# IoT-Enabled Soil-Moisture-Based Plant Watering System with NodeMCU ESP8266

SUBJECT- IOT  TOPIC- SENSORS & ACTUATORS

Lanisha Thomas Nadar
Class SE COMPS B
Roll no. 10678
Department of Computer Engineering
Fr. Conceicao Rodrigues College of Engineering, Bandra
Mumbai-400042, India
lanishathomas7606@gmail.com

Noella Pereira
Class SE COMPS B
Roll no. 10690
Department of Computer Engineering
Fr. Conceicao Rodrigues College of Engineering, Bandra
Mumbai-400042, India
pereiracnoella@gmail.com

*Abstract*— **In the age of technology, automation has permeated our lives, and the demand for remote control systems continues to grow. This report introduces an IoT-powered Automatic Plant Watering System, driven by the NodeMCU ESP8266 microcontroller and a soil moisture sensor. Users can monitor soil moisture levels through a mobile application, remotely activating the watering process. This system addresses the challenges of manual plant care and the need for efficient solutions. The project aims to deliver cost-effective and accessible intelligent farming tools, promoting sustainability and water conservation aligned with the United Nations' Sustainable Development Goals 2 (SDG-2). By offering precise soil moisture control, customization, and remote capabilities, it empowers users to maintain healthy crops, fostering a recent trend in plant care.**

***Keywords— NodeMCU ESP8266, IoT, Soil Moisture Sensor, Plant Irrigation, Automation***

## I. INTRODUCTION

Plants need proper and timely watering to stay healthy. Manual watering often depends on personal judgment, which may not always be accurate. The growth of IoT and low-cost microcontrollers has made it easier to design systems that monitor soil conditions and automate irrigation. Automated watering helps ensure plants receive the right amount of water, reduces wastage, and removes the need for constant supervision. IoT connectivity also allows users to check plant conditions remotely, thus enabling the system to become smarter and more accessible. The use of IoT also allows data collection over time, which can help in making informed decisions for better crop management. Manual watering can lead to over-watering or under-watering because soil moisture is not continuously monitored, affecting plant growth and wasting water. The objective is to develop an IoT-based automatic plant watering system using NodeMCU ESP8266 to monitor soil moisture in real time, automatically activate a water pump when moisture levels drop below a set threshold, and provide status information through IoT connectivity, with minimal human involvement.

## II. LITERATURE REVIEW

An automatic plant-watering system is designed to monitor soil moisture and supply water only when needed, helping prevent both over-watering and under-watering **[1]**. These systems typically use IoT technology to collect real-time moisture data from sensors and send control signals to a water pump when the soil becomes dry. The integration of IoT enables remote monitoring and control through mobile applications, improving usability and reducing manual effort **[1]**. Microcontroller-based designs allow the system to respond intelligently to changes in soil conditions by automating water flow using basic electronic components such as moisture sensors and relays **[2]**. Such systems are inexpensive, energy-efficient, and suitable for household gardening, agriculture, and greenhouse environments **[2].**

## III. METHODOLOGY

The methodology describes the components used and the step-by-step process followed to design, implement, and operate the IoT-based automatic plant watering system. This system automatically waters plants based on soil moisture level, while also allowing users to monitor moisture and control the pump remotely via the internet (Blynk).

### *A. Components*

#### *1. Required Components*

**NodeMCU ESP8266** - a small, low-cost microcontroller board that has built-in Wi-Fi. It is used in IoT projects because it can connect to the Internet wirelessly and interact with apps or cloud services. A microcontroller is an electronic device that acts like a tiny computer in many devices; consisting of a CPU- that processes the instructions, 2 memory – Flash Memory- that stores the program & RAM- the temporary working memory and I/O pins- that connect to sensors, etc.

**Soil Moisture Sensor** - an electronic device used to measure how much moisture is present in the soil.

**Relay Module** - an electronic switching device that lets a low-power controller (here, NodeMCU) safely turn ON/OFF high-power devices (here, water pump)

**Breadboard** - a reusable platform used to build and test electronic circuits without soldering.

**Jumper Wires** - are small electrical wires used to connect components on a breadboard or between modules (here, sensors, microcontrollers).

1. **Male-Male Jumper Wires** - Wires with pins on both ends, used to connect components that have female connectors, such as breadboards or module headers.
2. **Male-Female Jumper Wires** - Wires with a pin on one end (male) and a socket on the other end (female), used to connect a male pin component

(here, NodeMCU) to a breadboard or another module.

2. *Other Components*

**Mini water pump**
**Mini water pipe**
**9v battery clip**
**9v Battery**

B. *Process*

1. *Construction*

Place the NodeMCU ESP8266 on the breadboard and connect the Vin and GND pins to the breadboard. Place the soil moisture sensor on the breadboard. Connect the following pins of the Soil Moisture sensor to the pins of the NodeMCU as follows:-
VCC -> Vin
GND -> GND
Analog (A0) pin -> A0
A relay has:

- Input side → connects to NodeMCU
- Output side → connects to high-voltage device

Connect the following pins of the Relay module to the pins of the NodeMCU as follows:-
VCC -> 3.3V
GND -> GND
IN -> D3
Furthermore, connect the water pump to the Relay model, and the relay model to the motor and battery as per the given circuit diagram below. Also, place the soil moisture sensor into the soil.

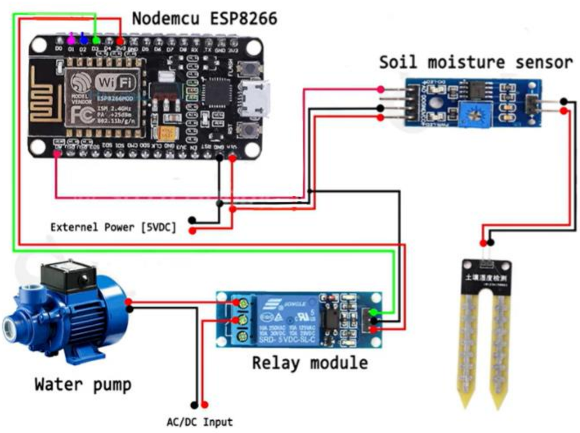


**Fig.1. Circuit Diagram**

The **ESP8266** is a Wi-Fi enabled microchip that contains a processor + Wi-Fi capability, thus having an advantage over Arduino. The **NodeMCU** is the name of an open-source firmware+development board that uses the ESP8266 chip, where 'Node' is inspired by Node.js, a JavaScript-style event programming and 'MCU' stands for Microcontroller Unit. It consists of a USB port, voltage regulator and GPIO pins. The **Soil moisture sensor** is a Resistive-type of sensor that consists of two metal probes. When placed inside the soil, wet soil conducts electricity better, and dry soil has more resistance. So the sensor measures the conductivity, converts it to a value and sends it to a microcontroller. Therefore,
More water = more conductivity → Higher reading
Less water = less conductivity → Lower reading
The **Relay module** acts as a middle device between the microcontroller and the output devices. Microcontrollers work at low voltage (3.3V–5V) and can only supply very small current. But devices like pumps and motors, require higher voltage and current (5V, 9V, 12V, 230V AC). Directly connecting them to a microcontroller would burn the board. The relay module consists of 2 LEDs, a red and a green indicator.
1) Power Indicator LED (red in colour)

- Shows that the relay module is powered
- ON → Module has power
- OFF → No power

This LED does NOT mean the relay is ON, just that the board is receiving power.
2) Relay Status (Signal) LED (green in colour)

- Shows whether the relay is activated
- ON → Relay is active → Output switched
- OFF → Relay is inactive

This LED tells you when your NodeMCU turns the relay ON. So, whenever the GPIO pin triggers the relay, the GREEN LED turns ON.

2. *Creation of the Blynk Web Dashboard*

Visit the Blynk website and create a new account using Gmail address. Sign in to that account and click the new Template button. Enter the project name and click the done button. Click the "Datastreams" tab and create two data streams for that. Use the information below:

- **Virtual PIN –> Name – Moisture value / PIN — V0 / MIN — 0 / MAX — 100**
- **Virtual PIN –> Name – Water pump / PIN — V1 / MIN — 0 / MAX — 1**

The, click the "Web dashboard" tab and include the one button and one Gauge widget to the dashboard. Arrange these widgets. Click the one-by-one settings buttons on these widgets and select the data streams created earlier. Click Save . Click the search icon button and create a "New device". For that, select the template created earlier.

The Blynk web dashboard has been created.

Connect this project to the computer, and upload the program for this project:

```
#define BLYNK_TEMPLATE_ID "TMPL3DD4wEhC7"

#include <ESP8266WiFi.h>

#include <BlynkSimpleEsp8266.h>
```

```
char auth[] = "DXreITrSyMFbMdPgTM07uawcaBbxO7Od";

char ssid[] = "motorola edge 50 pro_2224";

char password[] = "xxxxxxxx";

#define sensor A0

#define pump D3

void setup() {

 pinMode(pump, OUTPUT);

 Blynk.begin(auth, ssid, pass);

}

void loop() {

 int m = analogRead(sensor);

 m = map(m, 0, 1023, 100, 0);

 Blynk.virtualWrite(V0, m);

 digitalWrite(pump, (m <= 50) ? LOW : HIGH);

 Blynk.run();

}
```

Copy-paste the Blynk auth token present in the Blynk web dashboard. Enter **Your** WIFI SSID and password, select board and port and upload the above code to the Nodemcu board.

*To create the Blynk Mobile Dashboard:*

Download and install the Blynk app on your phone. Sign in to your account and click the template created in the Blynk web dashboard. Add the widget to the dashboard. Click the one-by-one widget and select the data streams created in the Blynk web dashboard.

The Blynk mobile dashboard has been created.

### 3. *Working*

The NodeMCU, relay, moisture sensor, and LCD are powered. Red LED on relay indicates that the module now has power. NodeMCU connects to Wi-Fi via programmed credentials. The Soil moisture sensor probes detect water content and converts the conductivity into an **analog voltage**. This analog signal is sent to NodeMCU A0 pin. NodeMCU reads moisture value (0–1023). NodeMCU compares moisture value with a preset threshold:

- Moisture < Threshold → Soil is dry → Need water
- Moisture ≥ Threshold → Soil is wet → No need to water

If soil is dry, NodeMCU sends HIGH signal to relay input pin:

- Relay triggers → Green LED turns ON
- Pump receives power → Water starts flowing

If soil is wet, NodeMCU sends LOW signal to relay input pin:

- Relay OFF → Pump stops

The pump draws water and supplies it to the plant. Once moisture level rises again, NodeMCU senses wetness and the relay stops pump. Thus, the system prevents overwatering. Now, the NodeMCU sends moisture readings to Blynk cloud. The user can view: Real-time moisture % and Pump status. The user can also manually control the pump if needed. Even if the user is faraway, the Blynk app gives full monitoring and control.

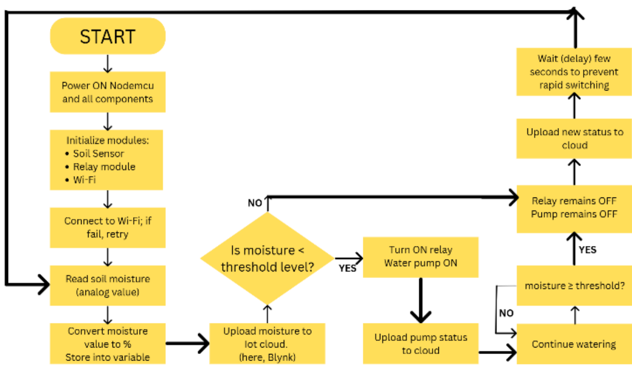


**Fig. 2. Flowchart 1**

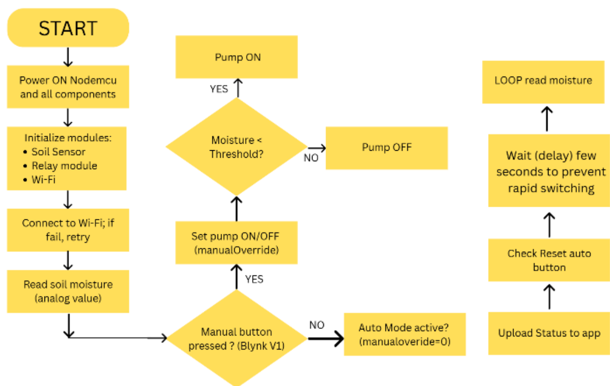


**Fig. 3. Flowchart 2-**
Explanation of the working with respect to the Virtual Buttons via. Blynk App

## IV. RESULTS

The system accurately sensed soil moisture and mapped readings to percentage values for meaningful interpretation. When moisture fell below the threshold, the controller activated the pump and halted irrigation once levels recovered, confirming effective closed-loop control. Manual override through Blynk worked reliably, allowing seamless switching between automatic and user-controlled modes. Overall, the system showed stable performance, low latency in response, and efficient water usage, making it suitable for small-scale smart irrigation.

## V. CONCLUSION

The IoT-enabled soil-moisture-based watering system using NodeMCU ESP8266 successfully monitored soil conditions, controlled irrigation automatically, and allowed reliable manual override via the Blynk app, resulting in efficient water usage and reduced human effort. To further improve performance, the system can be enhanced by expanding sensor coverage, incorporating long-term data logging and analytics, integrating weather prediction to optimize watering cycles, and employing solar energy for improved sustainability. Additional developments such as AI-based decision-making, alert notifications, and scalability for multi-field agricultural environments would further strengthen the reliability and applicability of this smart irrigation approach.

## VI. REFERENCES


[1] Razam, N. H. D. M., & Rais, S. S. (2023, December). IoT-enabled automatic plant watering system. In *2023 IEEE 8th International Conference on Recent Advances and Innovations in Engineering (ICRAIE)* (pp. 1-6). IEEE.

[2] Mayuree, M., Aishwarya, P., & Bagubali, A. (2019, March). Automatic plant watering system. In *2019 International Conference on Vision Towards Emerging Trends in Communication and Networking (ViTECoN)* (pp. 1-3). IEEE.

[3] https://srituhobby.com/how-to-make-a-plant-watering-system-with-the-nodemcu-esp8266-board-and-the-new-blynk-update/