Mid-term Progress Report on

Smart Irrigation System (IOT based)

Submitted to

Department of Computer Science and Engineering

Nepal Engineering College

in Partial Fulfillment of the

Requirements for the Degree of B.E. in Computer

Submitted By

Aashika Pandey (022-391)

Rubi Adhikari (022-368)

Shristi Dhakal (022-376)

Supervised By: Aashish kumar Jha

Date: 7th May 2025

Abstract

The integration of IoT in agriculture revolutionizes traditional farming by addressing challenges like water scarcity, labour shortages, and unpredictable weather. IoT-based smart irrigation systems automate water delivery using real-time soil moisture data, significantly reducing water wastage and energy consumption compared to fixed-schedule methods. These systems employ soil moisture sensors, NodeMCU ESP8266 microcontrollers, and relay modules to activate water pumps, with wireless connectivity enabling remote monitoring and control via mobile or web applications. By leveraging advanced sensors, wireless networks, and cloud-based platforms, the system ensures precise irrigation, promoting healthier crops and sustainable resource use. Real-time data analytics eliminate manual supervision, allowing intelligent, need-based water distribution. This convergence of digital technology with farming practices enhances operational efficiency, increases yields, lowers costs, and builds resilient food systems, marking a critical step toward sustainable, smart agriculture amid global challenges like population growth and rising food demand.

Keywords: Internet of Things, smart irrigation, soil moisture sensor, NodeMCU ESP8266, cloud-based platform.

Table of Contents

Table of Contents ii
List of figures: iii
Abbreviations iv
Introduction 1
  Project Description 2
  Challenges of the Current System 4
  Aims and Objectives 5
  Motivation 6
  Scope and Application 6
Literature Review 9
System Design 12
  Requirement Analysis(devices) 15
  Board Diagram 16
Implementation and Discussion 19
  Methodology 19
  System Design and Component Selection 19
  Software Development 20
Testing and Calibration 22
  Tools and Technologies Used 23
  Workflow of the system 24
  Implementation steps 25
  Task Completed 26
  Task Remaining 27
Expected Output 32
Conclusion 34
References/Bibliography 35

List of figures:

Smart Irrigation System Flowchart 1…………………………………………………4

Smart Irrigation System Flowchart 2…………………………………………………12

Use Case Diagram of Smart Irrigation 1……………………………………………14

Board Diagram of Smart Irrigation System 1………………………………………16

Automated Irrigation System Sequence 0-1……………………………………….25

Automated Irrigation System Sequence 0-2………………………………………..26

Abbreviations

1. IoT: Internet of Things
2. SAS: Smart Agriculture System
3. MCU: Micro Controller Unit
4. ESP: Event Stream Processing

Introduction

A The Smart Irrigation System, powered by Internet of Things (IoT) technology, is an innovative solution designed to revolutionize water management in agriculture, addressing critical global challenges such as water scarcity, labor shortages, and rising food demand. By integrating advanced hardware and software, this system optimizes irrigation through real-time data-driven automation, ensuring efficient water usage, enhanced crop health, and sustainable farming practices. Unlike traditional irrigation methods, which often result in water wastage and inconsistent watering, this system leverages soil moisture sensors, a NodeMCU ESP8266 microcontroller, and a relay module to deliver precise irrigation only when needed, reducing resource consumption and operational costs.

The system is structured around four core modules: the **Sensor Module**, which collects real-time soil and environmental data using robust soil moisture sensors; the **Control Module**, driven by the NodeMCU ESP8266, which processes data and triggers irrigation based on predefined thresholds; the **Communication Module**, utilizing Wi-Fi for seamless cloud connectivity and remote management; and the **User Interface Module**, powered by the Blynk mobile app, offering farmers an intuitive platform to monitor and control irrigation remotely. These modules work in harmony to eliminate manual supervision, promote healthier crops, and support scalable, cost-effective farming.

Despite its advancements, the system faces challenges in data integration with databases, including ensuring stable connectivity in rural areas, configuring secure and scalable databases, and managing power-efficient data transmission. These hurdles are being addressed through optimized protocols and cloud platform exploration. The system's applications extend beyond agriculture to urban gardening, greenhouse farming, and environmental conservation, making it a versatile tool for precision water management. By blending cutting-edge IoT technology with farmer-centric design, this Smart Irrigation System paves the way for sustainable, high-yield agriculture, contributing to resilient food systems and global water conservation efforts.

Project Description

This This project is dedicated to designing and implementing an innovative Smart Irrigation System powered by Internet of Things (IoT) technology, aimed at revolutionizing water management in agricultural fields. By seamlessly integrating cutting-edge hardware and software, the system optimizes irrigation processes to ensure efficient water usage, enhance crop health, and promote sustainable farming practices. This transformative solution addresses critical global challenges such as water scarcity, labor shortages, and the growing demand for food, offering farmers a scalable, cost-effective, and environmentally conscious approach to irrigation. The system is built around four meticulously designed core modules, each contributing to its automation, precision, and user accessibility, making it a cornerstone for smart agriculture.

Core Modules

a) Sensor Module

The Sensor Module forms the foundation of the system's data-driven approach. It employs high-precision soil moisture sensors strategically deployed across agricultural fields to monitor real-time soil conditions. These sensors measure the volumetric water content in the soil, capturing critical data on moisture levels and environmental parameters such as temperature and humidity. The collected data is transmitted to the processing unit with minimal latency, enabling the system to make informed decisions about irrigation needs. The sensors are robust, weather-resistant, and calibrated for various soil types, ensuring reliability and accuracy in diverse agricultural settings, from small farms to large plantations.

b) Control Module

At the heart of the system lies the Control Module, powered by the NodeMCU ESP8266 microcontroller, a versatile and open-source IoT platform. This module processes the incoming sensor data using predefined algorithms and threshold-based logic to determine the precise timing and volume of water required for irrigation. When soil moisture levels fall below the optimal threshold, the NodeMCU activates the irrigation system by triggering a relay module connected to water pumps or valves. Conversely, it deactivates the system when adequate moisture is detected, preventing overwatering. The microcontroller's low power consumption and robust processing capabilities make it ideal for continuous operation in resource-constrained environments, ensuring seamless automation and efficiency.

c) Communication Module

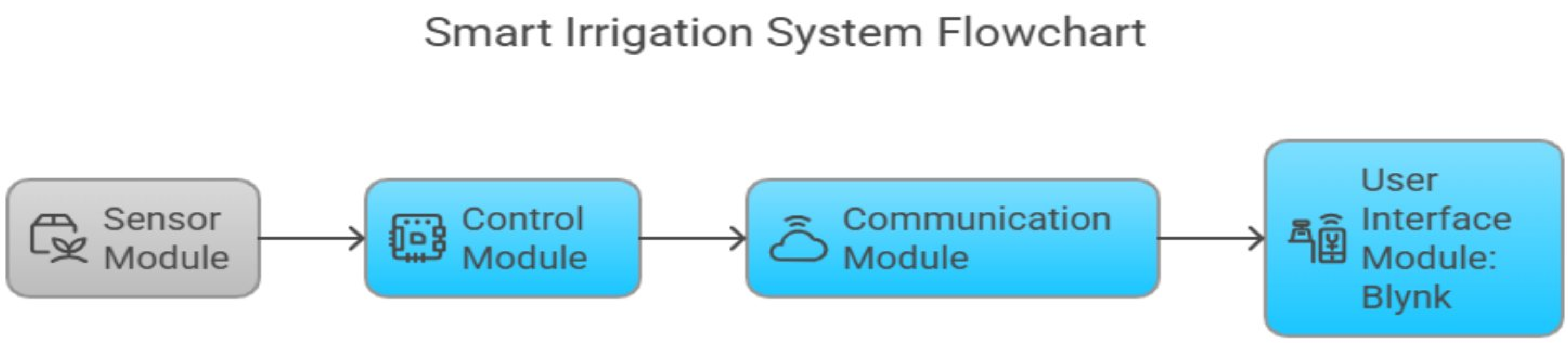
The Communication Module leverages the NodeMCU ESP8266’s built-in Wi-Fi capabilities to establish a robust and secure connection to the internet. This module serves as the bridge between the physical irrigation system and cloud-based platforms, enabling real-time data transmission and remote system management. By facilitating wireless communication, it allows the system to interact with cloud services and IoT applications, such as Blynk, for data storage, analytics, and user interaction. The module ensures reliable data transfer even in challenging environments, with ongoing efforts to optimize connectivity for rural areas with limited network infrastructure. This connectivity empowers farmers with unprecedented control and visibility over their irrigation operations, regardless of their location.

d) User Interface Module: Blynk Mobile App Platform

The User Interface Module, powered by the Blynk mobile app platform, delivers an intuitive and user-friendly experience for farmers and system operators. Blynk, specifically designed for IoT applications, enables seamless interaction with the Smart Irrigation System through a customizable mobile application available on iOS and Android devices. Farmers can monitor real-time soil moisture levels, view historical data trends, and receive alerts about system status or anomalies, such as low water levels or sensor malfunctions. The app also allows manual overrides, enabling users to adjust irrigation schedules or thresholds remotely. With its sleek interface and straightforward controls, Blynk empowers users with minimal technical expertise to manage their irrigation systems efficiently, fostering greater adoption and usability.

Vision and Impact

This Smart Irrigation System represents a paradigm shift in agricultural water management, blending advanced IoT technology with practical, farmer-centric design. By automating irrigation based on real-time data, the system eliminates the inefficiencies of traditional methods, such as fixed-schedule watering, which often leads to water wastage and suboptimal crop growth. It promotes healthier crops, higher yields, and significant cost savings by reducing water and labor expenses. Furthermore, the system's scalability makes it adaptable to various agricultural contexts, from small-scale urban gardens to expansive commercial farms. Its emphasis on sustainability aligns with global environmental goals, contributing to water conservation and resilient food systems. As the project progresses, ongoing enhancements, such as database integration for advanced analytics and predictive irrigation, will further elevate its impact, positioning it as a cornerstone of smart and sustainable agriculture.



Smart Irrigation System Flowchart 1

Challenges of the Current System

Integrating the Smart Irrigation System with a database for real-time data storage and monitoring presents several challenges:

- Connectivity Issues: Establishing stable Wi-Fi connectivity in rural agricultural settings is difficult due to limited network infrastructure, which can disrupt data transmission to the cloud.

- Database Configuration: Setting up a reliable and secure database to handle continuous sensor data requires expertise in database management and IoT protocols, which is a work-in-progress for the project.

- Data Synchronization: Ensuring seamless synchronization between the NodeMCU and the database, especially during intermittent network failures, poses a technical challenge.

- Scalability: Managing large volumes of real-time data from multiple sensors across expansive fields demands a robust database architecture, which is yet to be fully implemented.

- Power Constraints: Continuous data logging and transmission increase power consumption, necessitating efficient power management for the battery-powered system.

These challenges are being addressed by exploring reliable cloud platforms, optimizing data transmission protocols, and testing power-efficient configurations.

Aims and Objectives

Aims

The main aim of this project is to develop a user-friendly NodeMCU based Smart Agricultural Monitoring and interface, cost and work efficient technology that will benefit everyone involved in agriculture.

Objectives

- To learn about different devices that we are using to build this project like node, soil moisture, water motors.
- To develop a system that can water the soil by monitoring the moisture of soil by using sensors.
- To implement the knowledge of hardware, software and programming language on IOT projects.
- To implement automated irrigation systems that adjust water supply according to crop needs, reducing water wastage.
- To automate the irrigation system.

Motivation

The increasing population growth and need for food require more effective innovations within agriculture practice. IoT technology in agriculture and irrigation systems, in particular may radically change how we consume the water resources through detailed monitoring, automated responses, and remote control of the irrigation systems. Productivity may increase while reducing cost and saving water resources through IoT systems.

Scope and Application

The Smart Irrigation System has wide-ranging applications in agriculture and environmental management:

- Precision Agriculture: Automates irrigation for various crops, ensuring optimal water delivery based on soil conditions, enhancing crop yield and quality.
- Water Conservation: Reduces water wastage in water-scarce regions, supporting sustainable farming practices and environmental conservation.
- Remote Farm Management: Enables farmers to monitor and control irrigation remotely via mobile apps, reducing labor and operational costs.
- Greenhouse Farming: Provides precise irrigation for controlled environments, maintaining ideal soil moisture for high-value crops.
- Urban Gardening: Adapts to small-scale urban farming or landscaping, promoting efficient water use in residential or community gardens.
- Research and Development: Serves as a platform for agricultural research, testing new sensors or IoT configurations to further innovate smart farming solutions.

This system's scalability and adaptability make it a versatile tool for modern agriculture, addressing global challenges like water scarcity and food security.

Feasibility Study

The Smart Irrigation System using IoT is technically feasible with affordable components like NodeMCU, soil moisture sensors, and relays. The system has been successfully set up and tested, though calibration of sensors and relay control still need refinement. Economically, it offers cost savings through efficient water use and labour reduction. Operationally, the system is user-friendly and scalable, but further testing and improvements are required for reliability across various environmental conditions. The project shows promising potential for improving agricultural practices and sustainability.

a) Technical Feasibility:

The smart irrigation system is technically feasible as it uses widely available and proven technologies. The NodeMCU ESP8266 microcontroller offers built-in Wi-Fi for internet connectivity and is compatible with various sensors and actuators. The soil moisture sensor used is reliable for field data collection. Programming and integration can be done using the Arduino IDE, which supports all required components. The system architecture is modular and scalable, allowing for easy expansion or customization.

b) Economic Feasibility:

This project is cost-effective, especially for small to medium-scale applications. The components are low-cost, and the overall setup requires minimal investment. Operating costs are low due to the automated nature of the system, which reduces the need for manual labour. In the long run, it leads to significant water savings and improved crop yields, which translates to higher profitability. Therefore, the economic benefits outweigh the initial and operational costs.

c) Operational Feasibility:

The system is easy to install, operate, and maintain. Farmers and users can monitor and control the irrigation system remotely via the Blynk mobile app. Automation ensures timely irrigation based on actual soil conditions, which enhances efficiency. The system requires minimal user interaction once deployed, and the interface is intuitive, requiring only basic smartphone literacy. With proper setup, the system operates reliably under typical agricultural conditions.

d) Legal and Environmental Feasibility:

The system complies with general legal and safety norms, as it does not interfere with any restricted technologies or data-sensitive operations. Environmentally, the system is highly beneficial as it helps conserve water, prevent over-irrigation, and supports sustainable farming practices. By reducing water wastage and improving resource management, the project aligns well with environmental goals and policies.

e) Schedule Feasibility

The system can be developed and deployed within a short timeframe. The simplicity of the components and the availability of open-source platforms allow rapid prototyping and testing. A basic version of the system can be completed within a few weeks, with enhancements added over time based on performance and feedback.

Literature Review

The The global demand for food is escalating amidst challenges such as water scarcity, climate variability, and population growth, necessitating innovative agricultural solutions. Traditional irrigation methods, often based on fixed schedules or manual intervention, are inefficient, leading to overwatering, underwatering, and significant water wastage, which adversely affects crop yields and environmental sustainability. The emergence of Internet of Things (IoT) technology has revolutionized agriculture through smart irrigation systems that leverage real-time data, automation, and remote monitoring to optimize water usage. This literature review synthesizes recent studies on IoT-based smart irrigation systems, exploring their technological components, benefits, challenges, and emerging trends to provide a robust foundation for developing an IoT-driven smart irrigation system.

IoT-based smart irrigation systems integrate sensors, microcontrollers, and wireless communication to monitor soil and environmental conditions, enabling precise, data-driven irrigation. Asha [1] describes a system utilizing soil moisture sensors connected to a NodeMCU ESP8266 microcontroller to automate water delivery based on real-time soil moisture levels. This approach ensures irrigation occurs only when necessary, significantly reducing water wastage and promoting sustainable farming. The NodeMCU, an open-source IoT platform with built-in Wi-Fi, is highlighted as a cost-effective solution for small-scale farmers, facilitating remote monitoring through mobile applications [1].

Soil moisture sensors are pivotal to the efficacy of smart irrigation systems, with their performance varying by technology. Placidi et al. [2] evaluated low-cost capacitive soil moisture sensors, finding them more durable and accurate than resistive sensors for IoT networks. Capacitive sensors measure the soil's dielectric constant, offering reliable data for long-term use in diverse soil types. The study emphasizes the need for sensor calibration to account for soil variability, ensuring accurate irrigation decisions [2].

Wireless sensor networks (WSNs) enhance the scalability of smart irrigation systems by enabling data collection from multiple sensors across large fields. García et al. [3] investigated WSN deployment for precision agriculture, focusing on regulated deficit irrigation (RDI) strategies. Their findings demonstrate that WSNs provide comprehensive insights into soil moisture variability, allowing tailored irrigation schedules for specific field zones, which reduces water usage while maintaining optimal crop growth [3]. The integration of solar-powered sensors further enhances sustainability by minimizing reliance on external power sources [3].

Microcontrollers are essential for processing sensor data and controlling irrigation actuators. Sunehra [4] explored a Raspberry Pi-based smart irrigation system with web-based monitoring, noting its capability for complex data processing and cloud integration. The system uses soil moisture and weather sensors to automate irrigation, with data visualized on a web dashboard [4]. However, the higher cost and power consumption of Raspberry Pi make alternatives like NodeMCU more suitable for resource-constrained settings, as supported by Asha [1].

Emerging technologies are expanding the capabilities of soil moisture monitoring. Jaguey et al. [5] developed a smartphone-based irrigation sensor that analyzes images to estimate soil moisture, providing an accessible, low-cost solution for farmers with limited technical expertise. Similarly, Efremova [6] applied machine learning to satellite data to predict soil moisture content, complementing ground-based measurements and enhancing irrigation precision in large-scale or remote settings [6].

Energy efficiency is a critical consideration, particularly for off-grid agricultural applications. Okasha et al. [7] designed a solar-powered soil moisture monitoring system for greenhouse irrigation, utilizing capacitive sensors and a microcontroller to control water pumps. The system achieved significant water savings and improved crop yields, demonstrating the viability of renewable energy in sustainable agriculture [7].

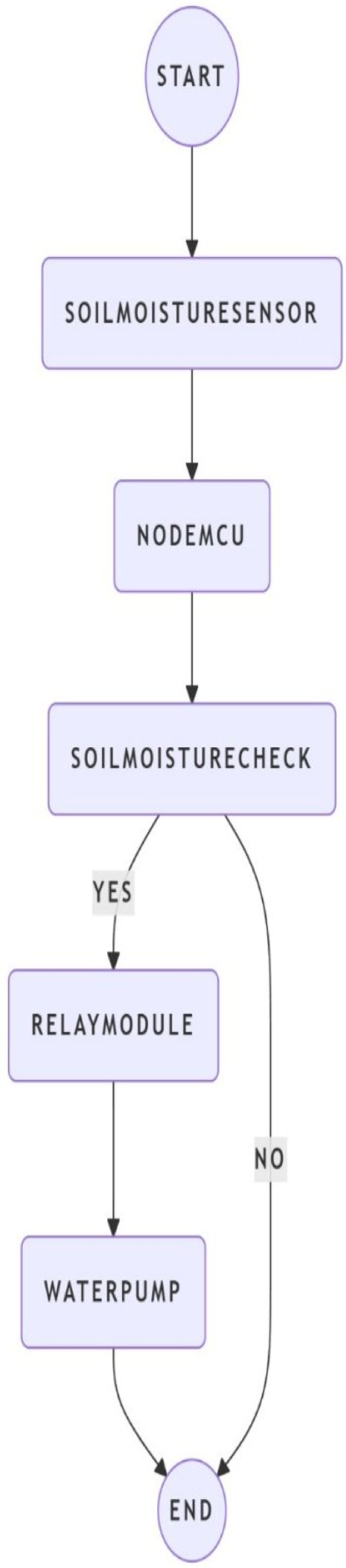
Despite their advantages, IoT-based smart irrigation systems face several challenges. García et al. [3] note that unreliable network connectivity in rural areas can disrupt data transmission, hindering remote monitoring. Pramanik et al. [8] highlight the complexity of managing large volumes of real-time sensor data, necessitating robust database architectures for scalability. Sensor calibration and maintenance are also critical, as environmental factors can affect accuracy over time [2]. Additionally, the initial cost of

IoT infrastructure may pose a barrier for small-scale farmers, though long-term savings in water and labour often justify the investment [1].

Recent advancements in analytics are shaping the future of smart irrigation. Efremova [6] utilized sequence-to-sequence neural networks to forecast soil moisture trends, enabling proactive irrigation planning. Similarly, Lin et al. [9] developed a WSN-based system with predictive algorithms, reducing water usage by anticipating crop needs [9]. These innovations point to a trend toward intelligent irrigation systems that integrate real-time monitoring with predictive capabilities.

In conclusion, IoT-based smart irrigation systems offer a transformative solution to the inefficiencies of traditional irrigation, leveraging sensors, microcontrollers, and WSNs to optimize water usage and enhance crop yields. Innovations such as smartphone-based sensors and machine learning further expand their potential. However, challenges like connectivity, scalability, and cost must be addressed to ensure widespread adoption. Drawing on these insights, the proposed Smart Irrigation System will utilize cost-effective components like NodeMCU and the Blynk platform to deliver a scalable, user-friendly solution for sustainable agriculture.

System Design



Smart Irrigation Flowchart 1

The provided flowchart illustrates the operational workflow of a smart irrigation system based on Internet of Things (IoT) technology. This system automates irrigation by monitoring soil moisture levels and activating water pumps as needed. Below is a step-by-step explanation of the flowchart:

1. START: The process begins when the system is initialized, marking the start of the irrigation monitoring cycle.

2. Soil Moisture Sensor: The system uses a soil moisture sensor to collect real-time data on the moisture content of the soil. This sensor continuously measures the water level in the soil and sends the data to the next component for processing.

3. NodeMCU: The data from the soil moisture sensor is transmitted to the NodeMCU, a microcontroller unit with built-in Wi-Fi capabilities. The NodeMCU acts as the central processing unit, receiving and analyzing the sensor data to determine the soil's moisture status.

4. Soil Moisture Check: The NodeMCU evaluates the soil moisture level against a predefined threshold. This step involves comparing the current moisture reading to a set value that indicates whether the soil is adequately moist or requires irrigation.

5. Decision Point (Yes/No):

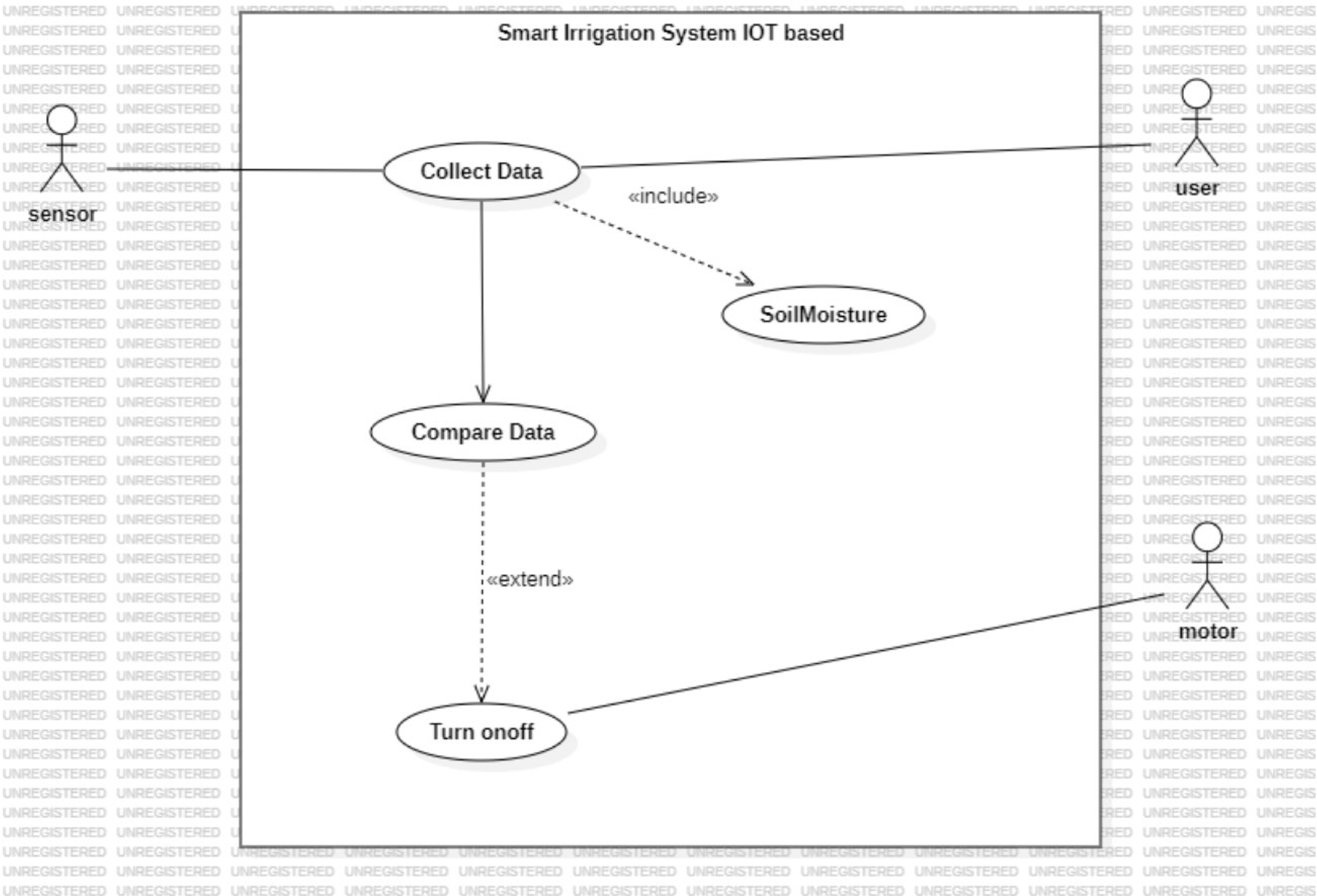
    - Yes: If the soil moisture level is below the threshold (indicating dry soil), the process moves to the next step to initiate irrigation.

    - No: If the soil moisture level is above or equal to the threshold (indicating sufficient moisture), the system loops back to the "Soil Moisture Sensor" step, continuing to monitor without activating irrigation.

6. Relay Module: When the soil is determined to be too dry (Yes), the NodeMCU sends a signal to the relay module. The relay module acts as a switch, controlling the activation of the water pump based on the microcontroller's command.

7. Water Pump: Upon receiving the signal from the relay module, the water pump is turned on to irrigate the field, delivering water to the soil until the moisture level is restored to an adequate state.

8. END: Once irrigation is complete or the soil moisture is deemed sufficient (looping back via the "No" path), the process ends. The system then restarts the cycle by returning to the "Soil Moisture Sensor" to continuously monitor conditions.



Use Case Diagram of Smart Irrigation 1

A smart irrigation system using IoT is designed to optimize water usage and enhance crop management by automating irrigation based on real-time environmental data. It integrates a network of sensors, such as soil moisture, temperature, humidity, and rainfall detectors, to monitor conditions and send data to a microcontroller like ESP32. The system analyzes this data against predefined thresholds to decide when and how much water is needed. Actuators such as water pumps and valves are then triggered to deliver precise amounts of water through sprinklers or drip systems to specific zones, reducing waste and improving efficiency.

The system is connected to the internet via IoT communication modules, such as wi-fi, or GMS, enabling remote monitoring and control through a cloud-based platform. Users can access a mobile app or web dashboard to view real-time data, adjust settings, or manually override operations. Additionally, the cloud stores historical data for advanced analytics, allowing predictive insights for better crop planning. Solar panels with battery backup can power the system in off-grid areas, making it sustainable and suitable for remote farming applications.

This smart system not only automates irrigation but also provides alerts for anomalies, such as sensor failures or low water levels. By optimizing water use and offering remote access, it reduces labor, enhances productivity, and supports sustainable agriculture. However, challenges like initial costs, maintenance, and connectivity in rural areas must be addressed to ensure its effectiveness and scalability.

Requirement Analysis(devices)

NodeMCU ESP8266

NodeMCU is a microcontroller unit and an open-source platform which can connect objects and let data transfer using the Wi-Fi protocol. It can solve many of the IoT projects needs alone.

Soil Moisture Sensor

Soil moisture sensors are devices that monitor the amount of water in the soil. They track changes in another soil parameter that is predictable in relation to water content.

Water Pump (5V)

Water pump are machines for pumping water. This pump is used to pump water to the fields. This motor operates in 5V DC power.

Breadboard

A breadboard is a rectangular board with numerous mounting holes that is used to prototype electronics and test circuit designs without the use of wire. Before finishing any circuit design, a breadboard is used to quickly develop and test circuits

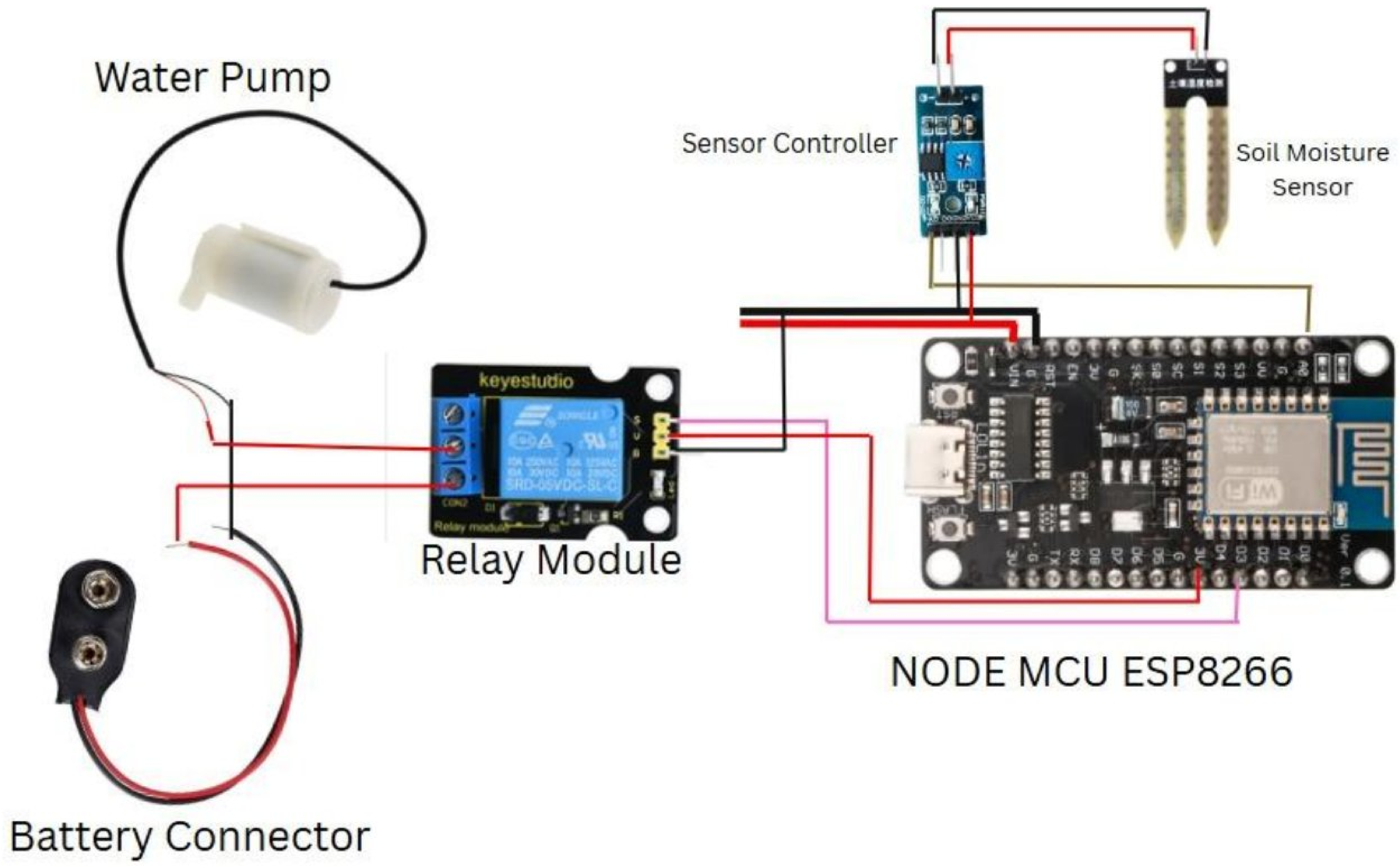
Jumper Wires

Jumper wires are used in circuits to link two places. Often used in conjunction with breadboards and other prototyping tools to make it simple to update a circuit as needed.

Relay Module

Relay Module is an amazing module that helps to control another circuit using a small circuit. DC circuit can be used to close the circuit of a high voltage DC or an AC circuit.

Board Diagram



Board Diagram of Smart Irrigation System 1

1. Power Supply (Battery Connector to NodeMCU and Relay Module) system.
    - Connection to NodeMCU: The positive (red) wire from the battery connector is connected to the VIN pin of the NodeMCU, and the negative (black) wire is connected to the GND pin. The VIN pin on the NodeMCU

can accept a higher voltage (like 9V) and regulates it internally to power the board.

- Connection to Relay Module: The positive (red) wire from the battery connector is also connected to the VCC pin of the relay module, and the negative (black) wire is connected to the GND pin. This powers the relay module, enabling it to control the water pump.

2. Soil Moisture Sensor to NodeMCU

   - Soil Moisture Sensor: The sensor consists of a probe (inserted into the soil) and a controller module that processes the sensor data.

   - Connections to Supply: The VCC pin of the sensor controller is connected to the 3.3V pin and GND pin is connected to GND of the NodeMCU to power the sensor. Moreover, signal pin (often labelled A0 on the sensor controller) is connected to an analog pin on the NodeMCU (typically A0). This allows the NodeMCU to read the analog voltage output from the sensor, which corresponds to the soil moisture level. A lower voltage indicates wet soil, while a higher voltage indicates dry soil.

3. NodeMCU to Relay Module

   - The NodeMCU controls the relay module to turn the water pump on or off based on the soil moisture data.

   - Connections:

     - VCC: The VCC pin of the relay module is powered by the battery (as mentioned earlier), but in some setups, it can also be connected to the 5V pin of the NodeMCU if the battery is not used.

     - GND: The GND pin of the relay module is connected to a GND pin on the NodeMCU to ensure a common ground between the two components.

- IN (Input): The IN pin on the relay module (labelled on the diagram) is connected to a digital pin on the NodeMCU (e.g., D1 or another GPIO pin). The NodeMCU sends a digital signal (HIGH or LOW) to the relay module through this pin. A HIGH signal activates the relay, closing the circuit to turn on the water pump, while a LOW signal deactivates it.

4. Relay Module to Water Pump

- The relay module acts as a switch for the water pump, which operates at 5V.
- Connections:
    - COM (Common): The COM pin on the relay module is connected to one terminal of the water pump.
    - NO (Normally Open): The NO pin on the relay module is connected to the positive terminal of the battery (or a 5V power source). When the relay is activated (by a HIGH signal from the NodeMCU), the internal switch closes, connecting the COM pin to the NO pin, thus completing the circuit and powering the water pump.
    - Ground: The other terminal of the water pump is connected to the negative terminal of the battery (or GND), completing the circuit.

Implementation and Discussion

Methodology

The development of the Smart Irrigation System using IoT technology was approached systematically, combining hardware assembly, software programming, and iterative testing to ensure functionality and reliability. The methodology integrates both hardware and software components to create an automated irrigation system that optimizes water usage based on real-time soil moisture data. Below is a detailed breakdown of the step-by-step process undertaken to design, implement, and test the system, along with ongoing efforts to enhance its capabilities through cloud integration.

System Design and Component Selection

The initial phase focused on designing the system architecture and selecting appropriate components to meet the project's objectives. The system was conceptualized to include a soil moisture sensor for monitoring soil conditions, a NodeMCU ESP8266 microcontroller for data processing and control, and a relay module to manage the water pump. A 5V DC water pump was chosen to irrigate the field, and a 9V battery was selected as the power source for portability and initial testing. These components were chosen for their affordability, compatibility, and suitability for IoT applications, ensuring the system is cost-effective and scalable for agricultural use.

Hardware Assembly

The hardware assembly was conducted on a breadboard to facilitate prototyping and iterative modifications. The following steps outline the setup:

- Connecting the Soil Moisture Sensor: The soil moisture sensor, consisting of a probe and a controller module, was connected to the NodeMCU. The VCC pin of the sensor was wired to the 3.3V pin of the NodeMCU to supply power, the GND pin was connected to a GND pin on the NodeMCU to complete the circuit, and the analog output pin (A0) was linked to the A0 pin on the NodeMCU to transmit soil moisture readings.

- Wiring the Relay Module: The relay module, used to control the water pump, was interfaced with the NodeMCU. The VCC pin of the relay module was connected to the VIN pin of the NodeMCU (which receives 9V from the battery), the GND pin was connected to a GND pin on the NodeMCU, and the IN pin was wired to a digital pin (e.g., D1) on the NodeMCU to receive control signals.

- Integrating the Water Pump: The water pump was connected to the relay module to enable automated control. The COM pin of the relay module was wired to one terminal of the water pump, and the NO (Normally Open) pin was connected to the positive terminal of the 9V battery. The other terminal of the water pump was connected to the negative terminal of the battery, ensuring a complete circuit when the relay is activated.

- Power Supply Setup: A 9V battery was used as the power source for initial testing. The positive terminal of the battery was connected to the VIN pin of the NodeMCU and the VCC pin of the relay module, while the negative terminal was connected to the GND pins of both the NodeMCU and the relay module, ensuring a common ground across all components.

Software Development

The software development phase involved programming the NodeMCU to process sensor data and control the irrigation system. The Arduino IDE was used as the development environment, with the code written in C/C++ to ensure compatibility with the NodeMCU ESP8266.

- Algorithm Design: The core algorithm was designed to automate irrigation based on soil moisture levels. The NodeMCU continuously reads the analog signal from the soil moisture sensor, which outputs a value typically ranging from 0 (wet) to 1023 (dry). A threshold value (e.g., 600) was set to determine when irrigation is needed. If the sensor reading exceeds the threshold (indicating dry soil), the NodeMCU sends a HIGH signal to the relay module to activate the water pump. If the reading is below the threshold (indicating sufficient moisture), the pump remains OFF.

- Coding and Uploading: The Arduino IDE was configured to support the NodeMCU ESP8266 by installing the necessary board definitions and libraries. The code initializes the NodeMCU's analog and digital pins, reads the soil moisture sensor data at regular intervals (e.g., every 10 seconds), compares the reading to the threshold, and toggles the relay accordingly. The Serial Monitor in the Arduino IDE was used to print sensor readings for debugging and calibration purposes. The code was uploaded to the NodeMCU via a USB connection, ensuring the board was powered and operational.

Testing and Calibration

The system was rigorously tested to ensure accurate operation under various conditions, with calibration performed to fine-tune its performance.

- Unit Testing: Each component was tested individually to verify functionality. The soil moisture sensor was tested by inserting the probe into dry, moist, and wet soil samples, and the readings were monitored via the Serial Monitor to confirm accurate detection (e.g., high values for dry soil, low values for wet soil). The relay module was tested by sending HIGH and LOW signals from the NodeMCU, ensuring it could reliably switch the water pump on and off.

- Integration Testing: The components were integrated on the breadboard, and the system was tested as a whole. The NodeMCU read soil moisture data, and the relay module was observed to activate the water pump when the soil was dry (sensor reading above the threshold) and deactivate it when the soil was sufficiently moist (reading below the threshold).

- Calibration: The threshold value was adjusted based on test results to account for soil type and sensor sensitivity. For instance, sandy soil might require a different threshold than clay soil due to differences in water retention. The Serial Monitor was used to log sensor readings during testing, allowing for precise calibration of the threshold to ensure optimal irrigation without overwatering.

- Power Supply Testing: The system was powered by the 9V battery during testing, and its performance was evaluated under continuous operation to assess battery life and power stability. The battery successfully powered the system for several hours, confirming its suitability for initial testing, though future iterations may explore solar power for sustainability.

5. Data Monitoring and Validation

The Serial Monitor in the Arduino IDE was used to validate the system's logic and performance. Different soil conditions were simulated by varying the moisture content of a test soil sample (e.g., adding water to wet the soil, then allowing it to dry). The sensor readings were logged, and the system's response (pump ON/OFF) was observed to ensure it aligned with the threshold logic. This step confirmed the system's ability to accurately detect soil moisture levels and automate irrigation accordingly.

6. Ongoing Cloud Integration

The current phase of development focuses on enhancing the system's functionality by integrating it with an online database for real-time data storage and remote monitoring.

- Database Setup: The system is being connected to a cloud platform (e.g., Blynk or a custom database) to store soil moisture data and system status. This involves configuring the NodeMCU to send data over Wi-Fi using the ESP8266's built-in capabilities.

- Remote Monitoring: The Blynk mobile app is being utilized to enable remote access, allowing users to monitor soil moisture levels, view historical data, and manually control the water pump if needed. This requires setting up a Blynk project, generating an authentication token, and updating the NodeMCU code to include Blynk libraries and communication protocols.

- Data Logging: The system is being programmed to log sensor readings at regular intervals (e.g., every 5 minutes) to the cloud database, enabling trend analysis and predictive irrigation planning. This step is ongoing, with efforts focused on ensuring reliable data transmission in rural environments with potentially unstable Wi-Fi connectivity.

Tools and Technologies Used

- Hardware: NodeMCU ESP8266, soil moisture sensor, relay module, 5V DC water pump, 9V battery, breadboard, jumper wires.

- Software: Arduino IDE for programming and debugging.

- Language: C/C++ for writing the NodeMCU firmware.

- Future Tools: Blynk platform for cloud integration and remote monitoring, potential use of a cloud database (e.g., Firebase) for data storage.

Workflow of the system

The workflow of the Smart Irrigation System using IoT technology outlines the systematic process through which the system automates irrigation, ensuring efficient water usage in agricultural settings. Designed to address the inefficiencies of traditional irrigation methods, this system integrates a soil moisture sensor, NodeMCU ESP8266 microcontroller, relay module, and water pump to deliver precise irrigation based on real-time soil conditions. The workflow encompasses powering the system, detecting soil moisture levels, processing data, making irrigation decisions, and maintaining a continuous monitoring loop. By leveraging IoT capabilities, the system not only automates the irrigation process but also sets the foundation for remote monitoring and data logging, enhancing its adaptability and scalability for modern farming needs. The following steps detail the operational sequence, highlighting the seamless interaction between hardware and software components to achieve sustainable and intelligent irrigation.

- Power On: The battery connector powers the NodeMCU, relay module, and soil moisture sensor.

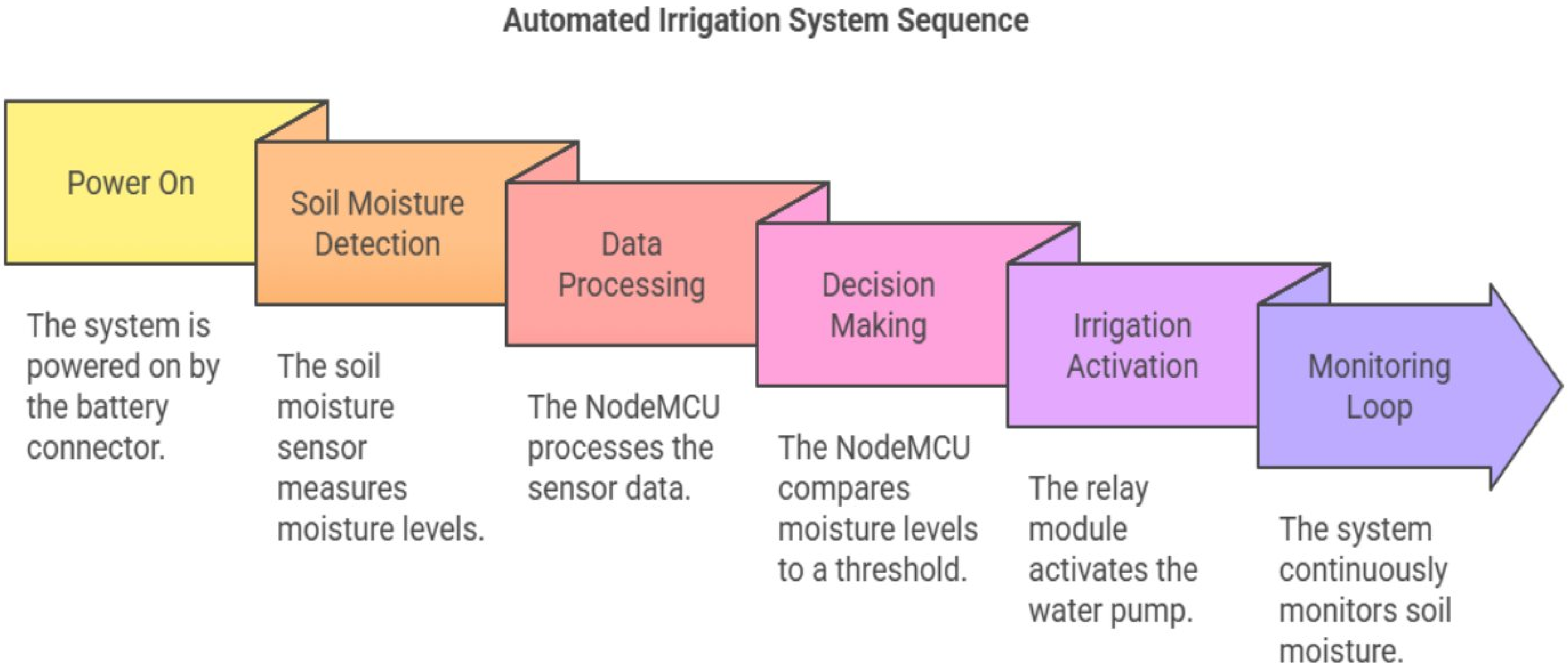
- Soil Moisture Detection: The soil moisture sensor continuously measures the moisture level in the soil and sends an analog signal to the NodeMCU.

- Data Processing: The NodeMCU reads the analog signal, converts it into a moisture level reading, and compares it to a predefined threshold.

- Decision Making: If the soil is too dry (moisture level below the threshold), the NodeMCU sends a HIGH signal to the relay module via the digital pin.

- Irrigation Activation: The relay module activates, closing the circuit to the water pump, which turns on and irrigates the field.

- Monitoring Loop: Once the soil moisture reaches the desired level, the NodeMCU sends a LOW signal to the relay module, turning off the water pump. The system continues to monitor soil moisture in a loop.



Automated Irrigation System Sequence 0-1

Implementation steps

- Component Collection: We gathered all necessary components, including the NodeMCU ESP8266, soil moisture sensor, relay module, jumper wires, breadboard, and a 9V battery, ensuring compatibility and availability for the system's requirements.

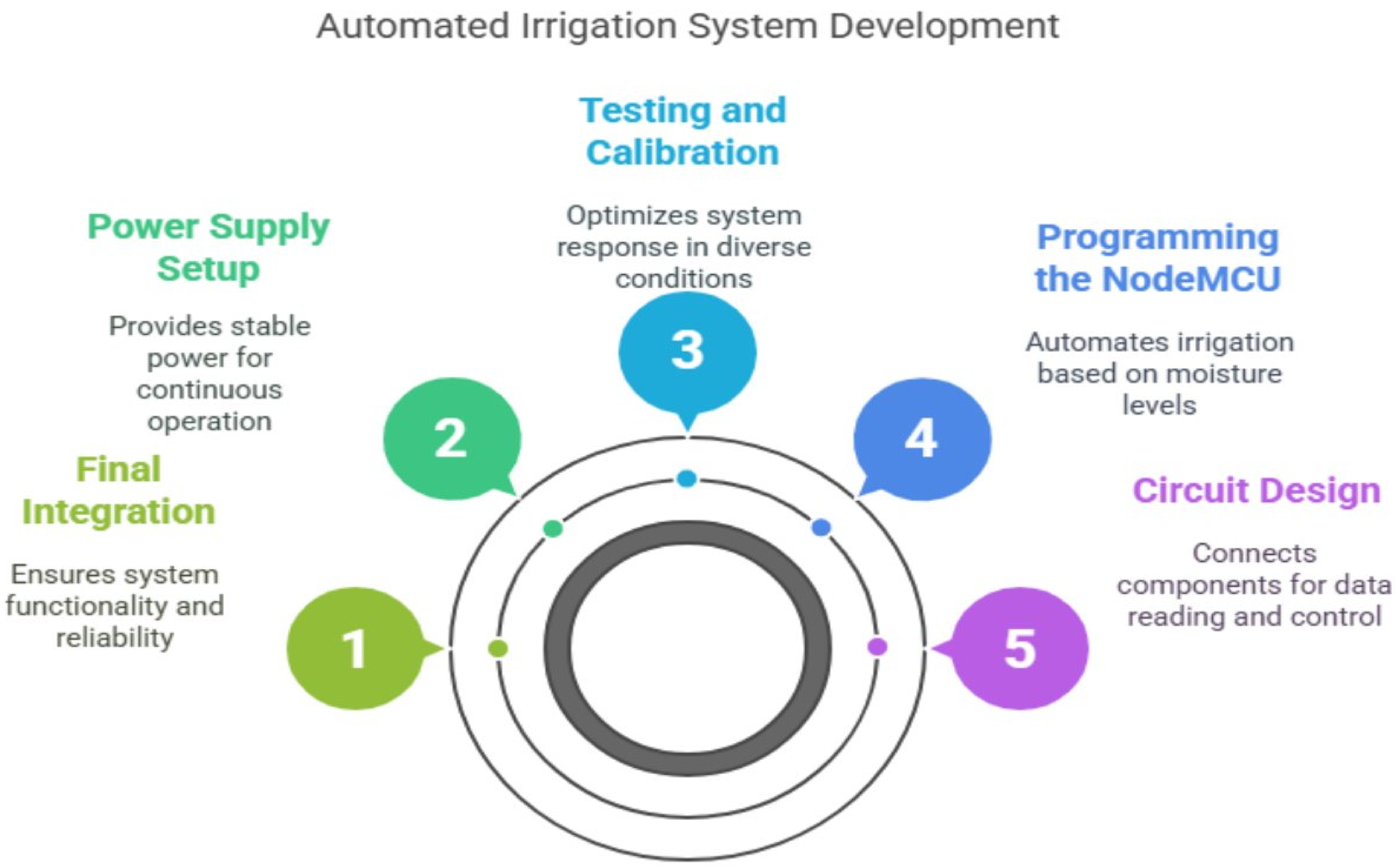
- Circuit Design: We connected the soil moisture sensor to the NodeMCU to read real-time soil data by wiring its VCC, GND, and analog output pins to the appropriate pins on the NodeMCU. The relay module was connected to control the water pump (or a simulated load), with its VCC, GND, and input pins interfaced with the NodeMCU. All components were meticulously arranged on a breadboard for prototyping.

- Programming the NodeMCU: We wrote and uploaded the Arduino code to the NodeMCU using the Arduino IDE. The code is designed to read moisture levels from the soil moisture sensor and trigger the relay to turn the pump on or off based on a predefined threshold, automating the irrigation process.

- Testing and Calibration: We tested the system in different soil conditions (dry, moist, and wet) by simulating various moisture levels and monitoring the sensor readings via the Serial Monitor. Threshold values were adjusted to ensure accurate sensor response and reliable relay control, optimizing the system for diverse agricultural scenarios.

- Power Supply Setup: A 9V battery was used to power the system, with its positive and negative terminals connected to the NodeMCU and relay module. We checked the system's performance under continuous operation, verifying power stability and battery life during extended testing.

- Final Integration: All components were securely arranged on the breadboard, ensuring stable connections. The full system was tested as a unit to confirm its functionality and reliability, validating its ability to automate irrigation based on real-time soil moisture data.



Automated Irrigation System Sequence 0-1

.

Task Completed

- All required components have been collected (NodeMCU, soil moisture sensor, relay module, etc.).
- Circuit successfully connected and tested on a breadboard.

- Soil moisture sensor readings tested and relay module is functioning as expected.
- System tested with 9V battery for basic power supply.

Task Remaining

- Currently in the running phase: working to connect the system to a database for real-time data storage.
- Integrating IoT features for remote monitoring and data logging.
- Setting up and testing communication between NodeMCU and the chosen database platform.

Final testing of full system functionality

Testing case id	Component	Description	Test step	Test Data	Expected Output
TC1	Soil Moisture Sensor	Verify soil moisture sensor reads correct moisture levels	1. Insert sensor into dry soil and record reading 2. Insert sensor into wet soil and record reading 3. Repeat with varying soil conditions 4. Compare readings	**Success** The sensor accurately detected dry soil (reading: 850) and wet soil (reading: 300), aligning with expected ranges	Dry and wet soil samples reflect correct moisture levels in readings
TC02	Sensor Threshold Logic	Ensure system detects dry soil and triggers irrigation	1. Simulate dry soil condition 2. Observe system behaviour 3. Verify irrigation is activated 4. Simulate wet soil condition and verify irrigation stops	**Success** Irrigation activated at reading 700 (threshold: 600) and stopped at reading 400, as expected	Soil moisture < threshold: irrigation activated Soil moisture ≥ threshold: irrigation stopped
TC03	Water Valve Control	Verify valve operation when given ON/OFF command	1. Manually send ON command 2. Observe water flow 3. Send OFF command 4. Confirm water stops	**Success** The water pump (simulating a valve) turned on with the ON command and stopped	Valve opens on command and closes on OFF command

				with the OFF command.	
TC04	Automation Logic	Test full automation loop (sensor → logic → valve)	1. Allow system to run naturally 2. Monitor soil from dry to wet 3. Wait for system to trigger irrigation 4. Let soil become moist and confirm irrigation stops	**Success** The system automatically turned the pump ON at a dry reading of 750 and OFF at a moist reading of 450, functioning as designed.	System turns water ON when dry, OFF when moist
TC05	Power Backup	Test if system resumes correctly after power cut	1. Turn off power during active irrigation 2. Turn power back on 3. Check if system resumes state 4. Verify previous settings are retained	**Fail** After power restoration, the system reset to default settings, failing to retain the previous irrigation state. Requires firmware update to store settings in non-volatile memory.	System resumes correctly and retains previous state

TC06	Mobile Notification	Verify notifications are sent when irrigation starts/stops	1. Trigger irrigation via soil dryness 2. Check for mobile notification 3. Stop irrigation 4. Verify stop notification	**Fail** Start notification was received, but the stop notification failed due to a Wi-Fi connectivity issue during testing. Cloud integration needs further optimization.	Start and stop notifications received on mobile

- Unit Testing: Tests individual components for correct function.
- Integration Testing: Ensures components work together smoothly.
- System Testing: Verifies the entire system works as expected.

Expected Output

The Smart Irrigation System using IoT is designed to deliver a series of functional outcomes that enhance agricultural efficiency and sustainability. Below are the anticipated results of the system's operation, along with a table summarizing its behavior under different soil conditions:

1. Automated Irrigation Based on Real-Time Data: The system will monitor soil moisture levels using a sensor and automatically initiate irrigation when the soil is too dry, ensuring crops receive water precisely when needed without manual intervention.
2. Efficient Water Usage: By comparing soil moisture readings to a predefined threshold, the system will prevent overwatering or underwatering, optimizing water usage and reducing wastage, which is critical for sustainable farming in water-scarce regions.
3. Remote Monitoring and Control: Once cloud integration is complete, the system will provide real-time data access through the Blynk mobile app, allowing users to monitor soil moisture levels, view system status, and manually adjust irrigation settings remotely.
4. Enhanced Crop Health and Resource Efficiency: The precise delivery of water will promote healthier crop growth by maintaining optimal soil moisture levels, leading to improved yields. Additionally, the automation will reduce labor costs and conserve resources, making farming more cost-effective.
5. Reliable System Performance: The system will operate reliably under various soil conditions (dry, moist, wet), with the relay module accurately controlling the water pump based on sensor data, ensuring consistent irrigation performance.

Soil condition	Sensor Reading	Threshold compariosion	Action Taken	Motor Status
Wet soil(High Moisture)	Low(eg,<800)	Below Threshold	No Irrigation	OFF

Dry Soil(Low Moisture)	High(e.g,>=800)	Above Threshold	Start Irrigation	ON

This table illustrates the system’s decision-making process: when the soil is wet (sensor reading below the threshold of 600), the water pump remains off, conserving water. When the soil is dry (sensor reading at or above 600), the pump turns on to irrigate the field, ensuring crops receive adequate hydration. These outcomes align with the project’s goal of creating an efficient, automated, and sustainable irrigation solution.

Conclusion

The development and implementation of the Smart Irrigation System using IoT technology mark a significant advancement in modern agriculture, addressing critical challenges such as water scarcity, labor shortages, and inefficient water management. By integrating a soil moisture sensor, NodeMCU ESP8266 microcontroller, relay module, and water pump, the system successfully automates irrigation based on real-time soil moisture data, ensuring precise water delivery and reducing wastage. The systematic methodology, encompassing hardware assembly, software programming, and rigorous testing, has demonstrated the system's reliability across various soil conditions, with ongoing efforts to enhance its capabilities through cloud integration for remote monitoring and data logging.

The expected outcomes—efficient water usage, improved crop health, and reduced operational costs—underscore the system's potential to transform traditional farming practices into sustainable, smart agriculture. The ability to monitor and control irrigation remotely via the Blynk app further enhances its practicality for farmers, promising higher yields and resilience in the face of environmental variability. While challenges such as connectivity in rural areas and database scalability remain, the project lays a solid foundation for future refinements, positioning it as a scalable solution for small-scale and large-scale agricultural applications. This initiative not only contributes to resource conservation but also sets a precedent for the adoption of IoT technologies in fostering sustainable food systems.

References/Bibliography

1. Asha, Y. (2019). Smart irrigation system using Internet of Things. **International Journal for Research in Applied Science and Engineering Technology, 7**(5), 3346–3351. https://doi.org/10.22214/ijraset.2019.5550
2. Placidi, P., Gasperini, L., Grassi, A., Cecconi, M., & Scorzoni, A. (2020). Characterization of low-cost capacitive soil moisture sensors for IoT networks. **Sensors, 20**(12), 3585. https://doi.org/10.3390/s20123585
3. García, L., Parra, L., Jiménez, J., Parra, M., Lloret, J., Mauri, P., & Lorenz, P. (2021). Deployment strategies of soil monitoring WSN for precision agriculture irrigation scheduling in rural areas. **Sensors, 21**(5), 1693. https://doi.org/10.3390/s21051693
4. Sunehra, D. (2019). Web-based smart irrigation system using Raspberry Pi. **International Journal of Advanced Research in Engineering & Technology, 10**(2), 1–10. https://doi.org/10.34218/IJARET.10.2.2019.006
5. Jaguey, J., Villa-Medina, J., Lopez-Guzman, A., & Porta-Gándara, M. (2015). Smartphone irrigation sensor. **IEEE Sensors Journal, 15**(9), 5122–5127. https://doi.org/10.1109/JSEN.2015.2435516
6. Efremova, N. (2019). Prediction of soil moisture content based on satellite data and sequence-to-sequence networks. **arXiv preprint arXiv:1907.03697**. https://doi.org/10.48550/arXiv.1907.03697
7. Okasha, A., Ibrahim, H., Elmetwalli, A., Khedher, K., Yaseen, Z., & Elsayed, S. (2021). Designing low-cost capacitive-based soil moisture sensor and smart monitoring unit operated by solar cells for greenhouse irrigation management. **Sensors, 21**(16), 5387. https://doi.org/10.3390/s21165387
8. Pramanik, M., Khanna, M., Singh, M., Singh, D., Sudhishri, S., Bhatia, A., & Ranjan, R. (2023). Evaluation of capacitance-based soil moisture sensors in IoT-based automatic basin irrigation system. **Research Square**. https://doi.org/10.21203/rs.3.rs-3043138/v1
9. Lin, L., Gu, W., Li, X., & Shen, G. (2017). Soil moisture monitoring system based on wireless sensor network. In **2017 2nd International Symposium on Advances in Electrical, Electronics and Computer Engineering (ISAEECE 2017)** (pp. 73–77). https://doi.org/10.2991/isaeece-17.2017.73