IOT Based Soil Monitoring and Automatic Irrigation System

Mohammad Shamiur Rahman Al Nahian (nahian76@gmail.com)
RUET: Rajshahi University of Engineering and Technology https://orcid.org/0000-0003-1036-4927
Arnab Piush Biswas
RUET: Rajshahi University of Engineering and Technology
Dr. J. C. Tsou
China University of Technology
Dr. Md. Hamidur Rahman
Department of Livestock

Research Article

Keywords: IOT, WSN, Automatic Control, Automation in agriculture, Arduino, NodeMCU, ESP8266

Posted Date: June 2nd, 2021

DOI: https://doi.org/10.21203/rs.3.rs-435834/v1

Version of Record: A version of this preprint was published at Acta Mechatronica on January 24th, 2022. See the published version at https://doi.org/10.22306/am.v6i3.71.

Abstract

To serve the humanity nowadays technology is playing a wonderful role and a man's basic and primary need is food indeed. It can be said that about more than 85% of people of Bangladesh are directly, indirectly depended on agriculture. Proper irrigation by water pump cannot be maintained due to frequent power outages, unavailability of grid lines in remote areas and scarcity/cost of fuel to run pumps. To make the sustainable irrigation system and field monitoring system for getting better crops growth as well as best production, this IOT based Automatic irrigation system is proposed. In this system IOT and WSN are used to control and monitor the irrigation system. IOT is used to obtain stored data monitoring and real time monitoring of various contents of soil. WSN is used to make a fully wireless system to make a user-friendly system to cultivate and irrigate water properly to the field. Different kinds of sensors are used. This report presents a fully automated drip irrigation system which is controlled and monitored by using "Thinkspeak Cloud Server". Temperature and the humidity content of the soil are frequently monitored. The system informs user about any abnormal conditions like less moisture content and temperature rise, even concentration of water by sending notifications through the wireless module.

Introduction

It is widely known that the resources of water are decreasing all over the world. On the other hand, rapid urbanization, population growth, industries and agriculture expansion increase the demand for fresh water. In the agriculture based countries including Bangladesh, for irrigation purpose water is used more than any other purpose, and the production rate can be decreased if any kind of hampering happened in water supply. The improvement of water usage efficiency without decreasing yield can be done by maintaining water management strategies & up-to-date technologies. It has become crying need for the agro-based countries to take more efficient technology in the field of agriculture to create better management of water resources. Digital Bangladesh concept that has led to tremendous growth in digital information storage, retrieval and communication. Now a day the concept of Internet of Things (IOT) has made human life more comfortable. Everyone is referring this system of inter-related computing devices, objects, things, animals, people, etc. Without human involvement the system is able sharing information over a network. The idea of IoT has been blooming since decades. For water savings function it has been proved that Wireless sensor network (WSN) system is very much helpful for irrigation management. WSN is the system which is a mesh of network of sensor nodes which are connected each other and the nodes directly collect data from the environment and provide real time data to the firm which is very much helpful for the farmers. Both as a data collection device and as a decision-making tool for real time monitoring this system can be used. The farmers are aware of water shortage or over watering may damage the yield. They need to understand when and how much amount of water is needed for specific crops. Most farmers have little knowledge of their farm and they are unaware of the methods of improving their productivity of agricultural practices. All these conflicts make it necessary to think of resolving support systems for agriculture. In order to overcome this problem, IoT based Wireless Sensor Network (WSN) for agriculture monitoring controls are applied. The internet/any kind of information sharing communication without cable connection between computers and other electronic devices can be done by Wireless Sensor Network (WSN) technology. A tremendous achievement has been found in agriculture environment with the help of Sensor Network System. It is said that in the 21st century, the most important technology is the WSN. WSN is a full package of a number of low-power, low-cost, multipurpose sensor nodes for a short and long distance wireless communication. Different network topologies and multichip communication is allowed by WSN. The effort and the complication can be cut down by WSN for monitoring environment. As a result of it the cost of water and labor can be reduced. Temperature, humidity, and soil moisture percentage and many more measurements can be remote by this technology. It seems that wireless outcomes are much better than the wired-based systems. Within this framework, IOT based wireless sensor network is a promising technology for irrigation management and soil monitoring by using soil conditions and actual weather on the basis of temperature and humidity of the area. A network of small devices which collect and process real time information from the fields in which they are deployed. The use of this technique makes the irrigation system & soil monitoring system independent of human intervention in terms of precise quantification, location and time of irrigation, and thus the establishment of an automatic irrigation system and soil monitoring system that is known as the smart irrigation and soil monitoring system for this reason. The purpose is to present several efficient irrigation systems and soil monitoring systems using IoT based wireless sensor networks, which can improve water use efficiency and also gives the correct condition of the soil by determining the timing of irrigation in an era of increasingly limited and costly water supplies. The second section can deal with the irrigation strategy that can be followed by an overview of smart irrigation by using wireless sensor networks. The artificial implementation of water in the field is known as irrigation. Irrigation comes in many forms. Many kind of efficient water supplying technology is replacing rapidly the old ones and applying it to the soil. Depending on how water is distributed throughout the field there are many different types of irrigation systems,

In this report a system has been developed to solve the problem of real time monitoring and stored data monitoring to investigate the soil condition at any time to take decision what types of crops should be grown and what should be done with the soil to get better and best production of the crops and also makes the whole system wirelessly automatic control over mobile phone which can reduce the cost of the labor as well the effort of a farmer.

- **Objectives**

1. To develop an IOT based automatic irrigation system having a low-cost equipment.
2. To monitor moisture contents at different conditions
3. To improve the system by using Mobile Phone App
4. To improve the system by using WSN (Wireless Sensor Network)

- **Scope and Limitations**

The scope of this project is

1. Monitoring of soil moisture content
2. Automatic Control system.
3. Real time monitoring of soil
4. Mobile based control system.
5. IOT Based platform

Limitations of this project is

1. The system can only be used via internet connection.
2. The system can be used with the help of batteries on the field where AC current is not available

System Overview

Figure 3.1 shows our proposed system. Our proposed system consists of 3 Nodes. Node 1 Consist of Arduino+Soil Sensor+NRF24L01 Module. Node 2 is consist of Arduino+NRF24L01+DHT11 Sensor+ESP8266 WiFi Module. Node 3 is consist of NodeMCU and Relay Module.

Figure 3.2 shows our working system. This diagram indicates how our 3 nodes are interconnected with each node.

So on the basis of our proposed system & working system, according to figure 3.1 & figure 3.2 we can see that there are 3 nodes in our system. In Node 1, capacitive soil moisture sensor. Arduino Uno & nRF24L01 module are mounted with each other. Arduino Uno collects the soil moisture data & sending data wirelessly in node 2. In node 2 Arduino Uno, nRF24L01, DHT11 Sensor, ESP8266 Wi-Fi module is mounted with each other. The node 2 receives data from node 1 and collects the temperature & humidity data from DHT11 & sending all the data in Thingspeak cloud server with the help of internet connection. Where all the data get stored for lifetime and it can be monitored at any time and we termed it as stored data monitoring. The Node 3 which is mounted with NodeMcu and relay module collects the data from the Thingspeak server and send wirelessly via Internet connection to a mobile app named Blynk which can be monitored in real time and the fact is called real time monitoring. The mounted relay can turn on or turn of the pump automatically as it is programmed. The pump can also be turned on or off by using the mobile app.

Thingspeak cloud server is an open server where any kind of data is stored by which all the system can be monitored. Figure 3.4 is showing how the data are plotted in the server and how it is storing all the data for monitoring.

Node 1: Node 1 is mounted with Capacitive Soil Moisture Sensor, nRF24L01 & Arduino Uno.

Capacitive Soil Moisture Sensor: The Sensor senses the data from soil.

Arduino Uno: The moisture data of soil is processed by Arduino Uno.

nRF24L01: It’s a wireless module. The processed data are sent wirelessly to node 2 by nRF24L01.

Node 2: Node 2 is mounted with Arduino Uno, nRF24L01, DHT11 & ESP8266 Wi-Fi module.

nRF24L01: This module receives the data which is sent by the node 1 and gives it to the arduino.

DHT11: This module collects the data of temperature and humidity and gives it to the arduino.

Arduiono Uno: It processes all the data.

ESP8266 Wi-Fi module: This module sends all the processed data to the thingspeak server via internet connection.

Node 3: It is mounted with Node MCU & Relay.

Node MCU: It receives the data from Thinkspeak Cloud Server and processes data and send them to the mobile app for real time monitoring and also gives command to the relay module.

Relay Module: This module can turn on or turn off the DC pump by the command of the Nodemcu.

- **Methods and tools used**

Table 3.1 is showing the specification of our System. Which components we used here and how does it operates.

Table 3.1: Specification of a System

Item	Specification
Arduino Uno	ATmega328P – 8 bit AVR family microcontroller, Operating Voltage 6-20V, DC Current on I/O Pin – 40ma.
NodeMcu	ESP-8266 32-bit, Operating Voltage- 3.3V, Input Voltage- 4-10V, Flash memory- 4 MB/64 KB
Development Platform	IA-32, x86-64, ARM.
Language Used	Arduino C++
Code development	Arduino Softwere

Table 3.1 describes the specification of our system. In our system, we use Arduino Uno as our mother controller. We use Arduino C++ as our operating language and Arduino software is used for code development.

Experimental Setup

IOT (internet of Things) part: Data are sent from Node 1 to Node 2. Node 2 receives the data & transfers the

data to Thingspeak cloud server through internet. These data are received by Node 3 via internet. This is real time data monitoring. WSN (Wireless Sensor Network) Part: Data is sent from node 1 to node 2 wirelessly. NRF4L01 is mounted with it

Automatic/Manual Control Of the Pump through Mobile App(Blynk): The pump can be turned on or turned off automatically. The pump also can operate manually by using mobile.

3.5 Sensing System

Capacitive soil moisture sensor senses soil moisture data of soil. The sensor is mounted in node 1. In Node 1, there is an Arduino Uno & NRF24L01 module which sends data wirelessly to Node 2.

The Capacitive Soil Moisture Sensor Has Three Pins

1. 5V VCC Pin.
2. GND Pin.
3. Analog Reading.

The 5V Vcc pin of the Sensor in connected with the Arduino from which the Sensor gets power to run the process. The Analog pin of the Sensor is connected with the A1 pin of the Arduino and the GND pin is connected with GND pin. When the Sensor is power up by 5V VCC then the Arduino gets the Sensor value through the pin A1 from the Analog Reading pin of the Sensor.

- **Circuit diagrams**

Node 1 is mounted with capacitive soil moisture sensor, Arduino Uno & NRF24L01 wifi module. There are 3 pins in capacitive soil moisture sensor which are directly connected with Arduino Uno. The pins of NRF24L01 are connected with Arduino Uno. So, the pins are connected to each other like this.

NRF24L01 ⟶	Arduino Uno
Pin CE ⟶	Pin 7
Pin CSN ⟶	Pin 8
Pin SCK ⟶	Pin 13
Pin MISO ⟶	Pin 12
Pin MOSI ⟶	Pin 11
Pin VCC ⟶	Pin 3.3V
Pin GND ⟶	Pin GND
Soil Moisture Sensor ⟶	Arduino Uno
Pin VCC ⟶	Pin 5V
Pin GND ⟶	Pin GND
Pin Analog Reading ⟶	Pin A1

Node 2 is mounted with DHT11 sensor, Arduino Uno, NRF24L01 wifi module and ESP8266Wifi Module. So, the pins are connected to each other like this.

NRF24L01 ⟶	Arduino UNO
Pin CE ⟶	Pin 7
Pin CSN ⟶	Pin 8
Pin SCK ⟶	Pin 13
Pin MISO ⟶	Pin 12
Pin MOSI ⟶	Pin 11
Pin VCC ⟶	Pin 3.3V
Pin GND ⟶	Pin GND
DHT11 Sensor ⟶	Arduino UNO
Pin VCC ⟶	Pin 5V
Pin GND	Pin GND
Pin DATA ⟶	Pin 4
ESP8266 WiFi Module ⟶	Arduino UNO
Pin RXD	Pin 3
Pin TXD ⟶	Pin 2
Pin VCC ⟶	Pin 3.3V
Pin GND ⟶	Pin GND
Pin CH_PD ⟶	Pin 3.3V

Node 3 is mounted with NodeMCU and Relay module. So, the pins are connected to each other like this.

Relay Module ⟶	NodeMCU
Pin EN ⟶	Pin D8
Pin VCC ⟶	Pin Vin
Pin GND ⟶	Pin GND

Figure 3.18 indicates the circuit diagram for Node 3 which is mounted with Nodemcu & Relay module. There are 38 pins in Nodemcu & 6 pins for Relay module.

- **Controlling System design**
  - **Algorithms and Flow chart**

Algorithm

(Soil Moisture Sensor+ Arduino+ nrf24L01 Module)

- Start
- Read the value of the Soil Moisture Sensor
- Processes the Value on Arduino
- Send the Value Through nrf24L01 Wirelessly

(nrf24L01 Module+ Arduino+ ESP8266 Wifi Module+ DHT11 sensor)

- Start
- Receive the value Through nrf24L01 Module from the soil moisture Sensor node
- Read the values of Temperature and Humidity from the DHT11 Sensor
- Processes the values on Arduino
- Send all Values to the Thingspeak Server Through ESP8266 Wifi Module

(Nodemcu+ Relay Module+ Mobile App)

- Start
- Read the values From the Thingspeak Server Through Nodemcu
- Processes the values on Nodemcu
- Send the values on Mobile App (Blynk)
- Taking Decision to turn the motor automatically on/off or Manually on/off.
- If The value of soil moisture is less than 60% the pump will automatically be turned on
- If the value of soil moisture is greater than 80% the pump will automatically be turned off
- The pump can be turned on/off through mobile app(Blynk) between the value 61% to 79%
- Stop

Working Principle of controlling System

Three major parts are involved here,

1. IOT (internet of Things) part: Data are sent from node 1 to Node 2. Node 2 receives the data & transfers the data to Thingspeak cloud server through internet. These data are received by node 3 via internet. This is real time data monitoring.

2. WSN (Wireless Sensor Network) Part: Data is sent from node 1 to node 2 wirelessly.nRF4L01 is mounted with it.
3. Automatic/Manual Control Of the Pump through Mobile App(Blynk): The pump can be turned on or turned off automatically. The pump also can operate manually by using mobile.

3.8 Cost Estimation

For Node 1

Node 1 is mounted with 1 unit Arduino Uno, 1 unit nrf24L01 & 1 unit capacitive soil moisture sensor. Total cost 850 tk.

Table 3.2: Cost Estimation for Node 1

Sl. No	Item Name	Specification	No. of Unit	Unit Cost (TK)	Total Cost (TK)
1	Arduino UNO	ATMEGA-328	1	350	850
2	Nrf24L01	single chip 2.4GHz	1	200	
3	Capacitive Soil Moisture Sensor	PH2.54-3P	1	300	

For Node 2

Node 2 is mounted with 1 unit Arduino Uno, 1 unit nrf24L01 , 1 unit DHT11 & 1 unit ESP8266 Wifi module. Total cost 900 tk.

Table 3.3: Cost Estimation for Node 2

Sl. No	Item Name	Specification	No. Of Unit	Unit Cost (TK)	Total Cost (TK)
1	Arduino UNO	ATMEGA-328	1	350	900
2	Nrf24L01	Single Chip 2.4GHz	1	200	
3	DHT11 Sensor	150	1	150	
4	ESP8266 Wifi Module	32-bit microcontroller with IEEE 802.11 b/g/n WiFi	1	200	

For Node 3

Node 3 is mounted with 1 unit Nodemcu & 1 unit Relay module. Total cost 500 tk.

Table 3.4: Cost Estimation for Node 3

Sl. No	Item Name	Specification	No. of Unit	Unit Cost(TK)	Total Cost(TK)
1	NodeMcu	ESP-8266 32-bit	1	400	500
2	Relay Module	5V switch	1	100	

Experimental Data

We have collected over 5000 data stored in the Thingspeak server which was collected in the indoor and outdoor situation at the format of excel

From the above data we come to an point on some different issues like when the values of humidity, temperature, soil moisture got changed. So on the basis of this we have made some data tables which has been given bellow.

Some Specific values of Temperature, Humidity & Soil Moisture Percentage from the 5000 data on May 23 have been given bellow:

Table 4.1

Date and Time	Temperature	Humidity	Soil Moisture Percentage
2020-05-23 04:45:00	26.88	84	88
2020-05-23 05:00:00	26.88	85	89
2020-05-23 05:15:00	28.35	85	89
2020-05-23 05:30:00	28.35	85	89
2020-05-23 05:45:00	28.84	85	89
2020-05-23 06:00:00	27.86	85	89
2020-05-23 06:15:00	27.37	86	90
2020-05-23 06:30:00	27.86	86	90
2020-05-23 06:45:00	26.88	86	90
2020-05-23 07:00:00	28.35	85	89
2020-05-23 07:15:00	27.37	83	89

Some Specific values of Temperature, Humidity & Soil Moisture Percentage from the 5000 data on June 03 have been given bellow:

Table 4.2

Date and Time	Temperature	Humidity	Soil Moisture Percentage
2020-06-03 04:45:00	28.84	83	77
2020-06-03 05:00:00	28.35	84	78
2020-06-03 05:15:00	28.84	84	78
2020-06-03 05:30:00	28.35	84	78
2020-06-03 05:45:00	28.84	84	78
2020-06-03 06:00:00	28.84	85	79
2020-06-03 06:15:00	28.84	85	79
2020-06-03 06:30:00	28.84	85	79
2020-06-03 06:45:00	29.33	85	79
2020-06-03 07:00:00	28.84	84	78
2020-06-03 07:15:00	28.84	84	78
2020-06-03 07:30:00	30.3	82	78

Some Specific values of Temperature, Humidity & Soil Moisture Percentage from the 5000 data on June 23 have been given bellow:

Table 4.3

Date and Time	Temperature	Humidity	Soil Moisture Percentage
2020-06-23 04:45:00	29.33	81	82
2020-06-23 05:00:00	29.33	82	83
2020-06-23 05:15:00	26.39	82	83
2020-06-23 05:30:00	28.84	82	83
2020-06-23 05:45:00	24.93	82	83
2020-06-23 06:00:00	29.33	83	84
2020-06-23 06:15:00	29.33	83	84
2020-06-23 06:30:00	29.33	83	84
2020-06-23 06:45:00	29.33	83	84
2020-06-23 07:00:00	28.84	82	83
2020-06-23 07:15:00	29.33	82	83
2020-06-23 07:30:00	28.84	80	82

Some Specific values of Temperature, Humidity & Soil Moisture Percentage from the 5000 data on July 04 have been given bellow:

Table 4.4

Date and Time	Temperature	Humidity	Soil Moisture Percentage
2020-07-04 04:45:00	29.33	76	83
2020-07-04 05:00:00	29.81	77	84
2020-07-04 05:15:00	29.81	77	84
2020-07-04 05:30:00	29.33	77	84
2020-07-04 05:45:00	29.81	77	84
2020-07-04 06:00:00	29.33	78	85
2020-07-04 06:15:00	29.81	78	85
2020-07-04 06:30:00	29.81	78	85
2020-07-04 06:45:00	29.81	78	85
2020-07-04 07:00:00	29.33	76	84
2020-07-04 07:15:00	29.33	76	84
2020-07-04 07:30:00	29.81	76	83

So by observing all the table values from different days but the time was from 5am to 7am, we found that at the time of this the values of humidity and soil moisture got increased every day.

Some Specific values of Temperature, Humidity & Soil Moisture Percentage from the 5000 data on May 28 have been given bellow:

Table 4.5

Date and Time	Temperature	Humidity	Soil Moisture Percentage
2020-05-28 07:30:00	28.35	84	82
2020-05-28 07:45:00	29.33	84	81
2020-05-28 08:00:00	29.33	87	82
2020-05-28 08:15:00	28.35	88	82
2020-05-28 08:30:00	28.84	88	83
2020-05-28 08:45:00	28.84	88	83
2020-05-28 09:00:00	28.35	89	84
2020-05-28 09:15:00	28.84	89	84
2020-05-28 09:30:00	27.86	88	83
2020-05-28 09:45:00	29.33	87	83
2020-05-28 10:00:00	27.86	87	83
2020-05-28 10:15:00	28.84	84	81
2020-05-28 10:30:00	29.33	84	81
2020-05-28 10:45:00	32.75	83	80
2020-05-28 11:00:00	29.81	83	80
2020-05-28 11:15:00	29.33	83	79
2020-05-28 11:30:00	28.84	83	79

Some Specific values of Temperature, Humidity & Soil Moisture Percentage from the 5000 data on June 12 have been given bellow:

Table 4.6

Date and Time	Temperature	Humidity	Soil Moisture Percentage
2020-06-12 14:30:00	28.84	80	85
2020-06-12 14:45:00	27.86	80	84
2020-06-12 15:00:00	28.84	85	86
2020-06-12 15:15:00	28.84	85	86
2020-06-12 15:30:00	28.84	85	85
2020-06-12 15:45:00	28.84	82	84
2020-06-12 16:00:00	28.84	79	84
2020-06-12 16:15:00	28.84	80	84
2020-06-12 16:30:00	28.84	80	84
2020-06-12 16:45:00	28.84	80	84

Some Specific values of Temperature, Humidity & Soil Moisture Percentage from the 5000 data on June 25 have been given bellow

Table 4.7

Date and Time	Temperature	Humidity	Soil Moisture Percentage
2020-06-25 10:30:00	29.33	82	77
2020-06-25 10:45:00	29.33	82	77
2020-06-25 11:00:00	27.86	87	80
2020-06-25 11:15:00	28.84	87	80
2020-06-25 11:30:00	28.35	86	79
2020-06-25 11:45:00	29.33	82	77
2020-06-25 12:00:00	28.84	81	76
2020-06-25 12:15:00	28.84	81	75
2020-06-25 12:30:00	28.35	81	75
2020-06-25 12:45:00	28.84	81	75

Some Specific values of Temperature, Humidity & Soil Moisture Percentage from the 5000 data on July 01 have been given bellow:

Table 4.8

Date and Time	Temperature	Humidity	Soil Moisture Percenatge
2020-07-01 17:45:00	29.33	75	92
2020-07-01 18:00:00	29.33	82	94
2020-07-01 18:15:00	27.86	85	95
2020-07-01 18:30:00	29.33	85	95
2020-07-01 18:45:00	29.33	85	95
2020-07-01 19:00:00	30.3	84	94
2020-07-01 19:15:00	28.84	83	93
2020-07-01 19:30:00	29.81	77	91
2020-07-01 19:45:00	29.81	75	91
2020-07-01 20:00:00	29.33	75	90
2020-07-01 20:15:00	29.81	76	90
2020-07-01 20:30:00	29.81	76	90
2020-07-01 20:45:00	28.84	76	90

Some Specific values of Temperature, Humidity & Soil Moisture Percentage from the 5000 data on July 10 have been given bellow:

Table 4.9

Date and Time	Temperature	Humidity	Soil Moisture Percentage
2020-07-10 09:30:00	29.81	79	78
2020-07-10 09:45:00	29.33	79	78
2020-07-10 10:00:00	29.81	85	80
2020-07-10 10:15:00	29.81	85	80
2020-07-10 10:30:00	29.33	85	80
2020-07-10 10:45:00	29.33	85	80
2020-07-10 11:00:00	29.81	84	78
2020-07-10 11:15:00	29.81	84	78
2020-07-10 11:30:00	29.33	83	77
2020-07-10 11:45:00	29.81	82	76
2020-07-10 12:00:00	29.81	79	75
2020-07-10 12:15:00	29.33	79	74
2020-07-10 12:30:00	29.81	79	74
2020-07-10 12:45:00	28.35	79	74

Some Specific values of Temperature, Humidity & Soil Moisture Percentage from the 5000 data on July 16 have been given bellow:

Table 4.10

Date and Time	Temperature	Humidity	Soil Moisture Percentage
2020-07-16 13:30:00	30.3	77	75
2020-07-16 13:45:00	30.79	77	75
2020-07-16 14:00:00	29.33	85	80
2020-07-16 14:15:00	30.3	85	80
2020-07-16 14:30:00	30.3	85	80
2020-07-16 14:45:00	28.84	80	77
2020-07-16 15:00:00	30.3	78	73
2020-07-16 15:15:00	30.3	78	73
2020-07-16 15:30:00	30.3	78	73
2020-07-16 15:45:00	29.81	78	73

So, from this section all the values of the data were from the rainy day, and the rain came for a certain time and the data is of that certain time. So, from the tables e found that at the time of raining the humidity and the soil moisture percentage increased.

Results

On the basis of the over 5000 data which was collected both on indoor and outdoor sides, we found that the values of the humidity and the soil moisture was decreasing

so fast in the outdoor side than the indoor side. It was taking so much time to decrease the humidity and soil moisture value in the indoor side.

Figure 4.3 represents the graphical view of the values of Table 4.1.

Figure 4.4 represents the graphical view of the values of Table 4.2

Figure 4.5 represents the graphical view of the values of Table 4.3

Figure 4.6 represents the graphical view of the values of Table 4.4

So from all the graphs which is based on the data of table 4.1, table 4.2, table 4.3 and table 4.4 we get that the humidity and the soil moisture percentage was increased in the morning of 5AM-7AM. That is mean the humidity and the soil moisture percentage always increase in the morning.

Figure 4.7 represents the graphical view of the values of Table 4.5

Figure 4.8 represents the graphical view of the values of Table 4.6

Figure 4.9 represents the graphical view of the values of Table 4.7

Figure 4.10 represents the graphical view of the values of Table 4.8

Figure 4.11 represents the graphical view of the values of Table 4.9

Figure 4.12 represents the graphical view of the values of Table 4.10

So from all the graphs which is based on the data of table5, table6, table7,table8 table9 and table10 we get that the humidity and the soil moisture percentage was increased in the rainy days. That is mean the humidity and the soil moisture percentage always increase in the rainy time.

After determining the boundaries of a given area, a soil moisture sampling plan should be developed. The initial sampling should be performed in a small grid; however, it is important to maintain a minimum of twenty sampling points per acre for a good soil moisture characterization, covering the entire area[16], Subsequent soil moisture sampling can be conducted using a grid-based system.

Discussion

During the time of our experiment we had faced some difficulties collecting data. The moisture content was suddenly fluctuated with the drastic change of outdoor & indoor temperature. For this scenario, there some error happened at the time of taking our moisture content. But after doing some calibration We could successfully complete our experiment.

Conclusion

The main achievement of our thesis is to build a system of real time monitoring and stored data monitoring of the soil condition during irrigation. As ours have an agriculturally based economy we have to be fully focused on maximum productivity. So, water wastage and soil monitoring during irrigation has to be done at a satisfactory rate so that maximum production can be ensured. The main objective of our thesis is to design a fully automated drip irrigation system and real time soil monitoring, stored data monitoring using IOT & WSN. The system provides an efficient monitoring of moisture, humidity and temperature content of soil. The data collected by the system can be used for further analysis purpose.

- **Improvements for the Future**

This project can be extended in future studies in order to improve the system in various aspects such as

This project can be used vastly in the rural areas of Bangladesh if Bangladesh agricultural ministry gives quality emphasize on this project. Then it would be made possible for the agricultural officers to monitor the farms without going to the lands. For this the farmers will be so much benefitted & at the same time production rate can be increased.

Online based warning system was not able to generate an alarm warning for insects. But it is very much needed thing to detect the insects for further decision making for the betterment of crop growth and another thing can be added in this research project and it is detection and distinguish the depredating insect sounds from other environmental sounds. For recognizing deficiencies, pests, diseases, and other detrimental agents in the vineyards an image processing method that can be added to the system.

Luminosity is a key factor of brightness analysis for estimating light radiation on plants can be applied for determining the sugar concentration, controlling the amount of sunlight received by the vines, and determining the optimum time for harvesting more accurately and timely. By utilizing luminosity sensors it can be measured that how much light radiation are falling to the crops or how it should be planted for better growth.

Declarations

1. *Funding*

Not Applicable.

2. *Competing interest:*

Not Applicable.

3. *Code Availbility:*

Arduino Program files

Program for Node 1 (Arduino+ Soil Moisture+ nrf24L01)

```
#include <SPI.h>

#include <nRF24L01.h>

#include <RF24.h>

RF24 radio(7,8);

int sensor = A1;

int sensorvalue;

int percentage;

const byte address[6] = "00001";

void setup() {
```

```
radio.begin();

radio.openWritingPipe(address);

radio.setPALevel(RF24_PA_MIN);

radio.stopListening();

pinMode(sensor, INPUT);

}

void loop() {

sensorvalue = analogRead(sensor);

percentage = map(sensorvalue,860,436,0,100);

radio.write(&percentage, sizeof(percentage));

}
```

Program for Node 2 (Arduino+ nrf24L01+ DHT11+ ESP8266 Wifi module)

```
#include <SoftwareSerial.h>

SoftwareSerial espSerial = SoftwareSerial(2,3); // arduino RX pin=2 arduino TX pin=3 connect the arduino RX pin to esp8266 module TX pin - connect the arduino TX pin to esp8266 module RX pin

#include <SimpleDHT.h>

#include <ESP8266_Lib.h>

#include <SPI.h>

#include <nRF24L01.h>

#include <RF24.h>

RF24 radio(7,8);// ce cns//

const byte address[6] = "00001";

int value;

int pinDHT11 = 4;

SimpleDHT11 dht11(pinDHT11);
```

```
String apiKey = "310ACMEQ7T9YN80T";

// replace with your channel's thingspeak WRITE API key

String ssid="Nahu55"; // Wifi network SSID

String password ="123456789"; // Wifi network password

#define ESP8266_BAUD 9600

ESP8266 wifi(&espSerial);

boolean DEBUG=true;

//============================================================================

 showResponce

void showResponse(int waitTime){

long t=millis();

char c;

while (t+waitTime>millis()){

if (espSerial.available()){

c=espSerial.read();

if (DEBUG) Serial.print(c);

}

}


}

//============================================================================

boolean thingSpeakWrite(float value1, float value2,float value3){

String cmd = "AT+CIPSTART=\"TCP\",\""; // TCP connection

cmd += "184.106.153.149"; // api.thingspeak.com
```

```
cmd += "\",80";
espSerial.println(cmd);
if (DEBUG) Serial.println(cmd);
if(espSerial.find("Error")){
if (DEBUG) Serial.println("AT+CIPSTART error");
return false;
}
String getStr = "GET /update?api_key="; // prepare GET string
getStr += apiKey;
getStr +="&field1=";
getStr += String(value1);
getStr +="&field2=";
getStr += String(value2);
getStr +="&field3=";
getStr += String(value3);
// ...
getStr += "\r\n\r\n\r\n";
// send data length
cmd = "AT+CIPSEND=";
cmd += String(getStr.length());
espSerial.println(cmd);
if (DEBUG) Serial.println(cmd);
delay(100);
if(espSerial.find(">")){
```

```
espSerial.print(getStr);

if (DEBUG) Serial.print(getStr);

}

else{

espSerial.println("AT+CIPCLOSE");

// alert user

if (DEBUG) Serial.println("AT+CIPCLOSE");

return false;

}

return true;

}

//=============================================================================
setup

void setup() {

DEBUG=true; // enable debug serial

Serial.begin(9600);

radio.begin();

radio.openReadingPipe(0, address);

radio.setPALevel(RF24_PA_MIN);

radio.startListening();

//pinMode(Temp_pin, INPUT);

pinMode(pinDHT11, INPUT);

//pinMode(Relay_Pin, OUTPUT);

espSerial.begin(9600); // enable software serial

// Your esp8266 module's speed is probably at 115200.
```

```
// For this reason the first time set the speed to 115200 or to your esp8266 configured speed

// and upload. Then change to 9600 and upload again

espSerial.println("AT+RST"); // Enable this line to reset the module;

showResponse(1000);

//espSerial.println("AT+UART_CUR=9600,8,1,0,0"); // Enable this line to set esp8266 serial speed to 9600 bps

//showResponse(1000);

espSerial.println("AT+CWMODE=1"); // set esp8266 as client

showResponse(1000);

espSerial.println("AT+CWJAP=\""+ssid+"\",\""+password+"\""); // set your home router SSID and password

showResponse(5000);

if (DEBUG) Serial.println("Setup completed");

}

// ====================================================================== loop

void loop() {

byte temperature = 0;

byte humidity = 0;

int err = SimpleDHTErrSuccess;

if ((err = dht11.read(&temperature, &humidity, NULL)) != SimpleDHTErrSuccess) {

Serial.print("Read DHT11 failed, err="); Serial.println(err);delay(1000);

return;

}

Serial.print((int)temperature);

Serial.println("Degree Centrigrade");

delay(1000);
```

```
Serial.print((int)humidity);

Serial.println("% Humidity");

delay(1000);

if (radio.available()){

radio.read(&value, sizeof(value));

}

Serial.print(value);

Serial.println("% Soil Moisture");

delay(1000);

thingSpeakWrite((int)temperature,(int)humidity,value);

// thingspeak needs 15 sec delay between updates,

delay(16000);
```

Program for Node 3 (Nodemcu+ Relay Module)

```
#define BLYNK_PRINT Serial // Comment this out to disable prints and save space

#include <BlynkSimpleEsp8266.h>

#include "ThingSpeak.h"

#include <ESP8266WiFi.h>

//Replace your wifi credentials here

char auth[] = "DcosB4cnDVf5Kiuli1_v9dj6tgrrU5Ta";

const char* ssid = "Nahu55";//Replace with your Wifi Name

const char* password = "123456789";// Replace with your wifi Password

//change your channel number here

unsigned long channel = 1090610;//Replace with your own ThingSpeak Account Channle ID

//1,2 and 3 are channel fields. You don't need to change if you are following this tutorial. However, you can modify it according to your application
```

```
unsigned int Temperature = 1;

unsigned int Humidity = 2;

unsigned int Soil = 3;

unsigned int Relay = 4;

WiFiClient client;

void setup() {

Serial.begin(9600);

delay(100);

pinMode(D8, OUTPUT);

//digitalWrite(D0, 0);

Blynk.begin(auth, ssid, password);

// We start by connecting to a WiFi network

Serial.println();

Serial.println();

Serial.print("Connecting to ");

Serial.println(ssid);

WiFi.begin(ssid, password);

while (WiFi.status() != WL_CONNECTED) {

delay(500);

Serial.print(".");

}

Serial.println("");

Serial.println("WiFi connected");

Serial.println("IP address: ");
```

```
Serial.println(WiFi.localIP());

Serial.print("Netmask: ");

Serial.println(WiFi.subnetMask());

Serial.print("Gateway: ");

Serial.println(WiFi.gatewayIP());

ThingSpeak.begin(client);

}

void loop() {

//get the last data of the fields

 int Temperature_1 = ThingSpeak.readFloatField(channel, Temperature);

 int Humidity_2 = ThingSpeak.readFloatField(channel, Humidity);

 int Soil_3 = ThingSpeak.readFloatField(channel, Soil);

 int Relay_4 = ThingSpeak.readFloatField(channel, Relay);

if(Soil_3 <= 60){

digitalWrite(D8, 1);

Serial.println("Motor Is On");

}

else if(Soil_3 >= 90){

digitalWrite(D8, 0);

Serial.println("Motor is Off");

}

Serial.println ("Temperature:");

Serial.println(Temperature_1);

Serial.println("Humidity:");
```

```
Serial.println(Humidity_2);

Serial.println ("Soil Moisture Percentage");

Serial.println(Soil_3);

Serial.println("Relay situation");

Serial.println(Relay_4);

delay(2000);

Blynk.virtualWrite(V1, Temperature_1);

Blynk.virtualWrite(V2, Humidity_2);

Blynk.virtualWrite(V3, Soil_3);

Blynk.run();
```

4. *Author's Contribution:*

In this Paper, The first Author, Mohammad Shamiur Rahman Al Nahian did the main things of the project. All the wiring, coding, testing, ,circuit desiging and finally data collection all the job have been done by him. The second Author, Arnab Piush Biswas works as research assistant with the first author. During data collection there was some problem and also having some coding problem, there the second author had some contribution. During circuit designing the second author had some electrical issue solved.

5. *Acknowledgement:*

All the praises and the supreme thanks belong to Allah, "Lord of All the Worlds, Most Beneficent and Ever-Merciful".

The authors wish to express their sincerest appreciation and heartiest gratitude to their supervisor, **Prof. Dr. Md. Rokunuzzaman**, for his valuable advice, guidance and assistance during the completion of this thesis. Without his advice and support, it would not be possible for us to prepare this report. We wish to thank some senior brothers and sisters of our department who helped us in preparing the report. We also want to mention some of our classmates who also have contributions to our report. In preparing of this report, we have got help from others as they wished to remain anonymous. We owe them with all gratitude.

Authors also like to thank the head of the department **Prof. Dr. Mhia Md Zaglul Shahadat**

and honorable professors of the Department for their valuable suggestion. We would be thankful if this report comes to any benefit of our teacher as well as any other students of our department.

References

1. Goap, Amarendra, Sharma, Deepak, Shukla, A. K., Rama Krishna, C., “An IoT based smart irrigation management system using Machine learning and open source technologies”, Computers and Electronics in Agriculture, vol.155, pp. 41-49, September, 2018
2. Nawandar, Neha K., Satpute, Vishal R., “IoT based low cost and intelligent module for smart irrigation system”, Computers and Electronics in Agriculture, vol.162, pp. 979-990, May, 2019.
3. Karimi, Navab Arabhosseini, Akbar Karimi, Mortaza Kianmehr, Mohammad Hossein., “Web-based monitoring system using Wireless Sensor Networks for traditional vineyards and grape drying buildings”, Computers and Electronics in Agriculture, vol.144, pp. 269-283, September, 2016
4. Moreno-Moreno, Carlos D. Brox-Jiménez, María Gersnoviez-Milla, Andrés A. Márquez-Moyano, Mariano Ortiz-López, Manuel A. Quiles-Latorre, Francisco J., “Wireless Sensor Network for Sustainable Agriculture”, Proceedings, vol.2, pp. 1302, 2018
5. Difallah, Wafa Benahmed, Khelifa Draoui, Belkacem Bounaama, Fateh., “Implementing wireless sensor networks for smart irrigation”, Taiwan Water Conservancy, vol.65, pp. 44-54, 2018
6. A Di Nisio, T. Di Noia, C. Guamieri Calo Carducci, M Spardavecchia., “Design of a Low Cost Multipurpose Wireless Sensor Network”
7. Zulkifli, C. Z. Noor, N. N., “Wireless sensor network and internet of things (IoT) solution in agriculture”, Pertanika Journal of Science and Technology, vol.25, pp. 91-100, 2018
8. Roy, Priyo Nath Armin, Maniza Kamruzzaman, S. M. Hoque, Md Emdadul., “A Supervisory Control of Home Appliances using Internet of Things”, 2nd International Conference on Electrical, Computer and Communication Engineering, ECCE, 2019
9. Ahmed, Hanady S. Ali, Abduladheem Abdulkareem., “Smart intensive care unit desgin based on wireless sensor network and internet of things”, Al-Sadiq International Conference on Multidisciplinary in IT and Communication Techniques Science and Applications, AIC-MITCSA 2016, pp. 41-49,2018
10. Kiani, Farzad Seyyedabbasi, Amir., “Wireless sensor network and Internet of Things in precision agriculture”, International Journal of Advanced Computer Science and Applications, vol.9, pp. 99-103, 2018
11. Sureephong, Pradorn Wiangnak, Patcharapong Wicha, Santichai., “The comparison of soil sensors for integrated creation of IOT-based Wetting front detector (WFD) with an efficient irrigation system to support precision farming”, 2nd Joint International Conference on Digital Arts, Media and Technology 2017: Digital Economy for Sustainable Growth, ICDAMT 2017, pp. 132-135,2017
12. Zhang, Peng Zhang, Qian Liu, Fusheng Li, Junqing Cao, Ning
13. Song, Changqing., “The Construction of the Integration of Water and Fertilizer Smart Water Saving Irrigation System Based on Big Data”, Proceedings - 2017 IEEE International Conference on Computational Science and Engineering and IEEE/IFIP International Conference on Embedded and Ubiquitous Computing, CSE and EUC 2017, vol.2, pp. 392-397, 2017

14. Wang, Jiamin Wang, MIlin Zheng, Kai Huang, Xin., “Model Checking nRF24L01-Based Internet of Things Systems”, Proceedings - 9th International Conference on Information Technology in Medicine and Education, ITME 2018, pp. 867-871, 2018
15. Wang, Yanping Chi, Zongtao., “System of wireless temperature and humidity monitoring based on Arduino Uno platform”, Proceedings - 2016 6th International Conference on Instrumentation and Measurement, Computer, Communication and Control, IMCCC 2016,pp. 770-773, 2016
16. Raut, R. Varma, H. Mulla, C. Pawar, Vijaya Rahul., “Soil Monitoring, Fertigation, and Irrigation System Using IoT for Agricultural Application”, Lecture Notes in Networks and Systems, vol.19, pp. 67-73, 2018
17. Zotarelli, Lincoln, Dukes, Michael D., Paranhos, Marcelo, “Minimum Number of Soil Moisture Sensors for Monitoring and Irrigation Purposes”, Edis, vol.2013, pp. 2-5, 2013.

Figures

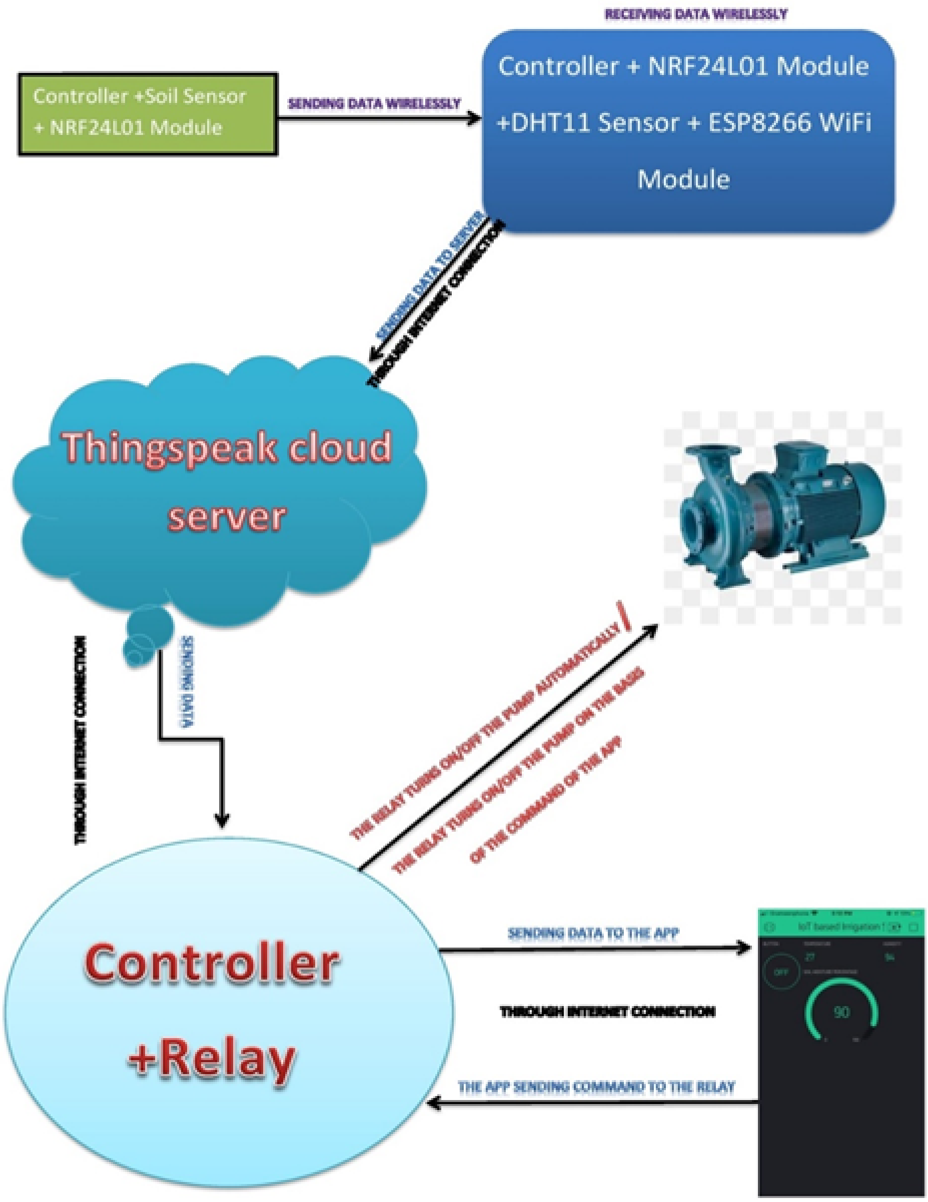


Figure 1

Proposed Model

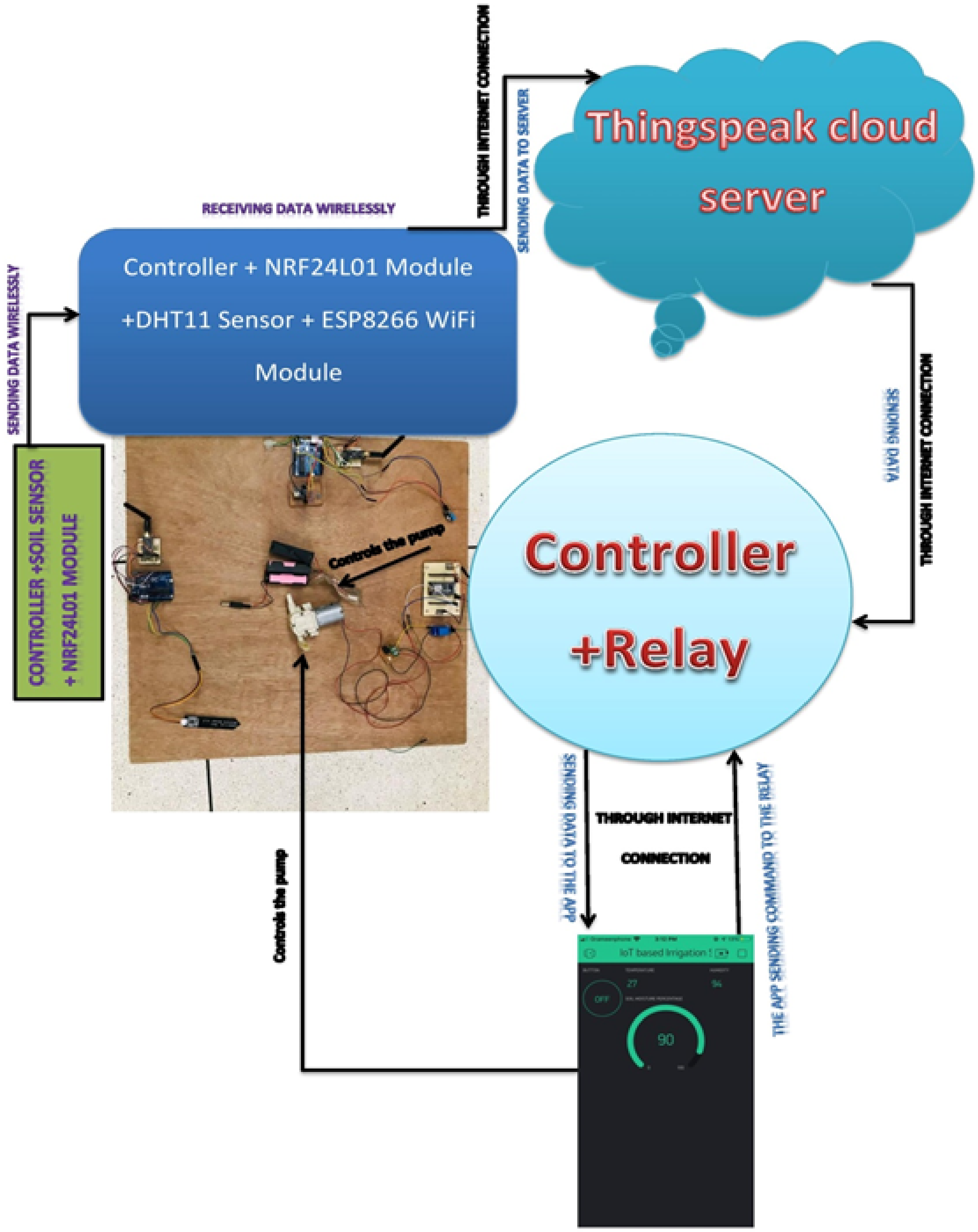


Figure 2

Working System

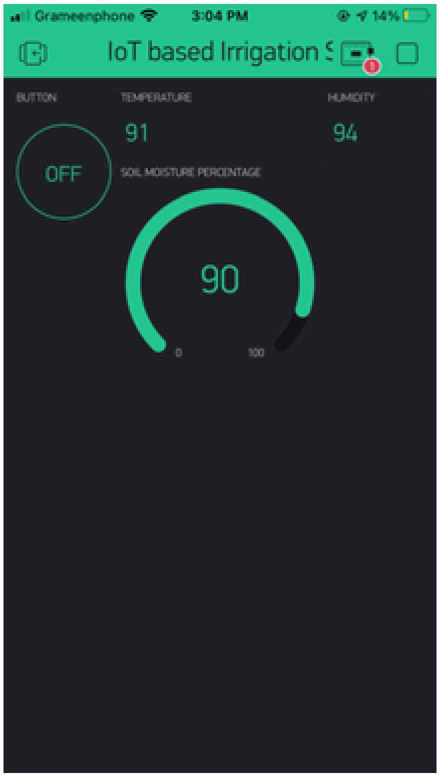


Figure 3

Mobile App (Blynk).

Figure 4

Thingspeak Cloud Server.

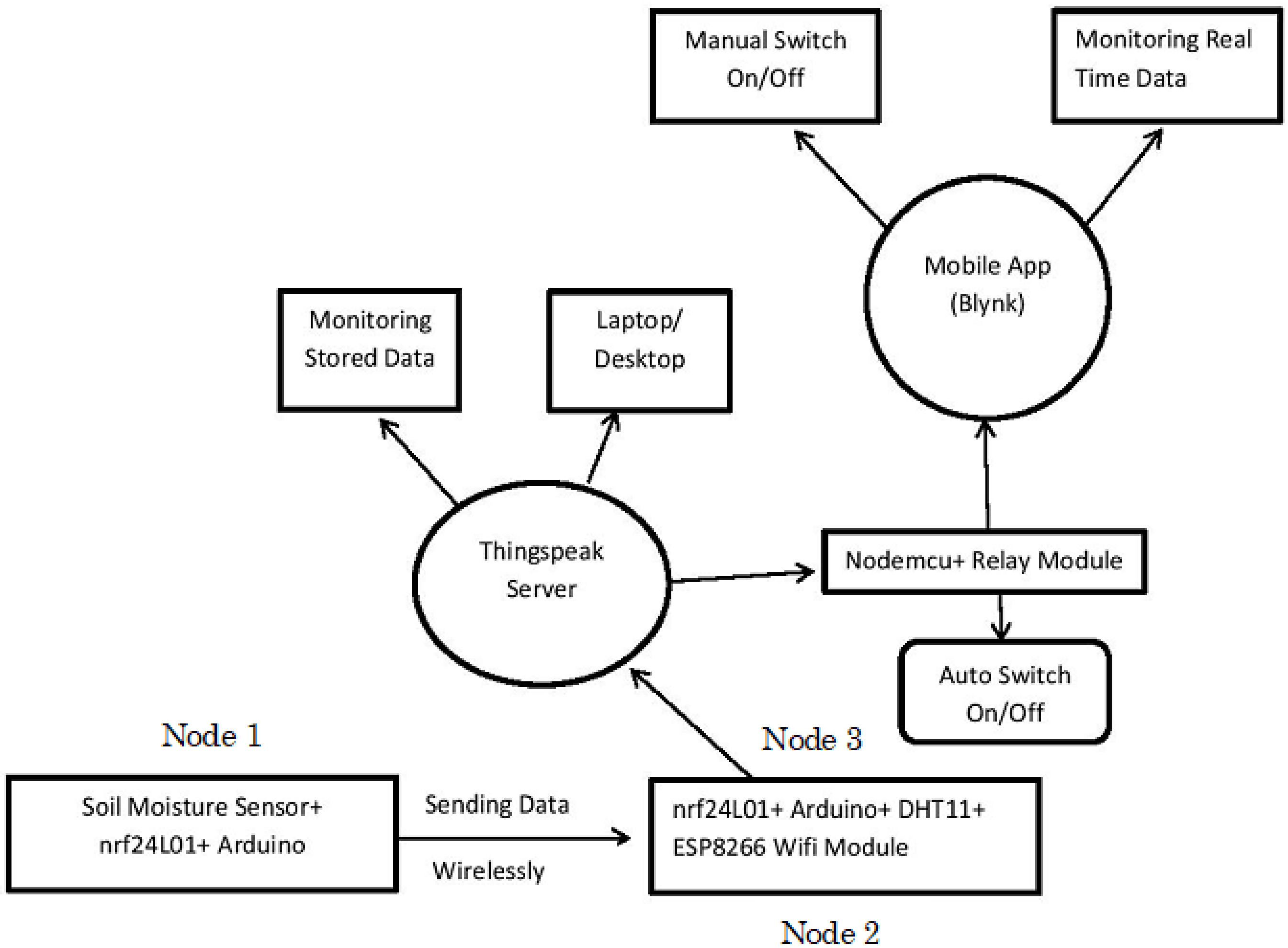


Figure 5

System Architecture

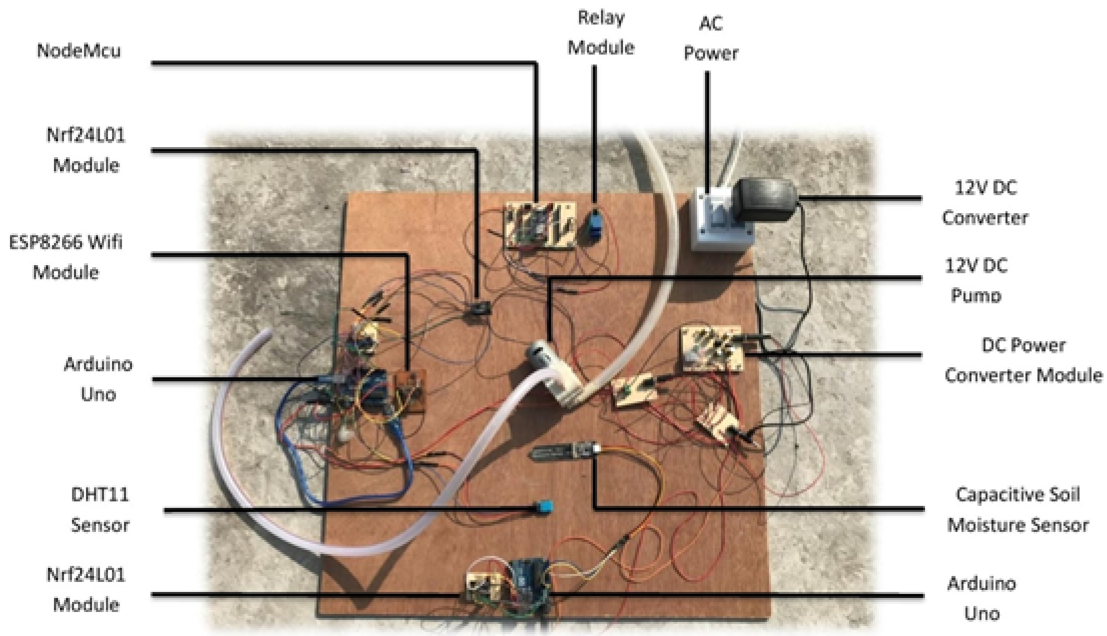


Figure 6

Experimental Setup

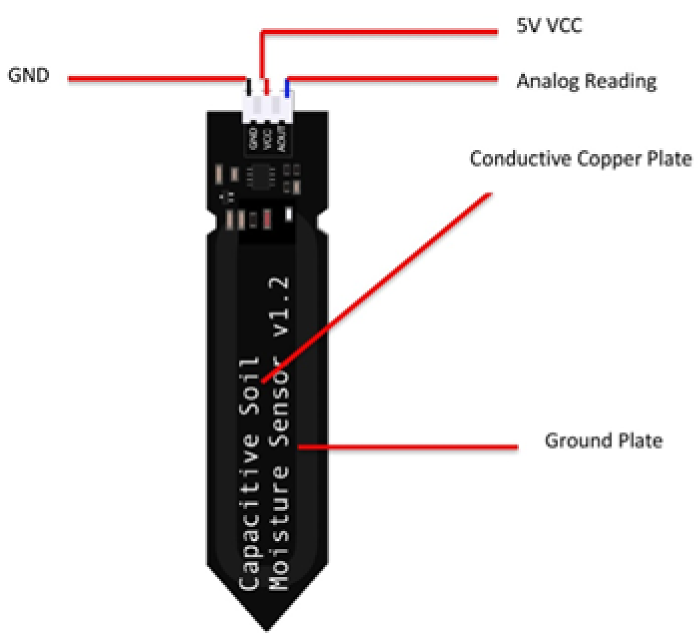


Figure 7

Sensing system Capacitive Soil Moisture Sensor.

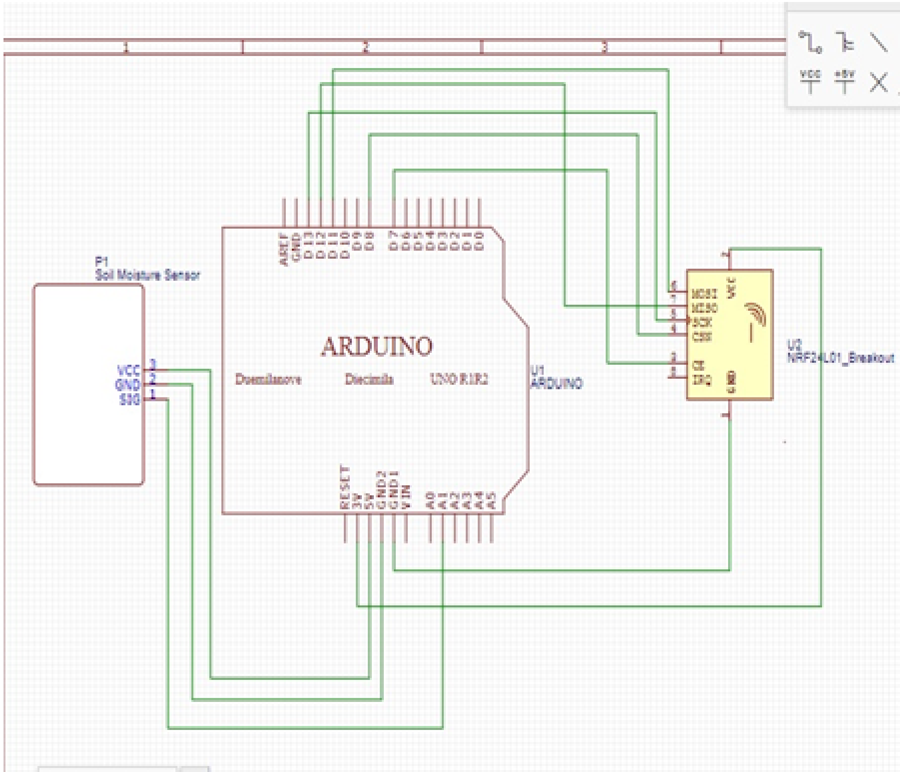


Figure 8

Circuit diagram of Node 1(Arduino+ nrf24L01+ Soil Moisture Sensor)

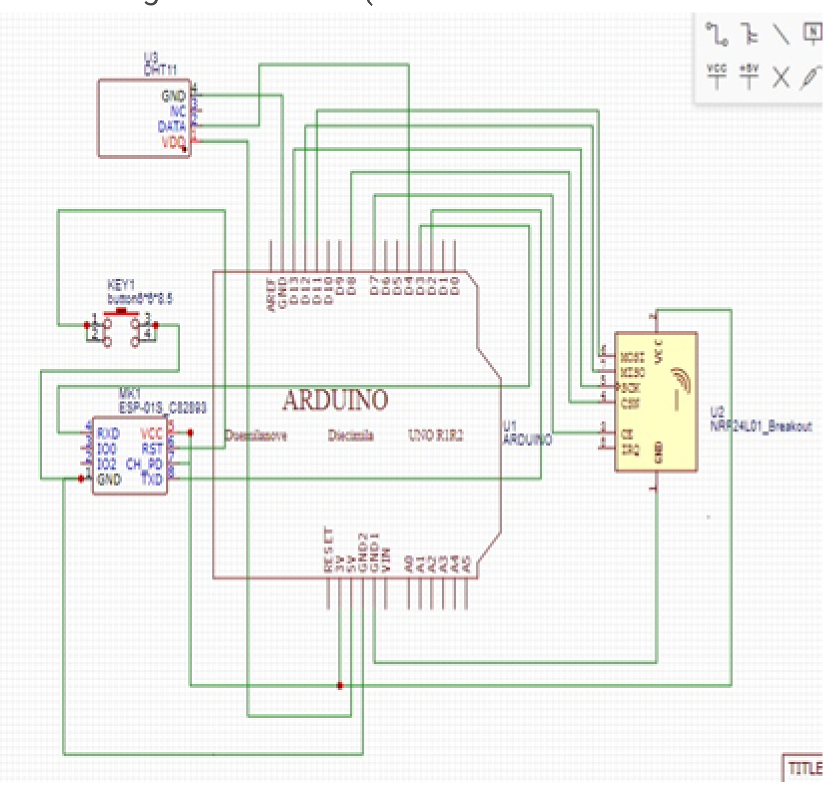


Figure 9

Circuit Diagram of Node 2(NRF24L01+ Arduino Uno+ DHT11+ ESP8266)

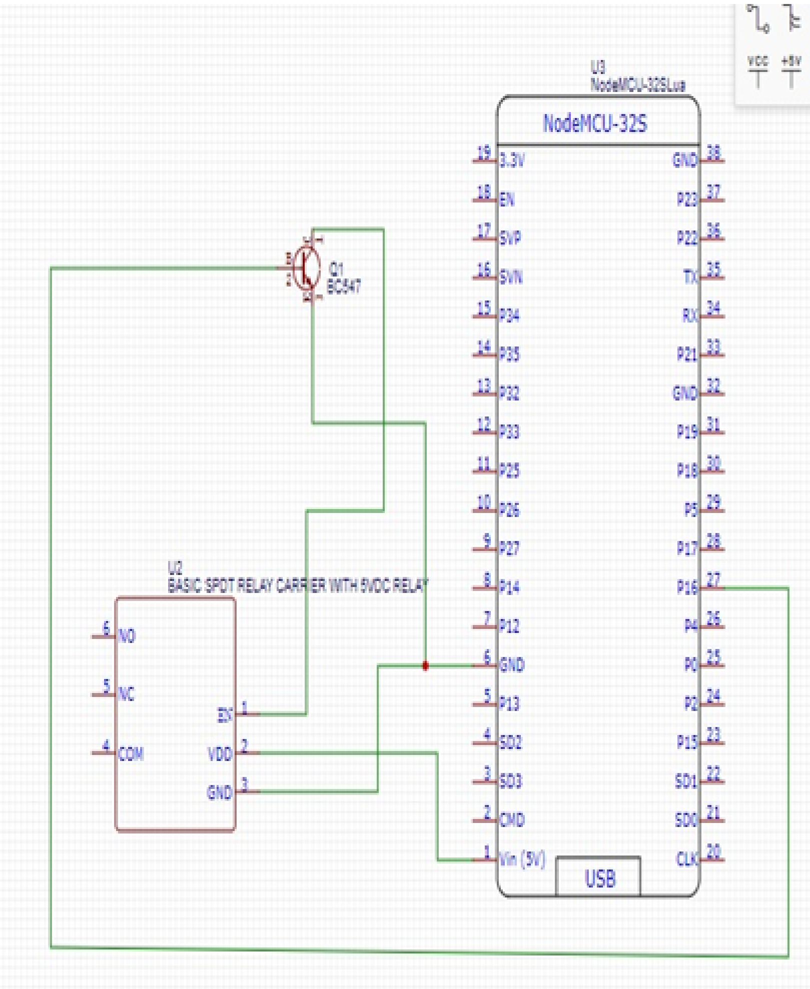


Figure 10

Circuit Diagram of Node 3(Nodemcu+ Relay Module)

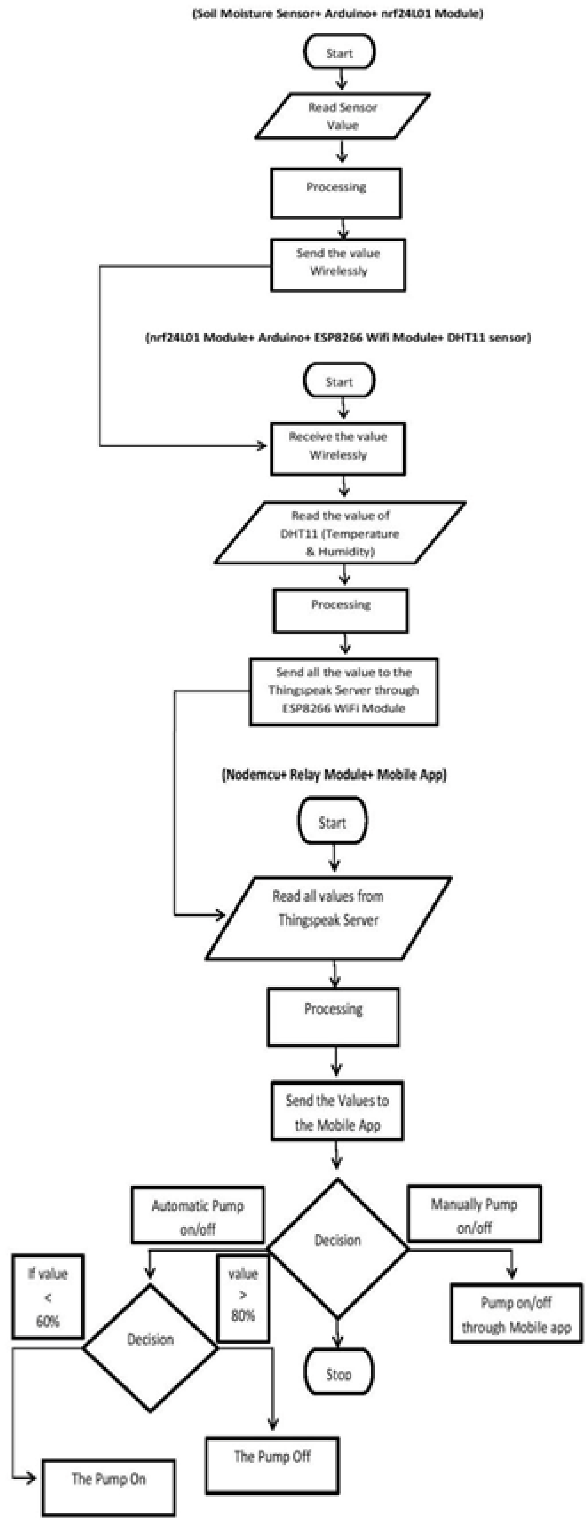
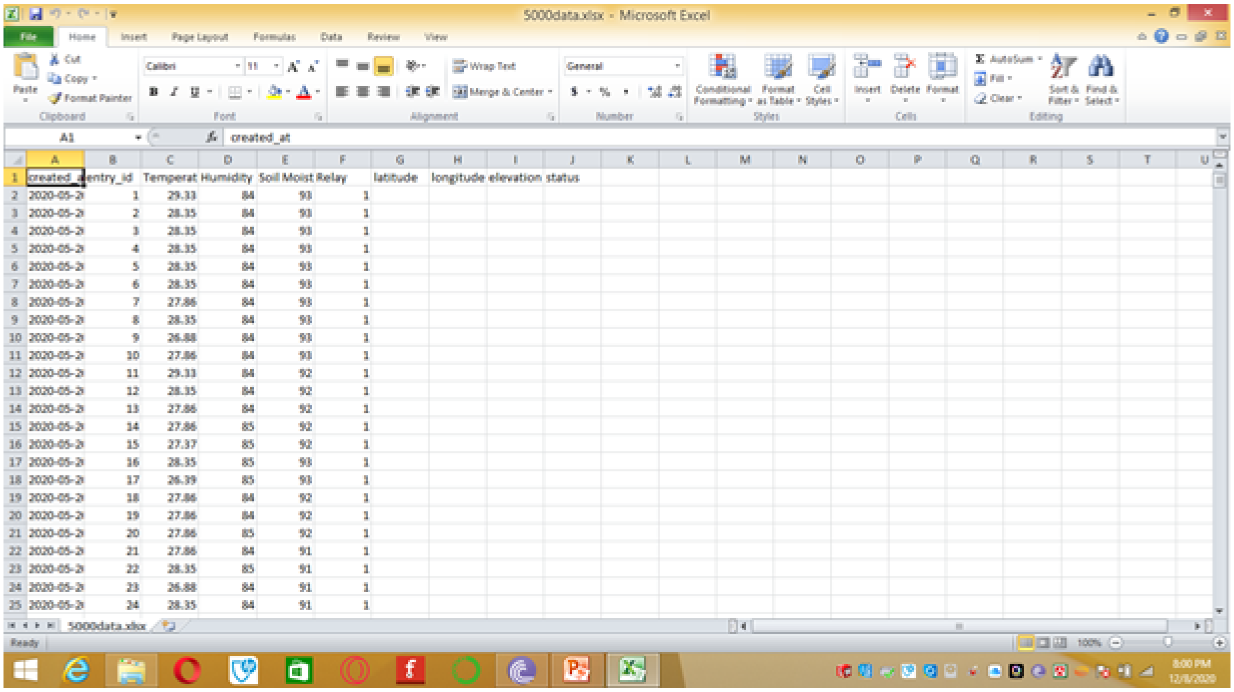


Figure 11

Flow Chart of the System



created_a	entry_id	Temperat	Humidity	Soil Moist	Relay	latitude	longitude	elevation	status
2020-05-2	1	29.33	84	93	1				
2020-05-2	2	28.35	84	93	1				
2020-05-2	3	28.35	84	93	1				
2020-05-2	4	28.35	84	93	1				
2020-05-2	5	28.35	84	93	1				
2020-05-2	6	28.35	84	93	1				
2020-05-2	7	27.86	84	93	1				
2020-05-2	8	28.35	84	93	1				
2020-05-2	9	26.88	84	93	1				
2020-05-2	10	27.86	84	93	1				
2020-05-2	11	29.33	84	92	1				
2020-05-2	12	28.35	84	92	1				
2020-05-2	13	27.86	84	92	1				
2020-05-2	14	27.86	85	92	1				
2020-05-2	15	27.37	85	92	1				
2020-05-2	16	28.35	85	93	1				
2020-05-2	17	26.39	85	93	1				
2020-05-2	18	27.86	84	92	1				
2020-05-2	19	27.86	84	92	1				
2020-05-2	20	27.86	85	92	1				
2020-05-2	21	27.86	84	91	1				
2020-05-2	22	28.35	85	91	1				
2020-05-2	23	26.88	84	91	1				
2020-05-2	24	28.35	84	91	1				

Figure 12

Collected Data.

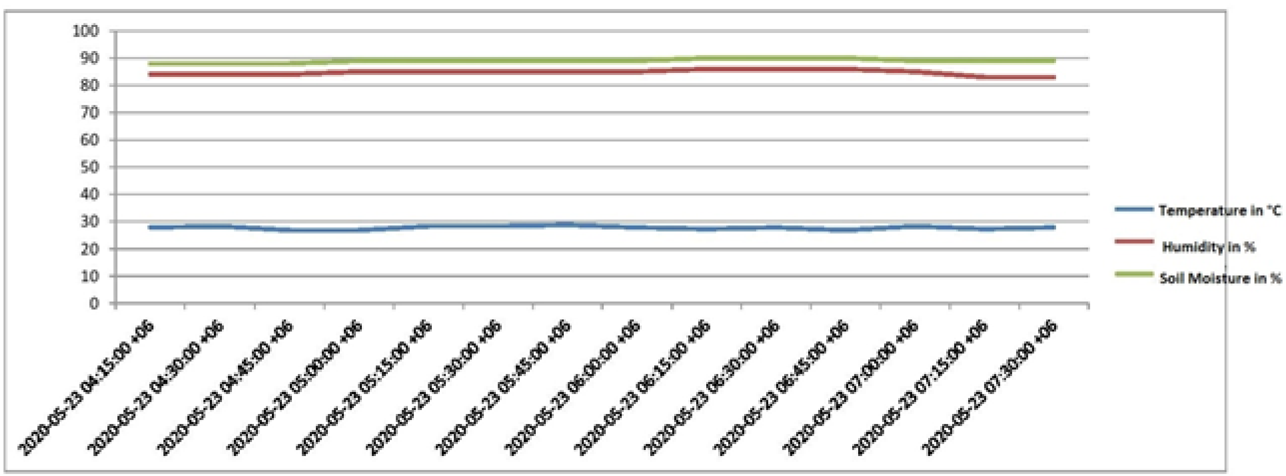


Figure 13

Graph of the indoor and outdoor position on the basis 0f over 5000 data

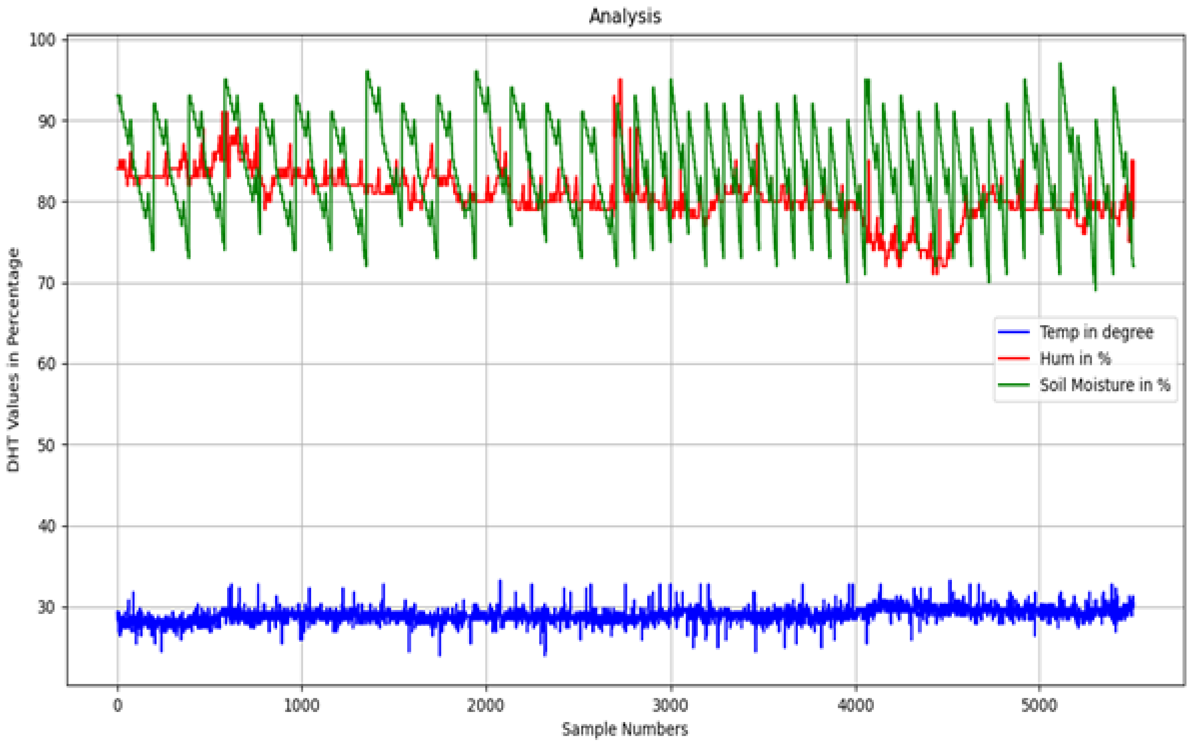


Figure 14

Graph of the data of Table 4.1

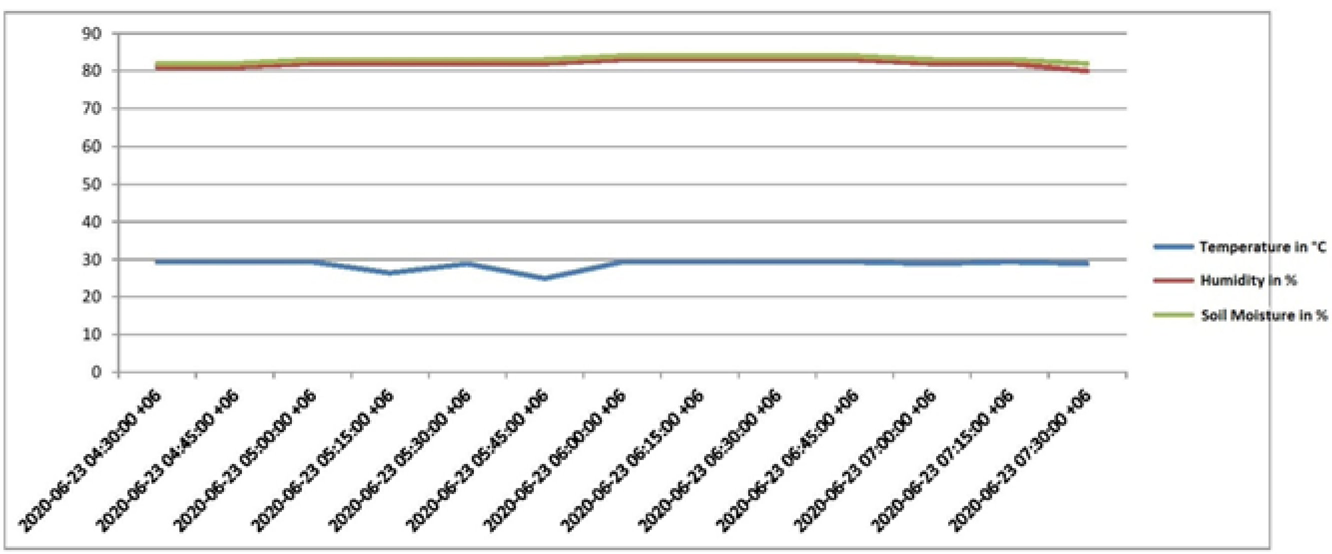


Figure 15

Graph of the data of Table 4.2

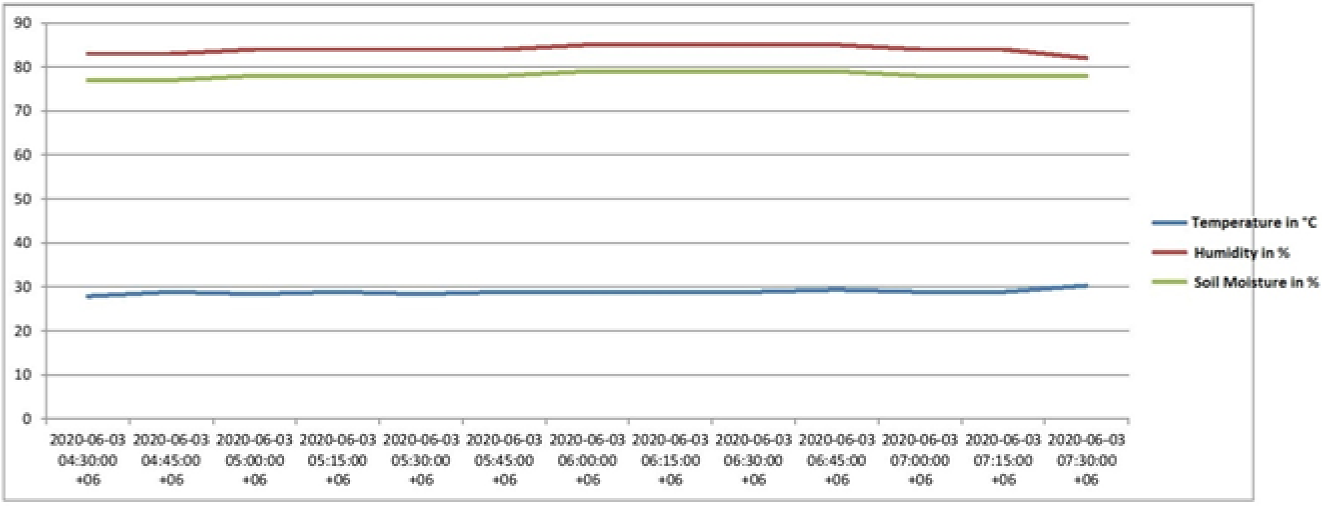


Figure 16

Graph of the data of Table 4.3

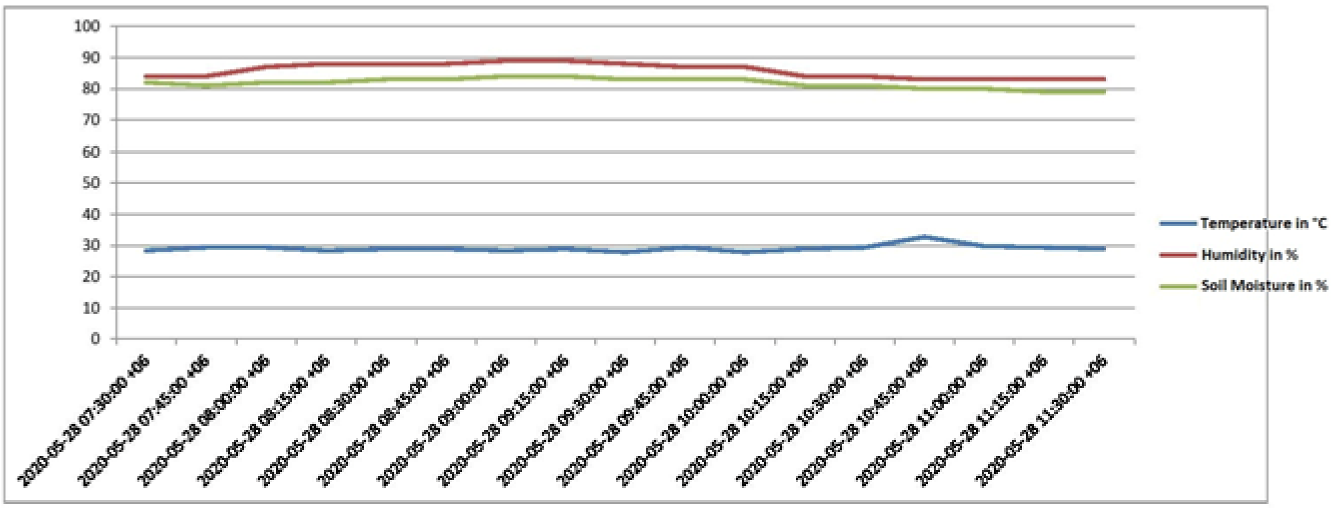


Figure 17

Graph of the data of Table 4.4

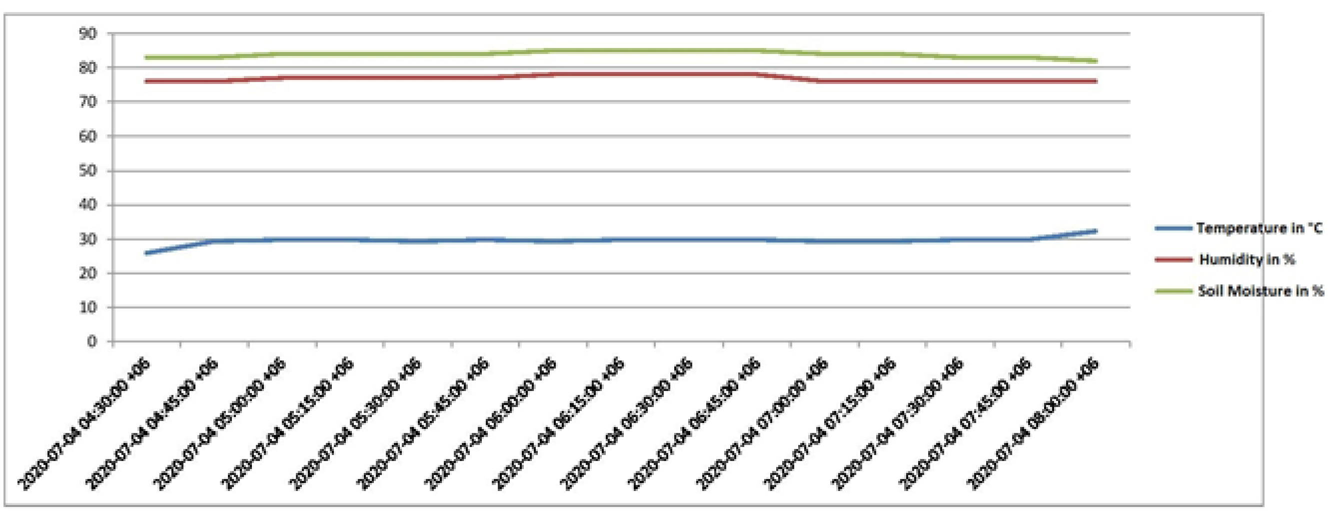


Figure 18

Graph of the data of Table 4.5

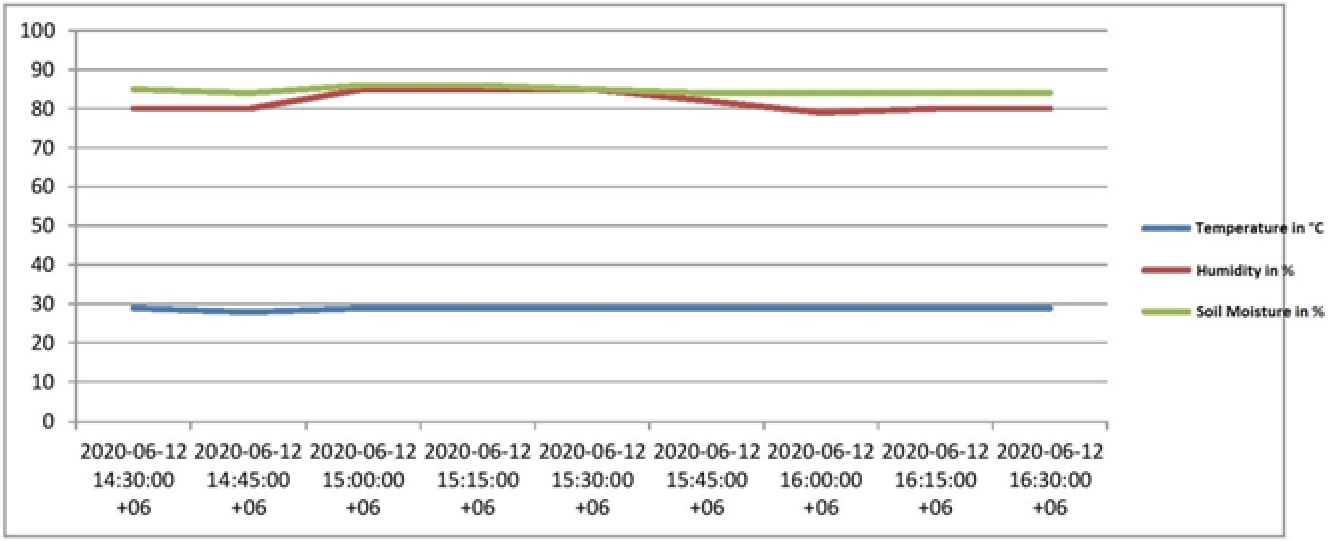


Figure 19

Graph of the data of table 4.6

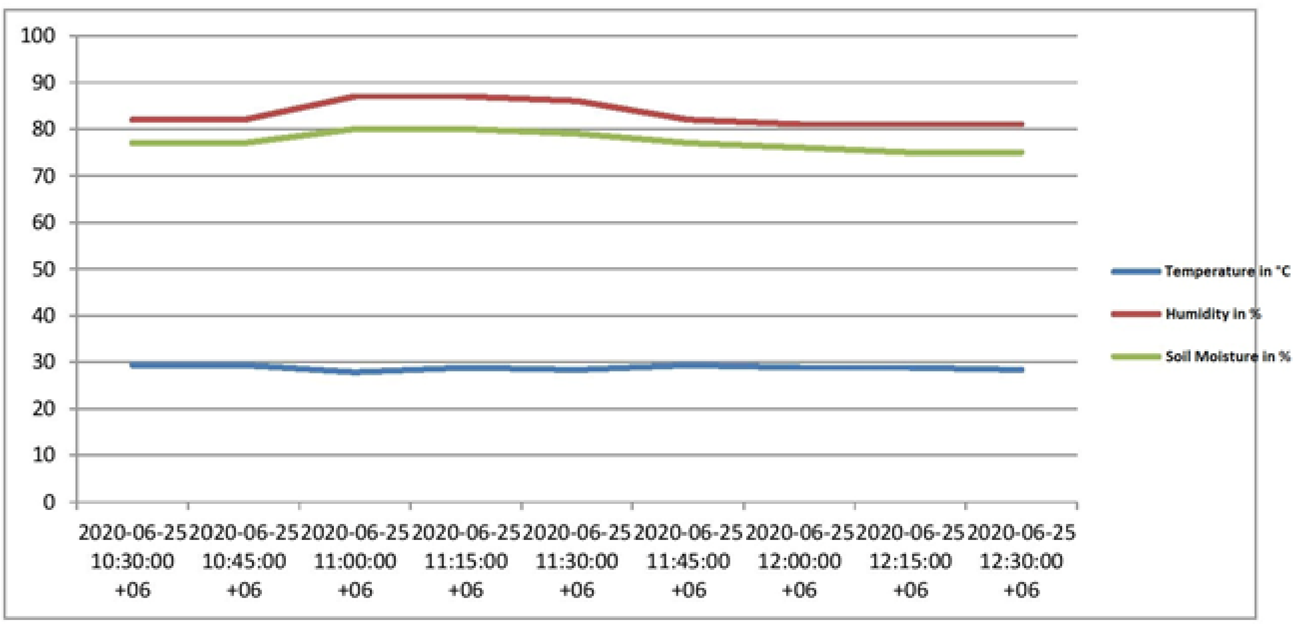


Figure 20

Graph of the data of Table 4.7

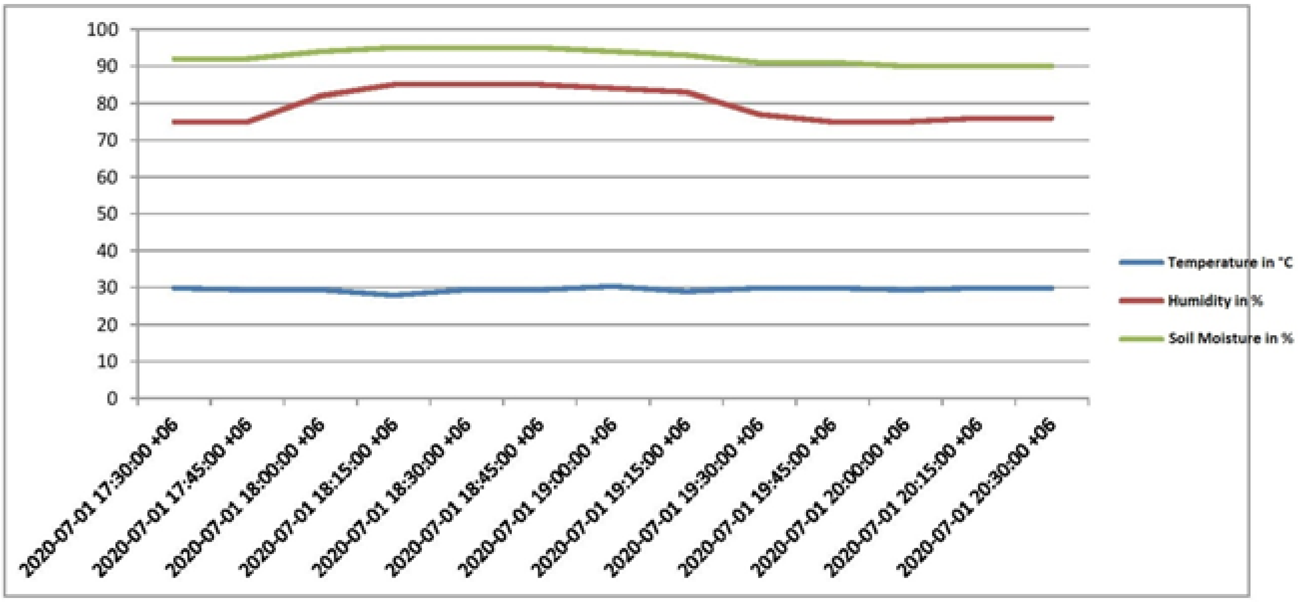


Figure 21

Graph of the data of Table 4.8

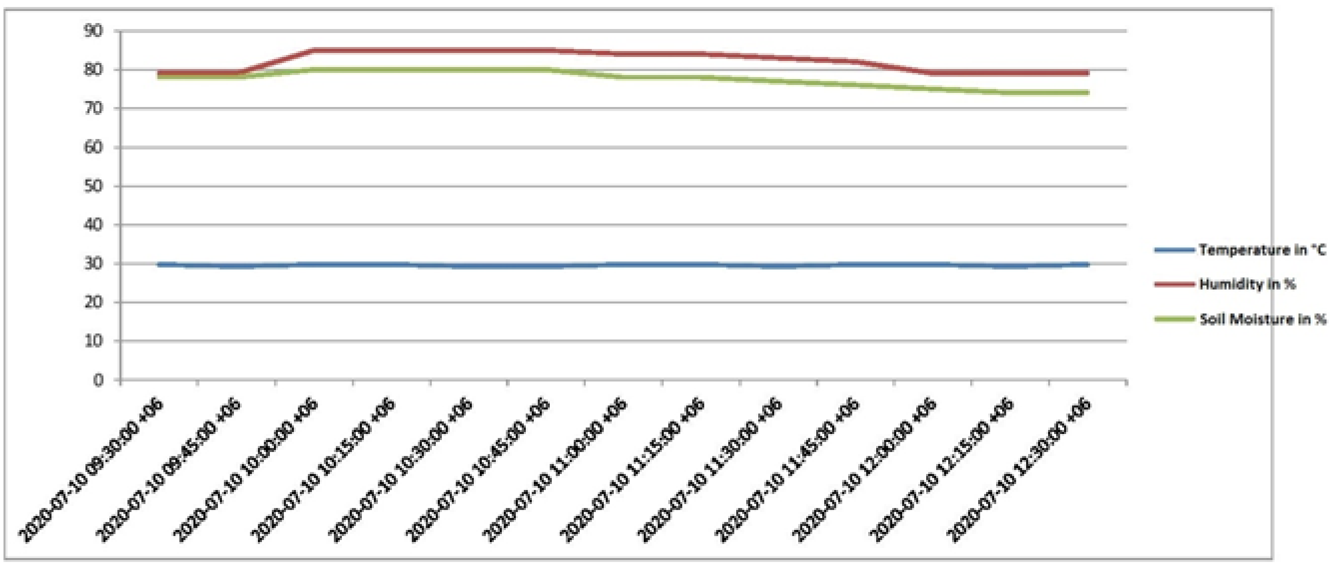


Figure 22

Graph of the data of Table 4.9

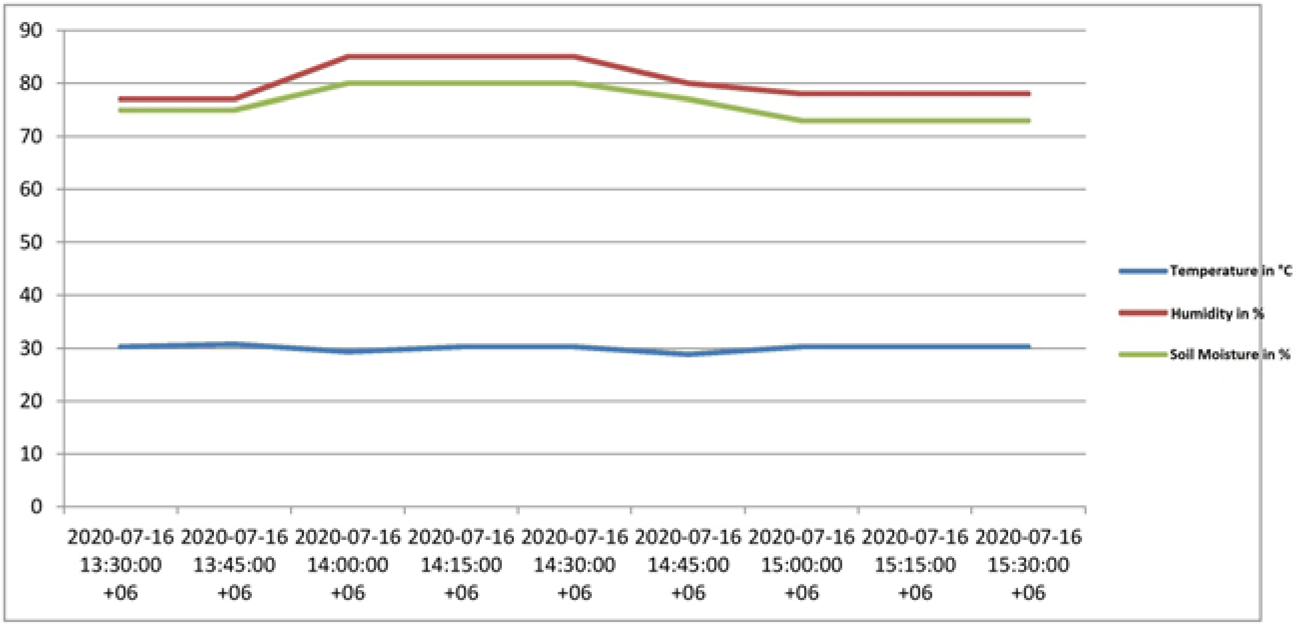


Figure 23

Graph of the data of Table 4.10