ACKNOWLEDGMENT

The completion of this project represented the culmination of a seemingly fruitful result of extraordinary effort, dedication, and hard work. By efficiently resolving the various laborious problems and hurdles involved in the project, one can be certain that it is the result of the joint effort of all involved.

It is a matter of deep honor and gratification to express our profound sense of gratefulness for the ceaseless guidance, indispensable advice, and special time devoted to us by our supervisor, Mrs. Imen Ben Chaaban, our professor at ISET'Com. The success of this project can be attributed to her constant support, kind guidance, and valuable recommendations.

Finally, we are thankful to everyone who contributed to our efforts especially all ISET'Com professors for the quality of the training they provided us and also the members of the jury for agreeing to judge this modest work.

Contents

List of figures

List of tables

List of abbreviation

IOT : internet of things.

GSM : Global System for Mobile Communication.

GPRS :General Packet Radio Services.

LORA : Long Range Wide Area.

MUC : Microcontroller unit.

RFID : the Radio Frequency Identification.

DAS: Data Acquisition Systems.

LCD : Liquid crystal display.

IDE : Integrated development environment.

GENERAL INTRODUCTION

One of the main factors contributing to water shortages in Tunisia is the country's arid and semi-arid climate, which limits the availability of natural surface water and groundwater resources. In addition, the increasing demand for water for agriculture, industry, and urbanization has put pressure on these resources.

To address these challenges, Tunisia has implemented a range of measures to improve water management in the agricultural sector. These measures include the usage of water-saving technologies, like implementing water pricing and allocation policies to encourage more efficient use of water and irrigation.

Traditional irrigation systems often apply a uniform amount of water to an entire irrigation system, regardless of the specific needs of the plants or the weather conditions. This can result in over- or under-watering, which can lead to unhealthy or dying plants and waste valuable resources, on the other hand smart irrigation systems are used to optimize the watering of plants and crops in order to conserve water and other resources.

In these conditions, farmers can better exploit water resources through mobile applications by using sensors and controllers integrated with the Internet of Things. It was therefore decided to carry out this project within this context.

This report will be divided into three chapters the first will be dedicated to present the context of the project, the second will describe the conception part, and the last chapter is for the implementation of the solution.

Chapter 1: Context of the project

Introduction

Our first chapter will introduce the main problem and the different systems proposed to solve it, as well as our proposed solution. After that, the technology used, IoT, and its different applications will be presented.

1. Problematic and study of the existing

1.1. Problematic

In the arid and semi-arid regions, countries like Tunisia face increasingly severe water shortage problems. Problems of water scarcity will intensify because of population growth, a rise in living standards, and accelerate deurbanization, which threaten the water supply in general and agriculture in particular and lead to both an increase in water consumption and pollution of water resources.

The continuing increase in demand by the urban sector has led to increased use of fresh water for domestic purposes on the one hand and the production of greater volumes of wastewater on the other.

Agriculture, in competition with other sectors, will face increasing problems of water quantity and quality, considering increasingly limited conventional water resources, growing future requirements, and a decrease in the volume of fresh water available for agriculture.

1.2. Study of the existing system

There is a variety of smart irrigation systems available, ranging from simple, standalone systems to complex, networked systems that can be controlled and monitored remotely.

A primary investigation is administered under the subsequent stages, like understanding the existing approaches, understanding the wants, and developing an abstract for the system.

Table 1 : Comparison table for some projects.

Projects	Abstract	Objectives	The material used	Type of connection used
AI-Based Yield prediction and smart irrigation[1]	This project presents different techniques and applications of artificial intelligence for yield prediction and smart irrigation.	-Optimum utilization of water. - Providing complete irrigation data through cloud computing. - Crop control by growth or yield. - Give the farmer the time to put the fertilizer	-Raspberry Pi - Soil moisture sensor - Temperature sensor - Smart sensor array for cotton field	GSM module
Control System for Greenhouse[2]	this project deal with the remote Measurement and Control System for Greenhouse Based on GSM-SMS	-remote measurement - remote control system for greenhouse based on PC-based -database system connected with base station - Criterion value of parameters to be measured in every base station is set by central station	-microcontroller -GSM module -sensors and actuators.	GSM-SMS
Based rain gun irrigation system[3]	In this project the irrigation will take place only when there will be intense requirement of water that save a large quantity of water.	- use automatic microcontroller in the irrigation process where it takes place only when needed	- Arduino Uno. - Two YL-69 soil moisture sensors. - usage of an operating system - the GPRS feature	-GSM module -GPRS

The table above shows a detailed comparison for some projects .

As we studied the related work presented, we realized that most of these systems used GSM, which has become less popular as a connected system as well as being economically unaffordable.

To solve these issues we need to make sure these objectives are achieved:

- ✓ Smart irrigation based on careful study
- ✓ Rationalize the use of water
- ✓ Work at the convenience of the farmer
- ✓ Improving product quality
- ✓ Saving energy used in traditional irrigation as and the cost
- ✓ Reducing or erasing human errors

In addition, we have also focused on the simplicity thanks to Lora technique as an efficient communication and connection technology of our solution as well as on reducing costs.

1.3. Proposed System

The system is a combination of hardware and software components. The hardware part consists of an embedded system that consists of three sensors connected to a controller, sensed and are displayed in a mobile application designed using MIT app inventor which is our software component.

The value of the conditions for soil moisture (if it is Wet, Dry, or Soggy Wet), is encoded in the MCU which is a platform device called Arduino Uno. When the soil moisture sensor detects the conditions of the soil, the MCU will power up the water pump and delivers the right amount of water to the plants. For the water source, there is a water level indicator installed that indicates if the water is at a high or already at a low level.

The mobile application is hosted online and consists of a database in which readings from sensors are inserted using the hardware.

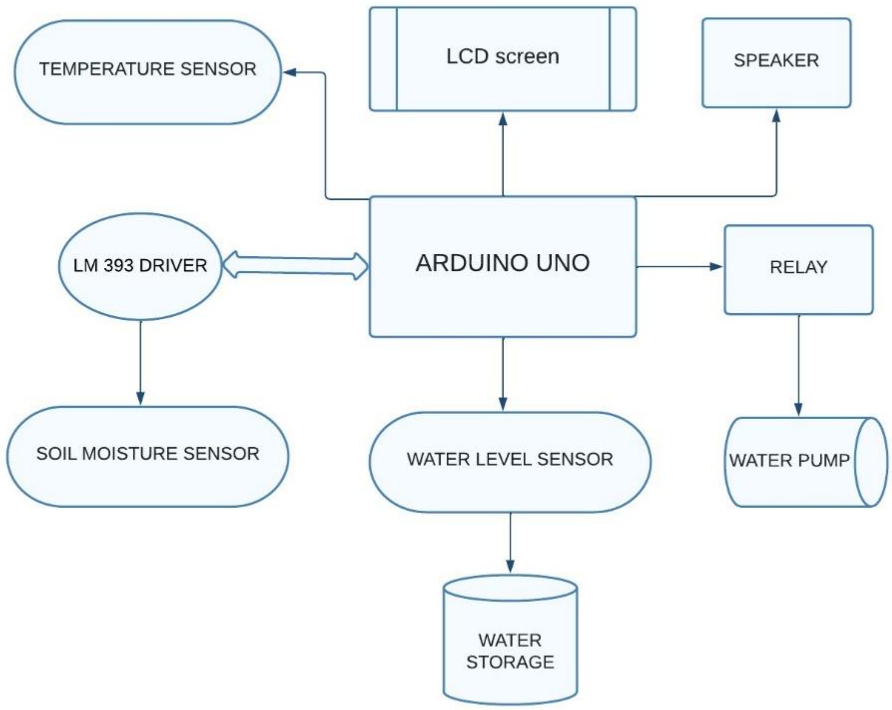

Figure 1: The block diagram of the Automatic Irrigation System

2. Generalities on the internet of things

2.1. Introduction

The IoT concept was coined by a member of the Radio Frequency Identification (RFID) development community in 1999, and it has recently become more relevant to the practical world largely because of the growth of mobile devices, embedded and ubiquitous communication, cloud computing, and data analytics.

These interconnected objects have data regularly collected, analyzed, and used to initiate action, providing a wealth of intelligence for planning, management, and decision-making. [4]

2.2. Definition of IoT

IoT is an internet of things where all things are interconnected. This sentence has two meanings. First, the core and the foundation of the IoT is still the internet. IoT is an extended network based on the internet. Second, the IoT connects anything at the user end for information exchange and communication.[5]

2.3. Iot architecture

There is four layers presenting the IoT architecture that can be divided as follows: Sensing Layer, Network Layer, Data processing Layer, and Application Layer.[6]

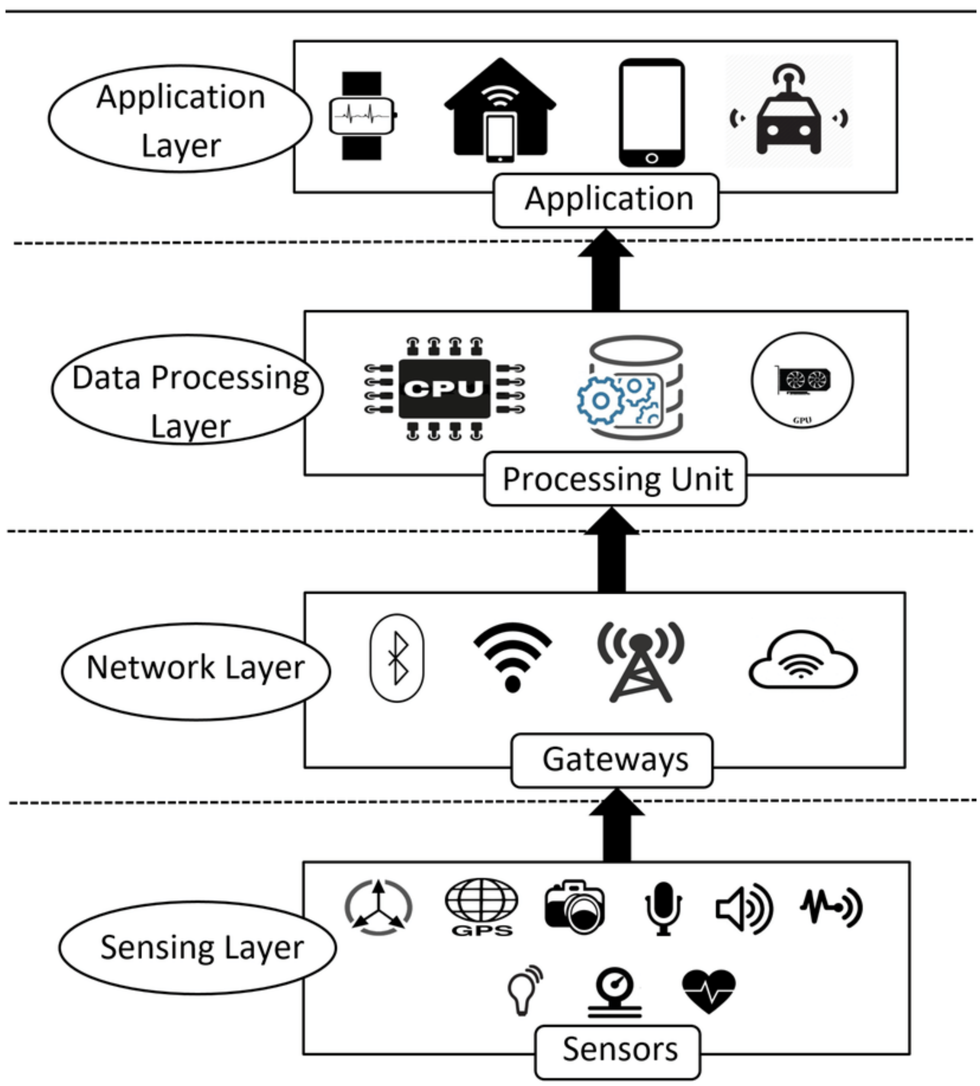

Figure 2: IoT Architecture Layers and Components.

2.3.1. Sensing Layer

Sensors, actuators, and devices are present in this Sensing layer. These Sensors or Actuators accept data (physical/environmental parameters), processes data, and emit data over the network.

2.3.2. Network layer

Internet/Network gateways and Data Acquisition Systems (DAS) are present in this layer. DAS performs data aggregation and conversion functions (Collecting data and aggregating data then converting analog data of sensors to digital data etc.). Advanced gateways which mainly open up a connection between Sensor networks and the Internet also perform many basic gateway functionalities like malware protection, filtering also sometimes decision-making based on inputted data and data management services, etc.

2.3.3. Data processing Layer

This is the processing unit of the IoT ecosystem. Here data is analyzed and pre-processed before sending it to the data center from where data is accessed by software applications often termed business applications where data is monitored and managed and further actions are also prepared. So here, Edge IT or edge analytics comes into the picture.

2.3.4. Application Layer

Data centers or the cloud is a management stage of data where data is managed and used by end-user applications like agriculture, health care, aerospace, farming, defense, etc.

2.4. Applications of IoT

The IoT finds application in various private and public aspects of life.

2.4.1. Healthcare

Wearable IoT devices provide a range of benefits to patients and healthcare providers alike. By extension, IoT enables healthcare professionals to monitor patients remotely. The devices can automatically collect patients’ health vitals like blood pressure, heart rate, temperature, and more.

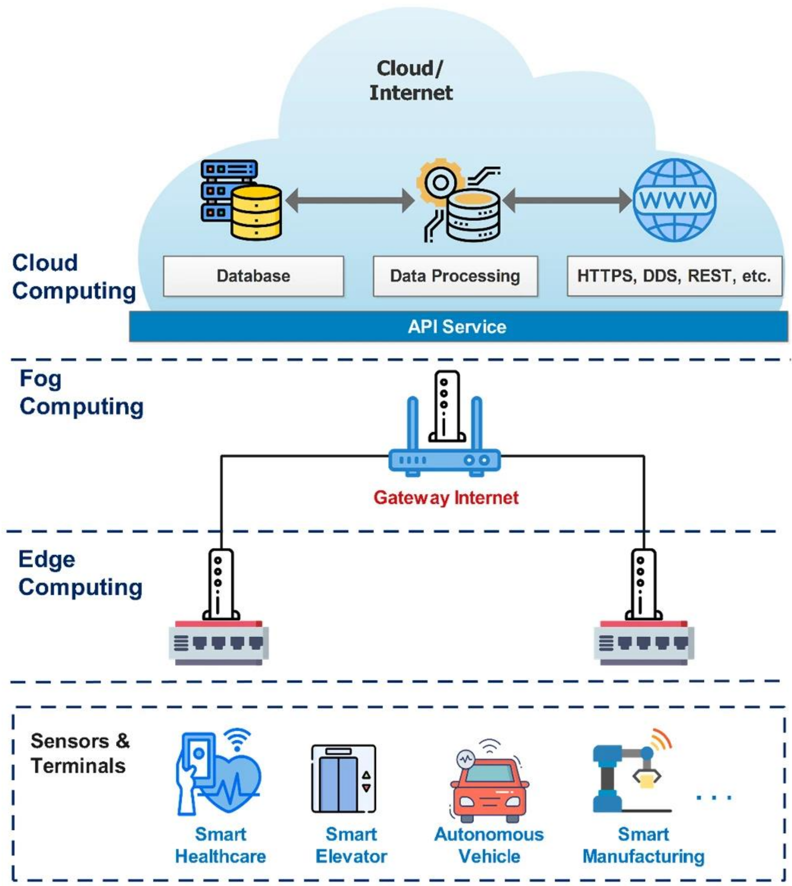

Figure 3: Smart healthcare

2.4.2. Transportation: IoT applications

Integrate personal and commercial vehicles by improving communication and information distribution. Besides connecting consumers and goods, it offers benefits such as route optimization, automobile tracking, weather monitoring, distance coverage, and more.

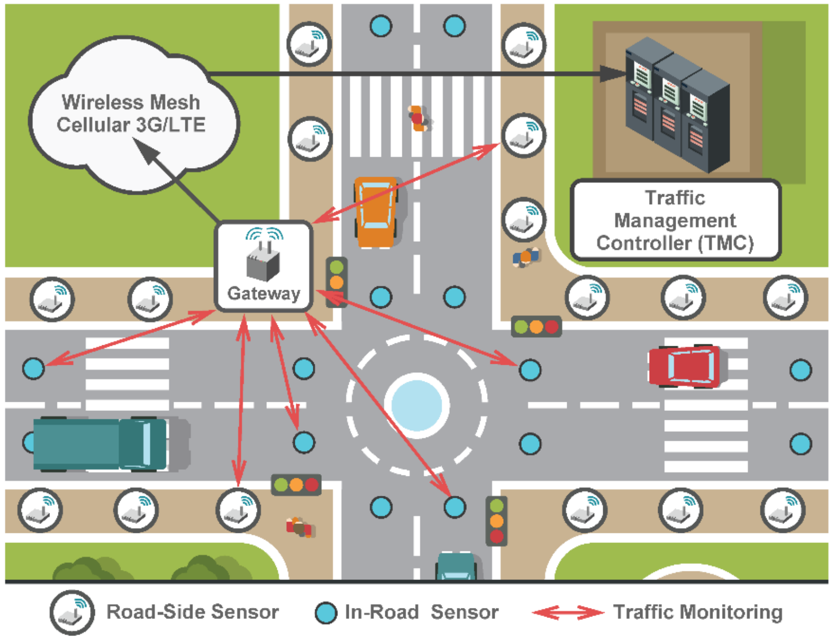

Figure 4: Smart Transportation

2.4.3. Smart Pollution Control

IoT devices and attached sensors are stationed at key city locations. They monitor pollution levels and periodically upload data to the IoT cloud. The system then processes the information to trigger public actions such as diversions or road closures.

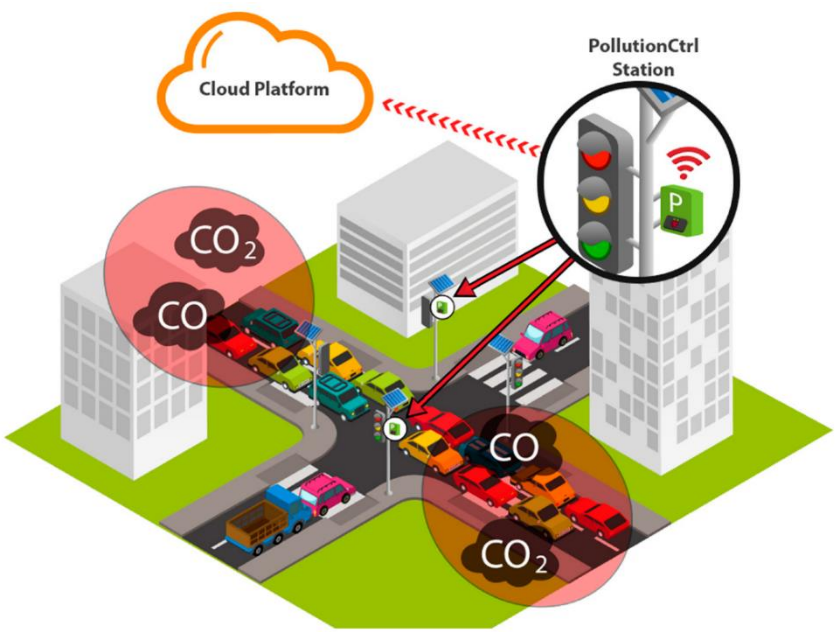

Figure 5: Smart pollution control

2.4.4. Agriculture

The ever-increasing world population drives up the demand for agricultural products. However, the migration of young people to big cities destabilizes the human resource required for agricultural development. IoT and related technologies can be pivotal in automating farming processes and fulfilling food demand.

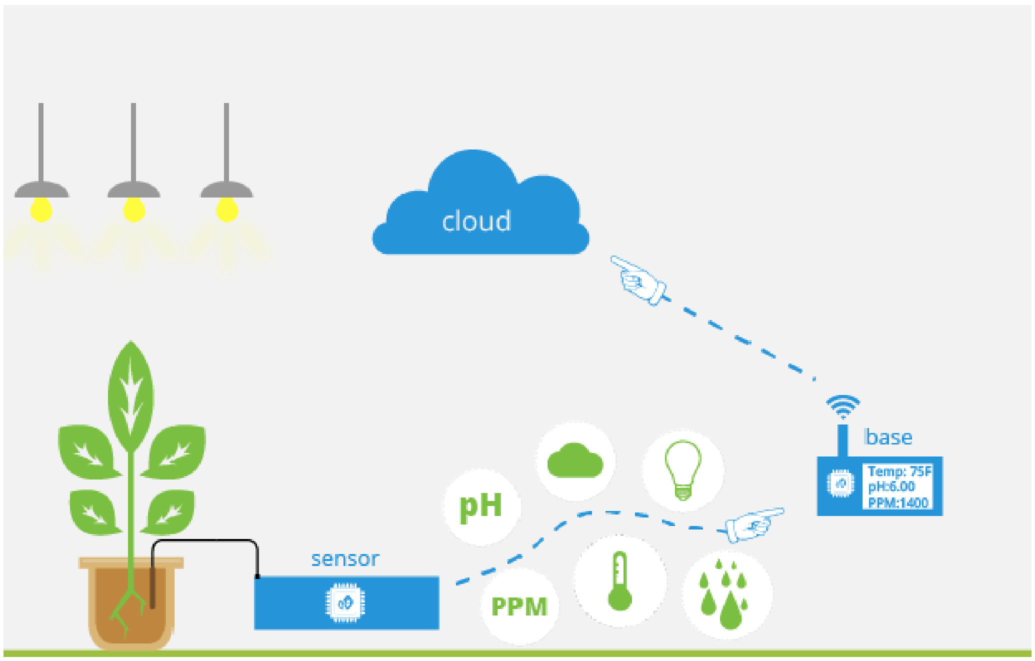

Figure 6: Internet of things in the agriculture

Conclusion

In this first chapter, we have provided a general overview of the problem and its existing solution, as well as our proposed one and the technology that will be used to address it. In the following chapter, we will provide a more detailed analysis of the project as a whole.

Chapter 2: Analysis and conception

Introduction

As any project carried out an analysis phase followed by a design phase is necessary. In this part, we will establish a complete analysis of the project while describing in a precise way the needs of the users as well as describe the design phase of the project to bring more details on the functionalities of the system as well as seeking to clarify the technical aspects.

1. Definition of UML

UML Unified Modeling Language unified object-modeling language and an object-oriented approach. It is the result of the fusion of three object-oriented methods Booch, OMT Object Modeling Technique and OOSE Object-Oriented Software Engineering designed respectively by Grady Booch ,James Rumbaugh, and Ivar Jacobson.[7]

2. Specification of requirements

In this part, we will identify the functional and non-functional requirements of our application to fully understand how our system works.

2.1. Functional requirements

The functional requirements express an action that the system must perform in response to a request. In other words, the functionalities that the solution offers to meet the needs of its customers.

In fact, the system must allow:

- ✓ Authentication: the user must log in as an administrator or as an ordinary user.
- ✓ User management: the admin can create delete and assign privileges to users.
- ✓ Sensor data acquisition: the application should automatically collect information from the sensors integrated into the irrigation system.

2.2. Non-functional requirements

Non-functional requirements are requirements that express a certain improvement of the services offered without any intervention in the application's business contributions.

Once the functional needs are satisfied the operational needs must be taken into account throughout the development of the application.

The non-functional needs of the implementation of this platform are the following:

- **Ergonomic constraint**: The application must have a simple and user-friendly interface so that the user can use it with the most comfort and efficiency.
- **Security**: Security measures to access the application, the user must authenticate, therefore a strong login and password are needed, which also allows the log out automatically when closing the application.
- **Performance**: This application must have acceptable response times to allow the realization of the different steps of the work in a user-friendly way.
- **Availability**: Our application must be available at all times.

3. Conception

3.1. Use case diagram

The description of the interaction is done according to the user's point of view and the use cases it allows to collect and describe the needs of the actors to the systems. It also allows facilitating the structuring of the users' needs and expresses the limits and objectives of the system.

- **Identification of the actors**

The actor is an external element that interacts with the system it can be a user or a third system (database computer program)

Table 2:Use case diagram description

Actor	Description
Admin	• Sign in • Manage users • Manage access rights • Consult data
User(farmer)	• Sign in • Add farm • Consult data

- **Global use case diagram**

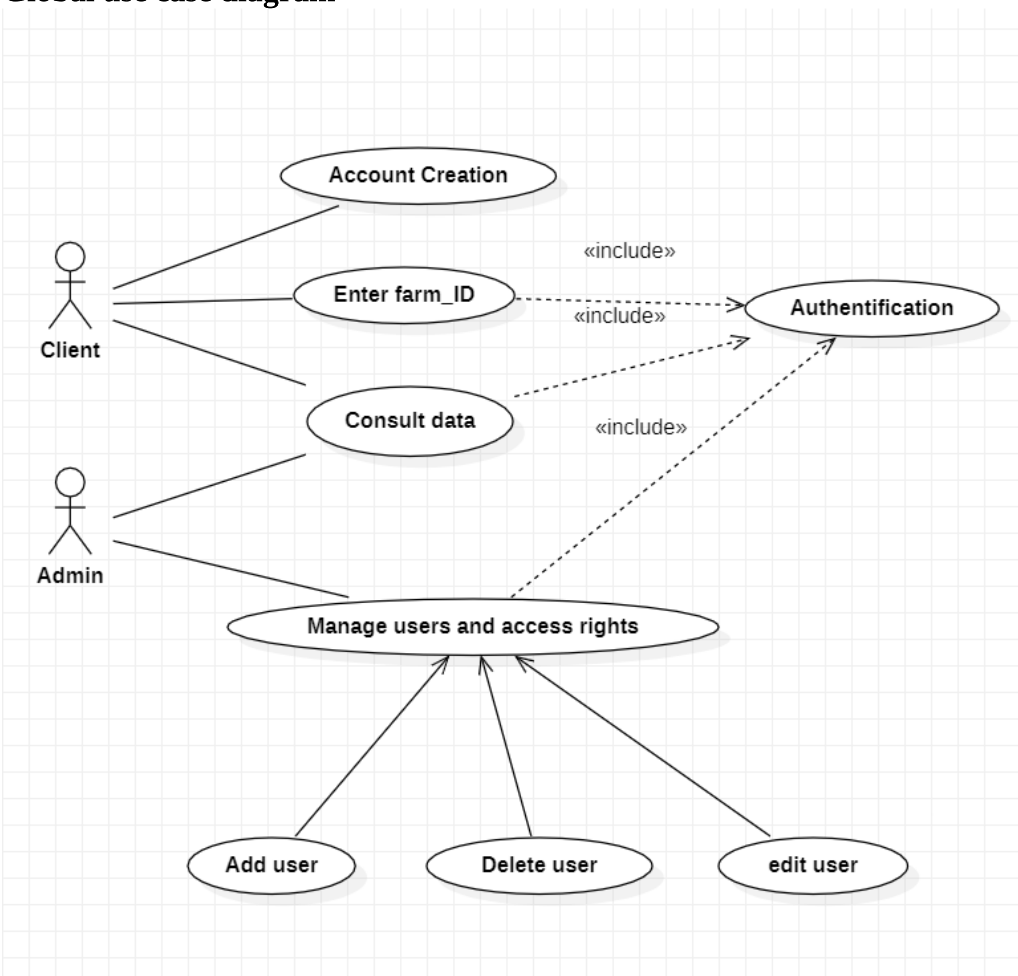

Figure 7: global use case diagram

In the figure7, we have given a global vision of the functional behavior of our system using the use case diagram. This diagram helps to describe the objectives of a system and the roles of the actors (users of the system)

Table 3: Global use case diagram description

Title	Use case diagram
Actors	Admin User
Preconditions	To be registered to the system
Post conditions	Internet connection required Authentication required
Description	• The user must authenticate • Admin must authenticate • the admin manages the user accounts and manages access rights • The user and the admin can consult the data

- **Authentification use case diagram**

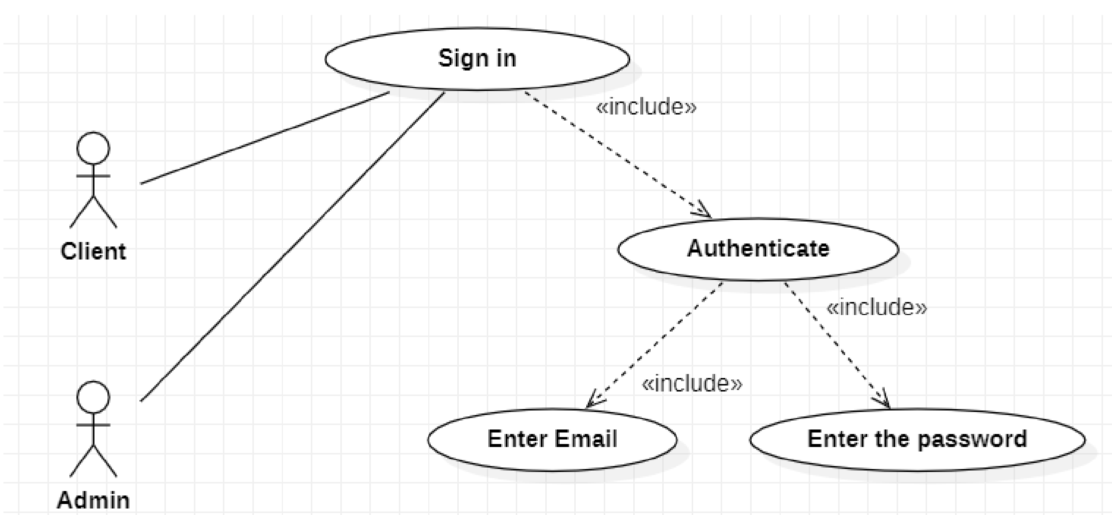

Figure 8: Authentication use case diagram

Table 4:Authentication use case diagram

Use cases	Authentication
Actors	User
Precondition	Be registered to the system
Post condition	Authenticated user
Scenario description	• The user of our application authenticates himself by entering his email and password. • The user validates the data entered by clicking on connect. • The system explores the database to check if the data entered by the user is valid.
Exception	If the email entered or the password are wrong an error message is displayed

3.2. Class diagram

The class diagram is a diagram used in software engineering to present the classes and interfaces of the system as well as the different relationships between them.

This diagram is part of the static part of UML because it abstracts the temporal and dynamic aspects. A class describes the responsibilities the behavior is the type of a set of objects. The elements of this set are the instances of the class.

The main elements of this static view are the classes and their generalization association relationship and several types of dependencies such as realization and use.

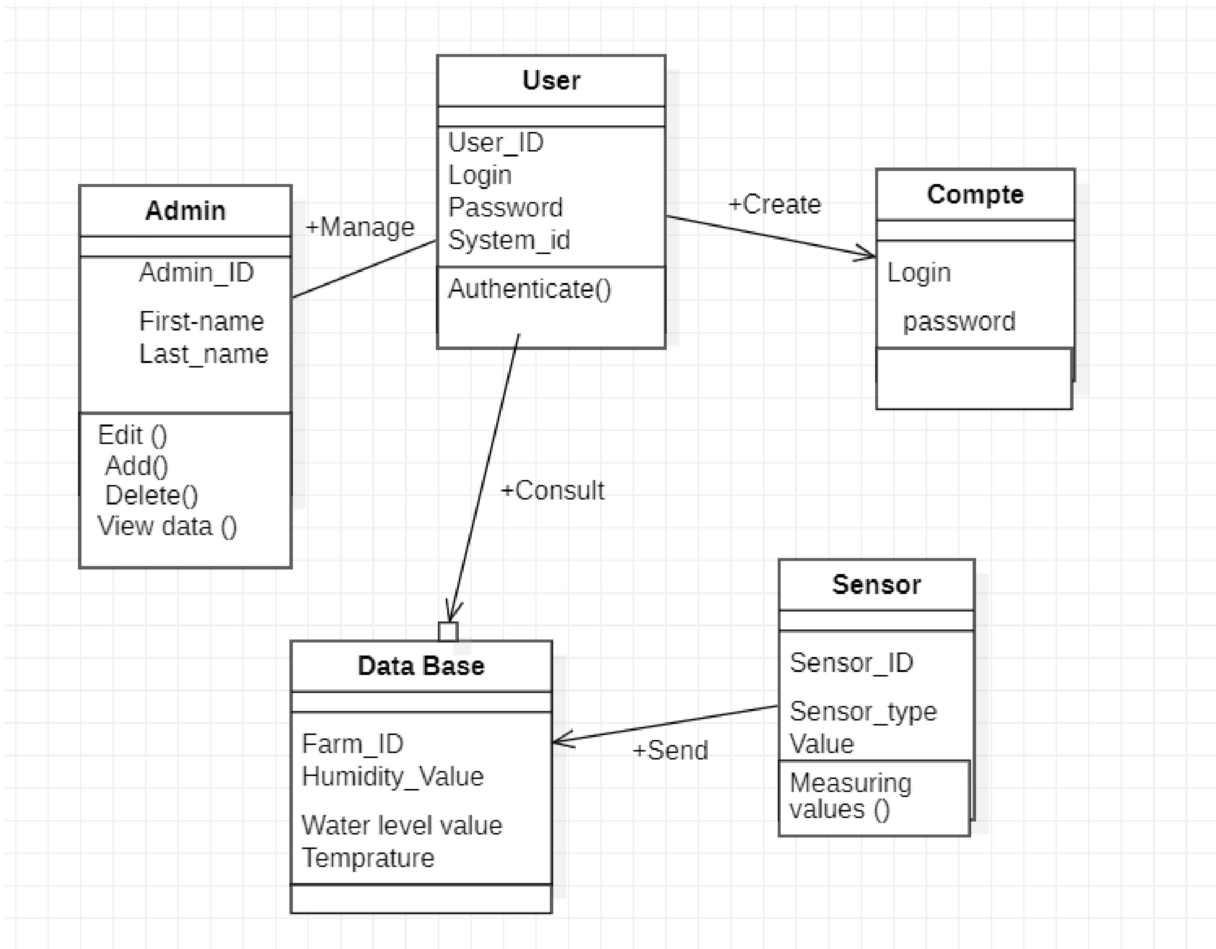

Figure 9: class diagram

The figure above represents the class diagram of our project; it describes the different classes of our system by treating the attributes and methods of each of them.

3.3. Sequence diagram

Sequence diagrams are used to establish a link between class diagrams and use case diagrams because it shows how objects communicate with each other to achieve the expected functionality. It shows the interactions between objects from a temporal point of view.

- Authentication sequence diagram

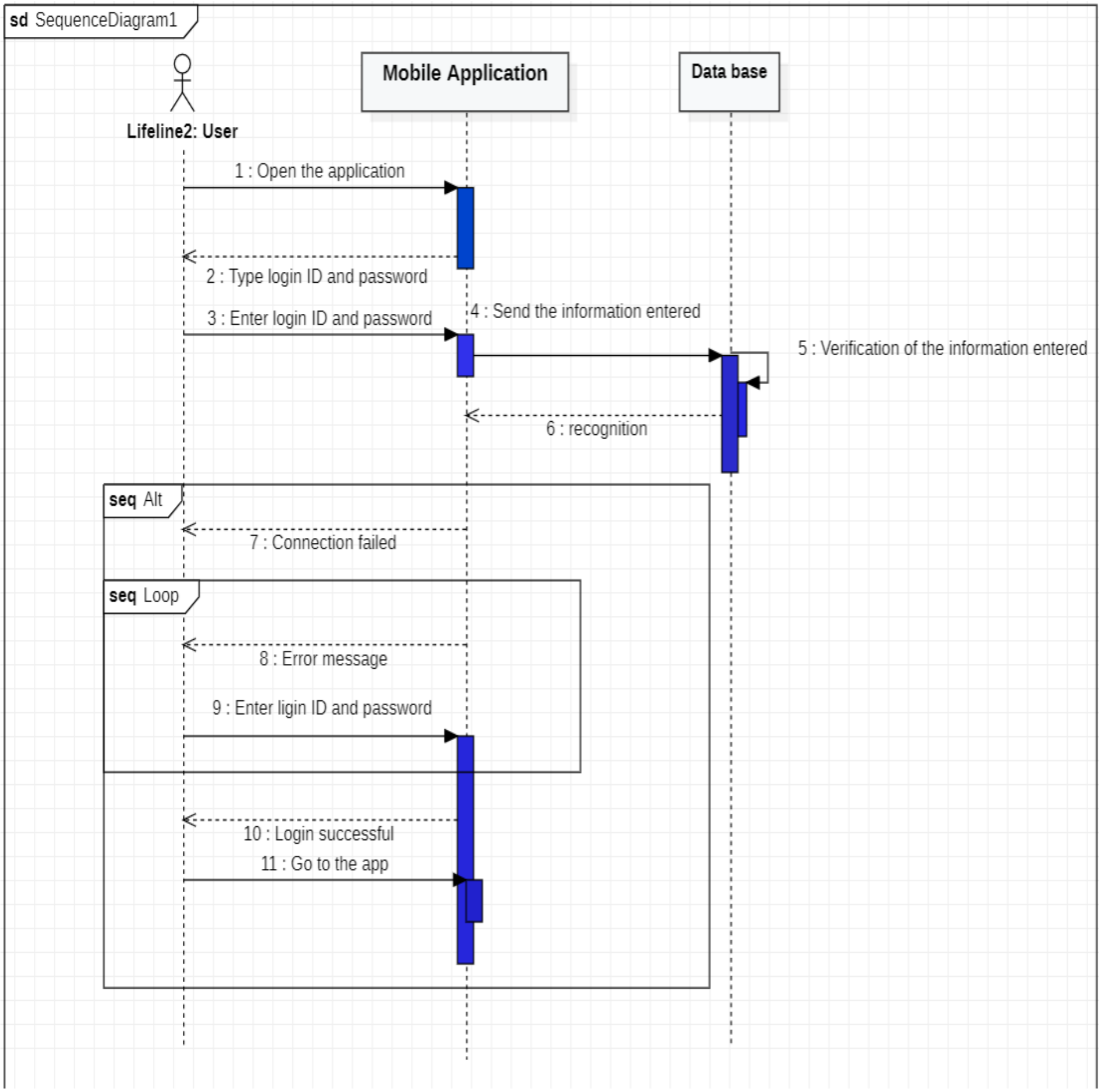

Figure 10:Authentication sequence diagram

The figure above represents the authentication sequence diagram; it illustrates the interactions between the different system actors to authenticate.

- Inscription sequence diagram

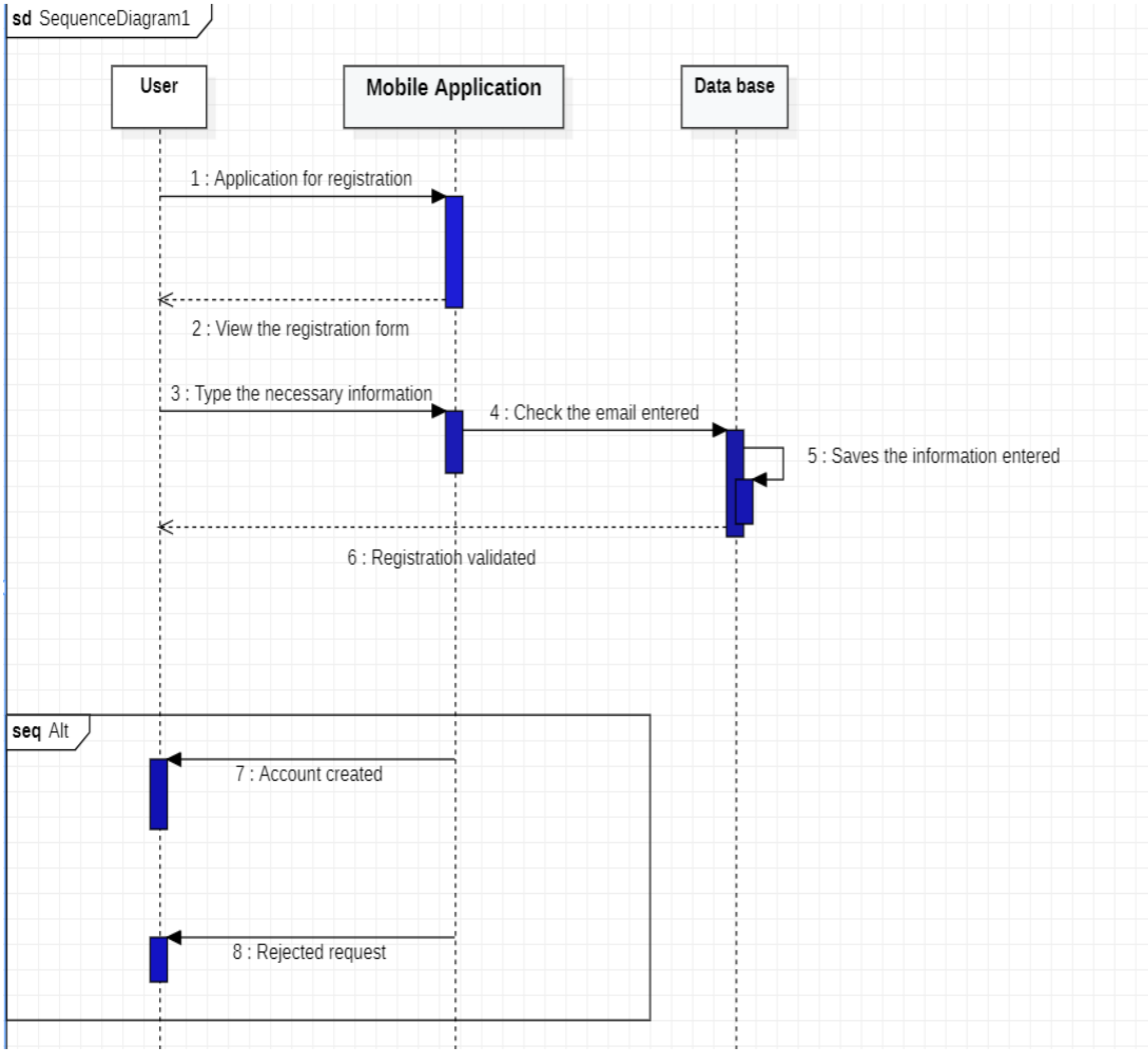

Figure 11: inscription sequence diagram

The figure represents the diagram of sequence of inscription explaining its process.

- System data consultation sequence diagram

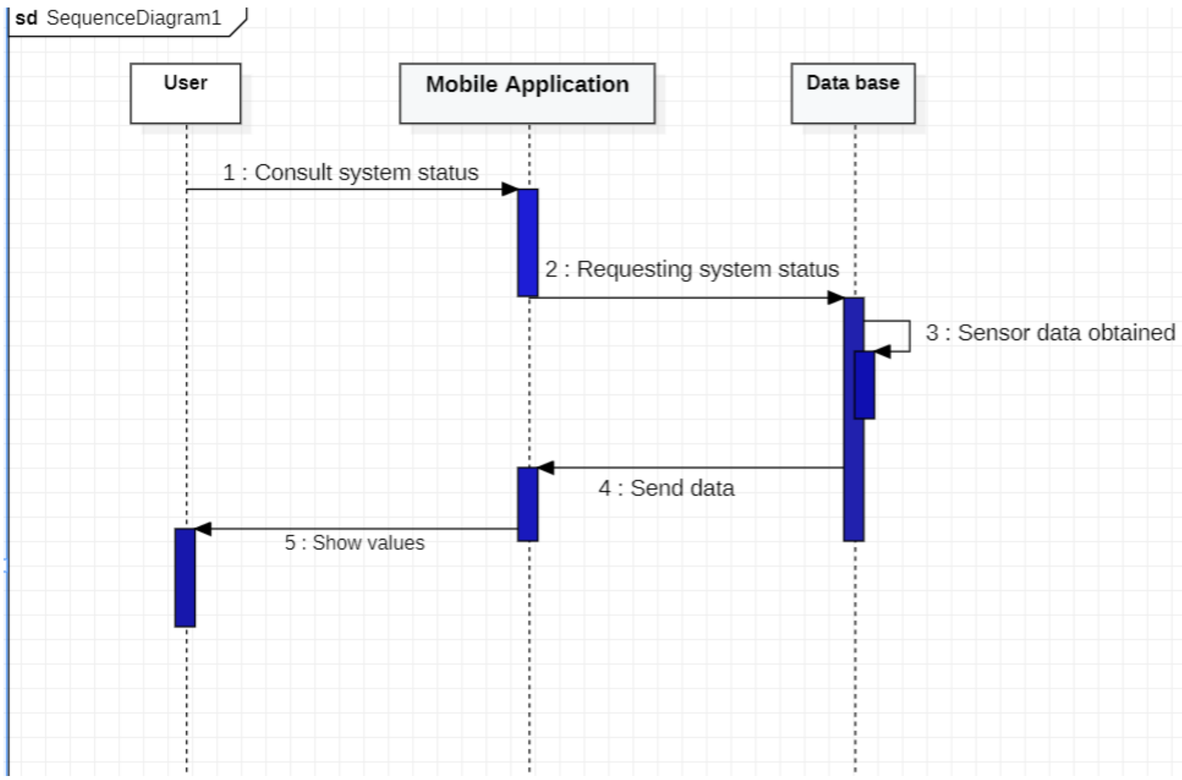

Figure 12: System data consultation sequence diagram

The figure above represents the consultation of system data.

Conclusion

In this chapter, we gave a rigorous vision of the system to be realized by determining the elements and their interactions. The result was presented in various diagrams.

In the following chapter, we will list the different steps of the development of our solution as well as the tests and the results found.

Chapter 3: System implementation

Introduction

In this chapter, we will present the implementation details of our proposed system, which is represented by the smart system deployed at the farm level according to our proposed architecture design, also the mobile application developed. Finally, we illustrate and discuss the obtained results.

1. The working environment

1.1.The hardware environment

The complete hardware of this work is shown in Figure 2, which contains:

- Arduino UNO

Arduino Uno is a microcontroller board based on the ATmega328P. It has 14 digital input/output pins, 6 analog inputs, a 16 MHz ceramic resonator , a USB connection, a power jack, an ICSP header and a reset button. It contains everything needed to support the microcontroller; simply connect it to a computer with a USB cable or power it with a AC-to-DC adapter or battery to get started. [8]

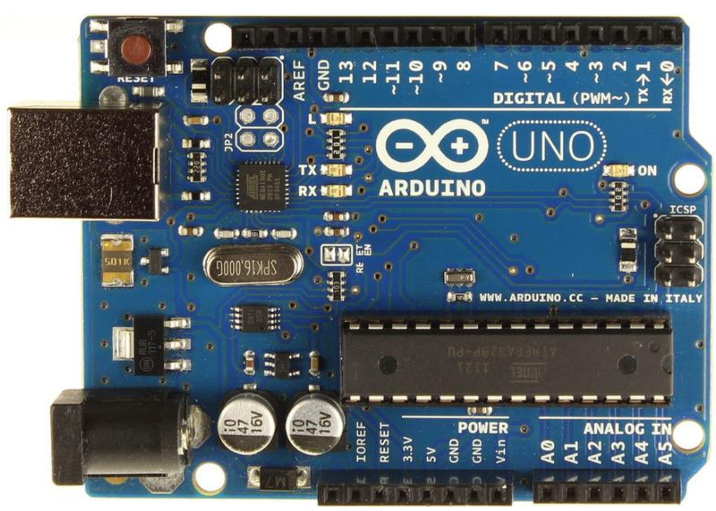

Figure 13:Arduino uno board

- Soil Moisture Sensor (with LM393 Driver)

The soil moisture sensor is used to measure the volumetric water content of soil. It is used to monitor soil moisture content to control irrigation on farms. A moisture sensor is used to sense the level of moisture content present in irrigation fields. It has a level detection module in which we can set a reference value. [9]

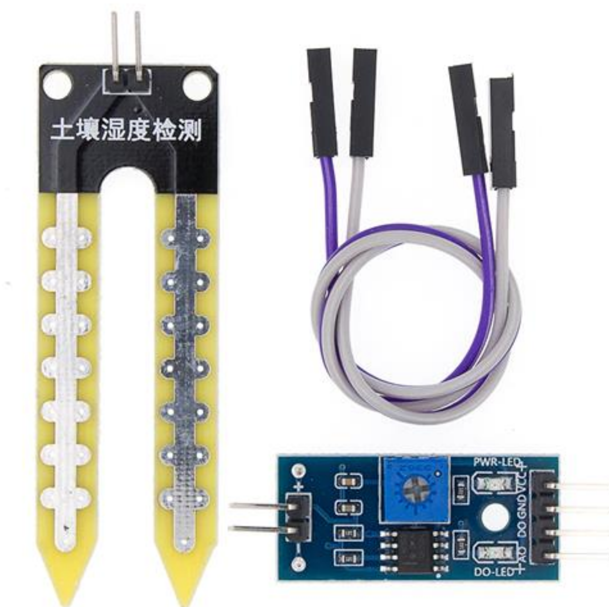

Figure 14:Soil Moisture Sensor

- DHT11 Temperature Sensor

The temperature range of DHT11 is from 0 to 50 degree Celsius with a 2-degree accuracy. It gives one reading for every second. It is necessary to reduce the watering frequency. That is when the weather gets cooler, less water is needed whereas vice versa in the other case.[10]

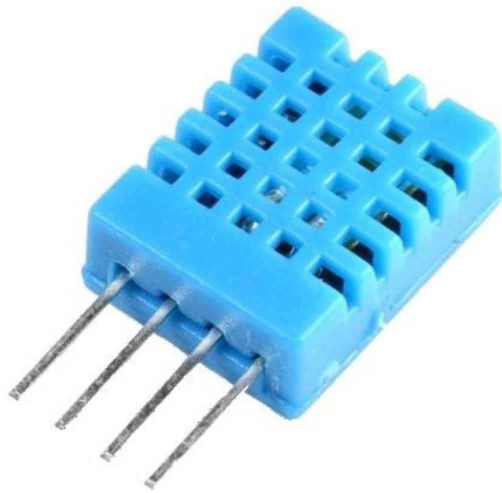

Figure 15:Temperature Sensor

- Water level sensor

The sensor has ten copper traces, five of which are power traces and five are sense traces. These traces are not connected but are bridged by water when submerged and act as a variable resistor whose resistance varies according to the water level.

This resistance is inversely proportional to the height of the water, that is, the more water the sensor is immersed in, results in better conductivity and will result in a lower resistance and vice versa. The sensor produces an output voltage according to the resistance, and this voltage is interpreted by a microcontroller to determine the water level.[11]

Figure 16:Water level sensor

- 16x2 LCD Display

This is a 16x2 LCD display screen with I2C interface. It is able to display 16x2 characters on 2 lines, white characters on blue background. This I2C 16x2 Arduino LCD Screen is using an I2C communication interface.

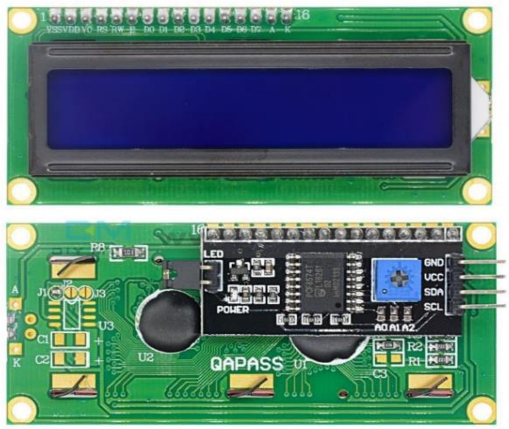

Figure 17 : 16x2 LCD Display

- Water Pump 5V 120L/H

This is Micro Submersible Water Pump DC 3V-5V, can be easily integrate to your water system project. The water pump works using water suction method, which drain the water through its inlet and released it through the outlet.[12]

Figure 18:Water Pump

- 5V Relay

The four-channel relay module contains four 5V relays and the associated switching and isolating components, which makes interfacing with a microcontroller or sensor easy with minimum components and connections. [13]

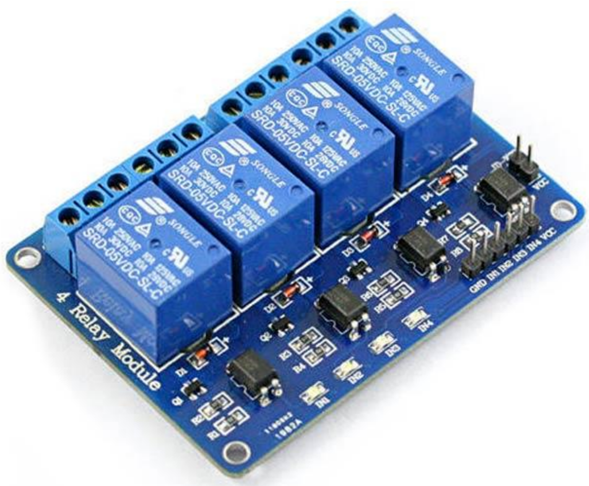

Figure 19 : 5V Relay

- Resistors

A resistor is a passive electrical component that resists the flow of electrical current, reducing the voltage and limiting the amount of current in a circuit.[14]

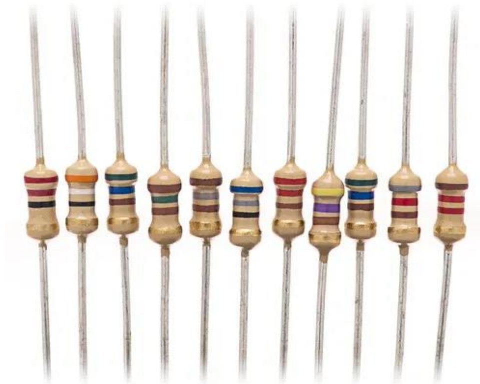

Figure 20:Resistors

- 7 colors LED module

The LED contains logic built-in to automatically flash the LED at different rates while cycling through the seven possible colors of red, green, blue, yellow, purple, cyan and white by mixing the different LED colors.[15]

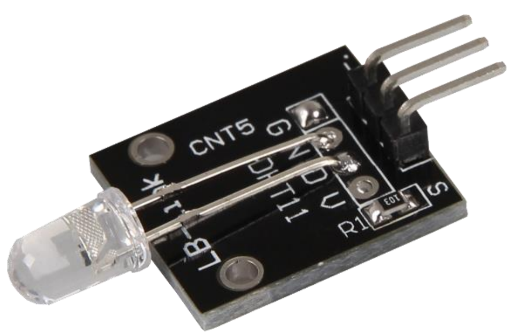

Figure 21: LED module

1.2. The software environment

1.2.1. The use of Arduino IDE 1.8.2

The open-source Arduino software (IDE) makes it easy to write code and upload it to the board. It runs on Windows, macOS, and Linux. The environment is written in Java and based on Processing and other open-source software. This software can be used with any Arduino board.

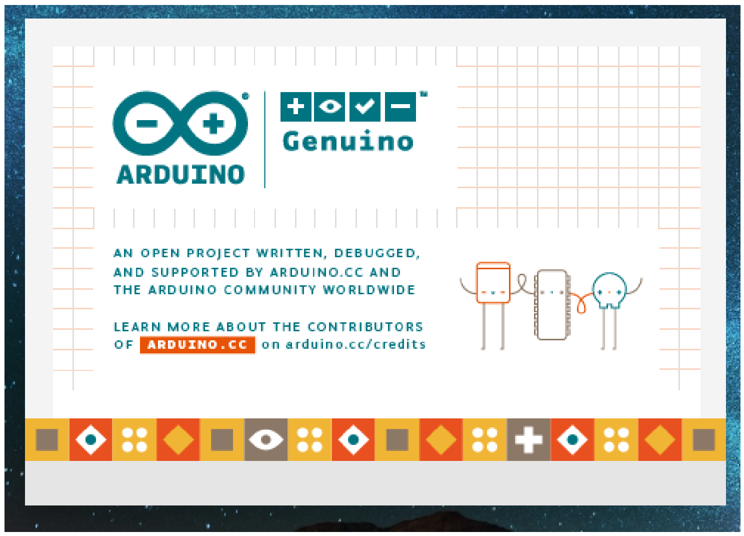

Figure 22:Arduino IDE Interface

1.2.2. Firebase

Firebase is a set of hosting services for any type of application (android, iOs, Javascript, Node.js, Java, Unity, PHP, C++...). It offers to host in NoSQL and in real-time databases, content, social self-notification (Google, Facebook, Twitter, and GitHub), and notification, or services, such as a real-time communication server. [16]

Figure 23: Firebase logo

1.2.3. MIT App Inventor

MIT App Inventor is an intuitive, visual programming environment that allows everyone to build fully functional apps for Android phones, iPhones, and Android/iOS tablets. Its blocks-based tool facilitates the creation of complex, high-impact apps in significantly less time than traditional programming environments. The MIT App Inventor project seeks to democratize software development by empowering all people, especially young people, to move from technology consumption to technology creation.[17]

Figure 24: MIT App Inventor logo

2. Realization

2.1. Embedded system development and tests result

2.1.1. Water level sensor implementation

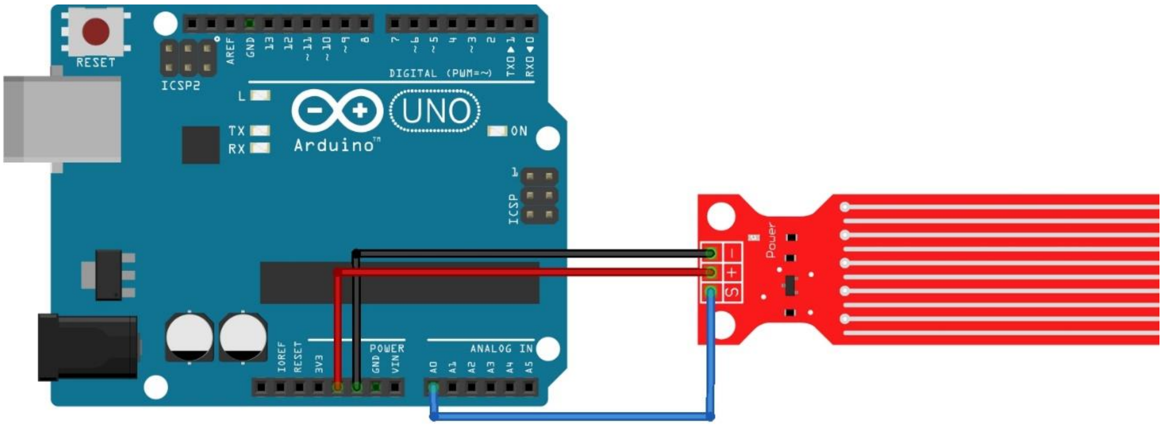

Figure 25:Water level sensor connection

Once put into a certain depth in the water, the pressure on the sensor's front surface is converted into the liquid level height and then will indicate if the liquid level in a fixed container is too high or too low according to the following thresholds:

- when the sensor is dry then the container is empty
- when the sensor is partially immersed in water then water level ~420
- when the sensor is fully immersed in water then water level >510

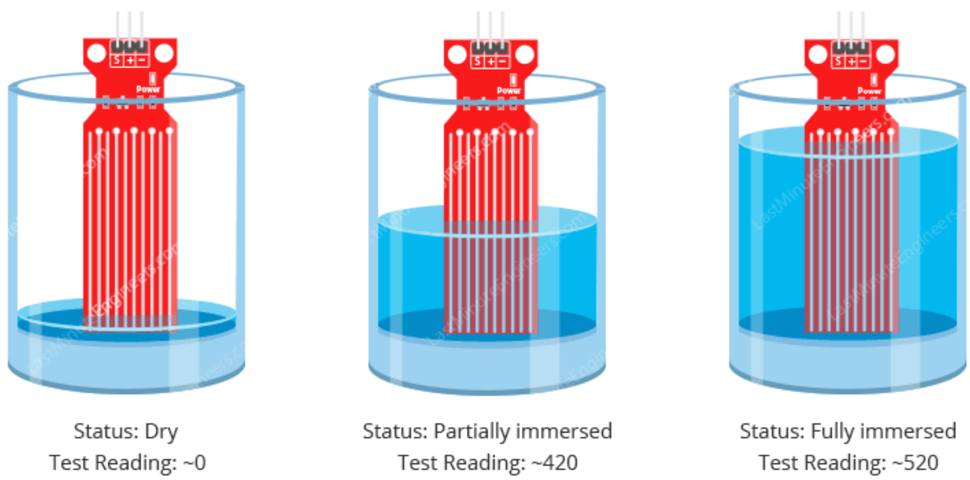

Figure 26:Water sensor's levels

To display these results we can use the LCD display by connecting it to the Arduino board.

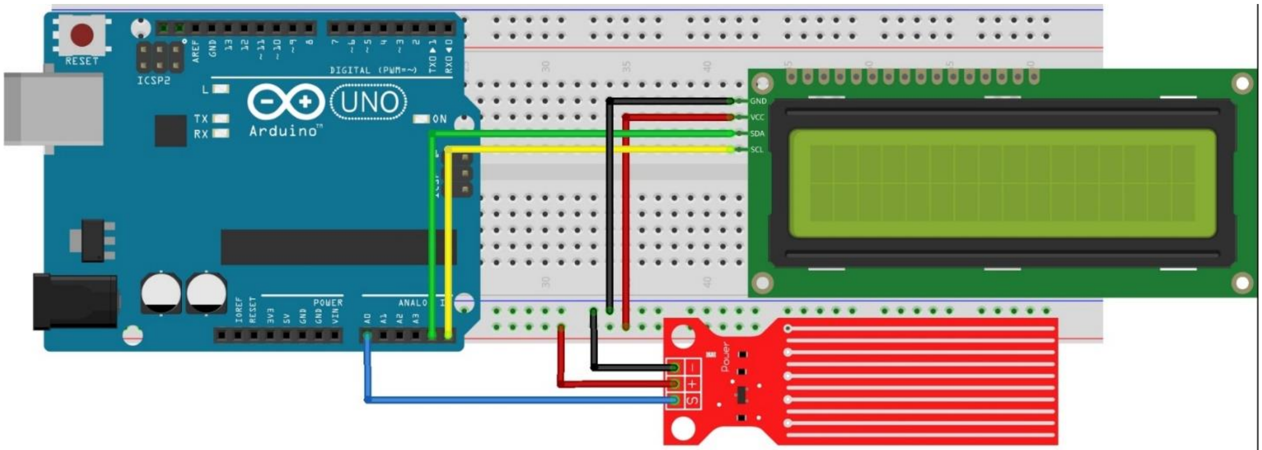

Figure 27: LCD display connection

After constructing the circuit, the code should be uploaded giving the following results:

- Low level:

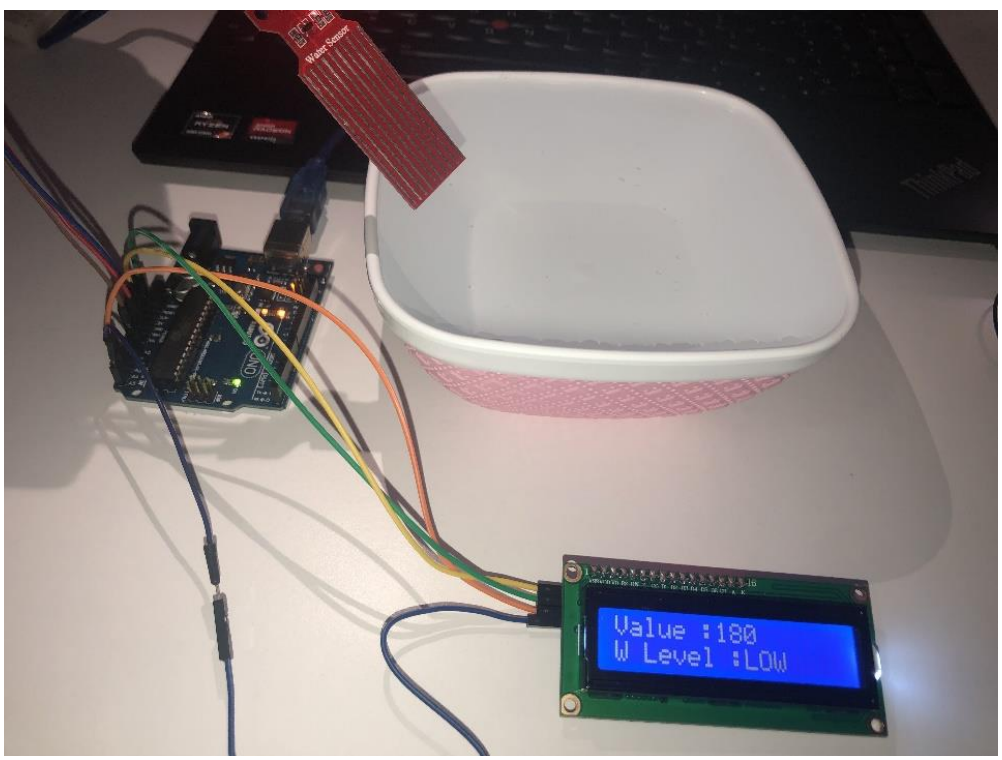

Figure 28: Water level sensor test(1)

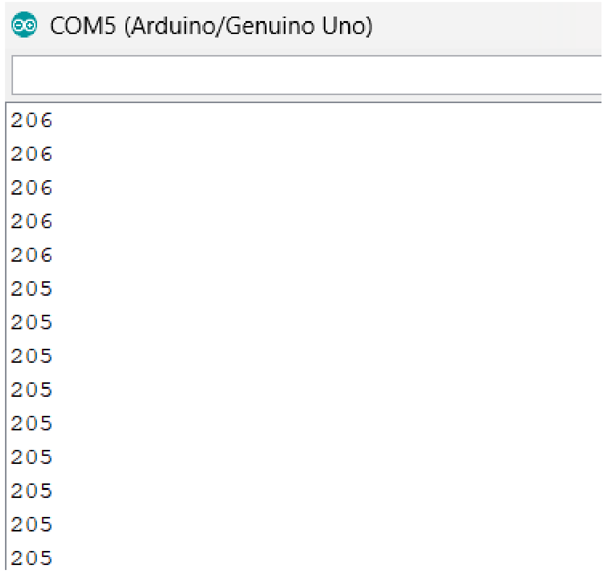

Figure 29: Output viewed by the Serial Monitor window(1)

- Medium level:

Figure 30: Water level sensor test(2)

358
358
358
358
358
372
379
410
417
458
495
497
494
495
499
499
496
493
491
491

Figure 31: Output viewed by the Serial Monitor window(2)

- High level :

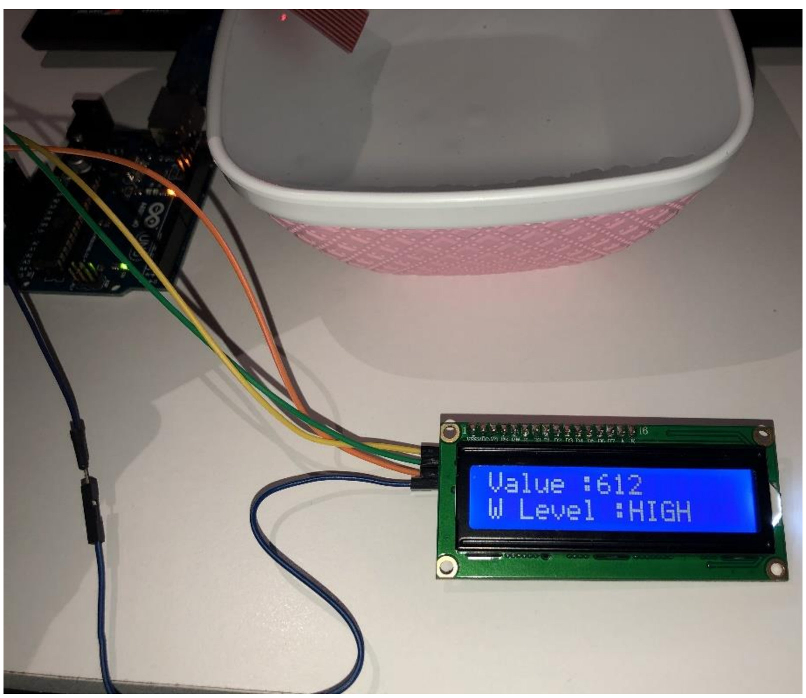

Figure 32: Water level sensor test(3)

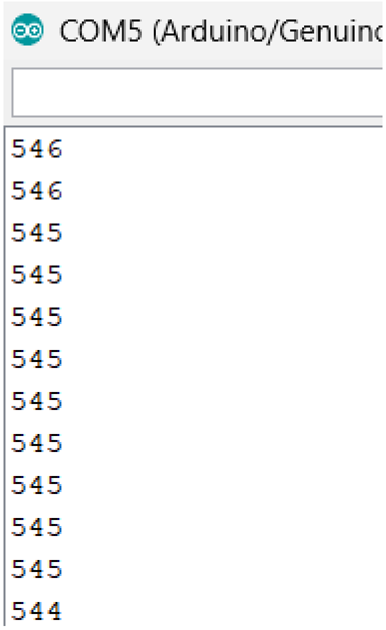

Figure 33:Output viewed by the Serial Monitor window(3)

2.1.2. Soil moisture sensor implementation

The sensor is connected to the Arduino card by the LM 393 driver it will determine how much moisture is present in the soil around it.

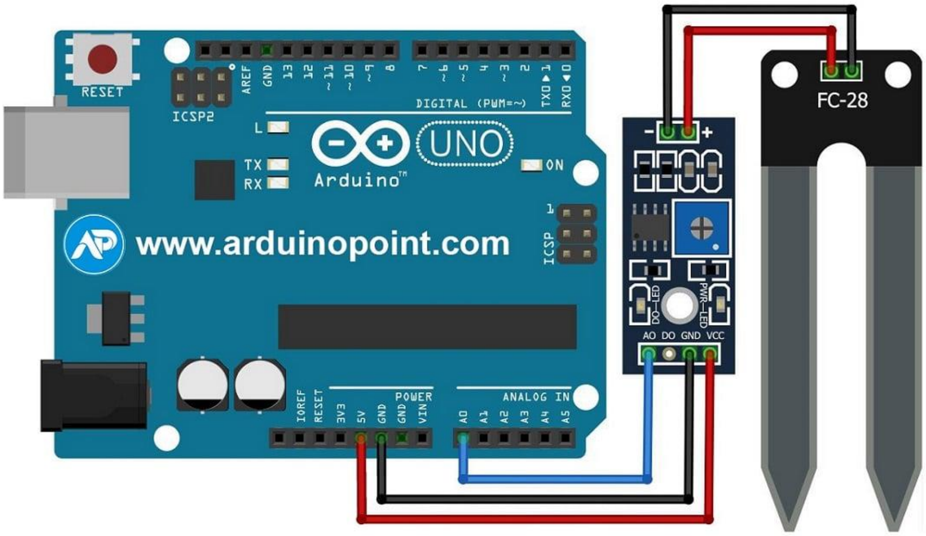

Figure 34:Soil moisture sensor connection

To display these results we can use the LCD display by connecting it to the Arduino board.

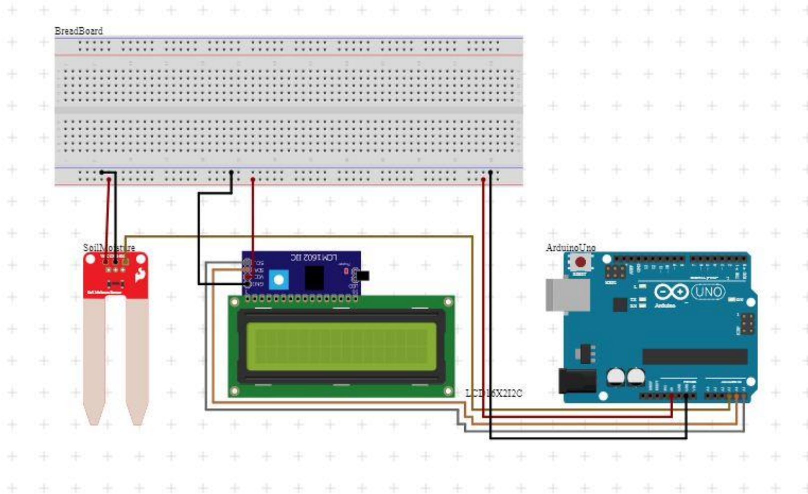

Figure 35: LCD display connection

This sensor conducts current through the soil using two probes, then reads the resistance to determine the moisture level. More water allows the soil to conduct electricity more easily (with less resistance), whereas dry soil does not (more resistance).

After constructing the circuit, the code should be uploaded giving the following results:

- If the moisture level < 300 then soil is Dry , the moisture level is Low and the pump is ON

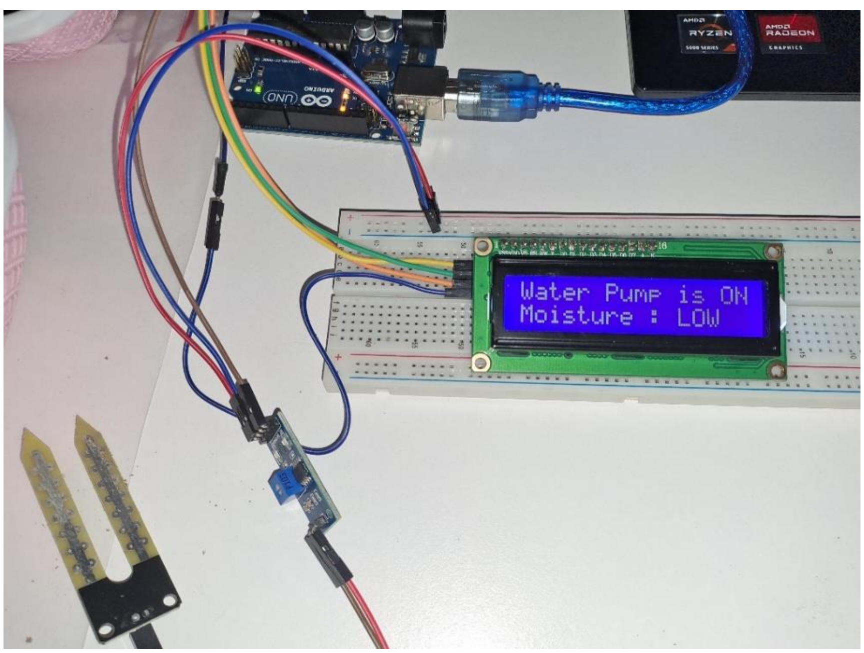

Figure 36:Soil moisture sensor test 1

- If the moisture level > 300 and < 950 then the moisture level is Medium and the pump is OFF

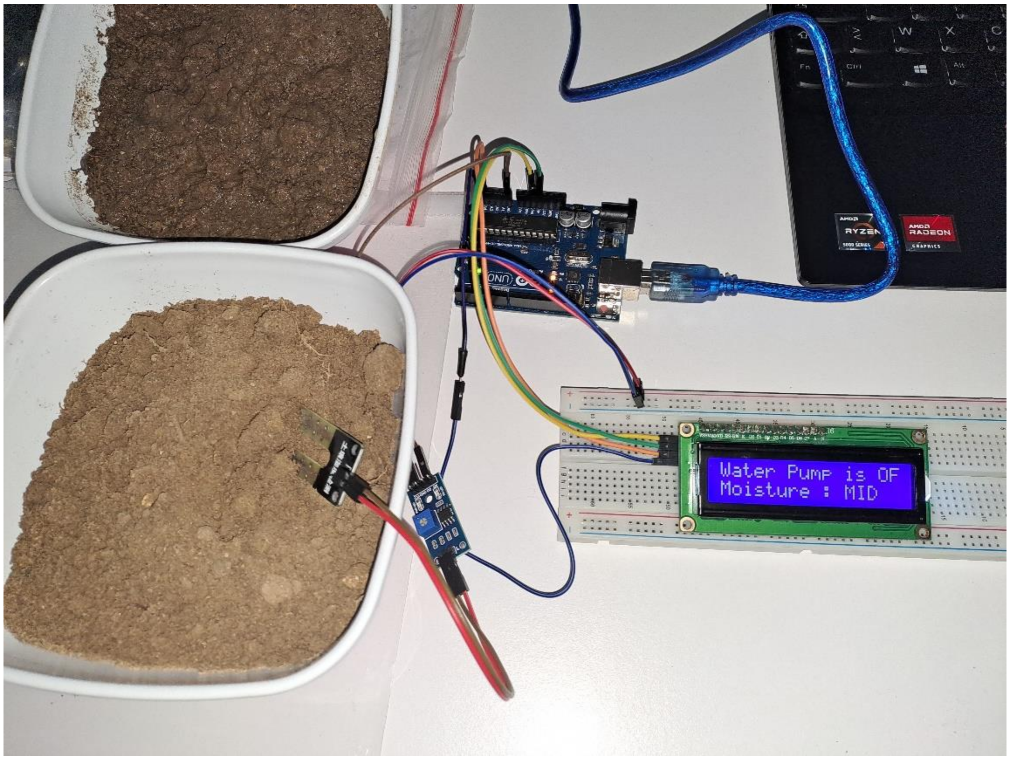

Figure 37:Soil moisture sensor test 2

- If the moisture level > 950 then the moisture level is High and the pump is OFF

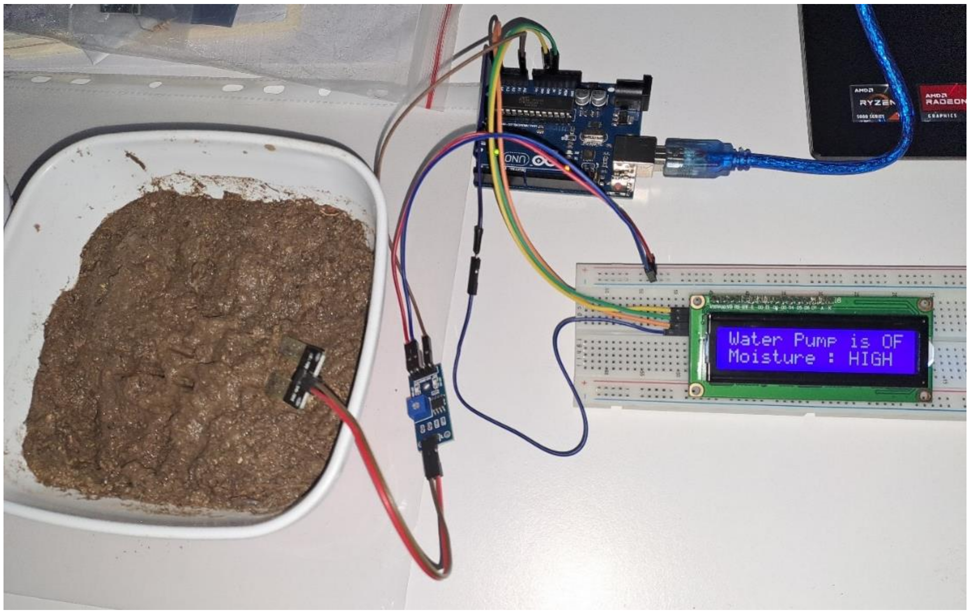

Figure 38:Soil moisture sensor test 3

2.1.3. The water pump and Relay Module implementation

A power relay module is an electrical switch that is operated by an electromagnet. The electromagnet is activated by a separate low-power signal from a micro controller. When activated, the electromagnet pulls to either open or close an electrical circuit which is the water pump.

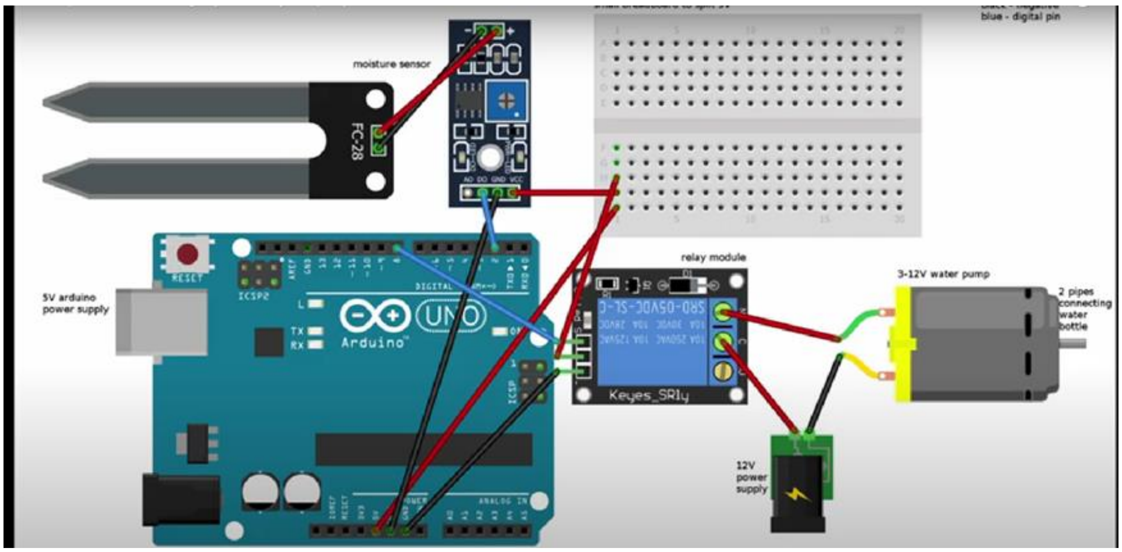

Figure 39: Relay module and water pump connection(1)

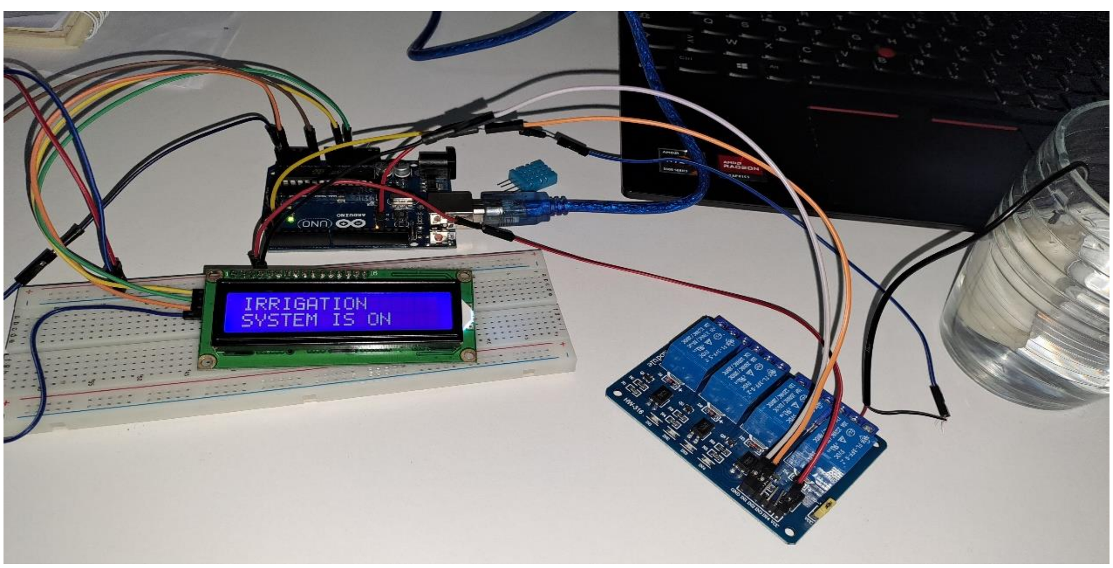

Figure 40: Relay module and water pump connection (2)

2.1.4. Global connection diagram

After the realization of the unit tests on each sensor we moved on to the development of the global code as well as the wiring the following diagram shows the wiring of the artificial irrigation system.

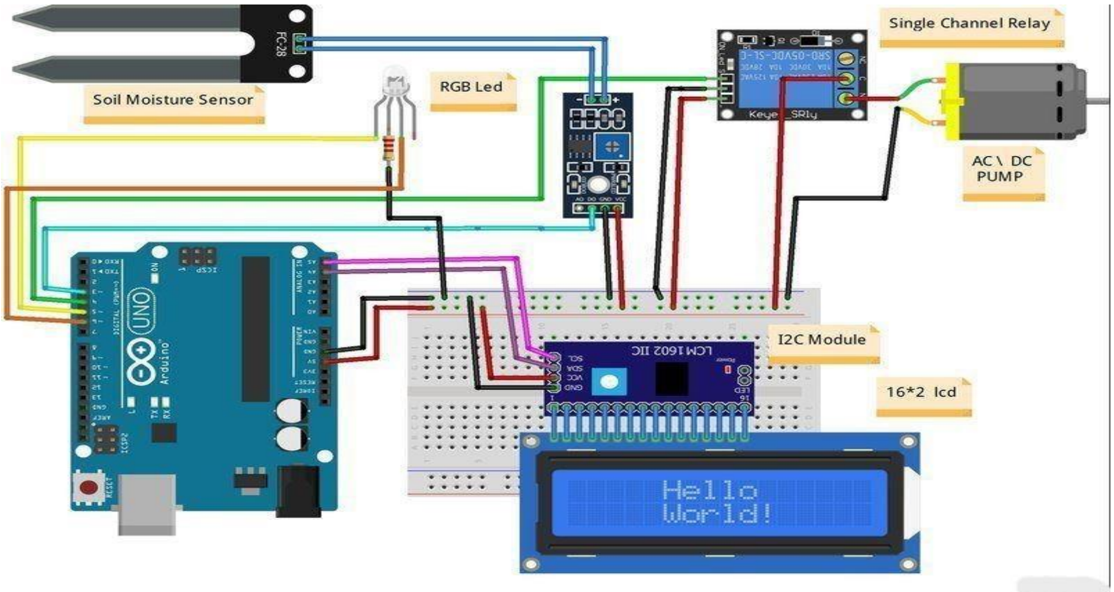

Figure 41: Global connection diagram(1)

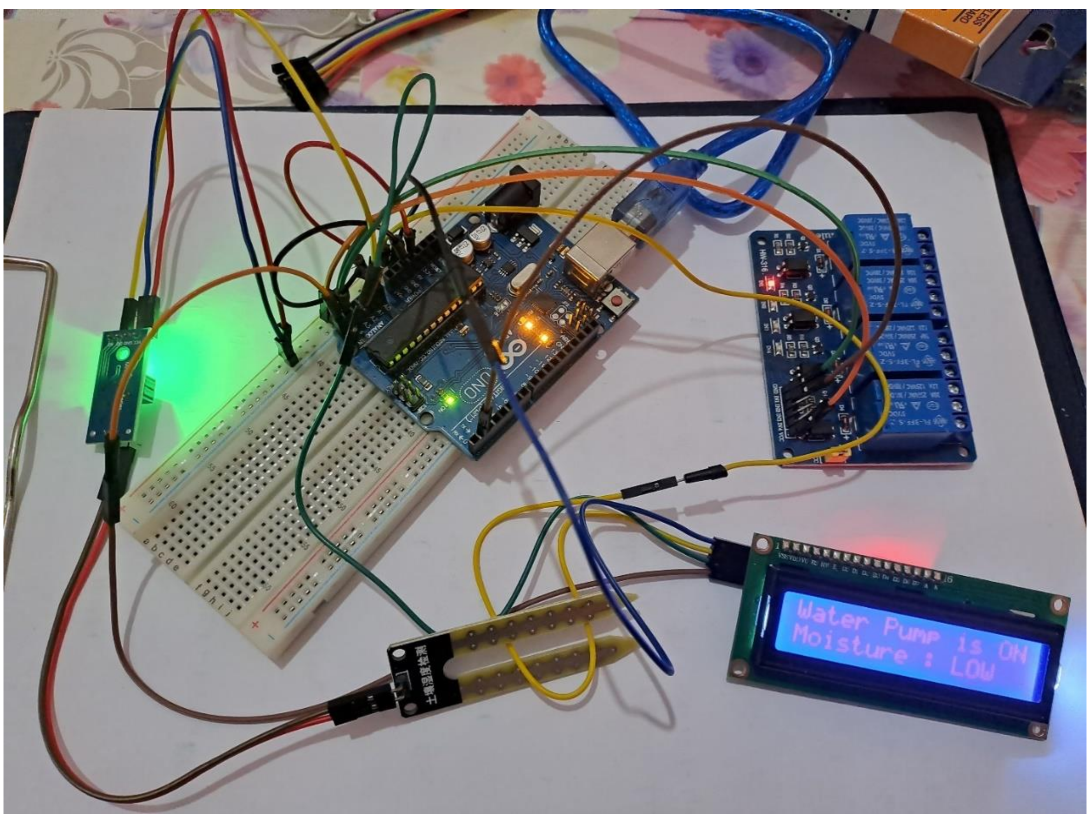

Figure 42:Global connection diagram(2)

This flow chart diagram explains the process of irrigation operation.

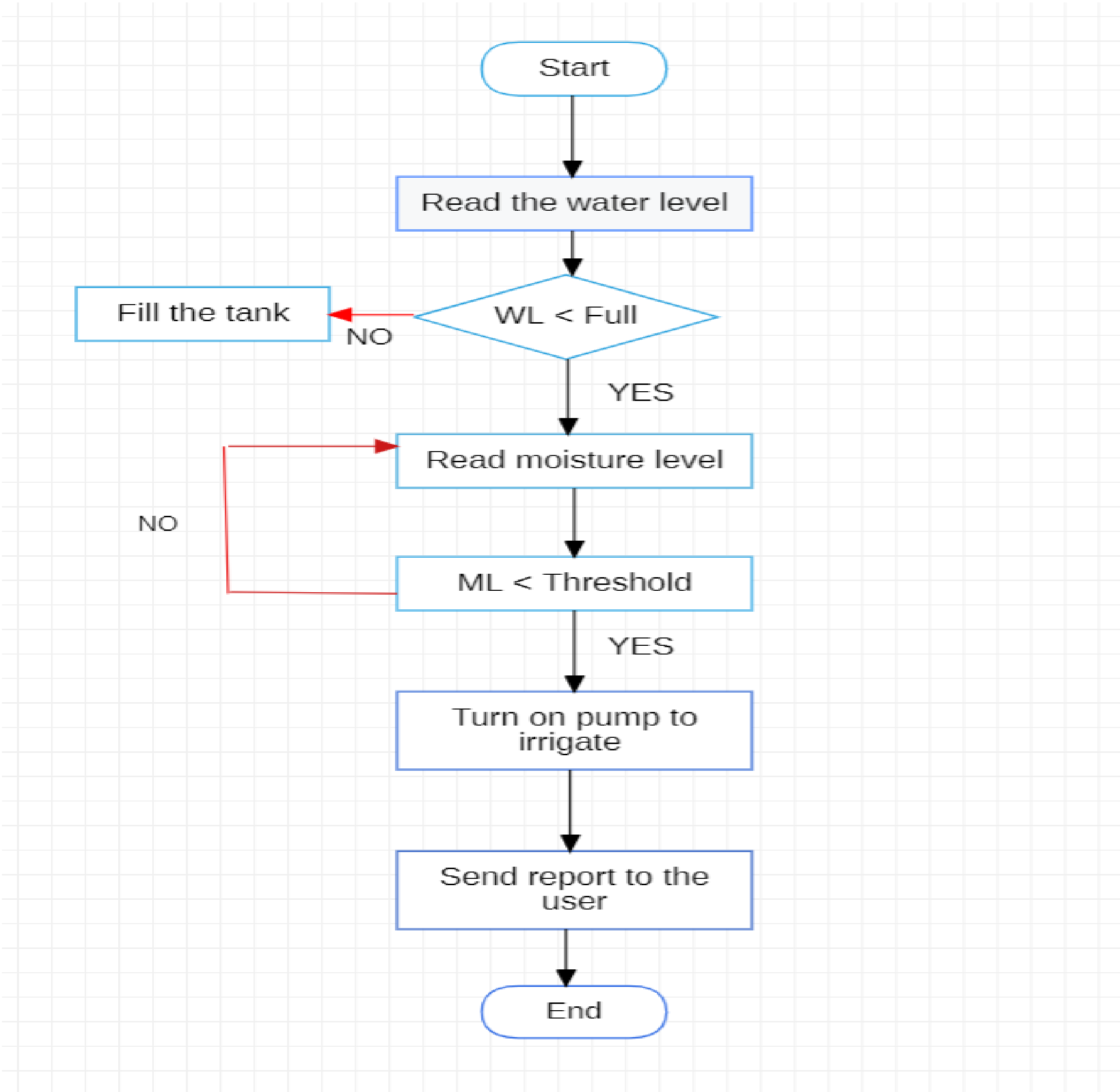

Figure 43:Fow chart diagram

2.2. Mobile application

The mobile application was created using MIT App Inventor. This application was designed to have multiple pages in order to show the data readout from different sensors. Aside from the monitoring pages, a page dedicated to identifying as an admin or as a farmer (user), and another interface for authentication.

Figure 44:Login Interface

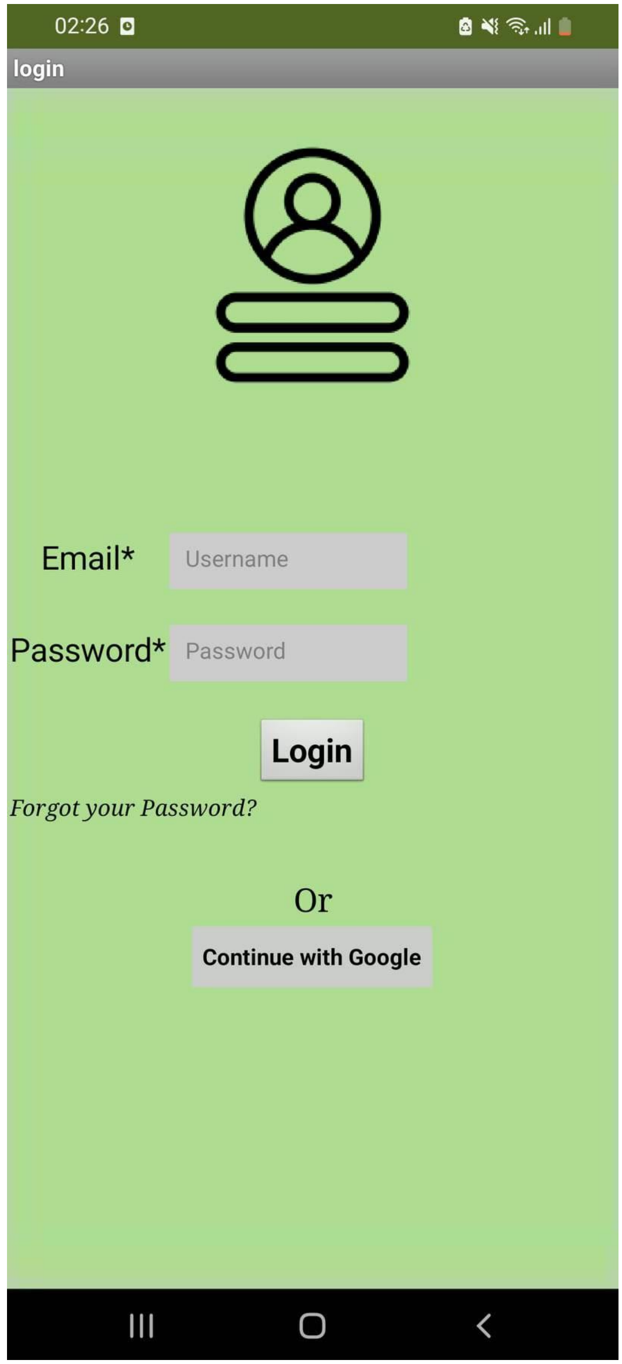

Figure 45: Authentication interface

The login interface (Figure 44) illustrates the possibility to choose to log in as an admin or as a farmer (user).

Figure 45 is the authentication interface where a user or an admin can have access to the application by inserting their user name or email and a password or just logging in with a Google account.

Figure 45:Admin authentication

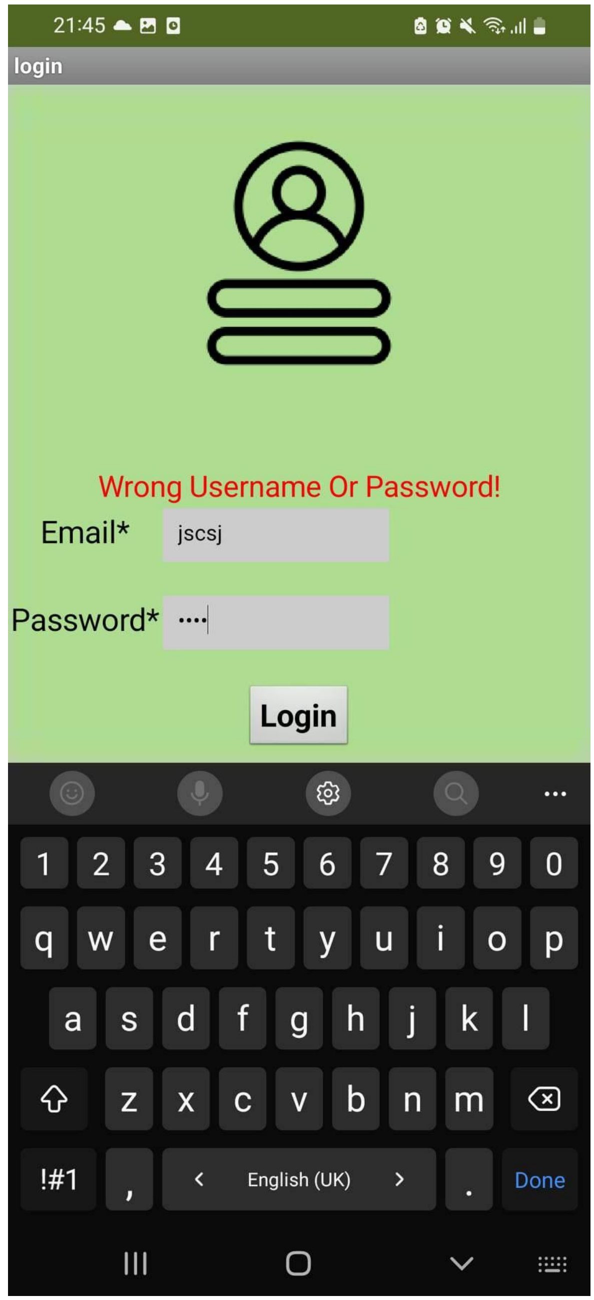

Figure 47:Error notification

The above figures illustrate an example of an admin authentication and an error message after logging in with an incorrect username or password.

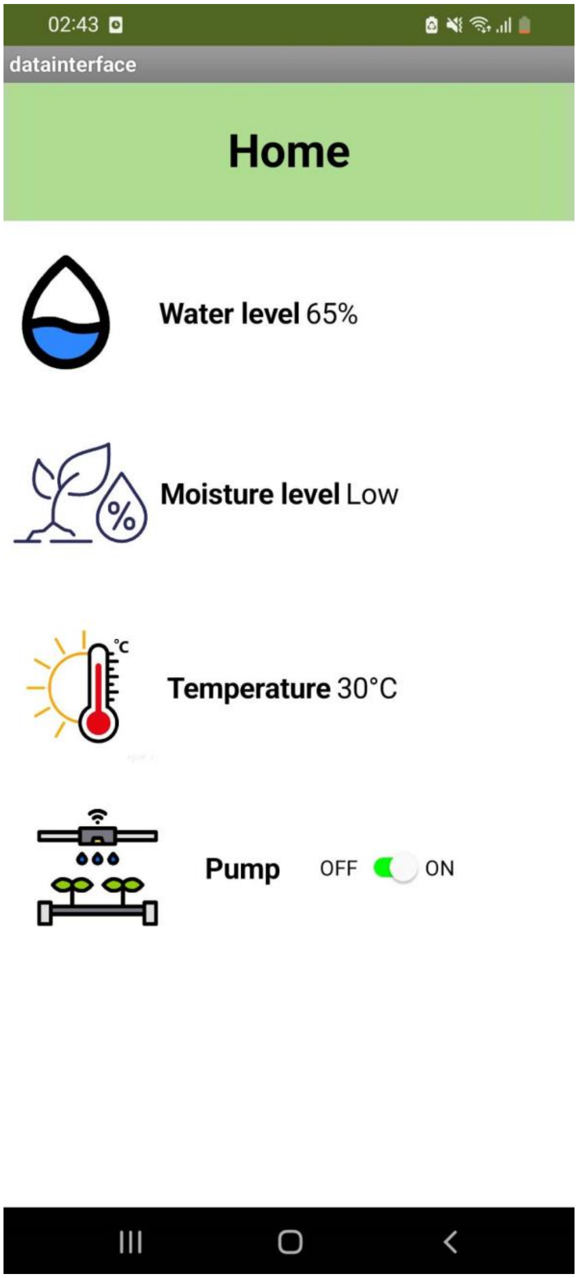

Figure 46: Home page1

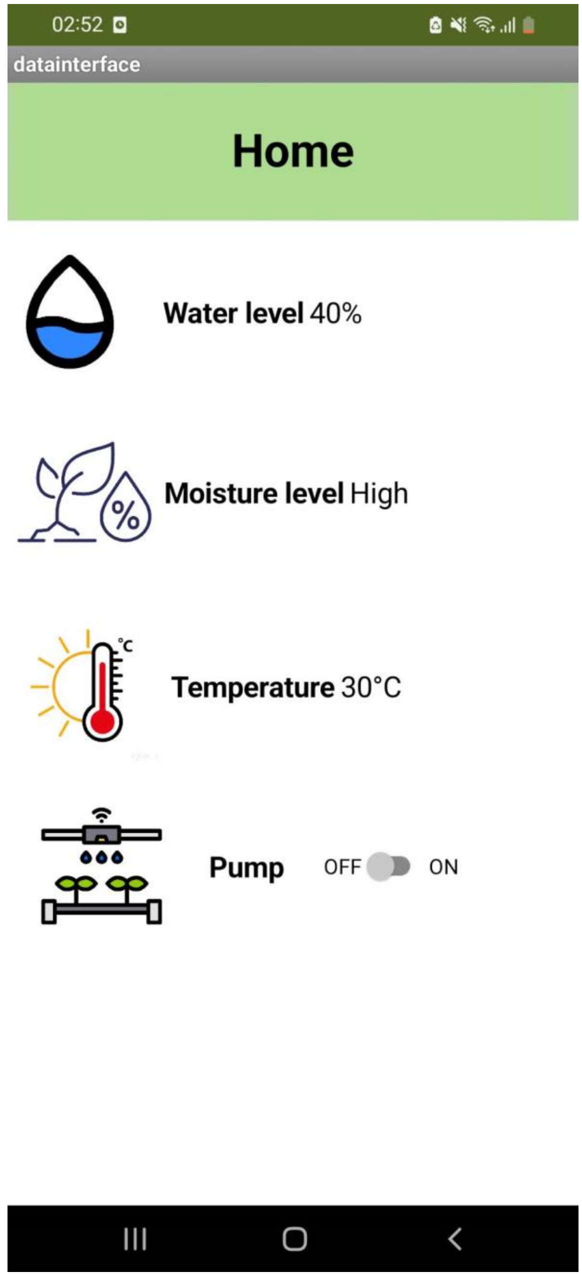

Figure 49: Home page2

The above figures are example of displayed data readout from different sensors.

The first case: (figure 48): shows the temperature value and the water level indicating the amount of water in a storage tank beside a low value of moisture, which leads to the pump turning on.

The second case: (figure 49): shows the temperature value and a decrease in the level of water in storage beside a high value of moisture after the irrigation which leads to the pump being turned off.

Conclusion

In this last chapter, we have presented the different stages of development of our proposed solution, including the development of the mobile application and the programming of the embedded system. In addition, we have explained, in detail, the general functioning of the project by doing the tests and interpreting the results obtained.

General Conclusion

The objective of this project is to develop an artificial irrigation system. This system has the role to measure the level of soil moisture and temperature as well as the level of water in the water storage tank and display them in real-time on an interface (mobile application) taking the decision to open or close the water pump.

Thus, in the first chapter of this report, we presented the project context and its objectives.

We have reserved a part to study the existing solutions, indicating the advantages and disadvantages of each of them and presenting the technology (IoT) related to our proposed solution.

In the second chapter, we discussed the functional and non-functional requirements and the treatment of the use case diagrams, the class diagram, and the sequence diagram. The activity diagram included in this part of the report describes the functioning of the application.

In the last chapter, we have chosen the hardware and software environments for the development of the solution. Then we described the realization of the mobile application, displaying its interfaces and its operating principle. Without forgetting the embedded part and the electrical connection.

Finally, we performed tests and interpreted the results found. These results show the achievement of the project objectives.

This work is not finished yet. The project may undergo modifications such as the use of LORA technology and the development of a mobile application that will help in the prediction of the weather and give the hand to the farmer to control the opening and closing of the pump.

Webography

[1] " Web site " : https://www.researchgate.net/publication/336819163_AI-Based_Yield_Prediction_and_Smart_Irrigation

[2] Ieee "Web site": https://ieeexplore.ieee.org/document/4350806

[3] Healthy water "Web site": https://www.cdc.gov/healthywater/other/agricultural/types.html

[4] "Web site" : https://www.imda.gov.sg/-/media/imda/files/industry-development/infrastructure/technology/internetofthings.pdf

[5] Huawei "Web site" : https://www.huawei.com/apac/huawei-ecoconnect/iot.html

[6] Hindawi "Web site" : https://www.hindawi.com/journals/jece/2017/9324035/

[7] Lucidchart "Web site" : https://www.lucidchart.com/pages/what-is-UML-unified-modeling-language

[8] Arduino store "Web site" : https://store.arduino.cc/products/arduino-uno-rev3

[9] Electronicsforu.com "Web site": https://www.electronicsforu.com/electronics-projects/prototypes/smart-irrigation-system

[10] Elprocus "Web site" : https://www.elprocus.com/a-brief-on-dht11-sensor/

[11] Lastminuteengineers "Web site": https://lastminuteengineers.com/water-level-sensor-arduino-

[12] Mybotic "Web site": https://www.mybotic.com.my/motor-and-motor-driver/water-pump/micro-submersible-water-pump-dc-3v-5v

[13] Components "Web site": https://components101.com/switches/5v-four-channel-relay-module-pinout-features-applications-working-datasheet

[14] Techtarget "Web site": https://www.techtarget.com/whatis/definition/resistor

[15] Protosupplie "Web site": https://protosupplies.com/product/7-color-flashing-led-module/

[16] Firebase "Web site": https://firebase.google.com/

[17] Appinventor "Web site": https://appinventor.mit.edu/

Annex