# Smart Plant Watering System

## Table of Contents

1. Introduction  
2. Objectives  
3. Features  
4. Hardware Requirements  
5. Software Requirements  
6. System Design  
 - Circuit Diagram  
 - Workflow Explanation  
7. Code Explanation  
8. Implementation  
9. Results  
10. Applications  
11. Future Improvements  
12. Conclusion

## 1. Introduction

The Smart Plant Watering System is an IoT-based project designed to automate plant irrigation.   
It monitors environmental parameters such as soil moisture, temperature, and humidity to make precise watering   
decisions. By integrating with the Blynk platform, users can monitor data and control the system remotely using   
their smartphones.

## 2. Objectives

- To reduce water wastage through intelligent irrigation.  
- To minimize human intervention in plant care.  
- To provide real-time data and control via IoT integration.

## 3. Features

1. Automatic Pump Control: Activates a pump when the soil moisture level is below the threshold.  
2. Real-Time Monitoring: Displays environmental parameters on an OLED screen and sends data to the Blynk app.  
3. Manual Control: Allows users to override the automatic pump via the app.  
4. Customizable Thresholds: Soil moisture thresholds can be adjusted remotely.  
5. Error Detection: Identifies and displays sensor malfunctions.  
6. Energy Efficiency: Uses minimal power by operating only when necessary.

## 4. Hardware Requirements

|  |  |  |
| --- | --- | --- |
| Component | Quantity | Purpose |
| ESP8266 Microcontroller | 1 | Main control and Wi-Fi connectivity |
| Soil Moisture Sensor | 1 | Measure soil moisture levels |
| DHT11 Sensor | 1 | Measure temperature and humidity |
| OLED Display (128x64) | 1 | Show sensor readings locally |
| Relay Module | 1 | Control water pump |
| Water Pump | 1 | Deliver water to the plants |
| Connecting Wires | As needed | Electrical connections |
| Power Supply | 1 | Provide power to components |

## 5. Software Requirements

- Arduino IDE: Programming the ESP8266.  
- Blynk App: Remote monitoring and control.  
- Libraries:  
 - BlynkSimpleEsp8266.h  
 - Adafruit\_GFX.h  
 - Adafruit\_SSD1306.h  
 - DHTesp.h  
 - EEPROM.h

## 6. System Design

### Circuit Diagram

The connections are as follows:  
1. Soil Moisture Sensor: Output to A0 of ESP8266.  
2. DHT11 Sensor: Data pin to D4.  
3. OLED Display: SDA to D1, SCL to D2.  
4. Relay Module: Control pin to pin 13.

### Workflow Explanation

1. The ESP8266 connects to Wi-Fi and synchronizes with the Blynk app.  
2. Sensors periodically measure soil moisture, temperature, and humidity.  
3. If soil moisture falls below the threshold, the relay activates the pump.  
4. The OLED screen and Blynk app display real-time data.  
5. Manual control and adjustments are handled via the app.

## 7. Code Explanation

The code is divided into the following sections:  
1. Initialization: Sets up the Wi-Fi connection, sensors, and display.  
2. Data Collection: Reads and processes data from the sensors.  
3. Automation Logic: Determines when to activate or deactivate the pump.  
4. Blynk Integration: Syncs data with the app and listens for user commands.  
5. Display Management: Shows real-time data and error messages on the OLED.

## 8. Implementation

1. Install the required libraries in the Arduino IDE.  
2. Write the code into the IDE, configure the Wi-Fi credentials, and upload it to the ESP8266.  
3. Assemble the circuit on a breadboard or PCB.  
4. Open the Blynk app, configure widgets, and monitor the system remotely.

## 9. Results

The system successfully:  
1. Monitors and displays real-time data.  
2. Activates the pump only when required, conserving water and electricity.  
3. Allows manual control and customization via the Blynk app.  
4. Detects and reports sensor issues effectively.

## 10. Applications

1. Gardens: Ensures plants are adequately watered without manual effort.  
2. Agriculture: Automates small-scale irrigation systems.  
3. Smart Homes: Integrates with home automation systems.

## 11. Future Improvements

1. Rain Detection: Add a rain sensor to prevent unnecessary watering.  
2. Water Level Monitoring: Include a water level sensor to avoid dry running.  
3. Solar Power: Integrate a solar panel for energy efficiency.  
4. Multi-Plant Support: Expand the system to monitor multiple plants.  
5. Mobile Notifications: Notify users of system status changes via push notifications.

## 12. Conclusion

The Smart Plant Watering System demonstrates the practical application of IoT in agriculture and gardening.   
It optimizes water usage, reduces human intervention, and provides a convenient way to monitor and manage   
plant care remotely.