DEMOCRATIC AND POPULAR REPUBLIC OF ALGERIA
Ministry of Higher Education and Scientific Research
Mohamed Khider University – BISKRA
Faculty of Exact Sciences, Natural Sciences and Life Sciences

Department of Computer Science

Order N°: SIOD15-Startup/M2/2023

Dissertation

Presented to obtain the academic master's degree in

Computer Science

Option: Information Systems, Optimization and Decision (SIOD)

IoT-Based web application for monitoring irrigation

By :
ANANI Maria

Discussed on 25/06/2023 before the jury composed of :

Sahli Sihem	MAA	President
Ouaar Hanane	MCB	Advisor
Mohammedi Amira	MCA	Examiner
Rahal Soulef	...	Examiner

Academic year 2022-2023

بسم الله الرحمن الرحيم

Acknowledgements

Praise be to Allah, Lord of the Worlds, and may prayers and peace be upon his Messenger. First of all, thanks to Allah for his facilitation of this work and for giving me the strength, and knowledge to complete this work.

After that, I would like to thank the person who has contributed to the discussion of this work.

Start with a big thanks to my supervisor, ***Dr. Ouar Hanane****, for her teaching, her supervision, her advice, and her guidance from the beginning to the end of this work. Thanks to her, I learned a lot of new techniques, I'm fortunate to have had the opportunity to work with her.*

I would like also to extend my heartfelt thanks to the jury members for their evaluation and assessment of this work. I will not let this opportunity pass without my thanks to all the people of computer science department, especially the teachers.

*A big thanks also to the two institutions "****L'ITDAS****" and, "****MakerSpace****" that have accepted me and provided me with invaluable support. Their guidance and assistance have been instrumental in shaping my path and enabling me to achieve my goals.*

I would like to thank my parents from the bottom of my heart for their love, encouragement, and support throughout my studies. They have always believed in me and made sacrifices to help me succeed. I am truly grateful for their presence in my life and the values they have put in me.

Thanks to my sisters and brothers for their constant love, encouragement, and support. Thanks for always being there for me.
Thanks to my friends, who have been by my side through both the good times and the challenging moments. They have provided me with comfort, advice, and companionship, making every experience more enjoyable. I cherish the memories we have created together and the bond we share.

Finally, I want to thank my "invisible soldiers" for their invaluable help and encouragement throughout this work.

MARIA ANANI

Dedicate

I dedicate this modest work to:

My beloved ***parents****, words cannot express the depth of my gratitude and love for you.*

To my amazing Husband ***Walid****.*

To my Dearest Sisters and Dear Brothers ***Kawther*** *,* ***Nour*** *,* ***Yacine*** *,and* ***Younes*** *.*

To my cherished Friends ***Marwa*** *,* ***SaMra****,* ***Hadjer*** *,* ***Ines*** *,and* ***Kawther*** *.*

To all my best familly.

ABSTRACT

The aim of this project is a web application with an easy-to-use dashboard for farmers, allowing them to monitor and manage plant irrigation in an automated system. The development of this application includes the integratio of sensors to collect data on soil moisture and temperature, using a micro-controller in order to communicate with the prototype. The MQTT protocol will be utilized for communication, and a cloud-based messaging service called HiveMQ will be leveraged. Ensuring a seamless integration and functionality of all system components is a priority in this prototype. Through the utilization of this experimental model, the performance of the system can be improved, any issues can be addressed, and a reliable operation can be ensured. By combining this technology with the Internet of Things, farmers are provided with the necessary tools to enhance crop yields, implement precise watering schedules in order to prevent water waste, and optimize their farming practices.

Key words: Web application, irrigation, automated system, real-time, remote monitoring, customization of water, internet of Thinge, prototype, sensors, microcontroller, MQTT protocol, cloud-based messaging service, HiveMQ.

Résumé

L'objectif de ce projet consiste à développer une application web avec une interface conviviale, afin de permettre aux agriculteurs de surveiller et de gérer l'irrigation des plantes dans un système automatisé. Le développement de cette application inclut l'intégration de capteurs pour la collecte de données sur l'humidité et la température du sol, en utilisant un micro-contrôleur pour la communication avec le prototype. La communication se fera à l'aide du protocole MQTT (Message Queuing Telemetry Transport), et le service de messagerie basé sur le cloud appelé HiveMQ sera exploité. Assurer l'intégration et le bon fonctionnement de tous les composants du système est une priorité dans ce prototype. En utilisant ce modèle expérimental, les performances du système pourront être améliorées, les problèmes pourront être résolus et un fonctionnement fiable pourra être assuré. Cette technologie, combinée à l'Internet des objets, offre aux agriculteurs les outils nécessaires pour améliorer le rendement de leurs cultures, mettre en place des plannings d'arrosage précis afin d'éviter le gaspillage et d'optimiser leur consommation d'eau.

Mots clés: application web, l'irrigation des plantes, Système automatisé, Temps réel, la surveillance à distance, gestion de l'eau, Internet des objets, Prototype, Capteurs, Microcontrôleur, le protocole MQTT, messagerie basé sur le cloud, HiveMQ.

ملخص:

يهدف المشروع إلى تطوير تطبيق ويب يتضمن لوحة تحكم سهلة االستخدام مصممة للمزارعين، بهدف مراقبة وإدارة ري النباتات في نظام آلي. يتمحور االهتمام حول توفير الوصول إلى البيانات في الوقت الحقيقي، وتمكين المراقبة عن بُعد، وإتاحة إمكانية تخصيص إعدادات إدارة المياه وفقًا الحتياجاتهم الخاصة ومتطلبات النباتات. من خالل استغالل قوة إنترنت األشياء (IoT)، سيتم بناء واختبار نموذج أولي يدمج أجهزة االستشعار لجمع البيانات حول رطوبة التربة ودرجة الحرارة والرطوبة. سيتم تواصل النموذج األولي مع متحكم دقيق باستخدام MQTT (نقل الرسائل عن بعد)،
وسيستفيد من خدمة الرسائل المستندة إلى السحابة والمعروفة باسم HiveMQ. سيتم التحقق من توافق مكونات النظام في هذا النموذج األولي، والتأكد من أنها تعمل بشكل صحيح وتقديم الفوائد المرجوة. عن طريق استخدام هذا النموذج التجريبي جنبًا إلى جنب مع إنترنت األشياء، يمكن تحسين أداء النظام وضمان عمله بشكل جيد وموثوق. تساعد هذه التقنية المزارعين في تحسين محاصيلهم ووضع جداول ري دقيقة تحول دون هدر المياه، وتحقيق االستهالك األمثل لها.

الكلمات المفتاحية : تطبيق ويب، ري النباتات، نظام آلي، الوقت الحقيقي، المراقبة عن بُعد، إدارة المياه، إنترنت األشياء، نموذج أولي، متحكم دقيق، المستندة إلى السحابة.

Summary

List of Figures

List of Tables

General introduction

Managing water resources effectively in regions with limited availability presents a significant challenge. To maximize crop yields, precise irrigation is crucial, ensuring the appropriate amount of water at the right timing. The advent of "smart irrigation" systems, incorporating automation, sensors, and artificial intelligence (AI), has revolutionized agricultural methods. These innovative systems can accurately sense and analyze plant water requirements, enabling efficient water usage. Despite their cost-effectiveness, the adoption of smart irrigation remains low among farmers, primarily due to a lack of awareness. Consequently, conventional irrigation practices persist, resulting in suboptimal water utilization in agriculture.

In recent years, the concept of Smart Agriculture has emerged, transforming traditional farming approaches through the integration of Internet of Things (IoT) technologies. This application of cutting-edge digital farming techniques has enabled the transition from manual and static farming to an intelligent and dynamic process, leading to increased productivity and reduced human supervision. A prototype smart irrigation system will be developed by integrating sensors, a microcontroller, and communication technologies. The sensors will be strategically placed in the soil to ensure accurate measurement of soil moisture levels, while the sensor data will be processed by the microcontroller, enabling intelligent decisions based on predefined algorithms. By establishing a network between the sensors, microcontroller, and a central control unit using communication technologies, real-time monitoring and control of the irrigation system will be enabled. The watering process will be automated by continuously monitoring soil moisture levels and regulating the water supply accordingly, considering factors such as weather conditions and plant requirements.

How can traditional irrigation practices be improved to address issues such as overwatering, underwatering, inefficient water usage, inaccurate timing?

What role does the automation and monitoring of irrigation play in addressing these problems?

Our main objective is to utilize IoT techniques for irrigation control, integrating sensors, microcontroller, and communication technologies to achieve efficient and precise management of irrigation processes. By continuously monitoring and regulating optimal soil moisture levels through automated watering, our proposed smart irrigation system exemplifies this transformative technology. Through real-time monitoring, automation, and optimization based on factors like soil moisture, weather conditions, and plant requirements, this system has the potential to enhance water efficiency. It not only saves an astounding 80% of water typically consumed in irrigation but also minimizes human labor.

The organization of this dissertation is as follows:

The first chapter focus on introduction of the concept of smart agriculture, information about irrigation, the Internet of Things (IoT), and the sensors utilized in IoT, is undertaken. Moreover, the emphasis is placed on related works in the realm of monitoring irrigation.

The second chapter concentrate on Design and Contribution. Our proposed architecture for addressing the issue is presented, providing a detailed explanation of its components and functionality. The proposed architecture is accompanied by a UML diagram, which illustrates the system's structure and the relationships between different modules.

The third chapter speak on the further discussion of the implementation and results of our project takes place. The development tools and frameworks employed are highlighted, along with the presentation of screenshots of the web application.

This work concludes with a summary of our findings and the provision of future work suggestions.

Chapter 1

State of art

1.1 Introduction

The efficient utilization of water is a challenging task, especially in areas where water availability is a major concern. .The chapter talks about the rise of smart irrigation systems that use advanced technologies like automation, sensors, and artificial intelligence (AI) to make better use of water. However, many farmers still use traditional methods because they don't know about these new technologies. This chapter explores how information technology can help in farming, especially the impact of AI and the Internet of Things (IoT) on smart farming. It looks at different studies on using IoT for monitoring irrigation and picks out important ideas for implementing IoT in smart farming. Overall and aims to give us a better understanding of how we can use new technologies to manage water, improve crops, and take care of the environment in sustainable farming.

1.2 Irrigation and Water Management

Irrigation plays a vital role in nourishing crops and promoting their growth. This age-old practice has been instrumental in boosting agricultural productivity and sus- taining human life in regions with limited rainfall[20]. Contemporary irrigation sys- tems encompass a wide spectrum, ranging from traditional manual methods like flood irrigation to advanced automated systems that leverage sensors and technology to optimize water usage[15].While irrigation offers significant benefits for agriculture, it is crucial to acknowledge its potential adverse effects on the environment, including soil erosion, water pollution, and the depletion of groundwater resources. Hence, it becomes imperative to adopt sustainable irrigation practices that strike a harmonious balance between crop requirements and environmental preservation.

1.3 Traditional Irrigation

In arid or semi-arid regions with minimal water availability, traditional methods like hand pumps, canals, tube wells, Check Basins, Furrow Irrigation, Strip Irrigation, and Basin Irrigation have historically been employed to manually supply water to fields. While these conventional approaches achieved satisfactory results in the past, they relied on human or animal intervention and encountered challenges such as inefficient water usage, excessive irrigation, and inadequate irrigation practices [24]. To address these issues, modern irrigation systems have been introduced, offering various methods such as drip irrigation, sprinkler irrigation, and pot irrigation Some modern irrigation methods are discussed below. However, adopting these modern systems requires farmers to invest in hardware equipment and undergo the setup process, which may necessitate assistance from providers [42]. Moreover, the maintenance of these irrigation setups can be complex and often requires timely servicing or replacement due to various factors [4].

1.4 Types of modern irrigation

By delving into the various types of modern irrigation, we gain valuable insights into how these methods contribute to sustainable water management in agriculture.

1.4.1 Sprinkler Irrigation

Sprinkler irrigation is an effective technique that involves the delivery of water to crops using a network of pipes and strategically placed sprinklers. The design of these sprinklers ensures the uniform distribution of water across the entire field, minimizing water wastage and maximizing crop productivity [5].

The figure 1.1 [16] show the method of sprinkler irrigation.

Figure 1.1: Sprinkler irrigation

1.4.2 Drip Irrigation

Drip irrigation stands out as the most efficient method for providing water and nutrients to crops. This system delivers water and nutrients directly to the plant's root zone, precisely supplying the right quantities at the appropriate times. By ensuring that each plant receives optimal care, drip irrigation enables farmers to achieve higher yields while conserving water, minimizing fertilizer usage, and reducing energy consumption [38].

The figure 1.2 [39] show the method of drip irrigation.

Figure 1.2: Drip irrigation

1.4.3 Center pivot irrigation

DThe center pivot system has become the preferred choice for agricultural irrigation due to its numerous advantages, including low labor and maintenance demands, convenience, flexibility, performance, and user-friendly operation. When designed and operated effectively, especially when equipped with high-efficiency water applicators, a center pivot system becomes an invaluable tool for conserving three crucial resources: water, energy, and time [44].

The figure 1.3 [32] show the method of center pivot irrigation.

Figure 1.3: Center pivot irrigation

1.5 The Basics of Irrigation calculating Water Needs

This section highlights the crucial parameters used to calculate and determine the amount of water required by plants.

1.5.1 Water Requirements in Irrigation

Water needs calculation in irrigation is a crucial step in determining the precise amount of water required for optimal plant growth and development within an irrigation system. It involves a comprehensive assessment of various factors to estimate the exact water requirements. Plant species, growth stage, environmental conditions, and soil characteristics play significant roles in this calculation process. For instance, different plant species have varying water demands, and the growth stage of a plant affects its water requirements as well. Environmental conditions such as temperature, humidity, wind speed, and solar radiation influence the rate of evapotranspiration, which affects water needs. Additionally, soil characteristics, including texture, structure, and water-holding capacity, determine the availability and retention of water in the root zone[28].

1.5.2 Evapotranspiration and Plant Water Consumption

Evapotranspiration, the emission of water vapor, is a combined result of two processes: evaporation and plant transpiration. Evaporation occurs as a purely physical phenomenon, while transpiration involves the release of water vapor by plants. During the period of vegetation cover, the recharge of the water table through precipitation can be restricted. This is because a significant portion of the water is consumed through evapotranspiration by vegetation. Evapotranspiration encompasses water losses attributed to climatic conditions, including evaporation from the soil surface and transpiration from plants[10].

Accurate estimation of evapotranspiration is essential for efficient water management in irrigation. This estimation considers various factors such as climatic conditions, vegetation characteristics, and plant growth stage. By estimating evapotranspiration accurately, irrigation practitioners can optimize resource utilization, promote sustainable agriculture, and conserve water resources. Effective irrigation scheduling based on proper estimation prevents over-irrigation and under-irrigation issues, ensuring optimal plant health and crop productivity[13].

The calculation equation for evapotranspiration (ET0) is given by [40]:

$$ET0 = \frac{0.408 \cdot \Delta \cdot (Rn - G) + \gamma \cdot \frac{900}{T+273} \cdot u2 \cdot (es - ea)}{\Delta + \gamma \cdot (1 + 0.34 \cdot u2)}$$

In this equation, the variables represent the following:

- ET0: Evapotranspiration
- Δ: Slope of the saturation vapor pressure-temperature curve

- Rn: Net radiation at the crop surface
- G: Soil heat flux density
- γ: Psychrometric constant
- T: Mean daily air temperature
- u2: Wind speed at 2 meters above the ground
- es: Saturation vapor pressure
- ea: Actual vapor pressure

This equation takes into account various meteorological factors such as net radiation, air temperature, wind speed, and vapor pressure. These parameters influence the rate of water loss from a well-watered reference crop and are used to calculate ETo[13].

the figure 1.4 [3] show plant transpiration and evaporation.

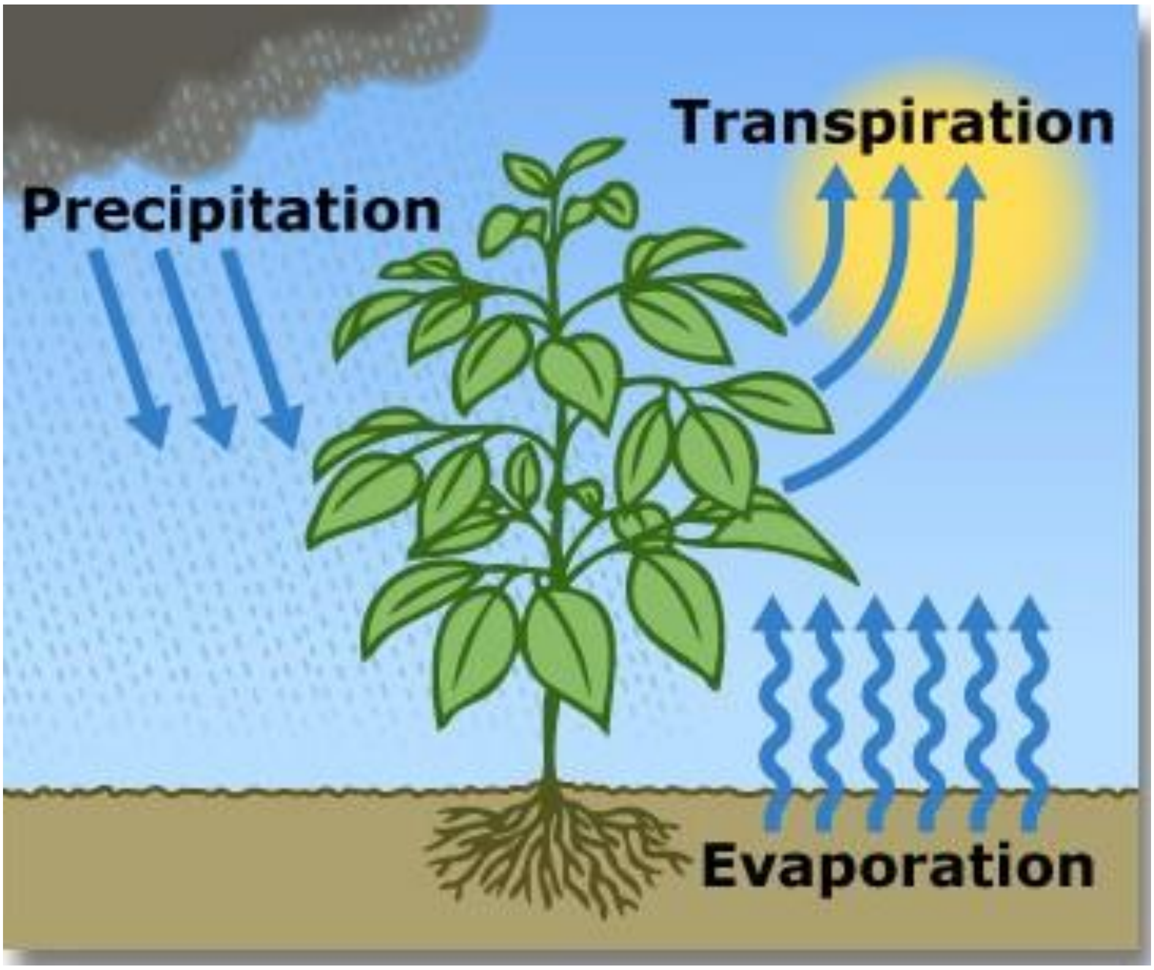


Figure 1.4: plant transpiration and evaporation

after that to To calculate the crop water requirements (ETc) using the crop coefficient (Kc), the equation is as follows:

$$ETc = Kc \times ETo$$

Where [40]:

- ETc represents the crop evapotranspiration,
- Kc is the crop coefficient specific to the crop and its growth stage,
- and ETo is the reference evapotranspiration.

1.6 Limitations of Modern Irrigation Systems

- Modern irrigation systems are expensive to install and maintain, requiring significant investment by farmers.
- While these types of techniques are often more efficient than traditional methods, they still require significant energy and water inputs. This will be a concern in areas with limited resources or where water scarcity is an issue.
- Rely on mechanical and electronic equipment that can be vulnerable to breakdowns, impacting crop yields and increasing maintenance costs.
- It has negative environmental impacts, such as soil erosion, water pollution, and habitat destruction.
- These systems can not be adaptable to changing weather patterns or soil conditions, limiting their ability to optimize water usage and crop yields.

The drawbacks of modern irrigation techniques have led to the emergence of smart irrigation systems that address these issues by using data, sensors, and automation to optimize water usage and reduce waste.

1.7 The Importance of Smart Irrigation Scheduling

The world is currently facing water scarcity, which is exacerbated by the growing global population and increasing demand for freshwater resources. With the current population at around 7.2 billion and projected to exceed 9 billion by 2050, the agriculture sector, particularly irrigation, consumes a significant amount of freshwater. However, due to the lack of cost-effective intelligent irrigation systems, developing countries tend to consume more water compared to developed countries to achieve the same crop yield. For instance, India possesses only about 4% of the world's freshwater resources but needs 2-4 times more water for some of its major agricultural products compared to countries like China and the USA. This situation calls for the urgent development of advanced technologies and smart strategies for the efficient utilization of freshwater resources. Figure 1.5 [33] provides a comparison of the efficiency of different irrigation techniques employed in farmland [33].

Figure 1.5: Various irrigation techniques and their efficiencies

1.8 Smart Agriculture Applications

Smart agriculture encompasses the wide-ranging implementation of artificial intelligence (AI) along with other digital technologies such as big data, the Internet of Things (IoT), and deep learning [36].

1.8.1 Definition of Internet of things (IoT)

The Internet of Things (IoT) is a network that links and integrates objects with the Internet, utilizing protocols that guarantee communication and information exchange across various devices. According to [25], the integration of IoT technology has led to improved automation processes and cost reduction. Automation not only reduces labor costs but also provides a continuous monitoring system, leading to cost reduction [41][37] in operation. Furthermore, automation has increased productivity by allowing smart farming activities to be performed even when the farmer is absent. This system regulates the optimal use of water [11] to maintain the required soil moisture level.

The Internet of Things (IoT) is a convergence of various technologies, including sensors, actuators, cloud services, and IoT protocols. The IoT architecture is composed of different layers that facilitate communication and information exchange between these technologies. This section aims to explore the functionality of each layer in the IoT architecture.

The figure 1.6 [9] illustrates the layers of the Internet of Things (IoT). It represents the different levels or stages involved in the IoT architecture.

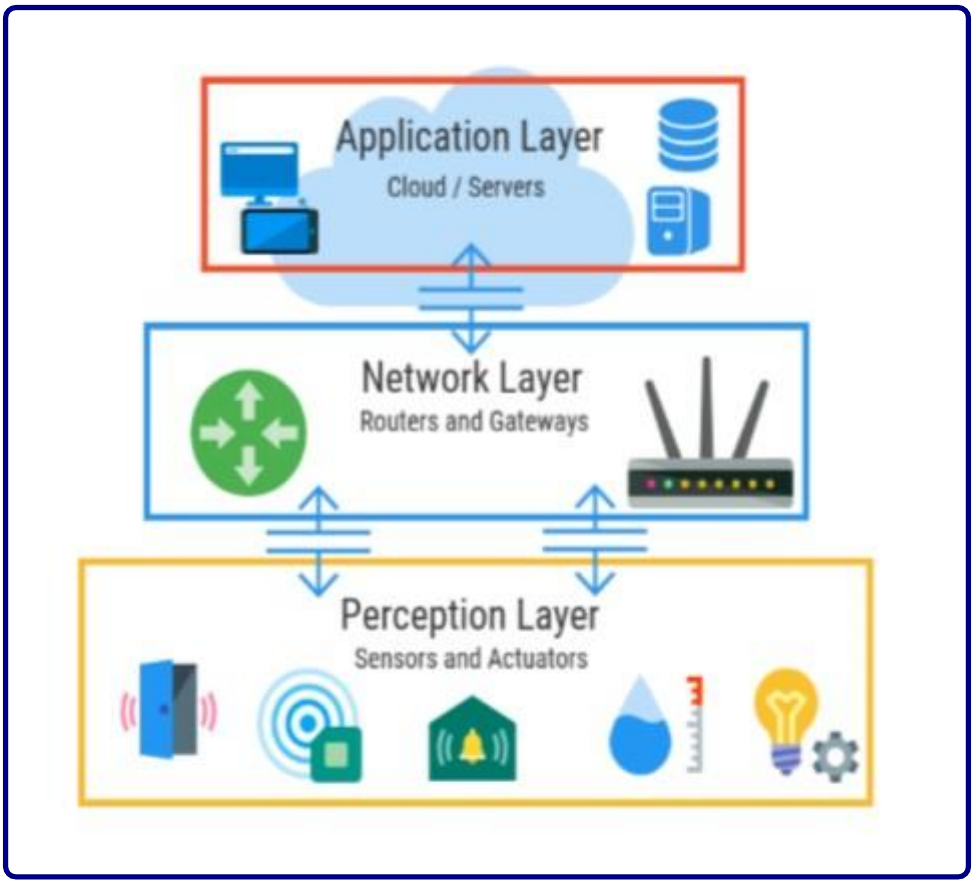


Figure 1.6: The fundamental Three Layer IoT Architecture

The three layers of IoT are:

- **Perception Layer:** This layer includes the devices and sensors that collect data from the physical world, such as temperature, humidity, and motion. The data collected by these devices is then sent to the next layer for processing.
- **Network Layer:** This layer is responsible for transmitting the data collected from the perception layer to the cloud or other systems for processing and analysis. It includes communication protocols, gateways, and other networking components that enable the transfer of data.
- **Application Layer:** This layer is where the data collected from the perception layer is analyzed, processed, and utilized to make decisions or perform actions. It includes the cloud infrastructure, analytics engines, and other software applications that enable the use of the collected data.

1.8.2 Overview of Communication Protocols in IoT

In the context of IoT, effective message transmission plays a vital role in enabling communication between different devices. This is crucial as it allows an IoT device to convey instructions or commands to other devices within the system, facilitating efficient system management and operation [31].

1.8.3 MQTT (Message Queue Telemetry Transport)

MQTT is a lightweight, open, and easy-to-implement client-server messaging transport protocol used in IoT. It operates over TCP/IP or other network protocols that offer ordered, lossless, bi-directional connections. MQTT follows a publish/subscribe message pattern, allowing for one-to-many message distribution. It is content-agnostic, meaning it does not depend on the payload content. The protocol offers three qualities of service for message delivery: "At most once" ensures best-effort delivery, where message loss can occur; "At least once" guarantees message arrival but may result in duplicate messages; and "Exactly once" ensures messages arrive exactly once. By utilizing these different levels of service, MQTT effectively reduces network traffic. Additionally, MQTT minimizes transport overhead and protocol exchange, resulting in efficient network utilization. It also includes a mechanism to notify interested parties of abnormal disconnections [7].

These factors contribute to the effectiveness and efficiency of MQTT in enabling seamless and reliable communication in IoT systems [8]:

- Lightweight and efficient protocol
- Publish/subscribe messaging pattern
- Scalability
- Reliable message delivery with different levels of quality of service (QoS)
- Low network bandwidth requirements
- Support for intermittent and unreliable networks
- Bi-directional communication capabilities
- Built-in support for security and authentication
- Wide range of platform and programming language support

The figure 1.7 [6] show the architecture of the MQTT protocol

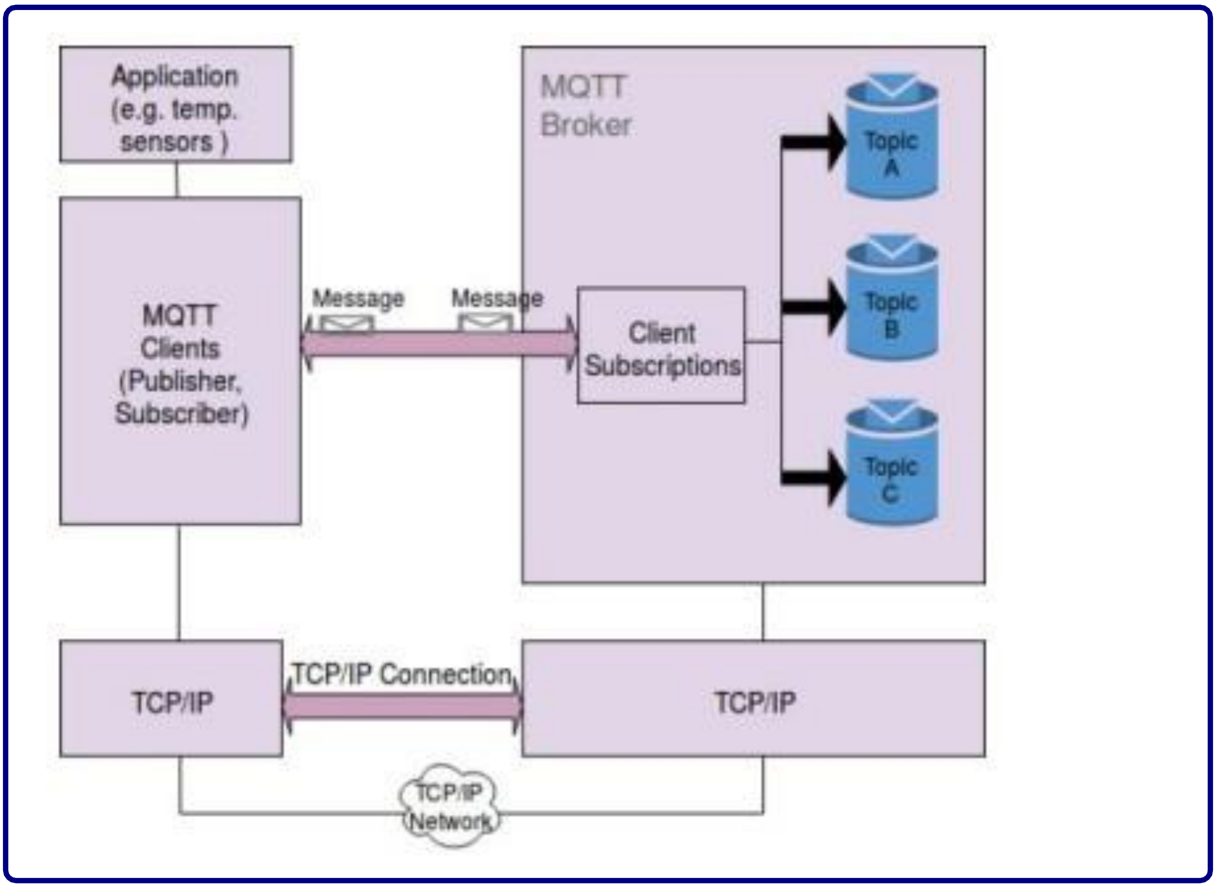


Figure 1.7: MQTT Architecture

1.8.4 SOAP (Simple Object Access Protocol)

is a widely used communication protocol in IoT systems for exchanging structured and extensible information. It offers a high level of extensibility and flexibility, making it suitable for integrating with existing systems and leveraging well-established XML-based tools and technologies. SOAP's adherence to XML standards ensures interoperability and enables seamless communication between diverse devices and platforms. Additionally, SOAP provides reliable messaging through built-in mechanisms such as message acknowledgments, retries, and error handling. This enhances the overall reliability and robustness of IoT applications[47].

The SOAP protocol offers significant advantages in various areas, including data-driven decision making, automation and efficiency, predictive maintenance, enhanced safety and security, and improved customer experience. By leveraging this protocol, organizations can make informed decisions based on data insights, automate processes to increase efficiency, proactively conduct maintenance to prevent issues, enhance safety measures, and provide superior customer experiences[14].

The the figure1.8 [46] show the SOAP Architecture:

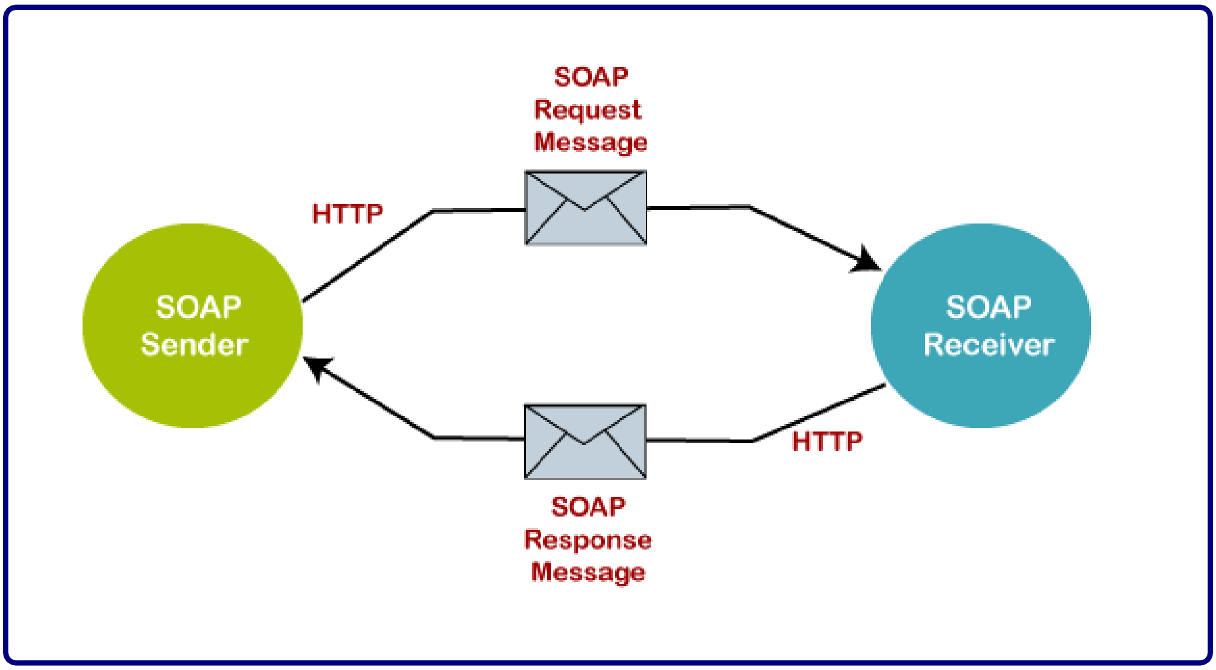


Figure 1.8: SOAP Architecture

1.8.5 The WebSocket protocol

WebSocket protocol created to address the limitations of continuous data exchange between clients and servers that were not adequately supported by HTTP. It introduces a bidirectional communication channel that operates through a single socket, enabling seamless and simultaneous data transmission between both ends. Unlike the HTTP protocol, WebSocket allows for asynchronous communication, allowing data to be sent at any time during the established connection. This enhanced capability ensures real-time and efficient data exchange between clients and servers[19].

WebSocket protocol offers significant advantages for IoT applications. It provides a bidirectional and persistent communication channel through a single socket, allowing real-time data exchange between client and server. This enables devices to send and receive data simultaneously, ensuring efficient and synchronized communication. WebSocket's low-latency and asynchronous nature further enhance its usability in IoT, enabling fast response times and efficient handling of multiple connections. Additionally, WebSocket is compatible with various programming languages, offers secure communication through encryption and authentication, and supports scalability. These features make WebSocket an ideal choice for seamless and reliable communication in IoT systems [23].

1.9 Artificial Intelligence in Agriculture

Artificial Intelligence (AI) techniques are extensively employed to address various challenges and optimize production and operational processes in the fields of agriculture, food, and bio-system engineering [1]. The Internet has become an essential tool for farmers, with 81% of them using it daily for agricultural activities such as accessing weather information, professional news, and banking services. Farm-related data constitutes 40% of the content consulted by farmers. Modern technologies, including smartphone applications, moisture sensors, thermal cameras, and drones, have seamlessly integrated into farm operations. The increasing data collected by these devices has led to an agricultural model that heavily relies on new technologies. However, effective management of this influx of information is necessary to derive valuable insights [48]. In the agricultural industry, Artificial Intelligence technologies are being adopted to promote healthier crop yields, pest control, monitoring of soil and growing conditions, data organization for farmers, workload assistance, and overall improvement across various tasks in the food supply chain [29].

The Figure 1.9 shows different fields of Artificial Intelligence in agriculture those:

- Crop yield prediction and price forecasts.
- Intelligent spraying.
- predictive insights.
- Agriculture robots.
- Crop and soil monitoring.
- disease diagnosis.

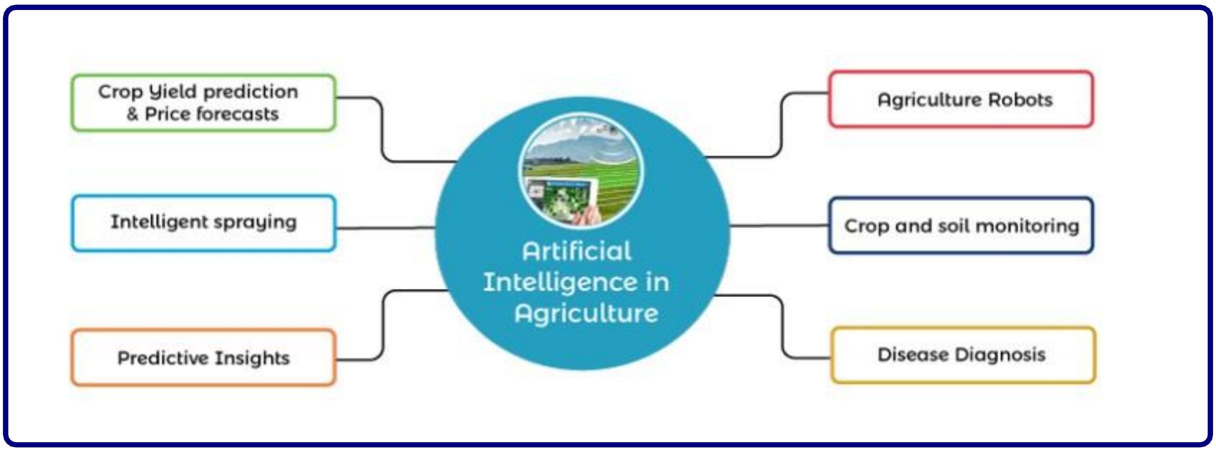


Figure 1.9: Artificial Intelligence in Agriculture

1.10 Deep learning in agriculture

Deep learning, a cutting-edge technology for image processing and data analysis, has emerged as a powerful tool with vast potential across various industries, including agriculture [2]. It utilizes nonlinear transformations to model complex patterns and extract high-level abstractions from raw data, a process known as feature learning. Deep learning enables the automatic extraction of features from lower-level components, resulting in higher-level components with more meaningful representations

[12]. In agriculture, recurrent neural networks (RNNs) and convolutional neural networks (CNNs) are commonly employed deep learning architectures. Deep learning has become increasingly crucial in agriculture as it aids farmers and researchers in optimizing crop yields, monitoring plant health, and detecting diseases and pests. Its application in the agricultural sector holds great promise for advancing productivity and sustainability in farming practices.

The Figure 1.10 [12] shows different fields of deep learning in agriculture those:

- Fruits counting
- Weed detection
- Soil management
- Water management
- Crop management
- yield prediction
- Disease detection

Figure 1.10: Deep learning applications in agriculture

1.11 Deep learning for WaterManagement

Water is an essential natural resource for agriculture that needs recycling for the continued and sustainable development of agriculture. Therefore, the agriculture field needs a DL technique to protect agriculture from water pollution. Traditional irrigation methods waste water due to excessive water use and unplanned water management.

Therefore, the authors provided an integrative approach that uses DL methods to improve the irrigation system in India's agriculture. The system consists of sensors that detect the soil's humidity and predict the irrigation needs of the soil. Evapotranspiration assessment uses DL techniques to predict upcoming water needs and to provide clues that can be helpful for real-time irrigation management. Thus, DL techniques can help farmers precisely manage their irrigation systems[12].

1.11.1 An Overview of few Deep Learning Algorithms Used in Smart Agriculture

Artificial Neural Networks (ANN)

Artificial Neural Networks (ANN) are a fundamental component of machine learning and have been extensively applied in various domains. ANN models are inspired by the structure and function of biological neurons and are designed to recognize patterns and make predictions based on input data. They consist of interconnected layers of artificial neurons that process and transmit information. One notable reference in this field is the book "Deep Learning" by Ian Goodfellow, Yoshua Bengio, and Aaron Courville [34], which provides a comprehensive overview of neural networks and their applications. Additionally, the paper "A Few Useful Things to Know About Machine Learning" by Pedro Domingos [22] offers insights into the key concepts and practical considerations of neural networks. Through the utilization of ANN, complex patterns and relationships can be learned, enabling tasks such as image recognition, natural language processing, and predictive analytics.

Recurrent Neural Networks (RNN)

Recurrent Neural Networks (RNN) are a specialized type of artificial neural network designed to handle sequential data, making them particularly effective for tasks involving time series analysis, speech recognition, and language modeling. Unlike traditional feedforward neural networks, RNNs introduce feedback connections, allowing information to persist and be shared across different time steps. A seminal paper in this area is "Understanding the Difficulty of Training Deep Feedforward Neural Networks" by Xavier Glorot and Yoshua Bengio [43], which discusses the challenges and solutions in training deep neural networks, including RNNs. Another notable reference is the paper "On the Difficulty of Training Recurrent Neural Networks" by Razvan Pascanu et al. [17], which delves into the theoretical aspects and optimization techniques for RNNs. By incorporating memory and context through recurrent connections, RNNs can capture long-term dependencies and generate meaningful predictions from sequential data.

Long Short-Term Memory (LSTM)

Long Short-Term Memory (LSTM) is a specific type of recurrent neural network architecture that addresses the vanishing gradient problem, allowing for more effective learning and retention of information over long sequences. LSTM networks are capable of capturing and remembering relevant information while selectively forgetting unnecessary details. The paper "Long Short-Term Memory" by Sepp Hochreiter and Jürgen Schmidhuber [26] presents the LSTM architecture and its advantages over traditional RNNs. Another reference, "Sequence to Sequence Learning with Neural Networks" by Ilya Sutskever et al. [27], explores the application of LSTM in machine translation tasks. By incorporating memory cells, input, output, and forget gates, LSTM networks can model complex sequences and perform tasks such as speech recognition, sentiment analysis, and time series forecasting.

1.12 Related Work

In this part, we will talk about some of the recent works and technologies that have been used for smart watering. By looking at what others have done before, we can benefit from their ideas and findings. From a technical standpoint, we will mention three works that are similar to ours and share similar goals.

1.12.1 Real Time Monitoring and Irrigation Control Using the WebSocket Protocol

The objective of this study is to develop a smart agriculture system using the Internet of Things (IoT) and the WebSocket protocol. The researchers aim to enable real-time communication for IoT devices by employing the WebSocket protocol, which offers universal compatibility. The study focuses on monitoring temperature and soil humidity while controlling the irrigation sluice in a rice field. Hardware components such as Arduino, Ethernet Shield, soil moisture sensor, and ultrasonic sensor are utilized, along with software components like Node.js and Socket.IO for WebSocket communication. The study also addresses energy consumption limitations and explores cloud storage for historical data in agriculture fields. By leveraging these technologies, the researchers aim to optimize resource usage, offer intelligent services, and enable real-time cloud connectivity for efficient smart agriculture practices[45].

The Functional flowchart used in this study :

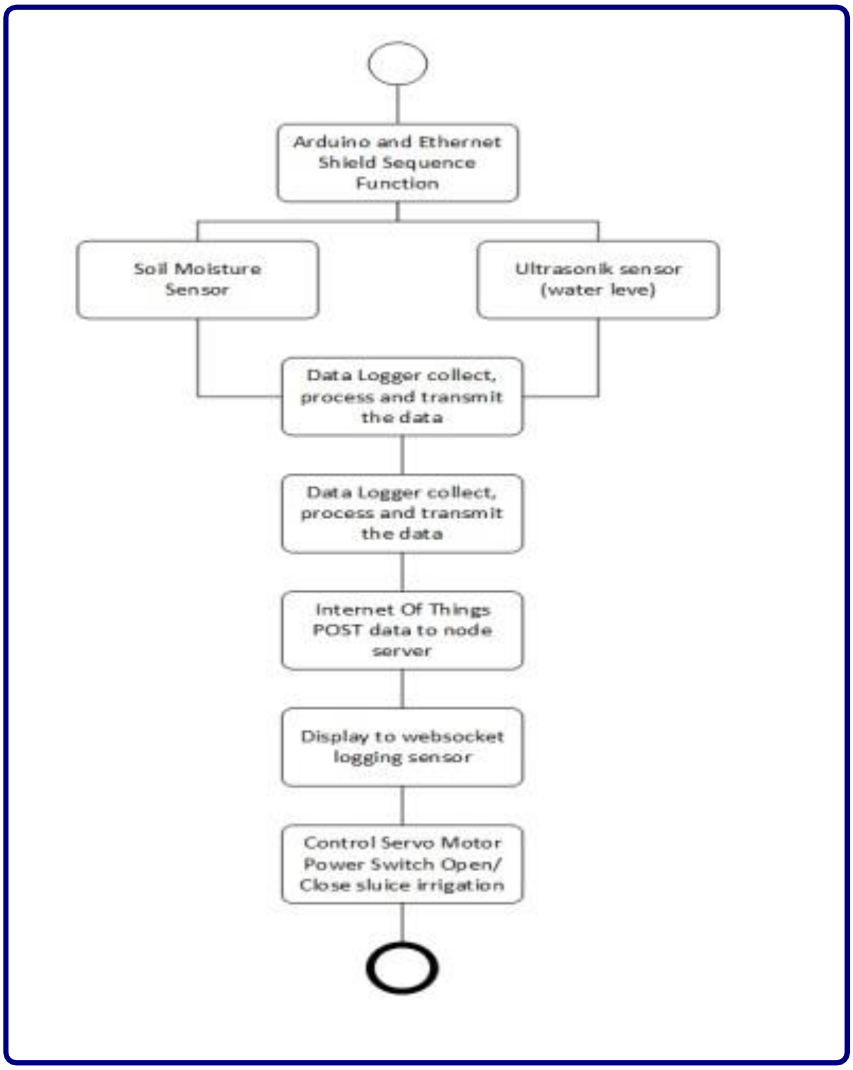


Figure 1.11: Functional flowchart

1.12.2 An Internet of Things and Cloud-Based Smart Irrigation System

The objective of the study is to present a cloud-based and IoT-based irrigation system that utilizes sensors for real-time data collection, cloud storage, and mobile applications for control and command. The authors aim to address the challenges of scientific irrigation scheduling by monitoring soil moisture and humidity levels near the root zone and initiating irrigation based on predefined parameters specific to the plant, soil type, and climate. The importance of this research lies in optimizing water usage and providing high-quality water to the root zone, ensuring efficient plant growth and development. By integrating cloud technology and IoT devices, the system offers a smart and automated approach to irrigation, enabling farmers to remotely monitor and control the irrigation process based on real-time data[30].

Diagram used in this study showing how an irrigation system using IoT and a certified cloud platform.

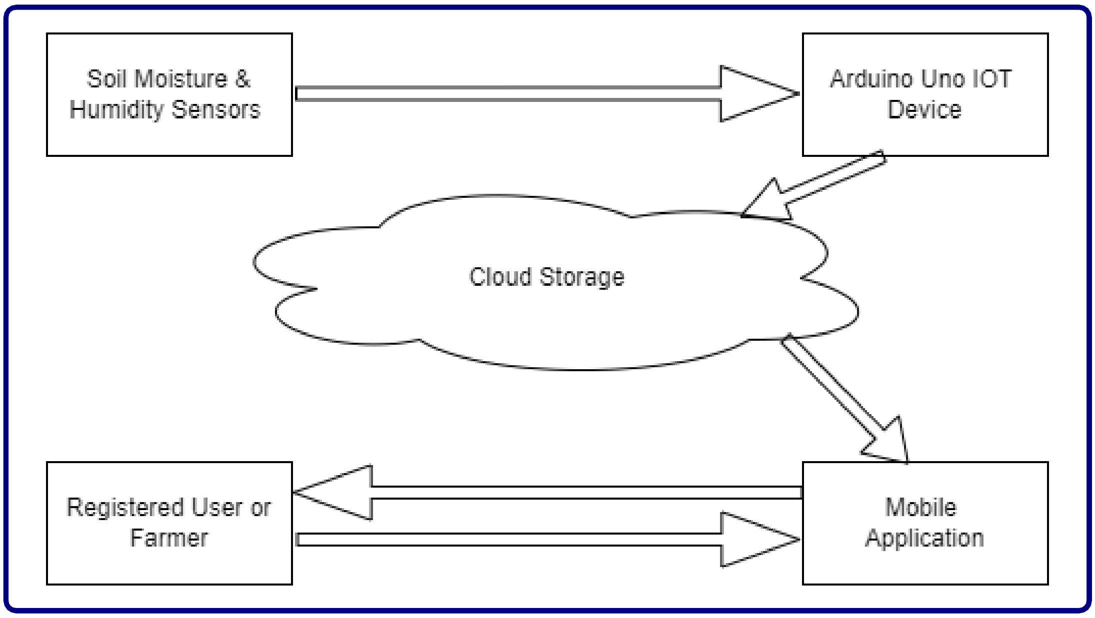


Figure 1.12: IoT and cloud Based Irrigation System

1.12.3 An IoT based smart irrigation management system using Machine learning and open source technologies

The objective of this study is to develop an IoT-based smart irrigation system using open-source technologies and a hybrid machine learning approach. The goal is to optimize water-resource utilization, improve irrigation efficiency, and reduce water wastage in precision farming. By integrating sensors and weather forecast data, the system aims to accurately predict soil moisture and provide real-time insights for irrigation management. This study is important in addressing global water scarcity challenges, promoting sustainability in agriculture, and enhancing crop productivity through informed irrigation practices. The use of open-source technologies makes the system cost-effective and accessible, contributing to the advancement of smart farming practices and efficient water usage [21].

The figure below illustrates the architecture utilized in this study:

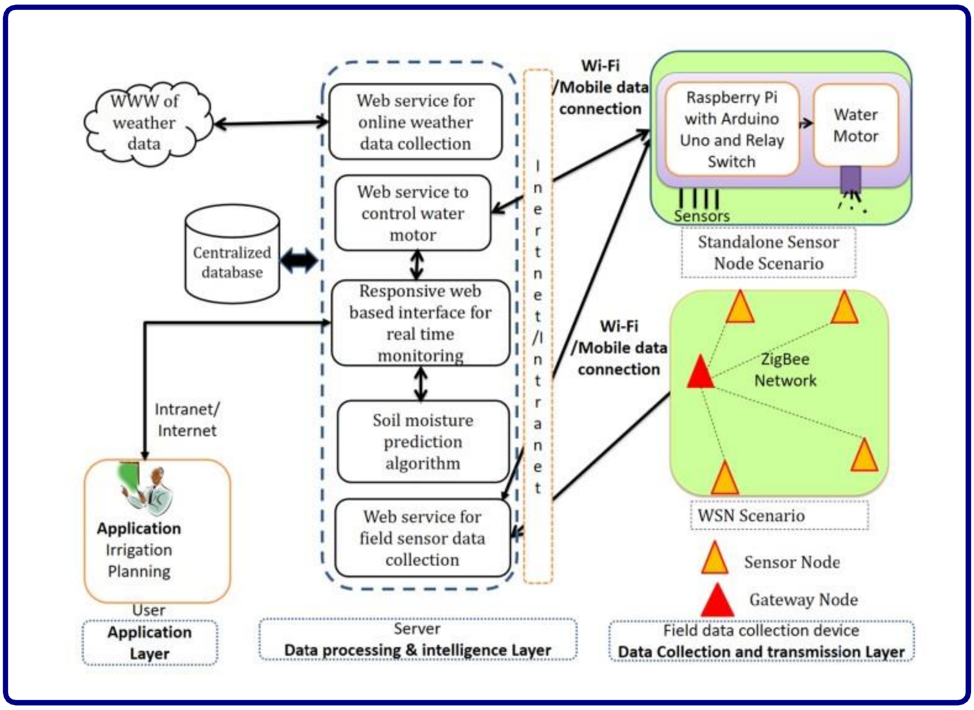


Figure 1.13: Architecture utilized in the study.

1.13 Analyse

Article Title	Focus	Technologies Used
Real Time Monitoring and Irrigation Control Using the Websocket Protocol[45]	Real-time monitoring and control of irrigation systems	Websocket protocol, IoT devices, sensors, web application
An Internet of Things and Cloud-Based Smart Irrigation System[30]	Smart irrigation system using IoT and cloud technologies	IoT devices, cloud computing, wireless communication, sensors
An IoT based smart irrigation management system using Machine learning and open source technologies[21]	Smart irrigation management using IoT and machine learning	IoT devices, machine learning algorithms, open source technologies

Table 1.1: Article Title, Focus, and Technologies Used

1.14 Contribution

Based on the mentioned studies "real-time monitoring and irrigation control using the WebSocket protocol," "IoT and cloud-based smart irrigation systems," and "IoT-based smart irrigation management systems utilizing machine learning and open-source technologies," we want to create a prototype of a smart irrigation system. The present system will use advanced communication protocols for real-time data acquisition and precise control. By combining IoT technology and cloud servers, we will collect and analyze scalable data for effective irrigation management. Using historical and real-time data, we will optimize irrigation schedules to conserve water and promote sustainable agriculture. Through seamless data exchange and coordination, we aim to overcome the limitations of traditional methods and leverage emerging technologies. The system will be controlled via a web application.

1.15 Conclusion

In this chapter, the utilization of IoT and Artificial Intelligence in smart irrigation systems to enhance efficiency and automation is explored. Different irrigation techniques and methods are explained, emphasizing the role played by IoT technology in monitoring and predicting crucial factors such as water quality, temperature, moisture, and soil conditions. These factors significantly influence the quality and yield of crops. The potential of IoT-based smart irrigation in improving traditional irrigation methods is demonstrated in the conclusion. In the next chapter, further details about the system will be provided, encompassing the methods and tools employed, as well as a comprehensive discussion on both the software and hardware components of the project, including the utilized architecture.

Chapter 2

System design

2.1 Introduction

The design phase is a critical step in the development of the connected irrigation system. During this phase, we will create a detailed plan that outlines how the system will be structured and how its various components will interact. This plan will serve as a guide for the implementation process, ensuring that the technical choices made are aligned with the desired outcomes. By carefully designing the architecture of the system, we can ensure that it is robust, efficient, and capable of effectively monitoring and managing irrigation. This design phase will lay the foundation for a successful implementation and maintenance of the connected irrigation system.

2.2 Objectives and principle of the proposed solution

Smart farming depends on technology and investment in agriculture and crop production. The biggest challenges include irrigation scheduling, which results in water waste, and the management of irrigation, especially for large farms. These issues require innovative solutions that incorporate technological advancements and sustainable farming practices to improve productivity and efficiency.

2.2.1 Objective of the project

The main objective of our system is to provide a solution to the problems associated with traditional irrigation practices in agriculture by:

- **Optimize water usage:** A smart irrigation system aims to reduce water waste by delivering the right amount of water at the right time to the plants.
- **Increase plant health:** By providing the right amount of water to plants, this system can help promote healthier and more robust plant growth.
- **Save time and effort:** The automatic irrigation can also help to simplify the irrigation process by automating the watering schedule, thereby reducing the need for manual effort and time investment from the farmer.
- **Improve efficiency:**By using sensors and data analysis, can optimize watering schedules and reduce water.

2.2.2 Principle of the proposed solution

The new proposed system is composed of three parts (Figure 2.1):

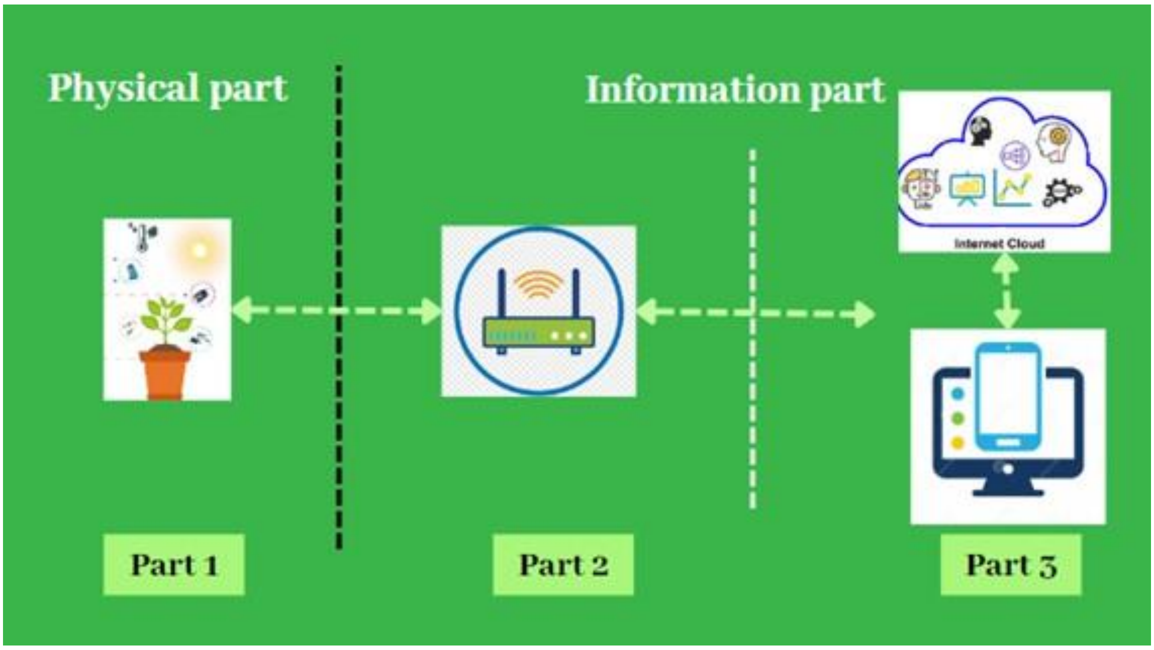


Figure 2.1: System element.

The principle of the proposed solution is to create an irrigation system that consists of three main components, each with a specific function:

- Part 1:The first component collects physical data from the farm and acts as a physical interface between the system and the farm, allowing for real actions to be taken based on system commands.
- Part 2:The second component serves as a gateway between the first and third components, facilitating the exchange of data and commands between them and sending data from the farm to the cloud.
- Part 3:The third component is a dynamic website hosted on the cloud, which provides real-time monitoring of the plant and generates statistics on climatic factors, allowing for remote manual irrigation management.

In summary, the proposed solution operates by collecting physical data from the farm, sending it to the cloud for analysis and decision-making, and providing real-time monitoring and management through a dynamic website.

2.3 General system structure

The new general system structure (figure 2.2) give as a complete picture of how different parts of a system fit together and collaborate to achieve its goals. It serves as a base for creating, building, and managing complex systems in various fields software, information, and control systems.

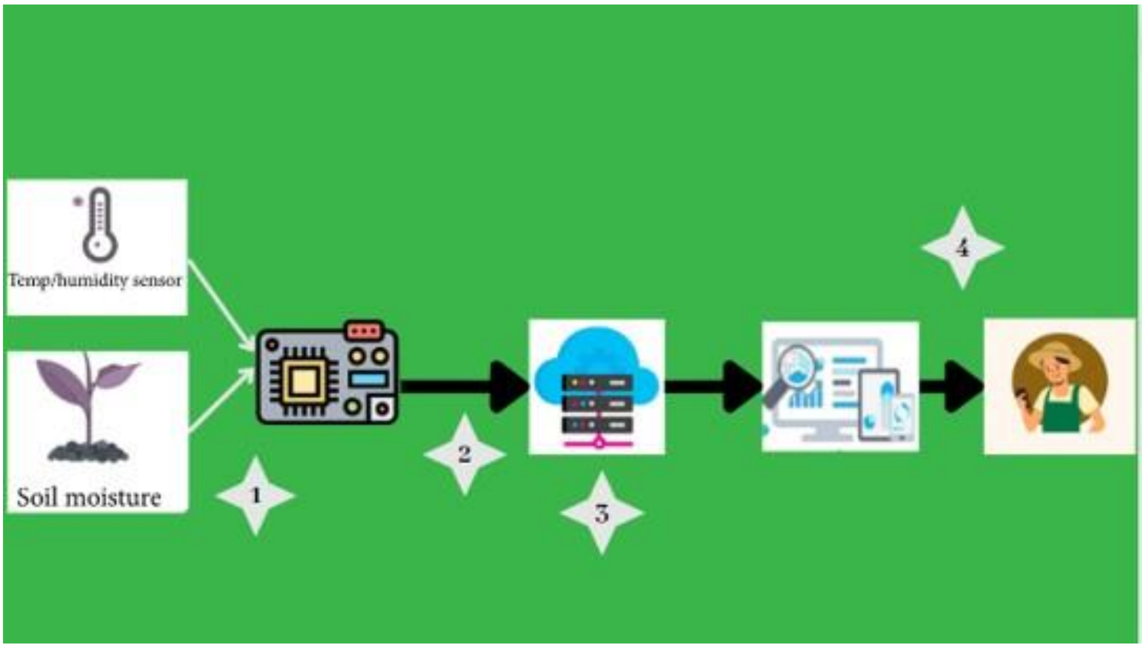


Figure 2.2: General system structure.

1. **Sensing Layer:** This layer is responsible for collecting data from various sensors, such as temperature, humidity, and soil moisture sensors. In your NodeMCU, you connect and read data from these sensors directly. The NodeMCU processes the sensor data and prepares it for further processing.

2. **Transport layer:** The transport layer handles the communication between the NodeMCU and the cloud-based server. In your system, the MQTT (Message Queuing Telemetry Transport) protocol is used to send the sensor data from the NodeMCU to the cloud server for further processing and storage.

3. **Processing layer:** This layer involves the processing of the collected data from the sensing layer. It includes tasks such as data cleaning, normalization, and feature engineering. In a smart irrigation system, this layer uses machine learning algorithms such as LSTM to make predictions about the optimal time for irrigation. The algorithm will be discussed in detail in the "Used Algorithm" section.

4. **Cloud layer:** This layer consists of a cloud-based server that receives and stores the collected sensor data from the NodeMCU. The server provides data storage, backup, and additional services such as data security and scalability. The server acts as a central repository for the collected data, making it accessible for further analysis or visualization.

5. **Application layer:** The application layer encompasses the user interface (UI) and the irrigation control system. In your system, the NodeMCU can utilize algorithms to analyze the collected sensor data locally and make decisions regarding irrigation control. The UI, which can be a dashboard or a web-based interface, allows the end-user to view the system's status, adjust irrigation settings, and receive notifications or alerts.

2.4 The communication protocol used

The protocol used in this system is illustrated , wherein the MQTT (Message Queuing Telemetry Transport) protocol has been chosen to be utilized for streamlined and effective communication.

The Figure 2.8 shows The communication protocol used:

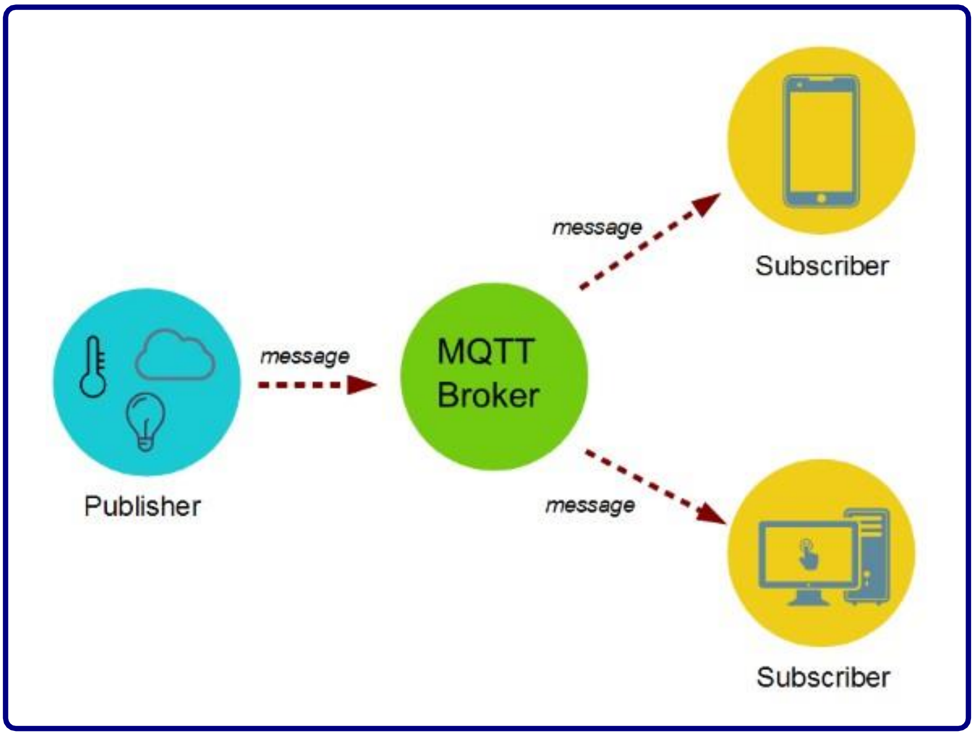


Figure 2.3: MQTT (Publish/Subscribe)protocol used.

Motivation for Using This Protocol: MQTT

- Simplified communication: Offers a lightweight and efficient protocol that simplifies data exchange between sensors, controllers, and the irrigation system, making communication more straightforward.
- Real-time updates: MQTT's publish/subscribe model enables instant communication, ensuring timely updates. This real-time capability allows for quick decision-making within our system.
- Scalability: Easily scalable, accommodating future growth and varying device counts. This flexibility is crucial as our system expands and evolves over time.
- Reliable message delivery: Guarantees reliable message delivery, minimizing data loss and ensuring that messages reach their intended destinations accurately.
- Robust performance: Demonstrates robust performance, even in challenging network conditions. This ensures that our smart irrigation system operates efficiently and consistently.

2.5 System architecture of the project

This system architecture serves as a guiding framework, illustrating how all the different components of the system collaborate and work.

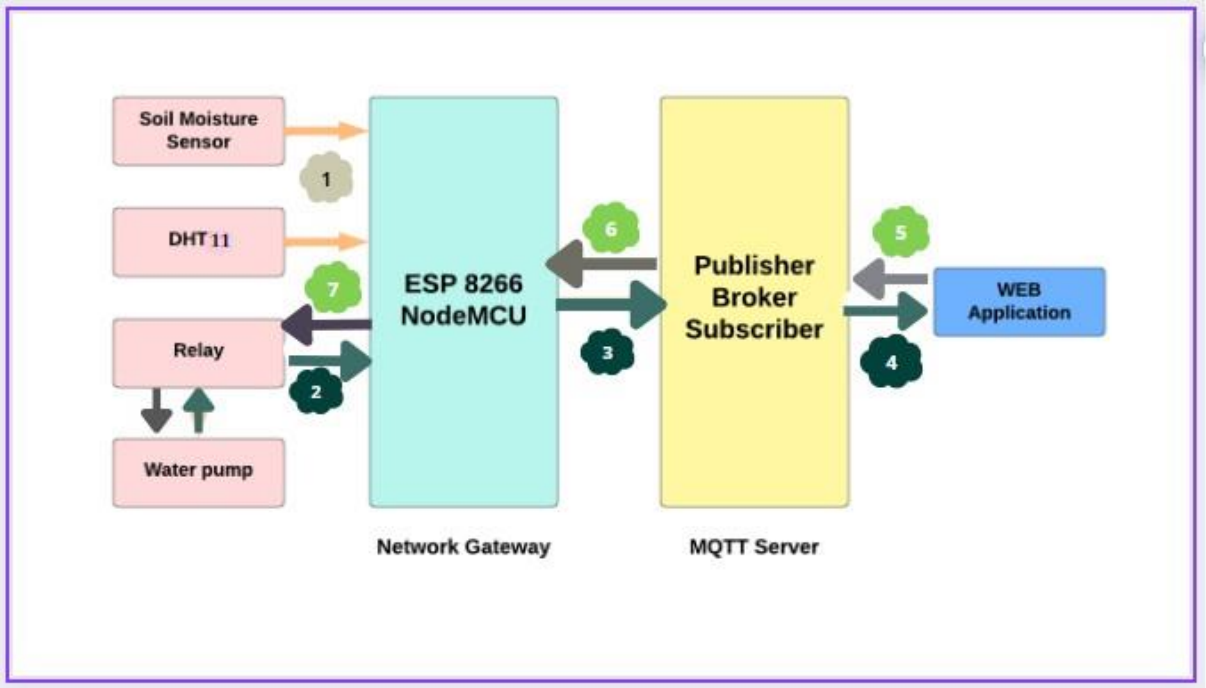


Figure 2.4: System architecture.

1. Flow from Soil Moisture Sensor and DHT11 to ESP8266:
   - The soil moisture sensor and DHT11 sensor collect data on soil moisture, temperature, and humidity.
   - The ESP8266 microcontroller reads the sensor data.
   - The ESP8266 receives data from the soil moisture sensor and DHT11 sensor.
2. Flow from Pump to Relay Module to ESP8266:
   - The pump is connected to a relay module.
   - The relay module is controlled by the ESP8266 microcontroller.
   - The ESP8266 sends control signals to the relay module to activate or deactivate the pump.
3. Flow from ESP8266 to HiveMQ Broker:
   - The ESP8266 acts as a MQTT client and connects to the HiveMQ MQTT broker.
   - The ESP8266 publishes the sensor data (soil moisture, temperature, humidity) to specific topics on the MQTT broker.
   - The MQTT broker receives and stores the published sensor data.
   - The ESP8266 subscribes to specific topics on the MQTT broker.

- The ESP8266 receives control commands or feedback messages from the web application through the MQTT broker.

4. Flow from Web Application to HiveMQ Broker:
   - The web application sends control commands or instructions to the MQTT broker (HiveMQ).
   - The MQTT broker receives and forwards the control commands to the appropriate subscribed devices, including the ESP8266.
5. Flow from ESP8266 to Relay(6) Module to Pump (7) :
   - The ESP8266 receives control commands from the MQTT broker.
   - The ESP8266 sends control signals to the relay module based on the received commands.
   - The relay module controls the pump based on the signals from the ESP8266, activating or deactivating the pump accordingly.

2.6 MQTT-based Automated irrigation System algorithms

In this part, the important algorithms that were used in my work are included: **Algorithm 1: Turn pump on**

Pump Activation that involves turning on the pump to allow water flow for irrigating the plant.

Algorithm 1: Turn Pump On

```
1: function TurnPumpOn
2:     SET pump ON
3: end function
```

Algorithm 2: Turn pump off

Pump Deactivation that is the process of turning off the pump to stop the water flow and end the irrigation process.

Algorithm 2: Turn Pump Off

```
1: function TurnPumpOff
2:     SET pump OFF
3: end function
```

Algorithm 3: Subscribe

The Subscribe algorithm enables the client to receive messages in a bidirectional MQTT communication system.

Algorithm 3: SUBSCRIBE

```
function SUBSCRIBE(client, topic)
    % Declare a function named SUBSCRIBE that takes two parameters: the MQTT client object and the topic to subscribe to.
    SET client.callback(mqtt_callback)
    % Set the callback function for the MQTT client object, specifying the function that will be called when a message is received.
    client.connect()
    % Establish a connection between the MQTT client and the MQTT broker.
    encoded_topic = topic.encode()
    % Encode the topic string into a format suitable for MQTT communication.
    client.subscribe(encoded_topic)
    % Instruct the MQTT client to subscribe to the encoded topic, enabling it to receive messages published on that topic.
    return client
    % Return the MQTT client object, allowing it to be used for further operations.
end function
```

Algorithm 4: ConnectMQTT

The ConnectMQTT algorithm establishes bidirectional communication by connecting the client to the MQTT broker

Algorithm 4: ConnectMQTT

```
function CONNECTMQTT
    % Declare ConnectMQTT function
    SET client = MQTTClient(
        client_id=b"123587495aa",
        server=b"77af8...hivemq.cloud",
        port=0,
        user=b"Test123@",
        password=b"Test123@",
        keepalive=7200,
        ssl=True,
        ssl_params={"server_hostname": "77aff..b.s2.eu.hivemq.cloud"},
    )
    client.connect()
    % Establish connection between MQTT client and broker
    return client
    % Return MQTT client object for further operations
end function
```

Algorithm 5: Publish

The Publish algorithm facilitates the transmission of data from the client to the MQTT broker.

Algorithm 5: PUBLISH

```
function PUBLISH(client, topic, value)
    % Declare PUBLISH function with parameters: client (MQTT client object), topic (topic to publish to), and value (data to be published)
    SET topic = id + "/" + topic
    % Concatenate unique identifier with topic
    client.publish(topic, value)
    % Publish the value to the specified topic via the MQTT client
end function
```

Algorithm 6: Verify Pump

The Verify Pump algorithm is used to check the status of the pump and listen for changes in the pump status.

Algorithm 6: VERIFY_PUMP

```
function VERIFY_PUMP
    % Function to verify the pump status
    SET pump_listening_time = 10
    % Set the time duration for listening to pump status
    SET client = connectMQTT()
    % Connect to MQTT broker
    SET topic = id + "/pump"
    % Construct the topic for pump status
    CALL subscribe(client, topic)
    % Subscribe to the pump status topic
    SET start_time = time.time()
    % Record the start time
    while True do
        % Start an infinite loop
        client.check_msg()
        % Check for incoming messages
        if time.time() - start_time >= pump_listening_time then
            % If the listening time has exceeded
            BREAK
            % Exit the loop
        end if
    end while
end function
```

Algorithm 7: MQTT Callback

The MQTT Callback algorithm is responsible for handling incoming messages from subscribed topics and performing actions based on the message content.

Algorithm 7: MQTT_CALLBACK

```
function MQTT_CALLBACK(topic, msg)
    % Function to handle MQTT messages
    PRINT("Received message from topic:", topic)
    % Print the received message topic
    PRINT("Message:", msg.decode())
    % Print the decoded message
    SET etat = etat_pump()
    % Get the current pump status
    PRINT("etat:", etat)
    % Print the current pump status
    if etat == "on" AND "off" in msg.decode() then
        % If the pump is currently on and the message contains "off"
        CALL turn_pump_off()
        % Turn off the pump
    end if
    if etat == "off" AND "on" in msg.decode() then
        % If the pump is currently off and the message contains "on"
        CALL turn_pump_on()
        % Turn on the pump
    end if
end function
```

Algorithm 8: Read Soil Moisture

The readSoilMoisture algorithm is used to measure and calculate the soil moisture level.

Algorithm 8: readSoilMoisture

```
function READSOILMOISTURE
    % Function to read soil moisture level
    SET adc = machine.ADC(0)
    % Initialize ADC (Analog-to-Digital Converter)
    SET SoilMoisture = adc.read()
    % Read the soil moisture value from ADC
    SET SoilMoisture = round(1 / (SoilMoisture / 1024) / 10 * 100, 2)
    % Convert the raw ADC value to a percentage value
    RETURN SoilMoisture
    % Return the calculated soil moisture level
end function
```

2.7 UML Diagram

In this section, the utilization of UML diagrams will be explored.

2.7.1 Use case diagram

In this section, we illustrate our platform functionalities which the user (Farmer) and manager can benefit from. Services and features are shown in the use case diagram below :

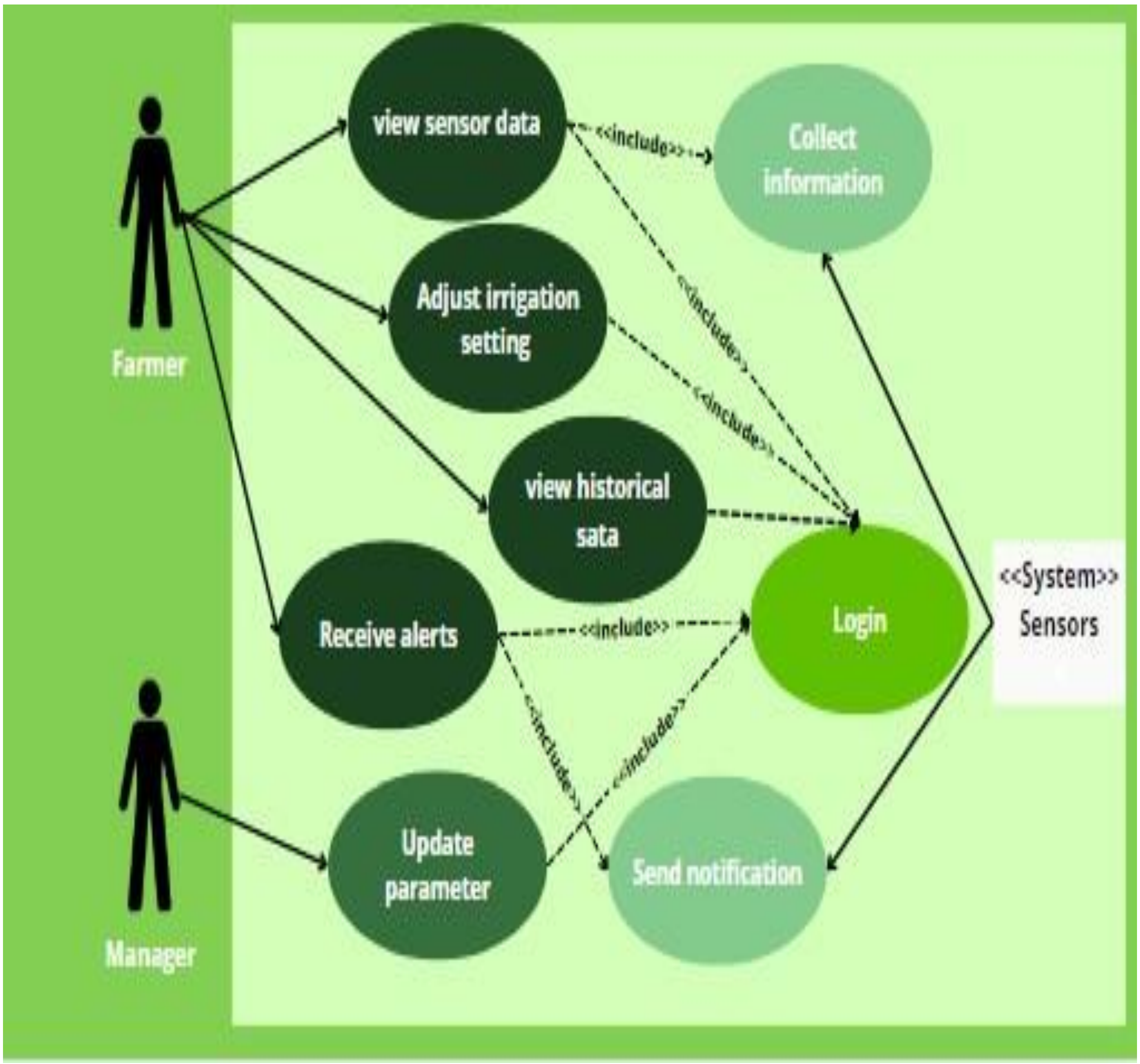


Figure 2.5: Use case diagram

Farmer

Functionality	Description
View Sensors Data	Users (Farmers) can access and view data collected by various sensors in the system. This includes data from soil moisture sensors, temperature sensors, and humidity sensors. They can monitor the current environmental conditions of their crops.
Adjust Irrigation Settings	Users (Farmers) have the ability to adjust the irrigation settings for their crops based on real-time weather and soil moisture data. They can modify parameters such as irrigation duration, frequency, and thresholds, ensuring optimal water management for their plants.
View Historical Data	Users (Farmers) can access and review the history of irrigation events. This includes information such as the start and stop times of irrigation cycles, as well as the amount of water used. It enables users to track irrigation patterns and monitor the water consumption of their crops.
Receive Alerts	Users (Farmers) can receive alerts and notifications from the system. They are notified when the soil moisture level is too low, indicating a need for irrigation. Additionally, they are alerted in case of any system errors or malfunctions, ensuring timely responses and interventions.

Table 2.1: Functionality description farmer

Manager

Functionality	Description
Update User Accounts	The manager has the capability to update user accounts. This functionality allows the manager to perform operations such as adding new user accounts, deleting existing user accounts, modifying user account details (e.g., username, email), and managing user permissions within the system.

Table 2.2: Functionality description manager

2.8 Class diagram

The key components of our system are represented by these classes, enabling user management, topic discussions, and notifications. Users of the system are represented by the "User" class, while the "Manager" class extends the "User" class and incorporates additional methods for managing farmers. The "Topics" class is responsible for handling topics or discussions pertaining to irrigation, whereas the "Notification" class is responsible for managing notifications sent to users. Together, these classes form the foundation of your application, fostering user interaction, facilitating communication, and promoting information sharing.

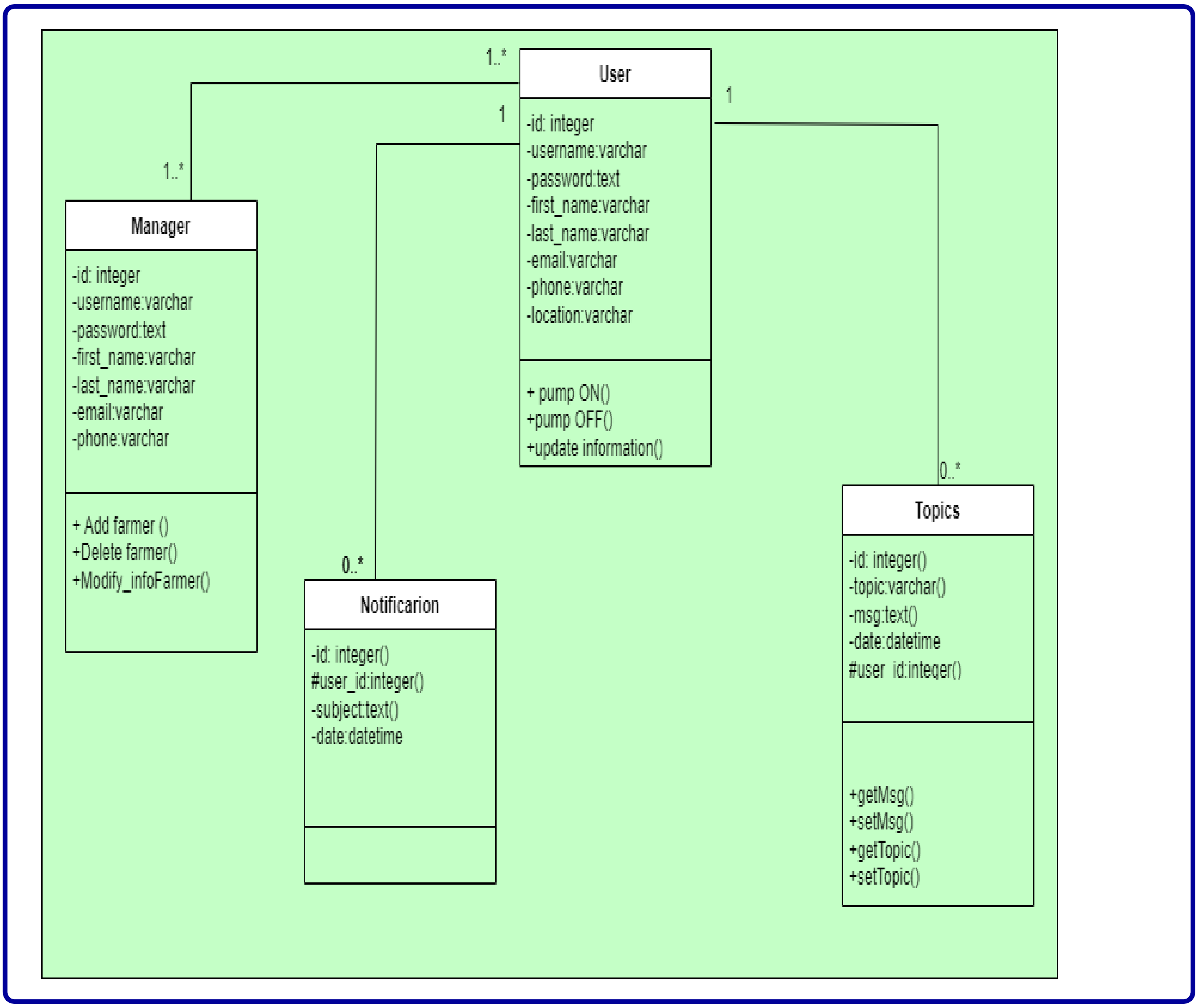


Figure 2.6: Class diagram.

2.9 Sequence diagram

this sequence describes the process of a farmer using a web page to control a remote pump:

2.9.1 Adjust pump state

- The farmer presses the "on/off" button on a web page.
- The web page publish the command after establishes a connection and subscription to an MQTT broker.
- The MQTT broker sends the command to a microcontroller (which is already connected and subscribed with MQTT broker)
- The microcontroller executes the command to operate the pump remotely.

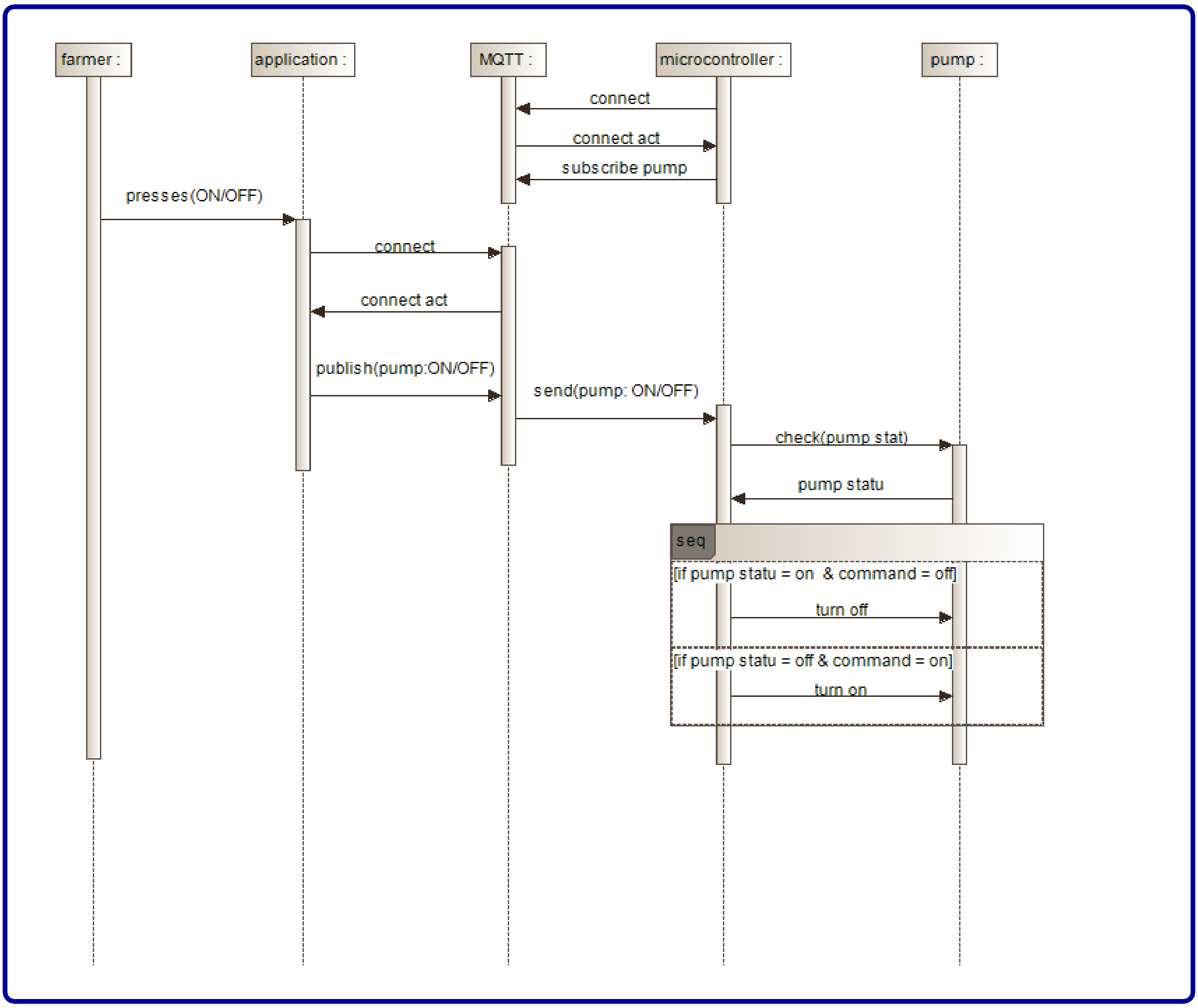


Figure 2.7: Adjust pump state

2.9.2 Storing data

When notifications need to be sent by the ESP8266, a connection is established, and the notification is transmitted by MQTT. Subsequently, MQTT forwards the notification to the server, which is already connected and shared. Upon receiving the notification from MQTT, the server sends an instruction to the database to store the notification.

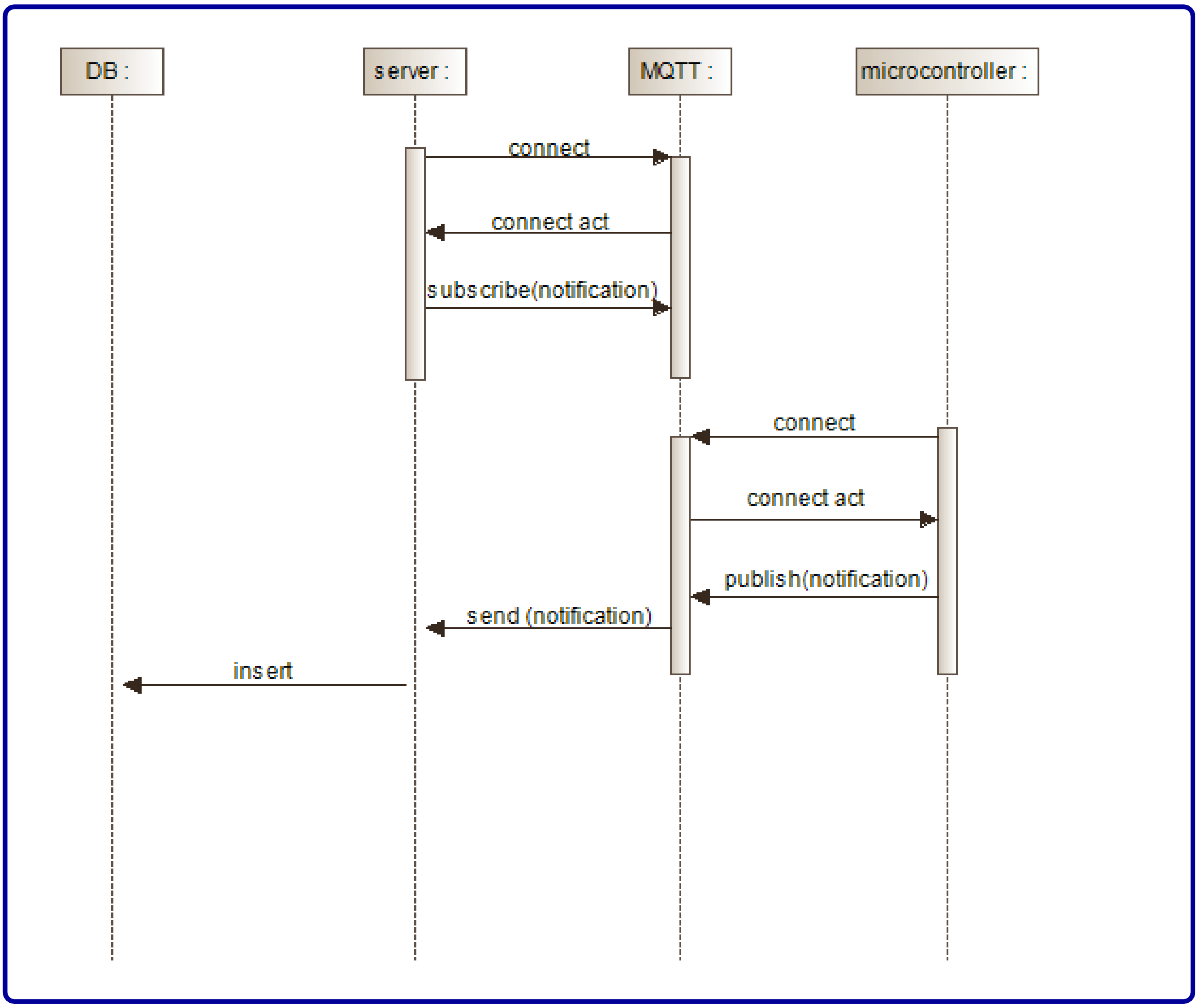


Figure 2.8: Storing data

2.10 Conclusion

In this chapter, an overview of our system architecture and the employed communication protocol has been provided. Our system is comprised of two key components: a Hardware section utilizing the Internet of Things (IoT) and a Software section consisting of a Web Platform and the chosen communication protocol. The challenge of water conservation is aimed to be addressed by integrating these technologies to offer an effective solution. In the following chapter, the implementation details of our system will be delved into, and the results achieved will be presented.

Chapter 3

System implementation

3.1 Introduction

In this chapter, the hardware tools and software environments employed for developing our system are introduced. Additionally, the various platforms used to implement different parts of the system are outlined, and screenshots of the results obtained from our work are included.

3.2 Physical interface for capture and actuation

The circuit diagram illustrates an IoT-based smart irrigation system that utilizes a DHT11 sensor, soil moisture sensor, and an ESP8266 NodeMCU. The system automatically monitors soil moisture levels and controls water flow for efficient irrigation. It incorporates a relay module to control the water pump/valve and an LED to indicate system activity. The NodeMCU enables internet connectivity for remote monitoring and control.

3.2.1 Physical diagram

A physical diagram is a representation that visually illustrates the physical components and connections of a system. In the case of a this system incorporating ESP8266, soil moisture sensor (such as DHT11), and a pump, the physical diagram would provide a clear overview of how these components are interconnected.

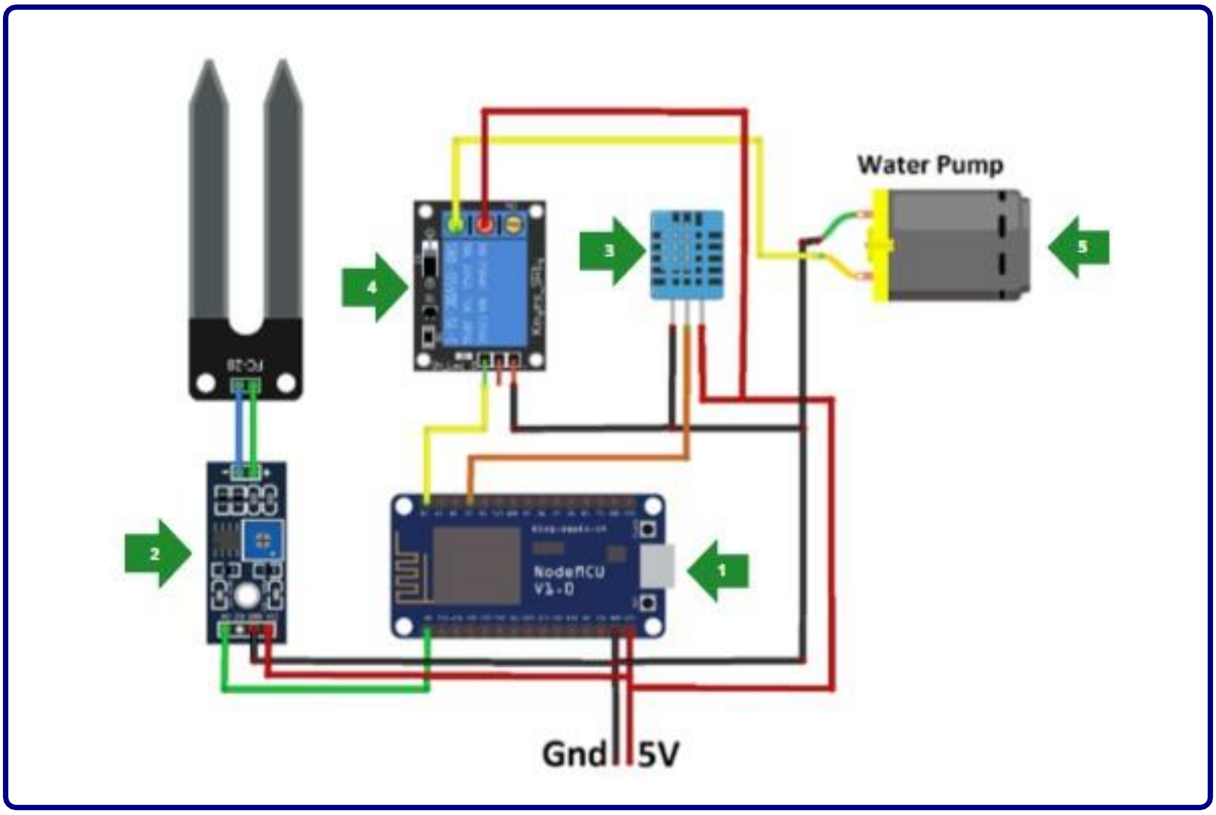


Figure 3.1: Physical global diagram

3.2.2 Components

In this part, the components numbered in the architecture 3.1 will be explained.

1. **NodeMCU ESP8266**

   The NodeMCU in a smart irrigation system collects sensor data, analyzes it, and makes decisions using predefined algorithms. It enables remote access, allowing users to monitor and adjust irrigation settings from anywhere. The integration of NodeMCU with other IoT devices and platforms enhances the scalability and potential of the smart irrigation system within a broader IoT ecosystem.elprocu

Figure 3.2: NodeMCU esp8266 board

2. **Sensors**

   - A soil moisture sensor **(2)** is a device that tells us how much water is in the soil. It's like a special tool that helps us understand if the soil is dry or wet. Farmers and gardeners use it to know when and how much to water their plants. The sensor is put into the ground, and it measures things like electricity or how well the soil holds water. This information is shown on a device or computer, so people can see if the soil has enough water for the plants to grow well. By using the sensor, we can avoid wasting water and make sure our plants are healthy[18].

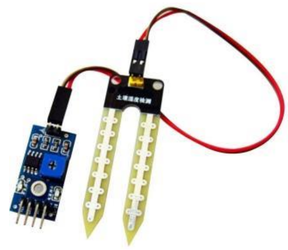


Figure 3.3: soil moisture sensor

- The DHT11 **(3)** is a popular sensor that tells us the temperature and humidity in a simple way. It is used in many things like checking the weather, controlling how warm or cool a room is, and making homes smarter. The DHT11 has a special part that measures temperature and another part that measures humidity. It can give us accurate readings by converting the measurements into digital signals. With the DHT11, we can know if it's hot or cold and how much moisture is in the air around us[35].

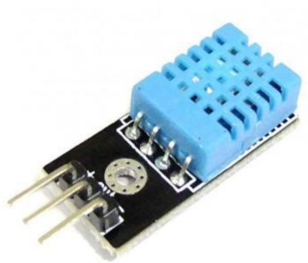


Figure 3.4: DHT11 sensor

- A relay module **(4)** is an electronic component used to remotely control an electrical load by utilizing a low-power electrical signal. It consists of an electric coil that can be activated by a low-intensity electrical current, along with a mechanical contact that can be opened or closed in response to the electrical signal. The relay module is commonly used to control electrical devices such as lights, motors, and pumps, as well as to protect electronic circuits by cutting off the power supply in cases of overload or short circuit. It can be controlled in various ways, such as using a microcontroller or a manual switch.

Figure 3.5: Relay module

3. **Actuators**

- Water Pump **(5)** : The pump is responsible for drawing water from a water source, such as a well, reservoir, or water storage tank. It ensures a continuous and reliable supply of water for irrigation purposes. It helps conserve water, supports system expansion, and requires regular maintenance.

Figure 3.6: Water Pump

3.3 Software environment and language

In this section, the Software environment and programming language utilized in this system will be discussed.

3.3.1 Software environment

Initiated by the software environment

Visual Studio Code (Code Editor)

VS Code, developed by Microsoft, is a code editor designed to facilitate the editing of source code files. It is known for its lightweight nature and offers a wide range of extensions that enhance the development experience for programmers. We opted for VS Code for our project due to its versatility in supporting various programming languages such as Python, HTML, JavaScript, and CSS.

Thonny

is a code editor developed to supports various programming languages such as Python and is particularly suitable for programming microcontroller boards like NodeMCU. Thonny's user-friendly interface and features make it an excellent choice for writing and uploading code to NodeMCU devices. We chose Thonny for our project due to its ease of use and compatibility with multiple programming languages, including the one used for programming the NodeMCU.

HiveMQ

HiveMQ is an MQTT (Message Queuing Telemetry Transport) broker or server that facilitates efficient and reliable communication between devices and applications in the Internet of Things (IoT) ecosystem. It provides a messaging protocol that enables lightweight and real-time data exchange, making it suitable for IoT applications where low bandwidth and intermittent connectivity may be present. HiveMQTT acts as a central hub for publishing and subscribing to MQTT messages, allowing devices to communicate with each other and with backend systems. It offers features such as secure connections, message persistence, and scalability, making it a popular choice for building robust and scalable IoT solutions.

Languages used

- PHP, CSS, and HTML are three fundamental technologies used in web development. HTML (Hypertext Markup Language) provides the structure and content of a webpage, defining elements such as headings, paragraphs, images, and links. CSS (Cascading Style Sheets) is responsible for the visual presentation and layout of a webpage, allowing developers to define colors, fonts, spacing, and other design aspects. PHP (Hypertext Preprocessor) is a server-side scripting language that adds dynamic functionality to websites, enabling tasks like form processing, database interactions, and user authentication. Together, these technologies form the backbone of web development, with HTML for structure, CSS for styling, and PHP for dynamic functionality, facilitating the creation of interactive and visually appealing web pages and applications.

- Python is a versatile and high-level programming language known for its simplicity and readability. It emphasizes code readability and uses a clean syntax, making it easy for beginners to learn and understand. Python supports multiple programming paradigms, including procedural, object-oriented, and functional programming. It has a large and active community, offering extensive libraries and frameworks that enhance its capabilities for various domains such as web development, data analysis, artificial intelligence, and scientific computing. Python's versatility, wide range of applications, and user-friendly nature have contributed to its popularity among developers for both small-scale scripting tasks and large-scale software development projects.

3.4 Prototype development and Web application integration

initiated by the Prototype development

3.4.1 Prototype development

The prototype aims to demonstrate the functionality and effectiveness of an IoT-based irrigation system, where sensors are used to collect data, microcontroller are employed for control and automation, and a monitoring system enables real-time tracking and analysis.

The experimental plant utilized in our project plays a significant role in the prototype development. This plant is specifically chosen to serve as an integral part of the IoT-based irrigation system. It is equipped with a soil moisture sensor that accurately measures the moisture levels in the soil. This vital information provides crucial insights into the plant's water requirements. Additionally, a temperature/humidity sensor is strategically placed to continuously monitor the environmental conditions surrounding the plant. The combination of these sensors offers a comprehensive understanding of the plant's immediate surroundings. By collecting and analyzing this data, our prototype aims to optimize the irrigation process and ensure the plant receives adequate water and ideal environmental conditions for growth.

The figure 3.7 show the prototype that was used in our project.

Figure 3.7: The proposed prototype

3.4.2 Connecting IoT-based Irrigation Monitoring to Haivemq Cloud

The Haivemq interface is an essential component of our IoT-based irrigation monitoring system. It consists of a port and URL that enable seamless communication between our devices and the Haivemq cloud platform. It acts as a bridge, connecting our irrigation system to the cloud. Through this interface, we can access real-time data on soil moisture levels, weather conditions, and the status of our irrigation processes. This valuable information empowers us to make informed decisions, optimize water usage, and improve the overall efficiency of our irrigation system. By integrating with the Haivemq cloud, we can leverage its robust computing capabilities to enhance the monitoring and control of our irrigation processes. It simplifies the management of our system and allows us to remotely monitor and manage our irrigation operations from anywhere.

When pressed on 1 , going to the information about the cluster.(see figure 3.9)

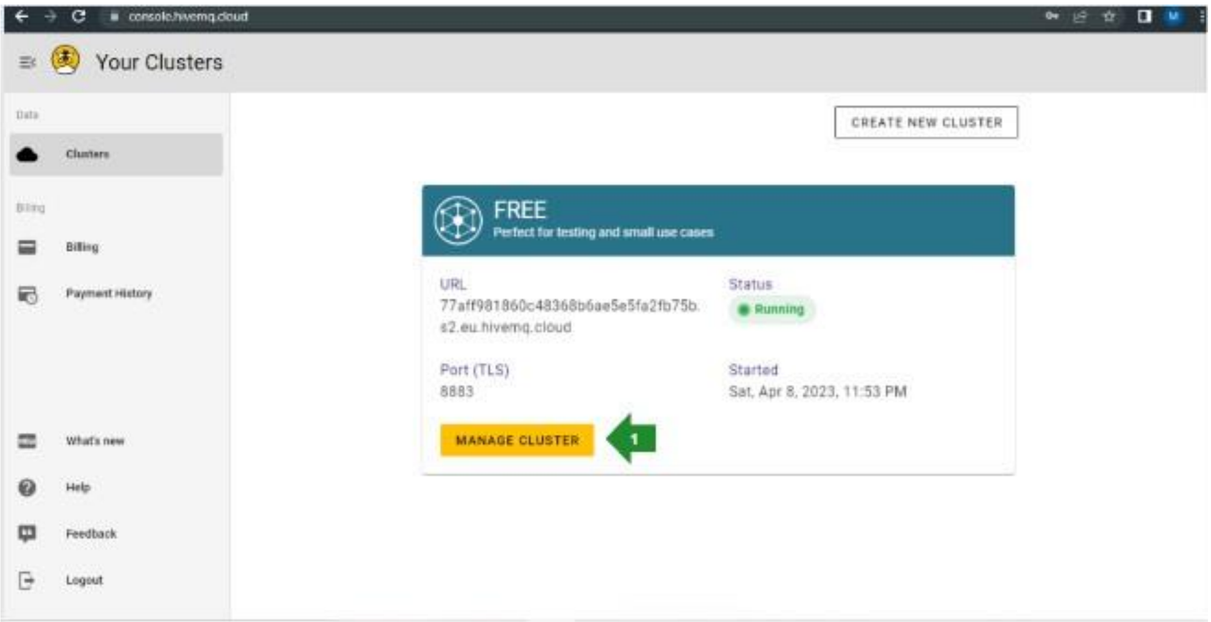


Figure 3.8: URL and port used

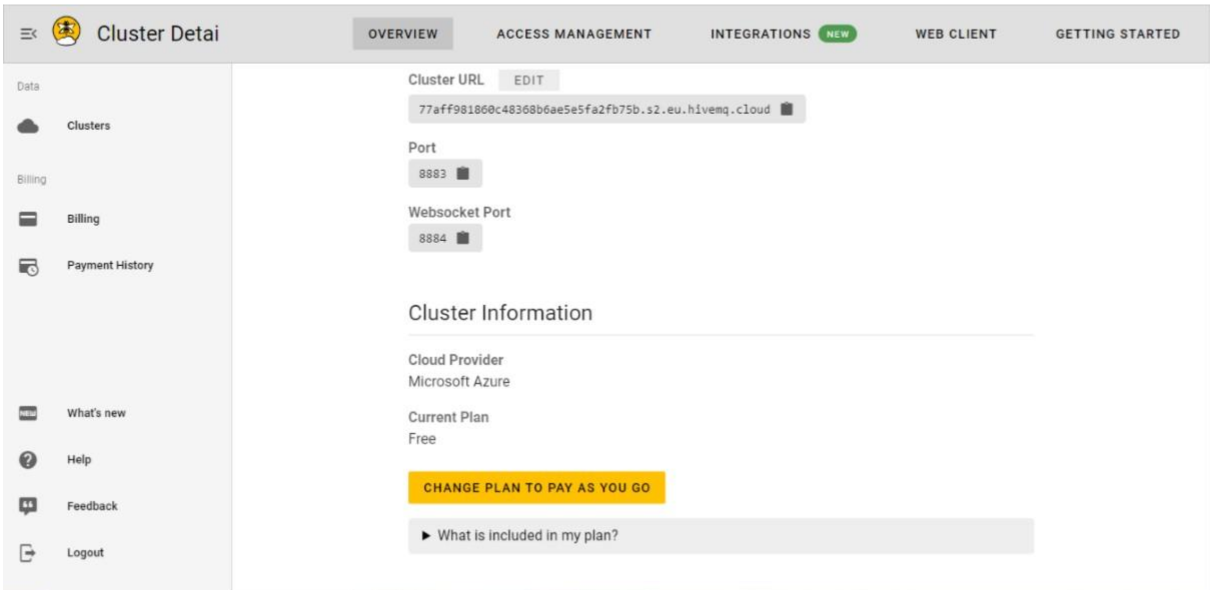


Figure 3.9: HaiveMQ information

3.4.3 Tables of our database

In this section, the tables of the database for our project that have been created will be presented.

logical data model:

- Manager (id_manager,username, password,firstName, lastName,email,phone)
- User(id_user,username,password,first_name,last_name,email,phone,location)
- Topics(id_topic,topic,msg,date,#id_user)
- Notification(id_note,message,date,#id_user)

#	Nom	Type	Interclassement	Attributs	Null	Valeur par défaut	Commentaires	Extra	Action
1	id	int(11)			Non	Aucun(e)		AUTO_INCREMENT	Modifier Supprimer Plus
2	username	varchar(45)	utf8mb4_general_ci		Non	Aucun(e)			Modifier Supprimer Plus
3	password	text	utf8mb4_general_ci		Non	Aucun(e)			Modifier Supprimer Plus
4	first_name	varchar(45)	utf8mb4_general_ci		Non	Aucun(e)			Modifier Supprimer Plus
5	last_name	varchar(45)	utf8mb4_general_ci		Non	Aucun(e)			Modifier Supprimer Plus
6	email	varchar(45)	utf8mb4_general_ci		Non	Aucun(e)			Modifier Supprimer Plus
7	phone	varchar(45)	utf8mb4_general_ci		Non	Aucun(e)			Modifier Supprimer Plus

Figure 3.10: Manager table

#	Nom	Type	Interclassement	Attributs	Null	Valeur par défaut	Commentaires	Extra	Action
1	id	int(11)			Non	Aucun(e)		AUTO_INCREMENT	Modifier Supprimer Plus
2	user_id	int(11)			Non	Aucun(e)			Modifier Supprimer Plus
3	subject	text	utf8mb4_general_ci		Non	Aucun(e)			Modifier Supprimer Plus
4	date	datetime			Non	Aucun(e)			Modifier Supprimer Plus

Figure 3.11: Notification table

#	Nom	Type	Interclassement	Attributs	Null	Valeur par défaut	Commentaires	Extra	Action
1	id	int(11)			Non	Aucun(e)		AUTO_INCREMENT	Modifier Supprimer Plus
2	topic	varchar(45)	utf8mb4_general_ci		Non	Aucun(e)			Modifier Supprimer Plus
3	msg	text	utf8mb4_general_ci		Non	Aucun(e)			Modifier Supprimer Plus
4	date	datetime			Non	Aucun(e)			Modifier Supprimer Plus
5	user_id	int(11)			Non	Aucun(e)			Modifier Supprimer Plus

Figure 3.12: Topics table

Figure 3.13: User table

3.4.4 Web application

The purpose of this application is to provide users with comprehensive information regarding their plant's status, which includes details about temperature, humidity, and the amount of water consumed.

3.4.5 Home page

When you open the application, you will be presented with a page where you can choose your role click on 1 as either a manager or a user.

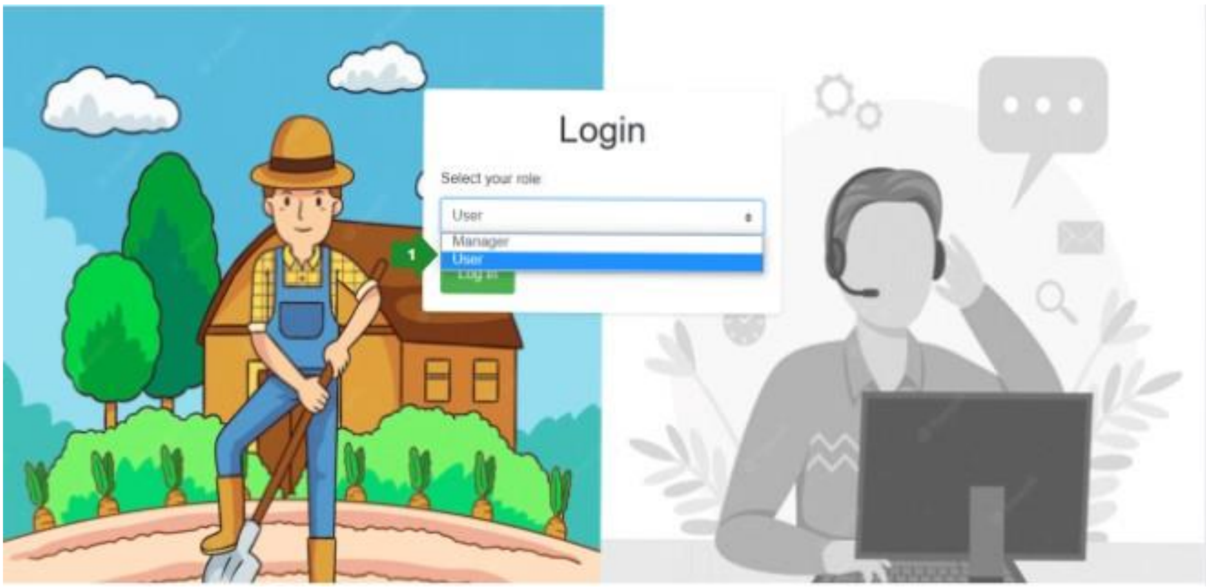


Figure 3.14: Select role

Manager space:

In this section, the interfaces of the manager are examined.

3.4.6 Login manager

Welcome to the manager login interface, this section is specifically designed for managers who have authorized access to the system. As a manager, logging in here gives you access to important tools to oversee and control different parts of the application. To get started, enter your login details securely to confirm your identity and gain access to the manager part, where you can manage users.

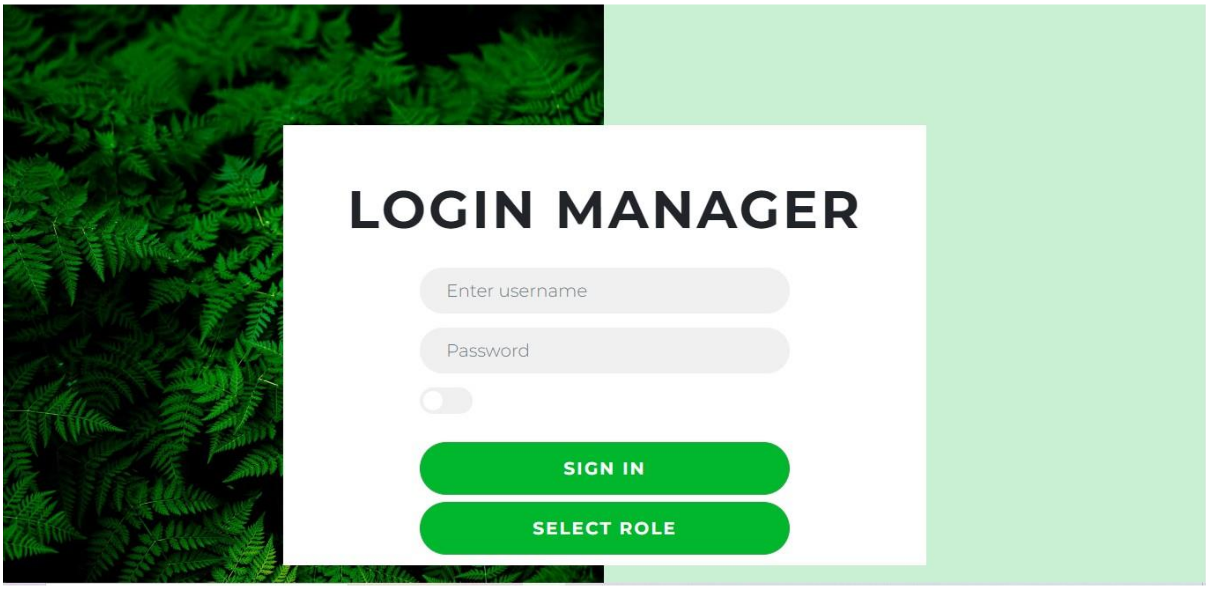


Figure 3.15: Login manager

3.4.7 List of Farmers

After the manager logs in , they can see a list of users (farmers). This "List of Farmers" interface shows important information about all the registered farmers in an organized way. It includes details like their names, contact information, location, and any other relevant data. This interface gives the manager a complete overview of the farmers associated with the system , easily communicate, and in this part the manager has the ability to add "**click on 2**", delete, and modify farmer records"**click on 3**" then Log out "**click on 1**" .

Figure 3.16: List of Farmers

3.4.8 Add Farmer

In the "Add Farmer" interface, the manager can enter the details of a new farmer to register them in the system. This interface allows the manager to input information such as the farmer’s name, contact details, location, and another data. After filling in the required fields, the manager can save the information to create a new farmer profile.

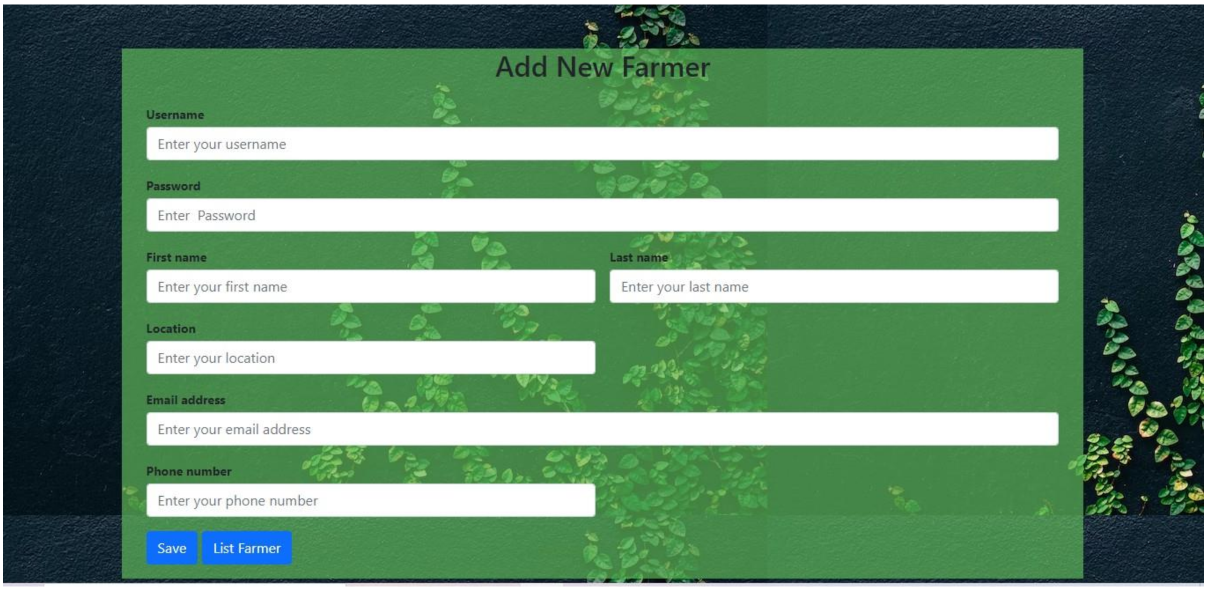


Figure 3.17: Add Farmer

Farmer space

In this section, the interfaces of the farmer are examined.

3.4.9 User Login

In the "User Login" interface, farmers can log in to their personal accounts. This interface ensures that only authorized farmers can access their own information and use the features available to them. Upon logging into the farmer's account, they gain access to their personalized dashboard, where they can find valuable insights about their plants and the current weather conditions.

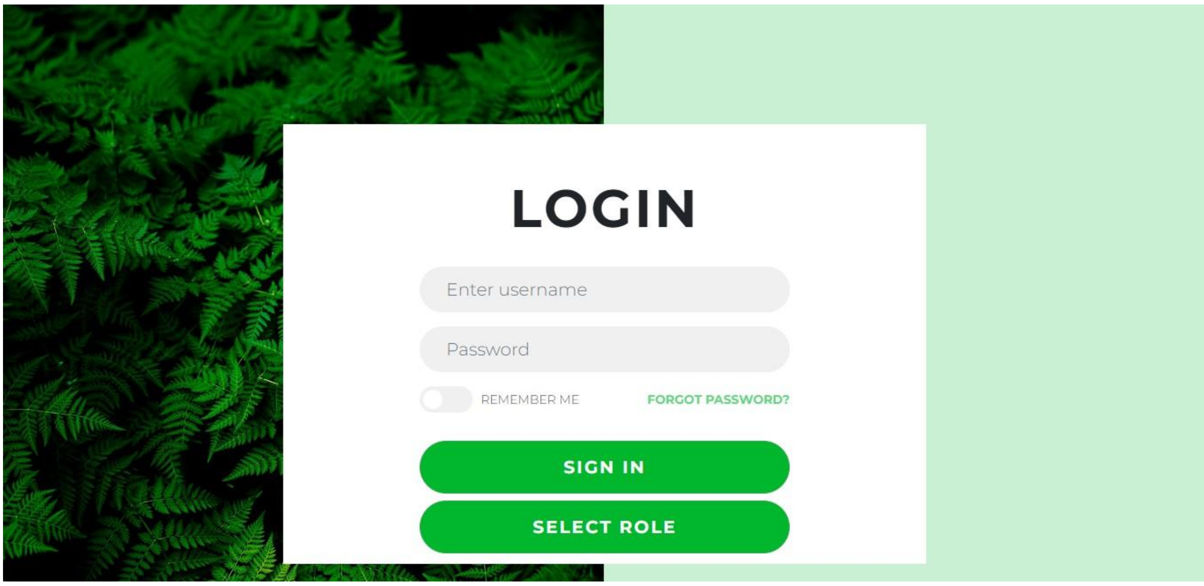


Figure 3.18: User Login

3.4.10 dashboard interface

The "dashboard interface" provides farmers with real-time information about their plant. It displays the current status of the plant, including essential details like soil moisture, temperature, humidity parameters. The interface also includes a feature that informs the farmer whether the plant requires irrigation or not, based on the analyzed data and predefined thresholds. Additionally, the dashboard presents a chart or graph that shows the average values of temperature, humidity, and soil moisture for each hour of the day. This allows farmers to track the trends and changes in these factors throughout the day, enabling them to make informed decisions about irrigation schedules and effectively manage their crops.

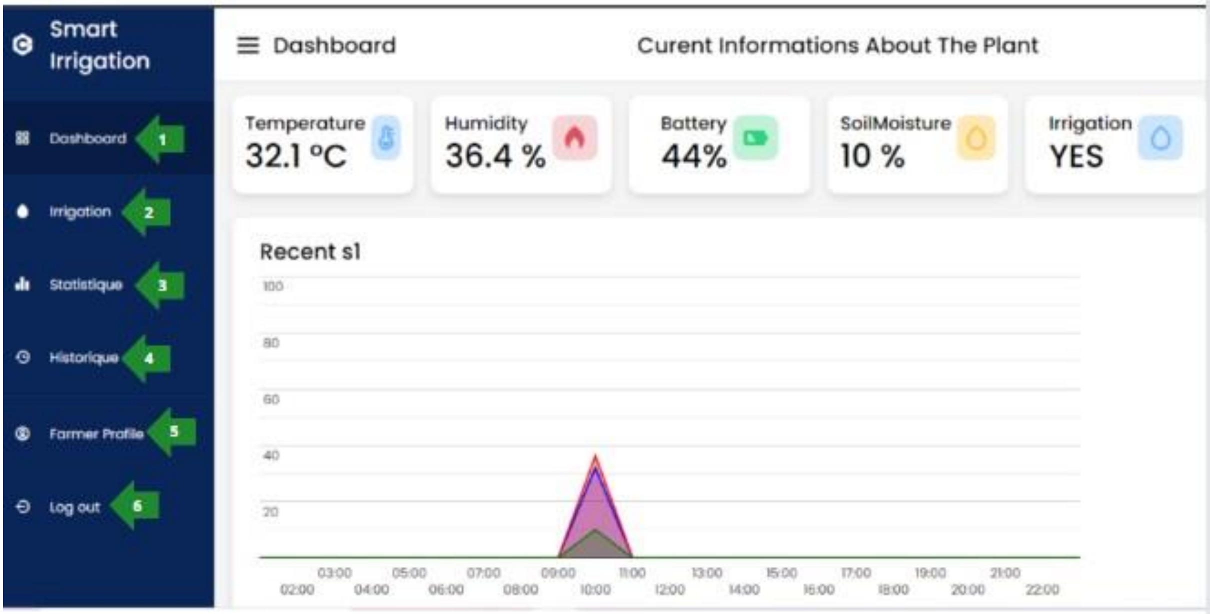


Figure 3.19: dashboard interface

Items dashboard are numbered:

The farmer is taken to the following by clicking on each number:

1. The dashboard for monitoring
2. The page of notifications and irrigation control
3. Statistics about real-time data
4. Historical daily data
5. Information about the farmer
6. Log out for the farmer

3.4.11 Notification interface

In the "Notification interface ", farmers receive messages and alerts about important events related to their plant. For instance, when the irrigation pump is activated, a notification is sent to the farmer, keeping them informed. Additionally, farmers can manually control the pump by activating or deactivating it through the interface. This interface ensures that farmers stay updated on irrigation activities and gives them the ability to have direct control over the pump as needed.

1. The notification block
2. The on/off switch of the pump

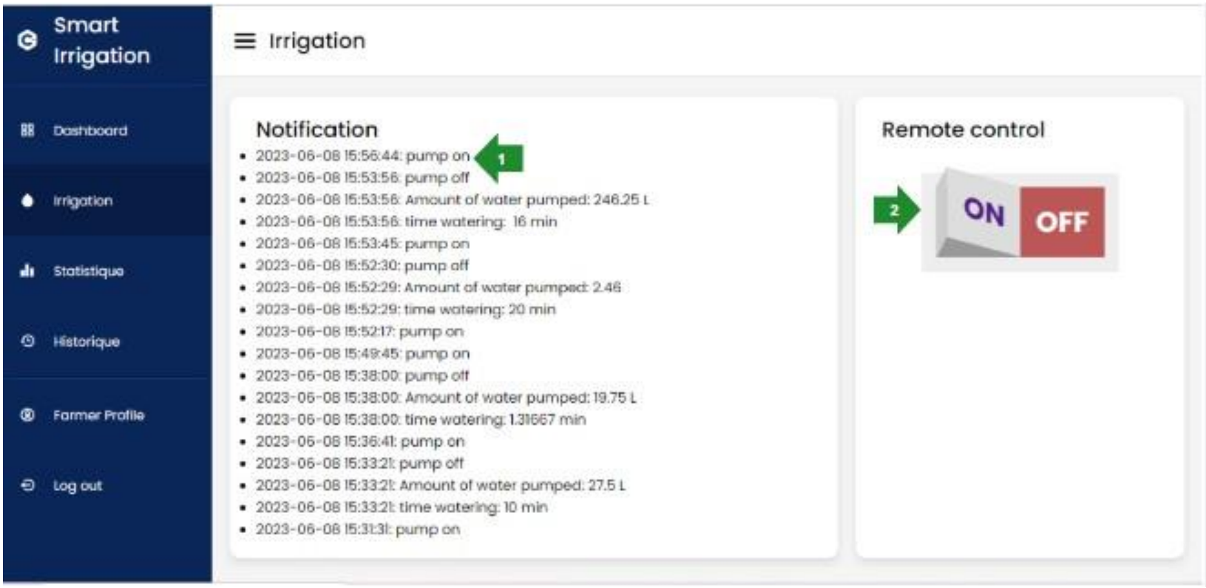


Figure 3.20: Notification interface

3.4.12 Real-time Statistics

In the "Real-time Statistics" interface, farmers can see live updates of the temperature, humidity, and soil moisture levels through easy-to-understand graphs. These graphs show the current information about the plant's conditions. The interface refreshes every 8 seconds to show the latest data, ensuring that farmers always have the most recent information about their crops. By keeping an eye on these real-time statistics, farmers can make timely decisions and take the right steps to keep their plants in the best possible conditions .

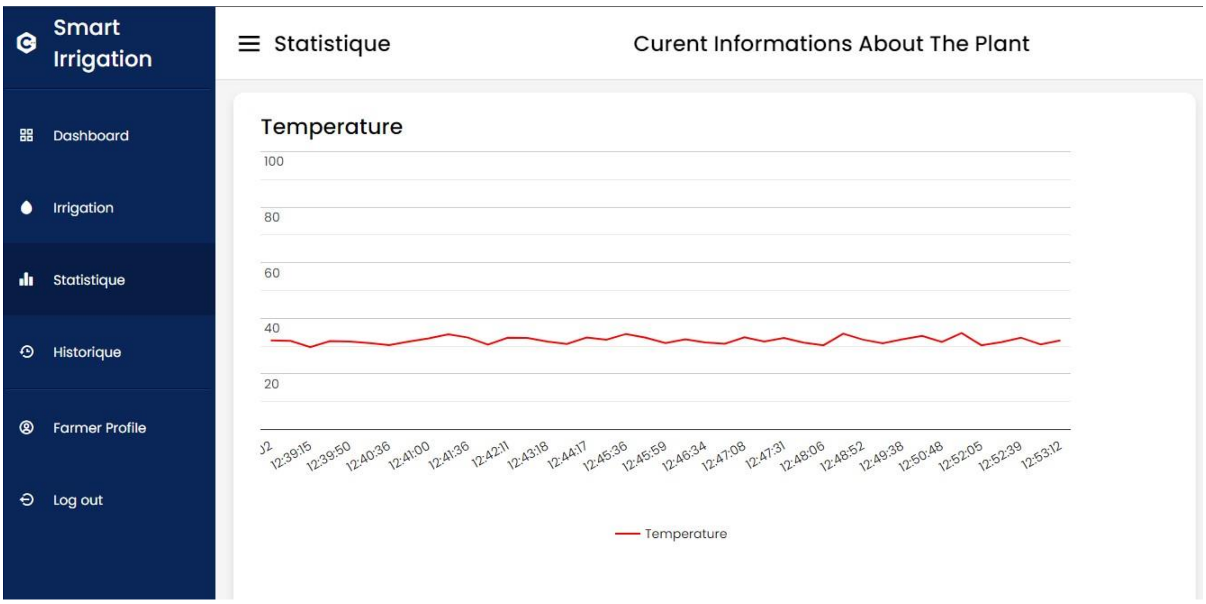


Figure 3.21: Real-time Statistics "Temperature"

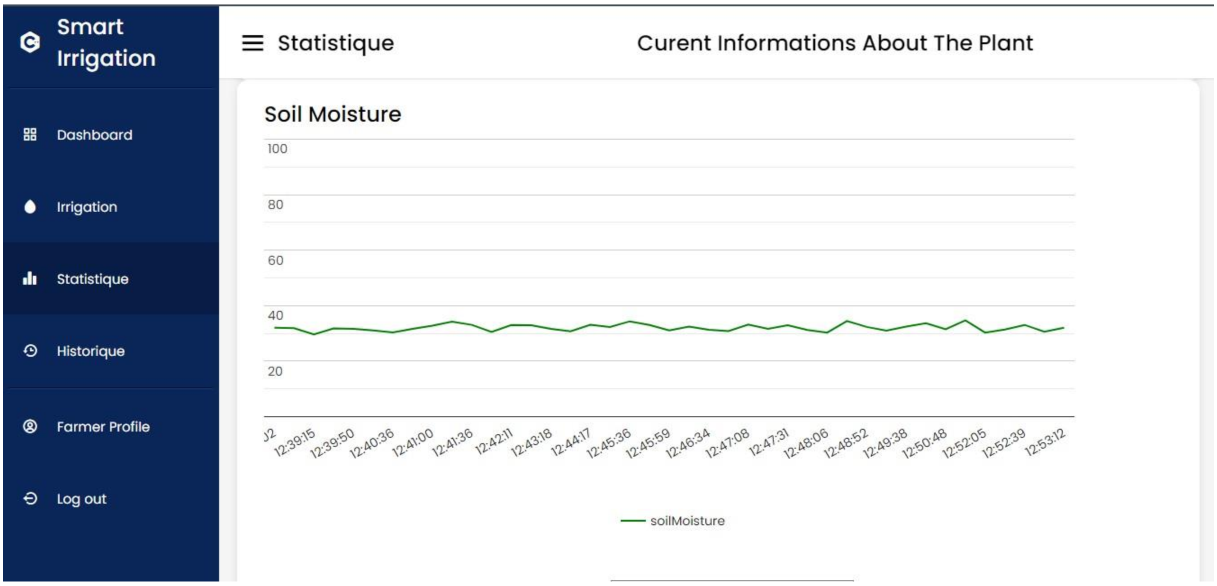


Figure 3.22: Real-time Statistics "Soil moisture"

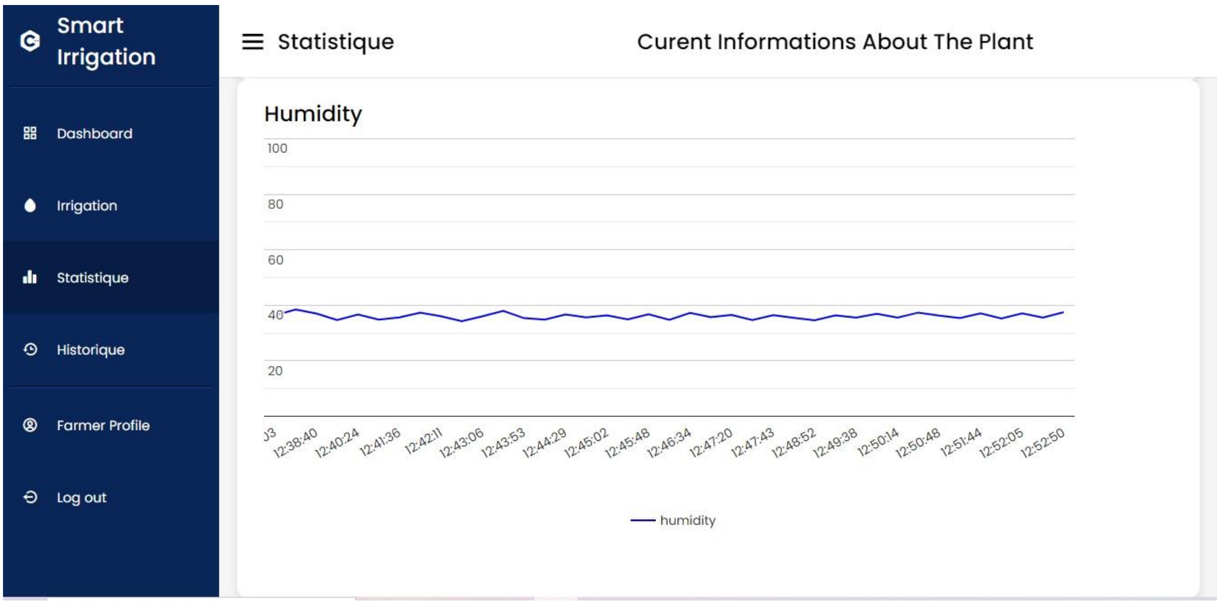
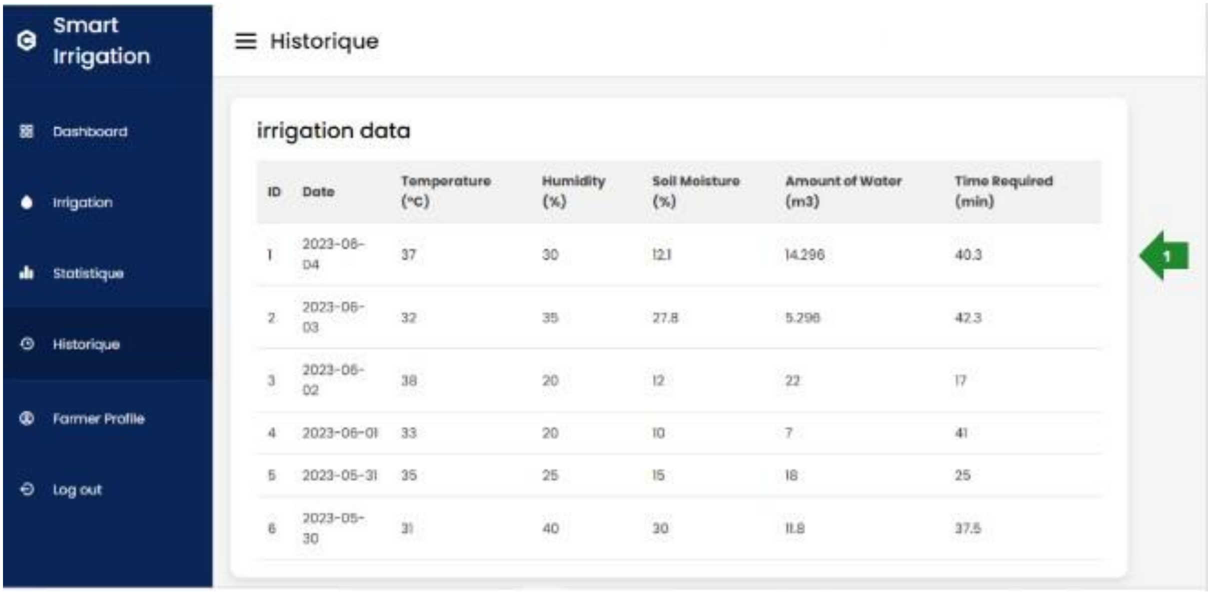


Figure 3.23: Real-time Statistics "Humidity"

3.4.13 Historical Data

In the "Historical Data" interface, farmers can find important information about the water they have used for irrigation and how long each irrigation session lasted. This interface keeps a record of past irrigation events, showing the total amount of water used over time and the duration of each irrigation session. By looking at this historical data, farmers can learn about their water usage patterns and make smart decisions to improve their irrigation practices. This helps them manage their water resources effectively, save water, and make sure their crops get the right amount of water at the right times.

1. The calculated average value of each parameter is determined for the entirety of a single day.



ID	Date	Temperature (°C)	Humidity (%)	Soil Moisture (%)	Amount of Water (m3)	Time Required (min)
1	2023-06-04	37	30	12.1	14.296	40.3
2	2023-06-03	32	35	27.8	5.296	42.3
3	2023-06-02	38	20	12	22	17
4	2023-06-01	33	20	10	7	41
5	2023-05-31	35	25	15	18	25
6	2023-05-30	31	40	30	11.8	37.5

Figure 3.24: Historical Data

3.4.14 Farmer Profile

In the "Farmer Profile" page, farmers can update their personal information. This page allows farmers to easily change details like their name, contact information, location, and other important data. By using this page, farmers can make sure their profile is accurate and has the most recent information.

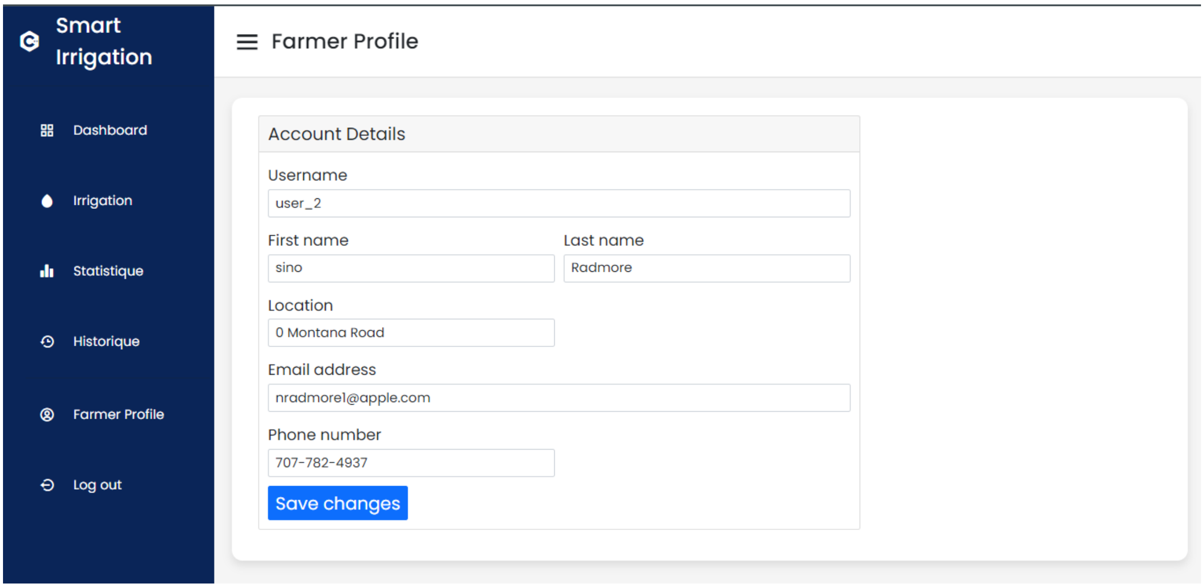


Figure 3.25: Farmer Profile

3.5 Conclusion

In this section presented an overview of how we implemented our irrigation monitoring application using IoT technology. We discussed the software and environments we used , we also highlighted the seamless connection made possible through the Haivemq interface and showcased the integration of our prototype with the cloud.In the end, we showcased the various interfaces of the application.

General Conclusion

Automated irrigation systems have revolutionized the way we manage water resources for agricultural and landscaping purposes. These systems incorporate advanced technologies, such as IoT and sensor-based monitoring, to optimize water usage and promote sustainable irrigation practices. By automating the irrigation process, these systems eliminate the need for manual intervention, save time and labor, and ensure precise and efficient water delivery to plants. The integration of sensors enables real-time monitoring of crucial factors like soil moisture, weather conditions, and plant water requirements, allowing for intelligent and data-driven irrigation decisions.

In this work, an IoT-based web application is being developed for the purpose of monitoring irrigation systems. The application leverages the power of IoT, utilizing interconnected devices and sensors to gather real-time data on various parameters, including soil moisture levels, temperature, and weather conditions. Through this IoT framework, the application enables seamless communication and data exchange between the irrigation system components and the web interface. This connectivity allows users to remotely monitor and control their irrigation systems, making adjustments based on the data received. By integrating IoT technology into the monitoring process, users can ensure timely and precise irrigation, optimize resource utilization, and improve overall efficiency in managing their irrigation systems.

Using IoT technology and web applications for monitoring irrigation offers several benefits. It enables remote monitoring and control of irrigation systems, allowing users to conveniently manage their irrigation processes. Real-time data collected through IoT devices helps in making informed decisions about irrigation scheduling and water usage. However, one disadvantage of not utilizing machine learning is the limitation in optimizing irrigation practices. Machine learning algorithms can analyze complex data patterns and dynamically adjust irrigation settings for better efficiency and adaptability.

future work

In future work, the focus will be on utilizing the collected sensor data to establish a comprehensive historical dataset. This dataset will serve as the foundation for developing a powerful machine learning model that can accurately analyze and predict water usage. By leveraging advanced algorithms and techniques, the model will enable precise calculation of the optimal irrigation requirements for various crops based on environmental factors and plant characteristics. Additionally, incorporating data visualization and analytics capabilities will provide valuable insights and recommendations for improving water management practices. The future work will involve fine-tuning the machine learning model, expanding its scope to cover different crops, and conducting thorough evaluations to ensure its effectiveness and applicability in real-world agricultural scenarios.

Bibliography

[1] Artificial intelligence in agriculture. *Artificial Intelligence in Agriculture*. Accessed on June 6, 2023.

[2] Assessing the influence of crop model structure on the performance of data assimilation for sugarcane. Accessed on June 3, 2023.

[3] Evapotranspiration is the sum of plant transpiration and evaporation. URL: https://www.usgs.gov/media/images/evapotranspiration-sum-plant-transpiration-and-evaporation. Accessed on June 7, 2023.

[4] Explication of conventional and smart irrigation systems with challenging factors. Accessed on June 1, 2023.

[5] Explication of conventional and smart irrigation systems with challenging factors. Accessed on June 7, 2023.

[6] Mqtt architecture broker controls the distribution of information and mainly acts as a mediator between the publisher and subscriber. Website. Accessed on June 7, 2023.

[7] The standard for iot messaging. https://mqtt.org/. Accessed on June 7, 2023.

[8] 7 benefits of mqtt protocol for iot. https://medium.com/@esperso/7-benefits-of-mqtt-protocol-for-iot-e463f6a97100, February 2020. Accessed on June 7, 2023.

[9] Younes Abbassi and Habib Benlahmer. Un aperçu sur la sécurité de l'internet des objets (iot). In *Colloque sur les Objets et systèmes Connectés-COC'2021*, 2021.

[10] Actu-Environnement. Évapotranspiration - définition - actu-environnement. Accessed on June 3, 2023.

[11] Fowzia Akhter, HR Siddiquei, Md Eshrat E Alahi, and SC Mukhopadhyay. Design and development of an iot-enabled portable phosphate detection system in water for smart agriculture. *Sensors and Actuators A: Physical*, 330:112861, 2021.

[12] Marwan Albahar. A survey on deep learning and its impact on agriculture: Challenges and opportunities. *Agriculture*, 13(3):540, 2023.

[13] Richard G Allen, Luis S Pereira, Dirk Raes, Martin Smith, et al. *Crop evapotranspiration-Guidelines for computing crop water requirements-FAO Irrigation and drainage paper 56*, volume 300. 1998.

[14] Don Box, David Ehnebuske, Gopal Kakivaya, Andrew Layman, Noah Mendelsohn, Henrik Frystyk Nielsen, Satish Thatte, and Dave Winer. Simple object access protocol (soap) 1.1, 2018.

[15] Erion Bwambale, Felix K Abagale, and Geophrey K Anornu. Smart irrigation monitoring and control strategies for improving water use efficiency in precision agriculture: A review. *Agricultural Water Management*, 260:107324, 2022.

[16] DAILY CIVIL. Sprinkler irrigation - advantages & disadvantages. May 2018.

[17] William W Cohen. *Machine Learning Proceedings 1994*. Elsevier, 2014.

[18] ElProCus. Soil moisture sensor: Pin configuration, working and applications.

[19] Ian Fette and Alexey Melnikov. The websocket protocol. Technical report, 2011.

[20] Conserve Energy Future. Types, methods and importance of irrigation (watering crops), 7 2022. Accessed on June 6, 2023.

[21] Amarendra Goap, Deepak Sharma, A Krishna Shukla, and C Rama Krishna. An iot based smart irrigation management system using machine learning and open source technologies. *Computers and electronics in agriculture*, 155:41–49, 2018.

[22] Patrice Godefroid, Michael Y Levin, and David Molnar. Sage: whitebox fuzzing for security testing. *Communications of the ACM*, 55(3):40–44, 2012.

[23] Sedat Görmüş, Hakan Aydın, and Güzin Ulutaş. Security for the internet of things: a survey of existing mechanisms, protocols and open research issues. *Journal of the Faculty of Engineering and Architecture of Gazi University*, 33(4):1247–1272, 2018.

[24] Ashutosh Gupta, Varun Krishna, Saarthak Gupta, and Jitesh Aggarwal. Android based solar powered automatic irrigation system. *Indian Journal of Science and Technology*, 9(47):2–7, 2016.

[25] M Han and H Zhang. Business intelligence architecture based on internet of things. *Journal of Theoretical & Applied Information Technology*, 50(1):90–95, 2013.

[26] Ronghang Hu, Jacob Andreas, Trevor Darrell, and Kate Saenko. Explainable neural computation via stack neural module networks. In *Proceedings of the European conference on computer vision (ECCV)*, pages 53–69, 2018.

[27] Ozan Irsoy and Claire Cardie. Deep recursive neural networks for compositionality in language. *Advances in neural information processing systems*, 27, 2014.

[28] S. K. Jain, P. K. Agarwal, and V. P. Singh. *Hydrology and water resources of India*, volume 57. Springer Science & Business Media, 2007.

[29] Kirtan Jha, Aalap Doshi, Poojan Patel, and Manan Shah. A comprehensive review on automation in agriculture using artificial intelligence. *Artificial Intelligence in Agriculture*, 2:1–12, 2019.

[30] Md Rezwanul Karim, Md Mahbub Hossain, Saptarshi Ghosh, Mohammad Sorowar Hossain, Mohammad Akhtar Hossain, and Mohitul Islam Chowdhury. Predicting the risk factors and the associated economic burden of childhood malnutrition in bangladesh. *Annals of the Romanian Society for Cell Biology*, 25(5):2716–2730, 2021. Accessed on June 10, 2023.

[31] Surapon Kraijak and Panwit Tuwanut. A survey on internet of things architecture, protocols, possible applications, security, privacy, real-world implementation and future trends. In *2015 IEEE 16th International Conference on Communication Technology (ICCT)*, pages 26–31. IEEE, 2015.

[32] Krishisewa. Use of iot based irrigation scheduling for smart farming - krishisewa.

[33] Krishisewa. Explication of conventional and smart irrigation systems with challenging factors. https://www.krishisewa.com/miscellaneous-articles/1018-use-of-iot-based-irrigation-scheduling-for-smart-farming.html, 6 2019. Accessed on June 1, 2023.

[34] Anders Krogh. What are artificial neural networks? *Nature biotechnology*, 26(2):195–197, 2008.

[35] N. A. M. Leh, M. S. A. M. Kamaldin, Z. Muhammad, and N. A. Kamarzaman. Smart irrigation system using internet of things. In *2019 IEEE 9th International Conference on System Engineering and Technology (ICSET)*, pages 96–101. IEEE, 2019.

[36] Qin Liu, Qing Yan, Jianying Tian, and Kexin Yuan. Key technologies and applications in intelligent agriculture. In *Journal of Physics: Conference Series*, volume 1757, page 012059. IOP Publishing, 2021.

[37] Neha K Nawandar and Vishal R Satpute. Iot based low cost and intelligent module for smart irrigation system. *Computers and electronics in agriculture*, 162:979–990, 2019.

[38] Netafim. Drip irrigation systems. Accessed on June 3, 2023.

[39] Netafim. Drip irrigation systems, 6 2023. Accessed on June 3, 2023.

[40] M. A. Salama, K. M. Yousef, and A. Z. Mostafa. Simple equation for estimating actual evapotranspiration using heat units for wheat in arid regions. *Journal of Radiation Research and Applied Sciences*, 8(3):418–427, 2015.

[41] Muhammad Saqib, Tarik Adnan Almohamad, and Raja Majid Mehmood. A low-cost information monitoring system for smart farming applications. *Sensors*, 20(8):2367, 2020.

[42] Nirmal Sengupta. Irrigation: traditional vs modern. *Economic and Political Weekly*, pages 1919–1938, 1985.

[43] Sören Sonnenburg, Gunnar Rätsch, Sebastian Henschel, Christian Widmer, Jonas Behr, Alexander Zien, Fabio de Bona, Alexander Binder, Christian Gehl, and Vojtěch Franc. The shogun machine learning toolbox. *The Journal of Machine Learning Research*, 11:1799–1802, 2010.

[44] William E Splinter. Center-pivot irrigation. *Scientific American*, 234(6):90–99, 1976.

[45] Sunardi Sunardi, Aulia Afif, and Fiftin Noviyanto. Real time monitoring and irrigation control using the websocket protocol. In *Proceedings of the 1st International Conference on Science and Technology for an Internet of Things, 20 October 2018, Yogyakarta, Indonesia*, 2019.

[46] Unknown. Introduction of soap and rest web services - javatpoint. *www.javatpoint.com*. Accessed on June 7, 2023.

[47] Unknown. What is soap? *App Architecture*, June 2022. Accessed on June 7, 2023.

[48] Cédric Villani, Yann Bonnet, Charly Berthet, François Levin, Marc Schoenauer, Anne Charlotte Cornut, and Bertrand Rondepierre. *Donner un sens à l'intelligence artificielle: pour une stratégie nationale et européenne*. Conseil national du numérique, 2018.