



RÉPUBLIQUE ALGÉRIENNE DÉMOCRATIQUE ET POPULAIRE
MINISTÈRE DE L'ENSEIGNEMENT SUPÉRIEUR ET DE LA RECHERCHE SCIENTIFIQUE

UNIVERSITÉ IBN-KHALDOUN DE TIARET

FACULTÉ DES SCIENCES APPLIQUEES
DÉPARTEMENT DE GENIE ELECTRIQUE

MEMOIRE DE FIN D'ETUDES

Pour l'obtention du diplôme de Master

Domaine: Sciences et Technologie

Filière: Automatique

Spécialité : Automatique et informatique Industrielle

THÈME

Intelligent Management of a Hydroponic Greenhouse Powered by a PV
System Using Internet of Things

Préparé par :
Benfatma hadjer
Boudjenah saliha ouiem

Devant le Jury:

Nom et prénoms	Grade	Qualité
H.BOUMEDIENE		Président
M. SEBAA	Pr	Examineur
H.BENABID		Examineur
T. ALLAQUI	Pr	Encadreur

Acknowledgment:

First and foremost I thank **Allah** for his blessings, showered on me in completing the project successfully.

I am heartily thankful to my teacher and supervisor **Mr. Allaoui Tayeb** whose encouragement, guidance and help from the initial to the final level enabled me to develop and understanding the subject.

I would also like to express my special thanks of gratitude to the **Members of Jury** who agreed to evaluate my work.

A very big thank you to **My Teachers and Monitors**, who contributed to my training during my studies at **Ibn Khaldoun University**, and especially **My Teachers** in the **Electric Engineering Department**.

Secondly, a special thanks to **Mr. Sofiane Maaradji** the student from previous promotion, **Mr. Mustapha Laoumir** and **Mr. Azouz Hakim** from **Computer Science Department** for their enlightening orientations, their constructive advice and their valuable help as well as all members of the **L2GEGI** laboratory.

I would also like to thank **My Dear Parents, Brothers, Sisters** and **Friends** who have always supported me.

My sincere thanks go to all those who have contributed directly or indirectly in the realization of this thesis.

Last but not least, I Like to thank **Me** for believing in me, I like to thank **Me** for doing all this hard work, I Like to thank **Me** for having no days off, I Like to thank **Me** for never quitting.

Table of content

Table of content :

Acknowledgment:	I
Table of content	II
List of figures	VI
List of tables:	IX
List of abbreviations:	X

General Introduction

General Introduction:	2
------------------------------------	----------

Chapter I: Hydroponics Greenhouses

I.1. Introduction:	5
I.2. Greenhouse:	5
I.2.1. Definition:	5
I.2.2. Choice of greenhouse:	5
I.2.3. Greenhouse architectures:	6
I.2.4. Classification of greenhouse:	9
I.2.5. Environmental conditions in greenhouse:	9
I.2.6. Control of Environmental conditions:	10
I.2.7. Pros of Greenhouse:	11
I.2.8. Cons of Greenhouse:	11
I.3. Hydroponic cultivation:	11
I.3.1. Definition:	11
I.3.2. History:	12
I.3.3. Passive and Active Hydroponic:	12
I.3.4. Different types of hydroponic system:	13
I.3.5. Benefits of Hydroponic:	19
I.4. Conclusion:	19

Table of content

Chapter II: Internet of Things

II.1 Introduction:	21
II.2 overview on internet of things:	22
II.2.1 Definition of IoT:	22
II.2.2 Characteristics:	23
II.2.3 IoT Architectures:	23
II.2.3.1 IoT Physical Architecture (IPA):	23
II.2.3.2 IoT Layered Architectures:	24
II.2.3.2.1 IoT 3 Layered Architecture:	25
II.2.3.2.2 IoT 5 Layered Architecture:	26
II.2.4 IoT Elements:	26
II.2.4.1 Identification:	27
II.2.4.2 Sensing:	27
II.2.4.3 Communication:	28
II.2.4.4 Computation:	28
II.2.4.5 Services:	28
II.2.4.6 Semantic:	28
II.2.5 How IoT Work:	28
II.2.6 IoT Application:	29
II.2.6.1 Building and Home automation:	30
II.2.6.2 Smart Cities:	30
II.2.6.3 Manufacturing Industry:	31
II.2.6.4 Health Care:	32
II.2.6.5 Smart Farming:	33
II.3 Conclusion:	33

Chapter III: Hardware and Software Description

III.1 Introduction:	35
III.2 Hardware part:	36
III.2.1 Choice of microcontroller:	36
III.2.2 Sensors:	36
III.2.2.1 Temperature & Humidity Sensor:	37
III.2.2.2 Light sensor:	38
III.2.2.3 TDS (Total Dissolved Solids) sensor:	39
III.2.2.4 DS18B20 Arduino Temperature sensor:	40

Table of content

III.2.2.5 PH sensor:	41
III.2.3 Actuators:.....	42
III.2.3.1 water pump:.....	42
III.2.3.2 Air pump:	43
III.2.3.3 Peristaltic pump:.....	44
III.2.3.4 Fan:.....	45
III.2.3.5 Relay:	45
III.2.3.6 LCD:.....	46
III.2.3.7 Heating:.....	47
III.2.3.8 Float:.....	48
III.2.3.9 Lighting:	48
III.2.3.10 ESP32 Camera:.....	49
III.2.4.1 PV:.....	50
III.2.4.2 Inverter :	51
III.2.4.3 Voltage Regulator:	52
III.2.4.4 Fuses and Circuit Breakers:.....	52
III.2.4.5 Level shifter:.....	53
III.3 Software part:.....	54
III.3.1 Arduino IDE:	54
III.3.2 Interfacing ESP32 with Arduino IDE:.....	55
III.3.3 Google Firebase:.....	56
III.3.4 Connecting Arduino to Firebase:.....	57
III.3.5 Android Application:	58
III.3.6 Android Studio:	58
III.3.7 Kotlin:.....	59
III.3.8 XML:	59
III.3.9 Building android app with android studio:	60
III.3.10 Connecting firebase with android studio:	62
III.4 Conclusion:	62

Chapter IV: Realization

IV.1 Introduction:.....	64
IV.2 Overall function of the circuit:.....	64
IV.3 Function of each part:	65
IV.3.1 Temperature and humidity:	65
IV.3.2 PH:.....	66

Table of content

IV.3.3 EC:.....	68
IV.3.4 Light:	70
IV.4 Circuit:	71
IV.4 Testing of sensors:	71
IV.4.1 Temperature sensor:	71
IV.4.2 Light sensor:	73
IV.4.3 EC and PH:.....	73
IV.5 Application:	74
IV.5.1. App interfaces:.....	74
IV.6 Conclusion:	80

General Conclusion

General Conclusion:.....	82
References:	84

List of figures

List of figures

Figure I. 1: Lean-To greenhouse	6
Figure I. 2: Gable greenhouse	7
Figure I. 3: Sawtooth greenhouse.....	7
Figure I. 4: Dome greenhouse	8
Figure I. 5: Tunnel greenhouse	8
Figure I. 6: Hydroponic cultivation.....	12
Figure I. 7: Wick system	13
Figure I. 8: Drip system.....	14
Figure I. 9: Ebb and Flow system	15
Figure I. 10: Water culture system	16
Figure I. 11: Nutrient Film Technique	17
Figure I. 12: Aeroponic system	18
Figure II. 1: internet of things	22
Figure II. 2: IoT Architectures	25
Figure II. 3: IoT elements.....	27
Figure II. 4: How internet of things Work diagram.	29
Figure II. 5: IoT for smart home	30
Figure II. 6: IoT for Smart city.....	31
Figure II. 7: IoT for Industry	32
Figure II. 8: IoT for healthcare.	32
Figure II. 9: Smart farming illustration.	33
Figure III. 1: Synoptic Diagram	35
Figure III. 2: ESP32Dev.Board Pinout.	36
Figure III. 3: DHT11 and DHT22 sensors.	37
Figure III. 4: LDR sensor.	38
Figure III. 5: TDS sensor.....	39

List of figures

Figure III. 6: DS18B20 Arduino Temperature sensor.....	40
Figure III. 7: PH sensor	42
Figure III. 8: Water Pump	43
Figure III. 9: Air Pump.....	44
Figure III. 10: Peristaltic Pump	44
Figure III. 11: Fan	45
Figure III. 12: Relay	46
Figure III. 13: LCD	47
Figure III. 14: Heating.....	48
Figure III. 15: Float	48
Figure III. 16: Light.....	48
Figure III. 17: ESP32-CAM	50
Figure III. 18: Solar Panel	51
Figure III. 19: Inverter.....	51
Figure III. 20: Regulator.....	52
Figure III. 21: Fuses	53
Figure III. 22: Fuses	53
Figure III. 23: TXS0108E level-shifter	54
Figure III. 24: Arduino logo	55
Figure III. 25: Google Firebase logo	56
Figure III. 26: Android Logo.....	58
Figure III. 27: Android Studio Logo	59
Figure III. 28: Kotlin Logo.....	59
Figure III. 29: XML File.	61
Figure III. 30: Kotlin File.....	61
Figure III. 31: Gradle File.	62
Figure IV. 1: Functionality of Greenhouse.	65
Figure IV. 2: Temperature and Humidity Flowchart.	66
Figure IV. 3: PH Flowchart.....	68
Figure IV. 4: EC Flowchart.....	69
Figure IV. 5: Light Flowchart.	70
Figure IV. 6: Electrical Circuit.....	71

List of figures

Figure IV. 7: Testing of DHT22 sensor.	72
Figure IV. 8: Result of test	72
Figure IV. 9: Testing of Light sensor.	73
Figure IV. 10: Result of test.	73
Figure IV. 11: Testing EC and PH.	74
Figure IV. 12: GreenTouch Logo.....	74
Figure IV. 13: SplashScreen.....	75
Figure IV. 14: Getting Started Interface.....	75
Figure IV. 15: Register Interface.....	75
Figure IV. 16: Login Interface	75
Figure IV. 17: Home Interface.	76
Figure IV. 18: Humidity and Temperature interface.....	76
Figure IV. 19: PH interface.	76
Figure IV. 20: EC interface.	78
Figure IV. 21: Light interface.....	78
Figure IV. 22: App services	78
Figure IV. 23: streaming interface.	79
Figure IV. 24: Searching interface.	79
Figure IV. 25: Dark mode	80

List of tables

List of tables:

Table I. 1: Passive and Active Hydroponic	13
Table I. 2: Wick system.....	14
Table I. 3: Drip system.	15
Table I. 4: Ebb and Flow system.....	16
Table I. 5: Water culture system.	17
Table I. 6: NFT system.....	18
Table I. 7: Aeroponic system.	19
Table III. 1: DHT11 and DHT22 sensors data sheet.....	38
Table III. 2: LDR data sheet.....	39
Table III. 3: TDS data sheet.	40
Table III. 4: DS18B20 Arduino Temperature sensor data sheet.....	41
Table III. 5: PH data sheet.....	42
Table III. 6: Water pump data sheet.....	43
Table III. 7: Air pump data sheet.	43
Table III. 8: Peristaltic pump data sheet.....	44
Table III. 9: Fan data sheet.	45
Table III. 10: Relay data sheet.....	46
Table III. 11: LCD data sheet.....	47
Table III. 12: ESP32-CAM datasheet.	50
Table III. 13: Inverter data sheet.	52
Table III. 14: TXS0108E level-shifter data sheet.....	54

List of abbreviations

List of abbreviations:

UN: United Nations.

PV: Photovoltaic.

IOT: Internet of Things.

DWC: Deep Water Culture.

NFT: Nutrient Film Technique.

NIC: National Intelligence Council.

RFID: Radio Frequency Identification.

LAN: Local Area Network.

IP: Internet Protocol.

IPA: IoT Physical Architecture.

ILA: IoT Layered Architecture.

3G: Third Generation.

4G: Fourth Generation.

LTE: Long Term Evolution.

DBMS: Data Base Management System.

ID: Identification.

EPC: Electronic Product Codes.

uCode: Ubiquitous Code.

IPv4: Internet Protocol Version 4.

IPv6: Internet Protocol Version 6.

RDF: Resource Description Framework.

OWL: Web Ontology Language.

WAN: Wide Area Network.

PH: Potential Hydrogen.

EC: Electrical Conductivity.

GPIO: General Purpose Input/output.

List of abbreviations

IO: Input Output.

TSMC: Taiwan Semiconductor Manufacturing Company.

DHT11: Digital Humidity Temperature Sensor.

DO: Digital Output.

AO: Analog Output.

AD: Analog Digital.

TDS: Total Dissolved Solids Sensor.

DC: Direct Current.

AC: Alternative Current.

LCD: Liquid Cristal Display.

IDE: Integrated Development Environment.

JSON: JavaScript Object Notation.

API: Application Programming Interface.

HTTP: Hyper Text Transfer Protocol.

OS: Operating System.

IOS: iPhone Operating System.

JDK: Java Developer Kit.

XML: Extensible Markup Language.

HTML: Hyper Text Markup Language.

Wi-Fi: Wireless Fidelity.

ICT: Integration of Communication Technology.

General Introduction:

General Introduction:

General Introduction:

“End hunger, achieve food security, improved nutrition and promote sustainable agriculture”, this is one of the UN’s goals. The UN explains: “It is time to rethink how we grow, share and consume our food”.

Right now, soils across the globe are undergoing rapid degradation. Climate change has increased the risks associated with disasters and put huge pressure on agriculture for quality and sustainable food production. [1]

In Algeria, only 3.4% of the total area is used for agriculture and only 25% of groundwater is being exploited in the south of Algeria. Due to the arid climate, it is difficult to exploit agricultural land in the south. The earth loses its fertility; it becomes fragile, and faces a lack of nutrients for plants. In addition, frequent dry and hot southern winds cause a significant damage to agriculture [2]. One solution to tackle this problem is to use greenhouses, where environmental variables can be controlled to obtain optimal conditions for plant growth. These controlled environmental conditions can be complemented with hydroponic, a method of growing plants without soil. [3]

Even though the hydroponic greenhouse has been used for many years, it is still being used in a traditional way. Therefore, there is a need to modernize greenhouses with the integration of ICT technologies based on Internet of Things (IoT) to improve the energy efficiency and quality of the produces [4].

In this context and as part of our professional master's training in Automation and Industrial Computing at the Ibn Khaldoun University of Tiaret and to apply our acquired knowledge and improve our skills, we have been proposed to expand and improve the project done last year by Ali Asmaa a smart hydroponic greenhouse powered by a PV panel, by adding an ESP32 controller to be able to connect to the cloud via Wi-Fi. Using this communication network, data detected by sensors will be sent and stored in firebase database and then displayed using android app to allow a real-time monitoring and supervision of the greenhouse.

To achieve this goal, we began to familiarize with the greenhouse its architecture, circuits and modules. We then started with a literature review and background study to

General Introduction:

acquire a good understanding on how IoT technology can be embedded into the greenhouse. Finally, we started the development and deployment of the application.

This thesis consists of four main chapters:

- ❖ Chapter 1 focuses on hydroponics and agricultural greenhouses; we talked as much as possible about greenhouses (types and classification, conditions, advantages, disadvantages).
- ❖ Chapter 2 provides an overview on IoT technology (definition, architectures, and components) and presented some of its applications.
- ❖ Chapter 3 gives a detailed description of the hardware and software developed in this project.
- ❖ Chapter 4 presents the setup of our system, its validation and testing and the results obtained.

Chapter I: Hydroponics Greenhouses

Chapter I: Hydroponics Greenhouses

I.1. Introduction:

In near future, agriculture will face some serious problems like availability of land and agricultural productivity, deforestation, and soil erosion.

A potential solution to overcome these challenges, is the large-scale adoption of greenhouses, where environmental variables can be controlled to create optimal conditions for plant growth. These controlled environmental conditions can be complemented with hydroponic, a method of growing plants without soil.[5] This is likely to contribute to the preservation of water resources, a year-round production, increasing yields, and minimizing the use of pesticides without any digging or weeding, thanks to off ground culture, which facilitates the agriculture. [6]

In the following section, we present a detailed overview about greenhouses and soilless cultivation.

I.2. Greenhouse:

I.2.1. Definition:

A greenhouse is a pseudo-closed structure, intended to create an auspicious climatic environment for the protection of plants. It has walls and transparent covers to facilitate the penetration of light to the plants. The use of a greenhouse makes it possible to manage weather conditions, temperature, light, water, carbon dioxide levels, relative humidity and plant nutrients, as well as air quality inside the greenhouse. [7]

The main benefit of greenhouse is the availability of crops all seasons despite the changing weather conditions (rain, wind, cold).

I.2.2. Choice of greenhouse:

The main standards for choosing the greenhouse are: [8]

- ❖ Energy saving.
- ❖ Transmission of useful radiation for photosynthesis.
- ❖ Sustainable.
- ❖ Easy to build, to operate and to maintain.

Chapter I: Hydroponics Greenhouses

I.2.3. Greenhouse architectures:

There are many shapes of greenhouse, the main types are listed below: [9]

- a- **Lean to:** It is placed against an existing wall usually oriented to south as shown in figure I.1, this type of shape is more common for hobby farming or small greenhouses.



Figure I. 1: Lean-To greenhouse. [10]

- b- **Gable:** this is the most common configuration and looks like a traditional house as shown in figure I.2. Its design simplicity creates a natural ventilation by adding an aperture at the top of the structure.



Figure I. 2: Gable greenhouse. [11]

C- Sawtooth: is a solution to increase passive ventilation .Thus, more appropriate for very warm places. An example of it is shown in figure I.3.



Figure I. 3: Sawtooth greenhouse. [12]

Chapter I: Hydroponics Greenhouses

d- Dome: has a very stable internal environment and an excellent light distribution as shown in figure I.4. It is also more robust against the wind. But, it is costly to fabricate.



Figure I. 4: Dome greenhouse. [13]

e- Tunnel: has a simple curved structure as shown in figure I.5. However, it has poor spacing and layout considerations. It also needs active ventilation.



Figure I. 5: Tunnel greenhouse. [14]

I.2.4. Classification of greenhouse:

Greenhouses are a technology-based investment. The smarter the technology used, the better the control of the environmental conditions is achieved. For helping people to choose the right investment for their needs and budget, there are three categories of greenhouses: [15]

a- Low cost or low-tech greenhouse:

The height of low technology greenhouses is less than 3 meters. Tunnel houses are the most common architecture. They do not have straight walls, they have poor ventilation. This structure is not expensive and simple to build with little or no automation is involved.

Although this type offers production benefits, it limits yield and does not provide control of pests and diseases.

b- Medium-tech greenhouse:

Generally, medium-level greenhouses are characterized by vertical walls between 2 and 4 meters high and a total height less than 5.5 meters. They are covered by a single or double skin plastic or glass film. They have ventilation on the roof or on the side walls or both.

These greenhouses offer a compromise between cost and productivity and represent a reasonable economic and environmental base for the industry. Production in medium-sized greenhouses can be more efficient than field production.

c- Hi-tech greenhouse:

High-tech greenhouses have a wall height of 4 meters or more, with the roof peak of up to 8 meters above ground level. These structures usually can achieve an advanced agricultural performance. They will have roof ventilation and may also have side wall vents. Cladding may be plastic film (single or double), polycarbonate sheeting or glass. The use of pesticides can be significantly reduced.

Greenhouses with high-technology provide a highly productive, sustainable environment for high-tech industry and fresh produce. The investment should be aimed to install high-tech greenhouses.

I.2.5. Environmental conditions in greenhouse:

The growth of plants in greenhouses requires special weather conditions, these conditions are:

Chapter I: Hydroponics Greenhouses

- a. Light.
- b. Carbon dioxide.
- c. Air circulation.
- d. Humidity.
- e. Temperature.

I.2.6. Control of Environmental conditions:

In the following, a brief overview on the control of these conditions is presented: [16]

a- **Light:**

Plants need typically at least 8 hours of light per day for a satisfactory growth and production. Because sunlight is significantly reduced during winter and in cloudy places, additional lighting is required.

The control of light can be achieved by a light source based on electric lamps or through the reflection of the sunlight. Reflected light is very important especially for urban gardens as lightning can be reduced or increased as necessary.

b- **Carbon dioxide:**

Commercial greenhouses commonly use CO₂ generators to maximize production. During the design of a CO₂ system, yields will only increase if CO₂ is the “limiting factor”. This means that if all other variables (light, fertilizer, temperature/humidity, ph...) are not optimal, the benefits of increased CO₂ levels may not be realized.

c- **Air circulation :**

Air circulation is also an important factor that affects the development of the plants, altering energy transfers, transpiration and CO₂ absorption, which affects leaf size, as well as stem growth and yield. The rate of photosynthesis can be increased by 40 percent if the wind speed increases by 10 to 100 centimeters by second.

d- **Humidity:**

Air and soil humidity are two important factors for plants growing. If the humidity is high, humid air can be vented to the outside. Watering the ground, using swamp cooler type air conditioners and a misting system to spray fog into the air, will help increase humidity.

e- **Temperature:**

Chapter I: Hydroponics Greenhouses

The temperature of the inside and outside environment, the ground and water must also within appropriate ranges.

To achieve cooling, it is necessary to create evaporation by using an internal shade curtain which can be closed during cloudy days and opened (manually or with small motors) on sunny days as necessary.

For the heating of the greenhouse, a thermostat is used to regulate the temperature and deliver the required amount of heat. It can also be heated using gas with central heating.

I.2.7. Pros of Greenhouse:

There are several advantages in using protected cultivation, the main are: [17]

- ❖ Off season production of vegetable and fruit crops.
- ❖ Maximize production and yield.
- ❖ Control of environmental conditions.
- ❖ Protection against pests and diseases.

However, this does not mean that greenhouses do not have drawbacks.

I.2.8. Cons of Greenhouse:

- ❖ Monitoring, maintenance and control are necessary.
- ❖ It can be expensive to construct. [18]

I.3. Hydroponic cultivation:

I.3.1. Definition:

Searching for the definition of hydroponic in various books and articles, we found that, Webster's New World College Dictionary, Fourth Edition 1999, defines hydroponic as "The science of growing or the production of plants in nutrient-rich solutions or moist inert material, instead of soil" [19].

The yield of plants grown in off-ground is higher than in the ground, thanks to the direct control of plant nutrients.

Chapter I: Hydroponics Greenhouses



Figure I. 6: Hydroponic cultivation. [20]

I.3.2. History:

During the past several decades, many amateur and commercial gardeners have become interested in growing plants with their roots in an artificial medium instead of soil. This method of growing plants is commonly known as "hydroponics." It is also sometimes referred to as nutrient-solution culture, soilless culture, water culture, gravel culture, and nutriculture.

Soilless culture of plants is not new, one of the first experiments in water culture was carried out by Woodward in England in 1699. The aim of these experiments was to determine whether water or the solid portion of the soil was responsible for plant growth. By the mid-nineteenth century, Sachs and Knop, the real pioneers in this field, developed a method of growing plants without soil.

In the late 1920's and early 1930's, Dr. W. F. Gerick was able to grow plants successfully on a large scale using a laboratory technique of solution culture. Dr. Gerick used the term "hydroponics" to describe this method of growing plants. Today, hydroponics is used in commercial production, but it is employed mostly in those areas where soil is lacking or unsuitable for plant growth. Hydroponics is also a tool in plant research as well as a fascinating hobby. [21]

I.3.3. Passive and Active Hydroponic:

Hydroponic can be either active or passive, Table I.1 gives a comparison of these methods: [22]

Chapter I: Hydroponics Greenhouses

Passive	Active
<ul style="list-style-type: none">* No pump is involved, due to the capillary action of the growing medium surrounding the roots (for the bringing of nutrient solution to plants).* Compared to active, the amount of oxygen is reduced.* Maximum growth and yield.* Optimal use of water.	<ul style="list-style-type: none">* Better yield.* Automation of plants nutrition and irrigation.* With a pump, it sends the nutrient solution to the roots.* Best plants health.* Work reduced.

Table I. 1: Passive and Active Hydroponic [22].

I.3.4. Different types of hydroponic system:

There are many types of hydroponic system [23]:

a- Wick system:

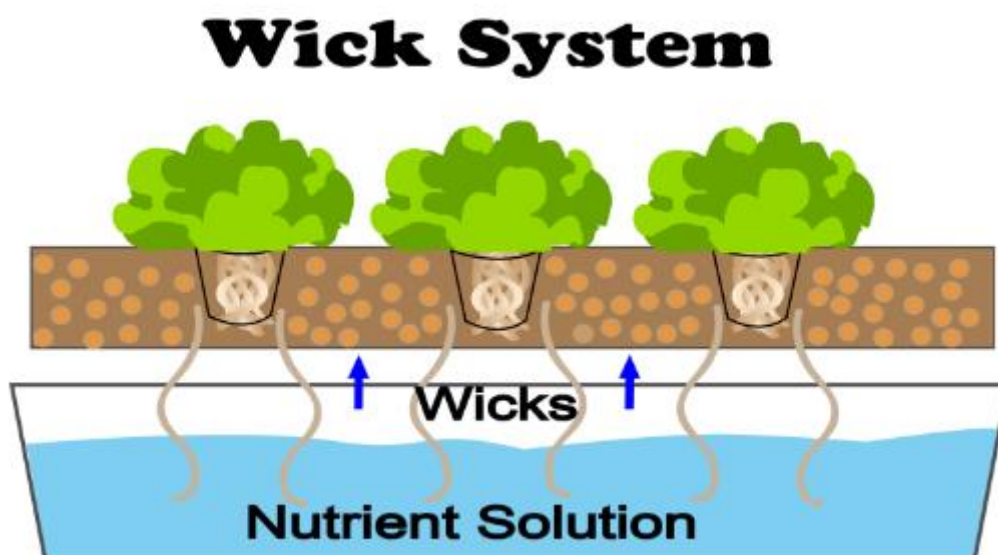


Figure I. 7: Wick system. [24]

Chapter I: Hydroponics Greenhouses

System	How it works?	Pros	Cons
Wick	*The bucket of plant is placed above a nutrient solution reservoir, and 2 to 4 wicks absorb solution from the reservoir to the plants roots. as shown in figure I.7	*Does not need pump or electricity to work. *Easy to build. *simplicity of using.	*less efficient when delivering nutrients to growing plant. *Do not really work well for larger plants, because it's need more water to grow. *Reservoir washing is absolute.

Table I. 2: Wick system. [24]

b- Drip system:

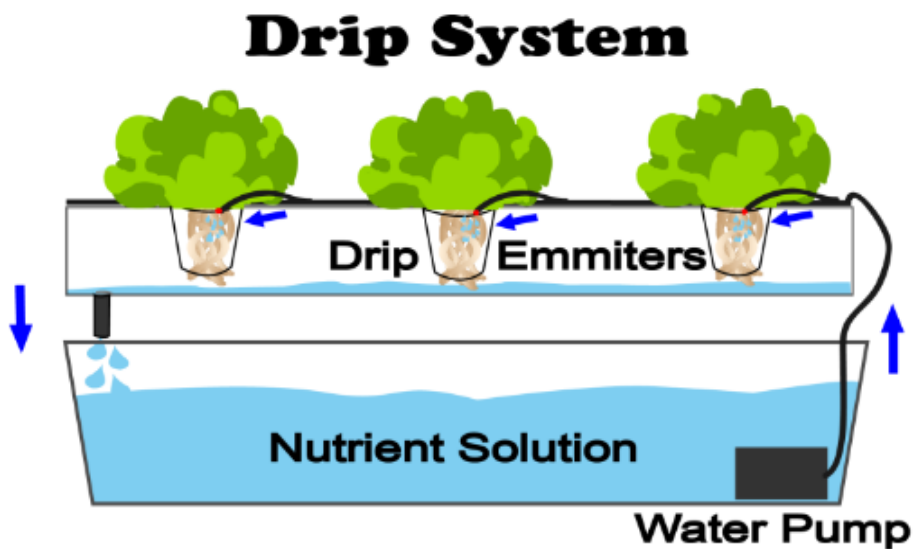


Figure I. 8: Drip system. [24]

Chapter I: Hydroponics Greenhouses

System	How to works?	Pros	Cons
Drip	*Nutrient solution is pumped up from the reservoir to the top of the growing media through tubing and dripped onto the plants roots as shown in figure I.8.	*Simpler to use. *The most widely used hydroponic system.	*Monitor the PH levels of nutrient solution, regularly.

Table I. 3: Drip system.

c- Ebb and Flow:

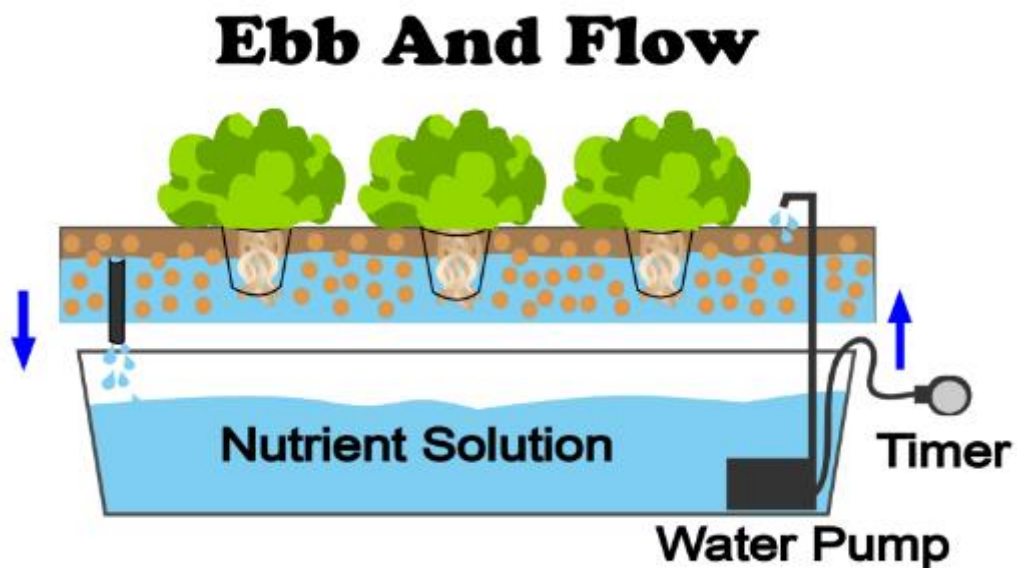


Figure I. 9: Ebb and Flow system. [24]

Chapter I: Hydroponics Greenhouses

System	How to works?	Pros	Cons
Ebb and Flow	*It works under the same concept to a “Drip system”, except the water pump is set up on a timer and nourishes the plants periodically. as shown in figure I.9	*Low cost. *Growing plants is very good.	*Vulnerable to power outages and pump failures.

Table I. 4: Ebb and Flow system. [24]

d- Water culture system:

Deep Water Culture (DWC)

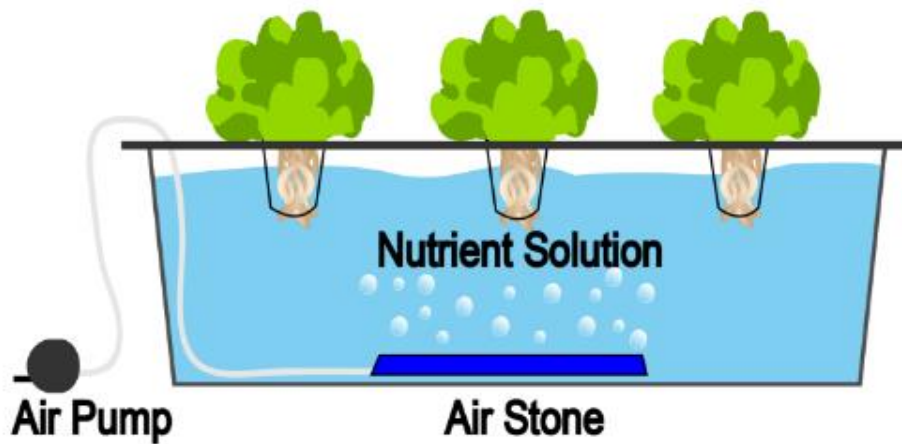


Figure I. 10: Water culture system [24].

Chapter I: Hydroponics Greenhouses

System	How to works?	Pros	Cons
DWC	<p>*Plants sit in buckets with roots extending into nutrient solution</p> <p>An aquarium pump and air stone or the action of water falling provides O₂ to the solution. as shown in figure I.10</p>	<p>*The using is simple.</p> <p>*Cost is very little.</p>	<p>*It needs a lot of water and electricity.</p>

Table I. 5: Water culture system. [24]

e- Nutrient Film Technique:

Nutrient Film Technique

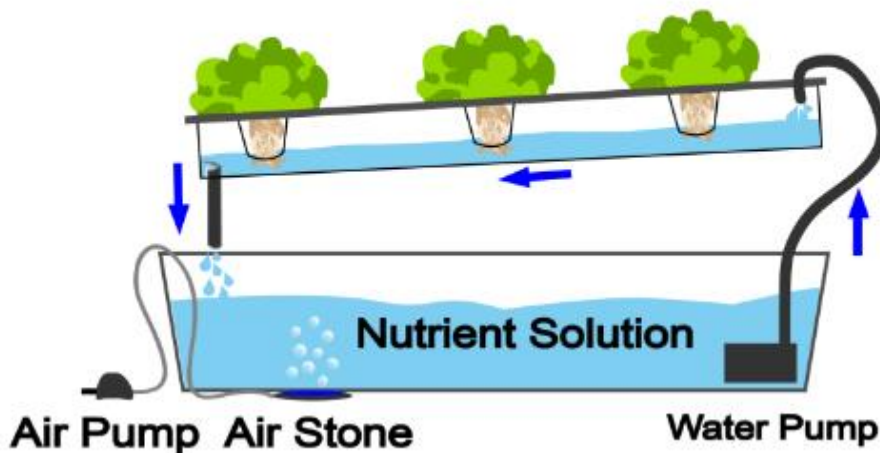


Figure I. 11: Nutrient Film Technique. [24]

Chapter I: Hydroponics Greenhouses

System	How to works?	Pros	Cons
NFT	*Plants and their roots sit extending into a channel, where a slim film of nutrient solution is distributed. as shown in figure I.11	*For small plants, it works well.	*If the flow of nutrient solution stops, the roots will dry out and become stressed very quickly.

Table I. 6: NFT system. [24]

f- Aeroponic system:

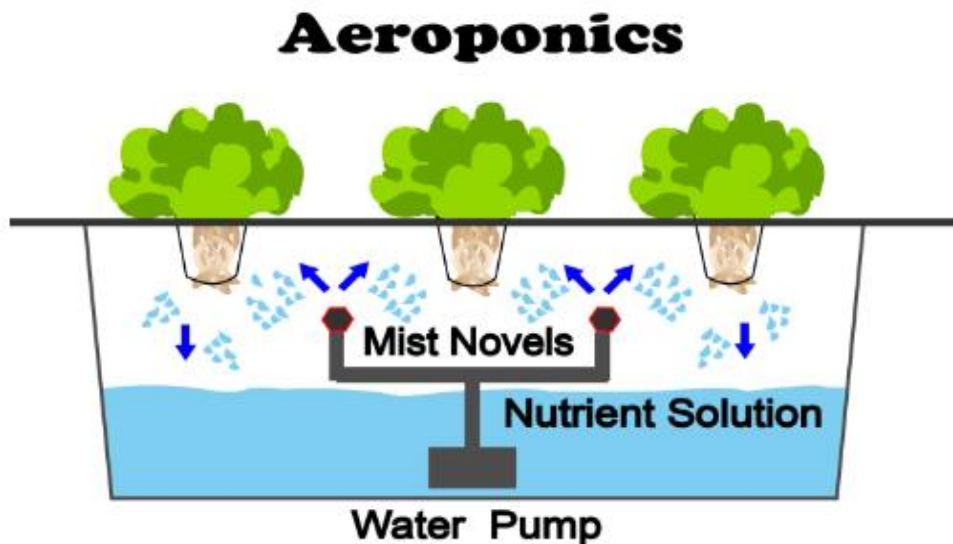


Figure I. 12: Aeroponic system. [24]

Chapter I: Hydroponics Greenhouses

System	How to works?	Pros	Cons
Aeroponic	*It nourishes plants and their roots in the air, which sprayed with nutrient solution. as shown in figure I.12	*Grow plants in small spaces. *Does not need a lot of water.	*The sprinkler heads must be clean regularly, to keep them from becoming clogged. *High cost.

Table I. 7: Aeroponic system. [24]

I.3.5. Benefits of Hydroponic:

- ❖ Hydroponic is water economy.
- ❖ Saving mineral fertilizers.
- ❖ Maximizes space (it needs smaller areas to produce hydroponic gardens).
- ❖ Higher yield.
- ❖ Doesn't need soil.
- ❖ Growing of crops is quickly.
- ❖ Well quality of food [25].

I.4. Conclusion:

In this chapter, we have presented a detailed overview about greenhouses (types and classification), we have also discussed how to choose and control a greenhouse, what conditions are involved, what are the advantages and disadvantages. In addition, we have briefly studied soilless cultivation.

Chapter II: Internet of Things

Chapter II: Internet of Things

II.1 Introduction:

Today there are billions of devices, sensors and controllers in homes, factories, hospitals, cars and farms that can be connected with each other to create incredible applications(as shown in figure II.1). Be the simple systems that turn on a light bulb or controllers that can adjust the output of a wind turbine, a flow monitor that can make adjustments on the fly in a manufacturing line or smart car that can report issues before they happen. How do we communicate with these devices? How do we link them together and create Smart systems? How do we analyze the data from these devices? The answer is IoT internet of things. [26]

According to Juniper Research’s latest report, the average number of connected devices per household in 2020 was 10. [27] And number of connected devices in 2021 was over 10 billion. [28] And According to GSMA Intelligence, there will be over 25 billion IoT connections by 2025. [29]

The “Smart agriculture” or “smart farming” market is predicted to be worth \$22 billion by 2025 as farmers worldwide seek to utilize RFID, GPS, drones, sensors, and more to gather actionable data and to automate every part of the process. [30]

This chapter is devoted to a general study of IoT systems to understand its features and its applications, in order to be able to develop an Internet of Things system for the intelligent management of hydroponic greenhouse.



Figure II. 1: internet of things. [31]

II.2 overview on internet of things:

II.2.1 Definition of IoT:

Defining the term IoT can be somewhat difficult because it has many definitions depending on who is defining the term. In this thesis we have chosen to follow the definition of IoT proposed by cloud.google.com The IoT is “the use of network-connected devices, embedded in the physical environment, to improve some existing process or to enable a new scenario not previously possible. These devices, or things, connect to the network to provide information they gather from environment through sensors, or to allow other systems to reach out and act on the world through actuators.” [32] And the one From the National Intelligence Council NIC “Internet of Things” to refer to the general idea of things, especially everyday objects, that are readable, recognizable, locatable, addressable, and/or controllable via the Internet—whether via RFID, wireless LAN, wide-area network, or other means. Everyday objects includes not only the electronic devices we encounter every day, and not only the products of higher technological development such as vehicles and equipment, but things that

Chapter II: Internet of Things

we do not ordinarily think of as electronic at all—such as food, clothing, and shelter; materials, parts, and subassemblies; commodities and luxury items; landmarks, boundaries, and monuments; and all the miscellany of commerce and culture.”[33]

II.2.2 Characteristics:

- 1) **Dynamic & Self Adapting:** IoT devices and systems may have the capability to dynamically adapt with the changing contexts and take actions based on their operating conditions, user’s context or sensed environment. Ex: the surveillance system is adapting itself based on context and changing conditions.
- 2) **Self Configuring:** allowing a large number of devices to work together to provide certain functionality.
- 3) **Inter Operable Communication Protocols:** support a number of interoperable communication protocols and can communicate with other devices and also with infrastructure.
- 4) **Unique Identity:** Each IoT device has a unique identity and a unique identifier (IP address).
- 5) **Integrated into Information Network:** that allow them to communicate and exchange data with other devices and systems.

II.2.3 IoT Architectures:

IoT architectures are mainly classified into two categories such as IoT physical architecture (IPA) and IoT layered architecture (ILA). [34]

II.2.3.1 IoT Physical Architecture (IPA):

The architecture of IoT consists of four building blocks, they are: [35]

A. Sensors:

Sensors are all accomplished the place, sensors sense the information from the environment

B. IoT Gateways and systems :

As the name properly states, it is an entry path to the internet for every one of the things/gadgets that we need to be associated with it. It gathers data from the sensor hubs, and it will transmit to the internet.

Chapter II: Internet of Things

C. Cloud server:

Data is transmitted through an entryway is kept away and handled securely inside the cloud server for example in server farm using data observation. To make gadgets and smart devices, keen activities are performed with the help of prepared information.

D. Mobile applications:

The natural portable applications will help end clients to control and screen gadgets (going from room indoor regulator to vehicle motors) from remote areas. These applications push the imperative data on your advanced mobile phones, tablet and so on.

II.2.3.2 IoT Layered Architectures:

There is no single and general agreement about the architecture of IoT that is agreed on by the whole world and researchers. Many and different architectures have been proposed by researchers as shown in figure II.2. According to some researchers, IoT architecture has three layers, but some researchers support the four-layer architecture. They think that, due to enhancement in IoT, the architecture of three layers cannot fulfill the requirements of applications. Due to a challenge in IoT regarding security and privacy, the architecture of five layers has also been proposed. It is considered that a recently proposed architecture can fulfill the requirements of IoT regarding security and privacy. [36]

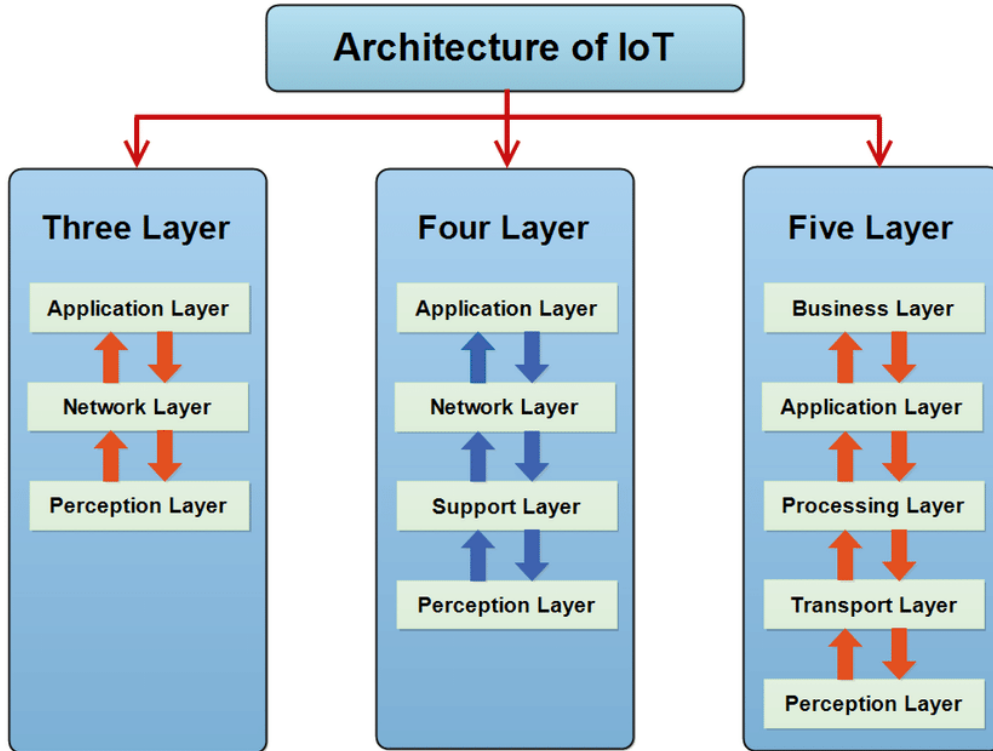


Figure II. 2: IoT Architectures. [36]

II.2.3.2.1 IoT 3 Layered Architecture:

The most basic architecture is a three-layer architecture. It was introduced in the early stages of research in this area. It has three layers, namely, the perception, network, and application layers. [37]

1. Perception Layer :

This layer also called as physical layer, gathers data/information and recognizes the physical world. In this layer all the actuators work according to the information which is collected by the sensors of different object in order to perform specific operations by the corresponding objects. [38]

2. Network Layer :

Network layer is the middle one, it establishes an interface link between application layer and perceptual layer. It is responsible for the initial processing of data, broadcasting of data and connecting devices. [39]

Chapter II: Internet of Things

3. Application Layer :

Application layer is the implementation of IoT. The working of sensors and actuators is achieved by application layer. We can understand it as software which works on and for the sensors other virtually intelligent objects. [40]

II.2.3.2.2 IoT 5 Layered Architecture:

. The difference between the Five Layer IoT Architecture and the Three Layer IoT Architecture are the addition of the following: transport, processing, and business.

The role of the perception and application layers is the same as the architecture with three layers. We outline the function of the remaining three layers.

4- Transport layer:

Takes the data from the perception layer and pass this data to the next layer which is processing layer and vice versa. This will done with the help of networks like LAN, wireless technology, 3G, 4G, LTE, RFID etc. [39]

5-Processing layer:

Which is third layer has to perform the major task because it will process all the information gathered by the perception layer. There is a huge amount of data which will be stored with the help of some techniques like cloud computing or any DBMS. Then it will analyze how to fetch data whenever required in order to complete the desired task. [41]

6-The business layer:

Manages the whole IoT system, including applications, business and profit models, and users' privacy. [37]

II.2.4 IoT Elements:

IoT provides many benefits and facilities to users. Thus, in order to use them properly, there is a need for some elements such as identification, sensing, communication, computation, services and semantic as shown in next figure

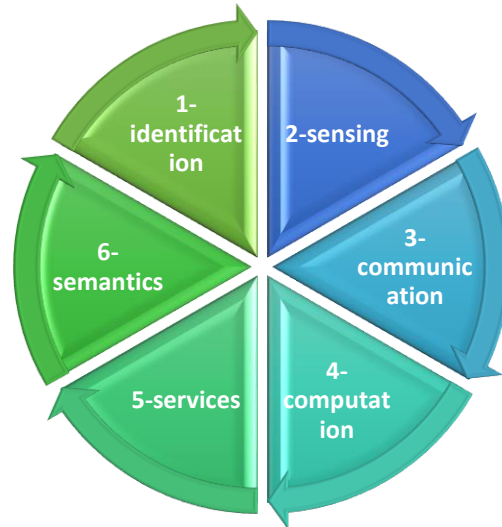


Figure II. 3: IoT elements.

II.2.4.1 Identification

Identification plays a central role in the Internet of Things since service requests can only add value to the customer if they are sent to the proper physical object within the network. Therefore, different methods are used for object identification. However, when addressing a specific IoT device, one has to distinguish between its ID and its address. The ID is considered the objects' name, for example 'S1' for a certain sensor, and can exist multiple times. Consequently, the address is necessary to identify objects within a network. Actually many identification methods exist such as electronic product codes (EPC) and ubiquitous code (uCode). To assign the unique address to each object, IPv6 is used. Firstly, IPv4 was used to assign the address but it could not fulfill the need of addressing due to large amount of IoT devices. Therefore, IPv6 is used because it uses 128 bit number addressing scheme [36-42]

II.2.4.2 Sensing:

This involves acquiring information from the environment and transferring it to a local, remote or Cloud-based database as an instance of the IoT sensor. We can identify intelligent devices, portable sensors or actuators. The collected information is transmitted to the storage medium. Numerous detection devices to collect data on objects such as RFID tags, actuators, portable sensors, smart sensors, etc. [43]

Chapter II: Internet of Things

II.2.4.3 Communication:

Communication involves the exchange of information between the IoT objects and allow them to interact with each other. Different technologies and protocols for communication are being used such as: Bluetooth, Zigbee, z-wave and Wi-Fi.

II.2.4.4 Computation:

Computation is considered as the “brain” of the IoT. It is the element that is responsible for processing operation. It constitutes the hardware platform such as Arduino, UDOO, FriendlyARM, Intel Galileo, Raspberry PI and Gadgeteer, etc. and the software platform such as TinyOS, LiteOS and RIoTOS, etc.

II.2.4.5 Services:

The different IoT services can be categorized as Identity related services, Information aggregation services, Collaborative-Aware services, and Ubiquitous services. [44]

Identity-related Services: most basic services which supports other services, Information Aggregation Services: collects raw data from sensors and supplies it to the IoT applications, Collaborative-Aware Services: rely on Information Aggregation Services and take decision on the collected data, and Ubiquitous Services: provide Collaborative-Aware Services anytime they are needed to anyone who needs them anywhere. [45]

II.2.4.6 Semantic:

The last IoT-enabling element is Semantics which describes the ability of different machines to extract and analyze relevant information from the same source. This information is then used to provide the service that was requested by the user. [39] Thus, Semantic Web technologies examples are the Resource Description Framework (RDF) and the Web Ontology Language (OWL).

II.2.5 How IoT Work:

The foremost component to consider in Internet of Things technology is sensor/devices. A sensor picks up all the minute details from an environment. The environment can have many complexities. What makes IoT security so great is these sensors that pick up even the most sensitive changes. These sensors are built in the devices which

Chapter II: Internet of Things

collect all the data to be used later. For instance, our phone is a device with built-in sensors like GPS, camera, etc.

Once the data is collected it is transferred to the cloud infrastructure (also known as IoT platforms). But to transfer the data, the devices will need a medium. That's when connections like Bluetooth, Wi-Fi, WAN, cellular networks, etc come into play. These mediums are all different and must be chosen wisely for best results. The effectiveness of IoT security highly depends on the speed and availability of these mediums.

After reaching the cloud infrastructure the data has to be analyzed so that the right action can be taken. This process is however considered one of the most crucial obstacles in front of IoT app development. The analysis can be as simple as checking the temperature of the AC or a complex one such as a situation where an intruder comes in and the device has to identify it through cameras. The IoT application is made such that it can process all the data at a fast rate to take immediate actions.

The last step is when the user is notified about the action with the help of a notification or an alert sound sent to the IoT mobile apps. This way the user will know that his command has been run through the systems [46], and that's what the figure II.4 resume.

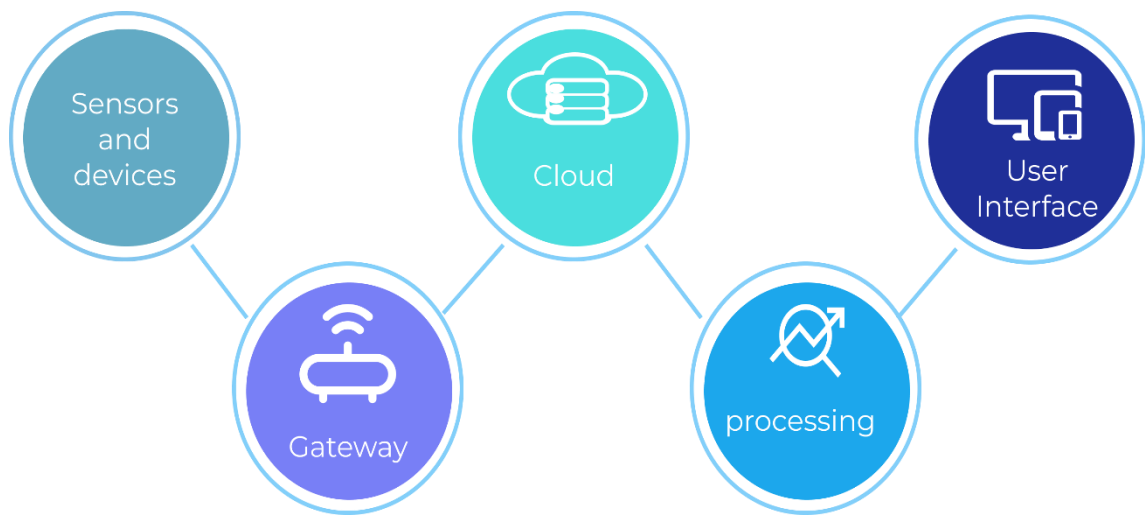


Figure II. 4: How internet of things Work diagram.

II.2.6 IoT Application:

There are several application domains which will be impacted by the Internet of we will state below some of the main ones.

II.2.6.1 Building and Home automation:

When we talk about the Internet of Things sub-field home automation, then we come to the point of controlling home appliances automatically from anyplace. From the enhancement of security to reducing energy & maintenance costs, the Texas instruments starts a new variety of products for their user, a wide range of innovative IoTs technologies for monitor & control of intelligent buildings & smart homes. The major applications related with the home automation side are: Access Control, Energy Optimization, Light & Temperature Control, and Predictive Maintenance, All the Devices are on a Single Network [47]. As shown in figure II.5



Figure II. 5: IoT for smart home. [48]

II.2.6.2 Smart Cities:

A smart city is an urbanization region that collects data using several digital and physical devices. The information collected from such devices is used efficiently to manage revenues, resources, and assets, etc., while the information obtained from such devices is utilized to boost performance throughout the city. Cloud-based Internet of Things (IoT) applications could help smart cities that contain information gathered from citizens, devices, homes, and other things. This information is processed and analyzed to monitor and manage

Chapter II: Internet of Things

transportation networks, electric utilities, resources management, water supply systems, waste management, and crime detection [49]. As shown in figure II.6.

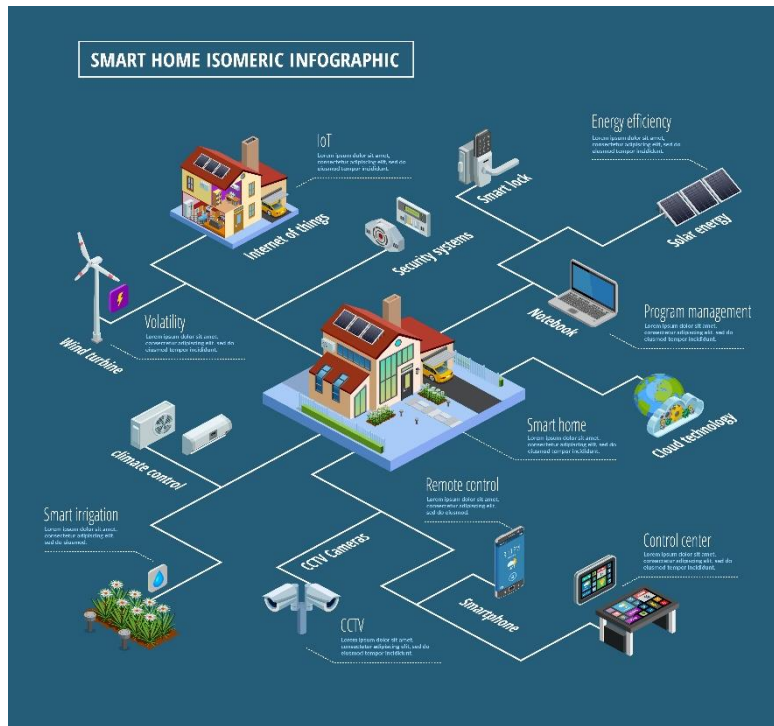


Figure II. 6: IoT for Smart city. [50]

II.2.6.3 Manufacturing Industry:

IoT ensures a more secure workplace. Through the use of wearable devices, IoT permits the assessment of the health condition of workers as well as the overseeing of hazardous activities that can end up in injuries. Staying updated regarding the inventory as well as the manufacturing tasks becomes quite a hassle, not to mention often not beneficial. IoT prompts mass customization by emerging as a source of real-time data that is needed for tasks like routing or shop floor scheduling. [51]



Figure II. 7: IoT for Industry. [52]

II.2.6.4 Health Care:

Internet of Things (IoT)-enabled devices have made remote monitoring in the healthcare sector possible, unleashing the potential to keep patients safe and healthy, and empowering physicians to deliver superlative care. It has also increased patient engagement and satisfaction as interactions with doctors have become easier and more efficient. Furthermore, remote monitoring of patient's health helps in reducing the length of hospital stay and prevents re-admissions. IoT also has a major impact on reducing healthcare costs significantly and improving treatment outcomes [53]. And so much more services as shown in figure II.8.



Figure II. 8: IoT for healthcare. [54]

Chapter II: Internet of Things

II.2.6.5 Smart Farming:

Smart Farming System assist farmers in getting Live Data (Temperature, Soil Moisture) for efficient environment monitoring which will enable them to expand their general surrounded and quality of products and be able to, most importantly, detect early stages of unwanted state. This is the purpose of that why modern agriculture involves the usage of smart tools and kits, from sowing to crop harvesting and even during storage and transportation. Timely reporting using a the value of sensors that makes the entire operation not only smart but also cost effective due to its precise monitoring capabilities. Different of autonomous tractors, harvesters, robotic weeders, drones, and satellites currently complement agriculture equipment as shown in figure II.9. Sensors can be installed and start collecting data in a short time, which is then available online for further analyses nearly immediately. Sensor the application of scientific crop and site-specific agriculture, as it supports precise data collection of every site. [55]



Figure II. 9: Smart farming illustration. [56]

II.3 Conclusion:

In this chapter we introduced the concept of IoT and gave an overview about it including its definition, architectures, elements and finally some of its applications.

IoT applications in smart farming and precision farming will enable the industry to increase operational efficiency, lower costs, reduce waste, and improve the quality of their yield.

Chapter III: Hardware and Software Description

III.1 Introduction:

Plants found in greenhouse are affected by various factors, such as PH, EC, and climatic conditions (temperature, humidity... etc.). In this prototype, we've been focused on design and implementation for monitored climate conditions and to control the different devices on output. Various inputs (sensors) and output (motors) are installed and connected to internet via controller circuit (ESP32) determined as data acquisition device.

A graphical user interface (GUI) has been designed using Android studio to retrieve and display the climatic conditions by sensing data and controlling them as well.

The following sections contain information on the hardware used in the smart system design as well as the software part.

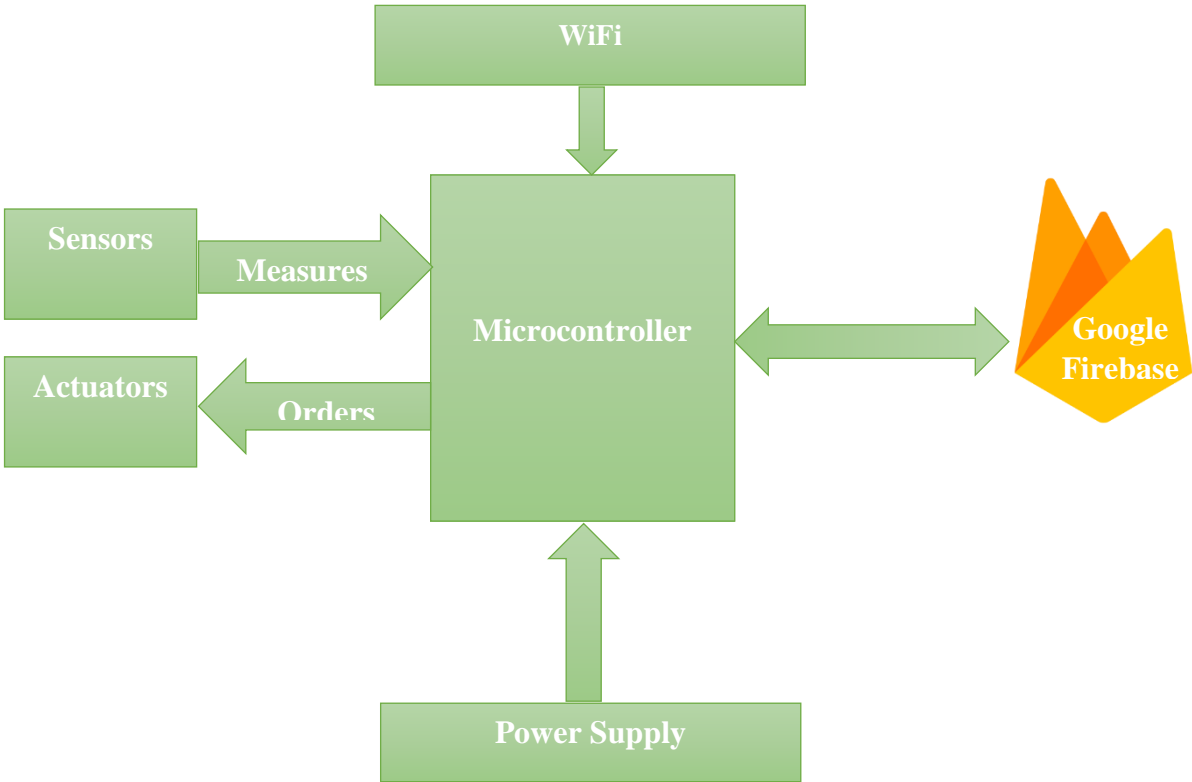


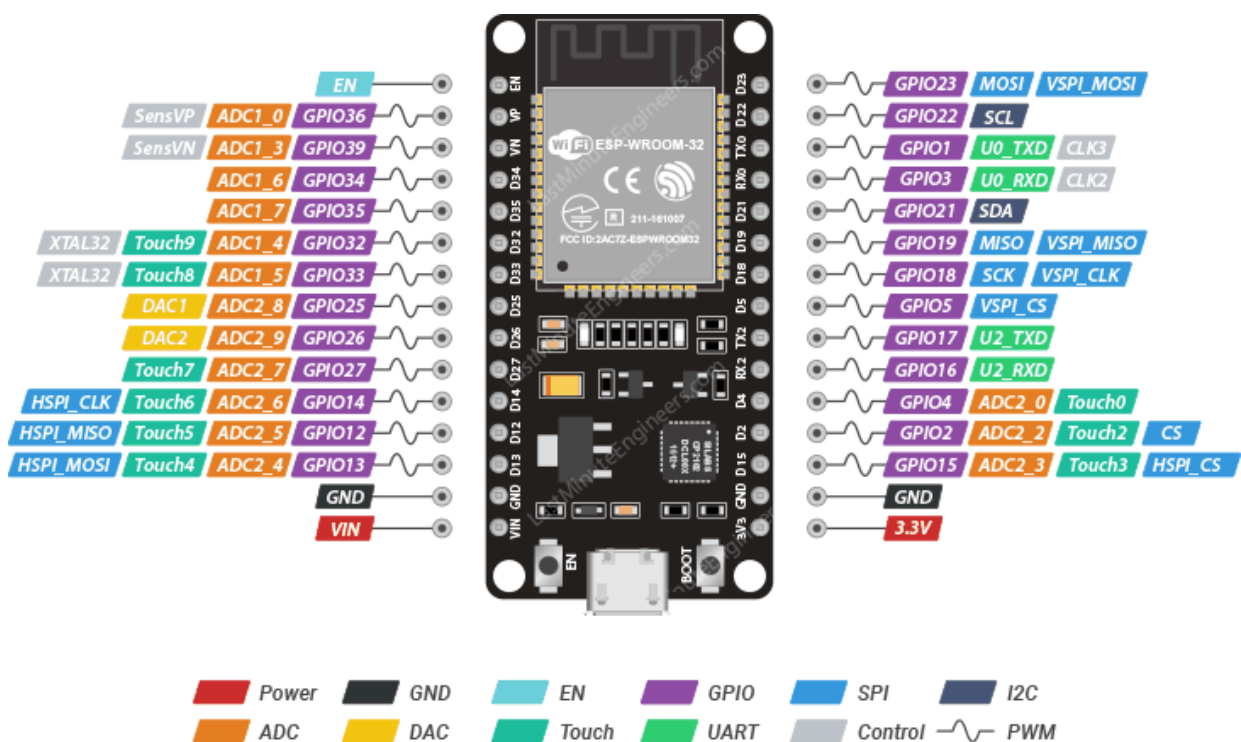
Figure III. 1: Synoptic Diagram

III.2 Hardware part:

III.2.1 Choice of microcontroller:

There is a variety of choices in the literature, we choose ESP32 because it has a lot more GPIOs than ESP8266. You won't have to juggle or multiplex the IO pins.

ESP32 is a single 2.4 GHz Wi-Fi-and-Bluetooth combo chip, powered by a 3.3V, designed with the TSMC ultra-low-power 40 nm technology and it is a highly-integrated solution for Wi-Fi-and-Bluetooth IoT applications. [57]



ESP32 Dev. Board Pinout



Figure III. 2: ESP32Dev.Board Pinout. [57]

III.2.2 Sensors:

To be able to control a smart greenhouse remotely from any device you need data. To do so you have to have sensors inside your greenhouse. What are the types of sensors used in the greenhouses?

III.2.2.1 Temperature & Humidity Sensor:

There is a low cost sensor of Temperature and humidity. It can be easily interfaced with any micro-controller, which is called DHT.

The DHT sensors consist of humidity sensing capacitor and a thermistor. They are very basic and slow.

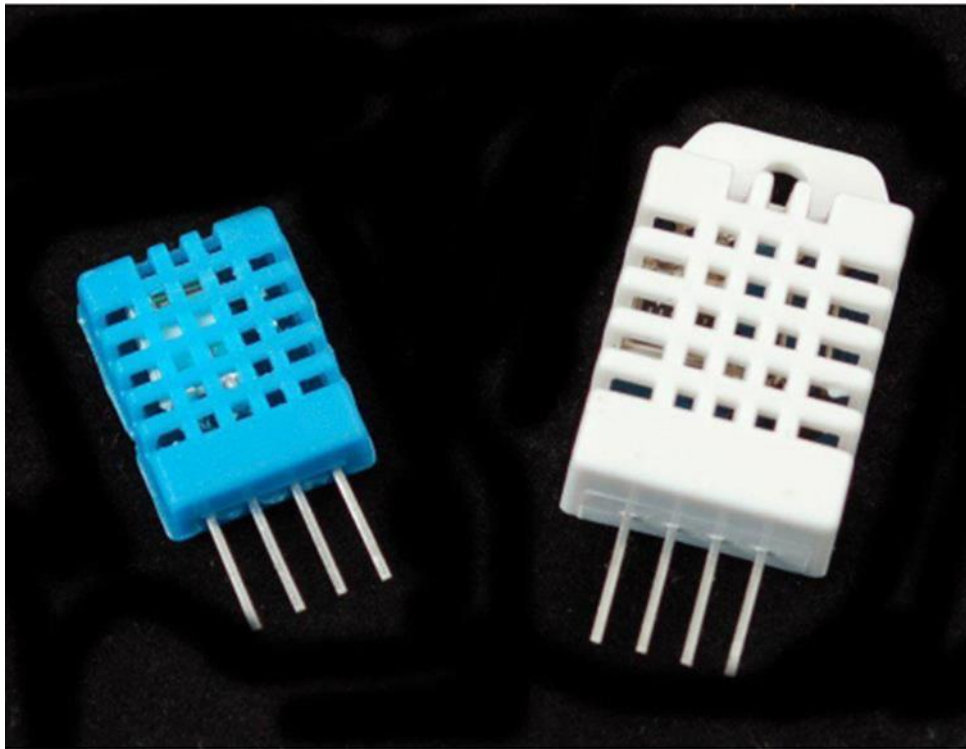


Figure III. 3: DHT11 and DHT22 sensors. [58]

We have two versions of the DHT sensor, they look a bit similar and have the same pin out, but have different characteristics. Here are the specs:

DHT11	DHT22
Ultra low cost	Low cost
3 to 5V power and I/O	3 to 5V power and I/O
2.5mA max current use during conversion (while requesting data)	2.5mA max current use during conversion (while requesting data)
Good for 20-80% humidity readings with 5% accuracy	Good for 0-100% humidity readings with 2- 5% accuracy
Good for 0-50°C temperature readings $\pm 2^\circ\text{C}$ accuracy	Good for -40 to 80°C temperature readings $\pm 0.5^\circ\text{C}$ accuracy

No more than 1 Hz sampling rate (once every second)	No more than 0.5 Hz sampling rate (once every 2 seconds)
Body size 15.5mm x 12mm x 5.5mm	Body size 15.1mm x 25mm x 7.7mm
4 pins with 0.1" spacing	4 pins with 0.1" spacing

Table III. 1: DHT11 and DHT22 sensors data sheet.

As you can see, the DHT22 is a little more accurate and good over a slightly larger range. That's why we choose it in our greenhouse for sensing the Temperature and Humidity. [58]

III.2.2.2 Light sensor:

LDR (Light Dependent Resistor) sensor:

This module is sensitive to the light, usually used for detecting the ambient brightness and light intensity. This sensor also is known as the Photoresistor sensor. This sensor module comes with 4 wires. Where the “DO” pin is a digital output pin that can be directly connected with the microcontroller, which goes high in the absence of light and it becomes low in the presence of light. And the “AO” pin is an analog output pin that can be connected with AD module that allows it to get more precise light intensity value. [59-60]

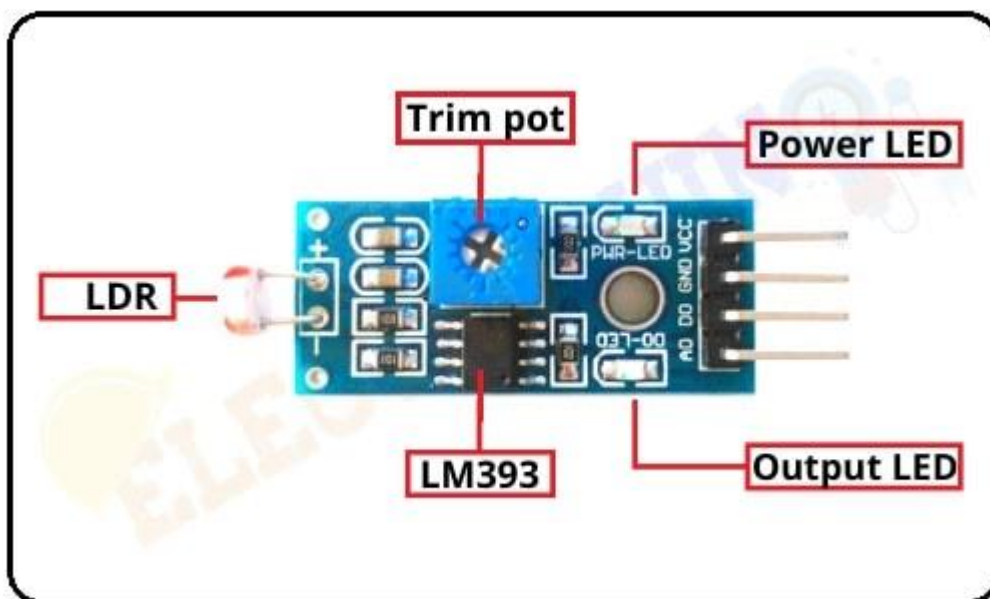


Figure III. 4: LDR sensor. [59]

Power supply	Positive: VCC=3.3-5V.
	Negative: GND.
DO	TTL switching signal output.
AO	Analog output.
Comparator chip	LM393.
Size(L*W)	Approx. (3.2cm x 1.4cm).
Sensitivity	Adjustable.
Indicator LED	Output and power LED indicator.
Module Pins	4 pins.

Table III. 2: LDR data sheet.

III.2.2.3 TDS (Total Dissolved Solids) sensor:

TDS indicates how many milligrams of soluble solids are dissolved in one liter of water. In general, the higher the TDS value, the more soluble solids are dissolved in water, and the less clean the water is. The price is affordable, and it is easy to use. This product supports 3.3-5.5V wide voltage input, and 0 - 2.3V analog voltage output, which makes it compatible with 5V or 3.3V control systems on boards. [61][62]

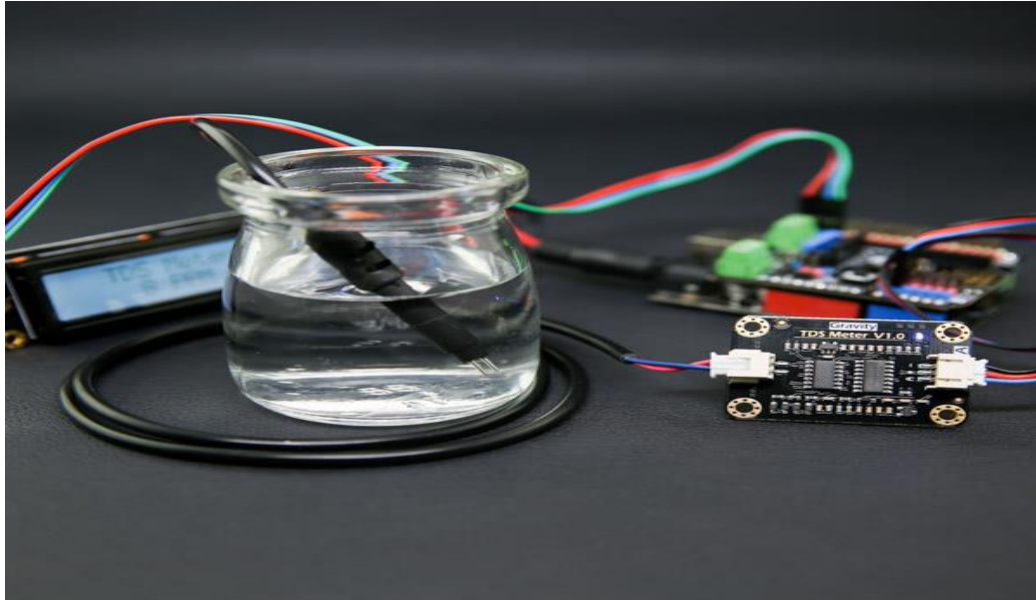


Figure III. 5: TDS sensor.

Product name	Gravity Analog TDS Sensor
Working voltage	3.3-5.5V
Working current	3-6Ma
TDS range	0-1000ppm
TDS accuracy	±10%F.S(25°C)
Board size	42*32mm
Output	analog 0-2.3V output

Table III. 3: TDS data sheet.

III.2.2.4 DS18B20 Arduino Temperature sensor:

This is a waterproofed version of the DS18B20 Arduino Temperature sensor. Handy for when you need to measure something far away, or in wet conditions. While the sensor is good up to 125°C the cable is jacketed in PVC so we suggest keeping it under 100°C. Because they are digital, you don't get any signal degradation even over long distances! Usable with 3.0-5.5V systems. [63]

This feature is useful for our project.



Figure III. 6: DS18B20 Arduino Temperature sensor. [63]

±0.5°C Accuracy from	-10°C to +85°C
Usable temperature range	-55 to 125°C (-67°F to +257°F)
9 to 12 bit	selectable resolution
Query time	is less than 750ms
3 wires interface	*Red wire – VCC *Black wire - GND *Yellow wire – DATA
Uses 1-Wire interface	requires only one digital pin for communication
Multiple sensors	can share one pin
Cable Diameter	4mm
Length	90cm

Table III. 4: DS18B20 Arduino Temperature sensor data sheet.

III.2.2.5 PH sensor:

If you worked with PH metering before you will know that PH values range from 0-14. Where PH 0 Will be very acidic, PH 7 will be neutral and PH 14 very alkaline. For hydroponic systems around 6 (for optimum plant nutrition take-up). This board has the ability to supply a voltage output to the analogue board that will represent a PH value just like any other sensor that will connect to an analog pin. Ideally, we want a PH 0 represent 0V and a PH of 14 to represent 5V. [64][65]



Figure III. 7: PH sensor. [65]

Heating voltage	$5 \pm 0.2V$ (AC.DC)
Working current	5-10mA
Detectable concentration range	PH0-14
Detection Temperature range	0-80 °C
Response time	$\leq 5s$
Component Power	$\leq 0.5W$
Settling Time	$\leq 60s$
Humidity	95% RH (nominal humidity 65% RH)
Working temperature	-10-50 °C (nominal temperature 20°C)
Module Size	42mm *32mm *20mm

Table III. 5: PH data sheet.

III.2.3 Actuators:

III.2.3.1 water pump:

We chose to pump water a 12V Mini DC Brushless Pump because it is suitable for our greenhouse. It has a small size, high efficiency, low consumption, low noise, and most importantly without pollution. [66]

Size	38*30*24mm
Working voltage	DC 12V
Power consumption	3.6W
Working current	450mA
waterproof level	IP68
Highest lift	350cm
Maximum flow	350L/H
Working life	>20000 hours
Working temperature	0- 75°C

Table III. 6: Water pump data sheet.



Figure III. 8: Water Pump. [66]

III.2.3.2 Air pump:

To control the humidity in the greenhouse we must use air pump.

Name	RS-028A
Frequency	50Hz
AC	220-240V
Puissance	6W
Q max	450L/H
H max	0.9m

Table III. 7: Air pump data sheet.



Figure III. 9: Air Pump. [72]

III.2.3.3 Peristaltic pump:

We use this type of actuator in our greenhouse to control the value of PH and EC in water. It is made of high quality material, durable and long service life. The flow direction of the peristaltic pump depends on the positive and negative wiring of the power supply, and the terminal can be arbitrarily selected. Also, it is easy to install and remove. [67]

Model	G528
Power Supply	DC 12V
Working Temperature	0-40°C
Flowing Range	0-23ml/min 0-65ml/min 0-150ml/min
Speed Range	0.1-60rpm
Curent	250-300mA
Motor Diameter	Approx. 32mm / 1.3in
Height	Approx. 23mm / 0.9in

Table III. 8: Peristaltic pump data sheet.



Figure III. 10: Peristaltic Pump. [67]

III.2.3.4 Fan:

This fan is used to reduce the temperature of the greenhouse if it is higher than the values at which the plant grows in a healthy way.

Working temperature	- 10 ~ + 70°C
Storage temperature	- 40 ~ + 75°C
Operating humidity	5~ 90% RH
Storage humidity	5~ 95% RH
Fan size	30 x 30 x 10 mm (l x w x h)
Rated voltage	DC 12V
Rated current	0.1A
Wind speed	2.5M/S
Air volume	2.8CFM
Wire Length	30cm

Table III. 9: Fan data sheet. [68]



Figure III. 11: Fan. [68]

III.2.3.5 Relay:

Relay module for switching power loads. The relay contacts are designed to switch loads up to 10A and 250VAC (30VDC). The control inputs are insulated with opto-couplers to minimize the noise perceived by the control circuit while the load switching is performed.

Chapter III: Hardware and Software Description

This module is ideal for switching AC current loads connected to the mains. This module is able to control various high current equipment for a long time and it is compatible with Arduino as well as a variety of microcontrollers. [69-70]

Relay Operation Voltage	250VAC/30VDC
Coil Voltage (Relay)	5V
Relay Operation Current	10A
Relay Activation Current	15mA~20mA
Channels	Relay Module 4 Channels
Dimensions	75x55x16.8 mm
Indicator led	For each channel

Table III. 10: Relay data sheet.



Figure III. 12: Relay. [70]

III.2.3.6 LCD:

LCD is one kind of electronic display module used in an extensive range of applications. It is the abbreviation of the English term "Liquid Crystal Display". The LCD display is in particular a visual interface between a system and a user. Its role is to transmit useful information from a system to a user. It will therefore display data likely to be exploited by the user of a system. [71]

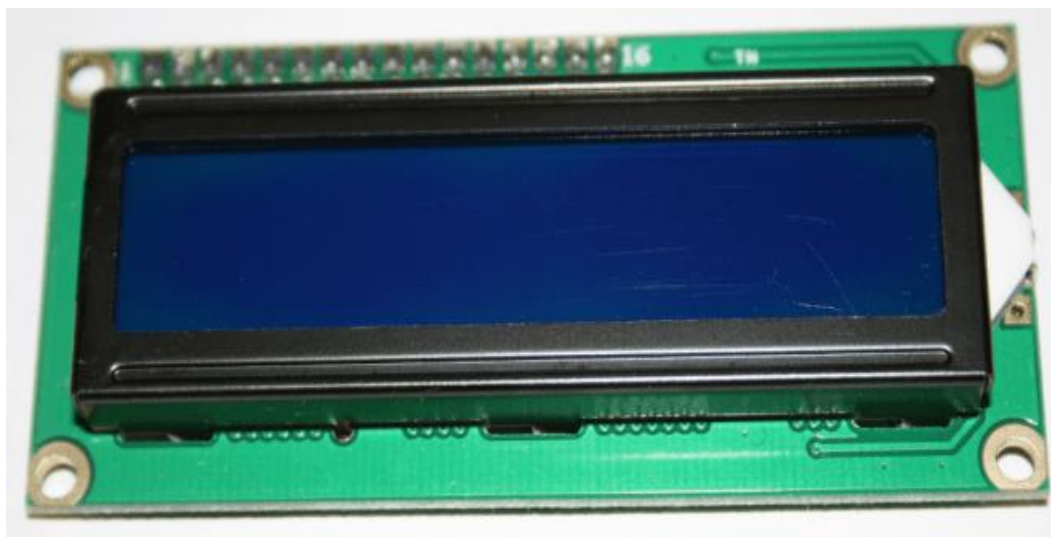


Figure III. 13: LCD. [71]

Product Name	Yellow Green STN 1602 Character LCD with Y-G Backlight
ModeINo	RYP1602A-8
LCD Type	YG screening
Led Backlight	Yellow-green
Dot Format	16 x 2
Dimension	80.0 x 36.0mm
Size	Assem: 75.0 x 31.0mm
	Character: 2.95 x 5.55mm
Operating Temperatures	-20-70°C
Storage Temperatures	-30-80°C
Nutrition	3.3V or 5V
Viewing Angle	6 hours

Table III. 11: LCD data sheet.

III.2.3.7 Heating:

This device was used to produce hot air, in order to control the temperature inside the building the greenhouse. [72]



Figure III. 14: Heating. [72]

III.2.3.8 Float:

The float is used to control the level of water in a reservoir, to ensure irrigation regular and continuous. [72]



Figure III. 15: Float. [72]

III.2.3.9 Lighting:

This colored light source was used to improve the plant growth, because each plant needs light of a certain color. [72]



Figure III. 16: Light. [72]

III.2.3.10 ESP32 Camera:

ESP32-CAM is a low-cost ESP32-based development board with onboard camera, small in size. It is an ideal solution for IoT application.

The board integrates WiFi, traditional Bluetooth and low power BLE, with 2 high performance 32-bit LX6 CPUs. It adopts 7-stage pipeline architecture, on-chip sensor, hall sensor, temperature sensor and so on, and its main frequency adjustment ranges from 80MHz to 240MHz.

Notes:

*please be sure that the power supply for the module should be at least 5V 2A, otherwise maybe there would be water ripple appearing on the image.

*ESP32 GPIO32 pin is used to control the power of the camera, so when the camera is in working, pull GPIO32 pin low.

*Since IO pin is connected to camera XCLK, it should be left floating in using, and do not connect it to high/low level.

*The product has been equipped with default firmware before leaving the factory, and we do not provide additional one's for you to download. So, please be cautious when you choose to burn other firmware. [73]

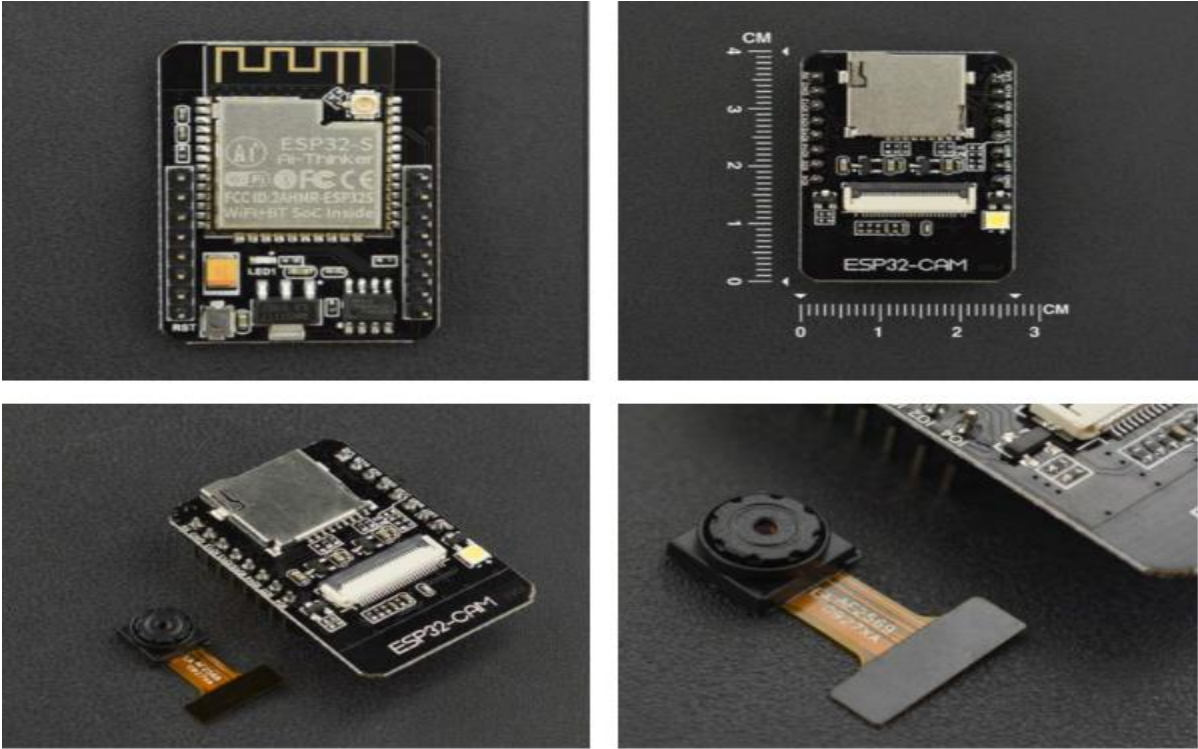


Figure III. 17: ESP32-CAM. [73]

SPI Flash	Default 32 Mbit
RAM	Built-in 520KB+external 4MPSRAM
Dimension	27*40.5*4.5 (±0.2) mm/1.06*1.59*0.18”
Bluetooth	Bluetooth 4.2 BR/EDR and BLE standards
Wi-Fi	802.11b/g/n/e/i
IO port	9
Security	WPA/WPA2/WPA2-Enterprise/WPS
Power supply range	5V
Operating temperature	-20 °C ~ 85 °C
Storage environment	-40 °C ~ 90 °C, < 90%RH
Weight	10g

Table III. 12: ESP32-CAM datasheet.

III.2.4.1 PV:

Today, there are different sources of energy to generate electricity such as solar energy. Solar energy is energy whose raw material is the sun. It is free, inexhaustible and moreover, available everywhere in the world. It collects photons from the sun and transforms them in electricity. Unfortunately, the sun does not shine 24 hours a day and solar energy depends on it to produce its energy. That’s why we must use an alternative energy source to meet our

Chapter III: Hardware and Software Description

needs during the night. In our project, we don't use solar panels just to power the greenhouse but we use it for many purposes like: lighting lamps, powering machines...etc. [74]



Figure III. 18: Solar Panel. [72]

III.2.4.2 Inverter:

The inverter is one of the key technologies. It is a device that transforms the DC power input to AC power. That's why we used Victron Energy Phoenix as an inverter for our greenhouse. [75]



Figure III. 19: Inverter. [72]

Power	180 VA
Continuos Power	175 W
Instant Power	350 W
Output Voltage	230 V AC +/-3%
Input Voltage	10.5-15.0 DC
Yield	91%
Waveform	Full sine

Table III. 13: Inverter data sheet.

III.2.4.3 Voltage Regulator:

Voltage regulator is a dispositive that maintains a constant output voltage irrespective of the changes in input supply or load conditions. That's what led us to use it in our greenhouse. It also displays protective functions such as overvoltage protection, short circuit protection, thermal shutdown, current limiting etc. [76]



Figure III. 20: Regulator. [72]

III.2.4.4 Fuses and Circuit Breakers:

To ensure the highest security for our project, there are many devices available which provide us a total range of protection for circuits such as a fuse, circuit breakers...etc.

Fuses and Circuit Breakers both serve the same purpose the protection of electrical circuits from extreme voltages or currents. They both interrupt the flow of electricity, but in very different ways from each other.

While a fuse is made of a piece of metal that melts when the flow of current through it is high , circuit breakers on the other hand, have internal switch mechanisms which stop the

Chapter III: Hardware and Software Description

flow of current once a fault has occurred. Not like a fuse, it is used to protect low current circuits as well as high voltage circuits. [77-78]



Figure III. 21: Fuses. [72]



Figure III. 22: Fuses. [72]

III.2.4.5 Level shifter:

The TXS0108E 8 channel (8-bit) bi-directional level-shifter (logic level converter) works with both open-drain and push-pull configurations. This 8-bit translator uses two separate configurable power-supply rails. The A port tracks the VA (VCCA) pin supply voltage. The VA pin accepts any supply voltage between 1.2 V and 3.6 V. The B port tracks the VB (VCCB) pin supply voltage. The VB pin accepts any supply voltage between 1.65 V and 5.5 V. [79]

A Port Voltage Range	1.2 to 3.6 V
Maximum Data Rates	*110 Mbps (Push Pull) *1.2 Mbps (Open Drain)
B Port Voltage Range	1.65 to 5.5 V
Board Dimensions	25mm x 15mm (0.985" x 0.591") / Long x Wide
Weight	2.6g (0.092oz)

Table III. 14: TXS0108E level-shifter data sheet.

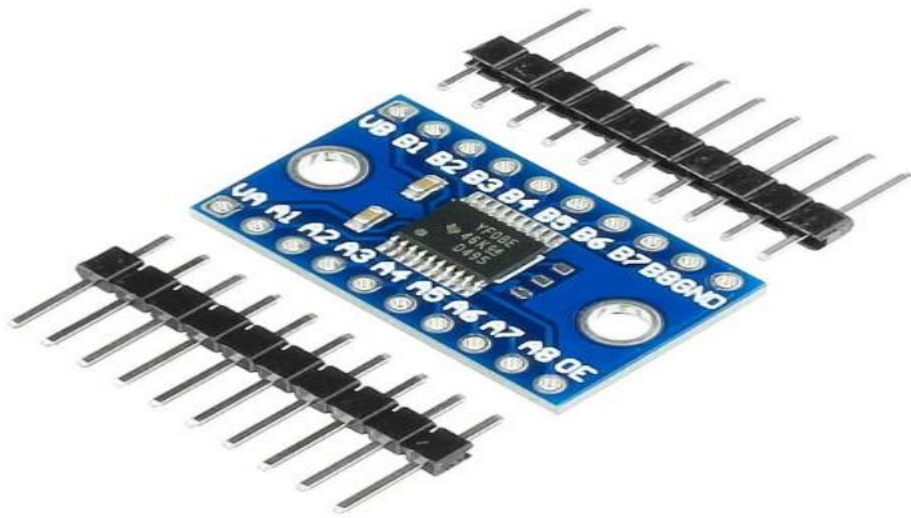


Figure III. 23: TXS0108E level-shifter. [79]

III.3 Software part:

III.3.1 Arduino IDE:

The arduino software (IDE) is an open source software, which is used to program the Arduino boards, and is an integrated development environment, developed by arduino.cc. Allow to write and upload code to arduino boards. And it consist of many libraries and a set of examples of mini projects. The IDE application is suitable for different operating systems such as Windows, Mac OS X, and Linux. It supports the programming languages C and C++.

[80-81]

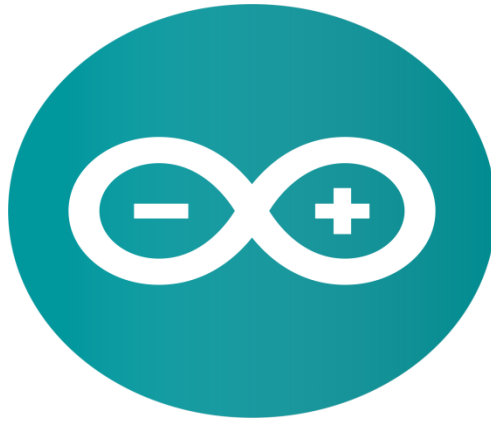


Figure III. 24: Arduino logo. [82]

Libraries used:

1. IOXhop_FirebzseESP32.h
2. WiFi.h
3. Adafruit_GFX.h
4. Adafruit_SSD1306.h
5. DallasTemperature.h
6. Wire.h
7. OneWire.h
8. SimpleTimer.h
9. ArduinoJson.h
10. DHT.h
11. LiquidCrystal_I2C.h

III.3.2 Interfacing ESP32 with Arduino IDE:

There's an add-on for the Arduino IDE that allows us to program the ESP32 using the Arduino IDE and its programming language.

To install the ESP32 board in an Arduino IDE, we execute the following instructions:

1. In the Arduino IDE, go to **File> Preferences**.
2. Enter the following into the "Additional Board Manager URLs" field:
3. https://dl.espressif.com/dl/package_esp32_index.json
4. Then, click the "OK" button.
5. Open the Boards Manager. Go to **Tools > Board > Boards Manager...**
6. Search for **ESP32** and press install button for the "**ESP32 by Espressif Systems**":

7. That's it. It should be installed after a few seconds. [83]

III.3.3 Google Firebase:

Firebase is considered as web application platform. It helps developers build high-quality apps. Data is stored in the realtime database as JSON (JavaScript Object Notation) which doesn't use query for inserting, updating, deleting or adding data to it. It is the backend of a system that is used as a database for storing data.



Figure III. 25: Google Firebase logo. [84]

Firebase is a platform that provides several services such as Realtime Database, Storage, Hosting and Authentication. In our application we worked with Realtime Database and Authentication. [85-86]

Real-time Database:

Firebase offers services like a real-time database and backend. An API is provided to the application developer which allows application data to be synchronized across clients and stored on Firebase's cloud. The client libraries are provided by the company which enables integration with Android, IOS, and JavaScript applications. Instead of typical HTTP requests, the Firebase Realtime Database uses data synchronization every time data changes, any connected device receive that update within milliseconds. Provide collaborative and immersive experiences without thinking about networking code. [86-87]

Authentication:

Chapter III: Hardware and Software Description

Firebase Auth supports social login provider like Facebook, Google GitHub and Twitter. It is a service that can authenticate users using only client-side code and it is a paid service. It also includes a user management system whereby developers can enable user authentication with email and password login stored with Firebase. [81]

III.3.4 Connecting Arduino to Firebase:

1- Adding needed libraries:

a- Firebase library:

Download firebase library from following link. [ArtronShop/IOXhop_FirebaseESP32: Get Set Push Stream value on Firebase via ESP32 \(github.com\)](https://github.com/ArtronShop/IOXhop_FirebaseESP32)

Then go back to Arduino ide and Open Sketch > include library > add.ZIP library and add ZIP library to Arduino IDE.

b- Adding the ArduinoJson library.

Open Tools > Manage Library and install ArduinoJson library.

c- The same way we add WiFi library.

2- Setup Firebase realtime database:

Go to firebase console from following link. <https://firebase.google.com/>

and create a project if you don't already have one. Once done go to Realtime Database in the left of the screen and create a Realtime Database. And use test mode as a Security rule.

3- Code explanation:

1- Go back to Arduino ide and open File > Examples >IOXhop_FirebzsESP32 master > FirebaseDemo_ESP32.

2- Get your firebase host from your realtime database and replace it in the Arduino code FIREBASE_HOST instead of 'example.firebaseio.com'

3- Then Open your Realtime Database and go to project overview > project settings > Service accounts > Database secrets and copy your Secret and paste in to your Arduino code FIREBASE_AUTH instead of 'token_or_secret'.

4- Then set your WIFI_SSID and WIFI_PASSWORD.

You can replace the loop and setup code according to your project needs and upload it to your board. [82][88]

III.3.5 Android Application:

Android is a powerful and modern OS platform, it competes with other platforms such as Apple's iOS, Microsoft's Windows Mobile, RIM OS...etc.

There are currently over 100000 apps in the Android Market, making it the second most popular development environment, behind iOS.

The applications are written for the most part in Java, and can use the hardware and the functionalities of the system via Java libraries developed by Google. As they can also be written using the Kotlin and C++ languages.

It has a very large community of developers who produce diverse and varied applications to extend the functionality of the operating system. [89]



Figure III. 26: Android Logo. [90]

III.3.6 Android Studio:

Android Studio provides a complete IDE, including an advanced code editor and app templates. It also contains tools for development, debugging, testing, and performance that make it faster and easier to develop apps. You can use Android Studio to test your apps with a large range of preconfigured emulators, or on your own mobile device. You can also build production apps and publish apps on the Google Play store. Android Studio is available for computers running Windows or Linux, and for Macs running macOS. The OpenJDK (Java Development Kit) is bundled with Android Studio. [91]



Figure III. 27: Android Studio Logo. [92]

III.3.7 Kotlin:

Kotlin is a concise language, with estimates of being less verbose than Java. Although it is a new language with the first version launched in 2016, the language is behind the big companies such as JetBrains and Google, which has joined as the recommended language for Android (because it is 100% Java compatible). Kotlin shows greater security. It also includes smart casting, higher order functions and extension functions. [93-94]



Figure III. 28: Kotlin Logo. [95]

III.3.8 XML:

XML (Extensible Markup Language) is a markup language much like HTML, but without predefined tags to use. Instead, you define your own tags designed specifically for your needs. This is a powerful way to store and transport data in a format that can be stored, searched, and shared. Most importantly, since the fundamental format of XML is standardized, if you share

or transmit XML across systems or platforms, either locally or over the internet, the recipient can still parse the data due to the standardized XML syntax. [96]

III.3.9 Building android app with android studio:

First step to do is to download Android studio. You can download it from the [Android Studio](#) page. And then follow the installation tutorials available on their official page. Once installation is done

1. Open Android Studio.
2. In the Welcome to Android Studio dialog, click Start a new Android Studio project.
3. Select Empty Activity. Click Next.
4. Give your application a name, such as My First App.
5. Make sure the Language is set to Kotlin.
6. Leave the defaults for the other fields.
7. Click Finish.

The creation of an android application is done in two parts:

- a. The part of programming that connects buttons and images with their functions and usually using java or kotlin as programming languages.
- b. The graphic part in terms of colors, shapes, tables and buttons etc.

In the 1st part we have 6 interfaces:

- ❖ Login Activity.kt
- ❖ Register Activity.kt
- ❖ Splashscreen.kt
- ❖ dashboard.kt
- ❖ lightActivity.kt
- ❖ PHActivity.kt
- ❖ ECActivity.kt
- ❖ MainActivity.kt(temp&hum)
- ❖ camActivity.kt
- ❖ aboutusActivity
- ❖ searchActivity

And each interface had a file .xml port its design according to the images, the shape and buttons to maintain a link between .xml design tools and their functions in .kt.

We also find gradle, which represents the emulator in Android Studio and it is very important, because all the libraries that we add in the file. kt must be existing in the compiler. [91]

Chapter III: Hardware and Software Description

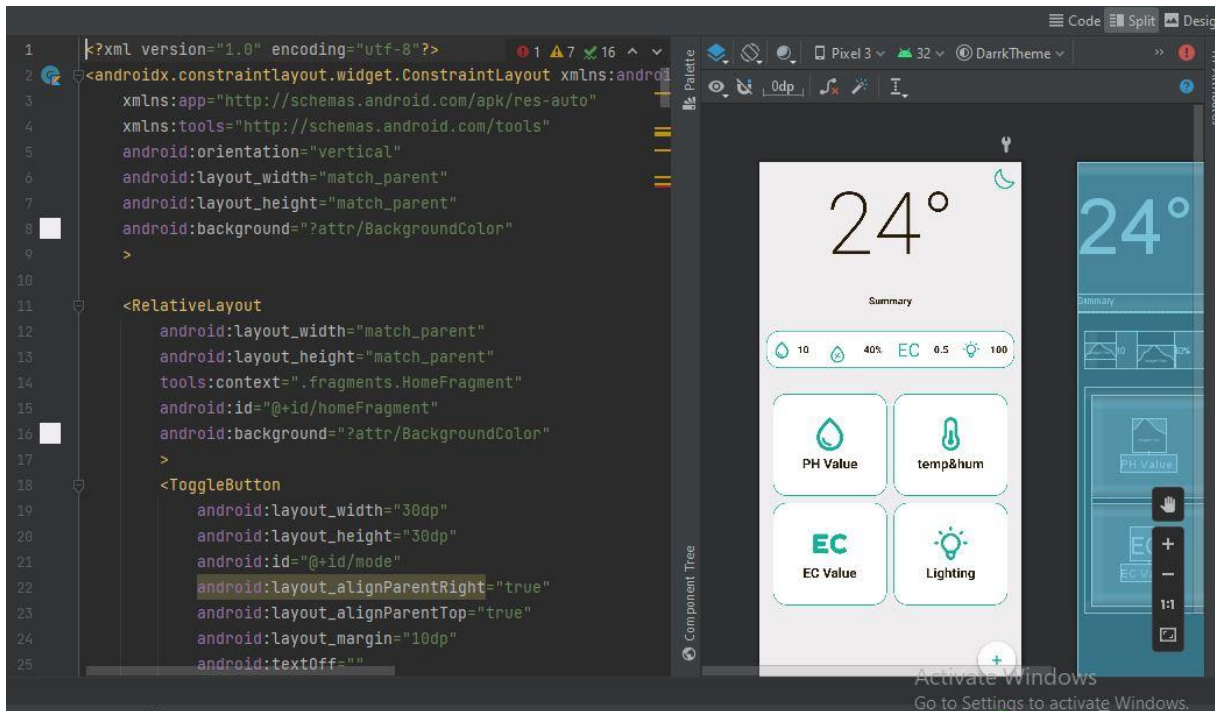


Figure III. 29: XML File.

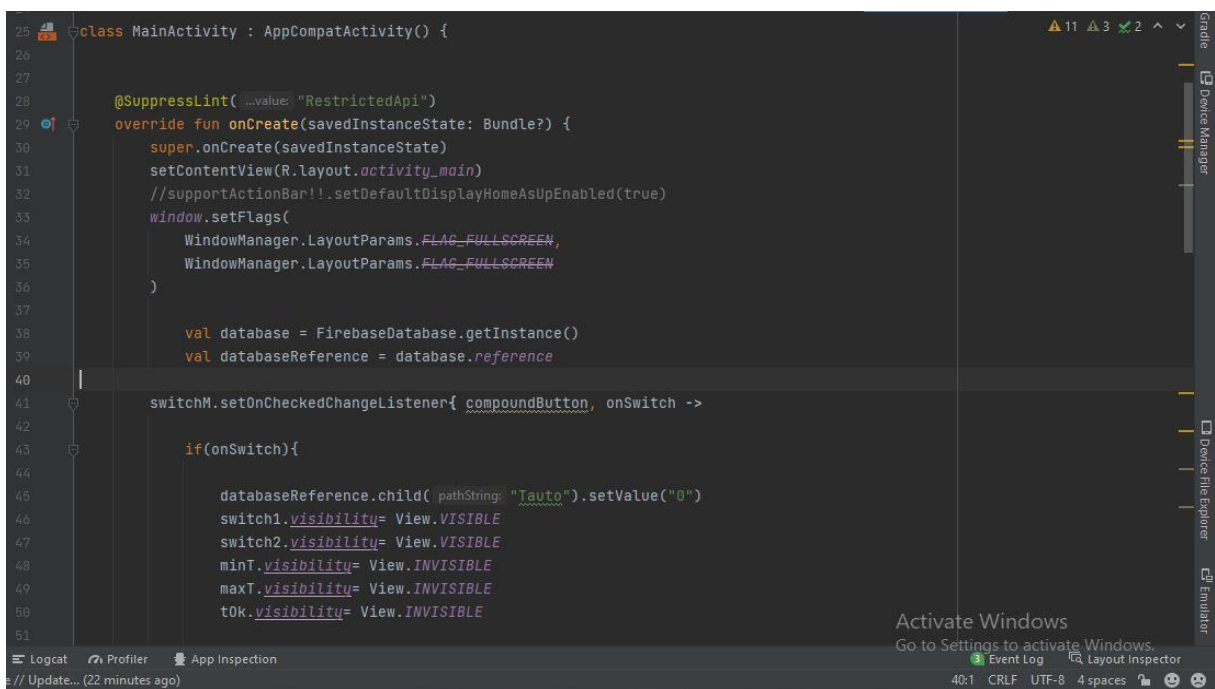
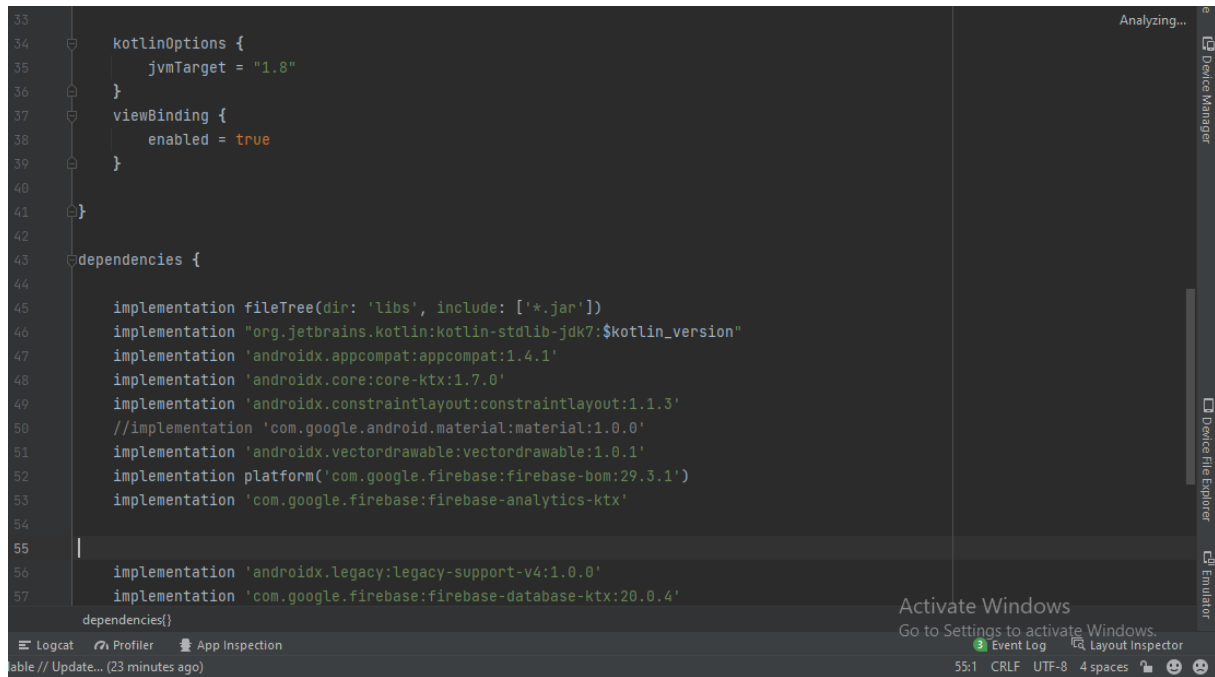


Figure III. 30: Kotlin File.



```
33
34     kotlinOptions {
35         jvmTarget = "1.8"
36     }
37     viewBinding {
38         enabled = true
39     }
40
41 }
42
43 dependencies {
44
45     implementation fileTree(dir: 'libs', include: ['*.jar'])
46     implementation "org.jetbrains.kotlin:kotlin-stdlib-jdk7:$kotlin_version"
47     implementation 'androidx.appcompat:appcompat:1.4.1'
48     implementation 'androidx.core:core-ktx:1.7.0'
49     implementation 'androidx.constraintlayout:constraintlayout:1.1.3'
50     //implementation 'com.google.android.material:material:1.0.0'
51     implementation 'androidx.vectordrawable:vectordrawable:1.0.1'
52     implementation platform('com.google.firebase:firebase-bom:29.3.1')
53     implementation 'com.google.firebase:firebase-analytics-ktx'
54
55
56     implementation 'androidx.legacy:legacy-support-v4:1.0.0'
57     implementation 'com.google.firebase:firebase-database-ktx:20.0.4'
58 }
dependencies{
dependencies{}
```

The screenshot shows an IDE window with a Gradle file. The code is as follows:

Figure III. 31: Gradle File.

III.3.10 Connecting firebase with android studio:

First make sure you have installed Google Repository version 26 or higher, using the following steps:

1. Click Tools > SDK Manager.
2. Click the SDK Tools tab.
3. Check the Google Repository checkbox, and click OK.
4. Click OK to install.
5. Click Background to complete the installation in the background, or wait for the installation to complete and click Finish.

You can now open and use the Assistant window in Android Studio by following these steps:

1. Click Tools > Firebase to open the Assistant window.
2. Click to expand one of the listed features (for example, Analytics), then click the Get Started tutorial to connect to Firebase and add the necessary code to your app. [97]

III.4 Conclusion:

At the end of this chapter, we were able to detail the needs of our project. We have started with a detailed synoptic diagram, presented the used hardware and software and we have finally explained how they work.

The purpose is to present the components of the smart greenhouse in order to get started with the implementation and realization of it. This implementation will be presented in the next chapter.

Chapter IV: Realization

Chapter IV: Realization

IV.1 Introduction:

Our project aims to remotely control a hydroponic greenhouse by an android application based on ESP32 electronic card through a Wi-Fi connection by using the Internet of Things. .

We have created an Android app that allows us to obtain data about the evolution of the "Temperature, Humidity, Light, PH value and EC electrical conductivity" parameters and to act if necessary.

In this last chapter, we presented the overall function of the circuit and of each part (temperature and humidity, PH, EC and luminosity), as well as our application and without forgetting the final results which we obtained during our tests.

IV.2 Overall function of the circuit:

The hydroponic smart greenhouse has multiple wired sensing units. Realtime values of temperature, humidity, PH, EC, and light are recorded and sent to the ESP32 controller, and sent to firebase realtime database afterwards.

An android application is also used to show realtime readings to the manager or owner of greenhouse and can be accessed from anywhere in the world using internet. Ranges like Temperature range or value of EC can also be changed using the smart phone, a manual override is also possible.

The Diagram below explains clearly the whole functionality of "Smart Greenhouse".

Chapter IV: Realization

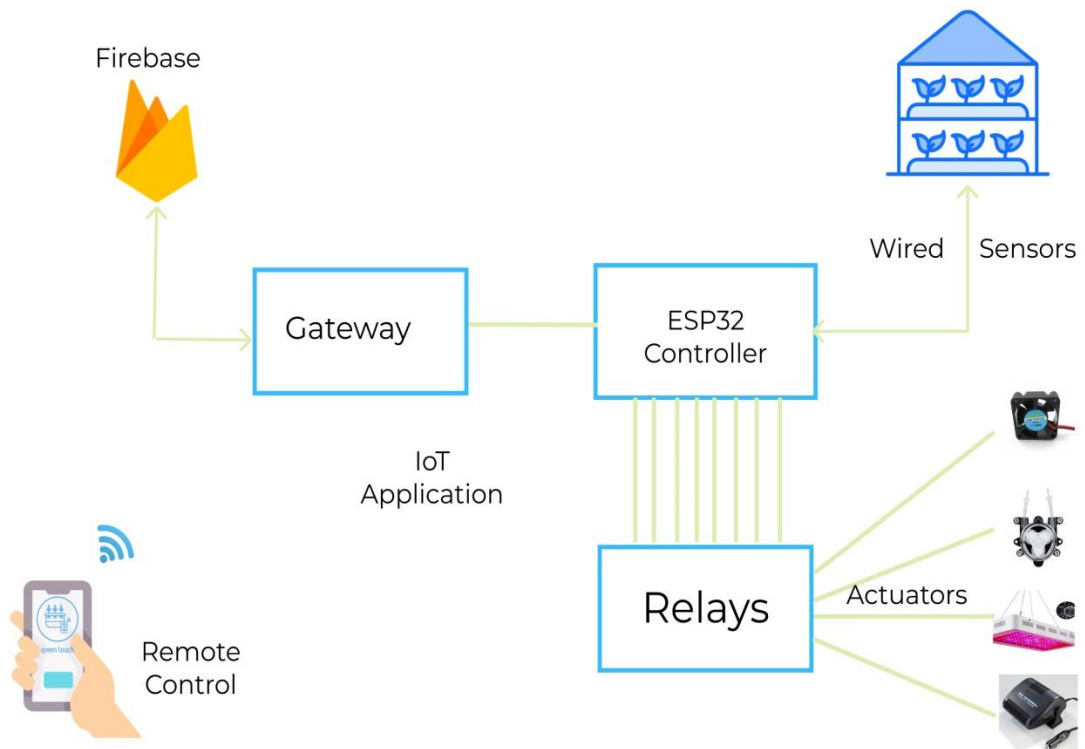


Figure IV. 1: Functionality of Greenhouse.

IV.3 Function of each part:

IV.3.1 Temperature and humidity:

As shown in diagram bellow, we get data from Firebase. First, we choose the mode (automatic/manual).

In automatic mode:

Our greenhouse has a temperature which must be between min and max for good plant growth and in this case, there is no need for ventilation or heating (Fan=off, heater=off).

If the temperature reaches the min and below, the fan and the heating will both be activated, otherwise the ventilation will be activated as soon as the temperature stabilizes.

In manual mode:

If (Fan-btn=1), ventilation is on, otherwise the fan is off.

And if (Heater-btn=1), the heating is on, if not it is off.

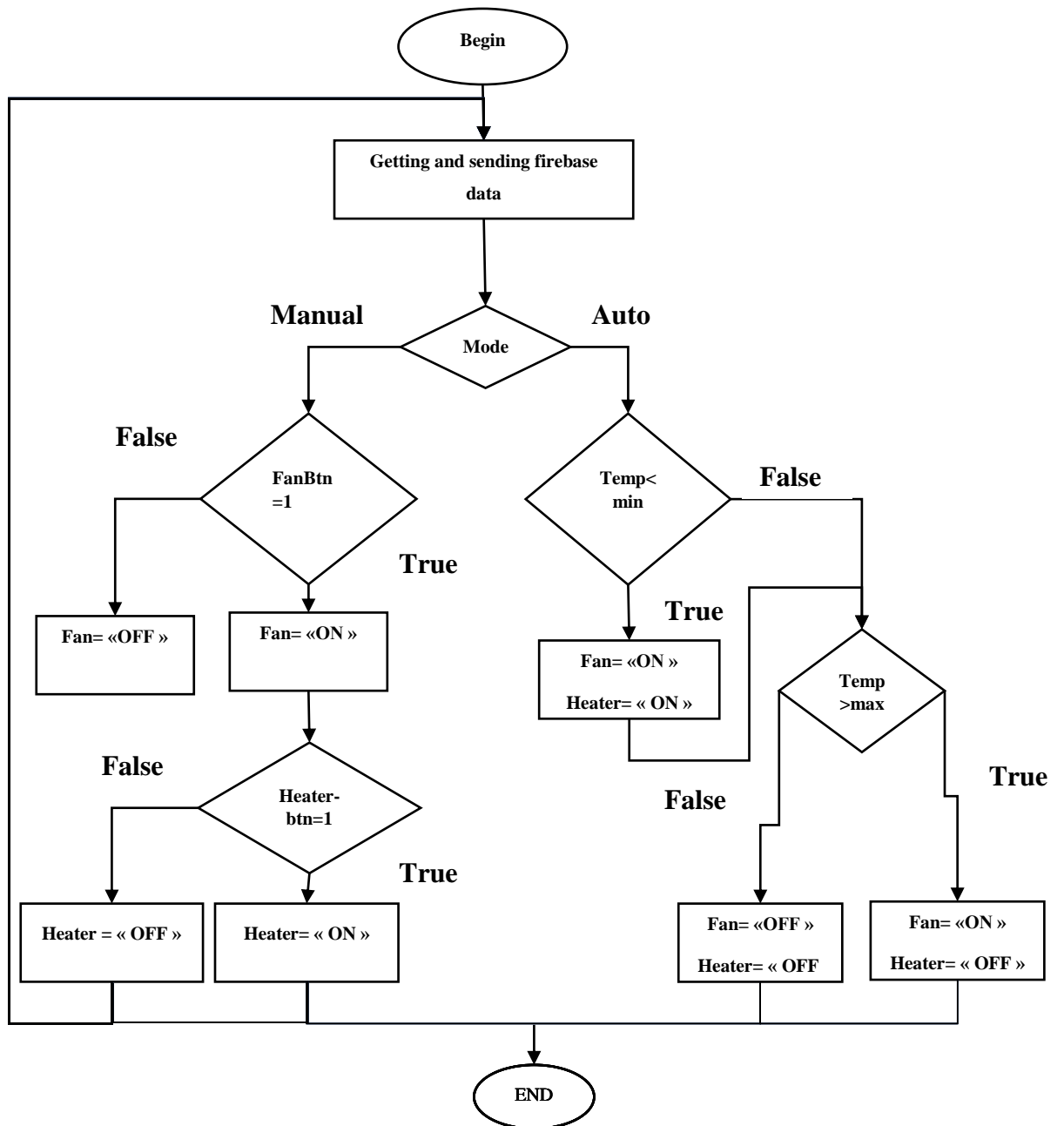


Figure IV. 2: Temperature and Humidity Flowchart.

IV.3.2 PH:

As shown in diagram bellow, we get also data from Firebase. First, we choose the mode (automatic/manual).

In automatic mode:

Chapter IV: Realization

If the PH value drops below a threshold (min), the system must trigger an alkaline pump.

If the PH reaches the max and above, the acid pump will be activated.

And if the PH value balances between min and max both alkaline and acid pumps are off.

In manual mode:

If (Alkaline-pump=1), pump is activated, otherwise the pump is off.

And if (Acid-pump=1), the pump is on, if not it is deactivated.

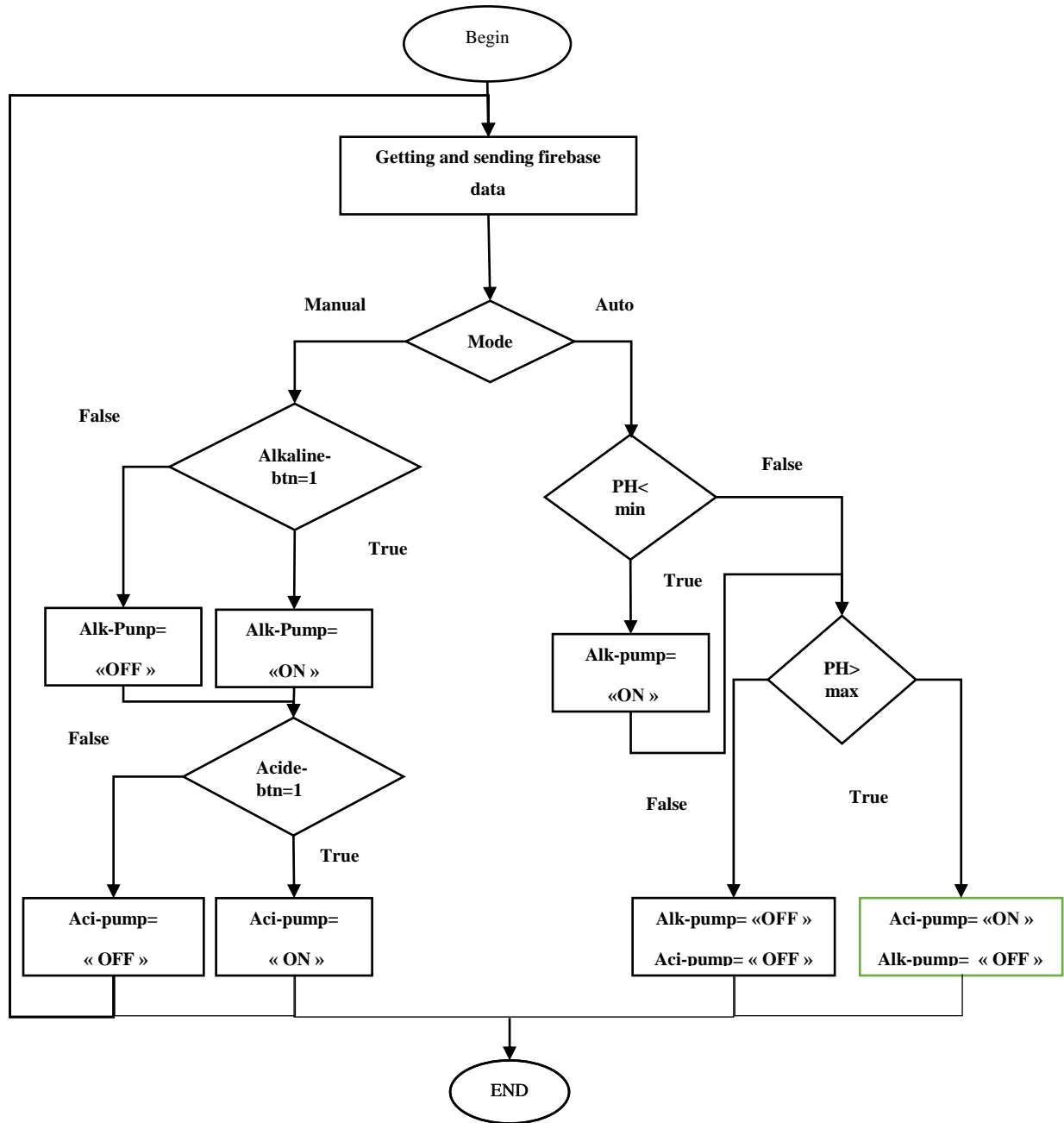


Figure IV. 3: PH Flowchart.

IV.3.3 EC:

First of all as shown in diagram bellow, we get data from Firebase and we choose the mode.

In automatic mode:

If the electrical conductivity is lower than the minimum value, the nutrient pump must be running, otherwise the pump is off.

Chapter IV: Realization

In manual mode:

If (Nutrient-btn=1), Nutrient-pump is on, if not it is off.

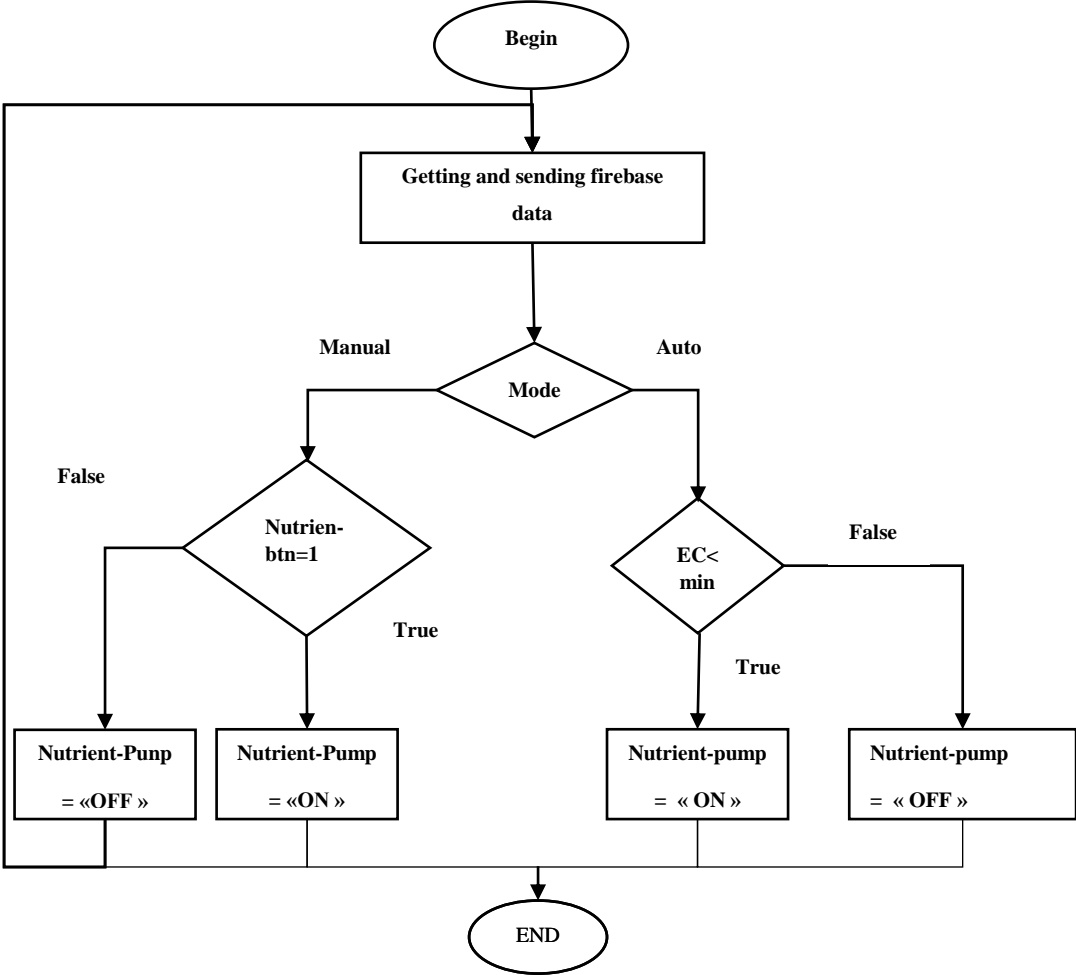


Figure IV. 4: EC Flowchart.

Chapter IV: Realization

IV.3.4 Light:

As shown in diagram bellow, in automatic mode:

In case that the brightness is below minimum, the lamp will be on, otherwise it will be off.

In manual mode:

If (lump=1), turn on the lamp, if not turn off.

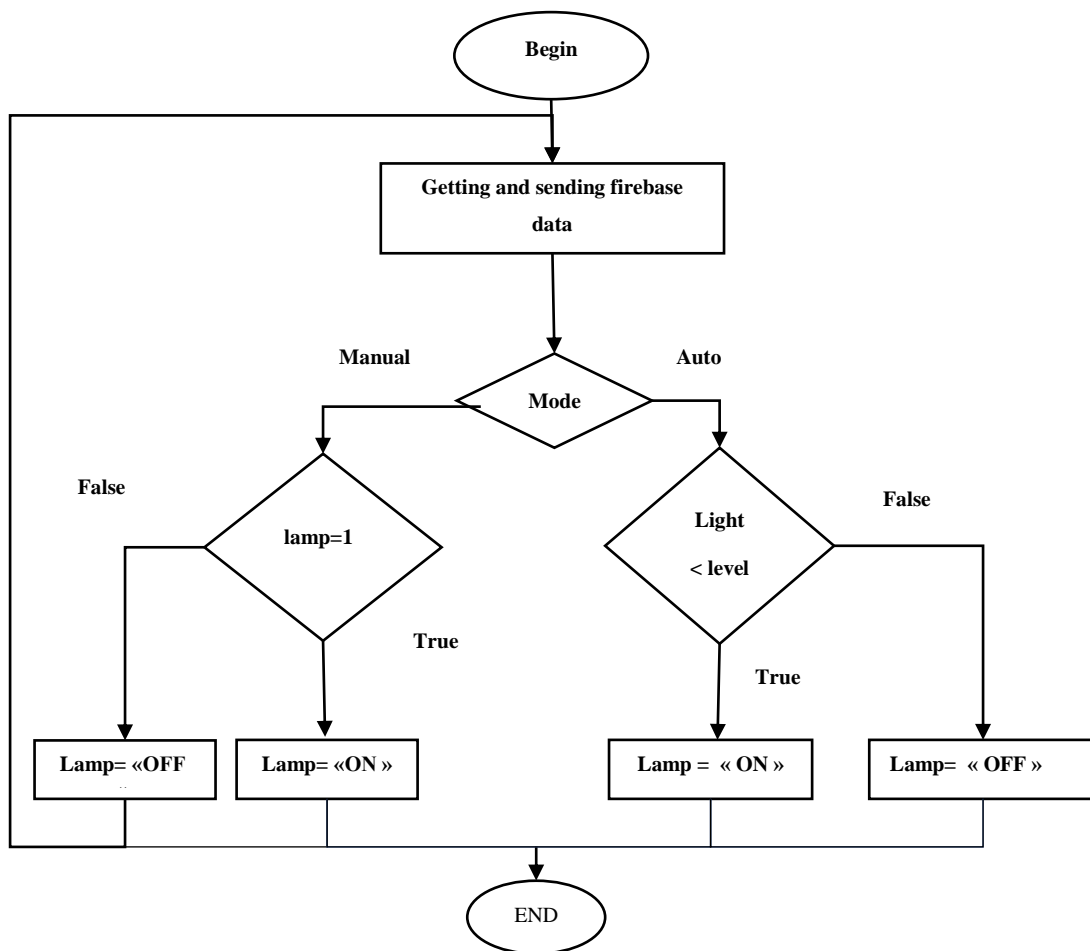


Figure IV. 5: Light Flowchart.

Chapter IV: Realization

IV.4 Circuit:

In next figure we present the global circuit used during our project:

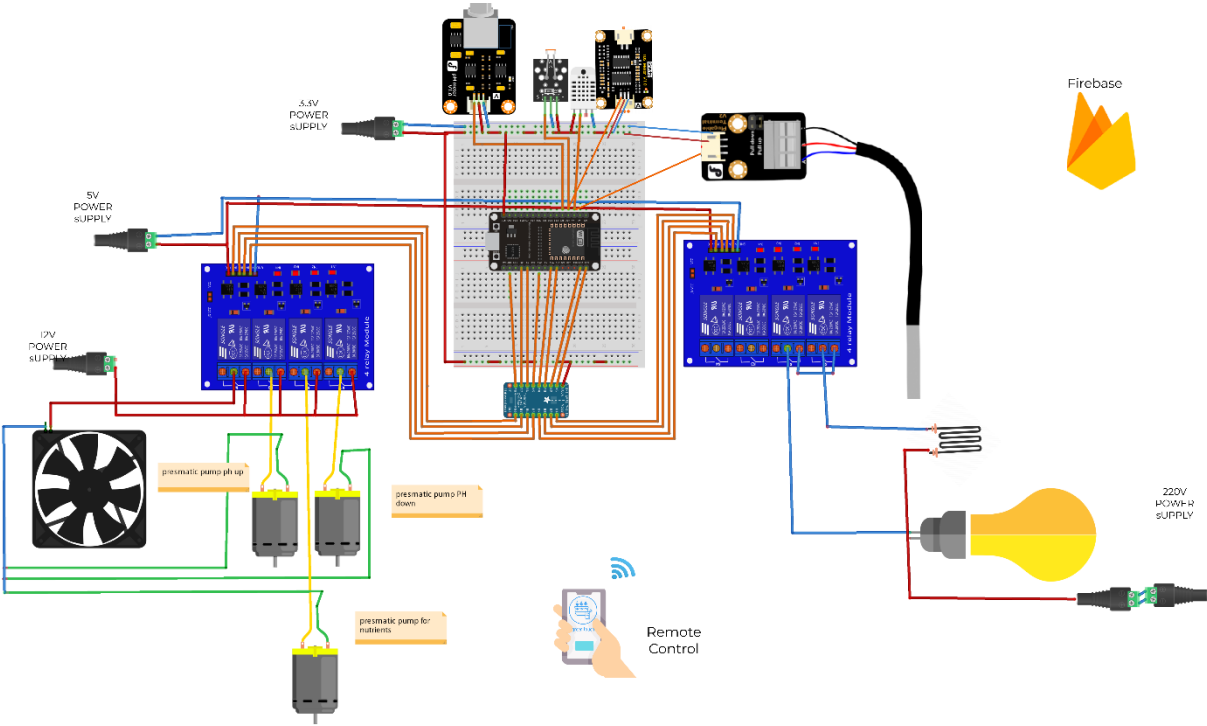


Figure IV. 6: Electrical Circuit.

IV.4 Testing of sensors:

To ensure the efficiency and proper functioning of electronic devices, we should to test them before using.

IV.4.1 Temperature sensor:

To test the temperature, we used a lighter.

Chapter IV: Realization

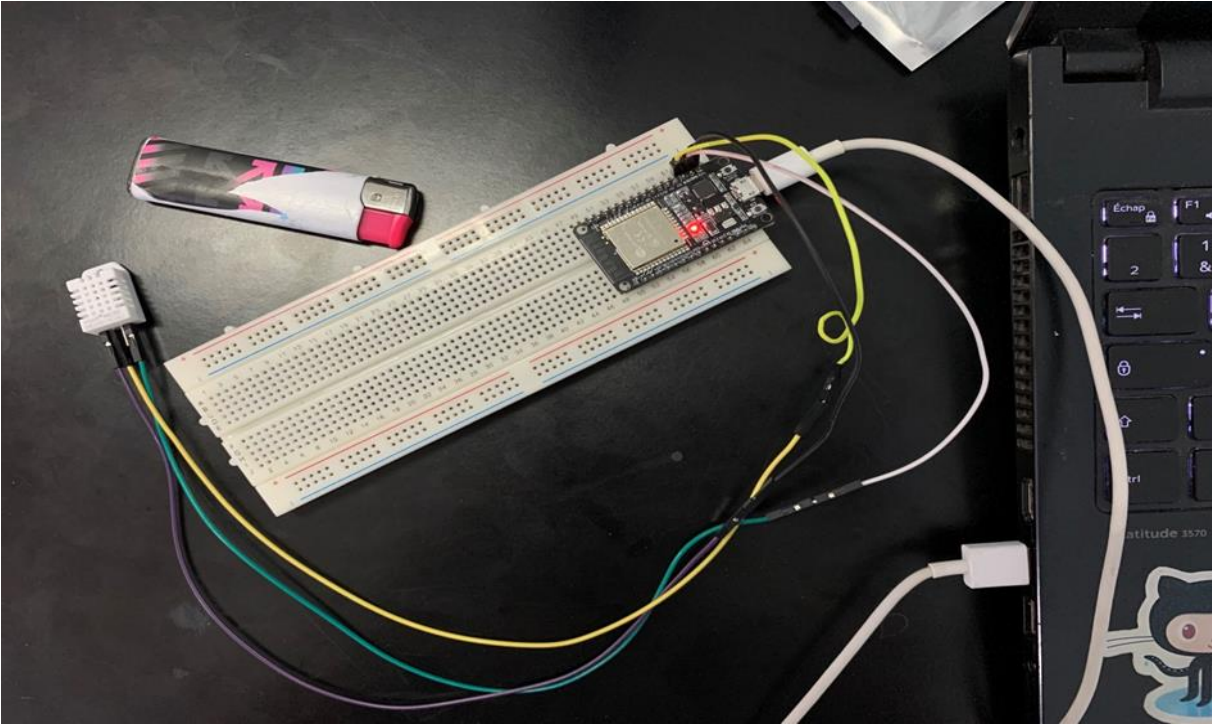


Figure IV. 7: Testing of DHT22 sensor.

```
COM5
Humidity:24%
Temperature:21°C
Humidity:24%
Temperature:22°C
Humidity:23%
Temperature:24°C
Humidity:22%
Temperature:30°C
Humidity:19%
Temperature:39°C
Humidity:19%
Temperature:45°C
Humidity:20%
```

Figure IV. 8: Result of test

As we light up the lighter we notice an evolution in the values displayed therefore. The DHT11 sensor is working properly.

Chapter IV: Realization

IV.4.2 Light sensor:

We covered the sensor to test the brightness.

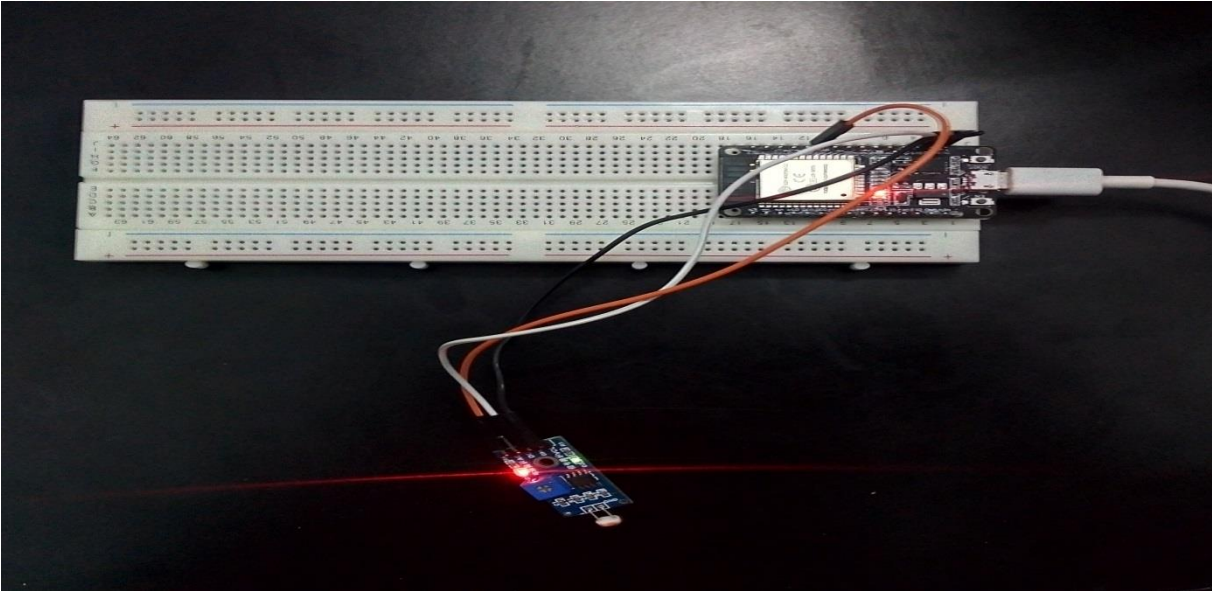


Figure IV. 9: Testing of Light sensor.

```
COM5
light:39
light: 37
light:36
light: 40
light:100
light:170
light:171
light:178
light:50
light:39
```

Figure IV. 10: Result of test

As you can see the change in the light values, which led us to say that the sensor works well.

IV.4.3 EC and PH:

The figure IV.11 mention the PH and EC sensors test

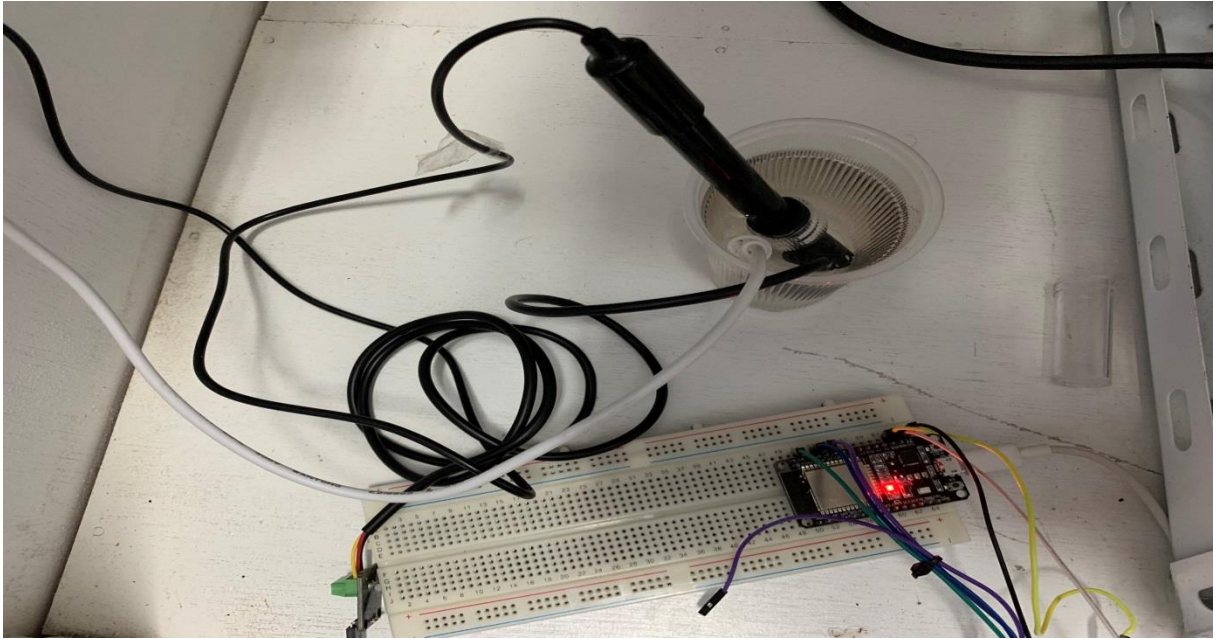


Figure IV. 11: Testing EC and PH.

IV.5 Application:

In the following, we present our work by exposing the different interfaces of the application. Starting first, with the app logo (Figure IV.12):



Figure IV. 12: GreenTouch Logo.

IV.5.1. App interfaces:

- * **SplashScreen:** This interface is the very first window displayed by the application, containing the application logo with a loading animation lasting 2 seconds. (Figure IV.13)

- * **Getting Started:** it's the interface displayed after splashScreen where u can choose either register or log in as shown in figure IV.14.

- * **Register:** To create an account to those who are not created an account in the application. By inserting their information as shown in Figure IV.15.

Chapter IV: Realization

* **Login:** For members who are already registered in the application. Where they insert their email and password as shown in Figure IV.16.



Figure IV. 13: SplashScreen.

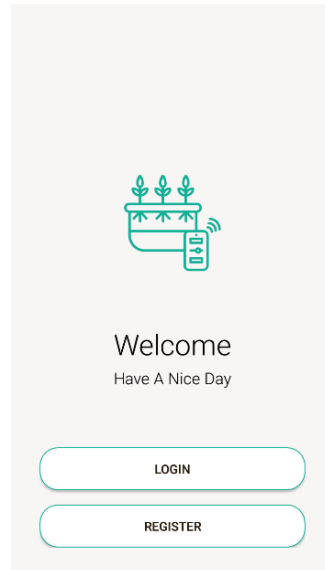


Figure IV. 14: Getting Started Interface.

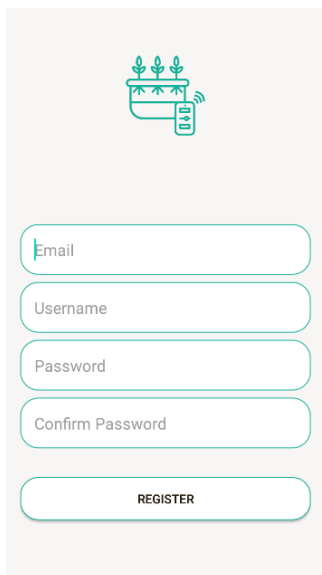


Figure IV. 15: Register Interface.

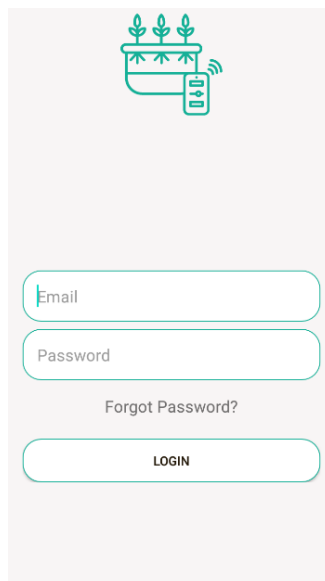


Figure IV. 16: Login Interface

*Dashboard:

Here is the graphical interface corresponding to the Home interface of the green touch app:

Chapter IV: Realization

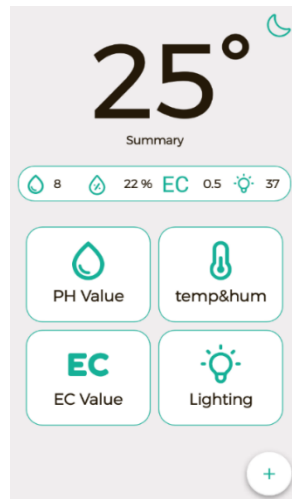


Figure IV. 17: Home Interface.

The graphical interface shows the different services provided by the application as well as a summary of the greenhouse data. The user can consult the temperature and humidity, The PH, EC, can check the Light level by clicking on its icon button to explore other features.

***parameters:**

The coming figures shows the interfaces related to each parameter.

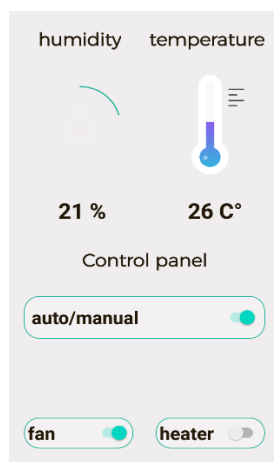


Figure IV. 18: Humidity and Temperature interface.

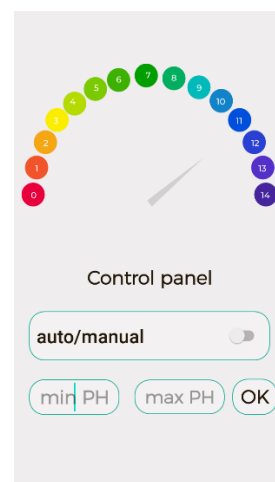


Figure IV. 19: PH interface.

Chapter IV: Realization

The Temperature interface shows the house temperature, its value is returned from the realtime database child 'Temperature', you can either choose automatic mode when you will be able to set minimum and maximum value of temperature or manual when you can turn on/off the fan or the heater according to your will. The state of switch defines the order, when it's off it sets the value of the corresponding child in realtime database to "0" otherwise "1". The same principle can be applied for all cases of Parameters.

Chapter IV: Realization

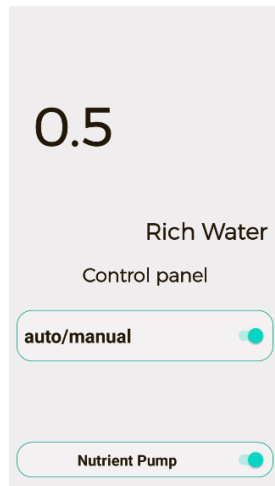


Figure IV. 20: EC interface.

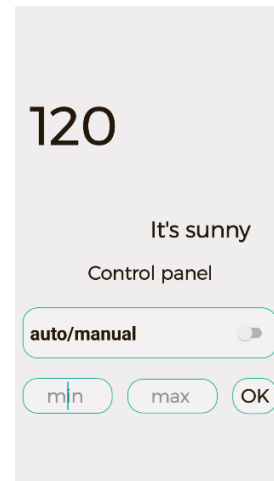


Figure IV. 21: Light interface.

By clicking on plus (+) button in the home interface as shown in the figure bellow you can explore other services such as:

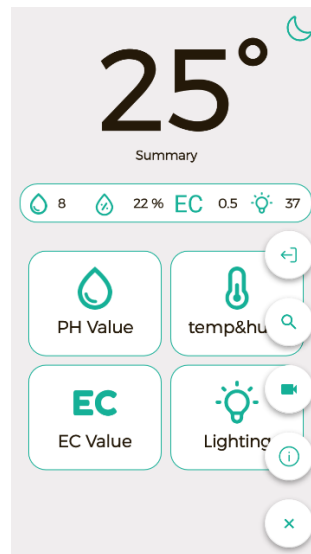


Figure IV. 22: App services

***Live Streaming:**

In order to get more information about your plant growth we added a live streaming interface as shown in figure IV.23. By clicking on the camera icon you can watch what's going inside of the greenhouse.

Chapter IV: Realization



Figure IV. 23: streaming interface.

***Search:**

Where you can enter the plant you wish to grow inside your greenhouse and get its main growth conditions and needs as shown below.

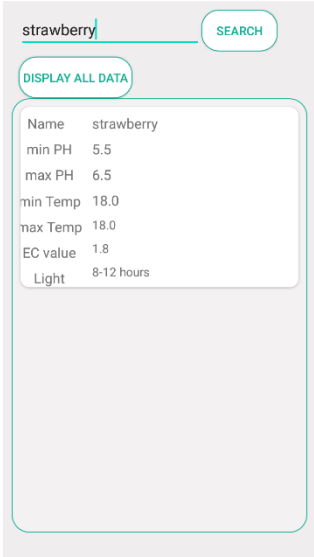


Figure IV. 24: Searching interface.

As you can logout of the app using logout icon button and to learn more about the app click the (!) button.

The button in the upper right is for switching modes (dark/light). The next figure shows the dark mode.

Chapter IV: Realization

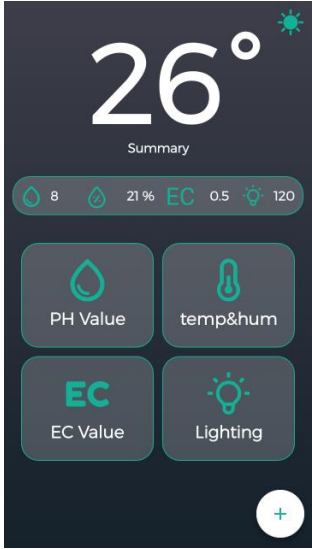


Figure IV. 25: Dark mode

IV.6 Conclusion:

This chapter gives an overview of our experimental study, we started with a presentation of how our system works, then with a description of our android application and then we tested and analyzed our results.

After our tests, we are satisfied.

General Conclusion

General Conclusion

General Conclusion:

Throughout our end-of-study project entitled: "Intelligent Management of a Hydroponic Greenhouse Powered by a PV System Using Internet of Things", we have been able to apply the knowledge and skills acquired during our university studies.

We have really enjoyed working on this project, because this project theme is interesting, and the project's outcome can be of great benefit to our daily life and to the society. The technology developed in this project can help modernize the farming and the agricultural sectors to achieve food security and improve the economy of the country.

Through this project, we have developed an Android App under the Android Studio environment which allows us to monitor and control the greenhouse parameters remotely via internet by using ESP32 Wi-Fi module.

This project has been a stimulating and motivating experience for the rest of our studies. We believe that we have glimpsed in this project a part of our future research and development activities.

Despite the huge time and efforts spent to achieve this modest work, we believe that our contribution is only the beginning of a long journey. The project could be developed further to enhance the proposed technology, including:

- ◆ Use of high quality sensors.
- ◆ Reinforce the data security part.
- ◆ Realization of a large-scale hydroponic system.
- ◆ Application improvement using artificial intelligence algorithms.

المخلص:

مع تقنيات التحكم في الدفيئة المائية، تصبح الحياة أسهل وأبسط، عندما تراقب الدفيئة الخاصة بك وتتأكد من نمو نباتك بشكل صحي.

الهدف من هذه الأطروحة هو المراقبة والتحكم عن بعد في دفيئة مائية مدعومة باللوحة الشمسية عبر تطبيقات أنترنت الأشياء IOT.

يتكون نظام الدفيئة المائية الذكية من إنشاء تطبيق أندرويد للمراقبة والتحكم في الدفيئة المجهزة بأجهزة استشعار مختلفة مثل درجة الحرارة الرطوبة والضوء ومستشعرات الناقلية الكهربائية EC ودرجة الحموضة PH والمشغلات مثل المضخات و كاميرا المراقبة و وحدة التحكم بطاقة ESP32.

الكلمات المفتاحية:

الزراعة المائية، الدفيئة، أنترنت الأشياء، التحكم عن بعد، المراقبة، تطبيق أندرويد، ESP32، Firebase

Abstract:

With hydroponic greenhouse controlling technologies, life is getting easier and simpler, when keep a watchful eye over your greenhouse and make sure your plant growth is healthy.

The aim of this thesis is to monitor and remotely control a Hydroponic Greenhouse powered by PV panel with applications of IOT.

The smart hydroponic greenhouse system consists of creating an Android Application for monitoring and controlling the greenhouse equipped with various sensors like temperature and humidity, light, EC, PH sensors and actuators such as pumps, surveillance camera, and the control unit ESP32 card.

Keywords:

Hydroponic, Greenhouse, Internet of Things, Remotely control, Monitor, Android App, ESP32, Firebase.

References

References:

- [1] [<https://sdg-tracker.org/zero-hunger>], consulted juin 12,2022.
- [2] [<https://monawa3at-mks.blogspot.com/2020/02/Agriculture-in-Algeria.html>],consulted juin 12,2022.
- [3] Miguel B. Fernandes ,Bertinho A. Costa ,João M. Lemos “Hydroponic Greenhouse Crop Optimization*”, 2018 13th APCA International Conference on Automatic Control and Soft Computing (CONTROLO) June 4-6, 2018, Ponta Delgada, Azores, Portugal.
- [4] [https://link.springer.com/chapter/10.1007/978-3-030-37794-6_10], consulted juin 12,2022.
- [5] Miguel B. Fernandes ,Bertinho A. Costa ,João M. Lemos “Hydroponic Greenhouse Crop Optimization*”, 2018 13th APCA International Conference on Automatic Control and Soft Computing (CONTROLO) June 4-6, 2018, Ponta Delgada, Azores, Portugal.
- [6] Author: admin@agrifarming.in “Hydroponic Greenhouse Farming a Full Guide” <https://gardeningtips.in/hydroponic-greenhouse-farming-a-full-guide>
- [7] « Automatisation intelligente des Serres basé sur un système d’IoT en utilisant Arduino », Melle Djeghaïbel Asma et Mr Maaradji Sofiane, mémoire pour l’obtention du diplôme de Master, Génie Electrique, Automatique et Informatique Industriel.
- [8] « Réalisation d’une serre hydroponique automatique», Melle Ali Asma, mémoire pour l’obtention du diplôme de Master, Génie Electrique, Automatique et Informatique Industriel.
- [9] “Hydroponic Greenhouse Crop Optimization”, Miguel Fernandes João Bordalo, Thesis to obtain the Master ,Electrical and Computer Engineering.
- [10][<https://www.gardenbuildingsdirect.co.uk/metal-greenhouses/polycarbonate-lean-to-greenhouse/>], consulted March 13, 2022.
- [11] [<https://www.gardenbuildingsdirect.co.uk/metal-greenhouses/billyoh-harvester-walk-in-aluminium-greenhouse/25902/>], consulted March 13, 2022.
- [12] “Film Greenhouse”, uploaded by Ally Zhai.
[<https://www.pinterest.co.uk/pin/326581410485550263/>]
- [13] [<https://greenhouseemporium.com/products/lumen-forge-20ft-geodesic-dome/>], March 20, 2022.
- [14] [<https://www.pinterest.co.uk/pin/292734044573178588/>], consulted April 02, 2022.
- [15] Abby Quillen, “Growing Soilless”,MARCH 16, 2015.
[<https://www.custommade.com/blog/introduction-to-hydroponics/>]

References

- [16] [<https://www.hydrofollies.com/PBCPPlayer.asp?ID=1885133>], consulted April 02, 2022.
- [17] J.Huete, « ADVANTAGE OF GREENHOUSES », posted at 13:04h in Nouvelles. [<https://jhuete.fr/advantages-greenhouse/>]
- [18] Susan Lundman, “The Purpose of a Greenhouse”, 21 July, 2017. [<https://www.gardenguides.com/13425530-the-purpose-of-a-greenhouse.html>]
- [19]” Hydroponics A Practical Guide for the Soilless Grower Second Edition”, Jones, J. Benton, 1930
- [20] [<https://www.pinterest.com/pin/381820874657174470/>], consulted April 02, 2022.
- [21] Butler, J.D., Oebker, N.F. (2006). Hydroponics as a Hobby Growing Plants without Soil. Circular 844. Information Office, College of Agriculture, University of Illinois, Urbana, IL61801
- [22] « Réalisation d'un système basé sur Internet des Objets pour le contrôle des serres intelligentes », Zahra Dafri , mémoire de Fin d'études Master, Informatique, Systèmes Informatiques.
- [23] “Types of greenhouses”, NSW GOVERNMENT, Department of Primary Industries. [<https://www.dpi.nsw.gov.au/agriculture/horticulture/greenhouse/structures-and-technology/types>]
- [24][<https://www.nosoilsolutions.com/6-different-types-hydroponic-systems/>], April 15, 2022.
- [25] “10 Benefits of Hydroponics & Its Impact on Agriculture”, Eden Green Technology, in AUGUST 31, 2020. [<https://www.edengreen.com/blog-collection/benefits-of-hydroponics>]

REFERENCES

- [26] Allen Goldberg “AWS IoT: Developing and Deploying an Internet of Things” [<https://www.coursera.org/lecture/aws-iot-developing-and-deploying-an-internet-of-things/introduction-to-aws-iot-8nysU>], consulted March 1, 2022.
- [27] How Many IoT Devices Are There in 2021? [More than Ever!], Published by Nick G, Updated on Jun 02, 2022 [techjury.net], consulted March 1, 2022.
- [28] Bojan Jovanovic “Internet of Things statistics for 2022 - Taking Things Apart” Updated: May 13, 2022[[dataprot.net](https://www.dataprot.net)], consulted May 4, 2022.
- [29] IoT Connections Forecast: The Rise of Enterprise, Monday 16 Dec 2019[[gsma.com](https://www.gsma.com)], consulted May 4, 2022.
- [30] SAM COOK, “60+ IoT statistics and facts”, updated: March 2, 2022, consulted May 4, 2022.

References

- [31] © DP, via Pixabay, posted May the 23th, 2015. Consulted March 1, 2022.
- [32] 12 Flavors of IoT Joanna F. DeFranco, the Pennsylvania State University PUBLISHED BY THE IEEE COMPUTER SOCIETY OCTOBER 2021
- [33] Disruptive Civil Technologies Six Technologies with Potential Impacts on US Interests out to 2025 Prepared by SRI Consulting Business Intelligence under the auspices of the National Intelligence Council April 2008
- [34] K. Lova Raju, Vijayaraghavan Veeramani «IoT Technologies in Agricultural Environment: A Survey » Article in Wireless Personal Communications • April 2020
- [35] <https://www.quora.com/What-is-IoT-architecture/>. Accessed March 5, 2022
- [36] “IoT Elements, Layered Architectures and Security Issues: A Comprehensive Survey” prepared by Muhammad Burhan , Rana Asif Rehman , Bilal Khan and Byung-Seo Kim 2,*ID1Department of Computer Science, National University of Computer and Emerging Sciences, Pakistan; Department of Software and Communications Engineering, Hongik University Published: 24 August 2018.
- [37] Pallavi Sethi¹ and Smruti R. Sarangi “Internet of Things: Architectures, Protocols, and Applications” Department of Computer Science, IIT Delhi, New Delhi, India
- [38] Yinghui H., Guanyu L., 2010. Descriptive Models for Internet of Things. IEEE International Conference on Intelligent Control and Information Processing, Dalian, China, Pages: 483- 486.
- [39] Miao W., Ting L., Fei L., ling S., Hui D., 2010. Research on the architecture of Internet of things. IEEE International Conference on Advanced Computer Theory and Engineering (ICACTE), Sichuan province, China, Pages: 484-487.
- [40] Krishan Kumar Goyal, Amit Garg, Ankur Rastogi, Saurabh Singhal “A Literature Survey on Internet of Things (IoT)”, January 2018, the International Journal of Advanced Manufacturing Technology 9(6):3663-3668

References

- [41] Pallavi Sethi and Smruti R. Sarangi, "Internet of Things: Architectures, Protocols, and Applications," Journal of Electrical and Computer Engineering Volume 2017, Article ID 9324035, 25 pages
- [42] Viviane Schmidt "Impact Analysis of the Internet of Things on the Value Chain in Manufacturing Industries" Baden-Württemberg Cooperative State University Mannheim, thesis for Bachelor of Science, June 2016
- [43] Syed Arshad Ali, Manzoor Ansari, Mansaf Alam "Resource Management Techniques for Cloud-Based IoT Environment" Internet of Things (IoT) (pp.63-87), December 2019.
- [44] M. Gigli and S. Koo, "Internet of Things: Services and Applications Categorization," Adv. Internet Things, vol. 01, no. 02, pp. 27–31, 2011.
- [45] Ankit Soni, Raksha Upadhyay and Anjana Jain "Internet of Things and Wireless Physical Layer Security: A Survey" Computer Communication, Networking and Internet Security (pp.115-123), 2017.
- [46] Sudeep Srivastava, Explained: What is Internet of Things (IoT) and how it Works? , December 23, 2021 [<https://appinventiv.com/blog/what-is-internet-of-things/>], consulted April 1, 2022.
- [47] Inam Ullah Khan, Muhammad Umar Shahzad, Muhammad Abul Hassan "Internet of Things (IoT): Applications in Home Automation», International journal of science engineering and advanced technology, IJSEAT vol.5 issue 1, ISSN 2321-6905, January 2017.
- [48] Designed by rawpixel, [freepik.com], consulted May 5, 2022.
- [49] Anweer Alam "Cloud-Based IoT Applications and Their Roles in Smart Cities" Smart Cities 4(3):1196-1219, September 2021.
- [50] Designed by macrovector. [freepik.com]. consulted May 5,2022
- [51] Role of IoT in the Manufacturing Industry Ritesh Pathak Oct 10, 2020 Updated on: Jul 05, 2021 <https://www.analyticssteps.com/blogs/role-iot-manufacturing-industry>.
- [52] Democratising data: how low-cost sensors are delivering IIoT for all By Lee Hibbert, November 9th, 2018[publitek.com], consulted May 5,2022

References

[53] Aryan Chaudhary, Saikat Mukhopadhyay “IoT, the Future of Health Care”, International Journal of Research in Engineering and Science (IJRES) ISSN (Online): 2320-9364, ISSN (Print): 2320-9356 www.ijres.org Volume 9 Issue 2 | 2021 | PP. 73-77.

[54] IoT in Healthcare: Remote patient monitoring, Mar, 2019 [sensiot.net], consulted May 5, 2022

[55] Dr.C.K.Gomathy, Mr.C. Saiganesh, Mr.B.saikiran Sri Chandrasekharendra Saraswathi Viswa Mahavidyalaya , Kanchipuram, “THE SMART FARMING SYSTEM BASED ON IOT”, International Journal of Scientific Research in Engineering and Management (IJSREM) Volume: 05 Issue: 10 | Oct – 2021, ISSN: 2582-3930.

[56] Designed by vectorpouch. [freepik.com], consulted May 5, 2022.

III.5 References:

[57] ESP32 Series Datasheet Copyright © 2022 Espressif Systems (Shanghai), consulted May 05, 2022

[https://www.espressif.com/sites/default/files/documentation/esp32_datasheet_en.pdf]

[58] [DHT11, DHT22 and AM2302 Sensors, Created by lady ada, Last updated on 2021-12-13 12:38:56 PM EST, Adafruit Industries] [<https://www.mouser.com/datasheet/2/737/dht-932870.pdf>]

[59] LDR sensor module | How LDR Sensor Works, 17 June 2020

[<https://www.electroduino.com/ldr-sensor-module-how-ldr-sensor-works/>]

[60] Arduino Photoresistor Detection Photosensitive Light Sensor, Rajguru Electronics, May 08, 2022

[<https://www.makershop.de/download/arduino-photoresistor-detection-photosensitive-light-sensor.pdf>]

[61][https://wiki.dfrobot.com/Gravity__Analog_TDS_Sensor___Meter_For_Arduino_SKU__SEN024] , consulted April 01, 2022

[62] [https://create.arduino.cc/projecthub/GAURAVK5/dfrobot-tds-meter-sensor-with-arduino-and-lcd-measure-92853d?ref=user&ref_id=808487&offset=0], consulted April 01, 2022

References

[63][https://wiki.dfrobot.com/Waterproof_DS18B20_Digital_Temperature_Sensor__SKU_DFR0198_] consulted juin,11 2022.

[64]

[<https://www.amazon.com/LiquidPH014SensorElectrodeArduino/dp/B01LWLHPRR#descriptionAndDetails>], consulted April 20, 2022

[65] How to use a PH probe and sensor. [<https://cdn.awsli.com.br/969/969921/arquivos/ph-sensor-ph-4502c.pdf>], consulted April 30, 2022

[66] [<https://alexnld.com/product/12v-mini-dc-brushless-garden-fountain-pump-hydrological-cycle-submersible-water-pump/>], consulted May 01, 2022

[67] [Peristaltic Pump, Durable Mini 32x23mm Laboratory Pump for Laboratory for Bioengineering(1 * 3(Hose)) : Amazon.in: Pet Supplies], consulted May 25, 2022

[68] [5/12/24V Power Supply Cooling Fan Radiator For Laser Module Heat Sink | Alexnld.com], consulted May 25, 2022

[69] [4 Channel Relay Module With Optocouplers - Manufacture and supplies of electronics (circuitarte.com)], consulted May 25, 2022

[70] [5v 4 channel | relay module robo-vikio (roboeq.ir)], consulted April 13, 2022

[71] [<https://plaisirarduino.fr/afficheur-lcd-comment-lexploiter/>], consulted April 13, 2022

[72] « Réalisation d'une serre hydroponique automatique», Melle Ali Asma, Mémoire pour l'obtention du diplôme de Master, Génie Electrique, Automatique et Informatique Industriel, université Ibn Khaldoun Tiaret, Année 2021.

[73] ESP32 CAM Development Board, consulted May 05, 2022

[https://media.digikey.com/pdf/Data%20Sheets/DFRobot%20PDFs/DFR0602_Web.pdf]

[74] « Etude d'une alimentation à base de panneaux photovoltaïques », M.BELLAGUE Abderrahmene et M. MOUSSA Ishak, Mémoire de fin d'étude pour l'obtention du diplôme d'Ingénieur, Electrotechnique, Energie et Environnement, Année 2019/2020.

[75] [<https://www.ayarteknik.com/victron-energy-off-grid-inverter-12-180-va>], consulted May 02, 2022

[76] [<https://electronicscoach.com/voltage-regulator.html>], consulted May 02, 2022

References

- [77] [<https://blog.steinerelectric.com/2016/07/key-differences-between-fuses-and-circuit-breakers/>], consulted May 02, 2022
- [78] [<https://www.elprocus.com/what-is-a-protection-device-different-types-of-protection-devices/#:~:text=Circuit%20Protection%20Devices,Fuse,current%20through%20it%20is%20high.>], consulted May 28, 2022
- [79] [<https://www.addicore.com/TXS0108E-p/ad284.htm>], consulted Juin 13, 2022
- [80] [<https://andprof.com/tools/what-is-arduino-software-ide-and-how-use-it/>], consulted May 28, 2022
- [81] [<https://www.javatpoint.com/arduino-ide>] /], consulted May 28, 2022.
- [82] [https://www.nicepng.com/ourpic/u2w7w7o0o0t4o0q8_arduino-1-logo-png-transparent-arduino-logo-png/], consulted May 28, 2022
- [83] [<https://randomnerdtutorials.com/installing-the-esp32-board-in-arduino-ide-windows-instructions/>], consulted May 19, 2022
- [84] [<https://icons8.com/icons/set/firebase-logo>], consulted May 19, 2022
- [85] The Firebase Realtime Database, Laurence Moroney, Chapter: First Online: 11 November 2017.
[https://link.springer.com/chapter/10.1007/978-1-4842-2943-9_3]
- [86] “Application of Firebase in Android App Development-A Study”, International Journal of Computer Applications (0975 – 8887), Volume 179 – No.46, June 2018, by “Chunnu Khawas” and “Pritam Shah”, Department of Computer Applications ,Sikkim University Gangtok, India.
- [87] [<https://firebase.google.com/docs/database>], consulted May 19, 2022
- [88] [<https://create.arduino.cc/projecthub/pulasthi-Narada/connecting-esp8266-to-firebase-to-send-receive-data-4adf66>], consulted May 19, 2022
- [89] « Conception et developpement d'un service mobile dedie a la telesurveillance des personnes à risque cardiovasculaire via un smarthphone », BELARBI Zineb Hannane ,AHTATACHE Rihab, Universite Abou Bekr Belkaid Tlemcen UABT , Département de Génie Électrique et Electronique ,Master en Génie Biomedical, Année 2016/2017

References

- [90] [<https://logos-marques.com/download/>], consulted May 19, 2022
- [91] [Build Your First Android App in Kotlin | Android Developers], consulted Juin 01, 2022
- [92][https://www.pngitem.com/middle/ihTJTbT_icon-android-studio-logo-hd-png-download/], consulted Juin 01, 2022
- [93] “Are you still smelling it? A comparative study between Java and Kotlin language”, Matheus Flauzino , Júlio Veríssimo, Ricardo Terra, Elder Cirilo, Vinicius H. S. Durelli and Rafael S. Durelli, Federal University of Lavras ,Lavras-MG, Brazil and Federal University of São João del Rei São João del Rei-MG, Brazil, consulted Juin 01, 2022
- [<http://professores.dcc.ufla.br/~durelli/sites/default/files/papers/SBCARS18.pdf>]
- [94] “A COMPARATIVE STUDY: JAVA VS KOTLIN PROGRAMMING IN ANDROID APPLICATION DEVELOPMENT”, International Journal of Advanced Research in Computer Science, Volume 9, No. 3, May-June 2018, Subham Bose, Aditi Kundu, Madhuleena Mukherjee and Madhurima Banerjee , Students, BCA, The Heritage Academy, Kolkata,India,[<https://pdfs.semanticscholar.org/c0ee/43434064520cdde7222318bf6c4d2db69177.pdf>] ,consulted Juin 03, 2022
- [95] [<https://freebiesupply.com/logos/kotlin-logo/>], consulted Juin 03, 2022
- [96] [https://developer.mozilla.org/en-US/docs/Web/XML/XML_introduction], consulted Juin 06, 2022
- [97] Connect to Firebase | Android Developers, [<https://developer.android.com/studio/write/firebase>], consulted Juin 06, 2022

Annexes

```
#include <IOXhop_FirebaseESP32.h>
#include <WiFi.h>
#include <Adafruit_GFX.h>
#include <Adafruit_SSD1306.h>
#include <DallasTemperature.h>
#include <Wire.h>
#include <OneWire.h>
#include <SimpleTimer.h>
#include <ArduinoJson.h>
#include <DHT.h>
#define dhtpin 18
#define DHTTYPE DHT22
DHT dht(dhtpin,DHTTYPE);
//#include <FirebaseESP32.h>
//-----Firebase info-----
#define FIREBASE_HOST
"https://temptest-ba24a-default-
rtbd.firebaseio.com/"
#define FIREBASE_AUTH
"QG8IOp8jMPUkgmnl3RRU6G21Ov60
97QCnUOJNF"
//----- Wifi INFO-----
#define WIFI_SSID "CDC-VISIO"
//DJAWEB_E1CF D-Link Idoom
4G_EB0ED
#define WIFI_PASSWORD "12345678"
//0101196 03738605 046241940
//-----Pin initialisation--
#define lfan 22
#define lheater 23
#define Lamp 15
#define EcPump 2
String BtnTauto;
String BtnLauto;
String BtnLamp;
String BtnEc;
String BtnEauto;
String BtnPauto;
String BtnAc;
String BtnAlk;
int ldr= 39;
int level;
int a=22;
int b=30;
String Btnfan = "0";
String Btnheater = "0";
//Ph
SimpleTimer timer;
float calibration_value = -30 ;
int phval = 0;
unsigned long int avgval;
int buffer_arr[10],temp;
float ph_act;
// ec
namespace pin {
const byte tds_sensor = 35;
```


Annexes

```
const byte one_wire_bus = 13; // Dallas
DallasTemperature
Temperature Sensor}

namespace device {

float aref = 3.3;}

namespace sensor {

float ec = 0;

unsigned int tds = 0;

float waterTemp = 0;

float ecCalibration = -1.25;}

OneWire oneWire(pin::one_wire_bus);

DallasTemperature
dallasTemperature(&oneWire);

void setup() {

//Serial Begin at 9600 Baud

Serial.begin(9600);

delay(1000);

WiFi.begin
(WIFI_SSID,WIFI_PASSWORD);

while (WiFi.status()
(WiFi.status()
!=WL_CONNECTED) {

delay(500);

Serial.print(".");}

pinMode(lfan,OUTPUT);

pinMode(lheater,OUTPUT);

pinMode (Lamp,OUTPUT);

pinMode (2,OUTPUT);

pinMode (4,OUTPUT);

pinMode (5,OUTPUT);

pinMode (15,OUTPUT);

pinMode (18,OUTPUT);

pinMode (19,OUTPUT);

pinMode (22,OUTPUT);

pinMode (23,OUTPUT);

digitalWrite(4,LOW);

digitalWrite(2,LOW);

digitalWrite(5,LOW);

digitalWrite(15,LOW);

digitalWrite(22,LOW);

digitalWrite(23,LOW);

digitalWrite(18,LOW);

digitalWrite(19,LOW);

dht.begin ();

dallasTemperature.begin();

Wire.begin();

Serial.println ("");

Serial.println ("WiFi Connected!");

Serial.println(WiFi.localIP());

Firebase.begin(FIREBASE_HOST,FIREB
ASE_AUTH);

}

void loop(){

delay(2000);

// Ph + niv
```

Annexes

```
timer.run(); // Initiates SimpleTimer

for(int i=0;i<10;i++) {
  buffer_arr[i]=analogRead(34);
  delay(30);}

for(int i=0;i<9;i++){
  for(int j=i+1;j<10;j++){
    if(buffer_arr[i]>buffer_arr[j]){
      temp=buffer_arr[i];
      buffer_arr[i]=buffer_arr[j];
      buffer_arr[j]=temp;
    }
  }
}

avgval=0;
for(int i=2;i<8;i++)
  avgval+=buffer_arr[i];
float volt=(float)avgval*14/4094/6;
//ph_act = 3.70 * volt + calibration_value;
Serial.print("ph");
Serial.println(abs(volt));
delay(1000);
BtnPauto= Firebase.getString("/Pauto");
BtnAc= Firebase.getString("/acide");
BtnAlk= Firebase.getString("/alkaline");
if(BtnPauto=="1"){
  if (abs(ph_act)<4){
    digitalWrite(5,HIGH);
    digitalWrite(4,LOW);
    delay(100);
    //digitalWrite(10,HIGH);
  }
  if (abs(ph_act)>8){
    digitalWrite(4,HIGH);
    digitalWrite(5,LOW);
    delay(100);
    //digitalWrite(12,HIGH);
  }else{
    if(BtnAc == "0"){
      Serial.println("Acide (OFF)");
      digitalWrite(4,LOW);
    }else {
      Serial.println("Acide (ON)");
      digitalWrite(4,HIGH);
    }
    if(BtnAlk == "0"){
      Serial.println("Alkaline (OFF)");
      digitalWrite(5,LOW);
    } else {
      Serial.println("Alkaline (ON)");
      digitalWrite(5,HIGH); } }
  // EC
  readTdsQuick();
  Serial.println(sensor::ec);
  delay(1000);
  Serial.println(sensor::waterTemp);
```

Annexes

```
delay(1000);
BtnEauto= Firebase.getString("/Eauto");
BtnEc=
Firebase.getString("/NutrientPump");
if(BtnEauto=="1") {
  if ( sensor::ec > 0.50)
  {
    digitalWrite(2,LOW);
    Serial.println("EC pump (OFF)");
  }
  if ( sensor::ec < 0.50){
    digitalWrite(2,HIGH);
    Serial.println("EC PUMP (ON)");
  }
}else
{if(BtnEc == "0"){
  Serial.println("EC pump (OFF)");
  digitalWrite(2,LOW);
}
else {
  Serial.println("EC PUMP (ON)");
  digitalWrite(2,HIGH);
}}
Serial.print("EC :");
Serial.println(sensor::ec);
String e= String(sensor::ec);
if (isnan(sensor::ec))
  Serial.println("ERROR");
else
  Firebase.setString("EC", e);
level= analogRead(ldr);
level= map(level,0,4094,0,255);
BtnLauto= Firebase.getString("/Lauto");
BtnLamp= Firebase.getString("/Lamp");
if(BtnLauto=="1") {
  if (level>170){
    digitalWrite(15,HIGH);
    Serial.println("lamp (ON"); }
  if (level<170){
    digitalWrite(15,LOW);
    Serial.println("lamp (OFF)");
  }}else{
  if(BtnLamp == "0"){
    Serial.println("lamp (OFF)");
    digitalWrite(15,LOW);}
  else {
    Serial.println("lamp (ON");
    digitalWrite(15,HIGH);}}
  Serial.print("lum :");
  Serial.println(level);
  String l= String(level);
  if (isnan(level))
    Serial.println("ERROR");
  else
```

Annexes

```

Firebase.setString("light", l);
int Hum = dht.readHumidity();
int Temp= dht.readTemperature();
Serial.print("temp :");
Serial.println(Temp);
Serial.print("hum :");
Serial.println(Hum);
String h= String(Hum);
String t= String(Temp);
if (isnan(Hum) || isnan(Temp)) {
  Serial.println("ERROR");}
else {
Firebase.setString("temperature", t);
Firebase.setString("humidity", h);}
BtnTauto= Firebase.getString("/Tauto");
Btnfan= Firebase.getString("/fan");
Btnheater= Firebase.getString("/heater");
// Tmin= Firebase.getString("/minT");
// Tmax= Firebase.getString("/maxT");
if(BtnTauto=="1") {
  if( Temp<a){
    digitalWrite(23,HIGH);
    digitalWrite(22,HIGH);}
  if (Temp>b){
    digitalWrite(22,HIGH);
    digitalWrite(23,LOW);}
    if (Temp>a && Temp<b){
      digitalWrite(23,LOW);
      digitalWrite(22,LOW);} }
  else {
    if(Btnfan == "0"){
      Serial.println("fan (OFF)");
      digitalWrite(22,LOW);}
    else {
      Serial.println("fan (ON)");
      digitalWrite(22,HIGH);}
    if(Btnheater == "0"){
      Serial.println("heater (OFF)");
      digitalWrite(23,LOW);}
    else {
      Serial.println("heater (ON)");
      digitalWrite(23,HIGH);} } }
  void readTdsQuick() {
    dallasTemperature.requestTemperatures();
    sensor::waterTemp =
    dallasTemperature.getTempCByIndex(0);
    // sensor::waterTemp =
    dht.readTemperature();
    float rawEc =
    analogRead(pin::tds_sensor) * device::aref
    / 4095.0; // read the analog value more
    stable by the median filtering algorithm,
    and convert to voltage value
    float temperatureCoefficient = 1.0 + 0.02
    * (sensor::waterTemp - 25.0); //
    temperature compensation formula:
    fFinalResult(25^C) =

```

Annexes

```
fFinalResult(current)/(1.0+0.02*(fTP-  
25.0));  
  
    sensor::ec      =      (rawEc      /  
temperatureCoefficient)      *  
    sensor::ecCalibration; // temperature and  
    calibration compensation  
  
    sensor::tds = (133.42 * pow(sensor::ec, 3)  
- 255.86 * sensor::ec * sensor::ec + 857.39  
* sensor::ec) * 0.5; //convert voltage value  
to tds value  
  
    Serial.print(F("TDS:"));  
    Serial.println(sensor::tds);  
  
    Serial.print(F("EC:"));  
    Serial.println(sensor::ec, 2);  
  
    Serial.print(F("Temperature:"));  
    Serial.println(sensor::waterTemp,2);  
  
}
```