SAKARYA UNIVERSITY
INSTITUTE OF SCIENCE AND TECHNOLOGY

AN IOT BASED SMART IRRIGATION RECOMMENDATION SYSTEM

M.Sc. THESIS

Waseem HAMDOON

Department	**:**	**COMPUTER AND INFORMATION ENGINEERING**
Supervisor	**:**	**Prof. Dr. Ahmet ZENGIN**

August-2022

SAKARYA UNIVERSITY
INSTITUTE OF SCIENCE AND TECHNOLOGY

AN IOT BASED SMART IRRIGATION RECOMMENDATION SYSTEM

M.Sc. THESIS

Waseem HAMDOON

Department	**:**	**COMPUTER AND INFORMATION ENGINEERING**
Field of Science	**:**	**COMPUTER ENGINEERING**
Supervisor	**:**	**Prof. Dr. Ahmet ZENGIN**

This thesis has been accepted unanimously by the examination committee on 04.08.2022

DECLARATION

I declare that all the data in this thesis was obtained by myself in academic rules, all visual and written information and results were presented in accordance with academic and ethical rules, there is no distortion in the presented data, in case of utilizing other people's works they were refereed properly to scientific norms, the data presented in this thesis has not been used in any other thesis in this university or in any other university.

Waseem HAMDOON

~~00.00.2022~~

ACKNOWLEDGMENTS

Before and after everything, I thank Allah Almighty for the blessing of completing this work and for everything.

Then, according to the saying of the Prophet Muhammad, may Allah bless him and grant him peace, "He who does not thank people does not thank Allah". I would like to thank my supervisor, Dr. Ahmet Zengin, for his unlimited support and for what I learned from him in terms of knowledge and good character.

Words cannot describe how grateful I am for my dear father and dear mother for their support, encouragement, care, and supplication.

Thank and gratitude to my dear wife and children for their significant support and patience and all my family members and loved ones, especially my brother Dr. Omar.

TABLE OF CONTENTS

LIST OF SYMBOLS AND ABBREVIATIONS

°C	: Celsius Degree
°F	: Fahrenheit Degree
ADC	: Analog-to-Digital Converter
AOT-compiled	: Ahead-Of-Time compilation
AU	: Actuator Unit
BaaS	: Backend as a Service
BLE	: Bluetooth Low Energy
CPU	: Central Processing Unit
SoC	: Socket on Chip
CSI	: Camera Serial Interface
DBaaS	: DataBase as a Service
DHT11	: Digital Humidity and Temperature sensor
DSI	: Display Serial Interface
ESP32	: Espressif Systems
GPIO	: General-Purpose Input/Output
HDMI	: High-Definition Multimedia Interface
I2C	: Inter-Integrated Circuit
IaaS	: Infrastructure as a Service
ICT	: Information and Communication Technology
IoT	: Internet of Things
JSON	: JavaScript Object Notation
LAN	: Local Area Network
LED	: Light-Emitting Diode
LLVM	: Low Level Virtual Machine
LoRaWAN	: Long Range Wide Area Network
LPWAN	: Low Power Wide Area Network

Micro SD	: Micro Secure Digital
NDK	: Native Development Kit
NFC	: Near Field Communication
NTC	: Negative Temperature Coefficient
PA	: Precision Agriculture
PaaS	: Platform as a Service
RAM	: Random Access Memory
RDBMS	: Relational DataBase Management System
RFID	: Radio-Frequency Identification
ROM	: Read Only Memory
SaaS	: Software as a Service
SBC	: Single-Board Computer
SPS	: Samples Per Second
SQL	: Structured Query Language
SU	: Sensor Unit
USB	: Universal Serial Bus
VDC	: Volts Direct Current
Wi-Fi	: Wireless Fidelity

LIST OF FIGURES

LIST OF TABLES

SUMMARY

Keywords: Internet of Things, Smart Agriculture, Cloud Computing, Smart Irrigation, Actuators, Sensors

The largest proportion of fresh water around the world is consumed in agriculture. As a result, there has been a continuous interest in proposing systems and solutions that achieve rationalization of water resources in agriculture without negatively affecting productivity. On the contrary, the solutions provided must take into account raising and improving productivity by using the least amount of water resources. The Internet of Things is one of the prominent recent technologies that contribute to providing many solutions in most fields, including agriculture in general and irrigation specifically.

This study proposes a framework for smart irrigation based on the Internet of Things, by building a prototype consisting of a control unit, water pumps, and several sensors. These devices are used to measure the soil's need for irrigation and to determine the appropriate amount of irrigation according to sensor data. As these values are sent through cloud computing to a mobile application that is installed on the user's mobile to allow the possibility of monitoring irrigation and controlling it from several aspects. This includes manual irrigation mode, determining the value at which irrigation is done automatically in automatic irrigation mode, and the appropriate amount of irrigation according to the relationship between different sensors.

This thesis made several contributions, including the adoption of the dynamic feature, whether in making the irrigation decision according to the type of plant or in determining the appropriate amount of irrigation according to the relationship between the sensor readings of each plant. In addition, records of irrigation operations are stored in the automated mode for the purposes of feedback and future improvement on the proposed system. In future work, other sensors such as rain sensor and water level sensor will be added to increase the automation of the proposed model, in addition to using artificial intelligence and deep learning techniques to make irrigation decisions and determine the appropriate amount based on the data that is already stored in the current proposed system.

YENİ BİR IOT TABANLI AKILLI SULAMA SİSTEMİ

ÖZET

Anahtar Kelimeler: Nesnelerin İnterneti, Akıllı Tarım, Bulut Bilişim, Akıllı Sulama, Aktüatörler, Sensörler

Dünyada tatlı suyun büyük kısmı tarımda tüketilmektedir. Tarımda su kaynaklarının kullanımının verimliliğini sağlayan sistem ve çözümler sürekli geliştirilmektedir. Önerilen çözümlerde, su kaynaklarının daha az kullanımı ile verimliliğin artırılması dikkate alınmalıdır. Nesnelerin İnterneti, genel olarak tarım ve özel olarak sulama dahil olmak üzere çoğu alanda birçok çözüm sağlamaya katkıda bulunan en son teknolojilerden biridir.

Bu tezde, bir kontrol ünitesi, su pompaları ve birkaç sensörden oluşan bir prototip oluşturarak Nesnelerin İnterneti tabanlı akıllı bir sulama sistemi çerçevesi önerilmektedir. Sistem, toprağın sulama ihtiyacını ölçmek ve sensör verilerine göre uygun sulama miktarını belirlemek için çeşitli donanımlar kullanılır. Sensörlerden gelen değerler, bulut bilişim aracılığıyla kullanıcının cep telefonunda kurulu bir mobil uygulamaya gönderilir ve böylece sulamanın izlenmesi ve kontrol edilmesine olanak tanır. İlave olarak, manuel ve otomatik sulama modlarında otomatik olarak sulamanın yapılacağı değer belirlenir ve farklı sensörler arasındaki bağıntıya göre uygun sulama miktarı gerçekleştirilir.

Bu tez çalışması, bitki türüne göre sulama kararının verilmesinde veya her bitkinin sensör değerleri arasındaki ilişkiye göre uygun sulama miktarının belirlenmesinde dinamik bir karar destek sistemi kullanır. Önerilen sistemde geri bildirim ve gelecekteki iyileştirme amacıyla otomatik modda sulama operasyonlarının kayıtlarının saklanması sağlanmaktadır. Gelecekteki çalışmalarda, önerilen modelin otomasyonunu artırmak için yağmur sensörü ve su seviye sensörü gibi diğer sensörler eklenecek, ayrıca yapay zeka ve derin öğrenme teknikleri kullanılarak sulama kararları verilecek ve elde edilen verilere dayalı olarak uygun miktar belirlenecektir.

CHAPTER 1. INTRODUCTION

This chapter contains an introduction, a problem statement, and motivation, as well as a literature review, contributions to this dissertation, and the thesis organization.

The problem statement in this chapter defines the major issue that this study is attempting to address.

1.1. Problem Statement and Motivation

1.1.1. Problem statement

Poor water management in irrigation leads to waste of water, as well as a decrease in the quality of the crops produced, especially if the irrigation is uneven [1] and does not take into account the needs of each plant or crop separately.

There is often more than one type of plant in the fields and requires each type has a different amount of water [2].

Farmers frequently pump more water than is required (excessive irrigation), resulting in decreased output as well as waste of water and energy [3].

One of the most significant recent trends in increasing the efficiency of agricultural operations and reducing waste of resources is the use of the Internet of Things in irrigation management in particular [3]. On the other hand, the literature on smart irrigation using the Internet of Things, focus on answering two major questions: When is the irrigation decision taken? How to determine the appropriate amount of irrigation? 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 on 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.

1.1.2. Motivation

Agriculture is the primary source of food and the largest consumer of freshwater, taking up to 70% of all water resources [4]. In a related context, concerns about global warming, have prompted researchers into developing water management strategies to ensure that water is available for food production and consumption. As a result, there has been an upsurge in studies on lowering irrigation water use [5]. The concept of "sustainable irrigation" has also emerged, which is one of the main concerns for preserving water resources through the application of wise policies in rationalizing water consumption [6].

On the other hand, Internet of Things (IoT) has expanded the number of devices on the Internet, this is due to the requirement to get information from these devices for various IoT applications, and the number of such devices is likely to continue to rise [7]. By 2025, the number of connected devices is estimated to reach 25.1 billion [8]. The IoT offers varied applications to monitor crop growth and support irrigation decisions [9], 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, such as irrigation for agriculture, which is not yet fully automated [10].

Consequently, it is expected that IoT systems and solutions will become increasingly important in the design of appropriate systems and solutions for every field, including agriculture. Without irrigation water that is intelligently managed to take into account the needs of each plant, it will be impossible to achieve sustainable irrigation or to improve the quality of crops.

1.2. Literature Review

In recent years, following the widespread adoption of the Internet of Things, many studies have proposed systems and solutions to implement the IoT in agriculture in general and irrigation management in particular. Table 1.1 shows a list of studies on irrigation management using the IoT.

Table 1.1. Literature review for smart irrigation

Reference	Brief Description
Bhattacharya et al. [11]	This paper is proposed a smart irrigation system based on the IoT using a soil moisture sensor only to collect soil condition data in real-time and then send it to a wireless web server where the data is analyzed using an IoT framework and then sent periodic notifications to an application installed on the farmer's smartphone.
Shekhar et al. [12]	An intelligent system based on the IoT is proposed, which uses only two sensors: one to assess soil moisture and the other to measure soil temperature. The sensor data is processed by an intelligent algorithm, which predicts the best irrigation decision.
Mohammad A. Abbadi et al. [13]	A system has been proposed that depends on the decision tree algorithm in making the irrigation decision for the soil. A model for the irrigation process is designed based on the soil properties and temperatures, but according to the researchers, the results of the proposed model cannot be generalized to all regions because of their different characteristics from the South Jordan Wadi region where the model was developed.
Benzaouia et al. [14]	A smart irrigation system based on fuzzy logic control was proposed as a case study for the eastern Moroccan climate zone, where the soil moisture sensor was used and temperature and sun radiation were acquired from the meteorological station.

Table 1.1. (Continued)

Kamienski et al. [9]	The researchers proposed a smart IoT-based water management technology for precision irrigation in agriculture, based on four pilot projects in Brazil and Europe, although their project achieved good results, according to the researchers, it involves custom-designed settings and component re-engineering to provide greater scalability with using minimal computational resources.
Mohamed E. et al. [15]	A smart irrigation system based on the Internet of Things was proposed, which uses sensors to obtain temperature, humidity, and soil moisture, as well as a multi-layer neural network that works according to predefined rules to determine the amount of irrigation water to be pumped into the farm with the expected operating time.
Rao et al. [16]	An intelligent system for field monitoring and automation based on IoT technology is proposed, which uses two sensors: soil moisture sensors and soil temperature, with the threshold values chosen to calibrate the sensors based on temperature and soil moisture values over the previous months, threshold values may differ depending on the crop, where the irrigation water pump is operated and the monitoring is done using a computer application.
Laura et al. [5]	The researchers presented a study on smart irrigation systems based on the Internet of Things. The study is a summary of the recent technologies in sensors as well as Internet of Things (IoT) systems for watering in precision agriculture. The researchers worked to find the most observed parameters to describe the quality of irrigation water, soil, and air conditions. Also mentioned are the most common nodes used in IoT systems, wireless networks for irrigating crops, and the most common wireless communication. Current trends in using IoT systems for crop management and water management also addressed.
Nidia et al. [18]	The researchers proposed a smart and green IoT framework that uses water balance as well as potential metric approaches to implement an irrigation management plan based on the crop, as well as information about irrigation systems from users, weather and soil moisture data from areas close to the fields being monitored. For soil condition forecasting, a system has been developed that uses training models based on two different methods: locally and globally.

It is noted in the literature reviews that some of them use one or two sensors while ignoring the correlation between the sensors in determining the appropriate amount. Also, ignore some of them the need for each plant or crop to irrigate independently of the other. In addition, some of the studies mentioned above depend on a fixed threshold value to make the irrigation decision, so it is not easy to adjust. Accordingly, this study attempts to solve these problems by building a dynamic system that uses a sufficient number of sensors and allows changing the value at which the irrigation decision is taken; in addition to that, it takes into account the importance of the relationship between the sensors in determining the appropriate amount of irrigation.

1.3. Aims of The Thesis

The research aim of this thesis is to develop a framework for a smart irrigation management system based on the Internet of Things with the use of cloud computing and a mobile application on the end user's phone, so that the use of this system can provide the end-user with flexible options to irrigate each field or crop.

The options can be automatic irrigation, manual irrigation, setting the periodicity to ensure that the soil needs irrigation in automatic irrigation mode, moreover, the option to monitor the soil moisture and temperature, in addition to the temperature and humidity of the field in real-time.

1.4. Contribution of The Dissertation

The following considerations are prioritized in the development of the proposed system: ease of use of the mobile application by the end-user, flexibility in adjusting the sensor settings, and accuracy in estimating the appropriate irrigation time and amount. In light of the foregoing, the contributions in this study can be summarized as follows:

- The ability to make irrigation decisions for each field or crop separately from other crops.

- The possibility of determining the appropriate amount to irrigate each crop based on different factors other than the soil moisture scale, which are: soil temperature, humidity and air temperature.
- The ability to obtain real-time information about the surroundings of each field.
- The irrigation decision can be taken in two ways: automatically or manually, depending on sensor readings.
- In the automatic irrigation mode, the ability to set the appropriate periodic duration time to re-check that the soil requires irrigation.
- In addition to the ability to save logs of readings in the cloud, however, the values of previous readings are stored inside the console for review in the event of a network connection outage.

1.5. Thesis Organization

The following is the structure of the thesis:

- Chapter 2 covers the Internet of Things and its architecture in general, as well as cloud computing, in addition smart agriculture and applications.
- Chapter 3 introduces the techniques utilized to implement the proposed system, in the form of hardware and software.
- Chapter 4 focuses on the proposed system and describe its architecture and how to represent it in addition its demonstration.
- Chapter 5 concludes the thesis and gives the future work.

CHAPTER 2. INTERNET OF THINGS AND AGRICULTURAL APPLICATIONS

This chapter provides an overview of the IoT and its common architecture, as well as a concept of cloud computing and its close relationship to the IoT. Smart agriculture is also defined, with some IoT applications in smart agriculture.

2.1. Internet of Things

2.1.1. An overview of the internet of things

The Internet of Things (IoT) is one of the major technological advancements in the recent decade, it tends to give "things" intelligence by automating tasks and transferring data without human interaction.

Although the widespread of the Internet of Things and considered one of the aspects of the Fourth Industrial Revolution, it does not have an agreed comprehensive definition [19], where it is defined sometimes from the aspect of its application and sometimes from the aspect of its relationship to the Internet and so on. The following are two selected definitions that are intended to comprise the overview of the Internet of Things as follows:

- The phrase "Internet of Things" is a two-word phrase, and the first definition can be viewed from this perspective. So, the "Internet" is described as a global network of a vast number of networks based on communication protocol standards, whereas "Things" refers to all items that are connected to that network based on the same standards [20]. As a result, the connecting and exchanging of data over the internet between physical objects, or "things," embedded with sensors, software, and other technologies is known as the Internet of Things (IoT). These things can be anything from common household items to high-tech industrial tools [21].

- However, a more abstract description of "Internet of Things" can be found in the second definition, which is; The “Internet of Things” concept refers to anything that can be accessed from anywhere at any time by anybody for any service over any network [22].

2.1.2. The importance of the internet of things

Traditional lifestyles have been converted into high-tech ones as a result of the Internet of Things (IoT). Examples of IoT-driven transformations are smart homes, smart cities, smart transportation, and smart industries [23].

The Internet of Things (IoT) is becoming increasingly important because of the benefits it provides, including: improving production efficiency, enhancing the quality of life, preserving the environment, promoting the digital economy, reducing the waste of resources and times, improving the decision-making process through real-time data, one of the prominent aspects in achieving sustainability, and automation of processes and tasks.

2.1.3. Architecture of the internet of things

The Internet of Things has a variety of suggested designs divided to fundamental levels (see figure 2.1.).

The following is a five-layer basic architecture for the Internet of Things [19] that will be used in this research:

1. Perception Layer: to detect and collect data about the environment by using sensors at the physical layer[20].

2. Data Link Layer: Several protocols are used at this layer to establish communication for the following layer (the network layer) [24], such as Bluetooth, ZigBee, BLE, Wi-Fi, Z-Wave, RFID, Cellular, Sigfox, Ethernet, NFC, LPWAN, LoRaWAN.
3. Network Layer: It's in charge of assigning data pathways for transmission over the network [25]. Switches, routers, firewalls, and other network equipment are found on this layer.
4. Transport Layer: Packet delivery command, multiplexing, byte routing, congestion avoidance, data integrity, and reliability over transmitted data are all provided by the transport layer.
5. Application Layer: This layer represents the IoT architecture's front end, where the majority of the technology's potential will be realized, as it provides IoT developers with the interfaces, platforms, and tools needed to build IoT applications like smart cities, smart health, intelligent transportation, and smart homes [25].

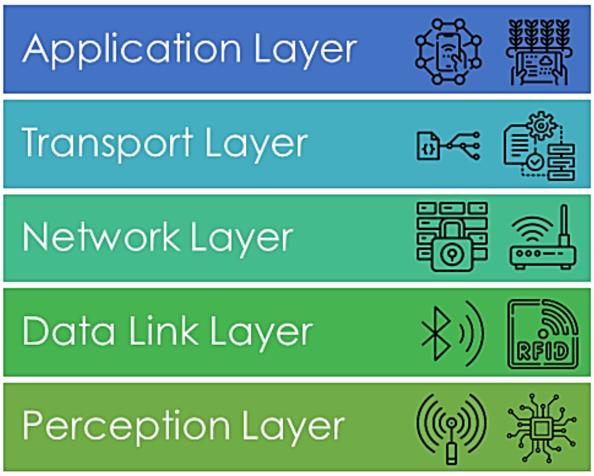


Figure 2.1. IoT Layers [25]

2.2. Cloud Computing

Cloud computing is a collection of virtual resources that may be accessed remotely, such as hardware, service platforms, or development platforms [26]. With the ability to dynamically reassign these resources to respond to changing loads, this leads to the optimum use of resources. A pay-per-use approach is commonly used to access this set of resources, with the Infrastructure Provider affording assurances through tailored SLAs [26].

Cloud computing aims to dynamically expand or reduce resources based on client load using software APIs with little service provider engagement [27].

Cloud computing provides a number of advantages, some of which can be highlighted as follows:

- It allowed small companies to benefit from business analytics that needed super computing resources, which were the preserve of large companies, at very low costs.
- Speed in marketing when computer resources are needed in any project, it is almost instantaneous access to those resources.
- It has greatly facilitated for institutions to expand the range of services provided, which depend in their work on customer requests.
- It introduced new categories of applications and services that did not exist before, such as interactive mobile applications that depend on real-time environment, context and location such as sensors, Additionally, parallel batch processing makes it possible to take advantage of extremely large computing power to evaluate large amounts of data in a short period of time, as well as business analysis that can use vast amounts of computer resources to understand customers and supply chains [28].

Clouds are usually classified into three basic types: private, public, and hybrid cloud [29]. A cloud condition might be a single cloud or a group of clouds. Customers frequently choose a cloud based on their ability to manage cloud frameworks and their security requirements. The physical foundation, the operating system platform, and the web application software being executed are the three tiers of a web server. One, two, or all of these layers may be present in a cloud container. SaaS (Software as a Service), PaaS (Platform as a Service), and IaaS (Infrastructure as a Service) are all examples of cloud service models. Hence, choosing a suitable cloud provider often boils down to deciding which layers you want to manage yourself and which should be left to the hosting provider [29].

The Internet of Things and cloud computing have a close relationship, as the Internet of Things has benefited from the advantages of cloud computing in terms of storage, processing, and communication. Development of many systems based on the Internet of Things with cloud computing are led into a variety of fields, including agriculture, since it combines IoT devices from sensors and actuators with the ability to store and follow up on the data created by it. The usage of the Internet of Things with the cloud has allowed the construction of high-speed information systems that function in real time or near to it [30] [31].

As seen in the Figure 2.2, Internet of Things (IoT) and cloud computing represent the next major step forward in the future internet [32].

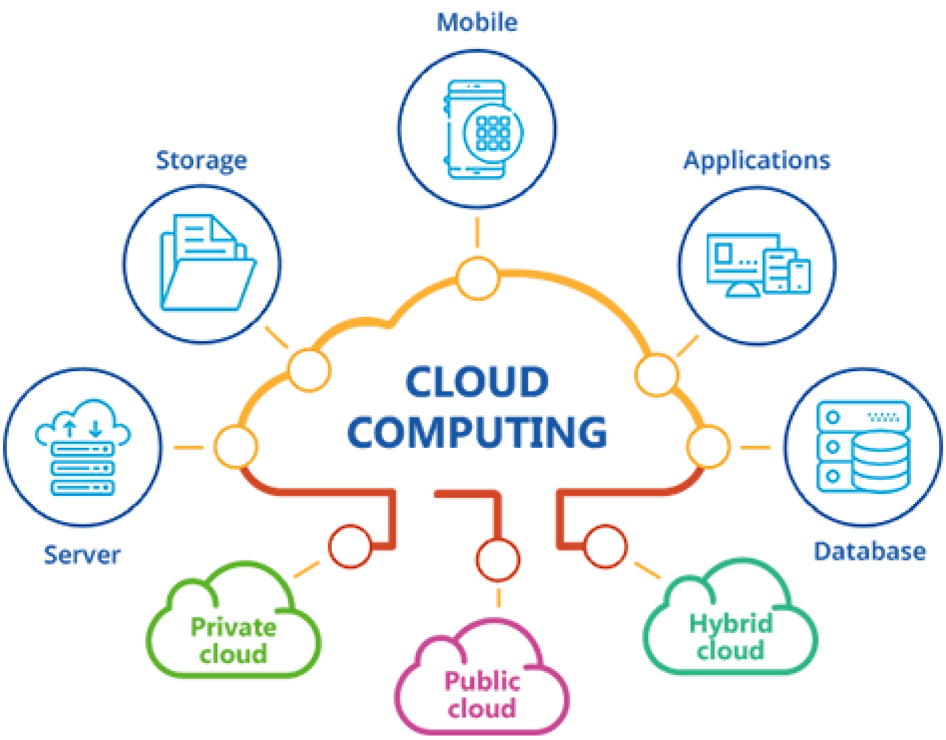


Figure 2.2. Cloud Computing [33]

2.3. Smart Agriculture

The rapid development of information and communication technology was reflected in most fields, including agriculture, where several terms appeared to denote new paths in agricultural technologies, including smart agriculture, electronic agriculture, and digital agriculture, where there are no fundamental differences except for some minor differences, according to the audience using these technologies [34].

Another broad term in this context is Precision Agriculture (PA) is a smart farming technique that enhances operational precision by providing each plant or animal exactly what it needs to thrive in the best possible way, maximizing overall performance while decreasing waste, inputs, and pollutants. In another word, is a complex technique that solely considers variables relevant to field conditions [31].

On the other hand, smart agriculture goes a step further by basing management responsibilities on data, which is reinforced by context and situational awareness and driven by real-time occurrences [35].

The Internet of Things, low-cost and enhanced sensors, actuators, and microprocessors, high-bandwidth wireless technologies, cloud ICT systems, big data analytics, artificial intelligence, and robots have all aided the progress of smart agriculture [31].

2.4. IoT Applications in Smart Agriculture

Similar to the rest of the sectors, it is expected that the agricultural sector would be one of the most prominent in benefiting from Internet of Things applications. Where the Internet of Things is used to deliver effective solutions for the agricultural industry, which contribute considerably to solving the issues of rising demand and workforce shortages in agriculture, as well as achieving agricultural sustainability [36].

The following are some examples of agricultural applications (see Figure 2.3):

- Smart water management.

- Smart monitoring of crops and the agricultural environment.
- Smart harvest support.
- Smart fertilization.
- Energy management in agriculture.
- Supply chain management in agricultural.
- Smart agricultural practices such as: Greenhouses, hydroponic, Aeroponics, Aquaponics, vertical agricultural, …, etc.

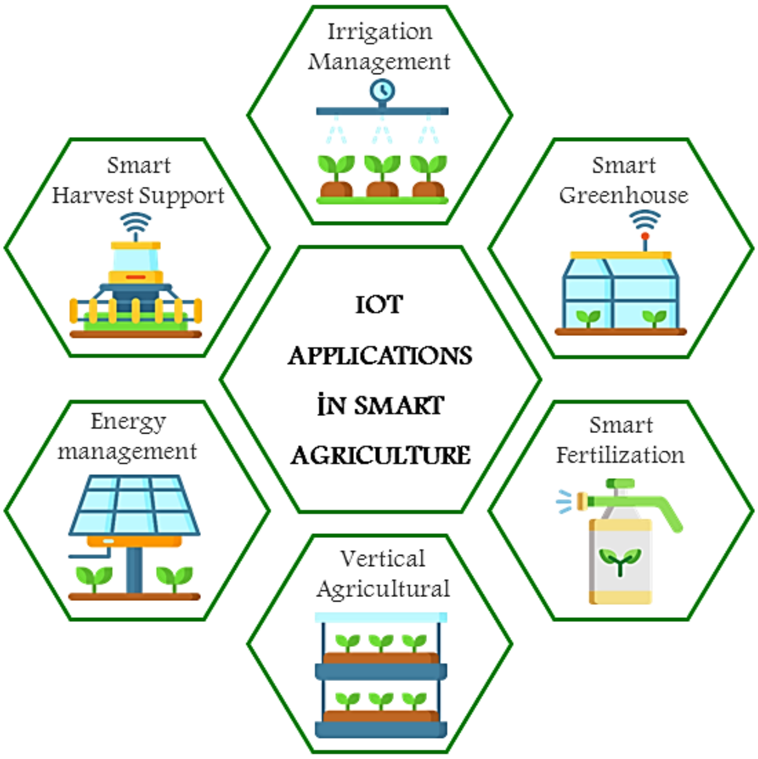


Figure 2.3. IoT Applications in Smart Agriculture

CHAPTER 3. HARDWARE AND SOFTWARE CONFIGURATION IN THE DEVELOPED APPLICATIONS

The techniques and device setup used in the proposed system are briefly discussed in this chapter.

3.1. Hardware

Most IoT-based systems are made up of three fundamental parts: a controller, a sensor, and an actuator.

3.1.1. Controller

It is usually used to automatically control devices and equipment [31]. 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” as shown in Figure 3.1. 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 [37] [38].

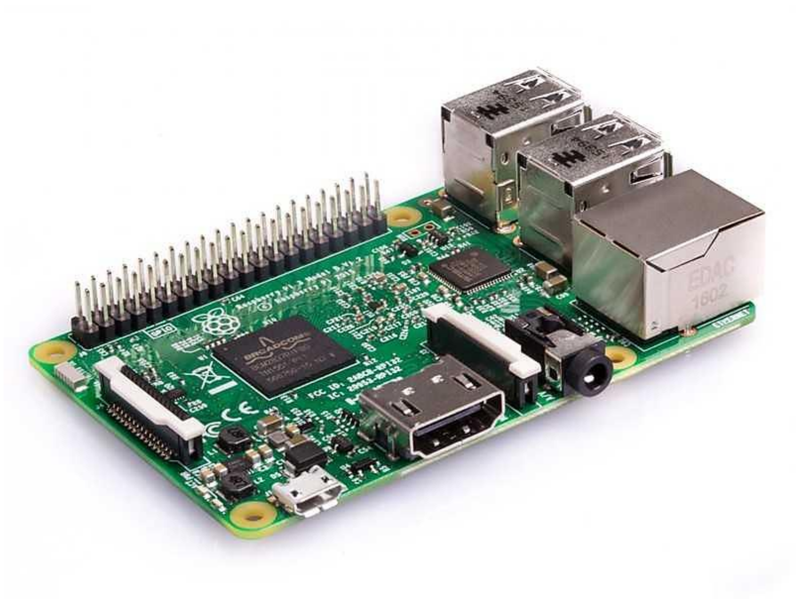


Figure 3.1. Raspberry Pi 3 Model B Figure [39]

Raspberry Pi 3 B Technical Specifications [39]:

- CPU: 1.2GHz 64bit Quad Core, SoC: Broadcom (BCM2837)
- RAM: 1GB
- Wireless: BCM43438, Bluetooth Low Energy (BLE) on board
- Ethernet: 100 Mbps
- GPIO: 40-pin
- USB: 4 ports
- Audio output: Headphone
- Video output: HDMI
- Camera Interface: CSI
- SD card port: Micro SD
- Power port: Micro USB, up to 2.5A

3.1.2. Sensors

Sensors are essential in automating any application since they collect data and process it. A sensor's complexity can range from the simplest to the most complicated. Sensors can be classified based on criteria including material type, conversion process, and the physical event they are intended to detect [31].

3.1.2.1. Soil moisture sensor

A “Capacitive Soil Moisture v2.0” sensor as shown in Figure 3.2. is used to measure the soil's moisture level. This capacitive sensor has an excellent responsiveness to local soil moisture changes and provides an effective correlation between gravimetric water content and output voltage [40] [41].

Its specifications can be summarized as follows [42]:

- Operating Voltage (VDC): 3.3 up to 5.5
- Output Voltage (VDC): 0 up to 3.0
- Current: 5mA
- Dimension: 98mm * 23mm (3.86in x 0.905in)
- Output type: Analog, hence an ADC converter is required to work with the Raspberry Pi.

Figure 3.2. Capacitive Soil Moisture v2.0 [42]

3.1.2.2. 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 hard environments such as thermometers, chemical solutions, industrial systems, soil, consumer products, and temperature sensing systems (see Figure 3.3). 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 [43] [44].

Its key specifications are as follows [43]:

- Temperature: between (-67°F to +257°F) or (-55°C up to +125°C)
- Accuracy range: ±0.5°C
- Operating voltage: 3V to 5V
- Communication: 1-Wire Interface (only one port pin)
- Conversion time: 750ms at 12-bit
- Resolution of the output (programmable): 9-bit up to 12-bit
- On-Board ROM stores a uniquely 64-bit serial number for each device.
- Communicates using 1-Wire method
- Enables multiplexing: Each sensor has a Unique 64-bit address

Figure 3.3. Soil Temperature Sensor (DS18B20) [43]

3.1.2.3. Air temperature and humidity sensor

As shown in Figure 3.4., sensor of type DHT11 is used to monitor and measure the temperature and humidity of the surrounding environment of a field or crop [44]. 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 [45] [46].

DHT11 specifications are as follows [46]:

- Voltage required for operation: 3.5V up to 5.5V
- Current required for operation: 0.3mA for measuring or 60uA for standby
- Output: Serial data
- Temperature: between 0°C and 50°C
- Accuracy range: ±1°C
- The Range of the Humidity: between 20% and 90%
- It has a resolution of 16-bits for Temperature and Humidity.

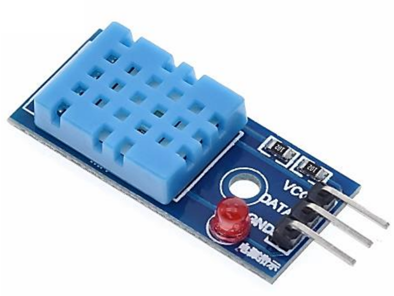


Figure 3.4. Air Temperature and Humidity Sensor (DHT11) [45]

3.1.3. Actuators

In the Internet of Things, actuators use data received by sensors and processed by software to control or take action in the system [47].

3.1.3.1. Water pump

A water pump is used for the purpose of irrigating the crop or field when the threshold limit is reached (see Figure 3.5.).

The water pump used has these specifications:

- Type: Submersible Pump
- Power: 5W
- Operating Voltage: 220V
- Frequency: 50Hz

- Max water temperature: 45
- High Max: 0.75M
- Dimensions: 35mm x 35mm x 30mm (1.37in x 1.37in x 1.18in)

Figure 3.5. Water Pump (Submersible) [48]

3.1.4. Other components

Other components are used, some of which are required and some of which are accessories.

3.1.4.1. Relay

Higher voltage modules can be controlled via relays on the Raspberry Pi. The relay "switch" is activated by a low-voltage pulse, it prevents the Pi from overheating because it can only handle 5V [49] (see Figure 3.6.).

Specifications of the relays used:

- Relay type: Toggle switch
- 2 Channel Relay Module with Optocoupler
- Each relay requires 15 - 20 mA for switching
- LEDs for indication of the relay status
- Operating Voltage: 3.3V or 5V
- Relay Maximum output: DC 30V/10A, AC 250V/10A

- Dimensions: 50 x 41 x 19 mm

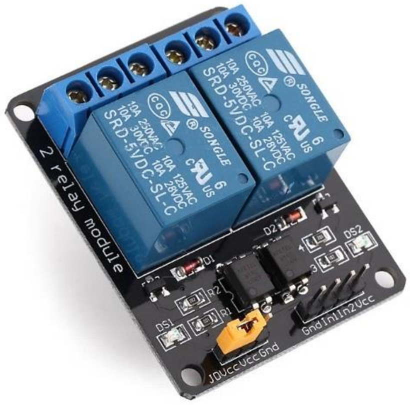


Figure 3.6. Relay Module with Optocoupler 5V (Dual Channels) [49]

3.1.4.2. Analog-to-digital converter

Since the Raspberry Pi 3B does not have an internal ADC, an external ADC is required to convert from analog to digital. The ADS1115 that shown in Figure 3.7. is used here as an external ADC.

The ADS1115 can be described as an analog to digital converter. It uses the I2C protocol and is four-channel with selectable addresses [50].

Specifications of the ADS1115 used:

- Resolution: 16 bits
- Voltage of operation (V): 2 to 5.5
- Data rate (Programmable): 8 to 860 SPS (samples per second)
- In continuous mode, it uses 150 µA, whereas in single-shot mode, it shuts down automatically.
- Reference voltage with low drift
- A built-in oscillator is utilized.
- Internally, there's a PGA (Programmable Gain Amplifier).
- The I2C interface allows for up to 4 pin-selectable addresses.
- Two differential channels or four single-ended channels.
- A comparator that can be programmed.

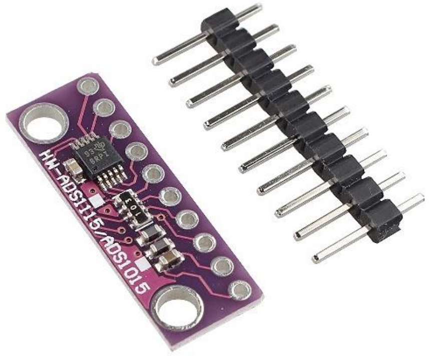


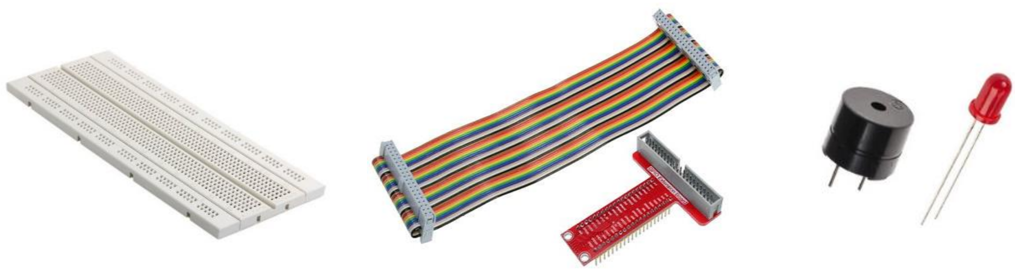
Figure 3.7. ADS1115 [51]

3.1.4.3. Accessories

The work can be made easier by a number of additional components that are not required.

The components as illustrated in Figure 3.8. can be summarized as follows:

- A breadboard: It’s a tool that is used to make electronic projects easier to work on.
- A Pi T-Cobbler Breakout for Raspberry Pi: It’s utilized in order to extend the Raspberry Pi's pins onto a breadboard.
- Red light-emitting diode (LED) and a passive buzzer: are used to add visual and auditory effects.



(a) (b) (c)

Figure 3.8. (a) Breadboard (b) GPIO T-Cobbler (c) LED and Buzzer

3.2. 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.

3.2.1. Raspberry pi requirements

3.2.1.1. Raspberry pi OS

Debian Linux distribution is the ideal operating system for Raspberry Pi devices, and regularly upgraded with an emphasis on reliability and performance; Also, it includes over 35,000 packages [52]. It's worth noting that it's based on the Debian (Linux distribution) and is likewise a free system [53].

Furthermore, users will find it easy to use due to the graphical interfaces that are given as shown in Figure 3.9.

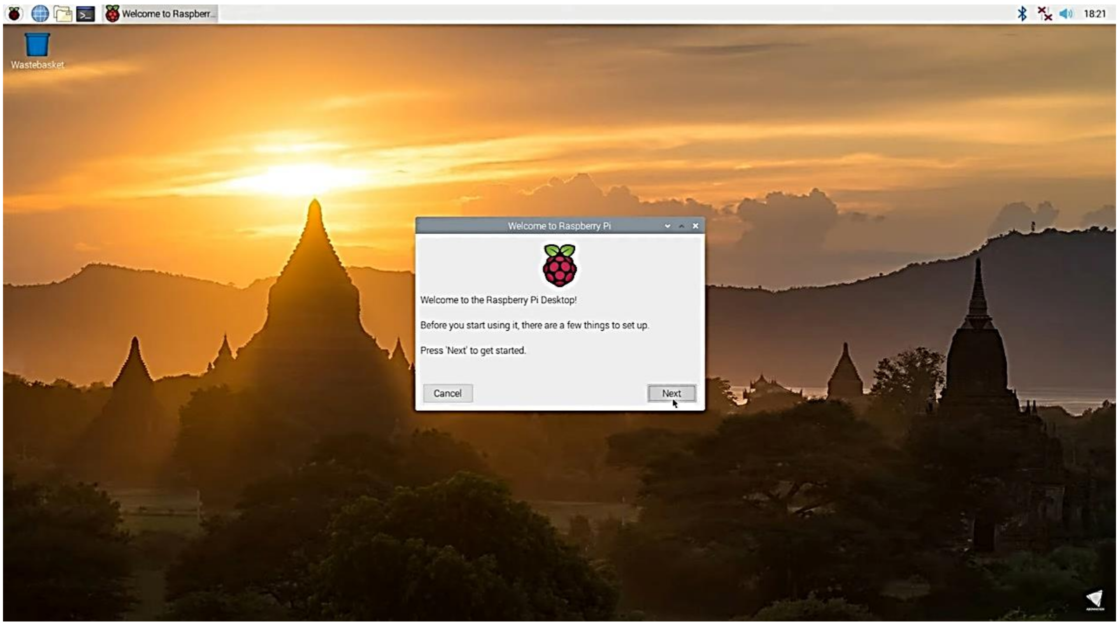


Figure 3.9. Raspberry Pi OS Desktop

3.2.1.2. Python

Python is a high-level programming language that combines power and clarity [54]. Due of this, it was included in the Raspberry Pi OS operating system since Python is a powerful tool that is easy to learn and use [55]. Version 3.7 is used in this research.

The following is a list of the most important features [56]:

- Simple language syntax makes it simple to learn.
- It is simple to read because it is punctuated-free.
- It's modular in order to make it simple to maintain.
- It has a large number of standard libraries that make it simple to integrate them into projects.
- It is interactive, as it includes a terminal for debugging and testing code snippets.
- It is not required to compile because is interpreted.
- It is portable in that it can run on a wide range of hardware platforms and has the same interface across all of them.
- It works on various operating systems such as Windows, Linux, Mac OS.
- Allows low-level modules to be added to the interpreter so that it can be customized.
- Versatile as it supports both procedural and object-oriented programming (OOP).
- Flexible, so that console programs, applications, and CGI (Common Gateway Interface) scripts to process web data and windowed graphical user interface (GUI) can be created.

3.2.1.3. TinyDB

TinyDB is a document-oriented, lightweight database, it’s written entirely in Python and does not require any external resources. its aimed at small apps that would be overwhelmed by a SQL database or an external database [57]. In addition, data is stored in a JSON file, which is suitable for dealing with the Internet of Things, as it is usually used to store important data and configuration data.

3.2.1.4. Visual studio code

It's a free and open-source code editor for creating and debugging desktop-based, modern web, and cloud applications. Microsoft introduced it, and it is regularly updated and developed [58]. It is quite popular among developers and supports a wide range of programming languages [59].

Furthermore, many add-ons are available for download to help with code management and handling [60].

3.2.2. Cloud hosting

Cloud hosting makes software and websites available via cloud resources [61], with minimal start-up costs, resource elasticity, and scale savings [62]. Firebase as cloud hosting services from Google is used in this research.

3.2.2.1. Firebase realtime database

It's a No-SQL, cloud-based database that syncs data in real time across all clients [63] [64].

The Firebase database has the following key features [65]:

- Data is stored as JSON,
- Offline functionality,
- In contrast to RDBMS, SQL is easily scalable to large database nodes.
- The scalability feature increases storage and performance.

3.2.2.2. Firebase authentication

In order to function properly, most applications require some type of user authentication, A user's identity may allow applications to securely store user data in the cloud and give a consistent, personalized experience across all of the user's devices, Firebase Authentication is tightly linked with the rest of the Firebase platform [66] [67].

3.2.3. Mobile application

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

3.2.3.1. Flutter

Mobile application development for various platforms has traditionally been tough since the developer must be superior to be able to design applications for Android and iOS. Therefore, the developers had to learn Java/Kotlin, Xcode, Swift, Eclipse, and other technologies at the same time [68] [63].

The Flutter framework from Google allows developers to create natively built, cross-platform mobile applications with a single codebase [69] [63]. As a result, the platform-independent emulator can run simultaneously on both Android and iOS devices.

Features of Flutter can be summarized as follows [70]:

- Open-source and free.
- It has a large community and resources.
- Each view component has its own rendering engine, which allows for applications to be that are as fast as native programs.
- It has a feature known as "hot-reload" that allows developers to quickly reload their work while they work.
- Languages used to build it are (C / C++, Dart), and uses the Skia Graphics Engine.
- C/C++ code is compiled with Android's NDK and iOS' LLVM, while Dart code is AOT-compiled into native code, see the (Figure 3.10.).

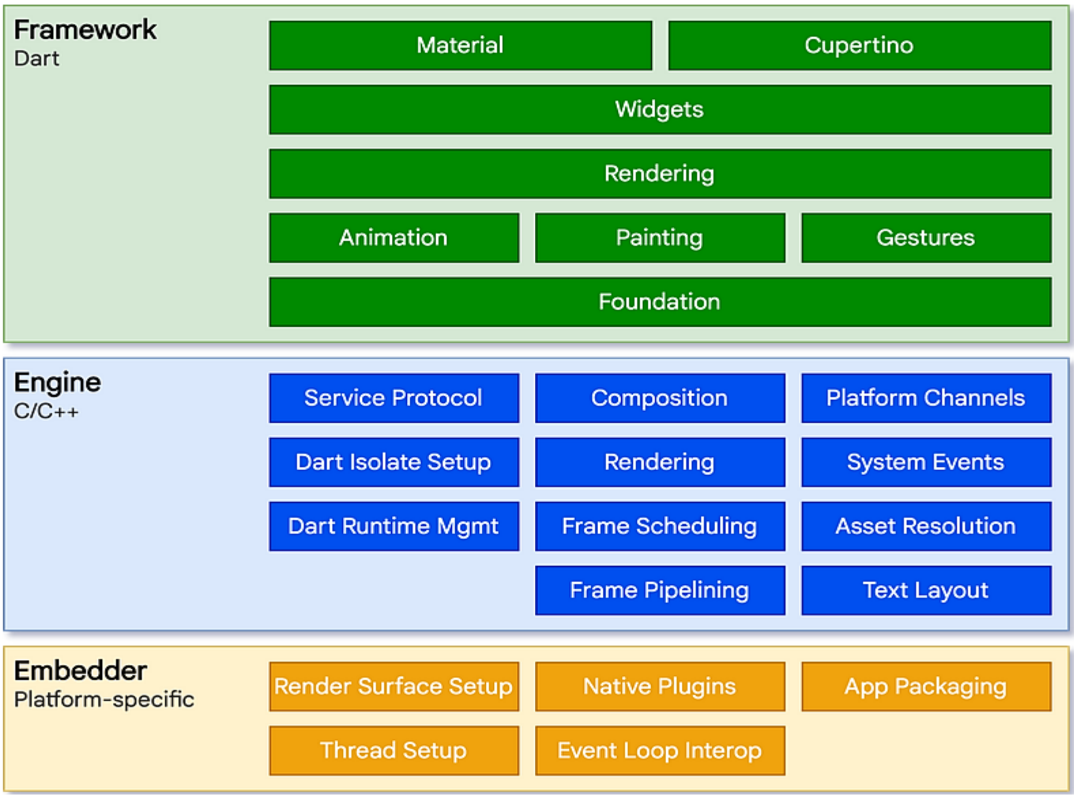


Figure 3.10. Architectural layers of Flutter [71]

3.2.3.2. Dart

In order to construct applications for Flutter, Dart is the programming language used in this thesis. It's developed and maintained by Google, it was originally intended as an alternative to JavaScript. Also, in an attempt to attract Java programmers, it uses a syntax that is similar to the Java programming language [72].

Dart has the following characteristics [73]:

- Easy to learn.
- Open source.
- Supports code sharing.
- Compiles to JavaScript.
- It has a lightweight text editor.
- Runs on the server and in the client.
- Types are supported, but they are not required.
- A large number of built-in libraries are available within it.

- With isolates, Dart allows for safe and simple concurrency.
- Small scripts and large, complicated applications can be written in Dart with ease.

CHAPTER 4. THE SMART IRRIGATION SYSTEM DESIGN

In this chapter, the proposed system architecture consisting of three layers will be shown and how each layer is implemented, and in the end, a demo application will be presented.

4.1. The Correlation Between Soil Moisture and Other Factors

The value of soil moisture in making irrigation decisions and calculating its amount is frequently used in Internet of Things irrigation systems. However, in addition to the soil moisture reading, additional factors influence the decision on whether irrigation is needed and the optimum amount to be applied. Among those influencing factors are: Soil temperature as all soil processes depend on temperature, Fertilizer efficiency seed germination plant growth, nutrient absorption, and decomposition diseases and insects are all affected by the soil thermal regime. In addition, for each crop there is an optimum temperature range is between two values called the lower and upper values, respectively. So, the soil temperature is a critical factor to be considered for effective irrigation scheduling [74] [75].

In the same context, the air temperature factor, which has a direct impact on soil temperature, can play a role in irrigation schedule. Furthermore, air humidity is a significant influence in irrigation scheduling, as the amount of water required for irrigation is lower when the weather is humidified according to dry. As a result, these factors must be considered together when making irrigation decision and irrigation amount.

On the other hand, those approved values must be dynamic and not fixed, that is, the required irrigation amount must be taken into account according to the type of each plant or crop independently from the other. It is important to take into account the dynamics of the required irrigation value according to the differences in those readings for the same plant.

In light of the foregoing, the proposed solution in the thesis took into account the different readings and not only the soil moisture reading, as well as taking into account the dynamic factor in the irrigation decision by determining a threshold value for each crop / plant. Also, the dynamic factor in the amount of irrigation was taken into consideration, which is the maximum value of soil moisture at which irrigation stops, through the establishment of four bands for sensors of soil temperature, air temperature and humidity, which allows determining this value dynamically with the possibility of adjusting it when needed as well.

4.2. 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 4.1.

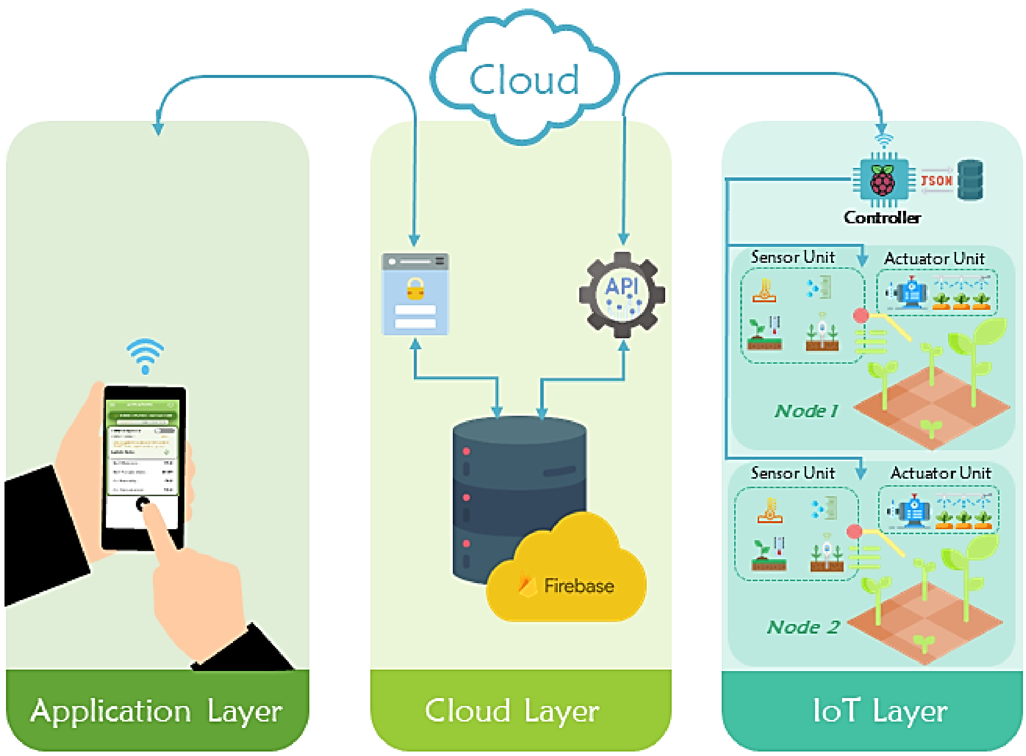


Figure 4.1. The architecture of the proposed smart irrigation system

Each layer of this architecture can be explained as follows. The Internet of Things (IoT) layer consists of software and hardware components that work together to represent this layer. The software consists of an irrigation management application that can be considered the firmware of this layer, in addition to libraries that facilitate dealing with the next layer, which is the cloud layer. As for the physical part in this layer, it consists of “Raspberry Pi 3 Model B” as a controller whose main job is to collect data from each node (a field or a plant) separately by the Sensor Unit (SU) and apply the appropriate decision by the Actuator Unit (AU). This model of Raspberry Pi also comes with a Wi-Fi module, which is necessary for the purpose of connecting to the Internet. An external memory is used to store configuration data, also to store the log of sensor readings and the status of each actuator. The sensors in (SU) used in each node are: a soil moisture sensor of a type "Capacitive Soil Moisture v2.0", a soil temperature sensor of a type "DS18B20", additionally a temperature and humidity sensor of the surrounding environment of a type "DHT11". While the Actuator Unit (AU) of each node consists of a water pump.

The cloud layer, which is the intermediate layer that connects the layer of the Internet of Things and the application layer, uses two cloud services from Firebase. These are: the Firebase Database Realtime service, which represents a database as a service (DBaaS) and its primary purpose is to store and retrieve data, and Firebase Authentication service, which represents a Backend as a Service (BaaS) to authenticate authorized users to access the database which hosted at the same layer using email and password.

The application layer comprises of a mobile application built with the Flutter framework and the Dart language for usage by the end-user in irrigation monitoring and control. The "Firebase Authentication" service in the cloud layer has been linked to this application's login screen in order to allow authorized access to the database of that layer.

4.3. System Implementation

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

4.3.1. IoT layer

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

a. Hardware implementation stage:

In order 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. Therefore, the items (sensors/components) of each node are connected to the GPIO pins as shown in the Table 4.1.

Table 4.1. A map of connecting components to the Raspberry Pi

Item	Item's pin	Intermediate Items	Raspberry Pi pin
Analog-to-Digital Converter (ADS1115)	V	------	Pin 1 – 3.3 V
	G	------	Pin 25 – GND
	SCL	------	Pin 5 – SCL1(I^2C)
	SDA	------	Pin 3 – SDA (I^2C)
Capacitive Soil Moisture v 2.0 [1]	GND	------	Pin 9 - GND
	VCC	------	Pin 17 – 3.3V
	AOUT	A0 (ADS1115)	------
Soil Temperature Sensor - DS18B20 [1, 2]	GND	------	Pin 14 - GND
	DATA	5k Ohm (Resistor 1)	Pin 7 – GPIO 4
	VDD	5k Ohm (Resistor 1)	Pin 17 – 3.3 V
Air Humidity & Temperature Sensor - DHT 11 [1]	VCC	------	Pin 2 – 5V
	DATA	------	Pin 18 - GPIO 18
	GND	------	Pin 6 – GND
Relay Module with Optocoupler (IN)	GND	------	Pin 6 - GND
	In 1 – Relay 1	------	Pin 32 – GPIO 12
	In 2 – Relay 2	------	Pin 38 – GPIO 20
	VCC	------	Pin 2 – 5V
Capacitive Soil Moisture v 2.0 [2]	GND	------	Pin 9 - GND
	VCC	------	Pin 17 – 3.3V
	AOUT	A1 (ADS1115)	------

Table 4.1. (continued)

Air Humidity & Temperature Sensor - DHT 11 [1]	VCC	------	Pin 2 – 5V
	DATA	------	Pin 23 - GPIO 23
	GND	------	Pin 6 - GND
LED	Cathode (-)	330 Ohm (Resistor 2)	Pin 9 – GND
	Anode (+)	------	Pin 36 – GPIO 16
Buzzer	Cathode (-)	------	Pin 9 – GND
	Anode (+)	------	Pin 37 – GPIO 26

On the other hand, the water pumps are connected to the relay module as indicated in the Table 4.2.

Table 4.2. Connecting the relay module to water pumps

AC Power Source	Relay Module (OUT)	Water Pump
VCC 220v	COM [Relay 1,2]	--------
-------	NO [Relay 1]	VCC [Water Pump 1]
-------	NO [Relay 2]	VCC [Water Pump 2]
GND	--------	GND [Water Pump 1,2]

b. Software implementation stage:

First of all, the operating system "Raspberry OS" is installed on the “Raspberry Pi 3 Model B” device, then The Visual Studio Code editor is installed to start writing an irrigation management application by Python 3.7 language.

Since it is good programming practice to divide large code into smaller files for easier code management and maintenance, the irrigation management application that runs on Raspberry Pi has been divided into files called modules in Python, meaning that these modules can be imported into any Python program [76].

Irrigation management application in the “IoT Layer” consists of several modules, each of these modules is used as shown in Table 4.3.:

Table 4.3. Modules for irrigation management application in the "IoT layer"

Module Name	Usage
IoT.py	To read from the sensor unit (SU) and send the appropriate command to the actuator unit (AU)
logger.py	To store sensors' readings and actuator status in the local memory of the “IoT layer”
firebase_utilties.py	To configure the connection to the “cloud layer” and CRUD operations
main_process.py	To implement the main scenario for irrigation management
Smart_Irrigation_Firmware.py	To execute the application

Another module called "calibration_Soil_Moisture_with_ADS1115.py" that must be used to calibrate each capacitive sensor of soil moisture before using it. Then, the minimum and maximum values for each sensor must be stored separately.

Furthermore, two irrigation management application files related are created to store data in the local memory in the “IoT layer” and use it as shown in the Table 4.4.

Table 4.4. Data storage files

File Name	Description
data_logger.json	To store sensor readings and actuator status
irrig_db.json	To store configuration settings

On the other hand, the flowchart in Figure 4.2. illustrates the work scenario of the irrigation management application in IoT layer.

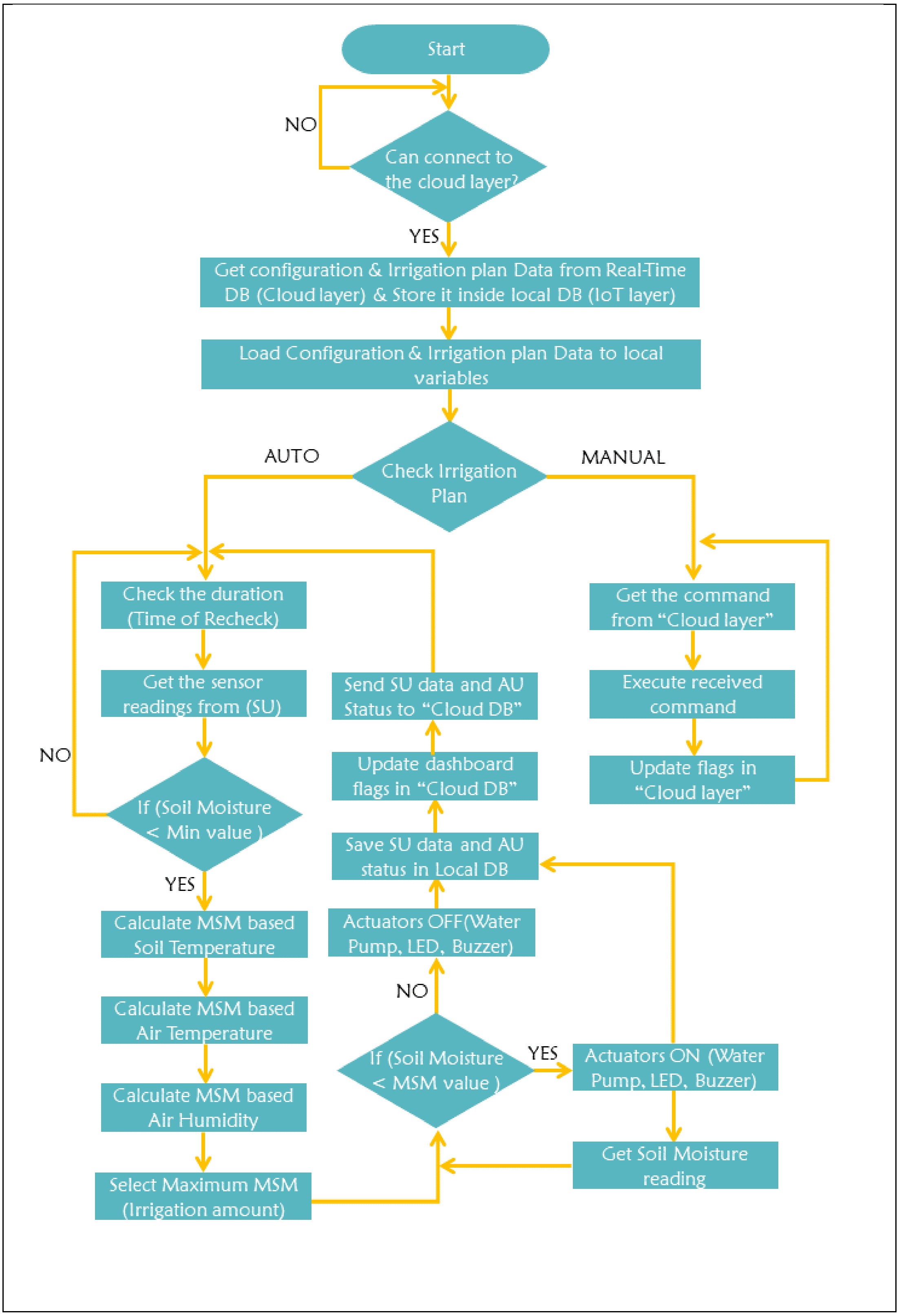


Figure 4.2. Flowchart of smart irrigation application in "IoT layer"

4.3.2. Cloud layer

Firebase allows the creation of an instance that can be used in a mobile application "application layer" as well as another instance that can be used in the "IoT layer". The main purpose of these instances is to provide access to data in real time so that when there is any update on the data stored in this database is all these updates reach users' devices in real time.

After this instance is created, it is included in the irrigation management application in the "IoT layer", and the same is true of the mobile application's instance.

However, in the "application layer", it is not directly connected to the cloud layer, but rather the verification service is passed first, in which the identity of the authorized user is verified by e-mail and password, and then the application can access the database data.

4.3.3. Application layer

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

In this layer, the mobile application has the following screens as in Table 4.5.

Table 4.5. Mobile application screens

Screen Name	Usage
Login	To create an account for the end-user and allow them to sign in to the application or sign into the application directly if an account already exists,and so access the real-time database in Firebase.
Dashboard	The application's home screen, from which the next five screens can be accessed or logged out.

Table 4.5. (continued)

Sensors	For each node, this screen displays readings from the sensors and the state of the actuators, as well as threshold values for each sensor that can be adjusted using this screen.
Actuators	To manually control the on/off of each water pump for each node, as well as to manually update the data with the ability to view the time and date of the last update operation.
Settings	To control the desired irrigation mode, automatic or manual, as well as set the "recheck" period in the automatic irrigation mode, In addition, each node (field/plant) can have a written description about it.
Logs	Displays the sensor data log and actuator status for each node with the date and time of each entry.

4.4. System Demonstration

In this section, each layer of the IoT-based smart irrigation system architecture will be demonstrated after its implementation as follows.

4.4.1. IoT layer:

The form of the hardware stage of the proposed system prototype in the IoT layer can be demonstrated in Figure 4.3.

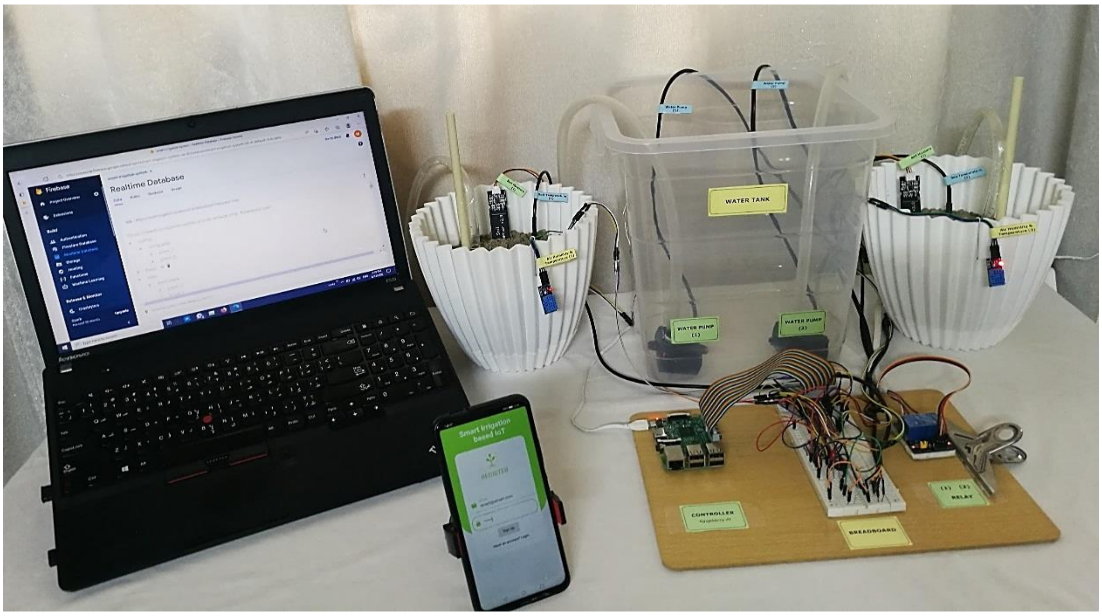


Figure 4.3. Prototype of the proposed system

As shown in the Figure 4.3., 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 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 automatic can be demonstrated in Figure 4.4.

```
----------------------------------------------------
[      STARTING THE SMART IRRIGATION SYSTEM        ]
----------------------------------------------------

Initial Step: Irrigation plan settings in the 'local database' are updated by firebase successfully.

Irrigation Mode: Auto ...

<<<<<<<<<<<<<<<< Node[ 1 ] >>>>>>>>>>>>>>>>

1. Start reading sensor data for plant ( 1 )...
  --> Current Soil moist is:  31.7  , and MSM is: 48.0
  --> Continue the previous state ..
2. Saving data in logger file ...
3. Sending data to firebase ...
  --> update last state data ...
  --> Insert new log data ...
------------------------------------------


<<<<<<<<<<<<<<<< Node[ 2 ] >>>>>>>>>>>>>>>>

1. Start reading sensor data for plant ( 2 )...
  --> Current Soil moist is:  8.2  , and MSM is: 50.0
  --> The plant ( 2 ) needs watering !!
  --> Plant ( 2 ) .. PUMP ON
2. Saving data in logger file ...
3. Sending data to firebase ...
  --> update last state data ...
  --> Insert new log data ...
```

Figure 4.4. Smart irrigation management application (IoT layer) - Auto irrigation mode

In contrast, and as shown in Figure 4.5., the flag state changes after receiving orders such as Pump On/Off, or update data.

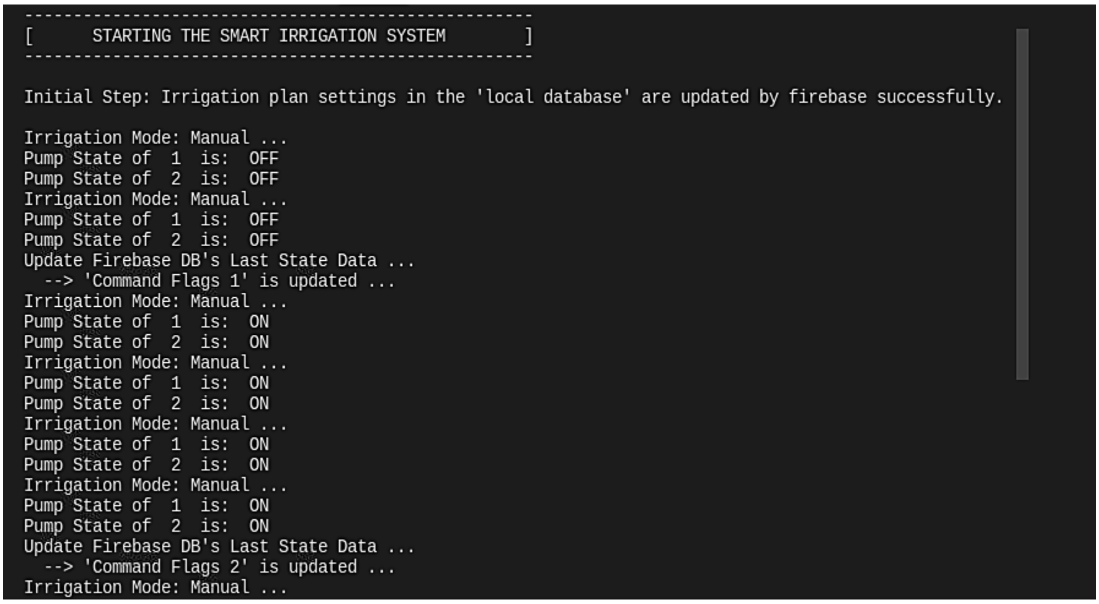


Figure 4.5. Smart irrigation management application (IoT layer) - Manual irrigation mode

4.4.2. Cloud layer:

Figure 4.6. illustrates how the Firebase real-time "No-SQL" database would look after the "Cloud layer" is implemented and connected to the IoT layer and application layer.

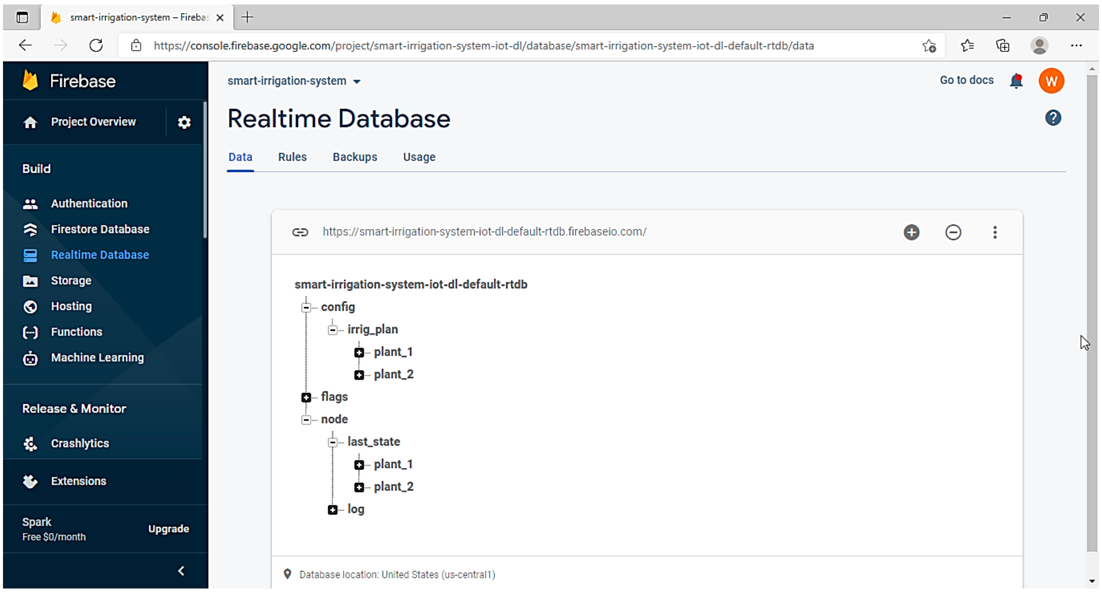


Figure 4.6. Firebase real-time "No-SQL" database

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, as seen in the Figure 4.7.

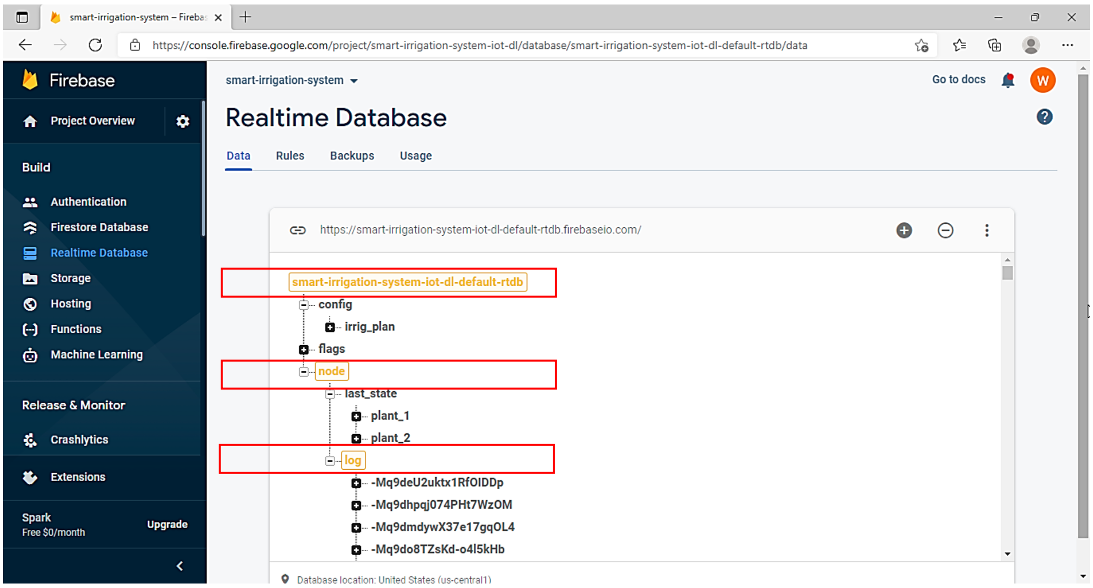


Figure 4.7. Highlighting changed data in Firebase real-time database

Some billable metrics (connections, storage, downloads) can also be accessed through the "Usage Tab" as illustrated in Figure 4.8.

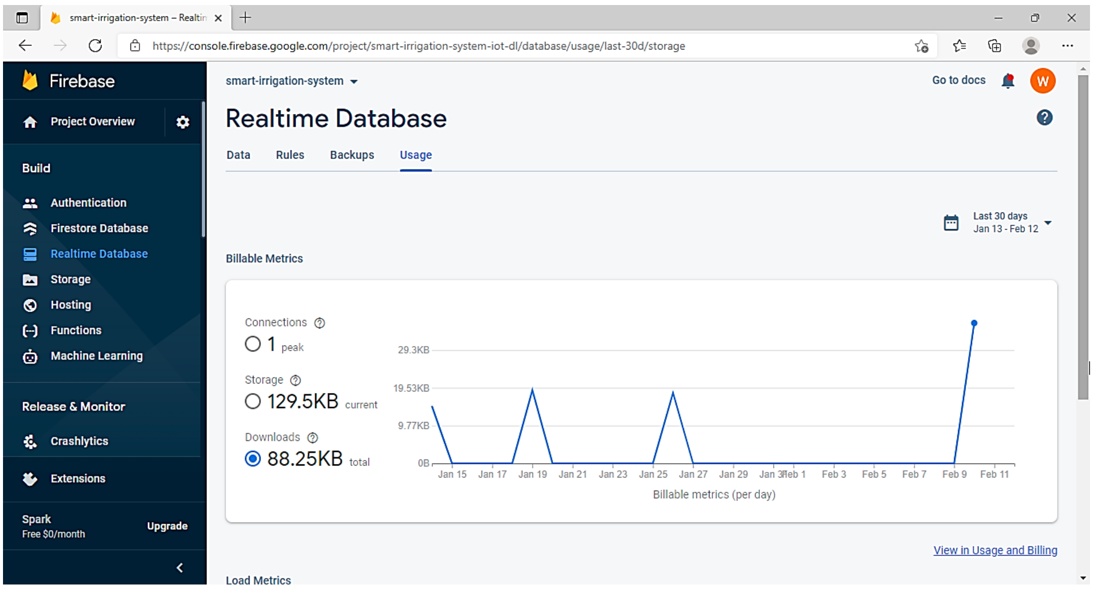


Figure 4.8. Billable metrics in the real-time database

On the other hand, the username/password authentication method is used to link the mobile application to the "cloud layer", as demonstrated in Figure 4.9.

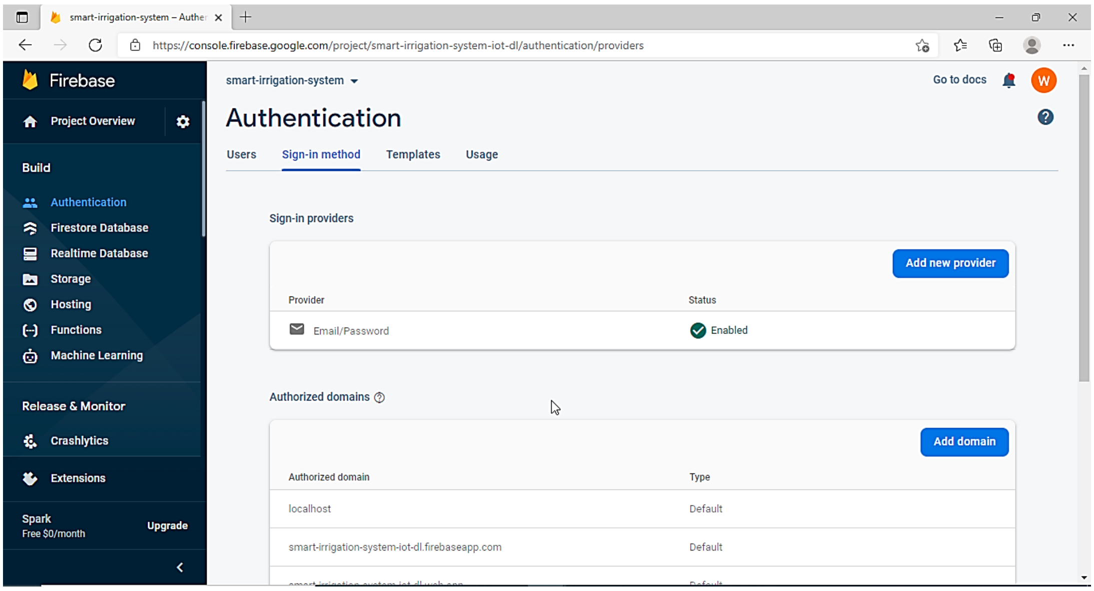


Figure 4.9. Authentication method

After the mobile application connects to the "cloud layer" and creates a new account for the first time, the created accounts are preserved as shown in the Figure 4.10.

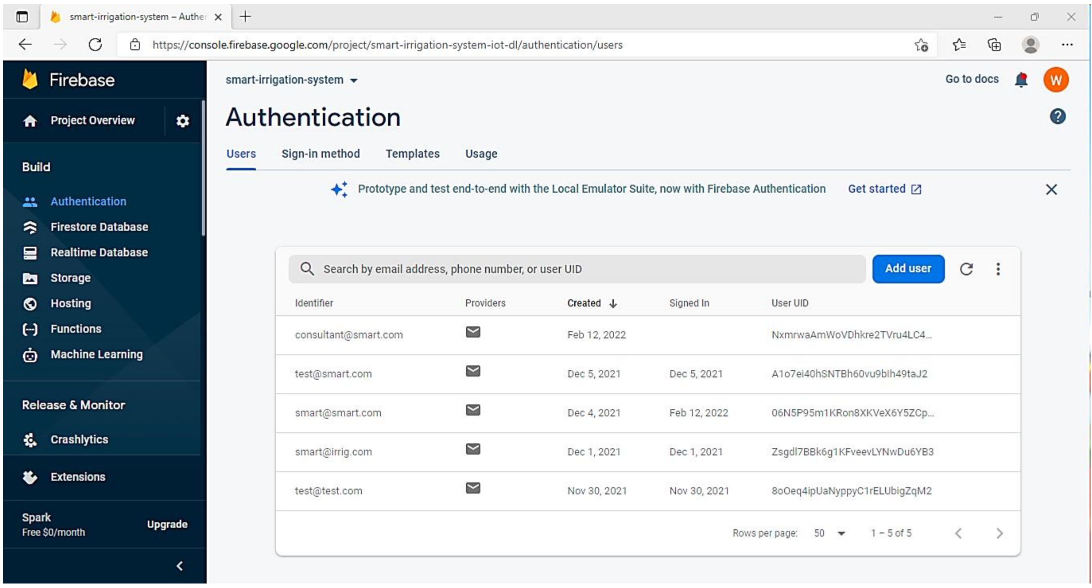


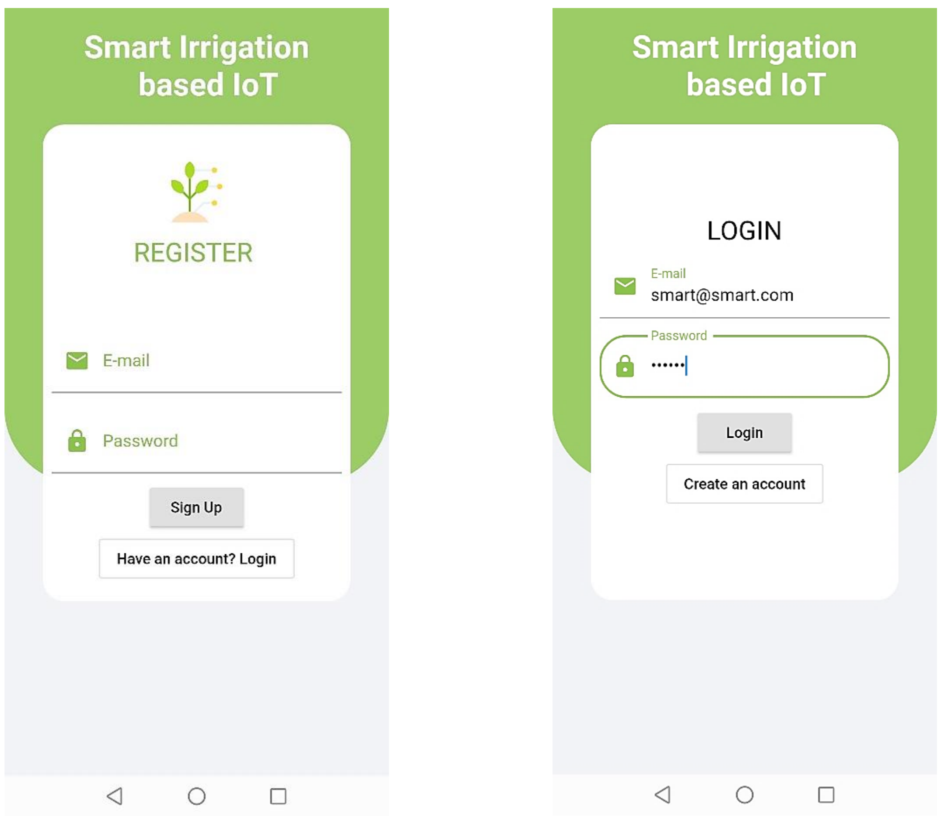
Figure 4.10. Users Identifiers using firebase Authentication

4.4.3. 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:

A new account can be created as indicated in (Figure 4.11. (a) Login screen – Sign up) by this screen. If an account and password are available, it is possible to log into the application through it as shown in (Figure 4.11. (b) Login screen – Log in). The registration data is sent to the authentication service of Firebase in the cloud layer, and if the user is authorized to enter, it can access the application. Therefore, it can access the data stored in the cloud database, which can be used to control and preview data from the IoT layer.



(a) (b)

Figure 4.11. (a) Login screen – Sign up, (b) Login screen – Login

- Main screen:

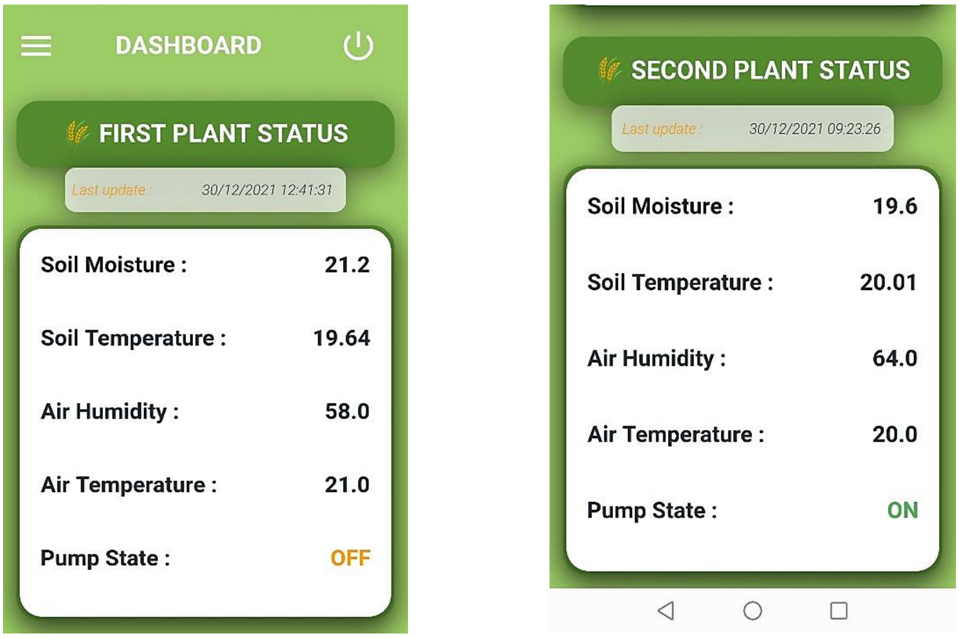
This screen appears after passing the entry; so that the user is taken to this screen, from where they can select the relevant screen from among the five available screens: the dashboard screen, the sensors screen, the actuators screen, the settings screen, and the logs screen, as seen in the Figure 4.12.

Figure 4.12. Main screen

- Dashboard screen:

This screen provides the end-user with the status and readings of the sensor unit represented by: the soil moisture sensor, soil temperature sensor, ambient temperature, and humidity sensor, in addition to the water pump status for each node (field/plant) (see Figure 4.13.).

On the other hand, this screen is divided into two sections, one for each node, and the data is updated in real-time in the automated irrigation mode, with the option to see the time and date of the update.



(a) (b)

Figure 4.13. Dashboard screen

- Sensors screen:

An end-user can view the threshold values and ranges for each sensor on this screen, which is demonstrated in Figure 4.14.

During the automatic irrigation mode, the critical reading for making the irrigation decision is the minimum value of the soil moisture sensor. This value represents the lowest value of soil moisture or percentage of water in the soil at the point at which the water pump is operated to irrigate the soil based on the reading value.

However, the high value of Maximum Soil Moisture (MSM), determines how much water is necessary for soil irrigation. When this value is achieved, the soil irrigation will be stopped; that is, the decision to turn off the pump is made on the basis of this value. This number indicates that the soil nodes have received the maximum or optimum amount of watering.

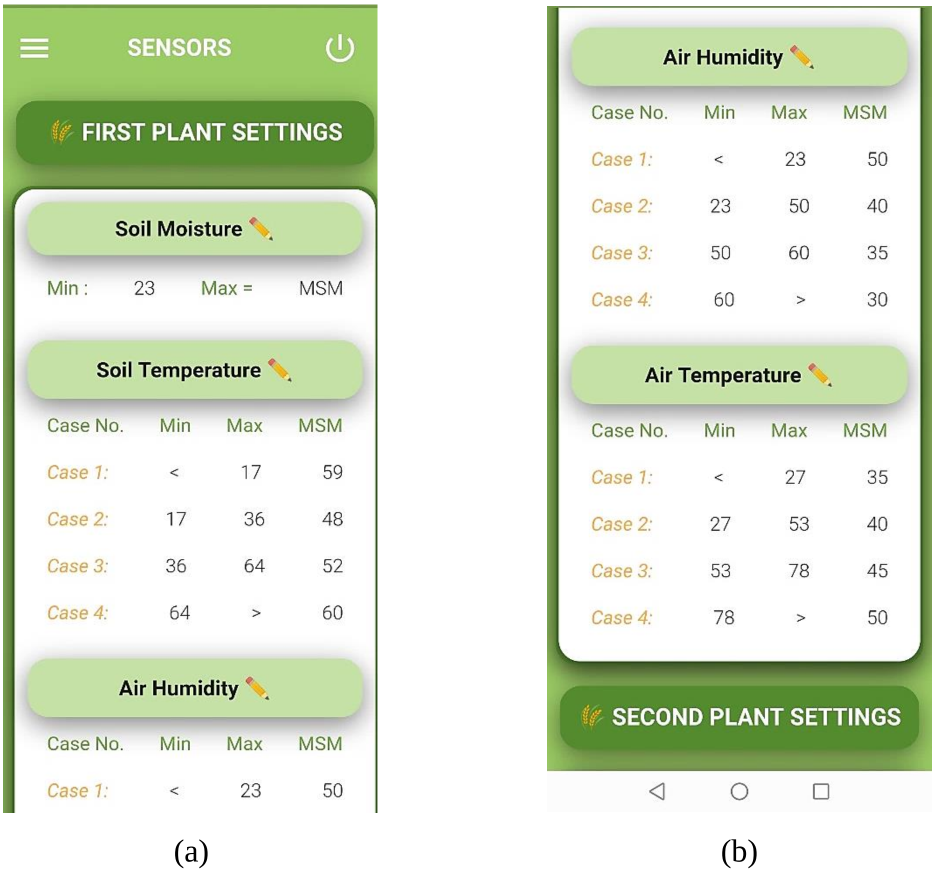
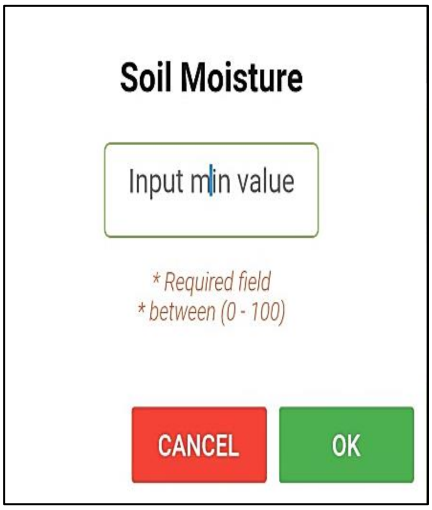
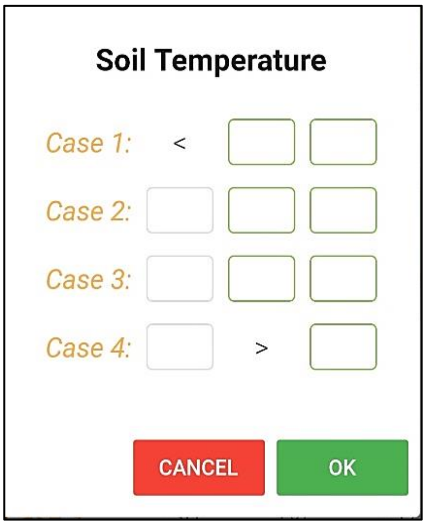


Figure 4.14. Sensors screen

In order to achieve dynamism in the irrigation decision, which is what is represented by the marginal value at which the irrigation decision will be taken, the possibility of specifying a value for each plant has been added, in addition to the possibility of adjusting it at any time, as shown in Figure 4.15. (a) The minimum value of soil moisture. In addition, the minimum and maximum sensor values for other sensors and for each level can be modified at any time as shown in Figure 4.15. (b) Ranges of soil temperature.





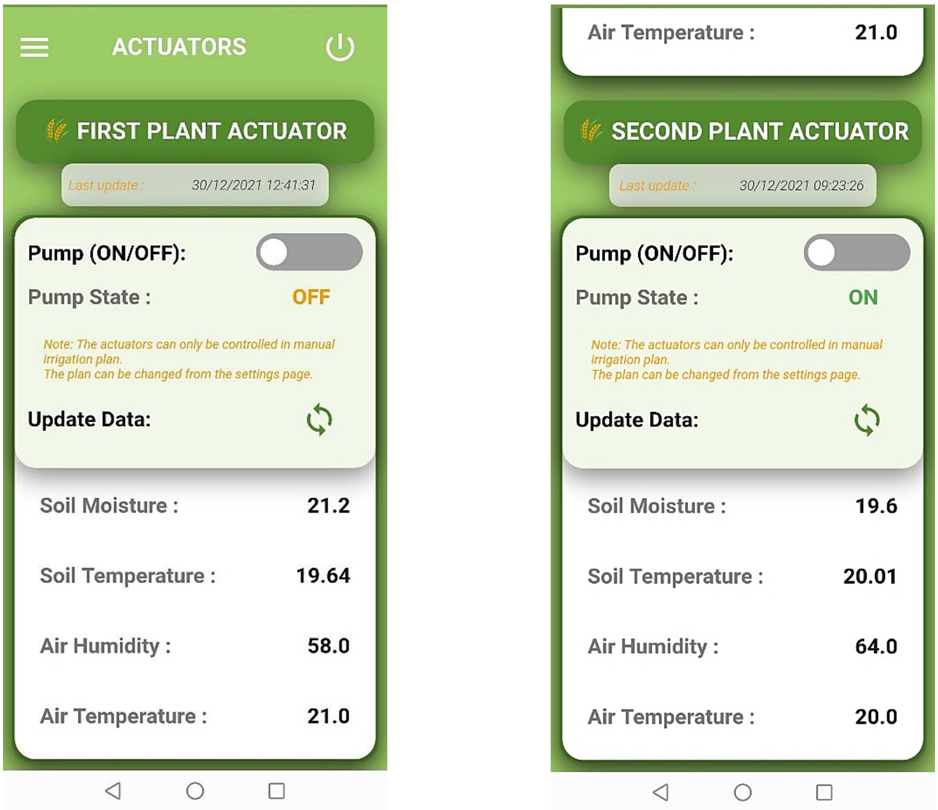
(a) (b)

Figure 4.15. (a) Minimum value of soil moisture, (b) Ranges of soil temperature

- Actuators screen:

This screen is important when irrigation mode is manual, each node (field/crop) has its own operator (water pump), as shown in Figure 4.16. It can be controlled manually through this screen, and the data can be updated manually. Observing the sensor readings, the status of the water pump, as well as display the time and date of the update are possible.

However, it should be noted that if the irrigation mode is automatic, it is not possible to control the turn on/off of the pump manually, as the on/off button of the water pump is disabled on this screen because the decision to control the operation of the pump is taken automatically.



(a) (b)

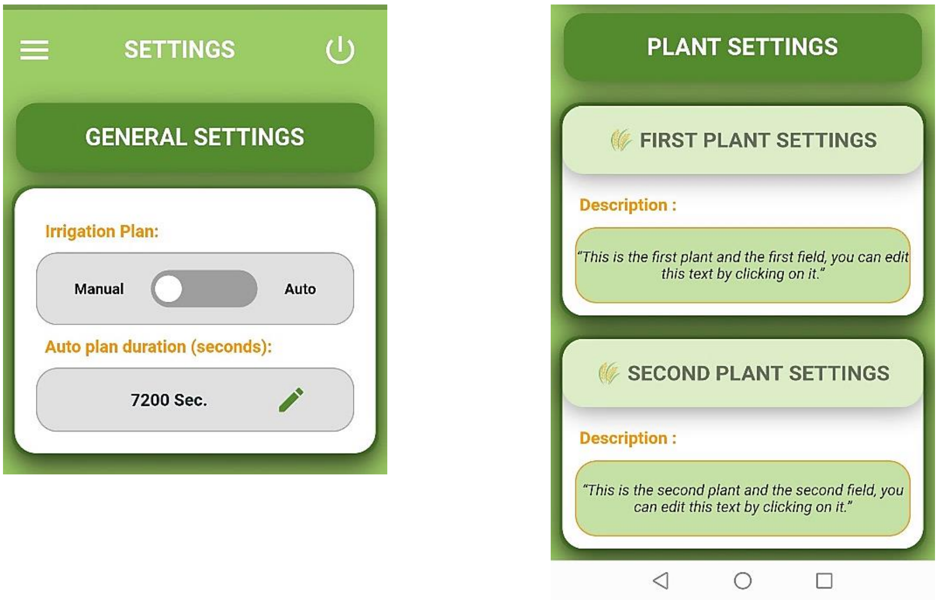
Figure 4.16. Actuators screen

- Settings screen:

One of the most important things about this screen is that you can choose the right irrigation plan. It can be switched between automatic and manual irrigation by clicking the irrigation plan button.

It is also possible to select the time period during the sensors will be reread to determine whether or not irrigation is needed, as well as the amount of irrigation that should be applied. This value has also been designed dynamically so that it can be modified according to the appropriate value at any time.

There can also be a description set for each node (field/plant) as shown in Figure 4.17. (b) to make it easier to identify the crop's name and other relevant parameters.



(a) (b)

Figure 4.17. (a) Settings screen – General settings, (b) Settings screen – Plant settings

- Logs screen:

This screen shows the records for each "Irrigation Need Check for each node in (Auto) Irrigation Mode".

Therefore, SU readings for each node (soil moisture sensor, soil temperature, ambient temperature, and humidity), AU unit or pump status, as well as the time stamp, are stored in the database as shown in Figure 4.18.

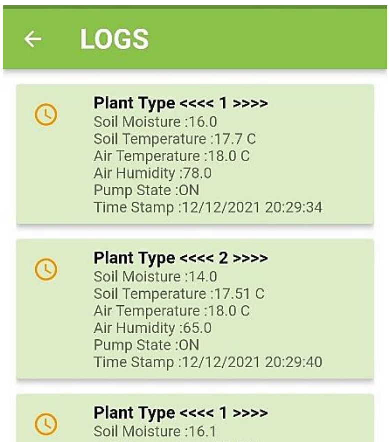


Figure 4.18. Logs screen

CHAPTER 5. CONCLUSIONS AND FUTURE WORK

In agriculture, rationalizing the use of water for irrigation and its intelligent management is one of the most important strategies for achieving long-term sustainability. This study deals with a general definition of the Internet of Things applications in smart agriculture in general and smart irrigation management in particular.

The sensors for soil moisture, soil temperature, and surrounding environment temperature and humidity are used in the proposed model in this study to make irrigation decisions and decide the appropriate amount of irrigation based on the information obtained from the sensors.

This thesis made several contributions, including its adoption of the dynamic factor in making the irrigation decision, meaning that the threshold value for taking the irrigation decision is not fixed, but rather variable, and it can be adjusted according to the type of plant and when needed. The same is true for the appropriate amount of irrigation, where the appropriate amount of irrigation is determined through the relationship of soil temperature readings, air temperature, and humidity with soil moisture, and based on those values, the value at which soil irrigation will be stopped is chosen, in addition to the possibility of the watering mode being manual or automatic.

In the future, we plan to add additional sensors, such as rain sensor and water level sensor, as well as an option to select the appropriate irrigation plan based on available amount, for example, basic irrigation, medium irrigation, or full irrigation based on how much water is in the tank and how much is needed.

Artificial intelligence and deep learning will be used to make irrigation decisions and decide the right amount based on the data that will be fed into the system during the training process.

REFERENCES

[1] 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.

[2] 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, pp. 87-91, 2018.

[3] 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.

[4] "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].

[5] 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.

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

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

[8] 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].

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

[10] 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.

[11] M. Bhattacharya, A. Roy and J. Pal, "Smart Irrigation System Using Internet of Things," Applications of Internet of Things, p. 119–129, 2021.

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

[13] 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.

[14] B. Mohammed, "An intelligent irrigation system based on fuzzy logic control: A case study for Moroccan oriental climate region," in 2nd international conference on Embedded Systems and Artificial Intelligence (ESAI'21), 2021.

[15] Karar, Mohamed E., et al. "IoT and neural network-based water pumping control system for smart irrigation." Information Sciences Letters, vol. 9, no. 2, pp. 107-112, 2020.

[16] 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), IEEE, pp. 478-483, 2018.

[17] L. M. Fernández-Ahumada, "Proposal for the design of monitoring and operating irrigation networks based on IoT, cloud computing and free hardware technologies," Sensors, vol. 19, p. 2318, 2019.

[18] G. S. Campos and Nidia, "Smart & green: An internet-of-things framework for smart irrigation," Sensors, vol. 20, p. 190, 2020.

[19] W. Kassab and K. A. Darabkh, "A–Z survey of Internet of Things: Architectures, protocols, applications, recent advances, future directions and recommendations," Journal of Network and Computer Applications, vol. 163, 2020.

[20] V. Aleksandrovičs, E. Filičevs and J. Kampars, "Internet of things: Structure, features and management," Information Technology and Management Science, vol. 19, p. 78–84, 2016.

[21] "What is IoT?," [Online]. Available: https://www.oracle.com/internet-of-things/what-is-iot/. [Access Date: 28.02.2022].

[22] D. G. INFSO, "Internet of Things in 2020: Roadmap for the Future," INFSO, vol. 4, 2008.

[23] S. Kumar, P. Tiwari and M. Zymbler, "Internet of Things is a revolutionary approach for future technology enhancement: a review," Journal of Big data, vol. 6, p. 1–21, 2019.

[24] P. Sethi and S. R. Sarangi, "Internet of things: architectures, protocols, and applications," Journal of Electrical and Computer Engineering, vol. 2017, pp. 1–25, 2017.

[25] M. R. Abdmeziem, D. Tandjaoui and I. Romdhani, "Architecting the internet of things: state of the art," Robots and Sensor Clouds, p. 55–75, 2016.

[26] L. M. Vaquero, "A break in the clouds: towards a cloud definition," ACM sigcomm computer communication review, vol. 39, p. 50–55, 2008.

[27] A. Dubey, D. Wagle and D. Jha, Delivering software as a service-The McKinsey Quarterly, 2007.

[28] M.-G. Avram, "Advantages and challenges of adopting cloud computing from an enterprise perspective," Procedia Technology, vol. 12, p. 529–534, 2014.

[29] D. Pooja, "Cloud computing-overview and its challenges," Int. J. Multidiscip, vol. 3085, p. 499–501, 2019.

[30] A. Botta, "Integration of cloud computing and internet of things: a survey," Future generation computer systems, vol. 56, p. 684–700, 2016.

[31] 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.

[32] L. M. Haji, "Impact of cloud computing and internet of things on the future internet," Technology Reports of Kansai University, vol. 62, p. 2179–2190, 2020.

[33] "Cloud computing is changing the way companies doing business," [Online]. Available: https://usdc.vn/cloud-computing-is-changing-the-way-companies-doing-business/. [Access Date: 07.05.2022].

[34] M. J. O'Grady, D. Langton and G. M. P. O'Hare, "Edge computing: A tractable model for smart agriculture?," Artificial Intelligence in Agriculture, vol. 3, p. 42–51, 2019.

[35] S. Wolfert, "Big data in smart farming–a review," Agricultural systems, vol. 153, p. 69–80, 2017.

[36] Mat, Ibrahim, et al. "Smart agriculture using internet of things." 2018 IEEE conference on open systems (ICOS). IEEE, pp. 54-59, 2018.

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

[38] 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].

[39] "Raspberry Pi 3 Model B," [Online]. Available: https://www.raspberrypi.com/products/raspberry-pi-3-model-b/. [Access Date: 30.01.2022].

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

[41] F. S. Muzdrikah, M. S. Nuha and F. A. Rizqi, "Calibration of capacitive soil moisture sensor (SKU:SEN0193)," 2018 4th International Conference on Science and Technology (ICST), pp. 1-6, 2018.

[42] "DFROBOT Capacitive Soil Moisture Sensor SKU:SEN0193 v.1.0," [Online]. Available: https://wiki.dfrobot.com/. [Access Date: 01.01.2022].

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

[44] R. A. Koestoer, "A simple method for calibration of temperature sensor DS18B20 waterproof in oil bath based on Arduino data acquisition system," In Proceedings of the AIP Conference Proceedings, Kuta, Indonesia, No. 1, vol. 2062, 2019.

[45] 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, Cyperspace, p. 10, 2020.

[46] "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].

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

[48] “JAINSONS PET PRODUCTS,” [Online]. Available: https://www.amazon.in/stores/JAINSONSPETPRODUCTS/. [Access Date: 07.05.2022].

[49] "Raspberry Pi: Control Relay switch via GPIO," 17 June 2018. [Online]. Available: https://tutorials-raspberrypi.com/raspberry-pi-control-relay-switch-via-gpio/. [Access Date: 01.01.2022].

[50] T. Insruments, "Ultra-Small, Low-Power, 16-Bit Analog-to-Digital Converter with Internal Reference," ADS1015 datasheet, May 2009.

[51] “ADS1015 12-Bit ADC,” [Online]. Available: https://www.pcboard.ca/ads1015-adc. [Access Date: 07.05.2022].

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

[53] 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.

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

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

[56] M. McGrath, "Python in easy steps," in Easy Steps, 2018.

[57] "Welcome to TinyDB! — TinyDB 4.6.1 documentation," [Online]. Available: https://tinydb.readthedocs.io/en/latest/. [Access Date: 03.02.2022].

[58] T. Kahlert and K. Giza, Visual Studio Code Tips & Tricks, vol. 1, Microsoft Deutschland GmbH, 2016.

[59] "Get Started Tutorial for Python in Visual Studio Code," [Online]. Available: https://code.visualstudio.com/docs/python/python-tutorial. [Access Date: 08.02.2022].

[60] M. Plainer, “Practical Study of Visual Studio Code Practical Course—Contributing to an Open-Source Project,” Semantic Scholar, 16-Mar-2021. [Online]. Available: https://www.semanticscholar.org/. [Accessed: 10-Aug-2022].

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

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

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

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

[65] A. M. Khachan and A. Özmen, "IMSSAP: After-school interactive mobile learning student support application," Computer Applications in Engineering Education, vol. 27, p. 543–552, 2019.

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

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

[68] R. Payne, Beginning App Development with Flutter: Create Cross-Platform Mobile Apps, Apress, 2019.

[69] W. Wu, React Native vs Flutter, Cross-platforms mobile application frameworks." Metropolia University of Applied Sciences, Finland, 2018.

[70] E. GÜLCÜOĞLU, A. B. USTUN and N. SEYHAN, "Comparison of Flutter and React Native Platforms," Journal of Internet Applications and Management, vol. 12, p. 129–143.

[71] "Flutter architectural overview," [Online]. Available: https://docs.flutter.dev/resources/architectural-overview.. [Access Date: 03.02.2022].

[72] 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, pp1-6, 2020.

[73] K. Walrath and S. Ladd, Dart: Up and Running: A New, Tool-Friendly Language for Structured Web Apps, O'Reilly Media, Inc.", 2012.

[74] A. A. Alli and O. E. Omofunmi, "A review of soil temperature under a controlled irrigation system," Journal of Research in Forestry, Wildlife and Environment, vol. 13, p. 50–59, 2021.

[75] J. Curiel Yuste, "Annual Q10 of soil respiration reflects plant phenological patterns as well as temperature sensitivity," Global Change Biology, vol. 10, p. 161–169, 2004.

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

RESUME

Name Surname : Waseem Mishaal Hamdoon

EDUCATION

Degree	School	Graduation Year
Bachelor	Mosul University/ College of Education	2010
Bachelor	Al-Hadba'a University College/ Computer Science	2005
High School	Al-Risalah High School	2000

JOB EXPERIENCE

Year	Place	Position
2012-2020	International Advanced Smart Systems company	Many Positions
2011-2012	Al-Maha company	IT
2006-2011	Pharmaceutical manufacturing company	Programmer

FOREIGN LANGUAGE

English

HOBBIES

Reading , Travel , Discovering New Technologies