SMART GREENHOUSE MONITORING AND CONTROL SYSTEM

Mangesh Ramesh Shinde*1, Onkar Mahadev Chougale*2, Sanket Dnyaneshwar Rajgude*3, Prof. Ayubkhan Mulani*4

*1,2,3,4Dept. Electronics And Telecommunications, Keystone School Of Engineering,
Pune, Maharashtra, India

DOI : https://www.doi.org/10.56726/IRJMETS37604

ABSTRACT

Smart Greenhouse Monitoring and Control Systems have become increasingly popular due to their ability to optimize plant growth and improve crop yields. In this study, we developed a Smart Greenhouse Monitoring and Control System using a Raspberry Pi 3, DHT11 sensor, soil moisture sensor, LDR sensor, fan, motor pump, LED light, and web-camera. The system continuously monitored and controlled the environmental conditions within the greenhouse, including temperature, humidity, soil moisture, and light intensity, while also providing live feed of the greenhouse on a web page. We evaluated the performance of the system in a greenhouse setting and found that it effectively maintained the desired environmental conditions for plant growth. The results demonstrate the potential of Smart Greenhouse Monitoring and Control Systems to improve the efficiency and productivity of greenhouse agriculture.

Keywords: Raspberry pi 3, Web Based Monitoring, DHT11 Sensor, Soil Moisture Sensor, LDR Sensor, Greenhouse Technology, Smart Greenhouse, Agriculture, Environmental Monitoring

I. INTRODUCTION

Greenhouse agriculture is a widely adopted practice for growing crops in controlled environments, particularly in regions with harsh weather conditions or limited land availability. However, maintaining optimal environmental conditions for plant growth in greenhouses can be challenging and often requires a significant amount of manual intervention. The use of technology such as Smart Greenhouse Monitoring and Control Systems can help automate and optimize the greenhouse environment, leading to improved crop yields and reduced labor costs. A Smart Greenhouse Monitoring and Control System typically consists of various sensors and control mechanisms that monitor and adjust environmental factors such as temperature, humidity, light intensity, and soil moisture. These systems use data analysis and decision-making algorithms to adjust environmental conditions in real-time, ensuring that the optimal conditions for plant growth are maintained. Additionally, many Smart Greenhouse Monitoring and Control Systems are web-based, allowing growers to remotely monitor and control the greenhouse environment from anywhere with an internet connection. In this study, we developed a Smart Greenhouse Monitoring and Control System using a Raspberry Pi 3, DHT11 sensor, soil moisture sensor, LDR sensor, fan, motor pump, LED light, and web-camera. The system continuously monitored and controlled the environmental conditions within the greenhouse, while also providing live feed of the greenhouse on a web page. We evaluated the performance of the system in a greenhouse setting and found that it effectively maintained the desired environmental conditions for plant growth. This paper presents the details of our Smart Greenhouse Monitoring and Control System, along with an evaluation of its performance and potential applications in greenhouse agriculture.

II. LITERATURE SURVEY

[1] One such study by Liu et al. (2019) used a Smart Greenhouse Monitoring and Control System to optimize environmental conditions for the growth of cherry tomatoes. The system used a combination of sensors, actuators, and control algorithms to maintain optimal conditions for plant growth, leading to a 24% increase in cherry tomato yield compared to traditional greenhouse methods.

[2] Another study by Cha et al. (2020) used a Smart Greenhouse Monitoring and Control System to grow lettuce in a vertical farming system. The system used various sensors to monitor environmental conditions and a control algorithm to adjust light intensity and nutrient supply, resulting in a 70% increase in lettuce yield compared to traditional growing methods.
A study by Jaiswal et al. (2020) developed a Smart Greenhouse Monitoring and Control System using an IoT-based approach, which allowed for remote monitoring and control of the greenhouse environment. The system used sensors to monitor temperature, humidity, light intensity, and soil moisture, and an automatic control algorithm to adjust these factors in real-time. The study found that the system was able to maintain optimal environmental conditions for plant growth, leading to improved crop yields and reduced labor costs.

In a study by Al-Ali et al. (2020), a Smart Greenhouse Monitoring and Control System was developed using a combination of a Raspberry Pi, Arduino, and various sensors to monitor and control the greenhouse environment. The system was able to maintain optimal environmental conditions for plant growth, leading to improved crop yields and reduced water usage. The study also found that the system was cost-effective, making it a viable option for small-scale farmers.

A study by Raza et al. (2021) developed a Smart Greenhouse Monitoring and Control System using a combination of machine learning algorithms and IoT-based sensors. The system used sensors to monitor environmental conditions, and machine learning algorithms to predict future environmental conditions and adjust the greenhouse environment accordingly. The study found that the system was able to maintain optimal environmental conditions for plant growth, leading to improved crop yields and reduced energy usage.

III. HARDWARE DESCRIPTION

**Raspberry Pi 3**
This serves as the main controller for the system and is responsible for collecting data from various sensors, processing the data, and controlling the various actuators in the greenhouse.

**DHT11 Sensor**
This sensor measures temperature and humidity inside the greenhouse and sends the data to the Raspberry Pi for processing.

**Soil Moisture Sensor**
This sensor measures the moisture level of the soil and sends the data to the Raspberry Pi for processing.

**LDR Sensor**
This sensor measures the light intensity inside the greenhouse and sends the data to the Raspberry Pi for processing.
Fan
This is an actuator that can be turned on and off by the Raspberry Pi to control the temperature and humidity inside the greenhouse.

Motor Pump
This is an actuator that can be turned on and off by the Raspberry Pi to control the water supply to the plants.

Led Light
This is an actuator that can be turned on and off by the Raspberry Pi to control the light intensity inside the greenhouse.

Web-Camera
This is used to capture live feed of the greenhouse which is displayed on the webpage.

IV. SOFTWARE DESCRIPTION
The Smart Greenhouse Monitoring and Control System developed in this study utilizes a combination of Python programming language and various open-source libraries to collect, process, and display data from the sensors in the greenhouse. The system also includes a web-based user interface for remote monitoring and control.

The software components of the system include:
1. Operating System: The Raspberry Pi runs the Raspbian operating system, which is a variant of the Linux operating system.
2. Python Programming Language: Python is used to write the software that runs on the Raspberry Pi. Python is a widely used programming language with a large number of open-source libraries and frameworks that make it easy to develop complex applications.
3. Flask Web Framework: Flask is a lightweight web framework that is used to build the web-based user interface. Flask allows for the development of dynamic web applications that can be accessed from any device with a web browser.
4. Libraries: The system uses various open-source libraries to communicate with the sensors and actuators. These libraries include:
   - Adafruit DHT Library: This library is used to communicate with the DHT11 sensor and retrieve temperature and humidity data.
   - GPIO Library: This library is used to communicate with the GPIO pins on the Raspberry Pi and control the actuators.
   - Flask-SocketIO Library: This library is used to establish a WebSocket connection between the web-based user interface and the Raspberry Pi, enabling real-time data streaming.

The software is designed to be modular and scalable, allowing for the addition of new sensors and actuators. The user interface is accessible from any device with a web browser, allowing for remote monitoring and control of the greenhouse environment. The software is also designed to be user-friendly, with a simple and intuitive interface that can be used by anyone, regardless of their technical expertise.

V. IMPLEMENTATION
To implement the Smart Greenhouse Monitoring and Control System, the following steps were taken:
1. Hardware Setup: The hardware components were assembled according to the hardware description provided in Section III. The sensors were connected to the GPIO pins on the Raspberry Pi, and the actuators were connected to relays that were controlled by the Raspberry Pi.
2. Software Installation: The Raspbian operating system was installed on the Raspberry Pi, and the necessary software libraries were installed using the Python Package Manager (pip). The Flask web framework was installed, and a web-based user interface was developed using HTML, CSS.
3. Sensor Data Collection: Python scripts were developed to collect data from the sensors at regular intervals. The scripts used the Adafruit DHT library to collect temperature and humidity data from the DHT11 sensor, the GPIO library to collect data from the soil moisture sensor and LDR sensor.
4. Actuator Control: Python scripts were developed to control the actuators based on the data collected from the sensors. For example, if the temperature inside the greenhouse exceeded a certain threshold, the fan would be turned on to cool down the greenhouse.

5. Web-Based User Interface: A web-based user interface was developed using Flask and HTML, CSS, and JavaScript. The interface displayed real-time data from the sensors, including temperature, humidity, soil moisture, and light intensity. The interface also provided controls for the actuators, allowing the user to turn on or off the fan, motor pump, and LED light.

6. Remote Access: The Raspberry Pi was connected to the internet using a wireless network adapter, allowing the user to access the web-based user interface from any device with a web browser.

The implementation of the Smart Greenhouse Monitoring and Control System was successful, and the system was able to effectively monitor and control the greenhouse environment. The system was tested for several weeks and was found to be reliable and easy to use.

![Flow Chart](image)

**Figure 3**: Project Implementation flow chart

**VI. RESULTS**

The web-based user interface provided real-time data visualization, allowing the user to monitor the greenhouse environment from any device with a web browser. The user interface also provided controls for the actuators, allowing the user to turn on or off the fan, motor pump, and LED light.

The system was able to effectively control the actuators based on the data collected from the sensors. For example, when the temperature inside the greenhouse exceeded a certain threshold, the fan was turned on to cool down the greenhouse. Similarly, when the soil moisture level dropped below a certain threshold, the motor pump was turned on to water the plants.

Overall, the Smart Greenhouse Monitoring and Control System was found to be reliable and easy to use. The system was able to maintain optimal conditions for plant growth and reduce the need for manual intervention.
The web-based user interface provided remote access to the system, allowing the user to monitor and control the greenhouse environment from anywhere with an internet connection.

VII. CONCLUSION

In this project, we designed and implemented a Smart Greenhouse Monitoring and Control System using a Raspberry Pi, sensors, and actuators. The system was able to effectively monitor and control the greenhouse environment, maintaining optimal conditions for plant growth.

The system was designed to collect data from sensors and control the actuators based on the data collected. The web-based user interface provided real-time data visualization and controls for the actuators, allowing the user to monitor and control the greenhouse environment remotely. The implementation of the system was successful, and the system was found to be reliable and easy to use. The system was able to maintain optimal
conditions for plant growth and reduce the need for manual intervention. The web-based user interface provided remote access to the system, allowing the user to monitor and control the greenhouse environment from anywhere with an internet connection.

**VIII. FUTURE SCOPE**

The Smart Greenhouse Monitoring and Control System has great potential for future development and expansion. Some possible future directions for this project include:

1. Integration of machine learning algorithms: Machine learning algorithms can be integrated into the system to predict plant growth and optimize the greenhouse environment based on the collected data. This can improve the efficiency and productivity of greenhouse agriculture.
2. Incorporation of additional sensors: Additional sensors such as CO2 sensors and pH sensors can be incorporated into the system to monitor other factors that can affect plant growth. This can provide a more comprehensive picture of the greenhouse environment and allow for better control.
3. Integration with other smart systems: The Smart Greenhouse Monitoring and Control System can be integrated with other smart systems such as weather forecasting systems and irrigation systems to provide even better control of the greenhouse environment.
4. Remote control via mobile app: A mobile app can be developed to provide remote control of the greenhouse environment, allowing users to monitor and control the system from their mobile devices.

Overall, the Smart Greenhouse Monitoring and Control System has great potential for future development and can be further improved to increase the efficiency and productivity of greenhouse agriculture.

**ACKNOWLEDGEMENT**

We would like to express our sincere thanks to all those who contributed to the successful completion of this project. Firstly, we would like to express our gratitude to our project guide Prof. Ayubkhan Mulani Sir, who provided us with invaluable guidance, support, and motivation throughout the project.

We would also like to thank our colleagues and friends who provided us with valuable feedback and suggestions.

**IX. REFERENCES**


