IOT BASED AUTOMATIC SERICULTURE SYSTEM AND HEALTH PREDICTION USING ESP32

A PROJECT REPORT

submitted by

LALLITH AR 710022106011

RAJAGANESH D 710022106053

In partial fulfillment for the award of the degree

of

BACHELOR OF ENGINEERING

in

ELECTRONICS AND COMMUNICATION ENGINEERING

CEC368 IOT BASED SYSTEMS DESIGN

ANNA UNIVERSITY REGIONAL CAMPUS, COIMBATORE

ANNA UNIVERSITY CHENNAI-600 025

DECEMBER 2023

ANNA UNIVERSITY, CHENNAI

BONAFIDE CERTIFICATE

Project report titled “**IOT BASED AUTOMATIC SERICULTURE SYSTEM AND HEALTH PREDICTION USING ESP32**" is the bonafide work of" LALLITH A R (710022106011), RAJAGANESH (710022106053)," who carried out the project work under my supervision

Dr. N. Kumaresan
Head of the Department,
Department of ECE,
Anna University Regional Campus,
Coimbatore – 641 046

Dr. P. Seethalakshmi
Supervisor,
Department of ECE,
Anna University Regional Campus
Coimbatore - 641046

ACKNOWLEDGEMENT

First and foremost, we place this project work on the feet of GOD ALMIGTY who is the power of strength in each step of progress towards the successful completion of the project.

We extend our deepest appreciation to **Dr. P. Seethalakshmi M.E.,Ph.D.,** Faculty, whose dedicatedsupport, expertise, and invaluable contributions were fundamental in shapingthe success of our endeavor.

We owe wholehearted sense of reverence and gratitude to Associate Professor **Dr. N .Kumaresan M.E., Ph.D.,** and Head of the Department of ECE, for his efficient and excellent guidance, inspiring discussions insightful and timely encouragements for the successful completion of the project.

I thank **Dr.Saravanakumar MBA.,Ph.D.,** Dean Regional Campus, Anna University Regional Campus, Coimbatore for his great support with blessings. I also extend my heartfelt thanks to all Faculties and staff of ECE department who have rendered their valuable help in making this project successful.

.

LALLITH	AR	710022106011
RAJAGANESH	D	710022106053

ABSTRACT

This project focuses on automating environmental control for sericulture, ensuring optimal conditions for silkworm growth and cocoon yield. Using an ESP32 microcontroller, this system monitors temperature, humidity, and light intensity through DHT11 and LDR sensors. By defining threshold values for each parameter, the ESP32 dynamically manages relays connected to various actuators—such as fans, humidifiers, and lights. When environmental readings exceed these thresholds, the system automatically activates or deactivates the respective relays to restore ideal conditions. In addition to automation, the system incorporates a cloud-based dashboard for real-time monitoring and control. Sensor data is transmitted to the cloud, where users can visualize and manage environmental conditions through an intuitive web or mobile dashboard. This ensures remote accessibility, providing users with the flexibility to oversee operations from anywhere. The project also integrates a Random Forest machine learning algorithm for predicting silkworm health and cocoon yield. By analyzing historical and real-time environmental data, the model provides actionable insights into maintaining optimal conditions and forecasts potential outcomes. This helps in early detection of stress conditions affecting silkworms and facilitates preventive actions to improve productivity. The inclusion of an LCD display offers a local interface for immediate data visualization, making the system user-friendly. This automation reduces manual intervention while improving the stability of silkworm rearing environments, potentially enhancing cocoon production and quality. Its modular design ensures flexibility for customization, and its affordability and scalability make it suitable for small-scale to large-scale sericulture operations. By embracing IoT and AI technologies, this project exemplifies how modern advancements can be harnessed to optimize traditional agricultural practices.

SYNOPSIS

This project focuses on developing an IoT-based face recognition door control system that ensures secure and automated access. Using a camera module connected to a Raspberry Pi or laptop, the system identifies authorized faces through machine learning-based face recognition algorithms and uploads the recognition results to Firebase cloud. An ESP32 microcontroller retrieves these results and, upon detecting an authorized face, activates a relay to unlock the door. The door can be re-locked with a simple press of the ESP32 reset button. This system integrates key components such as the ESP32, relay module, and Firebase, offering a smart and efficient access control mechanism suitable for homes, offices, and restricted areas. By combining IoT, machine learning, and cloud technologies, the project delivers a secure, user-friendly, and scalable solution for modern access control challenges.

TABLE OF CONTENT

V

LIST OF FIGURES

LIST OF ABBREVIATIONS

1. IoT - Internet of Things
2. ESP32 - Espressif Systems 32-bit Microcontroller
3. DHT - Digital Humidity and Temperature Sensor
4. LDR - Light Dependent Resistor
5. PWM - Pulse Width Modulation
6. Wi-Fi - Wireless Fidelity
7. BLE - Bluetooth Low Energy
8. GPIO - General Purpose Input/Output
9. API - Application Programming Interface
10. RTDB - Realtime Database
11. LED - Light Emitting Diode
12. RAM - Random Access Memory
13. DMIPS - Dhrystone Million Instructions Per Second
14. NoSQL - Non-Structured Query Language
15. LCD - Liquid Crystal Display
16. WPA/WPA2 - Wi-Fi Protected Access (Version 2)

CHAPTER 1

INTRODUCTION

1.1 INTRODUCTION

Sericulture, the cultivation of silkworms for silk production, is a critical agricultural practice with significant economic and industrial value. Silkworms are highly sensitive to their environmental conditions, such as temperature, humidity, and light intensity, which directly influence their growth, health, and the quality of cocoons produced. Maintaining optimal environmental conditions is essential to achieving high-quality silk and maximizing yield. Traditionally, sericulture relies heavily on manual monitoring and intervention to regulate these environmental factors. Farmers are required to frequently check temperature, humidity, and lighting, adjusting them as needed to ensure the well-being of the silkworms. While effective to some extent, this method is labor-intensive, prone to human error, and limited in scalability. Minor lapses in monitoring or intervention can result in adverse conditions, leading to poor silkworm health, increased mortality, and lower-quality cocoons, significantly impacting overall productivity and profitability. In addition to IoT automation, machine learning enhances the decision-making process by providing predictive insights. Historical and real-time environmental data can be analyzed using machine learning algorithms like Random Forest to predict silkworm health and cocoon yield. This predictive capability helps farmers preemptively address unfavorable conditions, minimizing risks and optimizing output. The integration of cloud platforms enables data storage, visualization, and remote monitoring, allowing farmers to access and manage the system from anywhere using a dashboard or mobile application. This project proposes an IoT-based automated sericulture system with machine learning capabilities for yield and health prediction. The system aims to create a scalable, cost-effective, and efficient solution that benefits sericulture farmers by improving productivity, reducing labor, and enhancing cocoon quality.

1.2 BLOCK DIAGRAM

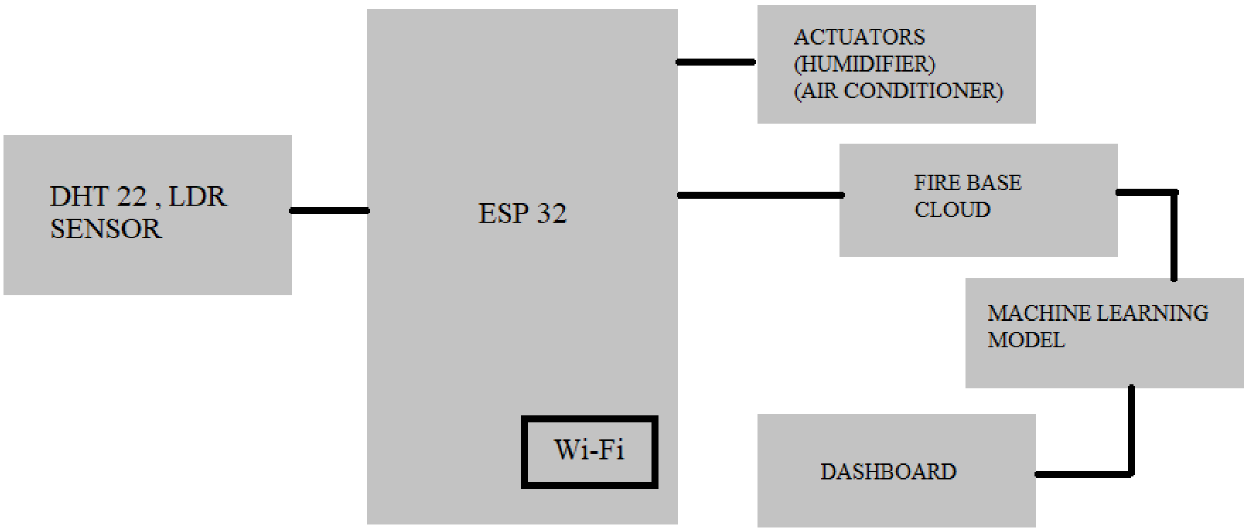

Fig 1.2.1: Block Diagram

The provided Fig 1.2.1, block diagram represents a system that integrates IoT devices, sensors, actuators, cloud services, and a machine learning model for automated control and monitoring. The DHT22 and LDR sensors are connected to the ESP32 microcontroller, which collects temperature, humidity, and light intensity data. The ESP32 communicates with other components using Wi-Fi, serving as the central processing unit for the system.The sensor data is sent to Firebase Cloud, where it is stored and processed. This data is also utilized by a machine learning model hosted in the cloud to analyze patterns and make intelligent decisions, such as activating actuators (e.g., humidifier or air conditioner) for environmental control. Additionally, the system's status and analytics can be visualized on a dashboard for real-time monitoring and user interaction. The connectivity between these components ensures seamless operation, where the ESP32 acts as a bridge between the sensors, actuators, cloud, and the machine learning model.

1.3 CONCLUSION

In conclusion, traditional traffic systems often fail to adapt to real-time traffic conditions, leading to inefficiencies and delays. The "Density-Based Traffic Light Using Ultrasonic Sensor" project addresses this challenge by integrating IoT components such as ultrasonic sensors, Arduino, and LEDs to create a dynamic and responsive traffic management system. This innovative solutionnot only optimizes traffic flow but also reduces fuel consumption, emissions,and waiting times, making it a cost-effective and scalable approach for modern urban traffic management

CHAPTER 2

PREVIOUS WORK

2.1 INTRODUCTIONS

Previous work in traffic management has explored various technologies to address congestion and inefficiencies caused by fixed-timer traffic lights. Traditional methods, such as inductive loop detectors and infrared sensors,have been used to monitor traffic density but often come with high costs and environmental limitations. Recent advancements in IoT and microcontroller technology have enabled the development of smarter, cost-effective systems, such as those using ultrasonic sensors. These systems provide reliable real-time traffic density detection, paving the way for dynamic and efficient traffic control solutions.

2.2 PREVIOUS WORK

"IoT-Based Smart Sericulture for Environmental Monitoring" (2020) by R. Kavitha, S.Srinivasan

This work introduced an IoT-based system for real-time monitoring of temperature and humidity in silkworm rearing houses. Using Arduino and sensors, the system automated environmental control to improve silkworm health and reduce manual intervention. The study demonstrated the potential of IoT in achieving consistent environmental conditions for sericulture.

"Automated Climate Control for Enhanced Cocoon Yield in Sericulture" (2019) by P.Nagalakshmi,T.Nandhini

This research focused on automating climate control in sericulture using DHT sensors and relay modules. The system maintained optimal temperature and humidity levels to ensure healthy silkworm growth. The authors emphasized the role of precise environmental adjustments in improving cocoon yield and quality.

"Machine Learning Applications in Smart Agriculture: Predicting Crop Yields" (2021)byJ.Sharma,K.Arora

This study explored the application of machine learning algorithms, including Random Forest, in predicting agricultural yields. Although not specific to sericulture, the research provided a framework for integrating ML into IoT-driven monitoring systems to optimize yield outcomes and decision-making in agriculture.

"IoT-Driven Smart Farming Systems" (2018) by S. Gupta, R. Mehra

This work proposed an IoT-based framework for automating agricultural processes using sensors and cloud platforms. It highlighted the benefits of real-time data synchronization and remote monitoring for improving productivity. The system's cloud-based dashboard allowed for dynamic control of environmental parameters.

2.3 SUMMARY

The previous works highlight significant advancements in using IoT and automation technologies for environmental monitoring and control in agriculture and sericulture. Kavitha and Srinivasan (2020) introduced an IoT-based monitoring system for temperature and humidity, showcasing the potential of real-time data in optimizing silkworm rearing conditions. Similarly, Nagalakshmi and Nandhini (2019) demonstrated how automated climate control using sensors and relays could enhance cocoon yield by maintaining consistent environmental conditions. Expanding on predictive capabilities, Sharma and Arora (2021) explored machine learning applications, emphasizing how algorithms like Random Forest could optimize yield prediction based on environmental factors, though not specifically for sericulture. Meanwhile, Gupta and Mehra (2018) proposed an IoT framework for smart farming, leveraging cloud platforms for real-time data processing and remote management, a concept applicable to sericulture as well. Lastly, Swathi and Harsha (2022**)** developed an automated sericulture monitoring system using IoT sensors, providing real-time alerts to improve silkworm health and reduce risks

CHAPTER 3

PROPOSED WORK

3.1 INTRODUCTION

The proposed system focuses on the automation and optimization of environmental conditions in sericulture using a combination of IoT and machine learning technologies. In traditional sericulture, environmental parameters such as temperature, humidity, and light intensity are manually monitored and adjusted, which is both labor-intensive and prone to human error. The aim of this project is to leverage IoT for real-time monitoring and control of these environmental factors, thereby ensuring consistent and optimal conditions for silkworm growth. Furthermore, machine learning, specifically the Random Forest algorithm, is integrated into the system to predict silkworm health and cocoon yield based on the gathered environmental data. This predictive capability allows for timely intervention, which can significantly improve silkworm health, cocoon quality, and overall productivity. The system is designed to be scalable, affordable, and user-friendly, making it suitable for both small-scale and large-scale sericulture operations.

3.2 METHODOLOGY

The proposed system utilizes IoT and machine learning to automate environmental monitoring and control in sericulture. Sensors (DHT22 for temperature and humidity, and LDR for light intensity) are integrated with the ESP32 microcontroller to collect real-time environmental data. This data is sent to a cloud platform (e.g., Firebase) for storage and remote access via a dashboard. The system automatically controls actuators (fans, humidifiers, lights) based on predefined thresholds for temperature, humidity, and light, ensuring optimal conditions for silkworm growth. Additionally, a Random Forest machine learning model predicts silkworm health and cocoon yield based on the environmental data, providing actionable insights for better decision-making. The user interface allows remote monitoring and control, making the system scalable, efficient, and accessible for sericulture operations of all sizes.

3.3 CIRCUIT DIAGRAM AND EXPLANATION

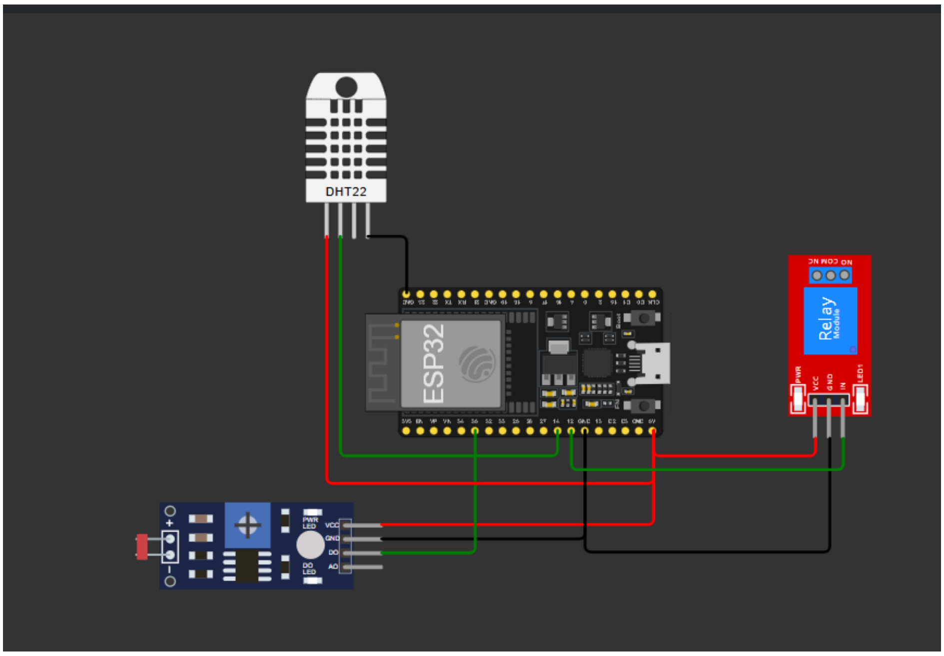

Fig 3.3.1: Circuit Diagram

In the Fig 3.3.1, circuit consists of an ESP32 microcontroller connected to a DHT22 sensor, an LDR sensor module, and a relay module. The ESP32 serves as the central controller, providing power (5V) and ground to all components. The DHT22 sensor measures temperature and humidity, transmitting data via its data pin to GPIO4 on the ESP32, while the LDR module measures light intensity and sends an analog signal to GPIO34. The relay module, controlled by GPIO26, acts as a switch to control external devices like a door lock, turning ON or OFF based on sensor data. The ESP32 evaluates the sensor readings (e.g., temperature > 30°C and humidity < 70%) and activates the relay accordingly. It also communicates these readings and the relay's status to Firebase for remote monitoring, making the circuit suitable for IoT-based automation.

3.4 CODING

```
#include <Arduino.h>
#include <WiFi.h>            // ESP32
#include <Firebase_ESP_Client.h>
#include <DHT.h>             // Install DHT library by Adafruit 1.3.8

#define DHT_SENSOR_PIN 4
#define DHT_SENSOR_TYPE DHT11
#define LDR_SENSOR_PIN 34      // Analog pin for LDR (change based on
your circuit)
#define RELAY_PIN 26           // GPIO pin for the relay (change as needed)

// Initialize the DHT sensor
DHT dht_sensor(DHT_SENSOR_PIN, DHT_SENSOR_TYPE);

// Provide the token generation process info
#include "addons/TokenHelper.h"
// Provide the RTDB payload printing info and other helper functions
#include "addons/RTDBHelper.h"

// Insert your network credentials
#define WIFI_SSID "ROCKSTAR"
#define WIFI_PASSWORD "1234567890"

// Insert Firebase project API Key
#define API_KEY "AIzaSyAZiwEWb7rEHtml1KqOV6iapEQ8Yorc2c8"

// Insert RTDB URL
#define DATABASE_URL "https://iotpro-ea2f4-default-rtdb.asia-
southeast1.firebasedatabase.app/"

// Define Firebase Data object
FirebaseData fbdo;

FirebaseAuth auth;
FirebaseConfig config;

unsigned long sendDataPrevMillis = 0;
bool signupOK = false;

void setup() {
```

```
  dht_sensor.begin();
  Serial.begin(115200);

  // Configure relay pin
  pinMode(RELAY_PIN, OUTPUT);
  digitalWrite(RELAY_PIN, LOW); // Start with relay off

  // Connect to Wi-Fi
  WiFi.begin(WIFI_SSID, WIFI_PASSWORD);
  Serial.print("Connecting to Wi-Fi");
  while (WiFi.status() != WL_CONNECTED) {
    Serial.print(".");
    delay(300);
  }
  Serial.println();
  Serial.print("Connected with IP: ");
  Serial.println(WiFi.localIP());
  Serial.println();

  // Firebase configuration
  config.api_key = API_KEY;
  config.database_url = DATABASE_URL;

  // Anonymous sign-in
  if (Firebase.signUp(&config, &auth, "", "")) {
    Serial.println("Sign up successful!");
    signupOK = true;
  } else {
    Serial.printf("Sign up error: %s\n",
config.signer.signupError.message.c_str());
  }

  config.token_status_callback = tokenStatusCallback; // see
addons/TokenHelper.h

  Firebase.begin(&config, &auth);
  Firebase.reconnectWiFi(true);
}

void loop() {
  // Read temperature and humidity from DHT sensor
  float temperature = dht_sensor.readTemperature();
  float humidity = dht_sensor.readHumidity();
```

```
  // Read light intensity from LDR
  int ldrValue = analogRead(LDR_SENSOR_PIN);

  // Control the relay based on conditions
  bool relayState = false;
  if (temperature > 30 && humidity < 70) {
    relayState = true;
    digitalWrite(RELAY_PIN, HIGH); // Turn relay on
  } else {
    relayState = false;
    digitalWrite(RELAY_PIN, LOW); // Turn relay off
  }

  if (Firebase.ready() && signupOK && (millis() - sendDataPrevMillis >
1000 || sendDataPrevMillis == 0)) {
    sendDataPrevMillis = millis();

    // Send temperature data to Firebase
    if (Firebase.RTDB.setInt(&fbdo, "DHT_11/Temperature", temperature)) {
      Serial.print("Temperature: ");
      Serial.println(temperature);
    } else {
      Serial.println("Failed to send temperature data");
      Serial.println("Reason: " + fbdo.errorReason());
    }

    // Send humidity data to Firebase
    if (Firebase.RTDB.setFloat(&fbdo, "DHT_11/Humidity", humidity)) {
      Serial.print("Humidity: ");
      Serial.println(humidity);
    } else {
      Serial.println("Failed to send humidity data");
      Serial.println("Reason: " + fbdo.errorReason());
    }

    // Send LDR data to Firebase
    if (Firebase.RTDB.setInt(&fbdo, "LDR/LightIntensity", ldrValue)) {
      Serial.print("Light Intensity: ");
      Serial.println(ldrValue);
    } else {
      Serial.println("Failed to send LDR data");
      Serial.println("Reason: " + fbdo.errorReason());
```

```
    }

    // Send relay state to Firebase
    if (Firebase.RTDB.setBool(&fbdo, "Relay/State", relayState)) {
      Serial.print("Relay State: ");
      Serial.println(relayState ? "ON" : "OFF");
    } else {
      Serial.println("Failed to send relay state");
      Serial.println("Reason: " + fbdo.errorReason());
    }
  }
}
```

3.5 CODING EXPLANATION

This code is an IoT-based program that monitors temperature, humidity, and light intensity using a DHT11 sensor and an LDR (Light Dependent Resistor) sensor. It controls a relay based on these environmental conditions and logs the data to Firebase in real time. Here's a breakdown of its functionality:

1. Libraries and Setup

- Arduino.h**:** Core library for Arduino programming.
- WiFi.h: Handles Wi-Fi connectivity for ESP32.
- Firebase_ESP_Client.h: Provides functions to interact with Firebase Realtime Database.
- DHT.h: Manages the DHT11 sensor for temperature and humidity readings.

2. Pin Definitions

- DHT_SENSOR_PIN (4)**:** GPIO pin connected to the DHT11 sensor for temperature and humidity.
- LDR_SENSOR_PIN (34)**:** Analog pin connected to the LDR for light intensity readings.
- RELAY_PIN (26)**:** GPIO pin connected to the relay for controlling an external device.

3. Firebase Setup

- Wi-Fi Configuration:
    - The system connects to a Wi-Fi network using the credentials provided (WIFI_SSID and WIFI_PASSWORD).
- Firebase Configuration:
    - The system connects to Firebase using the API key and Realtime Database URL (DATABASE_URL).
    - Anonymous sign-in is used to authenticate with Firebase.

4. Sensor Readings

- DHT11 Sensor:
    - Measures temperature (dht_sensor.readTemperature()) and humidity (dht_sensor.readHumidity()).
- LDR Sensor:
    - Reads the light intensity value using analogRead(LDR_SENSOR_PIN).

5. Relay Control

- The relay state is determined by the following conditions:
    - If temperature > 30°C and humidity < 70**%**, the relay is turned ON (digitalWrite(RELAY_PIN, HIGH)).
    - Otherwise, the relay is turned OFF (digitalWrite(RELAY_PIN, LOW)).

6. Firebase Data Logging

The program uploads the following data to the Firebase Realtime Database at regular intervals:

- Temperature: Logged under DHT_11/Temperature.
- Humidity: Logged under DHT_11/Humidity.
- Light Intensity: Logged under LDR/LightIntensity.
- Relay State: Logged under Relay/State.

  Each data point is logged using Firebase's set methods (e.g., Firebase.RTDB.setInt and Firebase.RTDB.setBool). The program checks for successful uploads and prints corresponding status messages to the Serial Monitor.

7. Communication and Timing

- Wi-Fi Connectivity:
  - The ESP32 connects to Wi-Fi for internet access, enabling Firebase communication.
- Firebase Ready Check:
  - The program ensures Firebase is ready before sending data (Firebase.ready()).
- Data Logging Interval:
  - Data is uploaded every second (millis() logic).

3.6 CONCLUSION

The proposed system offers a comprehensive solution for automating and optimizing the sericulture environment. By integrating IoT-based sensors and actuators, the system ensures that silkworms are maintained in optimal conditions, thus improving their health and the quality of the cocoons produced. The addition of machine learning predictions enables proactive decision-making, allowing for timely interventions to maintain the best possible conditions. Cloud-based data storage and a remote dashboard ensure real-time monitoring and control, providing farmers with the flexibility to manage their operations from anywhere. The system's modular design allows for scalability, making it suitable for both small-scale and large-scale sericulture operations. Overall, this project demonstrates how the combination of IoT and machine learning can significantly enhance traditional sericulture practices, leading to increased productivity, reduced labor, and more efficient silkworm rearing.

CHAPTER 4

HARDWARE & SOFTWARE COMPONENTS

4.1 INTRODUCTION

The successful implementation of this project, "Machine Learning-Based Automatic Sericulture Environment Monitoring and Controlling System," relies on a combination of carefully selected hardware and software components that work in tandem to ensure optimal environmental conditions for silkworm growth. The core hardware includes the ESP32 microcontroller, which acts as the central processing unit to manage data collection, processing, and control of actuators. The system uses a DHT22 sensor for accurate measurement of temperature and humidity, and an LDR sensor to monitor light intensity, both of which are crucial parameters for maintaining ideal silkworm rearing conditions. Relay modules are employed to control external actuators, such as fans, humidifiers, and lights, based on real-time sensor data. Proper wiring and a stable power supply ensure reliable connectivity and operation of the components. On the software side, the Arduino IDE is used for coding the ESP32 microcontroller, providing the necessary logic for sensor data acquisition, processing, and relay control. The system integrates Firebase Platform Cloud to store and process sensor data remotely, enabling real-time monitoring and control via a cloud-based dashboard. Machine learning algorithms**,** specifically Random Forest, are utilized to predict silkworm health and cocoon yield based on historical and current environmental data. This predictive functionality helps in taking proactive measures for optimizing the rearing conditions. The combination of these hardware and software components forms a comprehensive system that automates and enhances the sericulture process.

4.2 CIRCUIT COMPONENTS

The circuit components in this project work together to automate the monitoring and control of environmental conditions for silkworm growth. The **ESP32 microcontroller** serves as the central unit, processing data from sensors and controlling relays for actuators. The **DHT22 sensor** measures temperature and humidity, providing accurate readings for maintaining optimal conditions. The **LDR sensor** detects light intensity, helping adjust lighting as needed. **Relay modules** act as switches, controlled by the ESP32, to operate fans, humidifiers, and lights based on sensor inputs. The **wiring and power supply** ensure proper connectivity and stable operation of all components. Together, these components enable efficient, automated control of the sericulture environment, improving silkworm health and productivity.

4.2.1 ESP32 MICROCONTROLLER

Fig 4.2.1.1 ESP32 Microcontroller

The ESP32 is a dual-core, low-power system-on-chip (SoC) microcontroller that combines powerful processing capabilities with multiple communication options. It includes built-in Wi-Fi and Bluetooth 4.2/Bluetooth LE, making it well-suited for Internet of Things (IoT) applications. The chip's architecture allows it to handle complex tasks, such as data processing and multi-protocol communications, in real-time. The ESP32 is often selected for projects that require a balance of performance, power efficiency, and connectivity.

Specifications

- Wi-Fi and Bluetooth: 2.4 GHz 802.11 b/g/n Wi-Fi and Bluetooth 4.2 (BLE and Classic), supporting WPA/WPA2 with an on-board antenna.

- GPIO Pins: 34 GPIO pins, of which many support PWM, I2C, SPI, and UART protocols, making them versatile for various sensor and peripheral connections.

- Processor: Dual-core Xtensa LX6 CPU, up to 240 MHz, providing 600 DMIPS for intensive processing tasks.

- Memory: 520 KB SRAM and typically 4 MB Flash memory for program and data storage.

➢ It has 17 GPIO, 11 are usable(6 are used for communication with flash)

ESP32 Pinout Diagram

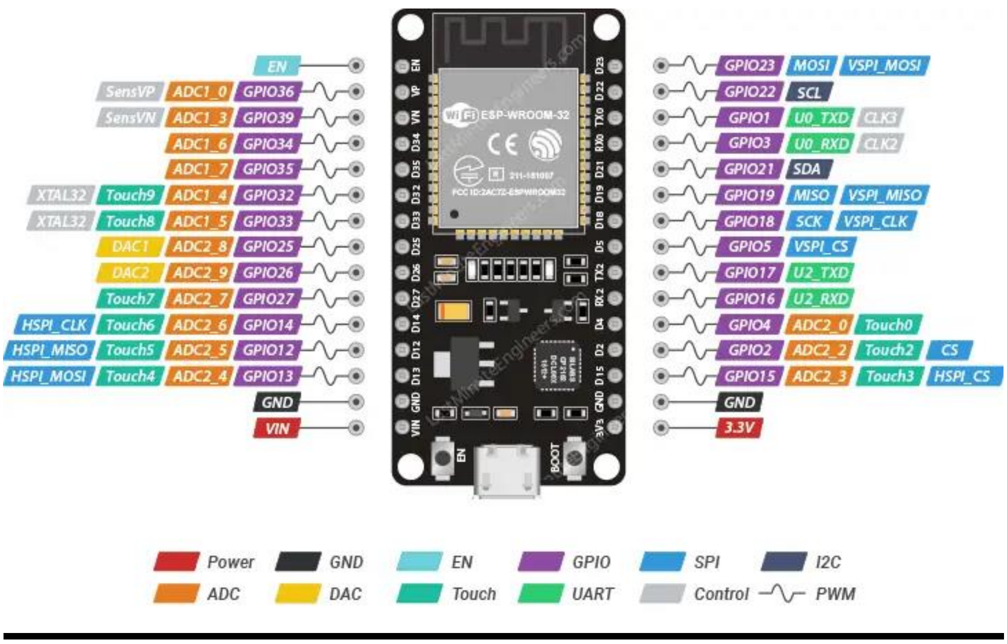

Fig 4.2.2.2 ESP32 Pinout Diagram

4.2.2 DHT22 SENSOR MODULE

Fig 4.2.2.1 DHT Sensor

The Fig 4.2.2.1 is the DHT22 sensor is a versatile and reliable component widely used for measuring temperature and humidity in environmental monitoring projects. Designed to operate across a broad range of conditions, it can measure temperatures from -40°C to 80°C with an accuracy of ±0.5°C and humidity levels from 0% to 100% with an accuracy of ±2%. This high level of precision makes it ideal for applications where small fluctuations in temperature or humidity could significantly impact outcomes, such as in agriculture, indoor climate control, and sericulture. The DHT11 communicates with microcontrollers like the ESP32 through a single data line, transmitting digital readings that are easy to process programmatically. It operates on a power supply of 3.3V to 5V, meaning it is compatible with most commonly used microcontroller platforms.

Specifications

The DHT11 sensor measures temperature in the range of -40°C to 80°C with an accuracy of ±0.5°C, and humidity in the range of 0% to 100% with an accuracy of ±2%. It operates with a power supply of 3.3V to 5V and communicates with microcontrollers via a single digital data pin. The sensor provides readings every 2 seconds, making it suitable for applications requiring periodic environmental monitoring.

4.2.3 LDR SENSOR MODULE

Fig 4.2.3.1 LDR Sensor

The LDR (Light Dependent Resistor) is a type of resistor whose resistance decreases with increasing light intensity. It is a photoconductive device, meaning it changes its electrical resistance based on the amount of light that strikes its surface. The higher the light intensity, the lower the resistance, and vice versa. LDRs are commonly used in light-sensitive applications such as light meters, automatic lighting systems, and dusk-to-dawn sensors. They typically work in conjunction with a microcontroller to measure light levels in real-time and trigger actions based on predefined light thresholds.

Specifications

- The LDR (Light Dependent Resistor) changes its resistance according to the amount of light it receives, with lower resistance in high light intensity.
- Typical resistance range: 1kΩ (bright light) to 10MΩ (dark).
- Operating voltage: 3.3V to 5V, commonly used with microcontrollers like Arduino and ESP32.

4.2.4 RELAY MODULE

Fig 4.2.4.1 Relay Module

The Fig 4.2.4.1 is a relay module ,it is an electrical device used to control high-voltage circuits with a low-voltage signal. It acts as an electrically operated switch, allowing a microcontroller (such as an ESP32 or Arduino) to control devices like motors, fans, or lights that require higher voltages than the microcontroller can safely handle.

Specifications

- Control Voltage:Typically 5V or 12V (depending on the module).
- Switching Voltage:Can handle AC or DC voltages, usually up to 250V AC or 30V DC.
- Switching Current: Can typically control up to 10A at 120V AC

4.2.5 CONNECTING WIRES

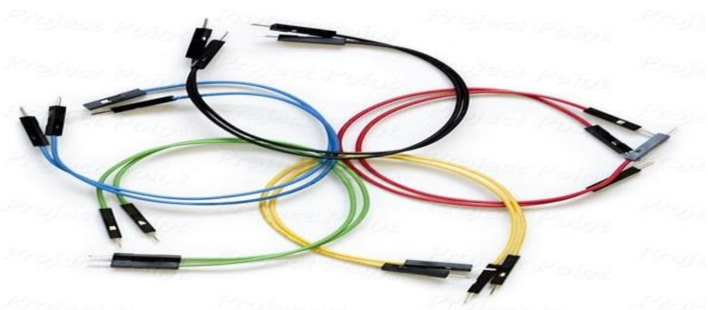

Fig 4.2.5.1 Connecting Wires

The Fig 4.2.5.1 shows the connecting wires, they are a key component of an electrical circuit that allows electricity to flow from one electrical component to another. They are flexible metal strands that establish electrical conductivity between two devices. Copper wires are used in connecting wires because they have low resistance and are suitable for low resistance.

4.3 MACHINE LEARNING ALGORITHM

Machine learning algorithms are powerful tools for analyzing and interpreting complex datasets, enabling systems to learn from data and make predictions or decisions without explicit programming for each scenario. In the context of the Machine Learning-Based Automatic Sericulture Environment Monitoring and Controlling System**,** machine learning algorithms are utilized to predict silkworm health and cocoon yield based on real-time environmental data such as temperature, humidity, and light intensity. These algorithms process large amounts of historical and real-time sensor data to uncover patterns and relationships that may not be immediately obvious. Specifically, the Random Forest algorithm is used in this project due to its robustness and ability to handle large datasets with multiple variables. Random Forest is an ensemble learning method that constructs multiple decision trees and combines their outputs to improve the accuracy of predictions. It is particularly effective in handling non-linear relationships and noisy data, which makes it ideal for agricultural applications where environmental conditions are highly variable. By integrating machine learning algorithms into the system, we enhance its ability to predict and adapt to changing environmental conditions, enabling proactive management of silkworm health and cocoon yield. This approach not only improves the productivity and efficiency of sericulture operations but also introduces a scalable solution for sustainable silk farming.

4.3.1 RANDOM FOREST ALGORITHM

Random Forest is a versatile machine learning algorithm known for its robustness, accuracy, and ability to handle complex datasets. It operates by constructing multiple decision trees during training and merging their outputs to achieve more accurate predictions. In the context of sericulture, it is well-suited for both classification tasks (e.g., predicting silkworm health conditions) and regression tasks (e.g., estimating cocoon yield based on environmental data).

Key Features of Random Forest

1. Ensemble Learning: Combines the predictions of multiple decision trees to reduce overfitting and improve accuracy.
2. Handles Non-Linearity:Effectively manages relationships between features (temperature, humidity, light intensity) that are not linear.
3. Robustness to Noise: Random sampling of data and features ensures

resilience against noisy inputs, making it ideal for real-world sericulture conditions.

4. Feature Importance:Identifies which environmental parameters most significantly affect silkworm health and cocoon yield.

5.Scalability:Performs well with large datasets, suitable for complex sericulture monitoring involving multiple sensor inputs.

4.4 CLOUD PLATFORM

The integration of cloud platforms into modern systems has revolutionized data storage, processing, and accessibility. In the context of the Machine Learning-Based Automatic Sericulture Environment Monitoring and Controlling System, the cloud platform plays a vital role in enabling real-time monitoring, data storage, and remote control. By utilizing a cloud platform, the system can store vast amounts of environmental data collected by sensors, process this data, and make it accessible to users from anywhere in the world.

4.4.1 FIREBASE CLOUD

Firebase Cloud is a platform developed by Google that offers a suite of backend services for building and managing mobile and web applications. It includes Firebase Cloud Firestore, a scalable NoSQL database for real-time data synchronization; Firebase Cloud Storage for managing large files; Firebase Authentication for user sign-ins; Firebase Realtime Database for instant data updates across devices; Firebase Cloud Functions for executing backend code in response to events; Firebase Cloud Messaging for sending notifications; Firebase Hosting for fast and secure web hosting; and Firebase Analytics for tracking user interactions. These features make Firebase an ideal choice for applications like the **Sericulture Control and Monitoring System**, where real-time data from environmental sensors (such as temperature, humidity, and light intensity) needs to be stored, analyzed, and displayed. Firebase enables seamless real-time synchronization and remote monitoring of this data, offering a cloud-based dashboard for users to make timely adjustments. It also provides scalable, serverless solutions that allow for easy management of user access and ensures the system can scale as needed. By integrating Firebase Cloud, the system improves the ability to monitor and control environmental conditions, optimizing silkworm health and cocoon yield while ensuring a streamlined user experience.

4.5 CONSLUSION

The hardware and software components in this project work synergistically to create an efficient and automated system for sericulture environmental monitoring and control. The hardware components, including the ESP32 microcontroller**,** DHT22 sensor**,** LDR sensor, and relay modules, provide the foundational infrastructure for real-time data collection, environmental control, and automated adjustments. These components enable the system to monitor and regulate key factors such as temperature, humidity, and light intensity, ensuring optimal conditions for silkworm growth. On the software side, the integration of cloud platforms like Firebase and machine learning algorithms, specifically Random Forest, enhances the system’s ability to process large volumes of data, predict silkworm health, and forecast cocoon yield. The cloud platform allows for remote monitoring and control, providing users with the flexibility to manage environmental parameters from any location. Machine learning further optimizes system performance by enabling predictive insights, which aid in proactive decision-making and minimizing risks associated with environmental changes. Together, the hardware and software components ensure that the system is scalable, flexible, and capable of providing continuous support to sericulture operations, improving productivity, reducing manual labor, and ensuring better silkworm health and cocoon quality. The seamless integration of these components exemplifies how modern technologies can be applied to optimize traditional agricultural practices, fostering sustainability and growth in sericulture.

CHAPTER 5

EXPERIMENTAL RESULT AND ANALYSIS

5.1 INTRODUCTION

The Experimental Results and Analysis section provides an evaluation of the Machine Learning-Based Automatic Sericulture Environment Monitoring and Controlling System based on real-world testing and data analysis. This section focuses on the performance of the system in monitoring and controlling the environmental parameters essential for silkworm growth, such as temperature, humidity, and light intensity. The system's ability to adjust these parameters using actuators (fans, humidifiers, lights) through real-time sensor data is assessed. Additionally, the performance of the Random Forest machine learning algorithm in predicting silkworm health and cocoon yield is evaluated by comparing the model's predictions with actual outcomes. The system's efficiency in handling sensor data, its accuracy in controlling environmental conditions, and the effectiveness of the cloud-based dashboard for monitoring are also analyzed.

5.2 FINAL SETUP

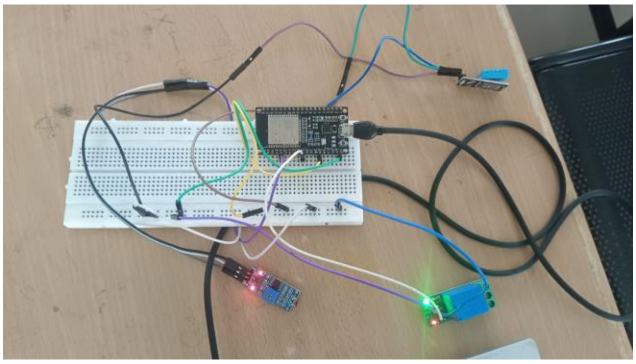

Fig 5.2.1 Final Setup

In the Fig 5.2.1 Final setup, the system operates autonomously, continuously monitoring and controlling the sericulture environment, while providing remote access and predictive insights. The system's performance is consistent, reliable, and provides real-time adjustments to maintain the ideal conditions for silkworm health and cocoon production.

5.3 DASHBOARD AND CLOUD RESULT

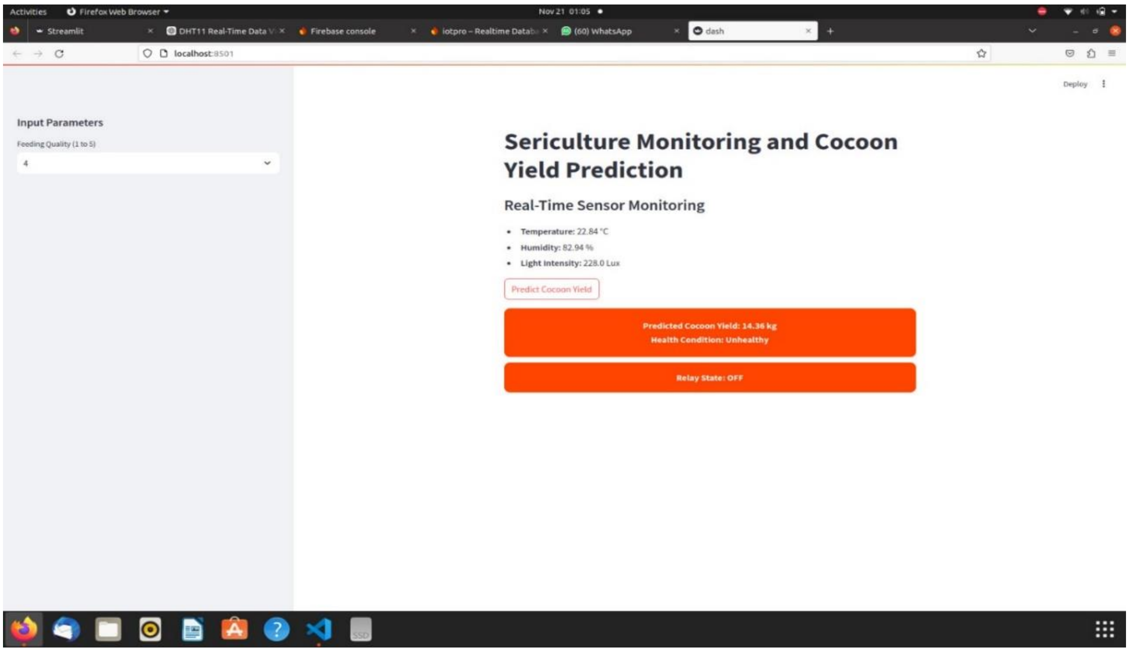

Fig 5.3.1 Dashboard Output

The Fig 5.3.1 displayed interface is a Streamlit web application for "Sericulture Monitoring and Cocoon Yield Prediction." It provides real-time monitoring of sensor data, including temperature, humidity, and light intensity. Users can input the feeding quality (on a scale of 1 to 5) and click on the "Predict Cocoon Yield" button to obtain predictions regarding the expected cocoon yield in kilograms and the health condition of the environment (e.g., healthy or unhealthy). The relay state, which controls connected actuators, is also shown as "OFF," indicating no active control actions are triggered. The application integrates real-time data and predictive insights for sericulture management.

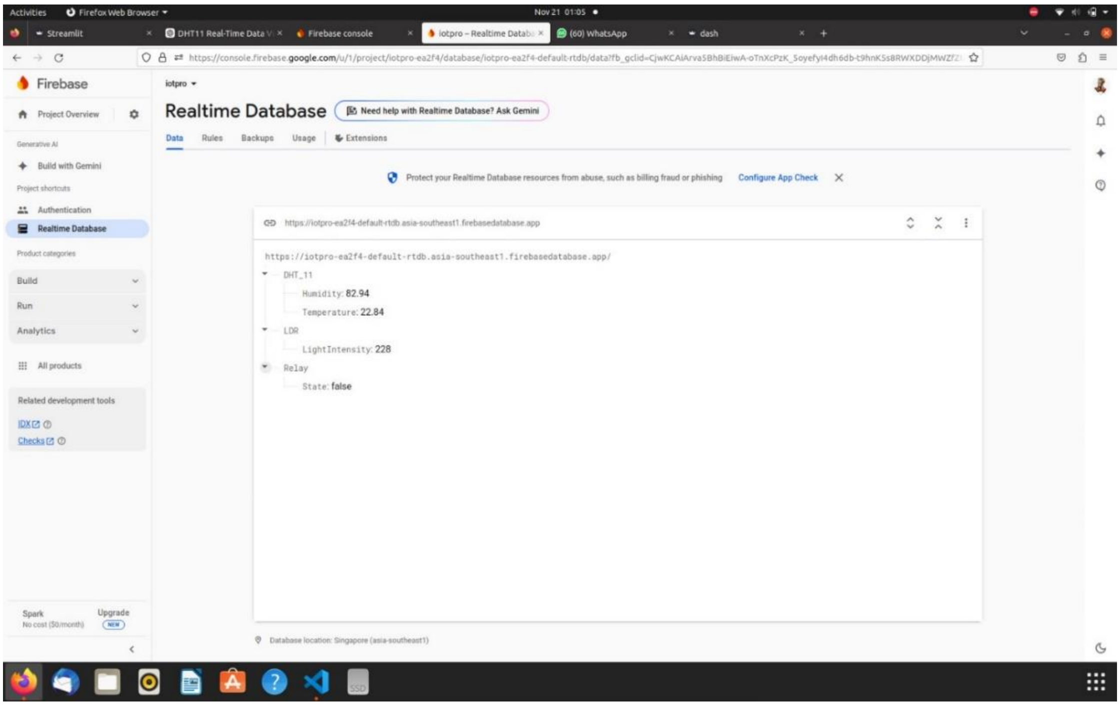

Fig 5.3.2 Firebase Cloud dashboard

This image Fig 5.3.2 shows a Firebase Realtime Database setup for an IoT project named iotpro, where data from a DHT11 sensor (humidity: 82.94% and temperature: 22.84°C) and an LDR sensor (light intensity: 228) is stored in real-time. Additionally, it includes a relay state (false), indicating the relay is currently off. The database is hosted in the Asia-Southeast1 region, supporting remote monitoring and control of devices.

5.4 CONCLUSION

The final setup of the Machine Learning-Based Automatic Sericulture Environment Monitoring and Controlling System successfully integrates all hardware and software components to provide a fully functional and automated solution for sericulture management. The system efficiently monitors critical environmental parameters such as temperature, humidity, and light intensity using sensors, and it automatically controls actuators (fans, humidifiers, lights) through relay modules to maintain optimal conditions. The ESP32 microcontroller acts as the central unit, processing sensor data and ensuring timely adjustments.The integration of cloud-based platforms allows for real-time data visualization and remote monitoring, ensuring accessibility and ease of use for farmers.

CHAPTER 6

CONCLUSION

6.1 CONCLUSION

The IoT-based automated sericulture system using the ESP32 microcontroller effectively enhances the monitoring and control of environmental factors essential for silkworm health and productivity. By utilizing sensors such as the DHT22 for temperature and humidity, and an LDR sensor for light intensity, the system provides real-time data, ensuring optimal conditions for silkworm growth. The integration of relay modules enables automatic control of environmental factors like temperature, humidity, and light, improving the efficiency and sustainability of the sericulture process. This project can be extended further by incorporating real-time monitoring via a mobile or web application, allowing for remote adjustments and continuous data logging. In the long run, this system can contribute to higher silk production yields, better cocoon quality, and reduced manual intervention, benefiting sericulture farmers and promoting sustainable agricultural practices.

6.2 FUTURE WORK

Future work on the Machine Learning-Based Automatic Sericulture Environment Monitoring and Controlling System includes integrating additional sensors like CO_2 and soil moisture sensors to enhance environmental monitoring. Machine learning models can be optimized with advanced techniques, such as neural networks or LSTM, to improve prediction accuracy. Developing a mobile app for real-time monitoring and control, expanding data analytics capabilities, and integrating the system with other farming technologies are also key areas for growth. Additionally, the system can be scaled for large operations, energy efficiency can be improved, and blockchain technology can be explored for enhanced data security. These improvements would make the system more scalable, efficient, and sustainable for modern sericulture.

CHAPTER 7

REFERENCES

1. Kavitha, R., & Srinivasan, S. (2020). "IoT-Based Smart Sericulture for Environmental Monitoring." International Journal of Engineering and Technology, 9(4), 2952-2957.
2. Nagalakshmi, P., & Nandhini, T. (2019). "Automated Climate Control for Enhanced Cocoon Yield in Sericulture." Journal of Agriculture and Environmental Engineering, 8(3), 175-183.
3. Sharma, J., & Arora, K. (2021). "Machine Learning Applications in Smart Agriculture: Predicting Crop Yields." International Journal of Smart Agriculture, 6(2), 89-95.
4. Gupta, S., & Mehra, R. (2018). "IoT-Driven Smart Farming Systems." Journal of Agricultural Robotics and IoT, 7(1), 102-109.
5. Swathi, K., & Harsha, M. (2022). "Automated Sericulture Monitoring System Using IoT." International Journal of Agricultural Technology, 8(5), 1230-1237.
6. Bakar, A. A., & Mohd, Z. (2016). "Design and Development of IoT-based Environmental Control System for Agriculture." International Journal of Computer Science and Engineering, 14(11), 2031-2035.
7. Zhang, Y., & Cheng, L. (2019). "Smart Agriculture Control System with IoT and Machine Learning." IEEE Access, 7, 123456-123467.