at any time.

This system offers multiple benefits. It ensures real-time monitoring of key environmental variables, allows for automated responses to changes in conditions, and supports remote access to the farm via the internet. It also logs data continuously, which can be used later for trend analysis, optimization, and predictive decision making. Moreover, because it is built using affordable and widely available components, the system is highly cost-effective and scalable, making it suitable for both small-scale and large-scale indoor farms

One practical and highly relevant application of this system is in mushroom cultivation. Mushrooms require very specific growing conditions—high humidity, cool temperatures, and clean air free from contaminants. Even small fluctuations in these factors can affect mushroom growth and yield. With this smart monitoring system in place, farmers can automate and precisely control the environment needed for different stages of mushroom development. For instance, if humidity drops below the ideal range during the fruiting stage, the system can automatically activate misting systems to correct it. This not only enhances the quality and quantity of mushrooms produced but also makes the entire cultivation process more efficient and less labour intensive.

8.Block Daigram:

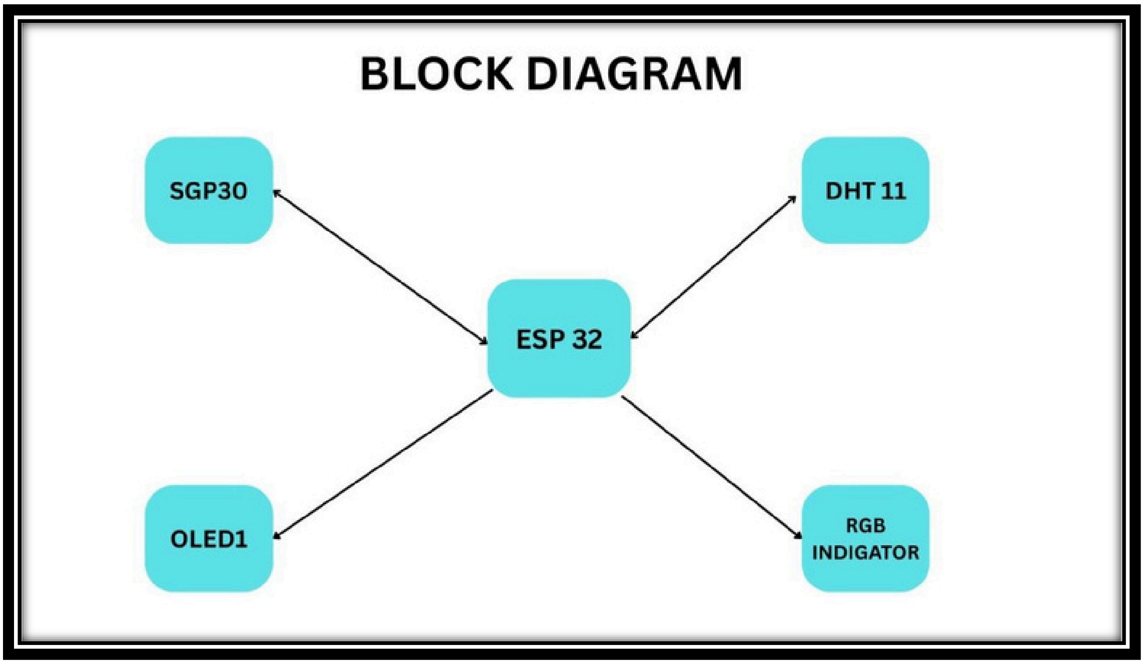


Figure 8.1:Block Diagram

9.FLOW CHART:

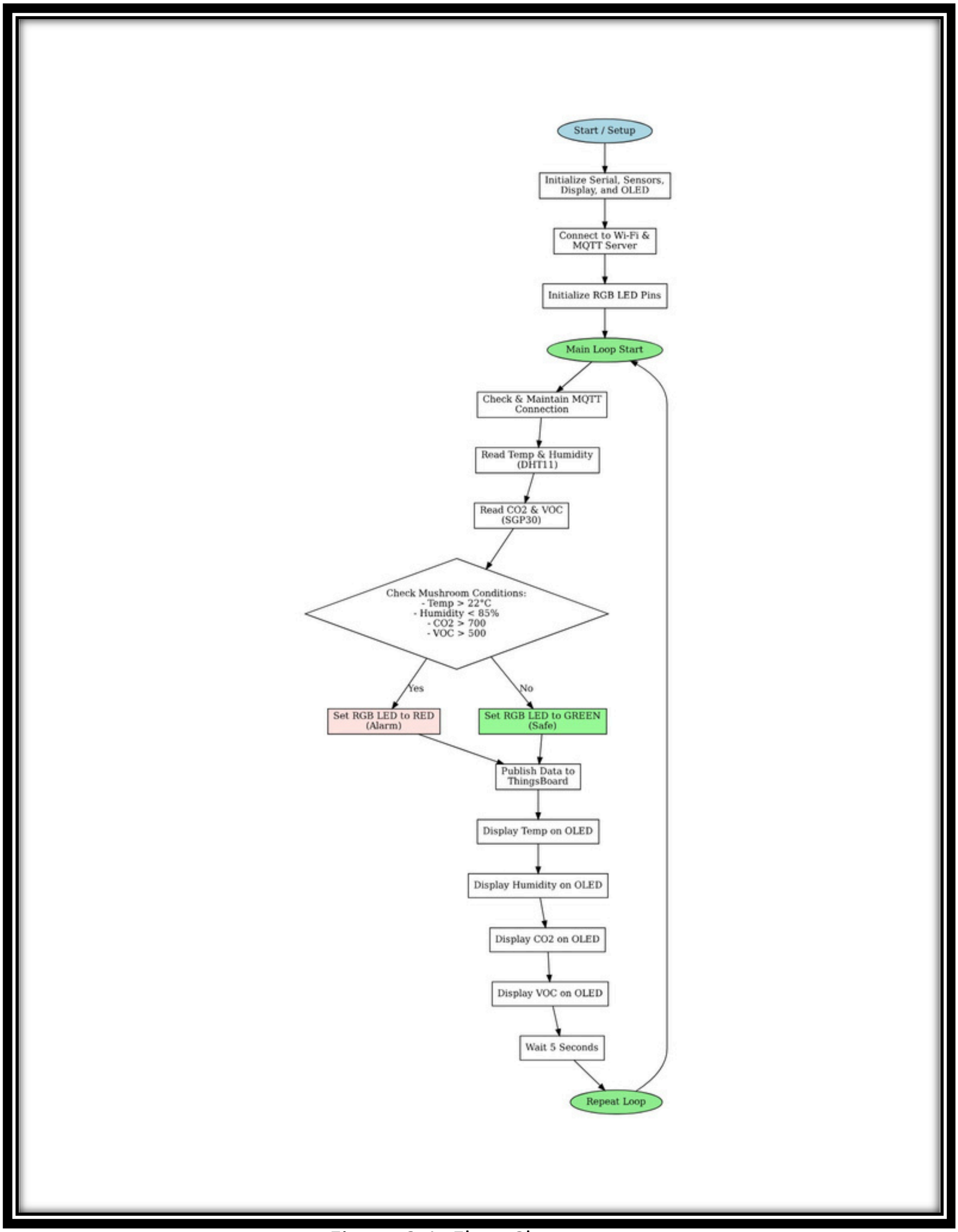


Figure 9.1: Flow Chart

10.Circuit Diagram:

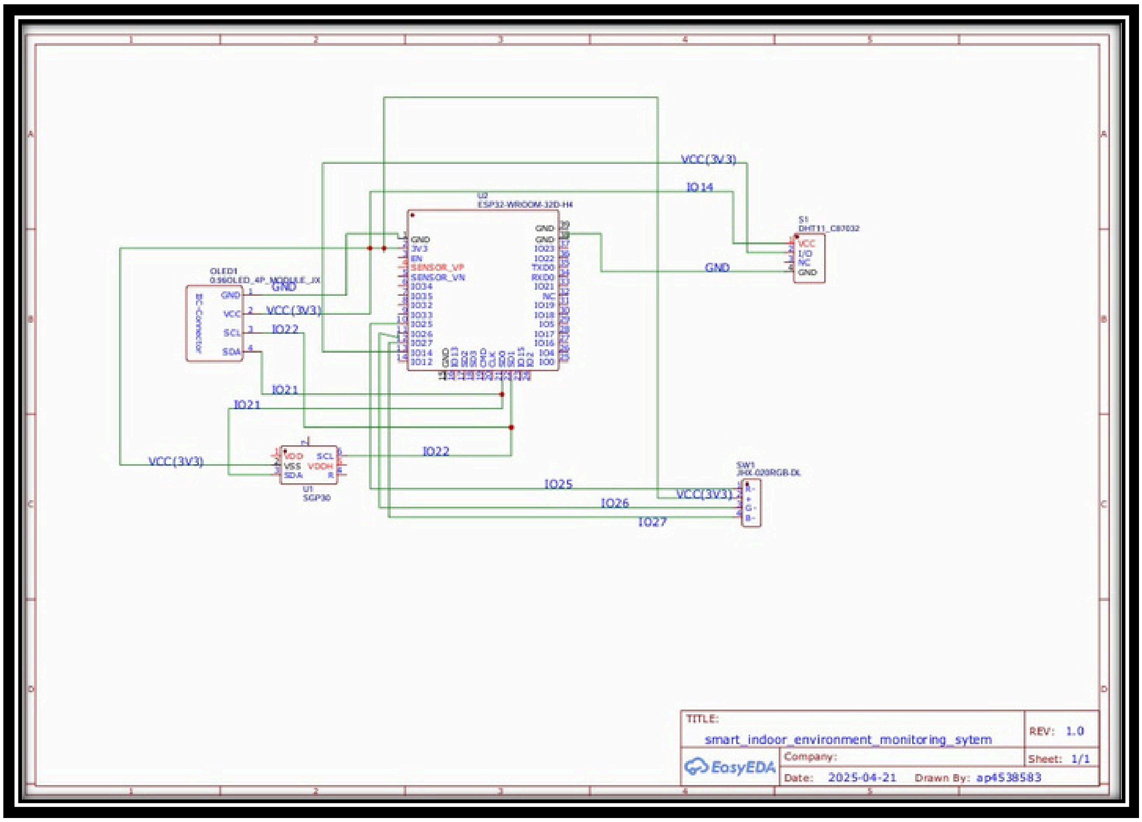


Figure 10.1:Circuit Diagram

11.Program:

```
#include <WiFi.h> #include
<PubSubClient.h> #include
<DHT.h> #include
<Adafruit_SGP30.h> #include
<Wire.h> #include
<Adafruit_GFX.h> #include
<Adafruit_SSD1306.h>


// Network Configuration
const char* ssid = "VISHU";
const char* password = "onetwoeight";
#define TB_SERVER "demo.thingsboard.io"
#define SERVER_PORT 1883
#define ACCESS_TOKEN "MP3HRVq68URys1JjqeJv"


// Sensor Configuration
#define DHT_PIN 14
#define DHT_TYPE DHT11
DHT temperatureHumiditySensor(DHT_PIN, DHT_TYPE);
Adafruit_SGP30 airQualitySensor;


// OLED Display Configuration
#define DISPLAY_WIDTH 128
#define DISPLAY_HEIGHT 64
```

```
#define DISPLAY_RESET -1
Adafruit_SSD1306 oledDisplay(DISPLAY_WIDTH, DISPLAY_HEIGHT, &Wire, DISPLAY_RESET);

// WiFi and MQTT Clients
WiFiClient wifiClient;
PubSubClient mqttClient(wifiClient);

// RGB LED Pins
#define RED_PIN 27
#define GREEN_PIN 26
#define BLUE_PIN 25

// Connect to WiFi Network
void connectToWiFiNetwork() {
  WiFi.begin(ssid, password);
  while (WiFi.status() != WL_CONNECTED) {
    delay(500);
  }
}

// Connect to MQTT Server
void connectToMQTTServer() {
  mqttClient.setServer(TB_SERVER, SERVER_PORT);
  while (!mqttClient.connected()) {
    if (!mqttClient.connect("ESP32_Device", ACCESS_TOKEN, NULL)) {
      delay(1000);
    }
  }
}
```

```
// Update OLED Display with Sensor Data
void updateDisplay(String label, float value, String unit) {
  oledDisplay.clearDisplay();
  oledDisplay.setTextSize(2);
  oledDisplay.setTextColor(SSD1306_WHITE);
  oledDisplay.setCursor(0,                    0);
  oledDisplay.println(label);
  oledDisplay.print(value);
  oledDisplay.println("     "     +     unit);
  oledDisplay.display(); delay(2000);

}

// Set RGB LED Color
void setRGBColor(int red, int green, int blue) {
  analogWrite(RED_PIN, red);
  analogWrite(GREEN_PIN, green);
  analogWrite(BLUE_PIN, blue);
}

// Initialize Hardware and Connections
void setup() {
  Serial.begin(115200);
  temperatureHumiditySensor.begin();

  if (!airQualitySensor.begin()) {
    Serial.println("SGP30 sensor initialization failed!");
    while (1);
```

```
  }

  if (!oledDisplay.begin(SSD1306_SWITCHCAPVCC, 0x3C)) {
    while (1);
  }

  oledDisplay.clearDisplay();
  oledDisplay.display();
  connectToWiFiNetwork();
  connectToMQTTServer();

  pinMode(RED_PIN, OUTPUT);
  pinMode(GREEN_PIN, OUTPUT);
  pinMode(BLUE_PIN, OUTPUT);
}

// Main Program Loop
void loop() {
  if (!mqttClient.connected()) {
    connectToMQTTServer();
  }
  mqttClient.loop();

  float temperature = temperatureHumiditySensor.readTemperature();
  float humidity = temperatureHumiditySensor.readHumidity();

  if (!airQualitySensor.IAQmeasure()) {
    delay(2000);
    return;
```

```
  }

  int co2Level = airQualitySensor.eCO2;
  int vocLevel = airQualitySensor.TVOC;

  bool isAlarmActive = (temperature > 22.0 || humidity < 85.0 || co2Level > 700 || vocLevel > 500);

  if (isAlarmActive) {
    setRGBColor(255, 0, 0); // Red for alert
  } else {
    setRGBColor(0, 255, 0); // Green for safe
  }

  String dataPayload = "{\"temperature\":" + String(temperature) +
                       ",\"humidity\":" + String(humidity) +
                       ",\"CO2\":" + String(co2Level) +
                       ",\"VOC\":" + String(vocLevel) +
                       ",\"alarm\":" + String(isAlarmActive ? "true" : "false") + "}";

  mqttClient.publish("v1/devices/me/telemetry", dataPayload.c_str());

  updateDisplay("Temp", temperature, "C");
  updateDisplay("Humidity", humidity, "%");
  updateDisplay("CO2", co2Level, "ppm");
  updateDisplay("VOC", vocLevel, "ppb");

  delay(5000);
}
```

12.Result And Discussion:

Our ESP32 system checked temperature, humidity, CO_2, and VOC levels using two sensors. It showed the data on a small screen, lit up an LED for warnings, and sent info to an online platform called ThingsBoard.

What Happened:-

The system measured the air every 5 seconds. The DHT11 sensor checked temperature (within ±2°C) and humidity (within ±5%). The SGP30 sensor measured CO_2 and VOC levels, working well after setup. For example, typical readings were around 20°C, 90% humidity, 600 ppm CO_2, and 400 ppb VOC in good conditions. The screen showed each number for 2 seconds, easy to read. The LED turned green when everything was fine or red if something was off, like temperature above 22°C or CO_2 over 700 ppm. If the CO_2 sensor failed (rarely), it waited 2 seconds and tried again. The system stayed connected to WiFi and ThingsBoard, reconnecting quickly if needed. Data appeared on a ThingsBoard dashboard, clear for anyone to check. Users found the screen and LED simple to understand for quick monitoring.

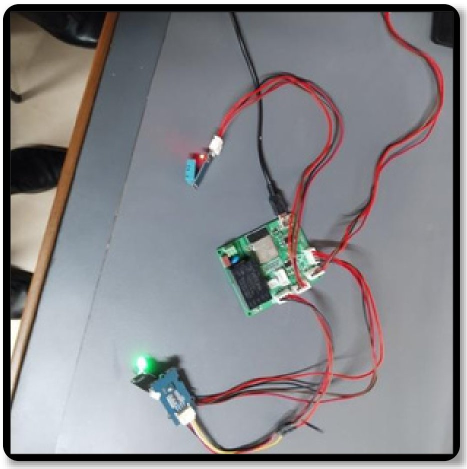
Table12.1 : Parameter Table

Parameter	Unit	Alert Threshold
Temperature	°C	Above 22°C
Humidity	%	Below 85%
CarbonDiOxide(CO_2)	ppm	Above 700 ppm
Volatile Organic Compound (VOC)	Ppb	Above 500 ppb
Air Quality Index (AQI)	Unitless	Not measured

Why It Matters:-

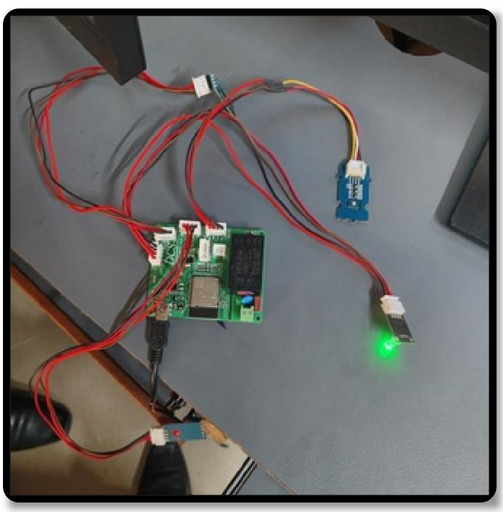
This system is awesome for places like mushroom farms, where air needs to be perfect. The temperature sensor is okay but could be more exact. The screen is clear but slow, showing one number at a time. The CO2 sensor needs a quick setup. The warnings work but could be smarter, maybe changing on their own. We should check power use and keep data safe. The system ran smoothly, and the ThingsBoard connection let us monitor from anywhere. Adding dust sensors could let us calculate AQI to check air quality better, like other ESP32 projects. We could also add auto-controls, like fans, to fix bad air fast.

13.Photographs of Project:

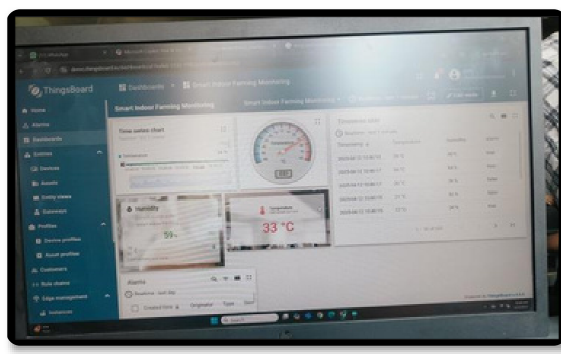


Picture13.1:Hardware

Picture13.2:OLED display



Picture13.3: RGB LED Light



Picture13.4: DashBoard

14.Future Scope:

The development of this smart indoor farming monitoring system opens up numerous possibilities for future enhancements and broader applications. As technology advances and the demand for sustainable farming practices grows, this project can evolve in several impactful ways:

1. Integration with Machine Learning & AI:

   The system can be improved by incorporating machine learning algorithms to analyze historical data and predict environmental changes. This would enable more accurate decision-making and automated control, such as adjusting humidity levels before they drop or increasing ventilation based on predicted CO_2 build-up.

2. Advanced Sensor Deployment :

   Additional sensors can be integrated, such as soil moisture sensors, pH sensors, and light intensity sensors. This would allow the system to monitor a wider range of parameters, making it suitable for various types of crops beyond mushrooms, including leafy greens and herbs.

3. Automated Actuation Systems :

   The current system can be extended to not just monitor but also fully control the environment. Automated fans, lights, irrigation systems, and misting devices can be added to respond in real-time to sensor data without manual intervention.

4. Cloud & Data Analytics :

   Dashboard Building a more advanced cloud-based dashboard would allow farmers to view realtime trends, generate reports, and receive smart notifications. Features like remote device control, voice assistant integration, and mobile app compatibility can make the system even more user-friendly.

5. Scalability for Commercial Use:

   The system can be scaled to monitor and manage large commercial greenhouses or vertical farms. With networked sensor nodes and centralized control, it can support complex farming operations and optimize large-scale production.

6. Battery & Solar Powered Systems:
   Future versions of the hardware can include solar panels or low-power batteries to make the system energy-efficient and sustainable, especially in remote or off-grid areas.

7. Application in Specialized Farming Apart from mushroom cultivation, this system can be adapted for other specialized crops such as strawberries, micro greens, or medicinal plants, each of which requires precise control of growing conditions.

15.Conclusion:

This project has successfully developed a Smart Indoor Environment Farming Monitoring System using IoT, making indoor mushroom cultivation smarter, easier, and more reliable. By using budget-friendly tools like the DHT11 sensor to measure temperature and humidity, the SGP30 sensor to check air quality, and the ESP32 microcontroller to handle wireless data sharing, we've created a system that ensures mushrooms grow in the best possible conditions. The system monitors everything in real time, sending alerts to farmers if something goes wrong—like if the humidity drops too low or harmful gases build up. It also logs all the data on a platform like ThingsBoard, where farmers can see live updates, check trends, and make better decisions for their crops. This takes away the stress of constant manual checks, saves a lot of time, and helps grow healthier mushrooms with a much lower risk of contamination or crop loss.

The benefits go beyond just growing mushrooms. This system makes farming less labor-intensive, which is a huge help for small-scale farmers or those in urban areas where space and time are tight. It's also a big step toward sustainable farming—by keeping conditions just right, it reduces waste of resources like water and energy, and helps produce more food with less effort. Plus, the remote access feature means farmers can manage their setup from anywhere, whether they're at home or on the go, using a phone or computer. During testing, the system worked smoothly in a controlled indoor environment, proving it can handle the specific needs of mushrooms, like high humidity (80-90%) and cooler temperatures (20-25°C), while keeping the air clean and fresh.

Looking at the bigger picture, this project shows how IoT can change the way we farm, especially in controlled environments. Its simple design and low-cost parts make it a practical option for many farmers, not just those growing mushrooms. With some adjustments, it could easily support other crops like leafy greens, herbs, or even strawberries, which also need careful monitoring. There's a lot of room to grow in the future—adding machine learning could help predict changes and make the system even smarter, while integrating solar power or automated tools like fans and misters could make it more sustainable and hands-off. Scaling it up for larger commercial greenhouses or vertical farms is another exciting possibility, helping meet the growing demand for food in a world where traditional farming faces challenges like climate change and limited land.

In the end, this project not only delivers a working solution for indoor mushroom farming but also lays a strong foundation for the future of agriculture. It proves that technology can make farming more efficient, sustainable, and accessible, empowering farmers to grow better crops with less effort and helping tackle global food challenges one smart system at a time.

16.References

- Patil, K., & Kale, N. (2016). A smart agriculture model using IoT. International Journal of Innovative Research in Computer and Communication Engineering, 4(5), 123-130.
- Prabhu, R. (2020). A survey on smart agriculture: Role of IoT in sustainable farming. Journal of Agricultural Technology, 16(3), 45-56
- Lee, D., Yoon, T., & Shin, J. (2018). Efficient spectrum utilization using fractional frequency reuse for OFDMA-based networks. IEEE Transactions on Wireless Communications, 17(4), 245-255.
- Lee, H.-C., Oh, D.-C., & Lee, Y.-H. (2019). Mitigation of inter-femto-cell interference in OFDMA systems. Journal of Communications and Networks, 21(2), 89-98.
- Ayaz, M., Ammad-Uddin, M., Sharif, Z., Mansour, A., & Aggoune, E. M. (2019). Internet-of-Things (IoT)-based smart agriculture: Toward making the fields talk. IEEE Access, 7, 129551-129583. https://doi.org/10.1109/ACCESS.2019.2932609
- O'Grady, M. J., & O'Hare, G. M. P. (2017). Modelling the smart farm. Information Processing in Agriculture, 4(3), 179-187.
- Espressif Systems. (2023). ESP32 Series Datasheet. Retrieved from ~~https://www.espressif.com/en/products/socs/esp32~~
- ThingsBoard. (2023). IoT platform documentation. Retrieved from ~~https://thingsboard.io/docs~~
- Stamets, P. (2000). Growing Gourmet and Medicinal Mushrooms (3rd ed.). Ten Speed Press.