Temperature and Humidity Monitoring System

Introduction

Project Overview

The **Temperature and Humidity Monitoring System** is a greenhouse monitoring project designed to provide real-time data on temperature, humidity, and day/night status. The project utilizes Arduino Yun, various sensors, and the Blynk platform for remote monitoring.

Objectives

The main objectives of the project are:

1. Monitor temperature and humidity inside the greenhouse.
2. Provide real-time data visualization through the Blynk app.
3. Implement email and notification alerts for temperature exceeding a threshold.
4. Track day and night conditions inside the greenhouse.

Components Used

- Arduino Yun
- LM35 Temperature Sensor
- Humidity Sensor
- Light Sensor
- Resistors
- Mini Breadboard
- Connecting Wires

System Implementation

Hardware Setup

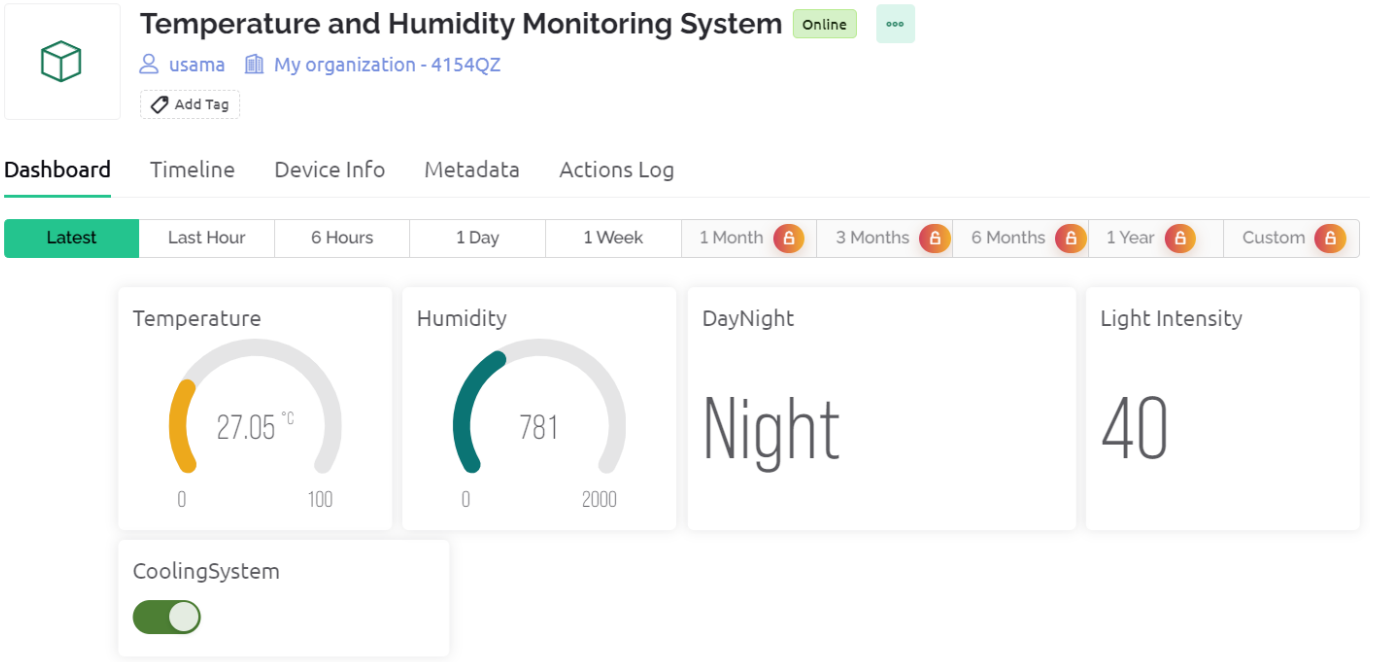
The system components are connected using the following configuration:

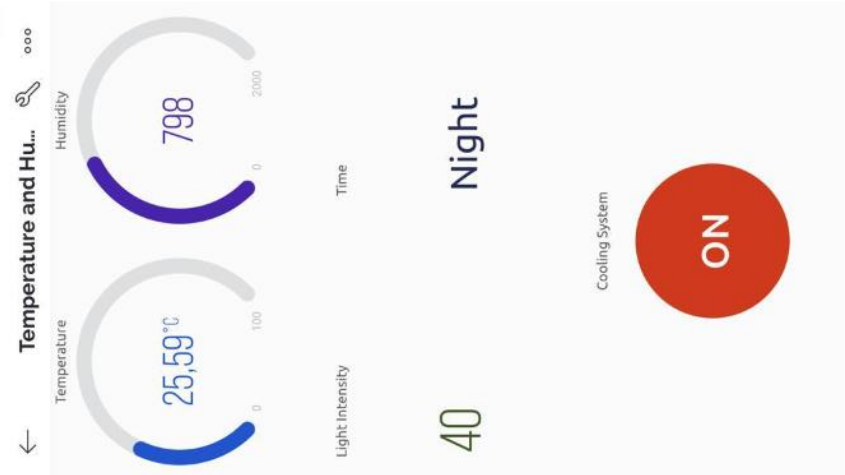
- LM35 sensor for temperature measurement.
- Humidity sensor for humidity measurement.

- Light sensor for monitoring light conditions.
- Arduino Yun for data processing and Blynk integration.

Blynk Integration

The Blynk platform is used for remote monitoring. The Arduino Yun is connected to the Blynk cloud using the provided authentication token. The Blynk app on the smartphone displays real-time temperature, humidity, and day/night status. Alerts are configured to notify users when the temperature exceeds a predefined threshold.





Coding

The Arduino code is structured to read sensor values, send data to the Blynk app, and trigger alerts. The code includes functions to determine day and night conditions based on light sensor readings. It also sends alerts to the Blynk app if the temperature exceeds a specified threshold. Here is the code used below:

```
#define BLYNK_PRINT Serial // Enables Serial Monitor
#define BLYNK_TEMPLATE_ID "TMPL4276vLRyT"
#define BLYNK_TEMPLATE_NAME "ArduinoTest"
#define BLYNK_AUTH_TOKEN "hzzlW8uQKl0DkBh6LE8nuMDq7wKeKdYn"

#include <Bridge.h>
#include <BlynkSimpleYun.h>

char auth[] = "hzzlW8uQKl0DkBh6LE8nuMDq7wKeKdYn";


int Lmax = 0;
int Lmin = 1023;
int L_value;
void setup()
{
  pinMode(A0, INPUT); // LM35 sensor
  pinMode(A1, INPUT); // Light sensor
  pinMode(A2, INPUT); // Humidity sensor
  pinMode(13,OUTPUT); //  Pin13 LED
  Bridge.begin();
  Blynk.begin(auth);
  Serial.begin(9600);


  int L; // Light Sensor calibratiion; only one time at the beginning
  Serial.println("Beginning Light Sensor calibration...");
  while (millis()<5000) {
    L = analogRead(A1);
    if (L>Lmax) Lmax = L;
    if (L<Lmin) Lmin = L;
    delay(10);
  }
  Serial.println("Calibration DONE!!");
  Serial.println(Lmin);
  Serial.println(Lmax);
  delay(2000);
```

```
}

void loop()
{
Blynk.run();

  delay(1000);
  float temperature = getTemperature();
  Blynk.virtualWrite(V1, temperature);  // Send temperature to Blynk app
  Serial.println(temperature);

  if(temperature > 30){
    Blynk.logEvent("temp_alert","Temp above 30 degree");
  }


  delay(1000);
  float lightvalue= analogRead(A1);
  Blynk.virtualWrite(V2, lightvalue); // Send LightValue to Blynk app
  Serial.println(lightvalue);

  delay(1000);
  float H_value = analogRead(A2);
  Blynk.virtualWrite(V4, H_value); // Send Humidity to Blynk app
  Serial.println(H_value);

  delay(1000);
  char* daynight = getdaynight();
  Blynk.virtualWrite(V3, daynight); // Send Day/Night status to Blynk app
  Serial.println(daynight);
  delay(500);

}


float getTemperature()
{
  int sensorValue = analogRead(A0);
  float voltage = (sensorValue / 1024.0) * 5.0;
  float temperatureC = ((voltage - 0.5) * 100.0)*-1;
  return temperatureC;
}


const char* getdaynight() {
```

```
  int a = analogRead(A1);

  if (a < 50) {
    return "Night";
  } else {
    return "Day";
  }
}


BLYNK_WRITE(V0){
int value = param.asInt();
if (value==1)
  digitalWrite(13,HIGH);
else
  digitalWrite(13,LOW);
}

BLYNK_CONNECTED(){
  Blynk.syncVirtual(V0);
}
```

Results and Observations

Real-Time Monitoring

The Blynk app provides a user-friendly interface for monitoring greenhouse conditions. Users can view temperature, humidity, and day/night status in real-time. The system also allows manual control of an output pin, turning a connected device on or off remotely.

Alert System

The implemented alert system sends notifications to the Blynk app and email when the temperature surpasses the defined threshold. This feature ensures timely responses to unfavorable conditions within the greenhouse.

Performance Metrics

The system's performance is evaluated based on its ability to provide accurate and timely data. The response time of the alert system and the reliability of sensor readings contribute to the overall success of the project.

Conclusion and Future Enhancements

Conclusion

The Temperature and Humidity Monitoring System effectively provides real-time data for greenhouse conditions. The integration with Blynk enhances remote monitoring and control capabilities.

Future Enhancements

To further improve the system, future enhancements may include:

- Integration of additional sensors for more comprehensive monitoring.
- Implementation of automated control systems for adjusting greenhouse conditions.
- Expansion of the alert system to include more sophisticated notification mechanisms.

The successful implementation of the project demonstrates its potential for optimizing greenhouse environments and promoting sustainable agricultural practices.

Group Members:
Usama ALI (1000053161)
Syed Muhammad Khizar ALAM (1000055853)