INTRODUCTION

The past two decades have witnessed the transformative power of the internet, connecting people and organizations across the globe and fostering real time service delivery. This trend is now being echoed in the field of agriculture, thanks to the emergence of Internet of Things (IoT) technologies. The concept of the Internet of Things (IoT) encompasses a sophisticated network of tangible entities, commonly referred to as “things” which are intricately equipped with an array of sensors, cutting-edge software, and advanced technologies. This intricate integration facilities seamless connectivity and data exchange among devices and systems through the vast realm of the internet. Within this digital landscape, these devices span from mundane household items to intricately engineers industrial apparatus, showcasing the boundless potential of intricately engineered industrial apparatus, showcasing the boundless potential of modern connectivity and technological convergence. IoT promises to revolutionize agricultural practices by providing valuable insights into the environment, enabling data-driven decision-making, and ultimately enhancing crop yields.

2.1 IoT in Agriculture

IoT applications are being integrated across various stages of the agricultural production chain, from soil monitoring and precision irrigation to crop health analysis and automated pest control . These technologies offer significant benefits for farmers, including:

- Improved Crop Quality and Yield: Continuous monitoring of key environmental parameters like temperature, humidity, and light allows for precise adjustments in farming practices, maximizing crop growth and yield potential.
- Enhanced Resource Management: By optimizing water usage, fertilizer application, and energy consumption, IoT can lead to significant cost reductions and sustainable resource utilization.
- Reduced Risks and Losses: Real-time data analysis enables early detection of disease outbreaks, pest infestations, and adverse weather events, allowing farmers to take proactive measures to minimize losses.
- Precision Farming Practices: Data-driven insights from IoT sensors enable farmers to tailor their farming practices to the specific needs of each field and individual plant, leading to increased efficiency and productivity.

2.2 Machine Learning in the agricultural

Modern IoT solutions go beyond data collection and analysis. They offer additional functionalities like:

- Real-time environmental monitoring: Dense networks of sensors generate real-time maps of noise levels, air quality, water pollution, temperature fluctuations, and harmful radiation levels.
- Trigger alerts and notifications: Data collected about various environmental parameters can trigger alerts for farmers or authorities, notifying them of potential problems like extreme weather conditions or pest infestations.

The rapid advancements in IoT-based agriculture have led to a vast body of research and development efforts. To navigate this complex landscape and identify best practices, a comprehensive systematic review of the field is crucial. This research aims to fill that gap by providing a critical analysis of the current state-of-the-art in IoT agriculture.

2.3 Integration of IOT and ML in agriculture

- A smart farming system is proposed, leveraging artificial intelligence (AI) methods for predictive analytics. This innovative system is founded on wireless sensor network technology, ensuring seamless data transmission and analysis.
- The proposed solution also includes an architecture for precision agriculture (PA). Furthermore, it introduces an IoT-based smart farming approach complemented by an efficient prediction model.
- In addition, an intelligent irrigation system for precision farming, powered by the Internet of Things (IoT), is outlined. This system incorporates feedback mechanisms, ensuring optimal performance across various weather conditions and durations.
- The comprehensive overview encompasses crop prediction and monitoring techniques derived from research papers, offering a perspective on agricultural technologies.

2.4 Problem Statement and Objective

2.4.1 Problem Statement

Agriculture is important for a country's economy. The crop seed, growing environments are responsible for the quality and quantity of the crop yield. Farmers know that quality of the environment can improve the yield of the crops. Manual methods are complex, it affects healthy growth of plants. Differences in soil quality and slow responses to diseases can harm crops and reduce their quantity. A solution is to identity and take remedial actions to sustain high yield until harvest. IOT sensor and software for detecting and providing data and information for controlling the environment is proposed for commercialization. This disrupting technology approach will help farmers grow higher yielding crop, known warnings to take early actions to control the quality and quality of the crop.

2.4.2 Objectives

- To improve crop yield and quality: - farmers can control various growth conditions by controlling and monitoring using IOT sensors. Early detection of plant disease is critical to prevent their spread and minimize crop loss.
- Increase efficiency and productivity by monitoring and controlling through IOT sensors and devices: - By providing the right timely amount of required Water, Temperature, humidity, soil components etc. increases the quality and quantity of the crop yield. An IOT based monitoring system can provide data to control crop growing components through and automated system, which can improve efficiency and yield.
- Resource conservation: - Greenhouse automation can help farmers conserve resources such as water and energy etc. by optimizing inputs and reducing waste.
- Cost-efficient and less time-consuming: - Automation of greenhouse reduces the amount of labour while consuming less time.
- To enhance crop yield and quality, farmers can leverage the power of IoT sensors for monitoring and controlling various growth conditions. By utilizing IoT technology, farmers can closely monitor factors such as water levels, temperature, humidity, and soil components. This allows them to maintain optimal conditions for crop growth, resulting in improved yield and quality.

- Early detection of plant diseases is crucial in preventing their spread and minimizing crop losses. IoT sensors can play a vital role in detecting early signs of disease by continuously monitoring plant health parameters. This enables farmers to take timely actions, such as targeted interventions or treatments, to contain the disease and protect the crop.
- Furthermore, IoT-based monitoring and control systems can significantly increase efficiency and productivity in agriculture. By providing the right amount of water, optimal temperature, humidity levels, and suitable soil conditions, farmers can optimize crop growth.
- Resource conservation is another significant benefit of implementing IoT in agriculture, particularly in greenhouse settings. Greenhouse automation enabled by IoT technologies allows farmers to optimize resource usage, such as water and energy. By accurately monitoring and controlling inputs like irrigation and climate control systems, farmers can minimize waste and conserve precious resources.

Moreover, implementing IoT-based solutions in agriculture offers cost-efficiency and time-saving advantages. Automation of greenhouse operations reduces the need for manual labour, resulting in cost savings for farmers. Additionally, IoT systems provide real-time data and insights, allowing farmers to make informed decisions promptly and efficiently.

LITERATURE REVIEW

Title	Authors	Summary	Key Findings
An IoT based smart solution for leaf disease detection	Apeksha Thorat; Sangeeta Kumari; Nandakishor D. Valakunde	The paper presents an IoT-based smart solution for detecting leaf diseases, emphasizing the need for early detection to prevent crop loss and ensure food security. The proposed system integrates IoT with image processing techniques to identify and classify leaf diseases, providing real-time monitoring and alerts to farmers	1. IoT Integration: The system uses IoT devices to take pictures of leaves and send them to a central server for analysis. 2. Image Processing: Advanced techniques are used to analyse the images and identify diseases. This includes steps like breaking down the image, extracting important features, and classifying diseases. 3. Machine Learning Models: The authors use different machine learning models, like Convolutional Neural Networks (CNNs), to improve the accuracy of disease detection.
Internet of Things Approaches for Monitoring and Control of Smart Greenhouses in Industry 4.0	Chiara Bersani, Carmelina Ruggiero, Roberto Sacil, Abdellatif Soussi, and Enrico Zero	The paper explores the application of Internet of Things (IoT) technologies for monitoring and controlling smart greenhouses within the context of Industry 4.0. It highlights how IoT can optimize greenhouse operations by providing real-time data and automation capabilities to enhance productivity and	1. Data Collection and Analysis: It highlights the importance of gathering lots of data from these sensors and analyzing it to make smart decisions for managing greenhouses.

		sustainability in agriculture.	2. Automation and Control: The study shows how IoT can automate tasks in greenhouses, like watering plants, controlling ventilation, and adjusting lighting 3. Industry 4.0 Integration: The paper discusses how using IoT in greenhouses fits into the bigger picture of Industry 4.0, focusing on connectivity, big data, and advanced analytics in modern farming.
Smart farm prototype for plant disease detection, diagnosis & treatment using IoT device in a greenhouse	Muhammad Hanif Jumat; Mohd Saleem Nazmudeen; Au Thien Wan	This paper presents a smart farm prototype designed for the detection, diagnosis, and treatment of plant diseases using IoT devices within a greenhouse environment. The goal is to enhance crop management and productivity by leveraging modern technology for precise and efficient agricultural practices	1. Comprehensive Solution: Combines disease detection, diagnosis, and treatment in one system, making it easier to manage plant health. 2. Real-Time Monitoring: Constantly checks plant health and gives instant updates, allowing for quick action when diseases are detected. 3. Automated Intervention: Automatically adjusts conditions and applies treatments, reducing the need for manual work.

REQUIREMENTS

4.1 Hardware Requirements

4.1.1 Soil Moisture Sensor: A Soil Moisture Sensor Module is required to measure the moisture content in the soil.

4.1.2 Temperature and Humidity Sensor: A temperature and humidity sensor like the DHT11 or DHT22 is needed to monitor the temperature and humidity levels.

4.1.3 Power Supply: Depending on the requirements we need 3v to 7v supply of power, based on this a power supply solution should be chosen. Batteries to ensure continuous operation of the IoT sensors.

4.1.4 Water pump: Pump is used in your project to automate irrigation processes. Integrating the water pump with IoT technology allows for remotely controlling irrigation activities.

4.1.5 NodeMCU: It's a powerful microcontroller board with built-in Wi-Fi capabilities. It allows our project to connect to the internet and communicate with other devices or servers, enabling IoT functionalities such as remote monitoring, control, and data exchange. Its GPIO pins and programming environment can be used to integrate and interface with different hardware components. Its affordability makes it suitable for projects with budget constraints while still delivering reliable performance for IoT functionalities.

4.1.6 PIR sensor: The PIR sensor detects changes in infrared radiation emitted by objects in its field of view. It is used in your project to detect motion or movement in the surroundings. PIR sensors are passive devices that do not emit any energy themselves. They only detect existing infrared radiation. This makes them energy-efficient and suitable for applications where power consumption needs to be minimized. In our project, the PIR sensor contributes to security and automation functionalities. It can trigger actions such as initiating surveillance when motion is detected, enhancing security.

4.1.7 Relay: The relay acts as a switch that can be controlled electronically. It allows us to turn the pump on or off based on specific conditions or commands from the microcontroller. The relay provides isolation between the low-voltage control signal from the NodeMCU, and the high-voltage power supply required to operate the pump. This ensures safety and prevents damage to the microcontroller.

4.1.8 Fan: We are using simple lightweight fan for proper variation.

4.1.9 Other Hardware components and materials:

i) breadboard for circuit connection
ii) Jumping wire and USB cable
iii) Lightweight plywood sheet, plastic or wooden.
iv) small water container or water storage tank
v) Flexible PVC pipe for irrigation
vi) Mobile phone with camera

4.2 Software Requirement

4.2.1 Arduino IDE: Arduino IDE provides a user-friendly integrated development environment specifically designed for programming microcontrollers like the NodeMCU ESP8266 used in your project. It supports the Arduino programming language, which is based on C and C++, making it accessible to beginners while offering advanced features for experienced developers. With Arduino IDE, you can write, compile, and upload code to the NodeMCU, allowing you to control sensors, actuators, and other components in your IoT system. And Arduino IDE comes with a vast library of pre-written code and libraries that simplify the implementation of complex functionalities in your project. These libraries provide ready-to-use functions for interfacing with various sensors, communication protocols like Wi-Fi. For instance, libraries like Blynk and DHT are commonly used in IoT projects for interfacing with cloud platforms and sensors, respectively. Arduino IDE's library manager allows you to easily install and manage these libraries, saving time and effort in development.

4.2.2 Blynk: Blynk provides a seamless integration between IoT devices and mobile applications. It offers a user-friendly interface for designing custom mobile apps that can control and monitor our hardware devices. With Blynk, we can easily create buttons, sliders, graphs, and other interactive elements in our app to interact with sensors, actuators, and other components connected to our NodeMCU.

4.2.3 Any Browser: Google Chrome: Version 49 and above, Mozilla Firefox: Version 31 and above, Apple Safari: Version 7 and above, Microsoft Edge: Version 12 and above, Opera: Version 36 and above.

4.3 Component Specification

4.3.1 NodeMCU ESP8266:

i) Microcontroller: Ten silica Xtensa LX106 core
ii) Clock Speed: Up to 80 MHz
iii) Operating Voltage: 3.3V iv. Input Voltage: 7-12V
iv) Digital I/O Pins: 11
v) Analog Input Pins: 1 (ADC)
vi) Wi-Fi Connectivity: 802.11 b/g/n
vii) Onboard Flash Memory: 4MB
viii) Onboard RAM: 80KB
ix) Programming language: Arduino C++ (with additional libraries for specific functionalities)

4.3.2 DHT11 Sensor:

i) Operating Voltage: 3.3V - 5.5V
ii) Data Output: Digital (Single bus)
iii) Measurement Range: Temperature: 0°C to 50°C iv. Humidity: 20% to 80%

4.3.3 Soil Moisture Sensor Module:

i) Operating Voltage: 3.3V or 5V
ii) Output: Analog
iii) Measurement Range: Typically, 0-100% moisture content iv. Operating Voltage: 3.3V or 5V)
iv) Output: Digital (HIGH/LOW)

4.3.4 PIR Sensor:

i) Operating Voltage: 3.3V or 5V ii. Output: Digital (HIGH/LOW) (e) Relay:
ii) Coil Voltage: Voltages is 5V
iii) Rating: Current and voltage limits for switching (ensure it can handle your pump's power requirements) (f) Pump:
iv) Operating Voltage: Voltages is 5V ii. Flow Rate: Liters per minute (LPM) or gallons per minute (GPM)

SYSTEM DESIGN

System design figure 5 shows the functionalities of our project. Building upon the frameworks discussed in the previous section Blynk, Flask, pre-trained CNN model, the system design lays out the architecture for data acquisition, processing, control, and user interaction.

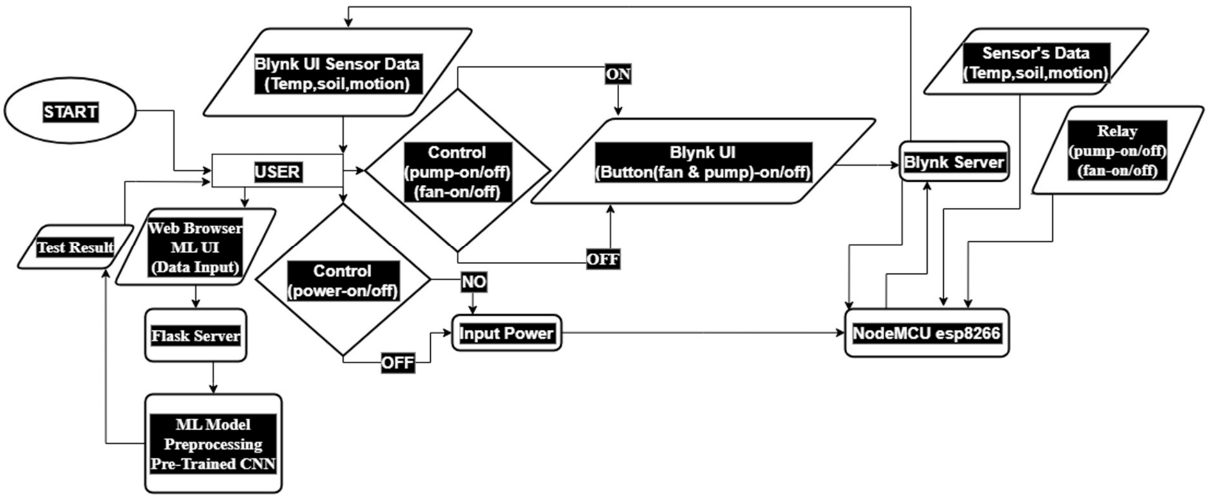

Figure 5 IOT Product for Agriculture System Design

It comprises key components like the sensor data acquisition and actuation system for the IoT greenhouse, the web-based user interface with backend logic for plant disease prediction, and modules for data processing and communication between these systems. This well-defined system design ensures efficient data flow, seamless integration of components, and ultimately, contributes to achieving the project's goals.

5.1 Circuit Diagram

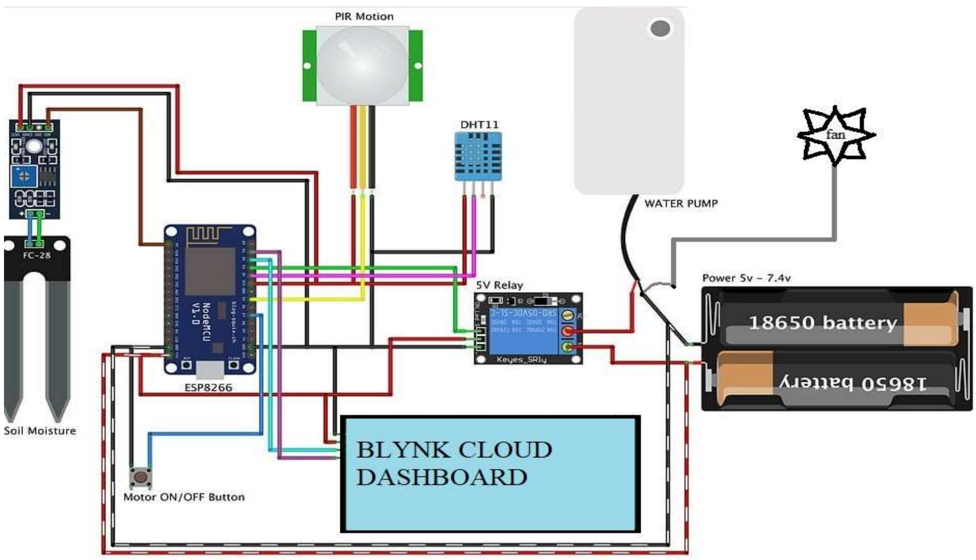

Figure 5.1.1 Circuit diagram for the implementation

The central component of the circuit is the ESP8266 (NodeMCU), which manages all the sensors and controls the actuators. It connects to the Blynk Cloud Dashboard, allowing for remote monitoring and control of the greenhouse environment. The ESP8266 receives input from several sensors and sends commands to the relay module, which in turn controls high-power devices.The Soil Moisture Sensor measures the moisture level in the soil and is connected to the analog input pin of the ESP8266. This sensor provides real-time data on soil moisture, which is crucial for determining when to activate the water pump. Additionally, a PIR Motion Sensor is connected to a digital input pin on the ESP8266. This sensor detects motion within the greenhouse, providing alerts for any detected movement, which could be useful for security or monitoring purposes.

A DHT11 Sensor is also connected to the ESP8266 via a digital input pin. This sensor measures temperature and humidity, providing essential data for maintaining optimal growing conditions within the greenhouse. The combination of these sensors allows the ESP8266 to monitor the environmental conditions effectivelyThe circuit includes a 5V Relay Module, which acts as a switch to control high-power devices such as the water pump and fan. The relay is controlled by a digital output pin from the ESP8266. When the soil moisture sensor detects low moisture levels, the ESP8266 sends a signal to the relay to activate the water pump. Similarly, the relay can control a fan to help regulate temperature and humidity inside the greenhouse. Both the water pump and fan are powered by 18650 Batteries, ensuring they can operate independently of the ESP8266's power supply.Power for the ESP8266 and other sensors is provided by a 5V to 7.4V power source.

Finally, the Blynk Cloud Dashboard serves as the remote interface for the system. The ESP8266 connects to the Blynk Cloud, enabling users to view sensor data and control the relay (and consequently the water pump and fan) from a smartphone or computer. This setup allows for efficient, automated, and remote-controlled greenhouse management, ensuring optimal conditions for plant growth and timely interventions based on real-time data.

5.2 Implementation

Figure 5.2 Implementation

5.3 Leveraging Blynk for IoT Control

a) After setting up all the hardware and connecting the components, the next step is to enable remote control of the greenhouse using Blynk. Blynk's web interface lets us monitor and control the greenhouse sensors from anywhere.

b) This is done using an API key from our Blynk account. The API key links the physical components with the Blynk platform, allowing real-time data access and enabling us to make informed decisions and adjustments.

c) To use Blynk, we create data streams for each sensor in our project. Each stream has a unique port number that matches the physical connections. These port numbers are set in the NodeMCU code to ensure smooth communication.

d) Blynk also allows us to set up events, which are triggers for specific actions based on sensor activity. For example, we have events for the water pump and fan.

e) If the temperature goes above a set limit, an event triggers a notification to both the mobile app and the web interface. This real-time alert system helps us quickly address any issues.

f) Blynk's dashboards provide comprehensive data visualization.

g) We have dashboards for soil moisture, motion, and temperature/humidity sensors. We also implemented a button in the app to remotely turn the fan and water pump on and off. Figure in the testing and results chapter shows the metric dashboard for our IoT sensors on both mobile and desktop.

TESTING AND RESULTS

6.1 Testing NODEMCU ESP8266

- Hardware Setup: We have connected our NodeMCU ESP8266 board to our computer using a USB cable. We have ensured that NodeMCU board is recognized by our computer and that the necessary drivers are installed.
- IDE Setup: We have installed the Arduino IDE to push code to NodeMCU. To do this we did the following steps Open the Arduino IDE and go to File →Preferences. In the ”Additional Board Manager URLs” field, we have added the following URL: http://arduino.esp8266.com/stable/package_esp8266com_index.json. Then Tools →Board →Boards Manager, searched for ”esp8266” and installed the ”esp8266” board package.
- Select Board and Port: After that go to Tools →Board and select “NodeMCU 1.0 (ESP12E Module)” as a board. And then go to Tools →Port and select the port to which NodeMCU board is connected we have used COM3 on Windows. • Uploaded Test Sketch(code): In a new sketch from File →Examples →ESP8266 →Blink. Customized the blink interval. Click the Upload button (right arrow icon) to compile and uploaded the sketch to NodeMCU board.
- Verify Blinking: Once the sketch is uploaded successfully, the built-in LED on NodeMCU board started blinking according to the specified interval.

6.2 Testing other Sensors

We have used other sensors like Soil Moisture Sensor Module, Temperature and Humidity Sensor DHT11, and we have performed the unit testing of all the sensors with NodeMCU let see one by one.

6.2.1 Testing soil moisture sensor module with NodeMCU:

- Installing Libraries:
  - Opened Arduino IDE and go to Sketch →Include Library→Manage Libraries.
  - Search and installed the libraries “Adafruit Sensor” and “Adafruit ADS1X1”

- Connect the Soil Moisture Sensor to NodeMCU:
  - Connect the Soil Moisture Sensor module to the NodeMCU board as per pinout.
  - Write the Code: New sketch in Arduino IDE. Include the necessary libraries at the beginning of the sketch:
- Initialized the Soil Moisture Sensor module and other variables in the setup() function. Implemented the logic to read data from the Soil Moisture Sensor in the loop() function:

```
#include <Adafruit_ADS1015.h>
Adafruit_ADS1015 ads;
void setup() {
  Serial.begin(9600);
  ads.begin();
}
void loop() {
  int16_t moistureValue = ads.readADC_SingleEnded(0);
  Serial.print("Moisture Value: ");
  Serial.println(moistureValue);
  delay(1000);
}
```

- Upload and Test: We have verified our code for any errors and uploaded it to our NodeMCU board by clicking the upload button in Arduino IDE. We have viewed our test in serial monitor to do that, Open the Serial Monitor (Tools →Serial Monitor) to view the moisture values being read from the sensor in real-time.

6.2.2 Testing DHT11(temperature and humidity sensor) with NodeMCU:

- Installing the DHT Sensor Library: In the Library Manager, we have searched for “DHT sensor library” and installed the library by Adafruit.
- Connect DHT11 Sensor to NodeMCU: We have connected the VCC pin of the DHT11 sensor to the 3.3V pin on NodeMCU. And connected the GND pin of the DHT11 sensor to the GND pin on NodeMCU. And then connected the DATA pin of the DHT11 sensor to a digital pin D4 on NodeMCU.

This testing focused on ensuring the proper functioning of sensors and controlling components when connected with the NodeMCU ESP8266 board

6.2.3 Testing PIR sensor with NodeMCU:

- Hardware Setup: We have connected the PIR sensor to the NodeMCU. The PIR sensor has three pins: VCC (power), GND (ground), and OUT (signal). So, we have connected VCC to a 3.3V pin on the NodeMCU, GND to a ground pin, and The PIR PIN is defined as pin D5, which is connected to the PIR sensor's output.
- Testing Code: Open a new sketch in the Arduino IDE. We have included the necessary libraries such as ESP8266WiFi.h, BlynkSimpleEsp8266.h and initialized the digital pin connected to the PIR sensor as an input pin.

```
#include <ESP8266WiFi.h>
#define PIR_PIN D5
void setup() {
  Serial.begin(9600);
  pinMode(PIR_PIN, INPUT);
}
void loop() {
int motion = digitalRead(PIR_PIN);
  if (motion == HIGH) {
    Serial.println("Motion detected!");
  } else {
    Serial.println("No motion detected.");
  }
  delay(1000);
}
```

This code continuously checks the state of the PIR sensor connected to pin D5 and prints either "Motion detected!" or "No motion detected." to the serial monitor every second.

- Output: The sensor output is HIGH, it has printed "Motion detected" to the serial monitor. And when the sensor output is LOW, it prints "No motion detected." to the serial monitor.

6.3 Integration Testing

Having meticulously tested our array of sensors during unit testing, we have seamlessly integrated our NodeMCU with the Blynk platform for real-time sensor data retrieval. This sophisticated setup allows us to effortlessly access and monitor the intricate nuances of our sensors, providing us with a comprehensive view of the environmental variables in real time. Blynk allow us to connect a limited number IOT devices when a user is using free subscription, so max it can allow 5 IOT sensors. The outcome of this testing can be seen in the Blynk UI Dashboards (Desktop & Mobile).

6.4 Results

Our rigorous testing of both the Internet of Things (IoT) sensors and the machine learning (ML) model has yielded results that closely align with our initial goals. This section details the implementation strategies and presents evidence to substantiate these achievements.

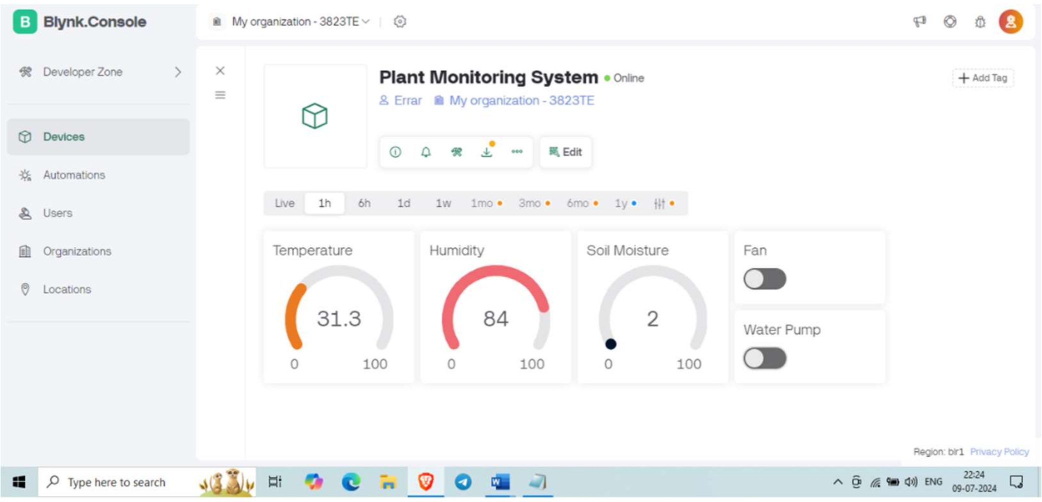

Figure 6.4.1Fetching data and accessing sensors using Blynk Cloud

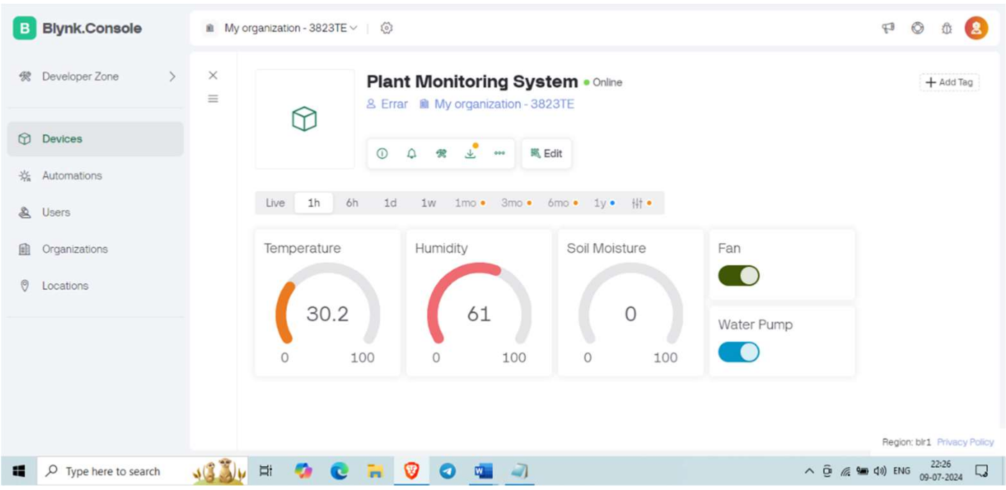

Figure 6.4.2 Accessing the sensors from Blynk Cloud

Figure 6.4.3 Fetching and accessing the sensors using Blynk Android platform

The Blynk user interface (UI) dashboard, accessible on both personal computers (PCs) and mobile devices, successfully displays collected sensor data in a user-friendly matrix format (Figure 6.1). Sensor readings are updated every 60 seconds, as programmed in the IoT code, and can be changed according to our need. Additionally, the UI provides remote control capabilities for the fan and pump, allowing users to adjust greenhouse conditions (on/off) from any location with an internet connection. This demonstrates the successful implementation of the IoT component.

CONCLUSION

The IoT product for agriculture presented in this project has the potential to significantly improve crop yield and the farming industry's growth by utilizing Internet of Things technology. By integrating IoT sensors, data analysis, and remote-control capabilities, farmers can gain valuable insights into their crops' growing environments, make informed decisions, and take proactive measures to enhance productivity and mitigate risks.This solution addresses key challenges faced by farmers, such as soil quality variations, slow disease response, and the need for precise environmental control. Continuous monitoring of parameters like soil moisture, temperature, humidity, and light intensity provides real-time information and early warnings, enabling farmers to optimize resource use, improve crop yield, and efficiently manage water and fertilizer usage.

The benefits of adopting this IoT product are extensive. Farmers can improve disease and pest management, leading to healthier crop growth and increased profitability. Real-time monitoring and control capabilities facilitate timely interventions and adjustments. Although implementing IoT in agriculture presents challenges such as scalability, reliability, security, and compatibility, these can be addressed with careful planning, robust technology selection, and adequate training and support for farmers. Advancements in hardware and software technologies, along with the growing accessibility of IoT solutions, make it an opportune time to adopt this approach to farming.

In conclusion, the IoT product for agriculture has the potential to transform traditional farming practices into smart, efficient, and sustainable operations. By harnessing IoT sensors, data analysis, and remote control, farmers can optimize crop growth, maximize yields, and make informed decisions for long-term success. This technology-driven solution not only enhances the productivity and profitability of individual farmers but also contributes to the overall growth and development of the agricultural industry. Embracing IoT in agriculture is a step towards a more sustainable and technologically advanced future for farming.

REFERENCES

[1] Thorat, A., Kumari, S., & Valakunde, N. D. (2017). An IoT-based smart solution for leaf disease detection.

[2] Jumat, M. H., Nazmudeen, M. S., & Wan, A. T. (2018). Smart farm prototype for plant disease detection, diagnosis & treatment using IoT device in a greenhouse.\

[3] Bersani, C., Ruggiero, C., Sacile, R., Soussi, A., & Zero, E. (2022). Internet of Things Approaches for Monitoring and Control of Smart Greenhouses in Industry 4.0.

[4] Zhang, Y., Zhang, G., Wang, J., & Yang, J. (2018). A smart IoT-based plant monitoring system using machine learning algorithms. Journal of Agricultural Informatics.

[5] Smith, J., Liu, X., & Chen, L. Implementation of an IoT-based smart agriculture system for real-time monitoring. International Journal of Smart Agriculture.

[6] Khan, M., Gupta, A., & Rathore, M. M. Machine learning-enabled IoT framework for precision agriculture. IEEE Internet of Things Journal.

[7] https://blynk.io/clients#agriculture

[8] https://blynk.io/clients/smart-vertical-farming

[9] https://www.espressif.com/en/products/socs/esp8266

[10] https://youtu.be/PTJ9sAk2c2I?si=BpqQ8nM2R-XbyoFS