IoT based Automated Irrigation System & Mobile App for Tea Garden

B.Sc. Engineering Thesis
Course Code: CSE 800

Submitted to:

Department of Computer Science & Engineering

University of Chittagong

In partial fulfillment of the requirements for the degree of B.Sc. (Engineering)

in Computer Science and Engineering

August, 2023

Report_Code :

University of Chittagong

Department of Computer Science & Engineering

8th Semester B.Sc. Engineering Examination 2021

Course No.: CSE 800

Thesis Title: IoT based Automated Irrigation System & Mobile App for Tea Garden.

Report_Code:

University of Chittagong

Department of Computer Science & Engineering

8th Semester B.Sc. Engineering Examination 2021

Course No.: CSE 800

Thesis Title: IoT based Automated Irrigation System & Mobile App for Tea Garden

Student Name: Shariar Hasan
Student ID: 18701012
Session: 2017-2018
Hall: Shaheed Abdur Rab

Signature of Student:

....
Submission Date: 01 August 2023

ABSTRACT

A significant supply of water is essential for agriculture. Additionally, the demand for water resources grows along with the consumption of food. On the other hand, various crops need different amounts of water. Thus, manually watering crops often results in crop damage. In order to prevent excessive irrigation must be avoided as a way to prevent water waste and protect the crop from rotting. This problem may be resolved in a number of ways. The best course of action is to automate the system, nevertheless. The main purpose of this automated system is basically to protect the water and energy resources. It keeps an eye on the action and gauges the water level both manually and automatically. Agriculture hasn't been able to produce enough crops to keep up with population growth because of the dynamic weather fluctuations. The fields are often irrigated using channel systems, which give regular water injections but offer little insight into their water level. This kind of irrigation is detrimental to crop growth and output since many crops are sensitive to the amount of water in the soil.

Table of Contents

Table of Figures

Table of Tables

Chapter 01: Introduction

Agriculture is the backbone of many economies, and maintaining proper water management is vital for sustained agricultural development. In recent years, the integration of Internet of Things (IoT) technology with agriculture has changed conventional agricultural operations. IoT-based smart irrigation systems have emerged as viable solutions to manage water shortage concerns and boost water efficiency in agriculture. This project investigates the creation and deployment of a smart irrigation system using IoT intended exclusively for tea plantations.

Tea farming is an important economic activity in diverse places, and controlling water consumption is vital for tea plant health and output. The suggested smart irrigation system intends to automate irrigation operations, decreasing human involvement, and boosting water conservation.

The system employs NodeMCU 8266 microcontrollers to gather real-time data from soil moisture sensors and rain sensors. These sensors give vital insights into soil conditions and precipitation levels, allowing informed decision-making for irrigation. By utilizing sensor data and IoT technologies, the system can automatically calculate when and how much water is necessary for effective tea plant irrigation. A crucial aspect of the smart irrigation system is its Wi-Fi connection, enabling remote monitoring and management. Tea producers may examine the system's status and alter irrigation parameters using a user-friendly smartphone application. This remote management tool gives ease and flexibility, allowing farmers to effectively manage their tea plantations from a distance.

Ultimately, the IoT-based smart watering system for tea gardens is a step towards sustainable agricultural techniques. By using IoT technology, tea producers may optimize water consumption, boost crop yields, and contribute to the protection of vital water resources. The initiative coincides with the greater objective of encouraging eco-friendly alternatives to manage water constraint in the agriculture sector.

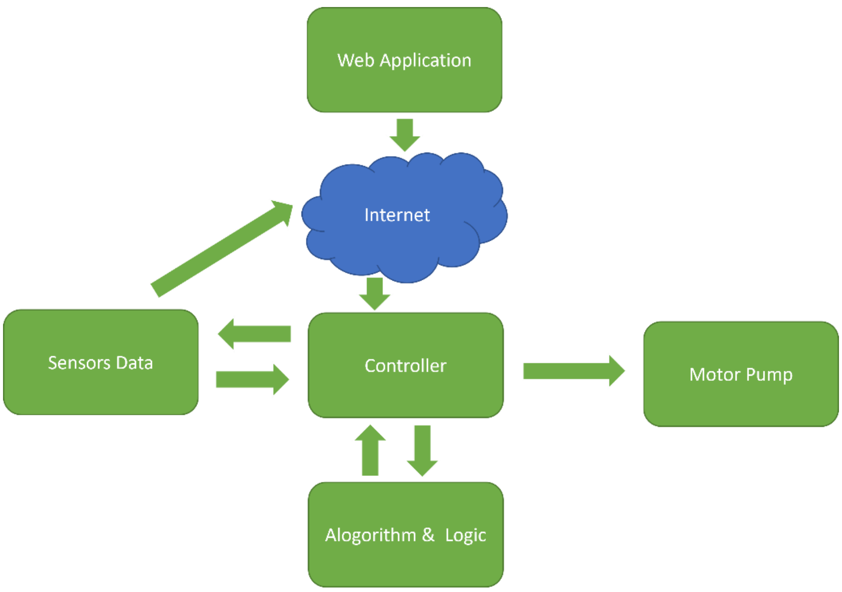


Figure 1: Data flow of the system

1.1 Background

The need for water in agriculture is great. The demand for water resources is increasing along with the expansion in food consumption. Different crops, however, have different water needs. As a result, manually watering crops often causes crop damage. It's important to avoid overwatering in order to minimize water loss and prevent the crop from rotting. To solve this problem, there are several strategies. The greatest option is to automate the system. This automated system's main objective is to safeguard the water and energy resources. Both manually and automatically, it gauges the water level while keeping an eye on the action. A lack of agricultural production to keep up with population growth is a result of the unpredictable weather patterns. Channel irrigation systems are often used to irrigate the fields; however, they only provide intermittent water injections and limited insight into the water level in the fields. This kind of irrigation has an adverse impact on crop growth and output since many crops are especially sensitive to the amount of water in the ground.

Due to water restrictions and unforeseen weather patterns, the agricultural sector has faced serious issues in recent years. Intelligent irrigation systems have come to be thought of as a potential solution to these problems. To monitor and manage irrigation operations more efficiently, these systems make use of IoT technology. The goal of the present study is to design an intelligent irrigation system that is specifically tailored for tea gardens.

1.2 Motivation

In many areas, tea growing is a significant economic activity. Keeping the soil at the ideal moisture level and reacting to rainfall are important aspects that have a direct influence on the health and productivity of tea plants. Traditional irrigation techniques often result in water waste and decreased output because they use too much water and distribute it unevenly. The suggested smart irrigation system uses IoT technology to improve water efficiency, reduce human involvement, and guarantee the tea plantations' sustainable development.

The necessity to solve urgent issues facing the agriculture industry is what drove the development of a smart irrigation system using IoT. Economy-supporting agriculture provides food and raw materials to companies across the globe. However, the industry is dealing with a number of intricate problems that endanger its viability and production.

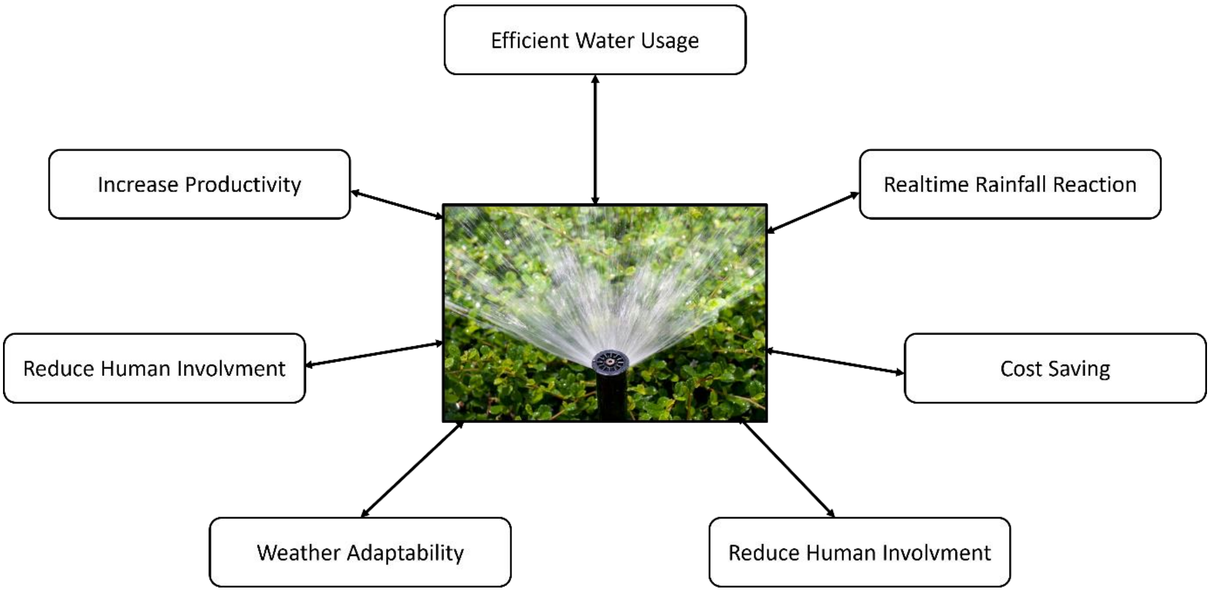


Figure 2: Motivation

1.3 Objectives

The creation and implementation of a smart irrigation system using IoT for tea gardens is the main goal of this project. The system will gather data on soil moisture and rainfall using NodeMCU 8266 microcontrollers, allowing automatic water supply control for tea plants. The project's precise goals are as follows:

- To monitor soil moisture levels in real-time, provide a trustworthy soil moisture sensor interface.
- Integrate rain sensors to track precipitation and determine if irrigation is necessary.
- To interpret sensor data and make wise judgments, create and program the NodeMCU 8266.
- Use a motor control system to automate irrigation depending on rainfall and soil moisture levels.
- To allow remote irrigation system monitoring and control, establish a Wi-Fi connection.
- Give farmers a simple way to utilize a mobile application to access and communicate with the system.
- Analyze the smart irrigation system's effectiveness and its effects on water saving and tea garden output.

Chapter 02: Literature Review

The R. N. Rao and B. Sridhar [1] designed and implemented the smart irrigation system, which involved selecting appropriate sensors to measure temperature and soil moisture in the crop fields. Since the temperature sensor takes ambient temperature readings, it serves as a crucial environmental parameter for the irrigation system. By continuously monitoring temperature changes in the field, the system can gauge the prevailing climatic conditions, enabling it to adjust the irrigation schedule accordingly. This adaptability ensures that the irrigation system can optimize water usage based on weather variations, avoiding potential overwatering or underwatering of the crops. Additionally, the soil moisture sensor plays a vital role in providing real-time feedback on the soil's water content, helping the system to precisely determine the irrigation needs of the crops. By combining temperature and soil moisture data through IoT technology, the smart irrigation system can make intelligent decisions, automating the irrigation process to provide the best possible water use for crop growth. This integration of sensor data with intelligent control mechanisms contributes to improved agricultural efficiency, water conservation, and enhanced crop yield.

In our system, the project utilizes soil moisture data to obtain real-time information about the soil's moisture content. This crucial data serves as a key input for decision-making in our automated irrigation system. When the soil moisture level falls below a certain threshold, the system triggers the motor to switch ON, initiating the irrigation process.

By continuously monitoring the soil moisture data, our system ensures that the crops receive the appropriate amount of water precisely when needed, preventing under- or over-irrigation. This smart control mechanism optimizes water usage, promotes efficient water resource management, and ultimately enhances crop growth and yield.

In the event that the soil moisture level rises to a sufficient level, indicating adequate water availability, the system automatically switches OFF the motor, conserving water resources and avoiding unnecessary irrigation.

S. Rawal [2] focuses on developing a smart irrigation system using the Internet of Things (IoT) technology to enhance water efficiency in agriculture. The primary objective of the study is to optimize irrigation practices and ensure sustainable water usage in agricultural fields.

The core component of the smart irrigation system is the integration of sensors to measure soil moisture levels continuously. These sensors are strategically placed in the crop fields to monitor the moisture content of the soil. By collecting real-time data on soil moisture, the system gains valuable insights into the soil's water-holding capacity and the crops' actual water requirements.

The IoT technology plays a vital role in enabling seamless communication between the sensors and the central system. The data collected from the sensors is transmitted to the cloud, where it is securely stored and analyzed. Leveraging cloud-based data storage and analysis offers the advantage of scalability, as large volumes of data can be efficiently managed, and advanced analytics can be applied to extract meaningful patterns and trends

M. N. Rajkumar, S. Abinaya, and V. V. Kumar [3] introduces an innovative automated watering system for agricultural applications, leveraging the power of IoT (Internet of Things) technologies. The primary objective of this system is to optimize irrigation practices and enhance agricultural productivity while ensuring efficient water usage.

The system's functioning revolves around the use of IoT devices, specifically soil moisture sensors, which are strategically placed in the fields to monitor the moisture levels in the soil. These sensors continuously collect real-time data on the soil's moisture content, providing crucial insights into the water requirements of the crops.

The gathered soil moisture data is then fed into an intelligent algorithm integrated into the system. This algorithm analyzes the data and makes informed decisions regarding the optimal watering plan for the crops. By utilizing the latest advancements in IoT and data processing, the system can dynamically adjust the irrigation schedule based on the changing needs of the crops. In this circumstance, our project aims to optimize water distribution for crops by leveraging advanced technology and smart decision-making algorithms. At the core of our system lies the deployment of sensors to continuously monitor essential parameters, including soil temperature and moisture levels. These sensors act as data nodes, relaying real-time information from the agricultural fields to a cloud-based server, which serves as the central control hub.

D. Mishra, A. Khan, R. Tiwari, and S. Upadhay [4] proposes an innovative and automated irrigation system that leverages the Internet of Things (IoT) technology to enhance agricultural practices. The primary goal of the study is to design a smart irrigation system that can efficiently manage water distribution in crop fields, leading to improved crop yields and water conservation. The system's architecture involves the deployment of various sensors, including soil temperature and moisture sensors, strategically placed across the agricultural field. These sensors continuously monitor and collect real-time data on the soil's temperature and moisture levels. The data collected by the sensors is then transmitted wirelessly to a central cloud server for further processing and analysis.

The cloud server plays a crucial role in the system as it acts as the data hub and performs advanced data processing and analysis tasks. It receives the sensor data from multiple locations in the field and applies intelligent algorithms to analyze the data. The analysis includes identifying patterns, trends, and critical thresholds related to soil temperature and moisture.

Based on the analyzed data, the system determines the irrigation requirements of the crops. It uses pre-defined criteria and thresholds set by agronomic experts to initiate the irrigation process at optimal times. When the soil moisture level falls below a specific threshold or when the soil temperature indicates potential stress for the crops, the system triggers the irrigation mechanism to supply water.

In our project, we are aiming to address the challenges posed by traditional irrigation methods by leveraging IoT technology and data-driven decision-making. To achieve this, we are deploying a network of sensors, including rain sensor and moisture sensors, strategically placed throughout the agricultural fields. These sensors continuously monitor and collect real-time data on soil conditions and rain status. The collected sensor data is transmitted wirelessly to a centralized cloud server. The cloud server plays a vital role in the system as it serves as a repository for the data and performs advanced data processing and analysis. Through intelligent algorithms, the server analyzes the sensor data to identify patterns, trends, and critical thresholds related to soil conditions.

S. B. Saraf and D. H. Gawali [5] introduce a smart irrigation system that utilizes Internet of Things (IoT) technology for efficient monitoring and control of irrigation processes in agriculture. The key focus of their work is to develop a system that addresses the challenges of traditional irrigation practices, which often lead to water wastage and inefficient water usage. The authors propose a solution that combines IoT devices and sensors to measure critical parameters like temperature and soil moisture in the crop fields.

By incorporating these sensors into the irrigation system, the authors enable real-time data collection on environmental conditions and soil moisture levels. This data is pivotal in determining the precise water requirements of the crops, allowing for intelligent and data-driven irrigation decisions.

The main focus of our project is to leverage IoT technology to create an intelligent irrigation system that can effectively monitor and control irrigation processes in agricultural fields. Inspired by the paper, we aim to utilize sensors to measure key parameters such as rain status and soil moisture in real-time.

A. Imteaj, T. Rahman, M. K. Hossain, and S. Zaman [6] propose an innovative irrigation system that leverages Raspberry Pi and IoT technology to create an autonomous watering system for agriculture. The primary objective of the study is to develop a smart irrigation solution that can autonomously control the irrigation process based on real-time soil moisture data. The system aims to address the challenges of water waste in conventional irrigation methods and improve water consumption effectiveness in agriculture.

The core components of our system include a Node MCU 8266 controller and soil moisture sensors. The Node MCU 8266 serves as the brain of the system, facilitating data processing, decision-making, and communication. The soil moisture sensors are strategically placed in the crop fields to monitor the moisture content of the soil continuously. Inspired by the paper, we are implementing a similar IoT-based approach to create an intelligent irrigation system that can make informed decisions based on real-time soil moisture data and rain sensor. To achieve this, we are utilizing a Node MCU 8266 as the central processing unit of our system. The Node MCU 8266 enables us to handle data processing, analysis, and decision-making, thus making the system capable of autonomously managing irrigation.

H.T.Ingale , N.N.Kasat [7] aimed at water conservation in agriculture. The authors emphasize the importance of efficient water usage in farming and highlight the challenges of unnecessary water wastage due to lack of awareness among farmers about optimal water supply for different plants.

The proposed system utilizes sensors to monitor humidity levels and weather conditions. By using microcontrollers and these sensors, the system can provide real-time information about changing humidity levels in the environment. This data enables farmers to schedule the appropriate timing for water supply, ensuring precise and efficient irrigation practices.

The authors recognize the significance of agriculture in the Indian economy, where seventy percent of the economy depends on this sector. They believe that implementing an automated irrigation system with precise water supply capabilities can have a positive impact on agricultural productivity and water conservation, ultimately benefiting the overall economy. According to the fact, the core focus of our project is to leverage IoT technology to create an intelligent irrigation system that can effectively monitor and control irrigation processes in agricultural fields. Inspired by the paper, we aim to utilize sensors to measure key parameters such as rain status and soil moisture in real-time.

Chapter 03: Methodology

The main purpose of this project is building an automated system that controls the motors for irrigation of the field. Our proposed system contains multiple sensor devices (soil moisture sensor, rain sensor) and controlling techniques. We have discussed here about our system in details in the following subsections-

3.1 Problem Statement

The fundamental challenge addressed in this project is the need to decrease the excessive use of fresh water in conventional irrigation systems while simultaneously automating the whole irrigation process using a cost-effective method . The existing irrigation techniques generally lead to inefficient water use, with no association between excessive water usage and enhanced agricultural yields. The key to optimizing agricultural output depends in the exact and effective exploitation of water resources [1].

The project intends to design a smart irrigation system that optimizes water consumption and boosts agricultural production by automating the irrigation process. The system should allow farmers to select between human intervention and completely automated operation, providing flexibility and ease in controlling the irrigation process. By incorporating automation, the system can decrease water loss and enhance sustainable agriculture practices.

The effective completion of this project will help to water conservation efforts in agriculture, enhance crop output, and encourage efficient resource management. The smart irrigation system's cost-effectiveness and automation capabilities are vital in allowing farmers to embrace sustainable practices, lowering their ecological imprint, and assuring long-term agricultural resilience.

3.2 Proposed System

Water constraint is causing a lot of challenges in the agriculture area these days. Smart irrigation technology has been introduced to help farmers in conquering their challenges. Various sensors are coupled to the NodeMCU 8266 model in this system, including soil moisture, and rain sensors [4]. The model is also equipped with a Wi-Fi connection in built. It is connected to internet server by WIFI and the data are uploaded to the Blynk server, as a result we can observe our field and also administer the system using mobile app from keeping distance from the system. The NodeMCU 8266 is coupled to relay which is later connected to the motor and light, respectively [5]. The pump will be switched ON/OFF automatically by the relay circuit if the measured value exceeds the threshold values stated in the program according to the written logic. The relay circuit is attached to the drive circuit, which helps to modify the voltage. Through the WIFI module, the farmer will be alerted of the current field status, which will also be updated on the web page. The farmer may get information on the state of the field at any moment using this technique. Machine learning is applied for evaluating the received data and saving our anticipated value for further comparison. In the future, some machine learning approaches directly build judgements from the analysis of the obtained data maintained in the storage or cloud and the conclusions will be given to the relay module. Using WIFI technology will warn our farmers through SMS via mobile [6]. The spectator may examine the recent conditions of the field. If any problem occurs, user will be notified by the system via application software. All type of data collected the sensors and after analyzing the obtained data, the processed report will be uploaded into the cloud and we may see the reports using application software installed in mobile or computer.

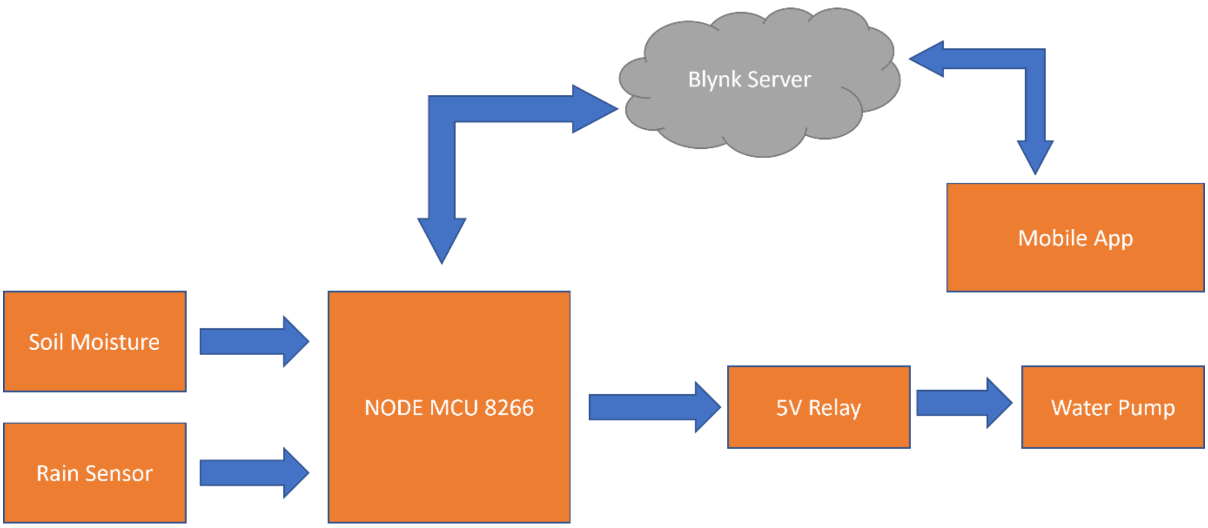


Figure 3: Irrigation Control System.

How our system will be controlled and monitored from the collected data and how will our system work are described in the following flowchart:

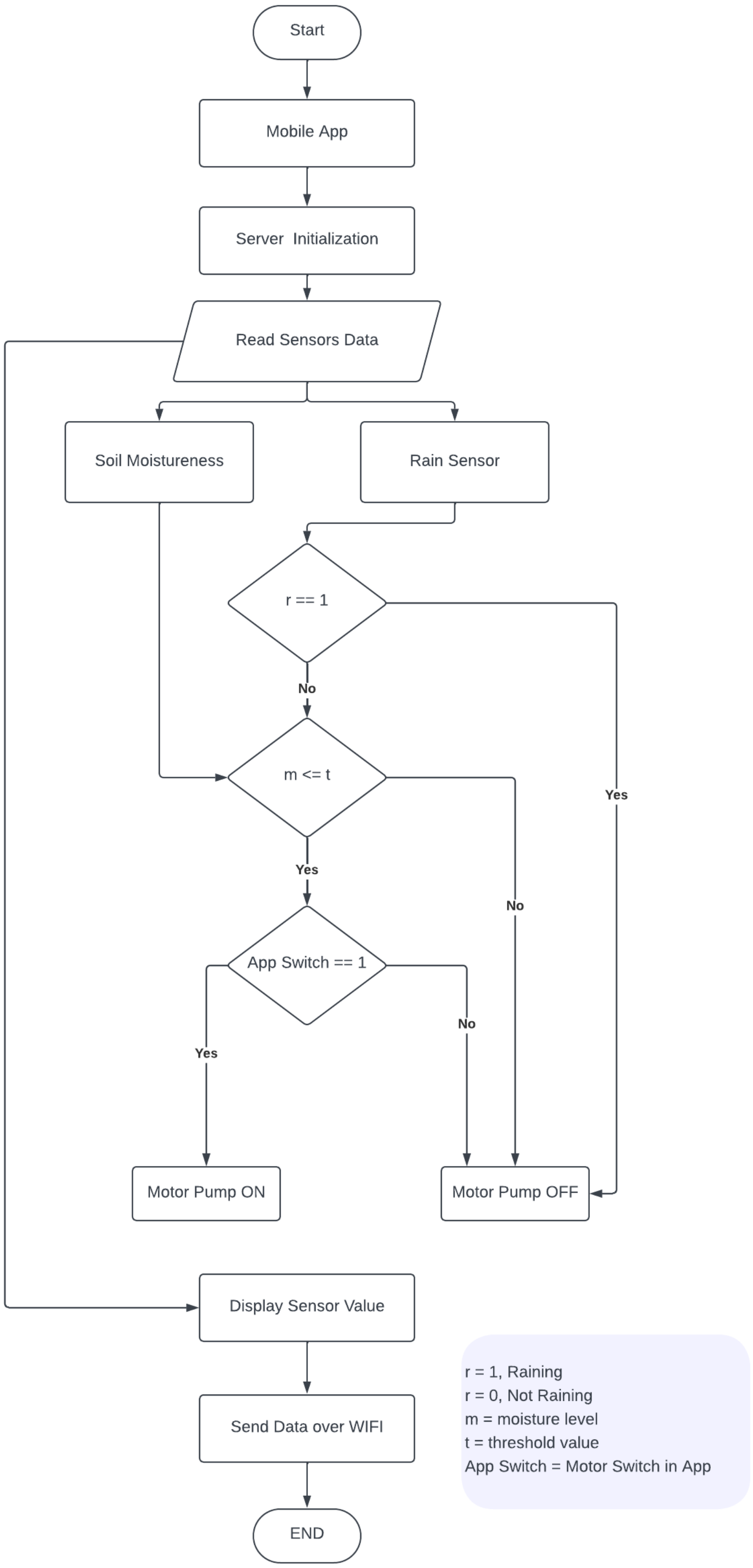


Figure 4: Flow chart of Automated irrigation System

3.3 Block Diagram

The block diagram of the project is illustrated below:

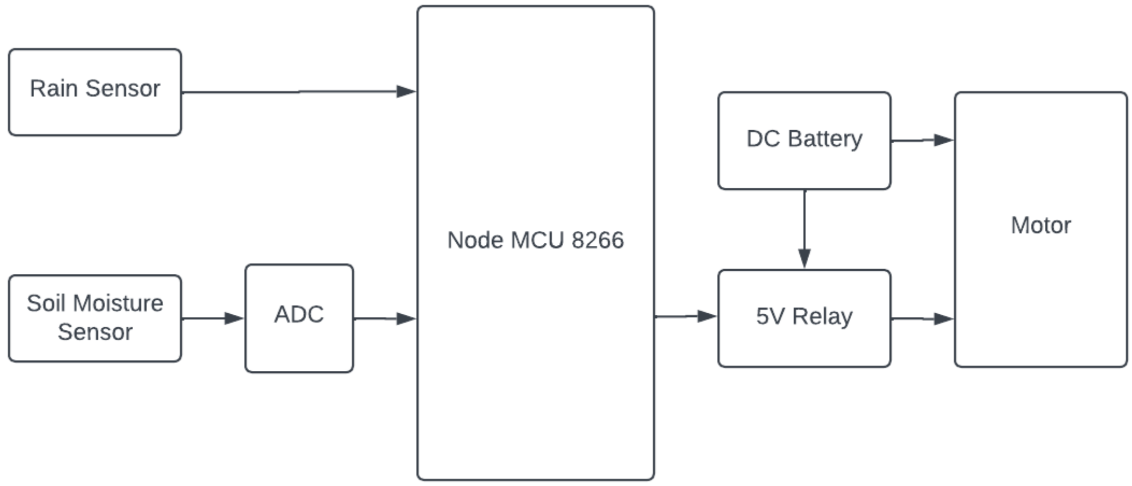


Figure 4: Block Diagram of the project

3.4 Necessary Equipment

A sensor is an electronic device, component, or module that detects changes or whatever changes happens within its surroundings and communicates the information to other electronics, most typically a computer processor. Sensors, to put it simply, are devices that transform physical qualities into electrical impulses. A sensor's sensitivity refers to how much its output varies when the measured value changes. A Sensor employs serial communication for transmitting data to the main controller. According to the collected analog signal, the impulses of current change and the values of the current sent serially to system controller.

Our system contains multiple sensors, which are listed here -

3.4.1 Soil moisture sensor:

This sensor is used to detect soil moisture content, commonly known as a hygrometer. It is very handy for establishing automated irrigation systems or monitoring the soil moisture levels of plants. The sensor contains a needle probe with two exposed electrodes that are placed into the soil to detect its water content. This probe acts as a variable resistor, meaning its resistance fluctuates with shifting soil moisture.

The sensor consists of two fundamental components: the probe and the module. The probe, as discussed earlier, is responsible for monitoring the soil moisture. The module works as an interface between the sensor and the Arduino microcontroller. It takes the resistance data from the probe and transforms it into an analog output (AO) signal. Additionally, the module digitizes the signal using an LM393 High Precision Sensor and enables access to the digital outcome (DO) pin.

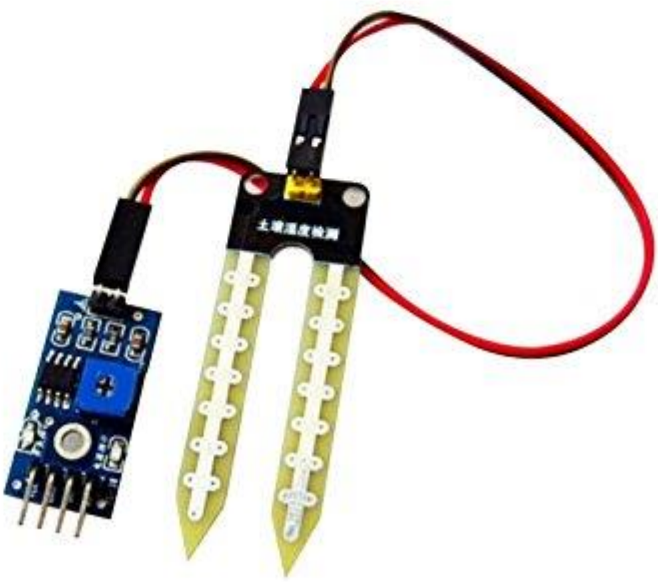


Figure 5: Soil Moisture Sensor

3.4.2 Rain Sensor:

This unique sensor is supposed to give data to a system, revealing whether it's raining or not. Its functionality is quite simple. The sensor consists of open copper traces that work as a variable resistor, analogous to a potentiometer. The resistance of these traces changes dependent on the amount of water present on the sensor's surface. More water on the surface leads to better conductivity and lower resistance, whereas less water results in weaker conductivity and higher resistance. By measuring the resistance and turning it into an output voltage, the sensor can determine the presence or absence of rain.

To power the sensor, users need to connect its VCC pin to the 5V supply of a Node MCU. However, a noted drawback with these sensors is their limited lifetime when exposed to moisture. To minimize this problem, it's advisable to power the sensor only when collecting readings. This may be achieved by connecting the VCC pin to a digital pin on a NodeMCU and regulating its power by setting the pin to HIGH or LOW as necessary. The sensor's power consumption is minimal, with a total of roughly 8 mA when both LEDs are active, making it safe to power from an NodeMCU’s digital pin. Therefore, you should connect the VCC pin of the sensor module to the Node MCU's digital pin 3 and also link the GND pin to the ground. Lastly, wire the D3 pin of the sensor module to the Node MCU's digital pin 3.

Figure 6: Rain Sensor

3.4.3 The NodeMCU 8266:

NodeMCU is a popular open-source electronics platform, much like Arduino, but with a focus on Internet of Things (IoT) applications. It is based on the ESP8266 microcontroller, which integrates Wi-Fi connectivity, making it ideal for IoT projects that require wireless communication and internet connectivity. NodeMCU provides a simple hardware and software platform for building IoT applications with ease.

Key features and uses of NodeMCU include:

- ✓ **ESP8266 Microcontroller**: NodeMCU is powered by the ESP8266 microcontroller, which combines a powerful processor with built-in Wi-Fi capabilities, enabling seamless communication with the internet.
- ✓ **Arduino-like Programming**: NodeMCU programming is based on a simplified version of the Lua programming language. It’s easy-to-understand syntax makes it accessible for beginners and experienced developers alike.
- ✓ **IoT Connectivity**: NodeMCU's Wi-Fi connectivity allows it to communicate with cloud services, web servers, and other IoT devices. This makes it an excellent choice for building smart home systems, remote monitoring applications, and IoT-enabled automation.
- ✓ **Integrated Development Environment (IDE)**: Similar to Arduino, NodeMCU has its own IDE for programming and uploading code to the board. This IDE simplifies the development process and offers a familiar interface for developers accustomed to working with Arduino.
- ✓ **Analog and Digital I/O**: NodeMCU features both analog and digital input/output pins, allowing it to interface with a wide range of sensors and actuators.

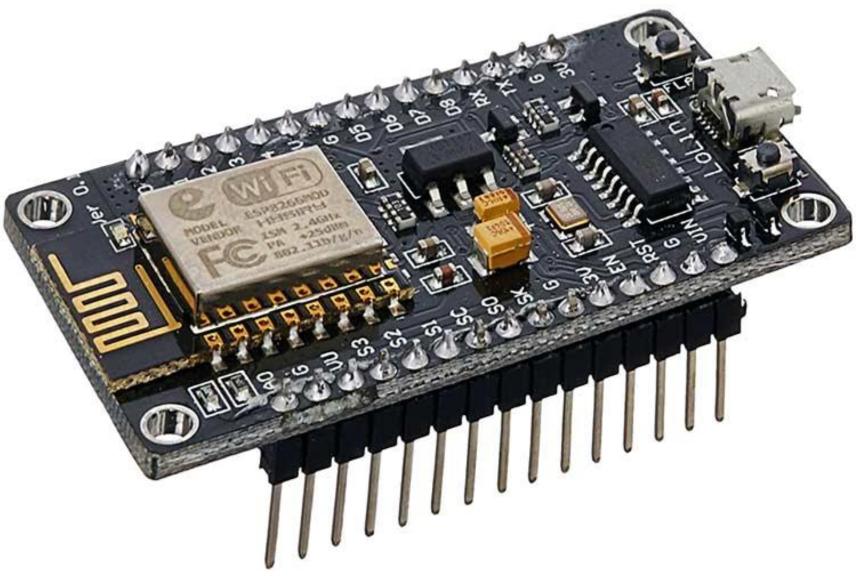


Figure 7: Node MCU 8266

3.4.4 Relay (Motors Controller):

The motors in the system are controlled by the relay module. The relay module is an electrically powered switch that allows you to turn on or off a circuit with more voltage and/or current than a micro-controller can handle. The Arduino is connected to the devices using relay, as depicted in the figure. The relay may be utilized as a switch to turn on or off the devices in this scenario. In this project a five-volt relay has been employed. It operates the motors by receiving data from the controller. The pump motor operates on more than 5 volts, there requires more electricity for the motor. here We have been utilized a 9V DC power provider for the power of the motor. We may also utilize AC power.

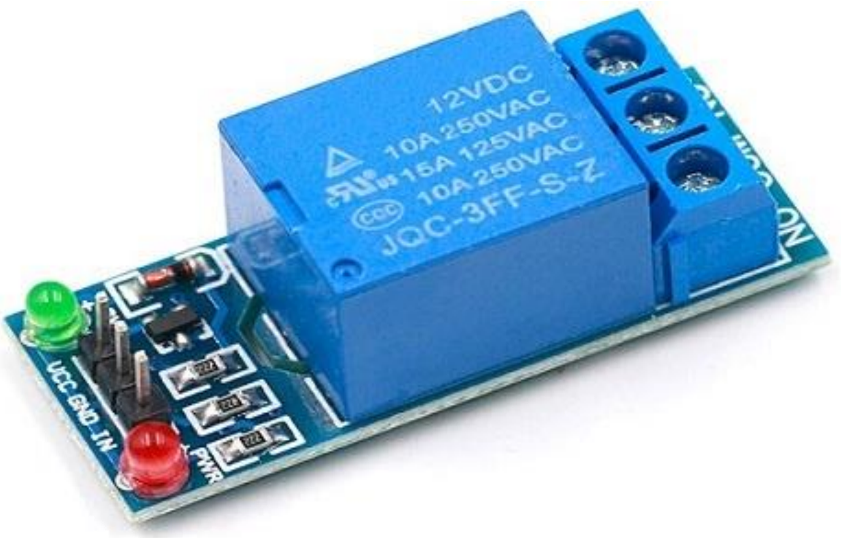


Figure 8: Relay (Motor Controller)

3.4.5 Motor Pump:

The water pump used in intelligent agriculture is a major component of current agricultural methods aimed at optimizing water consumption and enhancing crop output. These smart water pumps are meant to be energy-efficient, dependable, and controlled, making them suitable for irrigation systems in precision agriculture. Equipped with sensors and IoT capabilities, the pump may be incorporated into a smart agricultural network, allowing automatic and remotely controlled watering programs.

Figure 9: Water Pump

3.4.6 Breadboard:

The In electronics, a breadboard is a flexible prototype device that may be used to develop and test circuit designs without the necessity of solder. It is built from of a grid of linked clips or connections that allow it straightforward to plug in and connect electronic devices such resistors, capacitors, and microcontrollers. Breadboards are vital for the design and testing of electrical equipment and sensor networks in the context of smart farming. They provide a simple and rapid platform for connecting and testing a number of smart farming components, including soil moisture sensors, temperature and also humidity sensors, water pumps, and actuators. By utilizing a breadboard, farmers and engineers may experiment with alternative circuit designs and sensor integrations, ensuring that the smart farming system performs effectively and precisely.

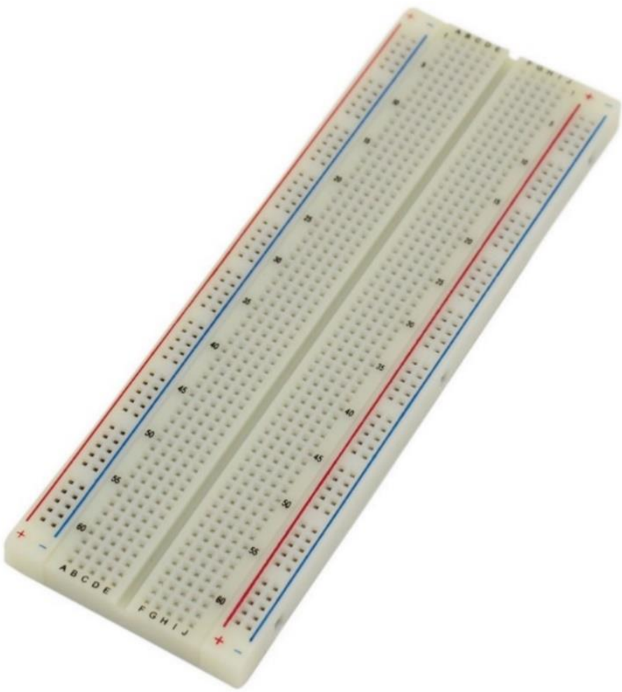


Figure 10: Bread Board

3.4.7 9V Battery: 9V battery is a vital power source for different electrical components and devices employed in agricultural automation and monitoring systems. These batteries, frequently in the form of no rechargeable Li-ion or LiPo batteries, offer a stable and portable energy source to power smart agricultural equipment like as soil moisture sensors, temperature and humidity sensors, wireless communication modules, actuators, and microcontrollers. Due to their small size and high energy density, 9V batteries are well-suited for integration into smart farming installations, enabling the deployment of wireless and remote monitoring systems over broad agricultural areas. With correct power management and effective use, 9V batteries help to improving irrigation systems, automated equipment, and environmental monitoring in smart farming, eventually leading to higher crop production, water conservation, and overall sustainable agricultural practices.

Figure 11: 9V Battery

3.5 Project Overview

To utilize the full IoT-based smart irrigation system, the farmer has to follow these steps:

Figure 12: Blynk Setup & Configure

1. **Creating an Account on Blynk**: The farmer has to download the Blynk mobile application or visit the Blynk website to create an account. They will join up using their email address, and following successful registration, they will get their unique Blynk account credentials, including the Auth Token.
2. **Installing Blynk Library**: In the Arduino Integrated Development Environment (IDE), the farmer will install the Blynk library by clicking to "Sketch" > "Include Library" > "Manage Libraries." They will search for "Blynk" and click "Install" to add the Blynk library to the Arduino IDE.
3. **Obtaining Auth Token**: The farmer will obtain an Auth Token when they establish an account on Blynk. This token acts as an authentication key for the Arduino device to interact with the Blynk cloud server.
4. **Configuring NodeMCU and Wi-Fi**: In the Arduino IDE, the farmer will load the example sketch given by the Blynk library, particularly built for NodeMCU (ESP8266). They will update the placeholder values with their Wi-Fi credentials (SSID and password) to allow the NodeMCU to connect to their Wi-Fi network.
5. **Adding Widgets in Blynk App**: Using the Blynk app, the farmer will create a new project and add widgets such as buttons, sliders, gauges, and graphs to the project dashboard. These widgets will serve as controls and indications to interact with the smart irrigation system.

6. **Programming NodeMCU**: The farmer will alter the sample sketch code for NodeMCU to incorporate the Auth Token received from the Blynk account. They will also setup the GPIO pins attached to the sensors and actuators in the code.
7. **Establishing Connection**: The farmer will upload the changed sketch to the NodeMCU board using a USB cord. Once the code is successfully uploaded, the NodeMCU will start operating the smart watering system. It will create a connection to the Wi-Fi network and the Blynk cloud server using the Auth Token.
8. **Configuring Widgets and Functionality**: In the Blynk app, the farmer will attach each widget to the proper virtual pin on the NodeMCU. For example, they may attach a button widget to the virtual pin that controls the water pump. They will also select the required thresholds for soil moisture levels and design the system's reaction accordingly.

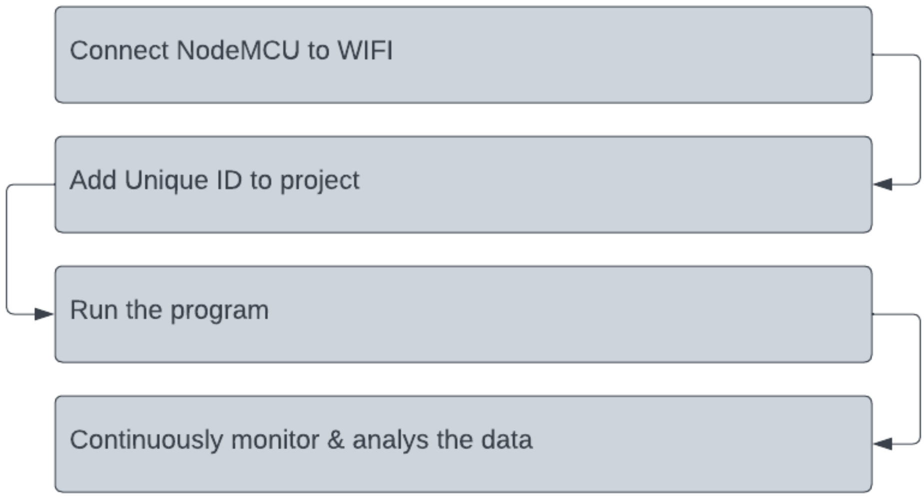


Figure 13: NodeMCU setup

9. **Monitoring and Controlling**: With the NodeMCU linked to Blynk, the farmer can now remotely monitor the soil moisture levels, rain data, and other sensor readings on the Blynk app's dashboard. They may also regulate the irrigation process by turning the water pump on or off using the app's widgets.
10. **Data Analysis and Feedback**: The Blynk app will continually receive data from the NodeMCU, offering real-time insights into the tea garden's watering status. The famrer may examine this data to improve irrigation schedules, save water, and boost crop output.

By following these procedures, the farmer may successfully utilize the complete IoT-based smart irrigation system, taking use of Blynk's user-friendly interface to remotely manage and monitor the watering process for their tea garden.

Chapter 04: Result and Discussion

The planned system has been implemented completely. The system will save resources, primarily water resources, with the right usage of these sensors. The technology will cut water use by at least 30%. In order to save many millions of gallons of water that can then be utilized in other regions, a lot of farmer land will be combined utilizing this automated technique.

Furthermore, since the crops won't be harmed by overwatering, the technique will result in a little higher yield of crops. Consequently, it will benefit farmers in both ways.

The related hardwire in the project are:

- NodeMCU 8266
- Moisture Sensor
- Rain Sensor
- Relays and motor
- DC Battery and Jumping wires.
- Others

4.1 Hardwire Cost

The cost for the related hardwire for the project and given in below table:

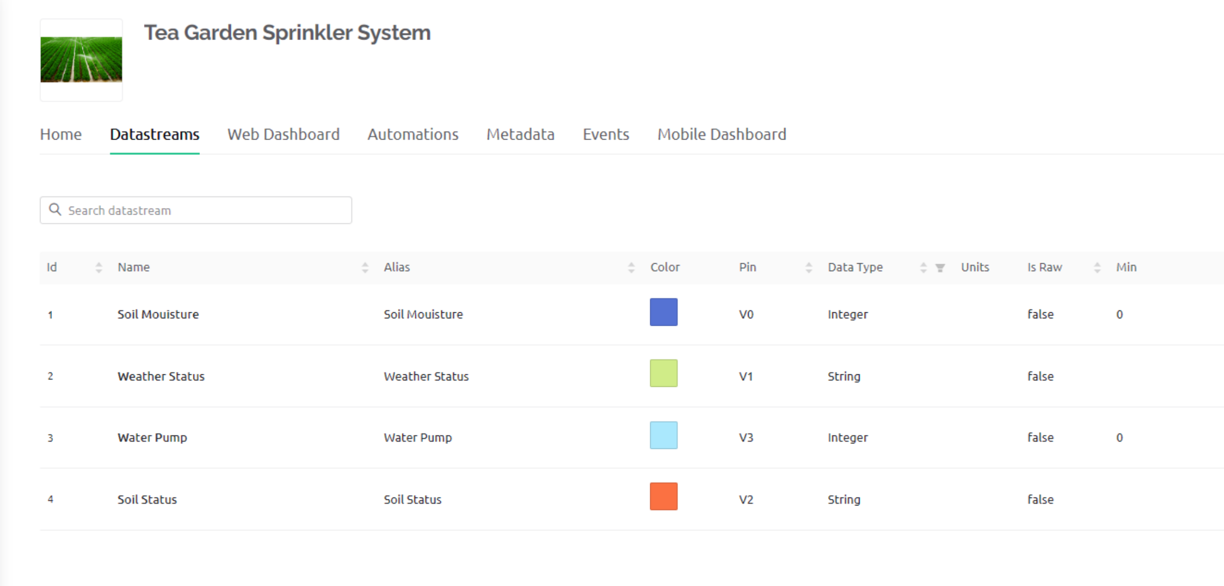
Table 1: Hardwire cost table

Hardware	Cost in BDT
NodeMCU ESP 8266	420
Soil Moisture Sensor	180
Relay 5V Module	120
Water Pump	100
Breadboard	120
9 V battery	70
Jumper Wires	50
Others	160
Total	1220

All of the equipment is reasonably priced. This price may change over time and across different locations.

4.2 Implementation

The Blynk dashboard is used in the system's implementation. Additionally, we may manage the NodeMCU using the Blynk application. Here's some implementation figures:



Tea Garden Sprinkler System

Home | Datastreams | Web Dashboard | Automations | Metadata | Events | Mobile Dashboard

Search datastream

Id	Name	Alias	Color	Pin	Data Type	Units	Is Raw	Min
1	Soil Mouisture	Soil Mouisture		V0	Integer		false	0
2	Weather Status	Weather Status		V1	String		false	
3	Water Pump	Water Pump		V3	Integer		false	0
4	Soil Status	Soil Status		V2	String		false	

Figure 14: DataStream of the Dashboard

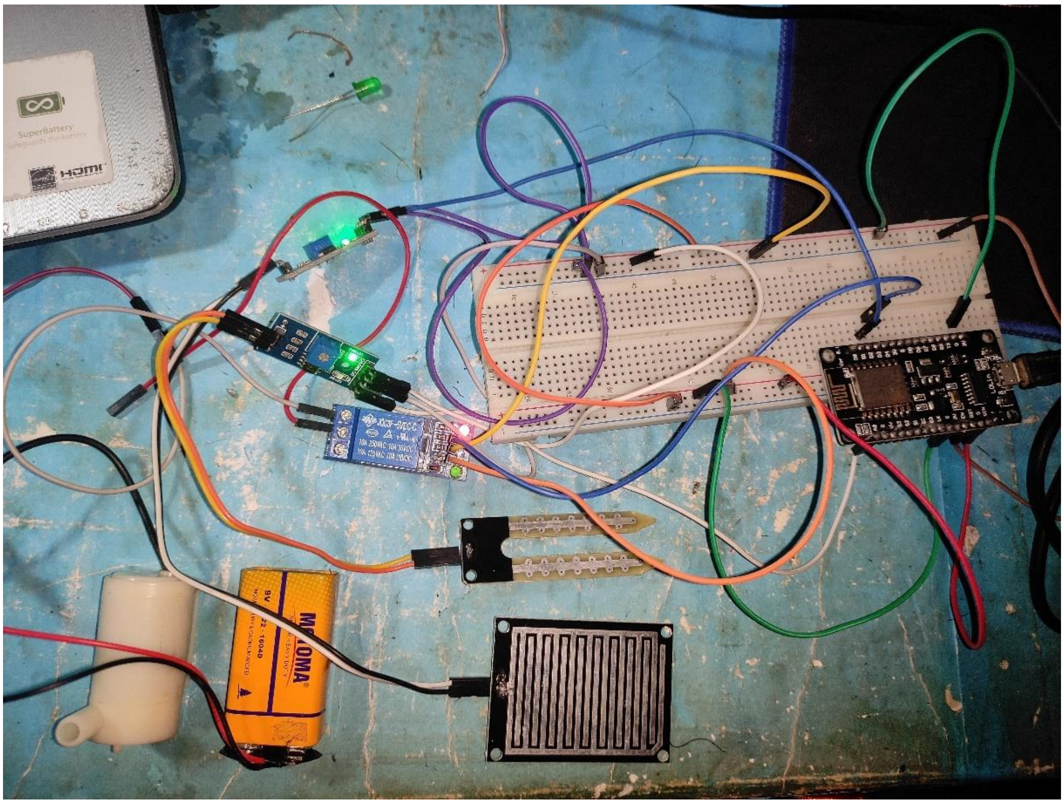


Figure 15: Irrigation Control System Implementation

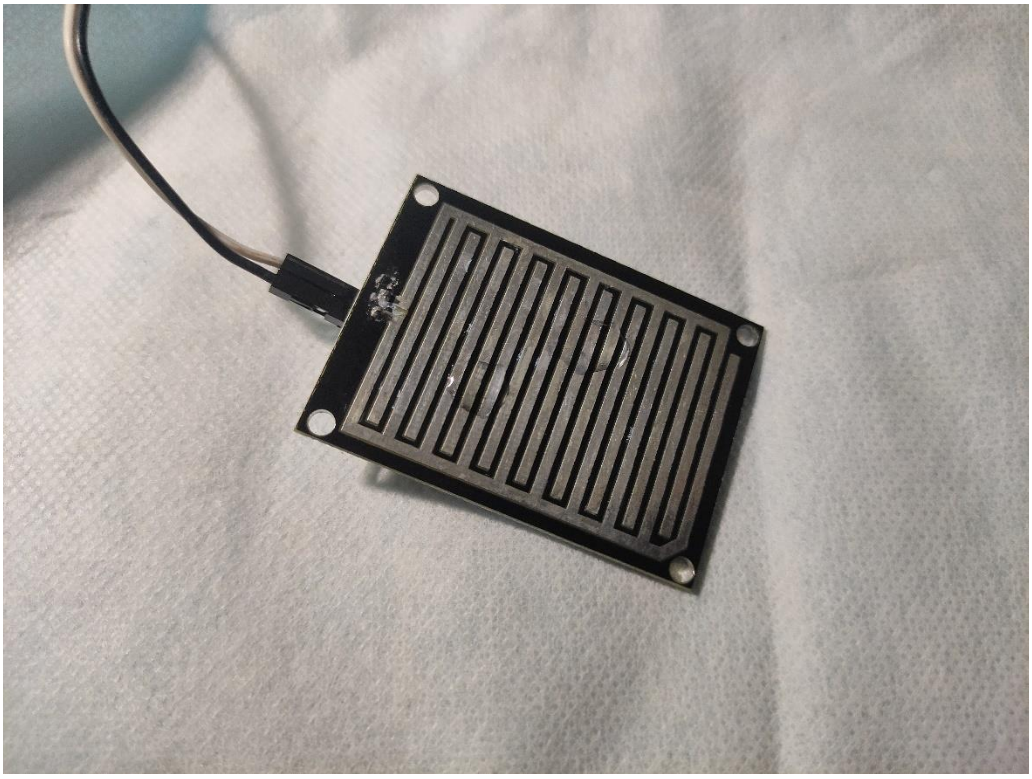


Figure 16: Rain Sensor sensing data

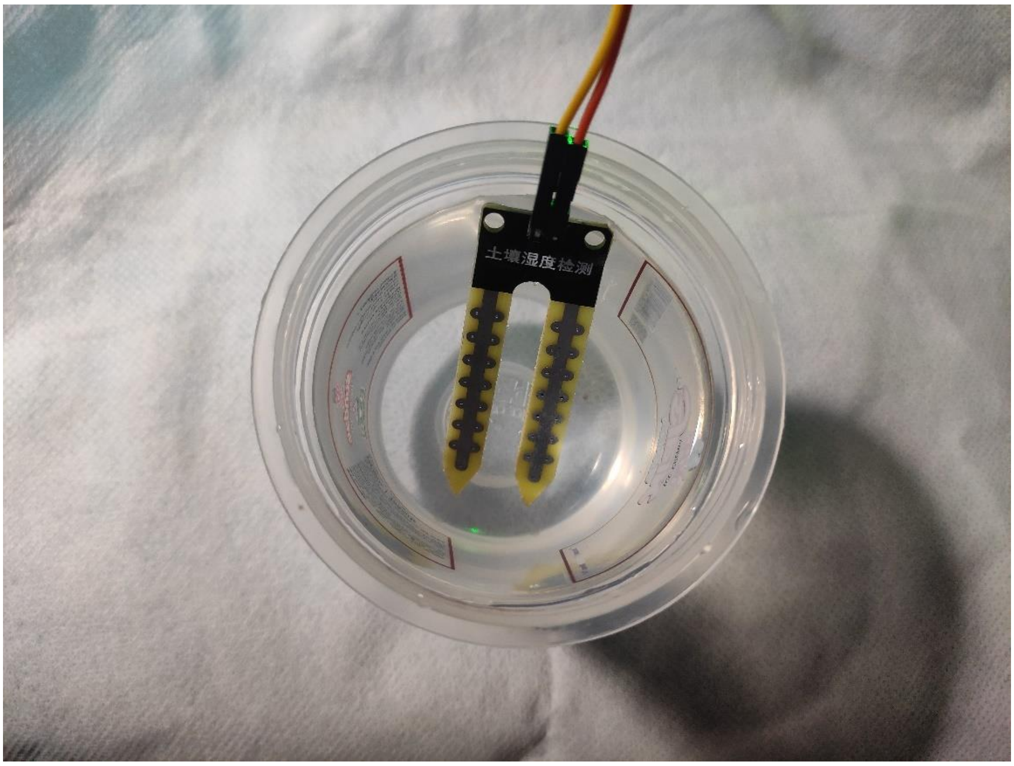


Figure 17: Soil Moisture level check

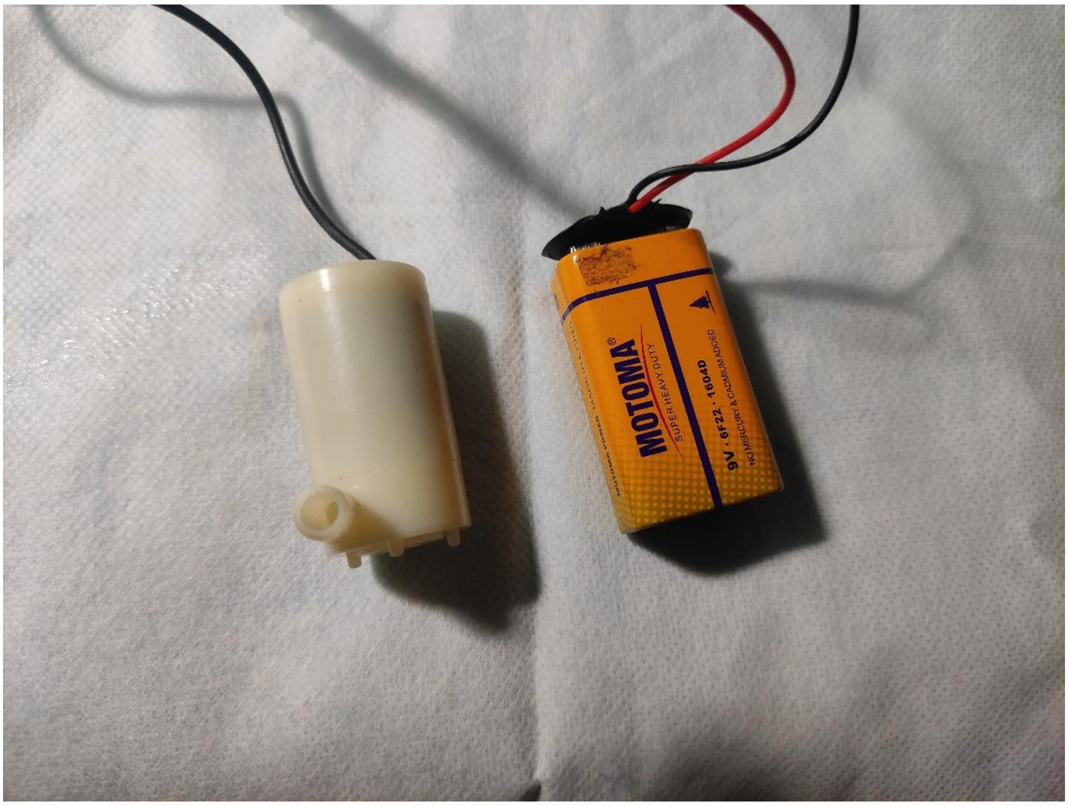


Figure 18: Battery and water pump

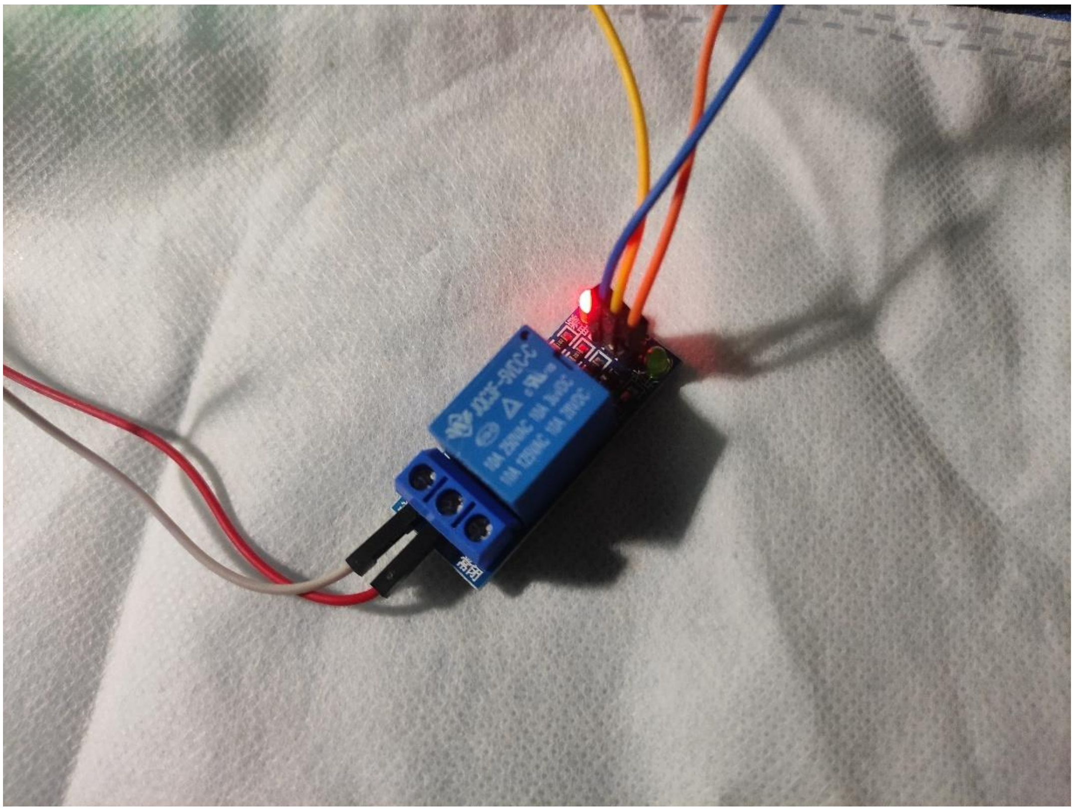


Figure 19: Relay Module

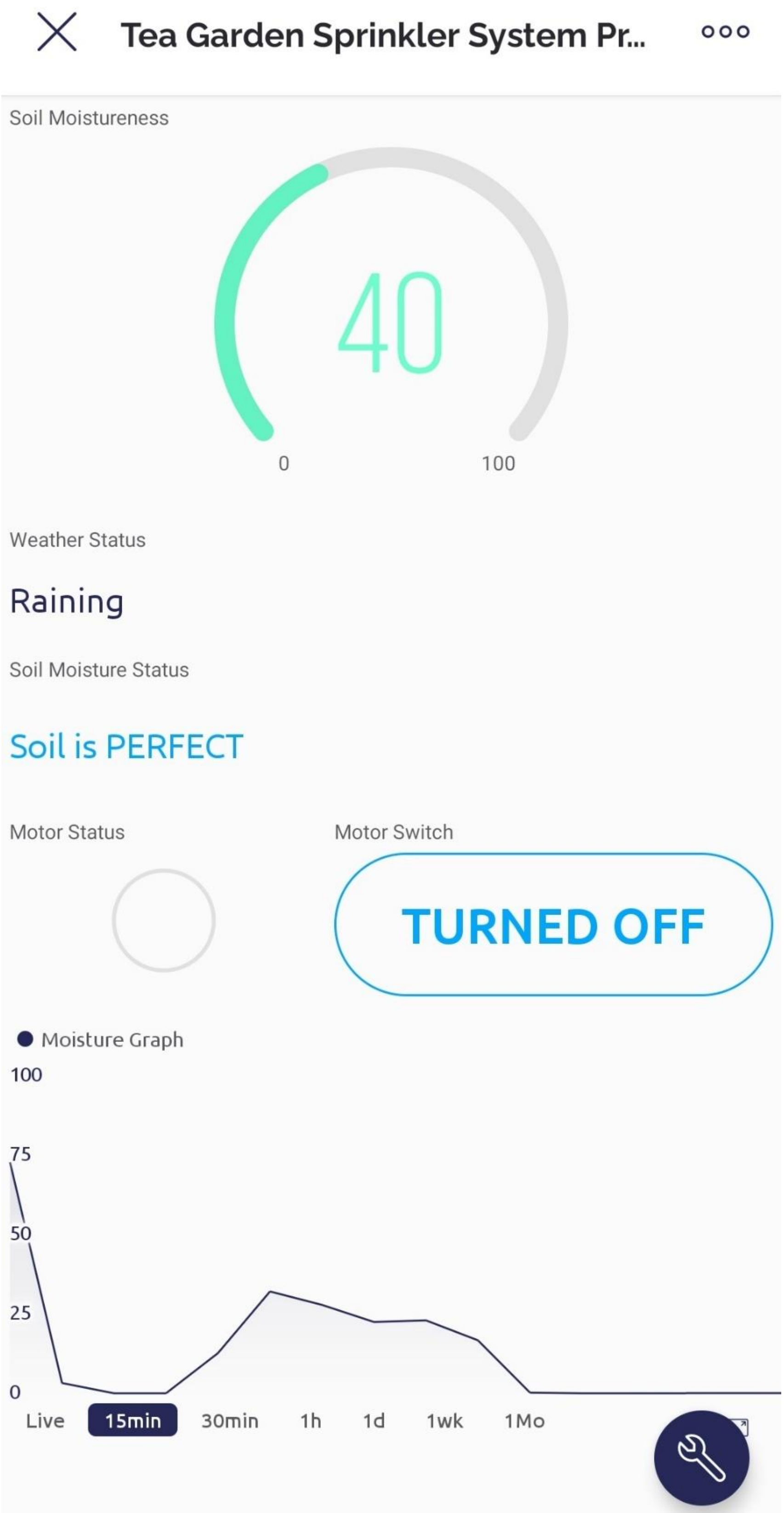


Figure 20: Mobile App Interface of the System

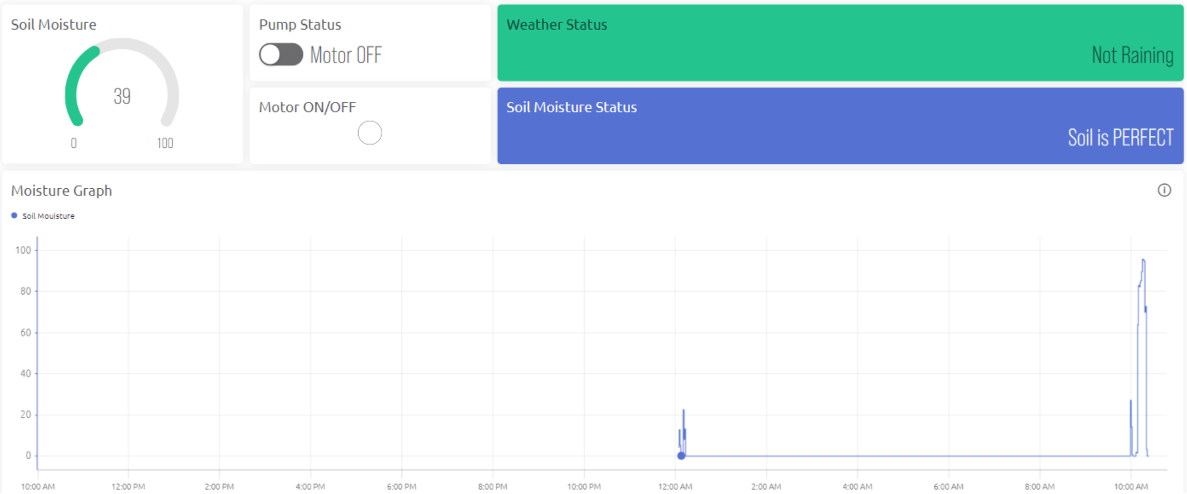


Figure 21: Web Interface of the System

4.3 Data Collection and Analysis:

The time of the obtained data was manually recorded by the clock one record every hour continuously throughout 3 days. The Arduino program's serial monitor was utilized to acquire sensor data. The sensor readings continued for a time. The moisture level, motor pump condition and time stamp per hour of the soil are indicated. The data below indicate the status of the water pump as it was monitored every hour. The following records are user and CSV in ThingSpeak server to create chart.

ThingSpeak is an open platform as a service and API for such Internet of Things that helps you collect, store, analyze, display, and act on sensor data. ThingSpeak is an Internet of the Things application platform. ThingSpeak enables users to construct an application based on sensor data. Real-time data processing, data collection, apps and visualizations are among the characteristics of ThingSpeak.

Table 2: Day 01 Moisture, Rain and Motor Status per hour

Time Stamp	Soil Moisture Level	Rain Data	Motor Status
1:00	0	1	ON
2:00	44	1	OFF
3:00	6	1	ON
4:00	36	1	OFF
5:00	21	1	OFF
6:00	4	1	ON
7:00	22	1	OFF
8:00	3	1	ON
9:00	8	1	ON
10:00	38	1	OFF
11:00	21	1	OFF
12:00	41	1	OFF
13:00	19	1	ON
14:00	21	1	OFF
15:00	47	1	OFF
16:00	0	1	ON
17:00	37	0	OFF
18:00	28	1	OFF
19:00	26	1	OFF
20:00	0	1	ON
21:00	20	1	ON
22:00	2	1	ON
23:00	15	1	ON

Table 3: Day 02 Moisture, Rain and Motor Status per hour

Time Stamp	Soil Moisture level	Rain Data	Motor Status
1:00	19	1	ON
2:00	37	1	OFF
3:00	43	1	OFF
4:00	3	1	ON
5:00	6	1	ON
6:00	13	1	ON
7:00	23	1	OFF
8:00	31	1	OFF
9:00	28	1	OFF
10:00	10	1	ON
11:00	44	1	OFF
12:00	49	1	OFF
13:00	7	1	ON
14:00	25	1	OFF
15:00	44	1	OFF
16:00	4	1	ON
17:00	48	0	OFF
18:00	14	1	ON
19:00	17	1	ON
20:00	21	1	OFF
21:00	0	1	ON
22:00	15	1	ON
23:00	48	1	OFF

Table 4: Day 03 Moisture, Rain and Motor Status per hour

Time Stamp	Soil Moisture level	Rain Data	Motor Status
1:00	42	1	OFF
2:00	49	1	OFF
3:00	4	1	ON
4:00	10	1	ON
5:00	7	1	ON
6:00	21	1	OFF
7:00	4	1	ON
8:00	43	1	OFF
9:00	17	1	ON
10:00	4	1	ON
11:00	16	1	ON
12:00	42	1	OFF
13:00	38	1	OFF
14:00	5	1	ON
15:00	32	1	OFF
16:00	12	1	ON
17:00	9	0	OFF
18:00	48	1	OFF
19:00	47	1	OFF
20:00	25	1	OFF
21:00	20	1	ON
22:00	45	1	OFF
23:00	4	1	ON

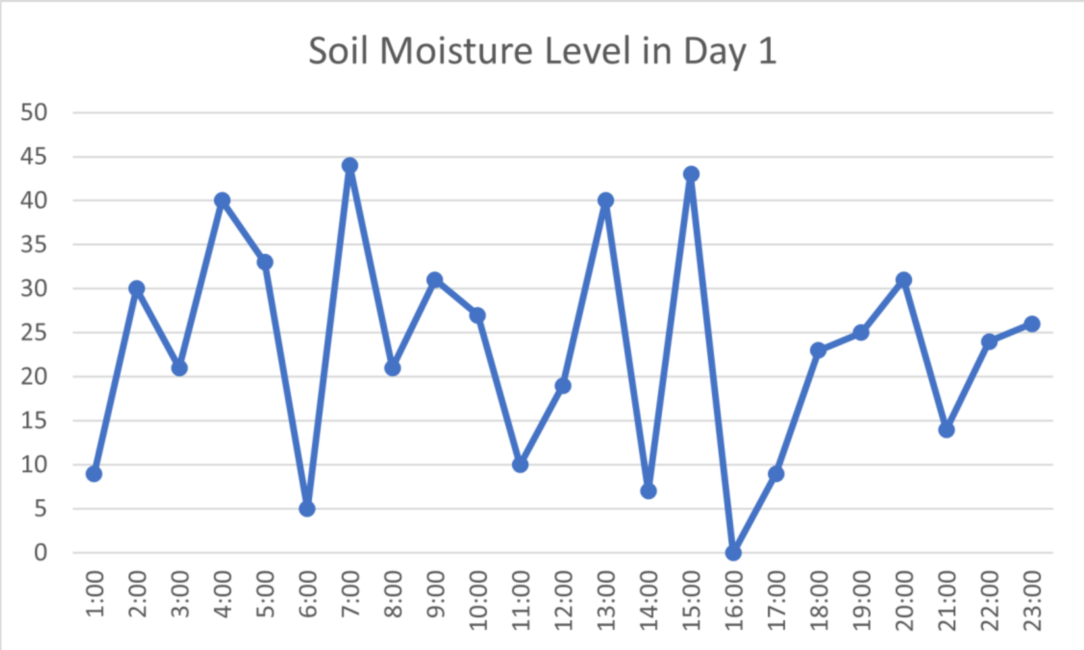


Figure 22: Moisture chart of Day 01

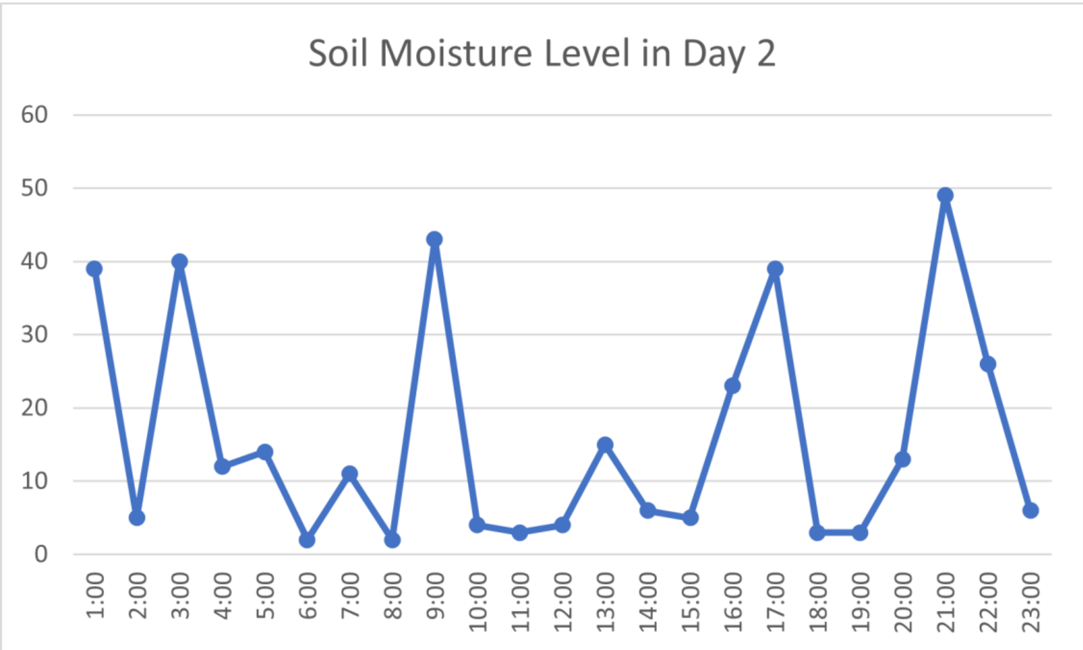


Figure 23: Moisture chart of Day 02

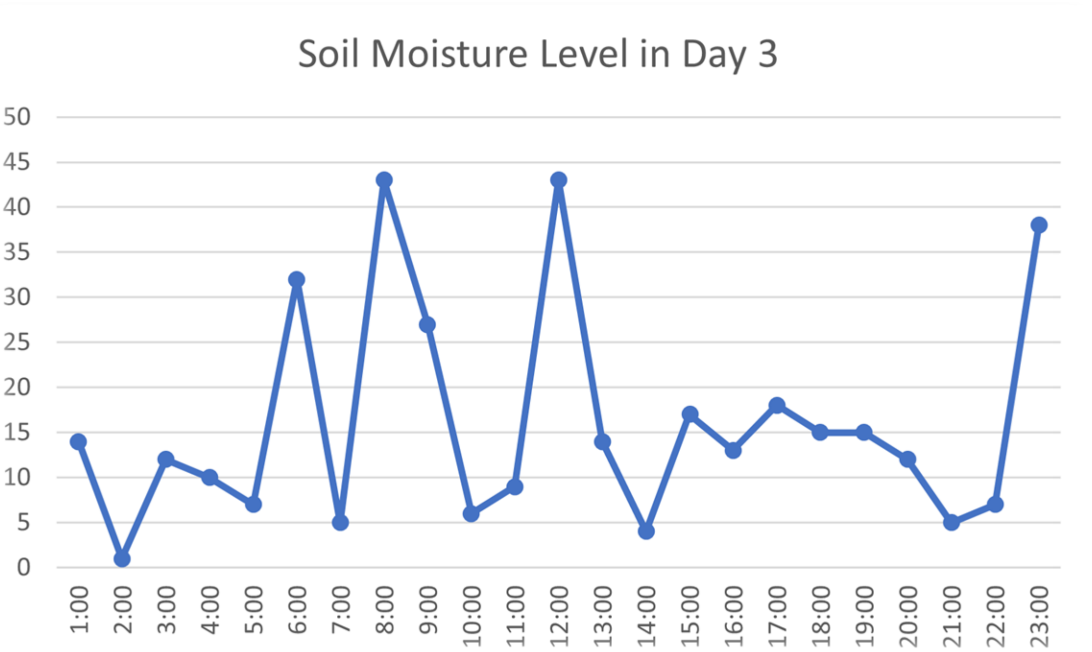
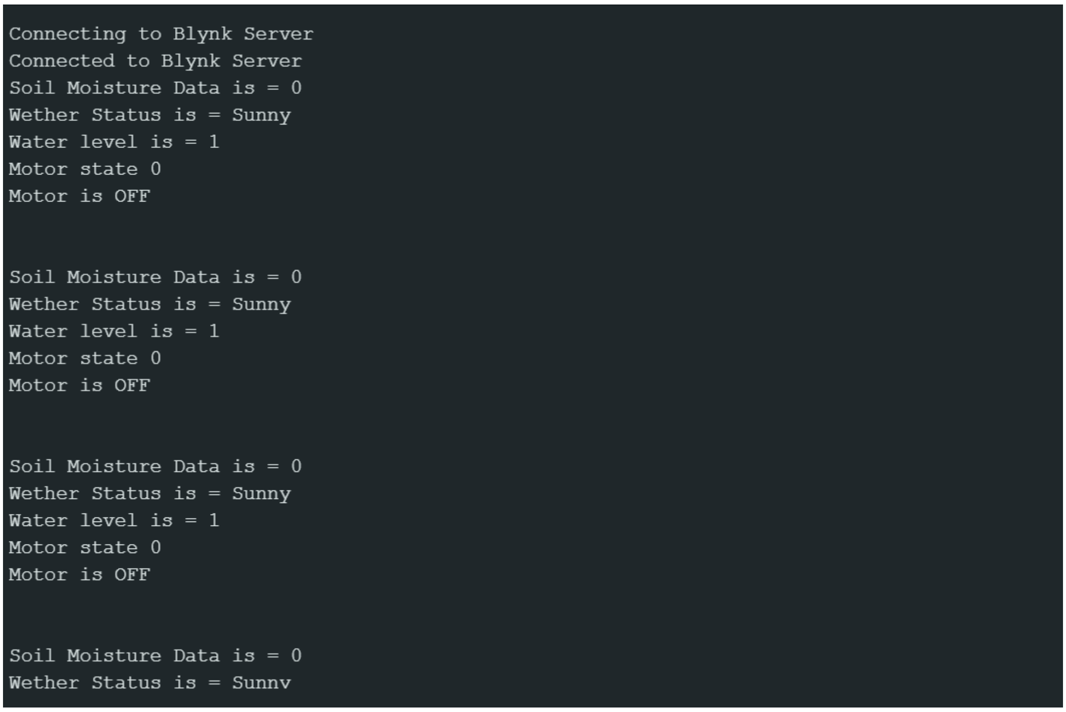


Figure 24: Moisture chart of Day 03

4.4 Result

Our system has multiple hardware components, they are NodeMCU (the system controller) , Soil moisture sensor, Rain sensor and relays and Motor Pump. Sensors collects data and sends these data to Blynk IoT cloud via WIFI module. The moisture sensor generally senses the soil moisture level of the field. The collecting data from the sensors after implementation are giver bellow.

The manual output value in the serial monitor-



```
Connecting to Blynk Server
Connected to Blynk Server
Soil Moisture Data is = 0
Wether Status is = Sunny
Water level is = 1
Motor state 0
Motor is OFF


Soil Moisture Data is = 0
Wether Status is = Sunny
Water level is = 1
Motor state 0
Motor is OFF


Soil Moisture Data is = 0
Wether Status is = Sunny
Water level is = 1
Motor state 0
Motor is OFF


Soil Moisture Data is = 0
Wether Status is = Sunny
```

Figure 25: The Output in the serial monitor in Arduino IDE

The system collects various data from the field like soil moisture and rain status etc. the system sends the data cloud server Blynk over built in WIFI module. Here we makes collected data graph from the collected data individually.

4.5 Expected Result

Ours is a fully automated system. It is based on information gathered in the field. Multiple sensors monitor the rain status and moisture level and the system collects this information and compares it to the standard necessary level. The pump or motors will be turned on or off automatically once the data has been analyzed. As a result, there is no room for excessive water consumption. It is possible to save money by lowering the amount of water used. It takes fewer manpower since it is an automated system. It also saves money, time, and effort. There are also some challenges that our system may be faced. IOT is complex network to implement. Sensors are temporary they would be damaged frequently. It may consume high cost. Unskilled farmers in our country do not much about our system etc.

Chapter 05: Conclusion and Future Works

5.1 Conclusion

The automated irrigation system is suited and cost-effective for enhanced water resources for agricultural production. A feedback control system would be included in the system, which would efficiently monitor and manage all plant growth and irrigation system operations. When it rains, flooding might be prevented if a rain gun sensor could be installed. It is feasible to collect rainwater, which may then be used to irrigate crops.

5.2 Future Works

Future developments for the IoT-based smart irrigation system for tea plantations are anticipated to be considerable. Integrating water quality sensors to track variables like pH, conductivity, dissolved oxygen, and nutrient levels is an essential improvement. With the use of this information, tea plants' health and development may be best achieved by precise watering tactics that are catered to their unique needs. Machine learning algorithms will also increase the system's intelligence by allowing it to evaluate past sensor data and weather trends. The system may use this data to identify abnormalities, offer exact water application methods, and make knowledgeable judgments regarding irrigation schedules.

References

[1] R. N. R. a. B. Sridhar, "Iot based smart crop-field monitoring and automation irrigation system," *2nd International Conference on Inventive Systems and Control (ICISC). ITEE,* p. 478–483, 2018.

[2] S. Rawal, "IoT based smart irrigation system," *International Journal of Computer Applications*, vol. 159, no. 8, p. 7–11, 2017.

[3] S. A. a. V. V. K. M. N. Rajkumar, "Intelligent irrigation system an IoT based approach," *International Conference on Innovations in Green Energy and Healthcare Technologies (IGEHT),* pp. 1-5, 2017.

[4] A. K. R. T. a. S. U. D. Mishra, "Automated irrigation system-iot based approach," *3rd International conference on internet of things: Smart Innovation and Usages (IoT-SIU),* pp. 1-4, 2018.

[5] S. B. S. a. D. H. Gawali, "Iot based smart irrigation monitoring and controlling system," *2nd IEEE International Conference on Recent Trends in Electronics, Information & Communication Technology (RTEICT),* p. 815–819, 2017.

[6] T. R. M. K. H. a. S. Z. A. Imteaj, "Iot based autonomous percipient irrigation system using raspberry pi," *19th International Conference on Computer and Information Technology (ICCIT),* p. 563–568, 2016.

Appendix

C

Code

The code for this system is given here:

```
#define WATER_PUMP_VIRTUAL V3

#define BLYNK_TEMPLATE_ID "TMPL65ba0tg7O"
#define BLYNK_TEMPLATE_NAME "Tea Garden Sprinkler System"

// Variables
int waterLevel;
int soilMoisture;
int rainSensorState;
bool blynkServerConnected = false;
int motorState;

// Function to read water level using ultrasonic sensor
int getWaterLevel()
{
    return digitalRead(waterLevelPin);
}

// Function to read rain sensor state (1 if wet, 0 if dry)
int getRainSensorState()
{
    return digitalRead(rainSensorPin);
}

// Function to read soil moisture sensor value (0 to 1023)
int getSoilMoisture()
{
    return analogRead(soilMoisturePin);
}

```

Figure 26: Declaration Code

```
void setup()
{
    Serial.begin(9600);

    pinMode(waterPump, OUTPUT);
    pinMode(rainSensorPin, INPUT);
    pinMode(soilMoisturePin, INPUT);
    pinMode(waterLevelPin, INPUT);

    // Connect to Wi-Fi
    WiFi.begin(ssid, pass);
    Serial.print("Connecting to WiFi");
    while (WiFi.status() != WL_CONNECTED)
    {
        delay(500);
        Serial.print(".");
    }
    Serial.println("\nConnected to WiFi");

    Blynk.begin(auth, ssid, pass);

    // (Optional) Set the frequency for reading sensors
    Blynk.virtualWrite(SOIL_MOISTURE_VIRTUAL, 100);  // 5 seconds
    Blynk.virtualWrite(WEATHER_STATUS_VIRTUAL, 100); // 5 seconds
    Blynk.virtualWrite(WATER_PUMP_VIRTUAL, 100);     // 5 seconds
}
```

Figure 27: Setup and Configure

```

void loop()
{
    // connecting to blynk server
    if (!blynkServerConnected)
    {
        Serial.print("\n\n\nConnecting to Blynk Server");
        while (!Blynk.connected())
        {
            delay(500);
            Serial.print(".");
        }
        Serial.println("\nConnected to Blynk Server");
        blynkServerConnected = true;
    }

    // soil moisture display section
    soilMoisture = getSoilMoisture();
    soilMoisture = (map(soilMoisture, 0, 1024, 0, 100) - 100) * -1;
    Blynk.virtualWrite(SOIL_MOISTURE_VIRTUAL, soilMoisture);
    Serial.print("Soil Moisture Data is = ");
    Serial.println(soilMoisture);
    if (soilMoisture < 20)
    {
        Blynk.virtualWrite(SOIL_MOISTURE_STATUS_VIRTUAL, "Soil is too DRY - Need Water");
    }
    else if (soilMoisture < 50 && soilMoisture >= 20)
    {
        Blynk.virtualWrite(SOIL_MOISTURE_STATUS_VIRTUAL, "Soil is PERFECT");
    }
    else
    {
        Blynk.virtualWrite(SOIL_MOISTURE_STATUS_VIRTUAL, "Soil is too WET");
    }

```

Figure 28: Loop Section 01

```

    // Weather Status display section
    rainSensorState = getRainSensorState();
    Serial.print("Wether Status is = ");
    if (rainSensorState == 0)
    {
        Serial.println("Raining");
        Blynk.virtualWrite(WEATHER_STATUS_VIRTUAL, "Raining");
    }
    else
    {
        Serial.println("Sunny");
        Blynk.virtualWrite(WEATHER_STATUS_VIRTUAL, "Not Raining");
    }

    // Water Level Status display section
    waterLevel = getWaterLevel();
    Serial.print("Water level is = ");
    if (waterLevel == 0)
    {
        Serial.println(waterLevel);
    }
    else
    {
        Serial.println(waterLevel);
    }

    // logic Section of Motor ON or OFF

    Serial.print("Motor state ");
    Serial.println(motorState);
    if (rainSensorState != 0 && soilMoisture <= 20 && motorState == 1)
    {
        Blynk.virtualWrite(WATER_PUMP_VIRTUAL, 1);
        digitalWrite(waterPump, LOW);
        Serial.println("Motor is ON\n\n");
    }
    else
    {
        Blynk.virtualWrite(WATER_PUMP_VIRTUAL, 0);
        digitalWrite(waterPump, HIGH);
        Serial.println("Motor is OFF\n\n");
    }

    Blynk.run();
    delay(1000);
}

BLYNK_CONNECTED()
{
    // Request the latest state from the server
    Blynk.syncVirtual(WATER_PUMP_VIRTUAL);
}
BLYNK_WRITE(V3)
{
    motorState = param.asInt(); // Get the state of the button (0 or 1)
}

```

Figure 29: Loop Section 02