SOFE 4610-Fall 2022
Design and Analysis of IoT Software Systems
Project Progress Evaluation

Title: Greenhouse Humidity and Temperature Sensing System

Dr. Ramiro Liscano
10/20/2022

Name	Student ID
Minhal Syed	100618744
Hemshikha Sultoo	100670616
Shahroze Butt	100701891

Table of Contents

1. Project Background: 3

2. Major Components: **Error! Bookmark not defined.**

3. Flow Chart: **4**

4. Device Demonstration Diagram: **5**

5. Device Demonstration Video: **5**

6. GitHub link **5**

6. Code **6-8**

7. Contribution matrix **9**

8. Reference: **9**

1. Project Background:

Our current project involves monitoring the temperature inside a greenhouse to ensure that it is always maintained at an ideal level of 27 °C/80 °F. However, it should only reach a maximum temperature of 32 °C/90 °F and a minimum of 24 °C/75 °F [1]. Since temperature and humidity have an inverse relationship, they can cooperate to ensure that plant health is constantly monitored to support healthy plant growth.

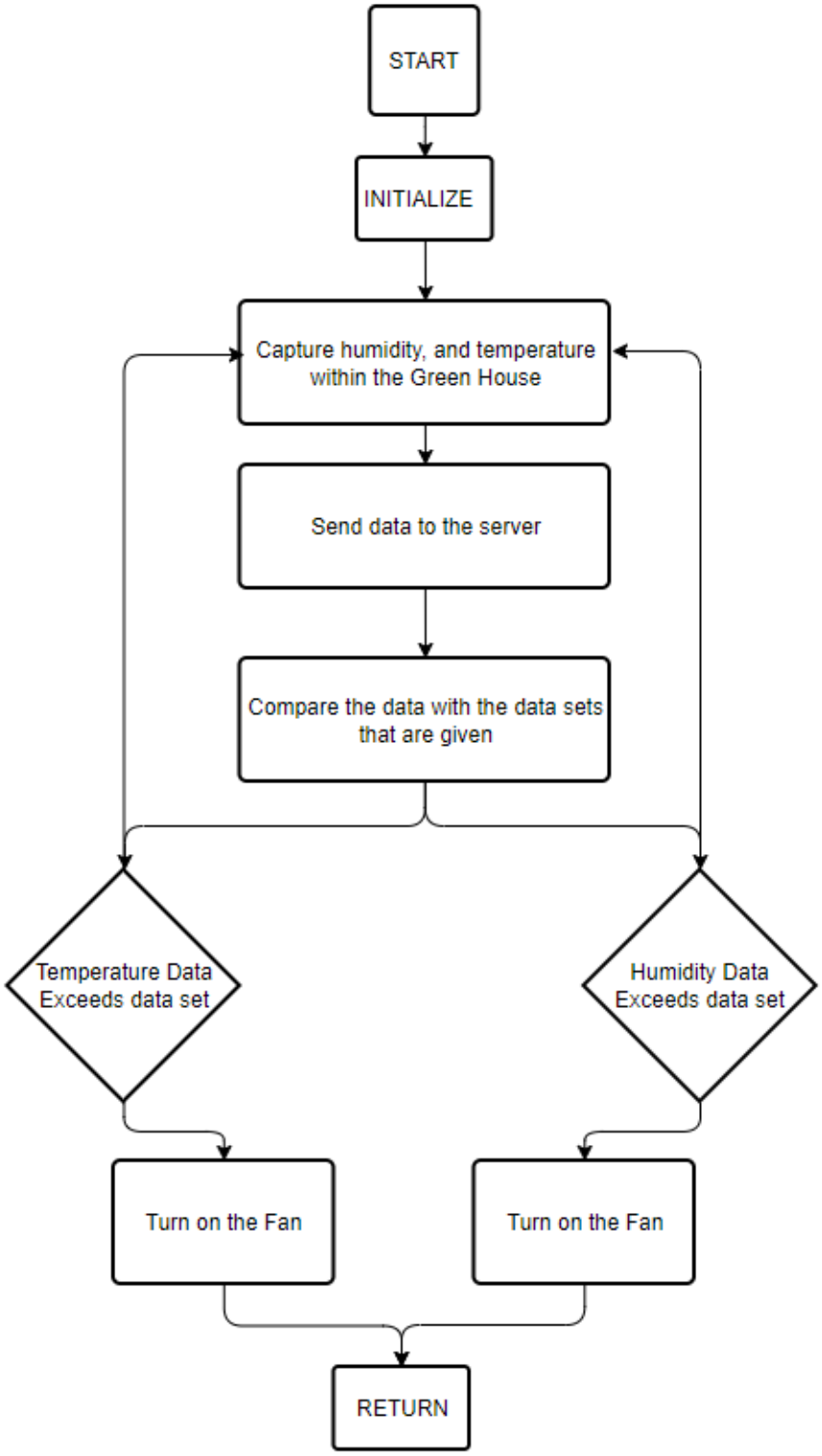
2. Major Components:

Due to configuration concerns between the pi 4 architecture and the most recent version of tensor flow, we decided to use our raspberry pi to download Buster and Python 3.7. The Raspberry Pi, the servo motor that also serves as a fan, and the DHT11 sensor, which measures temperature and humidity, are the main parts of our project. On the raspberry pi, the application is running as data is being gathered from the sensor, and it will then go through its decision-making process to decide what the fan should do.

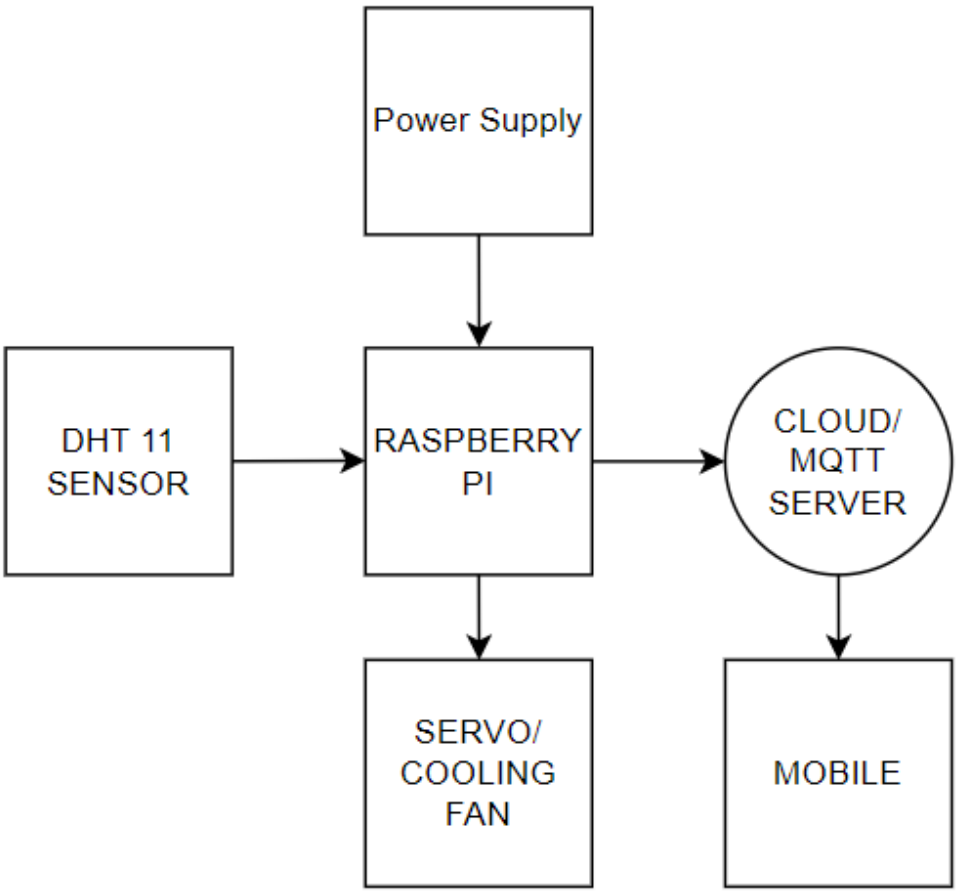
While loops were used in the programming to create the cases. If the temperature had risen and the fan had sensed that it had reached a certain temperature, it would run through a series of different scenarios. It would instruct the user to close the greenhouse door if the temperature had dropped because it would have appeared that the door was open and letting heat escape. It would simply monitor the temperature if there was no temperature flux, and nothing would happen. Once the environment's temperature has been regulated, the fan gradually turns off.

Use Cases	Details
UC-1: Increase in temperature	The infrastructure is continuously monitored by the system to check for sudden increases in humidity or decreases in temperature. When the temperature or humidity hasn't returned to normal, the system collects the data and turns the fan on
UC-2: Decrease in Temperature	When there is an abrupt rise in temperature and fall in humidity, when it keeps rising above a particular point, and when it hasn't returned to normal. The user would receive a warning to close the door to the green house.
UC-3: Stable environment	While data is being gathered and the environment hasn't changed all that much. The user could assess the app's stability by checking it.

3. Flow Chart:


START
INITIALIZE
Capture humidity, and temperature within the Green House
Send data to the server
Compare the data with the data sets that are given
Temperature Data Exceeds data set
Humidity Data Exceeds data set
Turn on the Fan
Turn on the Fan
RETURN

4. **Device Demonstration Diagram:**



5. **Device Demonstration Video:**

   https://drive.google.com/file/d/16koKhkYqnLJlhGhguqknjO1JHCBy_ARm/view?usp=share_link

6. **GitHub:**

   https://github.com/hsultoo/IoTProject.git

7. Code:

```
//Libraries
#include <DHT.h>
#include <Servo.h>


//Constants
#define DHTPIN 4     // what pin we're connected to
#define DHTTYPE DHT11   // DHT 11  (AM2302)
// Initialize DHT sensor for normal 16mhz Arduino
DHT dht(DHTPIN, DHTTYPE);
Servo s;




int chk;
float humidity;  //humidity
float temperature; //temperature


void setup()
{
 Serial.begin(9600);
 dht.begin();
 s.attach(9);
 s.write(90);


}


void motor(){
 for (int i=0; i<=180; i+=1){
```

```
    s.write(i);
    delay(15);
  }
}


void functions(){
  if (temperature > 26 && temperature <= 28){
    Serial.println("Greenhouse is running at optimal temperature and
humidity levels!");
    Serial.println("Ventilation system paused.....");
  }

  else if (temperature > 28){
    Serial.println("Overheat Alert!!!");
    Serial.println("Ventilation system triggered.....");
    Serial.println("");
    while(temperature > 28){
      motor();
      temperature = dht.readTemperature();
      Serial.println(temperature);
    }

    Serial.println("Optimum temperature reached. Ventilation system
paused!");
  }

  else if (temperature < 24){
    Serial.println("Temperature Drop Alert!!!");
    Serial.println("Please make sure that all the doors and ventilation
outlets are closed.");
  }
}
```

```
void loop()
{
   //Read data and store to humidity and temperature
   humidity = dht.readHumidity();
   temperature = dht.readTemperature();


   //Print readings
   Serial.print("");
   Serial.print("Humidity: ");
   Serial.print(humidity);
   Serial.print("%");
   Serial.println("");
   Serial.print("Temperature: ");
   Serial.print(temperature);
   Serial.print(" Celsius");
   Serial.println("");


   functions();


   delay(4000); //Delay by 4 seconds
}
```

8. Contribution matrix

Name	Percentage	Work
Hemshikha	33.3%	- **Coding/ Building device** - **Major Components** - **Device Demonstration Video** - **Format of Report** - **GitHub**
Minhal	33.3%	- **Coding/ Building device** - **Major Components** - **References** - **Slides** - **GitHub**
Shahroze	33.3%	- **Coding/ Building device** - **Flow Chart** - **Device Demonstration Diagram** - **Slides** - **GitHub**

9. Reference:

[1] N. Vercelletto, “How to get the right temperature and humidity level for healthy plants,” *HappySprout*, 31-Aug-2022. [Online]. Available: https://www.happysprout.com/outdoor-living/greenhouse-temprature-humidity/. [Accessed: 02-Nov-2022].