Middle East Journal of Science

https://dergipark.org.tr/mejs

MEJS

e-ISSN:2618-6136

Research Article

A PROPOSED SMART IRRIGATION MANAGEMENT SYSTEM BASED ON THE IOT AND CLOUD COMPUTING TECHNOLOGIES

Waseem HAMDOON* [1] ***Ahmet ZENGIN*** [1]

[1]Sakarya University, Faculty of Computer and Information Engineering, Sakarya, Türkiye
[*]Corresponding author: waseem.algburi@ogr.sakarya.edu.tr

Abstract: *Most fresh water is used in agriculture. There has been a constant interest in presenting systems and solutions that rationalize water resources in agriculture without reducing productivity. In contrast, the solutions must improve production while utilizing less water. On the other hand, The Internet of Things is a prominent recent technology that provides various solutions in many disciplines, including agriculture and irrigation. This paper proposes an Internet-of-Things-based architecture for smart irrigation by developing a prototype with a controller unit, water pumps, and sensors. These systems monitor the soil's irrigation needs and determine the right amount based on sensor data. As these values are delivered through cloud computing to a user's mobile app, irrigation may be monitored and controlled from multiple angles. This comprises manual irrigation mode and automatic irrigation mode and determines the right amount of irrigation based on sensor relationships.*

Keywords: *Internet of Things; Cloud Computing; Smart Irrigation; Actuators; Sensors.*

Received: July 22, 2022 *Accepted: June 26, 2023*

1. Introduction

Agriculture is the main source of food and, at the same time, the largest consumer of fresh water, consuming up to 70% of all water resources [1]. Where this high percentage of water consumption explains the reason for the increasing interest by researchers in the possibility of reducing the use of irrigation water [2]. Several concepts emerged around this, including "sustainable irrigation," one of the main concerns for preserving water resources through wise policies in rationalizing water consumption [3]. Contrarily, inefficient irrigation practices have several detrimental effects, the biggest among them water waste and a decrease in the quality of the crops that are produced, particularly when irrigation is irregular [8] and does not take into account the demands of each plant or crop separately. Mostly, there is more than one type of plant in the fields and each type requires a different amount of water [9]. In many cases, farmers pump more water than is required (excessive irrigation), which leads to lower production as well as the waste of water and energy [10].

On the other hand, the use of Internet of Things technology has grown significantly in recent years, and as a result, there have been more and more devices connected to the Internet as a result of the need to collect data from these devices for various Internet of Things applications [4]. It is estimated that the number of connected devices will reach 25.1 billion by 2025 [5]. In the field of irrigation, IoT offers various applications to monitor crop growth and support irrigation decisions [6], making it a logical choice for smart water management applications. Currently, despite the spread of IoT, there are still some challenges that prevent the widespread use of IoT for precision irrigation, such as the need to develop software for IoT-based smart applications, and the need to automate irrigation processes using dedicated platforms which is not yet fully automated [7], and so on.

In this study, a proposal for a smart irrigation system based on the Internet of Things was presented, through which the appropriate amount of irrigation can be controlled, as well as the appropriate irrigation time, considering the weather conditions of the crop and soil.

Although there are many studies or proposed systems in this field, however, there are problems in representing these proposed systems, most notably: (1) Adopting fixed boundary values at which the irrigation decision is taken, which cannot be changed easily or at any time according to the supervisor of the irrigation process. (2) The amount of irrigation is fixed for all crops/plants without taking into account the existence of a difference in the amount of water required for each, (3) The amount of irrigation is fixed - non-dynamic - for a single crop/plant without taking into account the difference in other factors affecting the soil, such as (soil temperature, air humidity, air temperature) and their impact on determining the appropriate amount based on those variable values. Accordingly, this study attempts to overcome these problems by proposing a new irrigation architecture based on the Internet of Things, as well as some other additional features that increase the efficiency of irrigation management.

2. Material and Methods

The techniques used to represent the proposed system are briefly discussed in this section.

2.1. Hardware

Most IoT-based systems are made up of three fundamental parts: a controller, a sensor, and an actuator, in addition to a middle layer to enable the control unit to connect to the Internet to send the data gathered as well as instructions that are required.

2.1.1 Controller

It is usually used to automatically control devices and equipment [11]. The controller in IoT applications is like the brain, processing inputs from sensors or other sources and producing outputs from one of the actuators. In this study, the “Raspberry Pi 3 Model B” is employed as a controller. The Raspberry Pi Foundation, in collaboration with Broadcom, has produced a line of miniature single-board computers (SBCs). Initially, the Raspberry Pi project aimed to promote basic computer science education in schools and impoverished countries. Its low cost, versatility, and open design make it popular in many fields, including weather monitoring [12][13].

2.1.2 Sensors

Sensors are essential in automating any application since they collect data and process it [11].

- **Soil Moisture Sensor**

A “Capacitive Soil Moisture v2.0” sensor is used to measure the soil's moisture level. This capacitive sensor has excellent responsiveness to local soil moisture changes and provides an effective correlation between gravimetric water content and output voltage [14][15].

- **Soil temperature sensor**

To measure the soil temperature, a sensor of the type DS18B20 is used. It is a digital temperature sensor, used in many applications and difficult environments. The startup resolution is 12 bits, where there is a resolution of 9, 10, 11, or 12 bits, which means the temperature sensor can accurately measure temperatures of 0.25°C, 0.125 °C, and even 0.0625°C [16][17].

- **Air temperature and humidity sensor**

A sensor of type DHT11 is used to monitor and measure the temperature and humidity of the surrounding environment of a field or crop [17]. The DHT11 is a popular temperature and humidity sensor that features a dedicated (NTC) for measuring temperature as well as an 8-bit microprocessor for outputting the values of temperature and humidity as serial data [18][19].

2.1.3 Actuators

On the Internet of Things, actuators use data received by sensors and processed by software to control or take action in the system [20].

- **Water pump**

A water pump is used for the purpose of irrigating the crop or field when the threshold limit is reached.

2.2. Software

It is typical for Internet of Things applications to be represented by a group of software programs. The following software packages are being utilized in this research.

2.2.1 Raspberry Pi Requirements

- **Raspberry Pi OS**

It is the ideal operating system for Raspberry Pi devices, and it is regularly upgraded with an emphasis on reliability and performance; Also, it includes over 35,000 packages [21]. It's worth noting that it's based on the Debian (Linux distribution) and is likewise a free system [22].

- **Python**

Python is a high-level programming language that combines power and clarity [23]. Due to this, it was included in the Raspberry Pi OS operating system since Python is a powerful tool that is easy to learn and use [24]. Version 3.7 is used in this research.

- **TinyDB**

TinyDB is a document-oriented, lightweight database; it's written entirely in Python and does not require any external resources.

2.2.2 Cloud hosting

Cloud hosting makes software and websites available via cloud resources [25] with minimal start-up costs, resource elasticity, and scale savings [26]. Firebase, a cloud hosting service from Google, is used in this research.

- **Firebase real-time database**

It's a No-SQL, cloud-based database that syncs data in real-time across all clients [27][28].

- **Firebase authentication**

Most applications require user authentication to function properly; a user's identity may allow apps to securely store user data in the cloud and give a consistent, personalized experience across all the user's devices; Firebase Authentication is tightly linked with the rest of the Firebase platform [29][30].

2.2.3 Mobile application

Flutter is used to build a mobile application that is used by the end-user to manage the irrigation process.

- **Flutter**

The Flutter framework from Google allows developers to create natively built, cross-platform mobile apps with a single codebase [31][27]. As a result, the platform-independent emulator can run simultaneously on both Android and iOS devices.

- **Dart**

It is the programming language that is employed to construct applications for Flutter. It was initially created and is now maintained by Google as a JavaScript replacement. Additionally, it adopts a syntax that is comparable to the Java programming language in an effort to attract Java programmers [32].

3. Results

In this section, the proposed three-layer system architecture, how each layer is implemented, and finally, what the proposed system will look like will be presented as the results of this study.

3.1. System Architecture

The architecture of the proposed smart irrigation system in this study can be described as consisting of three layers: The Internet of Things (IoT) layer, the cloud layer, and the application layer, as shown in Figure 1.

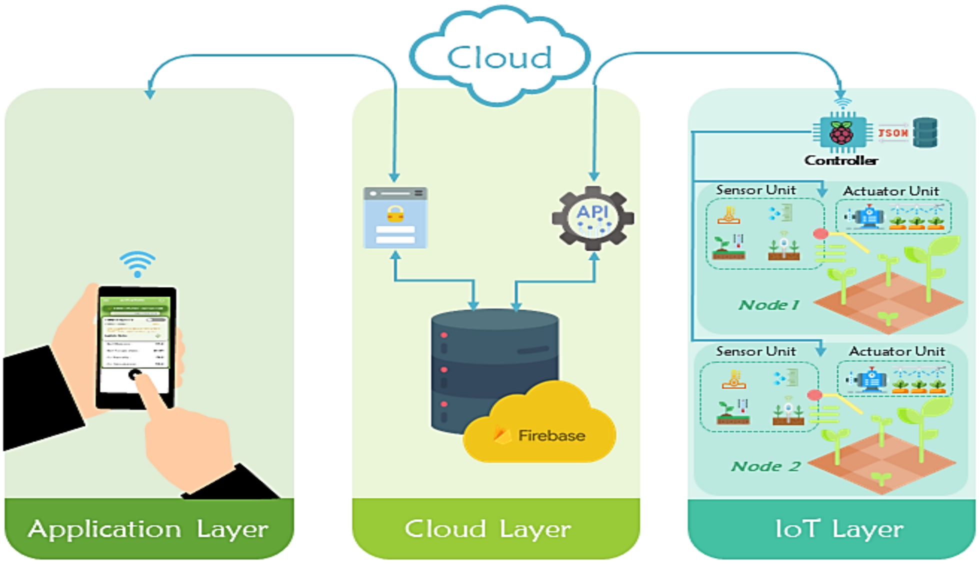


Figure 1. The architecture of the proposed smart irrigation system

3.2. System Implementation

This section describes the implementation of each layer of the proposed system.

3.2.1 IoT layer

The implementation of the IoT layer can be divided into two stages:

I. Hardware implementation stage:

To start the implementation of this stage, a "Pi T-Cobbler Breakout" is used to extend 40 GPIO pins on the "Raspberry Pi Model B" to the breadboard for the purposes of protecting those pins installed on the Raspberry Pi as well as facilitating their handling, that the items (sensors/components) of each node are connected to the GPIO pins.

II. Software implementation stage:

First, "Raspberry OS" is installed on "Raspberry Pi 3 Model B," and then "Visual Studio Code" is installed to start building a "Python 3.7" irrigation management application. Since it's good programming practice to divide enormous code into smaller files for better code management and maintenance, the irrigation control application has been broken into Python modules [33].

The flowchart for this stage can be seen in Figure 2.

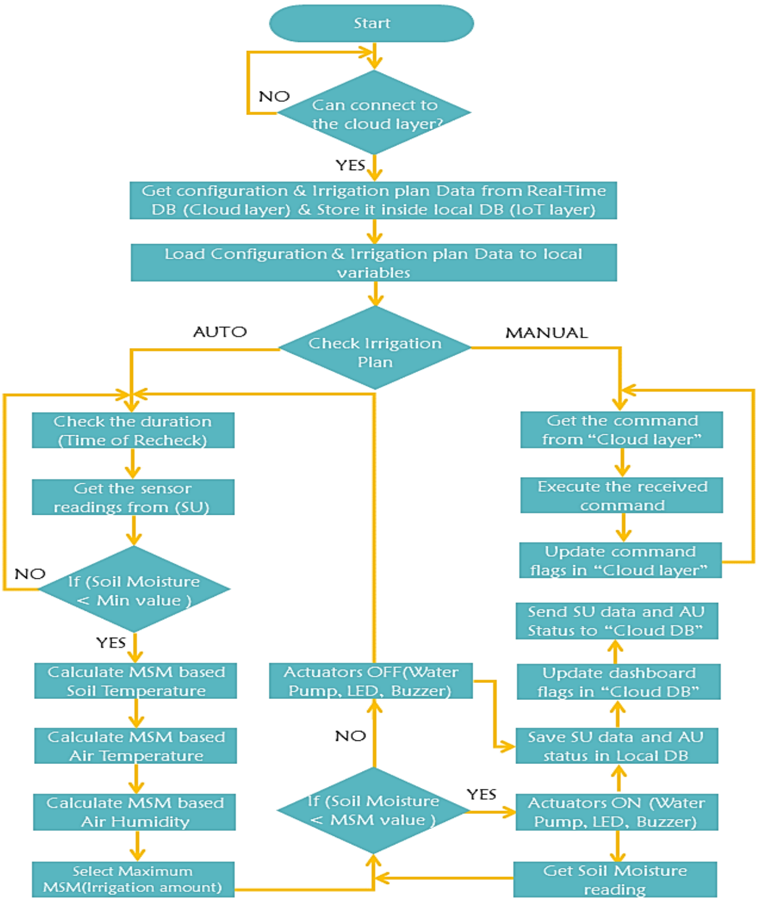


Figure 2. Flowchart of smart irrigation application in "IoT layer."

3.2.2 Cloud layer

Firebase can be used in a mobile app's "application layer" and "IoT layer." These instances enable real-time access to data so updates to the database reach users' devices in real-time. After this instance is generated, it's added to the irrigation management app's "IoT layer" and the mobile app's model. The "application layer" is not directly connected to the cloud layer; instead, the authorized user's e-mail address and password are verified by the verification service before the program is allowed to access the database data.

3.2.3 Application layer

"Android Studio" is used as an (IDE) to create a mobile application that allows the end-user control to the "nodes" of the "IoT layer" across the cloud layer.

3.3. System Demonstration

In this section, each layer of the IoT-based smart irrigation system architecture will be demonstrated after its implementation as follows. The proposed system prototype can be demonstrated in Figure. 3.

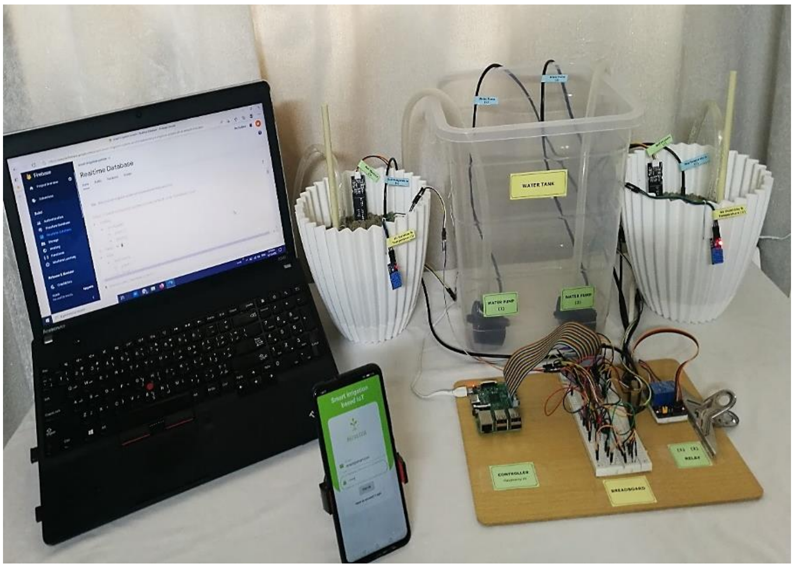
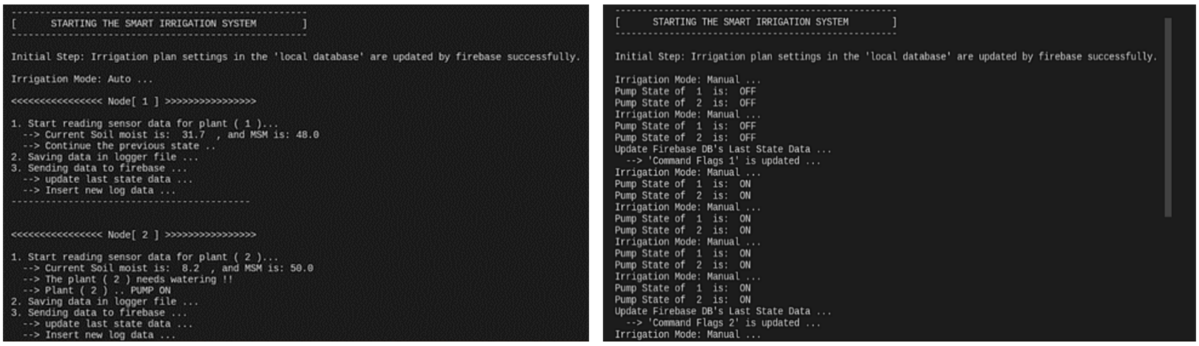


Figure 3. Prototype of the proposed system

3.3.1 IoT layer:

It is shown in Figure 3, the hardware stage in the IoT layer, where the sensors unit (SU) and water pumps were delivered to each node separately so that the decision to irrigate each node is made based on the desired mode (auto/manual) as well as on the sensor values and the appropriate amount of irrigation for each node. On the other hand, the software stage in the IoT layer of the smart irrigation management application when the irrigation mode is automatically selected is demonstrated in Figure 4. A.

In contrast, it is shown in Figure 4. B, the flag state changes after receiving orders such as Pump On/Off or update data.



(A) (B)

Figure 4. Smart irrigation management application (IoT layer) (A) Auto irrigation mode. (B) Manual irrigation mode.

3.3.2 Cloud layer:

Figure 5 illustrates how the Firebase real-time database would look after the “Cloud layer” is implemented and connected to the IoT layer and application layer. In addition, if the database data of this layer is updated by the “IoT layer” or the “application layer,” these changes will be indicated by highlighting the changed value.

Figure 5. Firebase real-time database

3.3.3 Application layer:

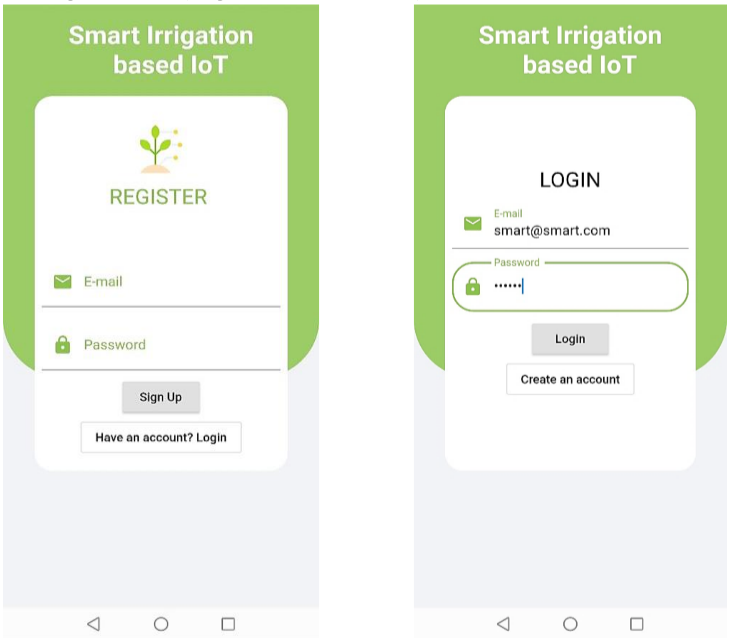
This layer reflects the actual testing of the proposed system's outputs and their conformity to the objective for which it was built, and it can be clarified using the screens below:

- **Login Screen:**

In this screen, registration data is transferred to Firebase's authentication service in the cloud layer. If the user is authorized to enter, it can access the application or the data stored in the cloud database, which can be used to control and preview IoT data.

- **Main Screen:**

This screen shows after passing the input; from here, the user can select the relevant screen from among the five possible screens: the dashboard screen, the sensors screen, the actuators screen, the settings screen, and the logs screen (Figure 6. A, 6. B, 7).



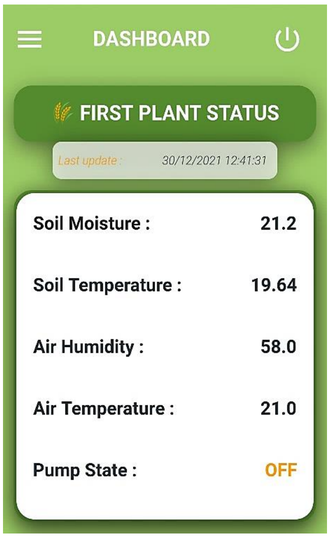
(A) (B)

Figure 6. Login screen (A) Sign up. (B) Login.

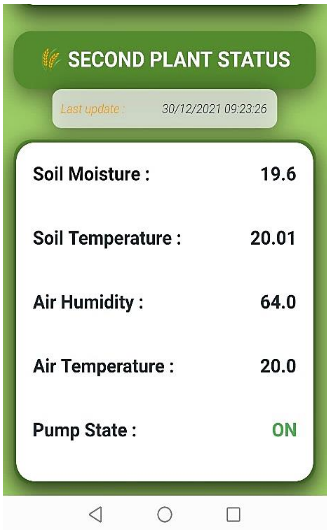
Figure 7. Main screen

- **Dashboard Screen:**

This screen displays the soil moisture sensor, soil temperature sensor, ambient temperature and humidity sensor, and water pump status for each node (field/plant) (Figure 8. A). In automated irrigation mode, this page is separated into two areas, one for each node, with the option to see the update time and date (Figure 8. B).



(A)



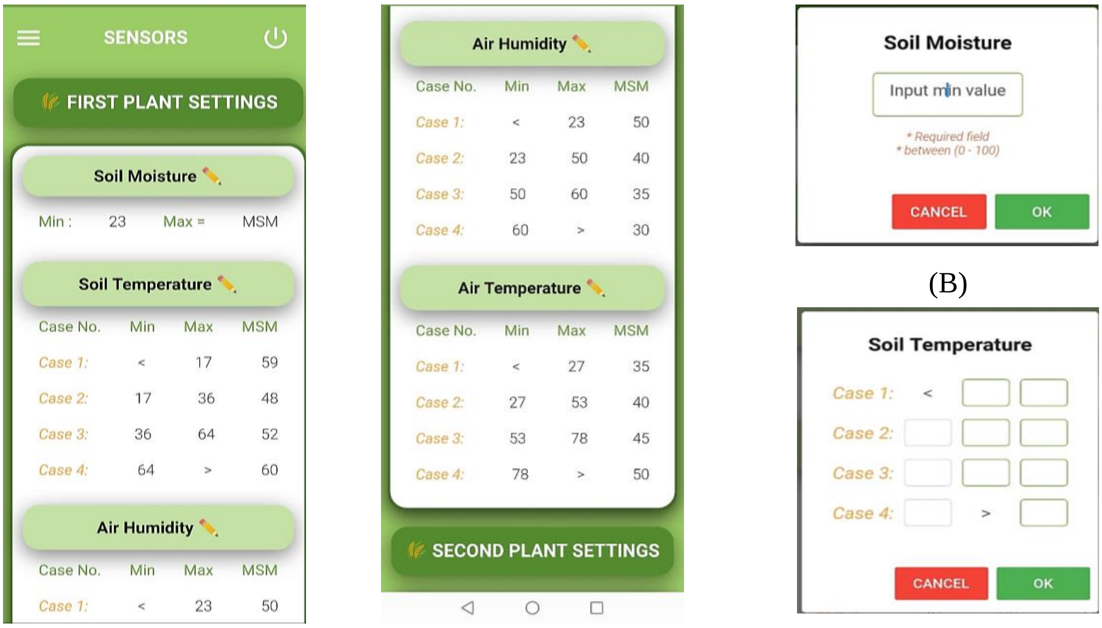
(B)

Figure 8. Dashboard screen (A) The readings of the first plant sensors. (B) The readings of the second plant sensors

- **Sensor's screen:**

This screen displays an end-user the sensor thresholds and ranges. In automatic irrigation mode, the minimum soil moisture sensor reading determines irrigation. It shows the lowest soil moisture or percentage of water in the soil at the point the water pump is operated to irrigate the soil.

The Maximum Soil Moisture (MSM) value influences how much water is needed for soil irrigation. When this value is reached, the soil irrigation pump is turned off.



(A) (B) (C)

Figure 9. Sensor's screen (A) Sensor readings. (B) Set the soil moisture value. (C) Set the soil temperature cases.

To achieve dynamism in the irrigation choice, represented by the marginal value, a value can be specified for each plant and adjusted at any moment, as illustrated in Figure 9. B, C shows how the minimum and maximum sensor values can be changed for additional sensors and levels.

- Actuators screen:

Figure 10 shows the irrigation mode screen when each node (field/crop) has its operator (water pump). This screen can be used to manually regulate and update the sensor readings, water pump status, and update time/date. In automatic irrigation mode, the on/off button of the water pump is disabled since the functioning of the pump is controlled automatically.

- **Settings Screen:**

This screen is important for choosing an irrigation plan. The irrigation plan button toggles automatic and manual irrigation. You can also choose the time period during which the sensors are reread to decide if irrigation is needed and how much water to apply. This value can be updated dynamically at any moment (see Figure 11).

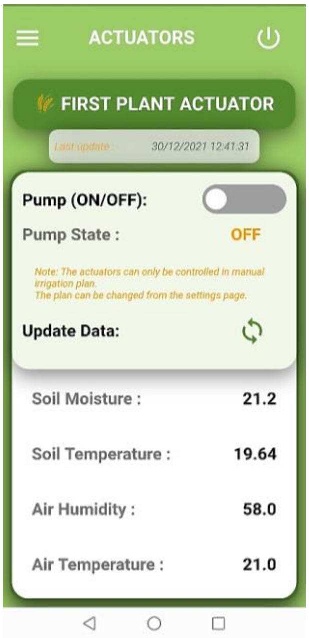


Figure 10. Actuators screen

Figure 11. Settings screen

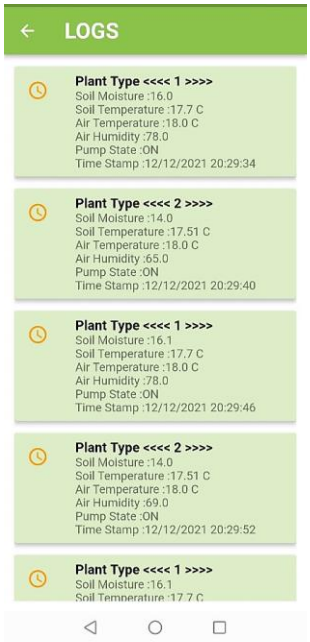


Figure 12. Logs screen

- **Logs Screen:**

This screen displays "Irrigation Need Check for each (Auto) Node" records. Figure 12 shows that the database stores SU values for each node (soil moisture sensor, soil temperature, ambient temperature, and humidity), AU unit or pump status, and the time stamp.

4. Discussion

The literature on smart irrigation using the Internet of Things focuses on answering two main questions: When is an irrigation decision made? How do you determine the appropriate amount of irrigation? Many of the proposed systems have been presented to answer the two previous questions, but they often fall into one of the following problems: (1) Adopting fixed limit values at which the irrigation decision is made, which cannot be changed easily or at any time according to the supervisor. Irrigation process. (2) The amount of irrigation is determined for all crops/plants without taking into account that there is a difference in the amount of water required for each of them, (3) The amount of irrigation is fixed - not dynamic - for one crop/plant without taking into account the difference in other factors affecting the soil such as (soil temperature, air humidity, air temperature) and their influence in determining the appropriate amount based on those variable values.

"Shekhar et al." proposed an intelligent system based on the Internet of Things, which uses only two sensors: one for assessing soil moisture and one for measuring soil temperature. The sensor data is processed by an intelligent algorithm, which predicts the best irrigation decision, but only two sensors were used, and the needs of each individual plant were ignored [34]. An irrigation system that uses the decision tree algorithm has been presented by "Mohammad A. Abbadi et al.," but although this model was developed based on soil characteristics and temperatures, the authors themselves say that because other locations differ from South Jordan Wadi, where the model was developed, the results cannot be generalized to all locations [35]. In addition, in the study of "Rao et al.," an intelligent field monitoring and automation system based on Internet of Things technology has been proposed, which uses two sensors: soil moisture sensors and soil temperature sensors, with bound values chosen to calibrate the sensors based on previous months' values, the irrigation water pump is operated and monitored using a computer application, However, he neglected to take into account the air temperature and humidity when determining the appropriate amount of irrigation [36].

In this study, a smart system that considers the dynamic factor in determining the appropriate amount of irrigation and the appropriate irrigation time, in addition to other advantages, was proposed to solve the representational issues associated with systems of this type. Practically and for the purpose of applying the algorithm in determining the appropriate amount for irrigation, the following table shows 4 cases for each sensor and its relationship to the soil moisture sensor. It was previously defined for the first plant, depending on either the opinion of experts or the data provided by the competent authorities in agriculture (see Table 1).

Table 1. Settings of the first plant sensors

Sensor	Case No.	Min	Max	MSM
Soil Moisture	----	19	Max(MSM)	----
Soil Temperature	Case 1	<17	17	35
	Case 2	17	36	45
	Case 3	36	64	55
	Case 4	64	>64	65
Air Temperature	Case 1	<10	10	40
	Case 2	10	26	50
	Case 3	26	42	60
	Case 4	42	>42	70
Air Humidity	Case 1	<25	25	65
	Case 2	25	45	55
	Case 3	45	65	45
	Case 4	65	>65	35

Table 2 displays a full day of sensor readings for the first plant. By selecting the correct MSM value for each sensor reading within its related case range, then selecting the maximum value among them, it is possible to determine the proper amount of irrigation to apply when the soil moisture sensor value reaches the minimum value.

Table 2. Read sensors for a full day of the first plant.

Soil Moisture	Soil Temperature	Air Temperature	Air Humidity	Pump State	Time
26.6	5	3.2	82	OFF	03:00
24.9	10	7.3	81	OFF	06:00
23.1	19	15.3	77	OFF	09:00
20.7	24	20.6	73	OFF	12:00
18.5	21	19.2	72	ON	15:00
35.4	16	13.5	75	OFF	18:00
32.5	13	10.7	79	OFF	21:00
29.2	9	5.9	81	OFF	00:00

Figure 13 shows the relationship between the moisture sensor and other sensors, as it is noted that when the reading of the soil moisture sensor reaches a value of 18.5 which is less than the minimum value at which irrigating must be performed, the system takes an automatic decision to irrigate, as for the amount, after determining the reading of each sensor, we have three values, which are (45, 50, 35), and therefore choose the maximum of which, which is (50).

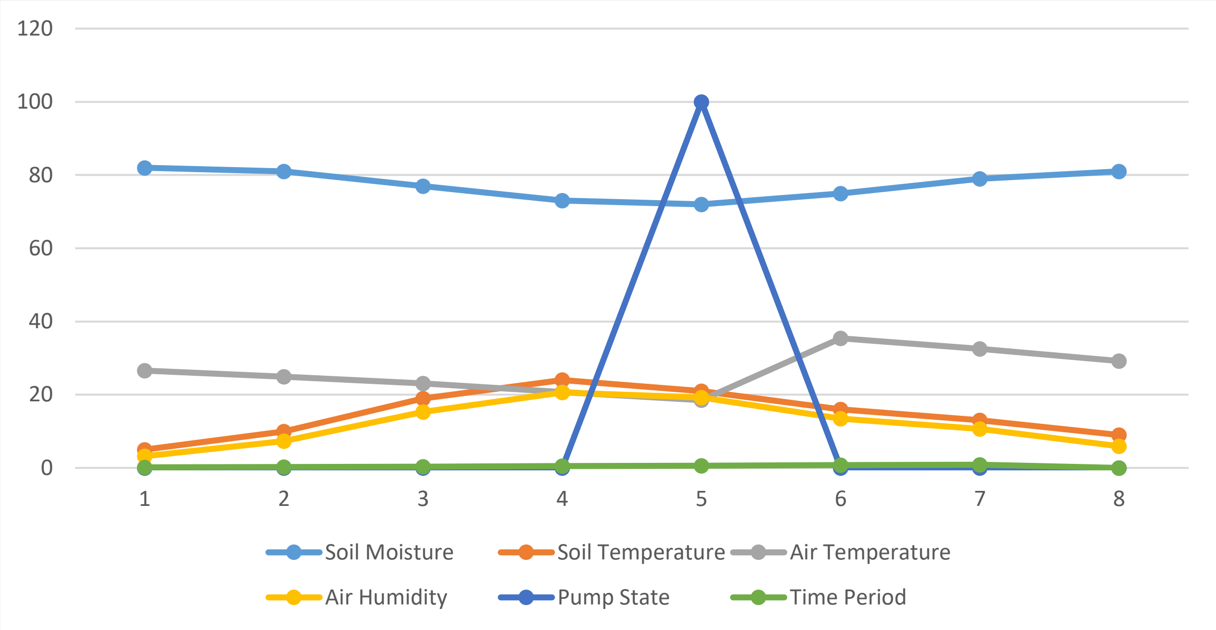


Figure 13. The relationship between reading sensors and determining the appropriate amount of irrigation MSM.

5. Conclusion

The proposed model in this work uses soil moisture, soil temperature, and surrounding temperature and humidity sensors to make irrigation judgments and decide the optimum quantity of irrigation. This study offered various contributions, including adopting a dynamic component in irrigation decision-making, meaning the threshold value is not constant but variable. It may be altered based on plant kind and requirement. The same is true for the appropriate amount of irrigation, which is determined by the relationship between soil temperature, air temperature, humidity, and soil moisture. Based on those values, the value at which soil irrigation will be stopped is chosen, and the watering mode can be manual or automatic.

In the future, it is planned to add rain and water level sensors, as well as an option to select the appropriate irrigation plan based on available quantity, such as basic, medium, or full irrigation, depending on how much water is in the tank and how much is needed. In addition, artificial intelligence and deep learning will be adopted to make irrigation decisions based on data entered into the system during training.

Declaration of Competing Interest

The authors declare no potential conflicts of interest related to the research, authorship, and publication of this article.

Ethical statements

The authors declare that this document does not require ethics committee approval or any special permission. Our study does not cause any harm to the environment and does not involve the use of animal or human subjects.

Authors' Contributions:

W.H.:50%

A.Z.:50%

All authors read and approved the final manuscript

References

[1] "AQUASTAT - FAO's Global Information System on Water and Agriculture," [Online]. Available: https://www.fao.org/aquastat/en/overview/methodology/water-use. [Access Date: 17.01.2022].

[2] L. García, "IoT-based smart irrigation systems: An overview on the recent trends on sensors and IoT systems for irrigation in precision agriculture," *Sensors*, vol. 20, p. 1042, 2020.

[3] J. F. Velasco-Muñoz, "Sustainable irrigation in agriculture: An analysis of global research," *Water*, vol. 11, p. 1758, 2019.

[4] M. Al-HUBAISHI, "SDN INTEGRATION FOR INTERNET OF THINGS," November 2019. [Online]. Available: https://tez.yok.gov.tr/UlusalTezMerkezi/. [Access Date: 23.01.2022].

[5] S. Kechiche, "IoT Connections Forecast: The Rise of Enterprise," 16 December 2019. [Online]. Available: https://data.gsmaintelligence.com/research/research/research-2019/iot-connections-forecast-the-rise-of-enterprise. [Access Date: 23.01.2022].

[6] C. Kamienski, "Smart water management platform: IoT-based precision irrigation for agriculture," *Sensors*, vol. 19, p. 276, 2019.

[7] N. K. Nawandar and V. R. Satpute, "IoT based low cost and intelligent module for smart irrigation system," *Computers and electronics in agriculture*, vol. 162, p. 979–990, 2019.

[8] M. Roopaei, P. Rad and K.-K. R. Choo, "Cloud of things in smart agriculture: Intelligent irrigation monitoring by thermal imaging," *IEEE Cloud computing*, vol. 4, p. 10–15, 2017.

[9] J. Kwok and Y. Sun, "A smart IoT-based irrigation system with automated plant recognition using deep learning," in Proceedings of the 10th international conference on computer modeling and Simulation, 2018.

[10] A. Goap, "An IoT-based smart irrigation management system using Machine learning and open source technologies," *Computers and electronics in agriculture*, vol. 155, p. 41–49, 2018.

[11] O. Friha, "Internet of Things for the Future of Smart Agriculture: A Comprehensive Survey of Emerging Technologies," *IEEE CAA J. Autom. Sinica*, vol. 8, p. 718–752, 2021.

[12] "About us," Raspberry Pi Foundation," [Online]. Available: https://www.raspberrypi.org/about/. [Access Date: 18.01.2022].

[13] L. Upton, "The Raspberry Pi in scientific research," [Online]. Available: https://www.raspberrypi.com/news/the-raspberry-pi-in-scientific-research/. [Access Date: 30.01.2022].

[14] P. Placidi, "Characterization of low-cost capacitive soil moisture sensors for IoT networks," *Sensors*, vol. 20, p. 3585, 2020.

[15] F. S. Muzdrikah, M. S. Nuha and F. A. Rizqi, "Calibration of capacitive soil moisture sensor (sku: Sen0193," in 2018 4th International Conference on Science and Technology (ICST, 2018.

[16] "Programmable Resolution 1-Wire Digital Thermometer (DS18B20) - Datasheet," [Online]. Available: https://datasheets.maximintegrated.com/en/ds/DS18B20.pdf. [Access Date: 30.01.2022].

[17] R. A. Koestoer, "A simple method for calibration of temperature sensor DS18B20 waterproof in oil bath based on Arduino data acquisition system," AIP Conference Proceedings, vol. 2062, 2019.

[18] A. I. Badran and M. Y. Kashmoola, "Smart Agriculture Using Internet of Things: A Survey," in IMDC-SDSP 2020: Proceedings of the 1st International Multi-Disciplinary Conference Theme: Sustainable Development and Smart Planning, IMDC-SDSP 2020, Cyberspace.

[19] "DHT 11 Humidity & Temperature - Datasheet," [Online]. Available: https://www.mouser.com/datasheet/2/758/DHT11-Technical-Data-Sheet-Translated-Version-1143054.pdf. [Access Date: 30.01.2022].

[20] A. Rayes and S. Salam, Internet of Things from hype to reality, Springer, 2017, p. 82–86.

[21] "Raspberry Pi Documentation - Raspberry Pi OS," [Online]. Available: https://www.raspberrypi.com/documentation/computers/os.html#introduction. [Access Date: 01.02.2022].

[22] G. Anand and A. K. Kumawat, "Object detection and position tracking in real time using Raspberry Pi," *Materials Today: Proceedings*, vol. 47, p. 3221–3226, 2021.

[23] "General Python FAQ," [Online]. Available: https://docs.python.org/3/faq/general.html#what-is-python. [Access Date: 01.02.2022].

[24] W. Donat and C. Krause, Learn Raspberry Pi Programming with Python, New York, NY 10013: Apress, 2014.

[25] D. Molnar and S. E. Schechter, "Self Hosting vs. Cloud Hosting: Accounting for the Security Impact of Hosting in the Cloud," WEIS, 2010.

[26] "What is cloud hosting?" [Online]. Available: https://www.ibm.com/cloud/learn/what-is-cloud-hosting. [Access Date: 01.02.2022].

[27] "Firebase Realtime Database | Firebase Documentation," [Online]. Available: https://firebase.google.com/docs/database/. [Access Date: 01.02.2022].

[28] L. Moroney, The firebase realtime database." The Definitive Guide to Firebase, Berkeley, CA: Apress, 2017, p. 51–71.

[29] L. Moroney, Using authentication in firebase." The Definitive Guide to Firebase, Berkeley, CA: Apress, 2017, p. 25–50.

[30] "Firebase Authentication | Firebase Documentation," [Online]. Available: https://firebase.google.com/docs/auth. [Access Date: 01.02.2022].

[31] W. Wu, React Native vs Flutter, Cross-platforms mobile application frameworks, 2018.

[32] G. I. Arb and K. Al-Majdi, "A Freights Status Management System Based on Dart and Flutter Programming Language," *Journal of Physics: Conference Series*, vol. 1530, 2020.

[33] "Modules — Python 3.7.12 documentation," [Online]. Available: https://docs.python.org/3.7/tutorial/modules.html. [Access Date: 07.02.2022].

[34] Y. Shekhar, "Intelligent IoT-based automated irrigation system," *International Journal of Applied Engineering Research*, vol. 12, p. 7306–7320, 2017.

[35] A. H. Blasi, M. A. Abbadi and R. Al-Huweimel, "Machine Learning Approach for an Automatic Irrigation System in Southern Jordan Valley," *Engineering, Technology & Applied Science Research*, vol. 11, p. 6609–6613, 2021.

[36] R. N. Rao and B. Sridhar, "IoT based smart crop-field monitoring and automation irrigation system," in 2018 2nd International Conference on Inventive Systems and Control (ICISC), 2018.