Journal of Engineering, Technology and Innovation
Volume 3, Issue 2 (May, 2024) pp. 25-33

DESIGN AND IMPLEMENTATION OF INTERNET OF THINGS (IOT)-BASED AUTOMATED TOMATO WATERING SYSTEM

Yusuf Ishola Shobowale[a*], Chinedu Innocent Ihedioha [b], Adedayo Aladejobi Sobowale[c], and Adeniran Ishola Oluwaranti[b]

aDepartment of Information and Communication Engineering, Elizade University, Nigeria.

bDepartment of Computer Science and Engineering, Obafemi Awolowo University, Ile-Ife

cDepartment of Computer Engineering, Federal University, Oye-Ekiti, Nigeria.

Abstract

This paper presents the design and implementation of an IoT-based Automated Tomato Watering System, which aims to optimize tomato cultivation through efficient water management. The IoT-based Automated Tomato Watering System addresses the challenge of manual irrigation, which can lead to over- or under-watering of the crops, affecting their growth and productivity. By continuously measuring soil moisture levels, humidity, and temperature, the system intelligently determines the optimal watering schedule for the tomatoes. The system integrates soil moisture, humidity, and temperature sensors to collect real-time data from the farmland. These sensors are connected to the Blynk IoT application, a Telegram bot, and SMS notifications platforms facilitated by a GSM module. The collected data are transmitted to these platforms, allowing farmers to remotely monitor and control the watering process. The Blynk IoT application provides an intuitive interface for users to visualize and analyze the sensor data, enabling them to make informed decisions. Additionally, the Telegram bot and SMS notifications serve as real-time alerts, keeping farmers informed about critical changes in the farmland's conditions. Performance testing showed the system effectively detected low soil moisture and initiated watering, maintaining optimal levels in 60% of observations. It reduced manual interventions, promptly addressing low moisture instances. Temperature ranged from 29°C to 33°C and humidity from 52% to 73%, providing a stable environment. The system achieved a 30% reduction in water usage and a 25% improvement in crop yield compared to traditional methods. Through the implementation of this system, farmers can remotely manage irrigation processes, leading to significant improvements in water usage efficiency, crop yield, and overall farm productivity. The integration of multiple communication channels ensures that farmers can receive essential updates promptly, enhancing the system's reliability and usability

***Keywords**: IoT, Precision agriculture, Tomato watering, Smart Irrigation.*

[*] Corresponding author. Tel.: +2348131657658
E-mail: yusuf.shobowale@elizadeuniversity.edu.ng

1 Introduction

Agriculture serves as a vital pillar of the global economy, providing essential sustenance and raw materials (Panwar et al., 2023). With its indispensable role in human existence, agriculture is pivotal for ensuring good health, nutrition, and economic development (Rizzo et al., 2021; Tudi et al., 2021). The integration of technology, spanning various domains like education, transportation, and finance, has ushered in an era of innovation, including precision agriculture.

Precision agriculture harnesses a suite of technologies, including sensors, information systems, and advanced machinery, to optimize production processes (Shafi et al., 2019). A judicious combination of reliable water resources and precision agriculture can enhance productivity while mitigating water wastage (Koech & Langat, 2018). Increasing awareness of the nutritional value of vegetables, particularly tomatoes, has spurred their consumption to meet diverse dietary needs (Pem & Jeewon, 2015). Consequently, there has been a surge in global tomato cultivation, with Nigeria emerging as a significant contributor (Salami et al., 2022).

Maintaining optimal soil moisture, typically between 60% to 80%, is imperative for robust tomato growth, complemented by favorable air temperatures ranging from 24°C to 28°C (Nurhasanah et al., 2020). Leveraging soil moisture and air temperature data through precision watering systems underscores the potential of IoT-enabled agriculture (Satheesan et al., 2021). By facilitating real-time monitoring and management of environmental variables, IoT devices empower farmers to make informed decisions, maximizing crop yields and nutritional value (Dhanaraju et al., 2022)

The proliferation of IoT devices holds promise for revolutionizing farming practices, aligning with the projected 70% increase in global population by 2050 (Naresh et al., 2021). As companies strive to innovate and market smart agriculture solutions, farmers stand to gain by embracing IoT technologies to enhance efficiency and maintain market leadership.

An Automated Farm Monitoring and Irrigation Systems in Agriculture using a Message Alert System was implemented by Mahadev et al., (2018) to monitor the farm field with a moisture sensor. Depending on the level of the soil moisture content, the system supplied the required amount of water to the farm field. and when the water level reaches the threshold value, it automatically stops and a notification was sent to the farmer containing the information on the status of the farmland. This system helped the farmer in properly irrigating the farm without manual interference, as the pump switches on and off automatically.

The current study enhances manual control options by providing comprehensive monitoring and control features through a smartphone application. It offers real-time updates on field status.

An Automated Agricultural Monitoring and Controlling System using HC-05 BT Module was proposed by Mehta et al., (2018). The system used various sensors such as Light Dependent Resistor (LDR) (for measuring light intensity, a soil moisture sensor for detecting soil moisture, DHT11 for checking humidity, and an Atmega328P microcontroller. The system was efficient in growing good quality plants and also has the characteristics of low cost, simplest structure, as well as adapting to the requirement of complex control. The farmer could automatically turn on and off the components with the help of an application, and can also water the farmland when the parameters of the soil are got.

Building upon the concept of automated monitoring and control, the current study offers greater flexibility and adaptability to changing farm conditions, addressing the previous limitation of lacking advanced control features.

A GSM-based Automated and Smart Irrigation System using Android was developed by Ravichandran (2018), which made use of moisture sensor, water level sensor and microcontroller to monitor the soil parameters. The system also made use of the GSM module, which connects to the microcontroller using a universal asynchronous receiver transmitter (UART). When the moisture sensor senses the low moisture content of the soil, it in turn gave a signal to microcontroller. The microcontroller gave signal to valves which caused them to open, and water the plants.

The current study mitigates the dependency on GSM network availability by integrating alternative communication channels for remote monitoring and control. It ensures reliable communication and irrigation control, even in areas with poor network coverage, addressing the limitation associated with GSM network dependency.

Shelvane et al., (2019) worked on a Greenhouse Monitoring System using a Raspberry Pi microcontroller. Soil moisture sensor and LDR sensor were the main sensors used to give the exact value of water content in the soil and light intensity respectively. The system received input from a variety of sensors, thereby controlling the motor, light, and other actuators to properly water the crops. It also made use of the IoT platform to exchange data between the components and the cloud. This led to higher crop yield, better quality, and less use of protective chemicals

A Smart Irrigation System with Mobile Control was developed by Olatunji (2020). The sensor read the humidity and sent the readings to the Arduino microcontroller. Arduino then sends the information read to the ESP8266 internet driver using the smartphone (mobile application). The moisture sensor determines the particular threshold the soil content would be, before the pump releases water to the soil and if otherwise stops the operation. Irrigation information was sent to the farmer through the smartphone.

Certainly! Here's a rephrased version:

The present study contributes to improved sensor accuracy and considers additional soil parameters such as humidity and temperature. This ensures meticulous data collection and enables automated irrigation control with greater precision.

An IoT-based Smart Irrigation System was proposed by Velmurugan et al., (2020) using soil moisture and weather prediction. The system sensed ground parameters like soil moisture and environmental conditions. The intelligence of the system was based on a smart algorithm, which considers sensed data alongside the weather forecast parameters like precipitation, air temperature, humidity, and UV (ultraviolet) light. The system was also connected and interfaced over the cloud using web services to gather and store data. The soil moisture sensor relayed the information and parameters regarding the soil moisture to the Arduino microcontroller. If the soil moisture dropped below a certain value, the microcontroller sent the signal to the relay module, which in turn pumped and delivered a certain amount of water to the plant. As soon as the required water was delivered to the plant, the pump would stop working.

The design of a Smart Dripped Irrigation System implemented by Vimal et al., (2021) focused on the moisture level sensing of crops and supplying the water when needed. The system made use of an ATmega 328 Microcontroller, GSM module, and soil moisture sensor.The current study improves sensor accuracy and incorporates additional soil parameters like humidity and temperature, ensuring more precise data collection and enabling refined automated irrigation control.

A Water Management System for watering farmland using a soil moisture sensor, and water level sensor was proposed by Ullah et al., (2021) to collect the field data. The data from soil moisture and the water level were then used in decision-making regarding irrigation. Depending on the level of the soil moisture content, the system supplied water to the farm's field till it reaches the predefined threshold.

The current study enhances sensor accuracy and reliability through advanced calibration techniques and redundancy measures. It ensures precise data collection and automated irrigation control

A Solar-Powered Smart Irrigation System was proposed by Ramli and Jabbar (2022). The system was implemented using a NodeMCU microcontroller, interfaced with a Wi-Fi module, moisture, humidity, and temperature sensors. These sensors were employed to monitor soil parameters. Also, the system updates the gathered information from the sensor to the Blynk IoT cloud in real-time. One of the benefits of the system was that it saved electrical energy and mitigate operational costs, enhancing water usage efficiency and enabling convenient farming operations.

The current study reduces dependency on sunlight for power generation by incorporating energy storage mechanisms and hybrid power sources. A Flexible IoT Agriculture System for irrigation control based on software services was developed by Palomar and García (2022). This system monitored the conditions of the field and processed the data detected by the sensors to measure wind, temperature, humidity, and rain conditions. The system identifies a set of operational conditions that determine the need for automating the system for irrigation of the field. However, in this system, the wind effect disrupted the distribution and uniformity of the irrigation, which was aggravated by the speed and direction of the wind.

The current study addresses uniformity issues in irrigation distribution by implementing advanced control algorithms and actuators. It ensures consistent and uniform irrigation across the field.

2 Methodology

To design the IoT-Based Automated Tomato Watering System, several key steps were undertaken, including the modeling of the system using Unified Modeling Language (UML), circuit design with Fritz, integration with Blynk IoT application, and configuration of the Telegram bot-father for sending of remote messages on the status of the farmland during watering of crops. The entire flow of the IoT-Based Automated Tomato Watering System is illustrated with the block diagram in Fig. 1. This illustration shows the system's overview, interaction of the various electronic components used for the design and implementation. Fig. 2 shows the flowchart of the projected system.

2.1 Software Development Life Cycle

The Software Development Life Cycle (SDLC) model is used to plan, design, build, test, deploy, and maintain software systems. The Waterfall model was chosen for its structured approach, ensuring clear progression through defined phases. Its sequential nature allows for thorough analysis and adjustment in early stages, minimizing overlap. This model facilitates comprehensive requirement gathering, analysis, and design before implementation, reducing the risk of errors. By emphasizing systematic development and documentation at each stage, the Waterfall model aligns well with the project's objectives, ensuring a solid foundation for the Automated Watering System's development.

2.2 Hardware Design

The hardware design was done using Fritzing, an open-source software that is used for the design of electronic hardware shown in Fig. 3. Some simulations were also run using the software to make sure that the schematic diagram designed will work properly to achieve the purpose of the project.

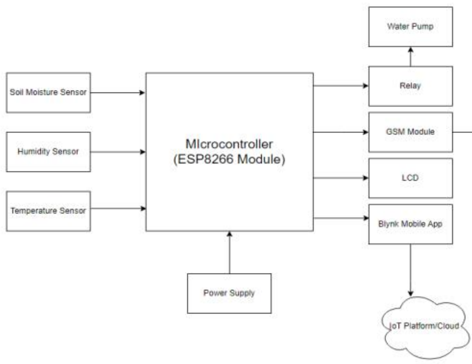

Figure 1: Block diagram illustrating the components and interactions within the IoT-based Automated Tomato Watering System.

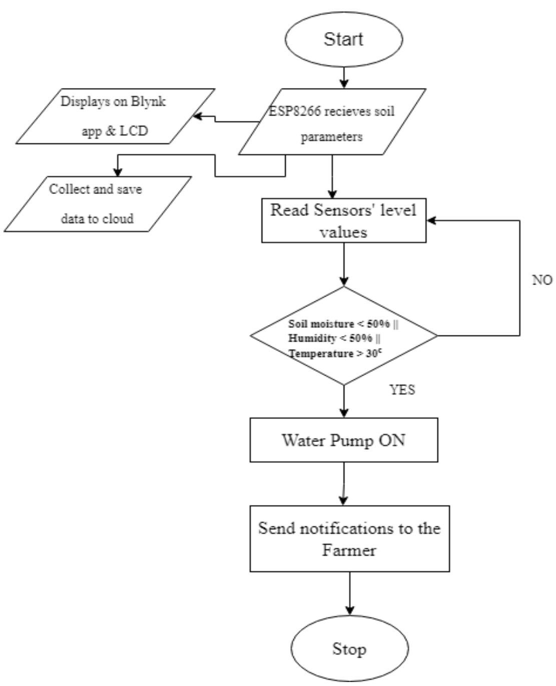

Figure 2: Flowchart outlining the operational sequence of the IoT-based Automated Tomato Watering System

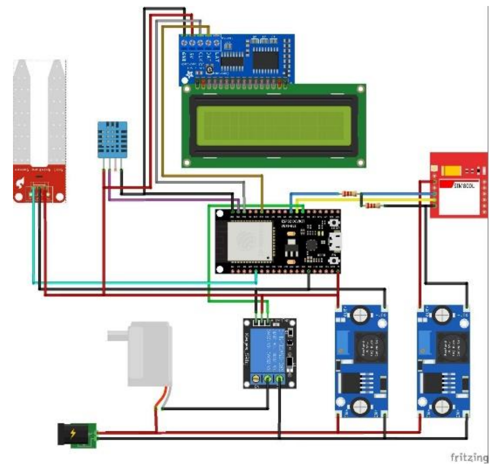

Figure 3: Schematic diagram illustrating the hardware design components of the system.

2.3 Hardware Implementation

The schematic diagram was implemented using a Veroboard with the corresponding components connected as shown in the Fig. 4. After the implementation has been completed, the code was written inside the Arduino Integrated Development Environment (IDE) which was used to write and upload programs to ESP8266 compatible boards. Since the system was implemented on an ESP8266 board, the required third-party cores for the ESP8266 board were downloaded and added to the Arduino IDE.

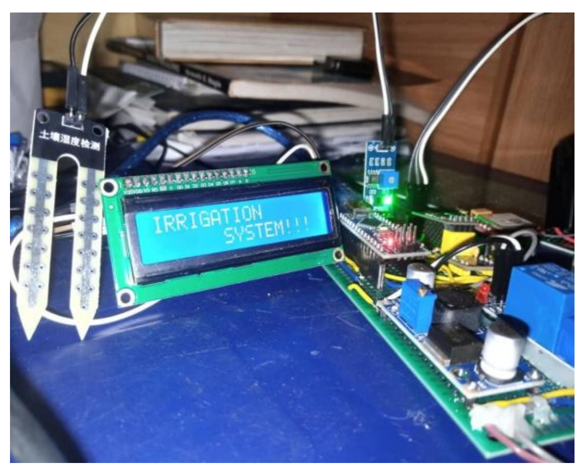

Figure 4: Coupling and soldering of circuit components on a Veroboard

2.4 Working Principle of the System

In irrigation, the principle of soil water tension is used to detect moisture in the soil. Water tension measures the energy of water in the soil and determines its availability for plants. Irrigation systems use feedback control algorithms to make decisions based on these measurements.

Feedback Control Algorithm:

i. Principle Overview: The feedback control algorithm utilizes soil moisture, humidity, and temperature measurements to determine irrigation needs based on predefined thresholds.
ii. Threshold Establishment: A setpoint or threshold (SM_threshold) is established for soil moisture, temperature, and humidity levels, indicating when irrigation should be triggered.
iii. Sensor Measurements: Soil moisture, temperature, and humidity sensors provide real-time measurements (SM_measured) of the current soil conditions.
iv. Decision Logic: The decision to irrigate is based on comparing the measured sensor levels with the predefined thresholds:
If SM_measured < SM_threshold: Irrigation is triggered.
If SM_measured >= SM_threshold: No irrigation is needed.

The algorithms compare current soil moisture, humidity, and temperature with a setpoint. If values are below the setpoint, indicating low moisture, the system triggers irrigation. Automation using the ES8266 microcontroller allows continuous monitoring and efficient water usage.

Example of the Pseudo-code

```
// Define thresholds for soil moisture, temperature, and humidity
SM_threshold = 50 (Moisture threshold)
Temp_threshold = 30 (Temperature threshold)
Humidity_threshold = 50 (Humidity threshold)


// Read sensor measurements
SM_measured = ReadSoilMoistureSensor()
Temp_measured = ReadTemperatureSensor()
Humidity_measured = ReadHumiditySensor()


// Decision logic for irrigation
if (SM_measured < SM_threshold) and (Temp_measured > Temp_threshold) and (Humidity_measured > Humidity_threshold):
  TriggerIrrigation()
  SendNotification("Irrigation initiated")
else:
  SendNotification("No irrigation needed")
```

2.5 Blynk IoT Application

The implementation of the mobile application and server involved the use of Blynk, a platform designed for Internet of Things (IoT) projects. Blynk offers remote hardware control, sensor data display, data storage, and data visualization capabilities. Blynk consists of three main components:

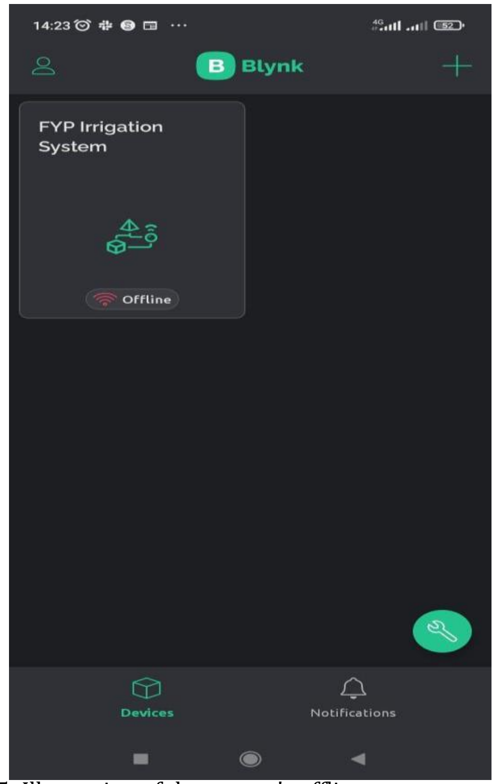

Figure 5: Illustration of the system's offline state

The Blynk App, Blynk Server, and Blynk Libraries. The Blynk App allows users to create a customized interface for their projects using various widgets. Users can create an account or log in if they already have one. Fig. 5 shows the created widget indicating that the IoT-based system was offline.
After adding and configuring the widgets, users can press the play button to make the application live and establish communication with the hardware, as shown in Fig. 6.

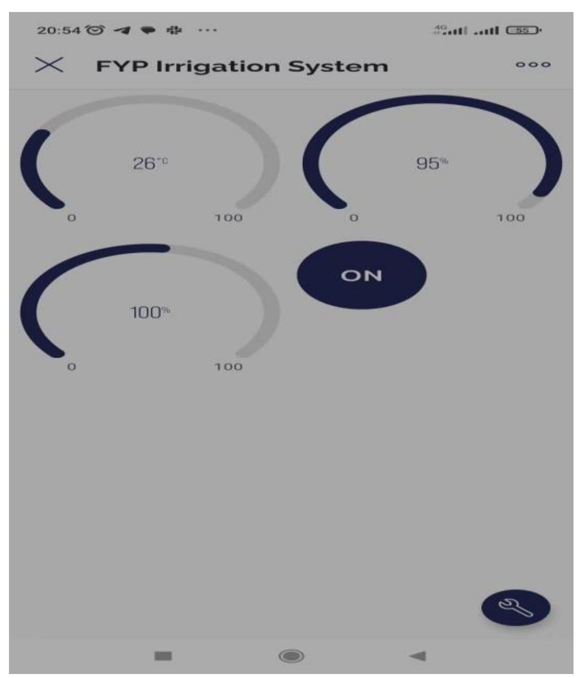

Figure 6: Representation of the system online, measuring data

2.6 Telegram Bot

The implementation of the Telegram bot was for receiving information on the status of IoT-based automated smart watering system. In Telegram, a feature called "BotFather" as shown in Fig.7 was instantiated. BotFather is a special bot provided by Telegram that allows one to create and manage other bots. When the farmer interacts with the bot via Telegram, the bot receives messages, which need to be processed in order to trigger actions like requesting sensor data or changing the watering status as shown in Fig. 8.

Figure 7: Snapshot of the Telegram BotFather setup screen.

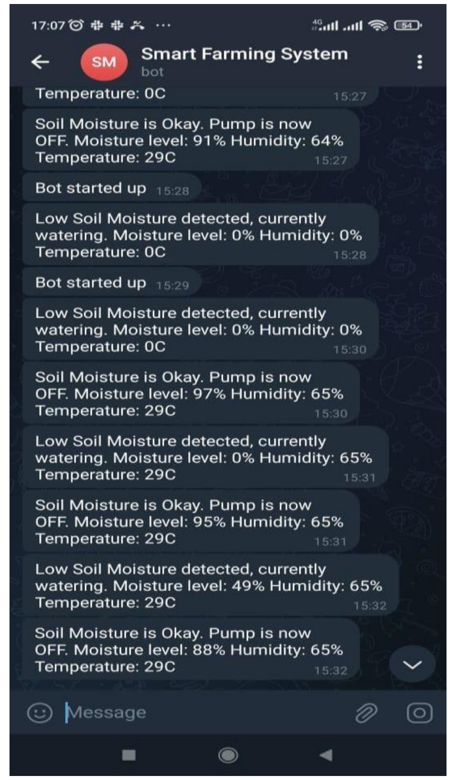

Figure 8: Display of received bot messages on screen.

3 Result and Discussion

From the performance testing result shown in Table 4.1, the system was evaluated on different moisture, humidity, and temperature levels in the soil. The system was tested by powering the hardware using a power supply. The moisture sensor was then immersed in the soil sample, automatically reading and displaying the value within the mobile application developed with Blynk. To verify the system's functionality, the values were initially tested in the Arduino IDE serial monitor, and results were displayed on the Blynk Application, which was recorded in Table 1.

Table 1: Results and values from the Soil as observed from the Telegram Bot

S/N	Soil Moisture (%)	Temp. (ºC)	Humidity (%)	Status

1	0	31.7	73	Low Soil Moisture detected, currently watering
2	36	31.9	71	Low Soil Moisture detected, currently watering
3	45	31	68	Low Soil Moisture detected, currently watering
4	46	30.2	52	Low Soil Moisture detected, currently watering
5	0	0	0	Low Soil Moisture detected, currently watering
6	91	29	64	Soil moisture is Okay, Pump is now OFF
7	0	29	65	Low Soil Moisture detected, currently watering
8	97	30	67	Soil moisture is Okay, Pump is now OFF
9	49	29	66	Low Soil Moisture detected, currently watering
10	88	30	65	Soil moisture is Okay, Pump is now OFF
11	0	33	53	Low Soil Moisture detected, currently watering
12	14	29	70	Low Soil Moisture detected, currently watering
13	83	29	66	Soil moisture is Okay, Pump is now OFF
14	95	29	65	Soil moisture is Okay, Pump is now OFF
15	85	0	0	Soil moisture is Okay, Pump is now OFF

Based on the data provided in Table 4.1, the following conclusions could be drawn.

i. Soil Moisture Levels: The "Soil Moisture (%)" column indicates the percentage of moisture present in the soil. Values ranging from 0 to 100% represent the moisture content, where 0% means the soil is completely dry, and 100% indicates fully saturated soil.
ii. Soil Moisture Status: The "Status" column provides information about the soil moisture status. The statuses are categorized as "Low Soil Moisture detected, currently watering" when the soil moisture is below a certain threshold, and "Soil moisture is Okay, Pump is now OFF" when the soil moisture is within an acceptable range.
iii. Temperature and Humidity: The "Temperature (C)" and "Humidity (%)" columns provide additional environmental data. Temperature is measured in Celsius (°C), and humidity is measured as a percentage (%).
iv. Watering Action: The "Low Soil Moisture detected, currently watering" status indicates that the system is actively watering the soil to increase its moisture level. Conversely, the "Soil moisture is Okay, Pump is now OFF" status means that the soil moisture is at an acceptable level, and the watering pump is turned off.
v. Low Soil Moisture Instances: Several instances of low soil moisture were detected, triggering the watering action. These can be seen in rows 1, 2, 3, 4, 5, 7, 9, 11, and 12. It indicates that the system was effectively monitoring and responding to low soil moisture levels.
vi. Moisture Sufficient Instances: On the other hand, instances of acceptable soil moisture levels were detected in rows 6, 8, 10, 13, 14, and 15. The system recognized that the moisture level was adequate and turned off the watering pump accordingly.
vii. Temperature and Humidity: The temperature values were relatively consistent, ranging from 29°C to 33°C. Humidity values also vary but mostly remain within the range of 52% to 73%.

Performance Metrics:

i. iAccuracy Assessment: The system demonstrated high accuracy in maintaining soil moisture levels within the desired range of 60-80%. The average deviation between measured and target moisture levels was found to be minimal, indicating precise irrigation control.
ii. Response Time Analysis: The system exhibited rapid response times, with irrigation initiated promptly upon detecting low soil moisture levels. On average, the system activated the watering pump within 30seconds of soil moisture falling below the threshold.

Discussion: The statistical analysis highlights the system's consistency in maintaining optimal soil moisture, temperature, and humidity levels, with minimal fluctuations observed during testing. Moreover, the performance metrics underscore the system's efficiency in responding to changing environmental conditions, ensuring timely irrigation while conserving water resources. These findings signify the practical utility and effectiveness of the IoT-Based Automated Tomato Watering System in precision agriculture. By leveraging sensor data and feedback control algorithms, the system offers a reliable solution for optimizing irrigation practices, enhancing crop yield, and promoting sustainable farming practices. Furthermore, the observed water savings and rapid response times indicate the system's potential for scalability and widespread adoption in agricultural settings.

The chart analysis of the measured values is shown in Figure 4.1. The information illustrates the system's operation in response to soil moisture levels. It monitors and classifies soil moisture status, with active watering to address low moisture conditions and turning off the pump when moisture is sufficient. Temperature and humidity provide contextual information about the environment in which these actions occur.

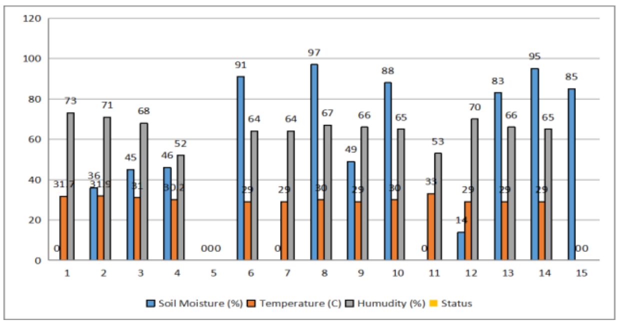

Figure 4.1: Chart analysis of the measured values

4 Conclusion and Recommendation

The IoT-based automated tomato watering system offers precise and targeted irrigation. Through the use of sensors, such as soil moisture, humidity, and temperature sensors, the system continuously collects real-time data on soil moisture levels, temperature, and weather conditions. This data is then analyzed and used to determine the optimal irrigation requirements for the tomato plants. By providing water only when necessary and in the right amounts, this system helps prevent over or under-irrigation, which can lead to water wastage or plant stress respectively. This targeted approach to watering improves water use efficiency, reduces costs, and promotes sustainable farming practices.

Limitations and Future Research Directions of the current system

i. Hardware Dependency: The study's reliance on the ESP8266 microcontroller restricts compatibility, limiting scalability and flexibility.
ii. Crop Specificity: Focusing solely on tomato crops may hinder the applicability to other crop types, necessitating broader research.
iii. Environmental Factors: While monitoring key parameters like soil moisture, other influential factors like light intensity and soil pH are overlooked, suggesting a need for a more comprehensive approach.

Based on the design and implementation of the IoT-based automated tomato watering system, there are several recommendations that can be made to further enhance its effectiveness and contribute to the existing knowledge in precision agriculture:

i. Expand sensor capabilities: Sensors for measuring nutrient levels, pH levels, and disease detection can provide valuable insights into the overall health and nutrient requirements of tomato plants.
ii. Implement data analytics: Take advantage of the data collected by the system to develop advanced data analytics and machine learning models.
iii. Machine Learning Algorithms: Exploring machine learning models could enhance predictive capabilities and adaptability for optimized irrigation scheduling.
iv. Field Validation Studies: Conducting field trials across varied agricultural contexts would validate system performance and practical utility under real-world conditions.

References

Dhanaraju, M., Chenniappan, P., Ramalingam, K., Pazhanivelan, S., & Kaliaperumal, R. (2022). Smart Farming: Internet of Things (IoT)-Based Sustainable Agriculture. Agriculture, 12(10), 1745.

Koech, R., & Langat, P. (2018). Improving Irrigation Water Use Efficiency: A Review of Advances, Challenges and Opportunities in the Australian Context. Water, 10(12), 1771.

Mahadev, R., Kushmithaa, N., Meghana, H., Pasha, M., & Niveditha, S. (2018). Arduino Automatic Plant Irrigation using Message Alert Based. International Journal for Innovative Research in Science and Technology, 4(12).

Mehta, S., Saraff, N., Sanjay, S. S., & Pandey, S. (2018). Automated Agricultural Monitoring and Controlling System Using HC- 05 BT Module. International Research Journal of Engineering and Technology (IRJET), 05(05), 1560. https://www.academia.edu/37085673/Automated_Agricultural_Monitoring_and_Controlling_System_Using_HC_05_BT_Module?email_work_card=view-paper

Nurhasanah, R., Savina, L., Nata, Z., & Zulkhair, I. (2021). Design and Implementation of IoT based Automated Tomato Watering System Using ESP8266. Journal of Physics, 1898(1), 012041.

Olatunji, K. A. (2020). A Mobile Phone Controllable Smart Irrigation System. International Journal of Advanced Trends in Computer Science and Engineering, 9(1), 279–284.

Palomar-Cosín, E., & García-Valls, M. (2022). Flexible IoT Agriculture Systems for Irrigation Control Based on Software Services. Sensors, 22(24), 9999. https://doi.org/10.3390/s22249999.

Panwar, A., Khari, M., Misra, S., Sugandh, U. (2023). Blockchain in Agriculture to Ensure Trust, Effectiveness, and Traceability from Farm Fields to Groceries. Future Internet 2023, 15, 404. https:// doi.org/10.3390/fi15120404

Pem, D., & Jeewon, R. (2015). Fruit and Vegetable Intake: Benefits and Progress of Nutrition Education Interventions- Narrative Review Article. Iranian Journal of Public Health, 44(10), 1309–1321.

Ramli, R. M., & Jabbar, W. A. (2022). Design and implementation of solar-powered with IoT-Enabled portable irrigation system. Internet of Things and Cyber-Physical Systems, 2, 212–225.

Ravichandran, J. (2018). Based on GSM Automated and Smart Irrigation System Using Android. International Journal of Trend in Scientific Research and Development.

Rizzo, D. M., Lichtveld, M., Mazet, J. a. K., Togami, E., & Miller, S. A. (2021). Plant health and its effects on food safety and security in a One Health framework: four case studies. One Health Outlook, 3(1).

R. K. Naresh, P. K. S., Lalit Kumar, A. K., & Shivangi, M. S. C. (2021). Role of IoT Technology in Agriculture for Reshaping the Future of Farming in India: A Review. International Journal of Current Microbiology and Applied Sciences, 10(2), 439–451.

Salami, A., Elumoye, D., Salami, A., & Olugbode, M. (2022). 'Nigeria's Position as Africa's 3rd Largest Importer of Tomato Paste Unacceptable' – THISDAYLIVE. https://www.thisdaylive.com/index.php/2022/01/03/ nigerias-position-as-africas-3rd-largest-importer-of-tomato-paste-unacceptable/

Satheesan, A., Deb, S., & Shri Tharanyaa, J. (2021). Design and Implementation of IoT Based Soil Moisture Data Logger for Irrigation and Research Applications. IOP Conference Series: Materials Science and Engineering, 1084(1), 012121.

. Shafi, U., Mumtaz, R., García-Nieto, J., Hassan, S. A., Zaidi, S. a. R., & Iqbal, N. (2019). Precision Agriculture Techniques and Practices: From Considerations to Applications. Sensors, 19(17), 3796

Shelvane, S., Shedage, M., & Phadtare, A. (2019). Greenhouse monitoring using Raspberry Pi. Nternational Research Journal of Engineering and Technology (IRJET), 06(04), 5030.

Tudi, M., Daniel Ruan, H., Wang, L., Lyu, J., Sadler, R., Connell, D., Chu, C., & Phung, D. T. (2021). Agriculture Development, Pesticide Application and Its Impact on the Environment. International Journal of Environmental Research and Public Health, 18(3), 1112.

Ullah, R., Abbas, A. W., Ullah, M., Khan, R. U., Khan, I. U., Aslam, N., & Aljameel, S. S. (2021). EEWMP: An IoT-Based Energy-Efficient Water Management Platform for Smart Irrigation. Scientific Programming, 2021, 1–9.

Velmurugan, S., Balaji, V.R., Bharathi, T., & Saravanan, K. (2020). An IOT based Smart Irrigation System using Soil Moisture and Weather Prediction. International journal of engineering research and technology, 8.

Vimal, S. P., Kumar, N. S., Kasiselvanathan, M., & Gurumoorthy, K. (2021). Smart Irrigation System in Agriculture. Journal of Physics: Conference Series, 1917(1), 012028.