PEOPLE'S DEMOCRATIC REPUBLIC OF ALGERIA
Ministry of Higher Education and Scientific Research

UNIVERSITY ECHAHID HAMMA LAKHDAR - EL OUED
FACULTY OF EXACT SCIENCES
Computer Science department

End of study memory
Presented for the Diploma of

ACADEMIC MASTER

Domain: **Mathematics and Computer Science**
Industry: **Computer Science**
Specialty: **Distributed Systems and Artificial Intelligence**

Theme

IoT Based SMART Irrigation System

Presented by:
Mohamed Nadhir Abid
Abdel Majid Derdega

Sustained on ....-Juin-2018 from the jury:

Miss. Chourouk Guettas MA (...) Supervisor Univ. El Oued

Mr/..... .... MA(...) President Univ. El Oued

Mr/..... .... MA(...) Reporter Univ. El Oued

Academic year 2017-2018.

ACKNOWLEDGEMENT

To MY Parents

Who offered their lives for me ...

Thanks To

The super special Teacher
Chourouk Guettas

For all the time and effort and encouragement

My Family Brothers and sisters: To my beloved grandmother Metira and my aunt Fatima and Dr.Khadija, Houdhiefa, Hind, Younes, Abdel badie, Ali, Toufik, and Amine

For their support and patient

My Friends

M.Anter, H.yahia, H.Younes, A.Bensalem, D.Majid, H.Aissam, M.Tarek, and all of my friends that i did not mention by name thanks

To Sahel Family

The ones who helped me a lot A.Farouk and Z.M.Mariem

A.M.Nadhir

To My Teacher

Chourouk Guettas

Thanks for your great support and continues care

To my family

The pure spirit of my grandmother, to my uncles and aunts, my brothers and sisters : Choaib, Yacine, Sara, Chourouk, Aroua, Logine.

For the support and care

To Sahel Family

The ones who helped me a lot A.Farouk and Z.M.Mariem

My Friends

A.M.Nadhir, K.Abde Assatar, H.Aissam, M.Tarek, H.Abdallah, T.Lazhar, D.Walid, A.Abdel Basset,B.Yazid and those who i did not mention by name thanks

D.Abdel Majid

ABSTRACT

Agriculture has been always hard and tedious job. This study is an effort to make it easy and less time consuming especially in deserted areas where heat and large spaces are inconveniencing criteria. To reach these objectives we applied different technologies such as Artificial Intelligence, Internet of Things and Cloud computing to realize smart irrigation system. Based on received information from sensors detecting weather condition, soil moister, and irrigation conditions of the plant, the system using intelligent agent decides whether to irrigate the plants or not. The irrigation system is enabled also with a manual control option that works in real time with time scheduling possibility.

Key words: Artificial Intelligence, IoT, Cloud Computing, Agent, Irrigation, Real time

Résume

L'agriculture a toujours été difficile et fastidieuse. Cette étude est un effort pour la rendre facile et consume moins de temps en particulier dans les zones désertes où la chaleur et les grands espaces sont des critères gênants. Pour atteindre ces objectifs, nous avons appliqué différentes technologies telles que l'intelligence artificielle, l'Internet des objets et l'informatique en nuage pour réaliser un système d'irrigation intelligent. Sur la base des informations reçues des capteurs détectant les conditions météorologiques, l'humidité du sol et les conditions d'irrigation de l'usine, le système utilisant un agent intelligent décide d'irriguer ou non les plantes. Le système d'irrigation est également activé avec une option de contrôle manuel qui fonctionne en temps réel avec possibilité de programmation horaire.

Mots clés : Intelligence artificiel, IoT, Cloud Computing, Agent, Irrigation, Temps réel

ملخص

الزراعة كانت دائما عملا شاقا و مستهلكا للوقت ، هذه الدراسة هي محاولة لجعلها اسهل و قل استهلاكا للوقت ، خاصة في المناطق الصحراوية ، حيث تكون الحرارة مرتفعة و المساحات كبيرة و هذه المعايير غير ملائمة للزراعة. و للوصول الى هذه الاهداف قمنا بتطبيق مختلف التقنيات مثل الذكاء الاصطناعي و انترنت الاشياء و كذا الحوسبة السحابية ، لانشاء نظام ري ذكي . انطلاقا من معلومات مستقبلة من الحساسات التي تستكشف حالة الطقس و رطوبة الارض و كذا شروط الري للنبتة ، يحدد النظام الذي يستخدم العامل الذكي ما اذا كان على النظام سقى النبتة ام لا . نظامنا هذا يمكن التحكم فيه بطريقة يدوية في الوقت الحقيقي مع امكانية جدولة الوقت .

الكلمات المفتاحية : ذكاء اصطناعي، انترنت الاشياء، الحوسبة السحابية، عامل ذكي، سقي، زراعة، الوقت الفعلي

CONTENTS

LIST OF FIGURES

LIST OF TABLES

GENERAL INTRODUCTION

One of the best innovation that human made is agriculture. Agriculture was and still the main reason that humans don't starve from hunger. The process of agriculture was always very tiring and time consuming in all human history. This process has changed and developed step by step from period to period. Adopting various solutions from handy solutions to mechanical ones ending with the automatic solutions today.

Todays solutions were not always enough even with the enormous and huge technological development. There are some performing solutions but in each there is always a lacking; whether its their high price or their poor adaptability. The problem we want to solve is the irrigation problem in agriculture. What time should the irrigation be, how much time should it take, and how to save water.

Our objective is to provide an easy using system that, can help the workers to provide time and effort. Yet with providing a good yield, that leads us to ask some questions for the study like, how the system will make the process easier, how the system will make the farmer save more water.

The importance of this study resides in merging nowadays technologies in agriculture field in our deserted region as an effort to improve the overall process taking into consideration the hard conditions of this area; such huge locomotion distance, soil nature, hot weather and time and water consumption. Without overlooking, of course, the importance of agriculture in Algeria's current economic.

In this work and starting from the previous criteria we decided that the system should be: Firstly remote, secondly has database, finally automatic. The remote part was divided into controlling and monitoring, using the Internet of Things (IOT) technology. The database is google firebase that has the months and days of irrigation. For water saving part, we implemented an automatic system that can calculate the right time considering the weather and decides how long the water will be applied using a rule base.

Other works similar to our system failed to do an automated system and some failed to make a cheap system. So some similar systems are Online farming based on embedded system [5], and IoT based automatic drip irrigation system [6].

In solving this problem we did not disregard the human choice so the system can work on both human orders or by itself.

This report will contains four chapters

- **Chapter 1 :** Will contain a background of the subject and more details about our motivation.
- **Chapter 2 :** Will contain other similar works and their evaluation, with the hypothesis and limitation of our study.
- **Chapter 3 :** Contains the architecture of the system and some a definition of the hardware parts.
- **Chapter 4 :** The last chapter will be about the implementation of the system and the obtained results.

Finally the general conclusion and our perspectives

CHAPTER 1

BACKGROUND

Introduction

To begin with, on one hand, artificial intelligence is a massive subject and a huge research domain, developing steadily. On the other hand, an interesting technology that has been around for a quiet time known as Internet. In the beginning, the internet enabled us, to connect with each other making the world smaller. The fast development of artificial intelligence and all other technologies allowed us to invent smart devices that we carry with us daily (smart phones, smart watches ...). These devices connected to the internet made a new step to new technologies..

The development was not exclusive for those small devices, but for all machines that we use daily such as: Tv, refrigerators, ... etc, which put us in front of a world full of connected smart devices. Nowadays, the smart devices in general are connected to the internet more than humans, by a ratio of 1.8 for each person.

The emergence of the internet of things come from those two big technologies mentioned above. Artificial intelligence gave us the ability to make smart devices denoted with the internet capability to communicate creating smart things(devices) with the power to communicate.

The communication has two categories : human to machine and machine to machine (M2M). The interaction human to machine is to provide some services from the cloud. Cloud Computing is an important technology that we will discuss in this chapter with other technologies mentioned above.

In this chapter we will be defining the main elements and notions in our project along with the definition of our problem and the represented solutions in today's reality, trying to realize a new perception using what the technology is offering.

1.1 Artificial Intelligence

Artificial intelligence is one of the biggest domains and we are going to briefly define it and talk about its history.

1.1.1 Definition and history

Artificial Intelligence

Artificial Intelligence AI, doesn't have one simple obvious definition; it is divided to four approaches, **thinking humanly, acting humanly, thinking rationally, acting rationally**. Every each of those approaches give the computer the ability to possess some capabilities [2].

The goal of work in artificial intelligence is to build machines that perform tasks normally requiring human intelligence, defined by Nilsson, Nils J. (1971). Research scientists in Artificial Intelligence try to get machines to exhibit behavior that we call intelligent behavior when we observe it in human beings by Slagle, James R. (1971). Studying the structure of information and the structure of problem solving processes independently of applications and independently of its realization in animals or humans, What is or should be, by McCarthy, John (1974). There is lot of definition and every author give a definition to AI depending on his point of view and philosophy [7].

Historically it appeared as a result of us Homo-sapiens trying to understand how we think. Work started in earnest soon after World War II, and the name **AI** was first used in 1956. Coined by John McCarthy, but the journey began much before that, with Vannevar Bush's seminal work *As We May Think.* Five years later Alan Turing wrote a paper on the notion of machines being able to simulate human beings and the ability to do intelligent things such as playing Chess [8].

Over the past sixty years AI main advances were in search algorithms, machine learning, and integrating statistical analysis into understanding the world at large. In AI field Expectations seem to always outpace the results in reality. We've built software that can beat humans in some games, but still not close to pass Turing test, building big Expert systems but not comparable to human experts [9].

1.2 Agent

The agent can be a human or a machine based on the domain. Also the study case.

1.2.1 Definition

The field of agents has many conflicting definitions, it's been defined as "AGENT - We distinguish two usages. The most general usage is to mean an AUTONOMOUS, self-contained, REACTIVE, PRO-ACTIVE, computer system,typically with a central focus of control, that is able to communicate with other AGENTS via some ACL. A more specific usage is to mean a computer system that is either conceptualized or implemented in term of concepts more usually applied to humans" [10].

Another definition is "An agent is able to take initiative and exercise a non-trivial degree of control over its' own actions" [11]. ***Figure.1.1***

In general an agent is anything that can be viewed as perceiving its environment through sensors and acting upon that environment through effectors.

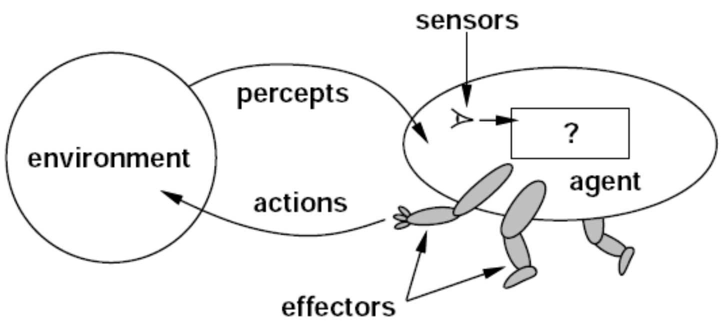


Figure 1.1: Agent [1]

Autonomy

The assumption that, although we generally intend AGENTS to act on our behalf, they nevertheless act without direct human or other intervention, and have some kind of control over their internal state... [10].

"Autonomous: An agent is able to take initiative and exercise a non-trivial degree of control over its' own actions" [11].

Reactive

Reactive as Capable of maintaining an ongoing interaction with the environment, and responding in a timely fashion to changes that occur in it.

Pro-active

A Pro-active agent is Capable of taking the initiative; not driven solely by events, but capable of generating goals and acting RATIONALLY to achieve them [10].

1.3 Classes of intelligent Agent

Intelligent agents are assembled into groups. based on their degree of perceived intelligence, capability and the way of interaction.

- Simple reflex agents
- Model based reflex agents
- Goal based agents
- Utility based agents
- Learning agents

Simple reflex agent

Simple reflex agent takes action based on only the current environment situation it maps the current percept into proper action ignoring the history of percepts.The mapping process could be simply a table-based or by any rule based matching algorithm. Depending on conditionaction rule, **if** condition **then** action. Some reflex agents can also contain information on their current state which allows them to disregard conditions. Also called situationaction rules, productions, or ifthen rules [2].***Figure.1.2***

Model based reflex agent

Model based reflex agent Needs memory for storing the precept history, it uses the precept history to help revealing the current unobservant aspects of the environment, allowing it the ability to handle a partial environment. It's stored current state maintain some kind of structure which describes the part of the world which cannot be seen. This knowledge about "how the world evolves" is called a model of the world, therefore the name "model-based agent" [12].***Figure.1.3***

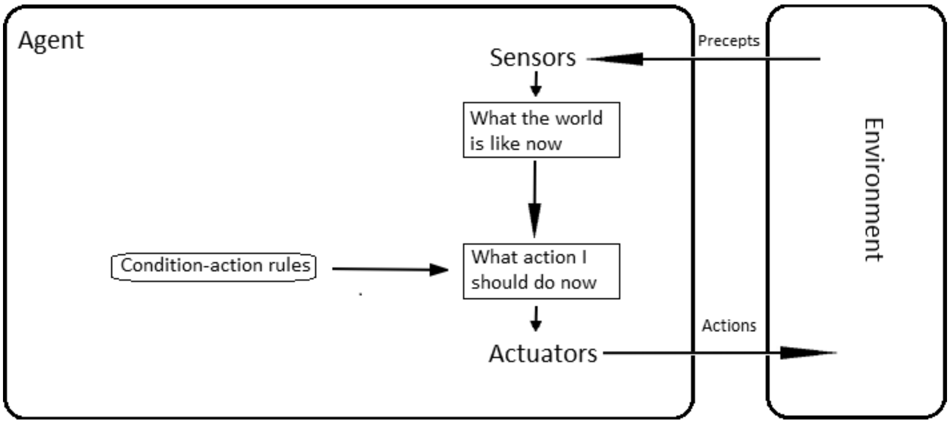


Figure 1.2: Simple reflex agent architecture [2]

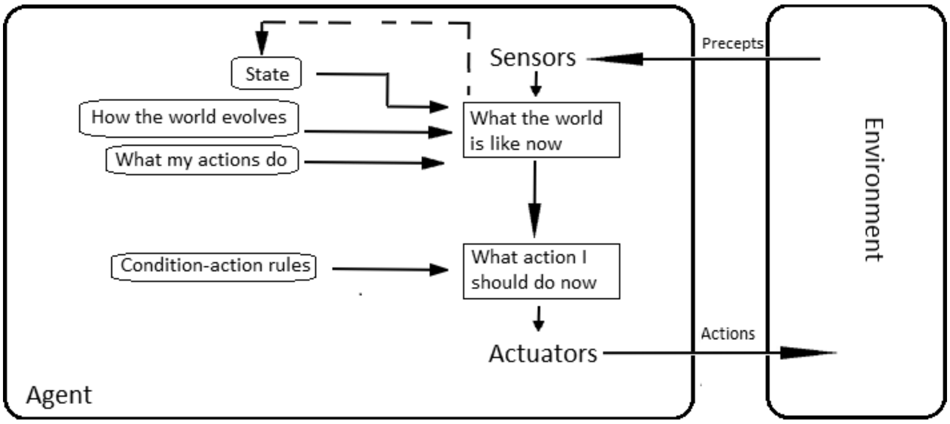


Figure 1.3: Model based reflex agent architecture [2]

Goal based agent

Knowing about the current state of the environment is not always enough to decide what to do.In other words, as well as a current state description, the agent needs some sort of goal information, which describes situations that are desirable. it must be able to reason and may be able to plan. This type of agent is described as Pro-active [13].***Figure.1.4***

Utility based agent

Goals alone are not really enough to generate high-quality behavior. Goals just provide a crude distinction between happy and unhappy states, whereas a more general performance measure should allow a comparison of different world states (or sequences of states) according to exactly how happy they would make the agent if they could be achieved. Because happy does not sound very scientific, the customary terminology is to say that if one world state is preferred to another, then it has higher utility for the agent [11].***Figure.1.5***

Learning agent

Agent may in some circumstances learn from past occurrences in the environment to predict the future and in some cases. Learning has an advantage that it allows the agents to initially

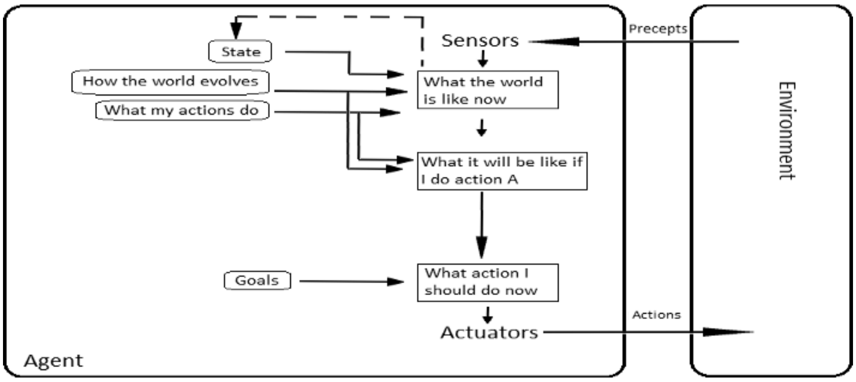


Figure 1.4: Goal based agent architecture [2]

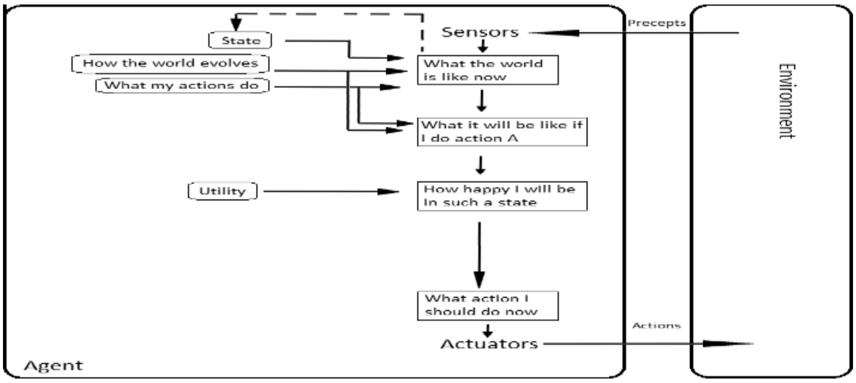


Figure 1.5: Utility based agent architecture [2]

operate in unknown environments and to become more competent than its initial knowledge alone might allow. The most important distinction is between the "learning element", which is responsible for making improvements, and the "performance element", which is responsible for selecting external actions. The learning element uses feedback from the "critic" on how the agent is doing and determines how the performance element should be modified to do better in the future [12].***Figure.1.6***

1.4 Internet of Things

Internet of Things (IoT), is defined by the ITU(International Telecommunication Union) and IREC (IoT European Research Cluster) as a dynamic global network infrastructure with self-configuring capabilities. Based on standard and interoperable communication protocols where physical and virtual "things" have identities. Physical attributes and virtual personalities, use intelligent interfaces and are seamlessly integrated into the information network. The same as AI, IoT have several definitions, **Kevin Ashton** an expert on digital innovation: "An open and comprehensive network of intelligent objects that have the capacity to auto organize, share information, data and resources, reacting and acting in face of situations and changes in the environment".

When the IoT was founded?. That term *Internet of things (IoT)* was coined by Kevin Ashton, Executive Director of the Auto-ID center in Massachute Institute of Technology (MIT)

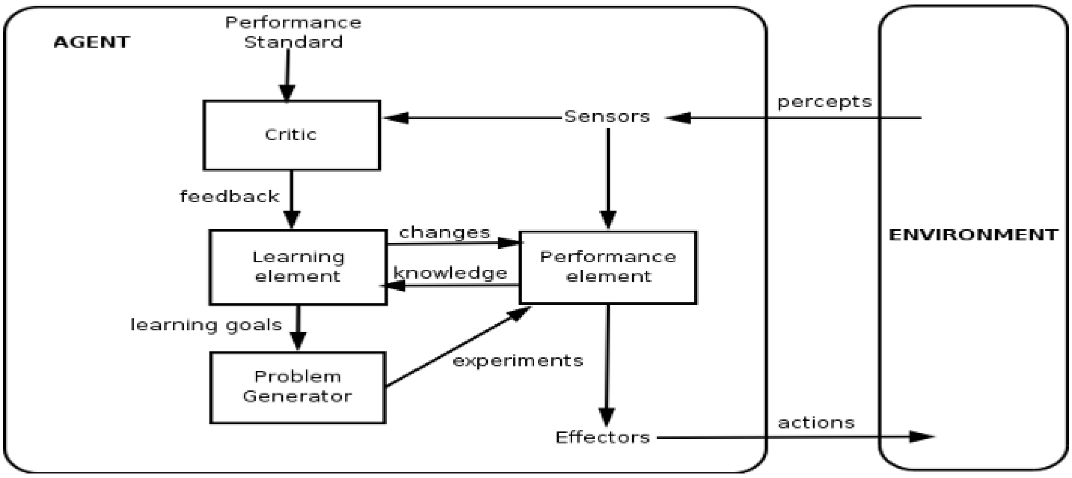


Figure 1.6: Learning agent architecture [2]

in 1999. In the same year Neil Gershenfeld first time spoken about IoT in his book "When Things Start to Think", and MIT auto-ID lab helped to develop the Electronic Product Code. In 2000, LG announced it's first Internet of refrigerator plans.

Two years later in 2002, The Ambient Orb created by David Rose and others in a spin-off from the MIT Media Lab is released into wild with NY (New York) Times Magazine naming it as one of the Ideas of Year.

After RFID (Radio-frequency Identification) was deployed on a massive scale by the US Department of Defense in their Savi program and Wal-Mart in the commercial world (2003-2004). The UNs International Telecommunications Union (ITU) published its first report on the Internet of Things topic in 2005.

The big year for IoT was 2008, where it was Recognized by the EU (European Union) and the first European IoT conference was held. In the same year a group of companies launched the IPSO Alliance to promote the use of IP in networks of Smart Objects and to enable the Internet of Things. The FCC (Federal Communication Commission) voted 5-0 to approve opening the use of the white space spectrum.

Between(2008-2009) the IoT was born according to Ciscos Business Solutions Group, at least in the same year US National Intelligence Council listed the IoT as one of the 6th Disruptive Civil Technologies with potential impacts on US interests out to 2025 [14, 15].

1.4.1 Area of applications

Today we relay in technology for our daily tasks, and our lives basics such as : Agriculture, education, energy, cars, health care and cities.

- **Agriculture** : In this important domain IoT will allow us more sophisticated and remote control of farm, better sensing and monitoring of production [16].
- **Education** : That domain, the fate of humanity depend on it, so engineers need to understand well the education programs. The integration of IoT make the education more interactive, and students be actively engaged which was proven scientifically a better way

to learn. IoT have more benefits; learn at your own place,focus on relevant content only, richer and interactive content. Content are recorded ,meaning any time, any place [17].

- **Energy** : Energy consumption is really going huge especially with all of this technology advancing. IoT giving us the ability to economise the energy, by offering the power of decisions making to the things "objects", so they be able to decide whenever is a good time to turn on or off. Also enabling the remotely control (e.g, Lights switch on once you walk into the room and switch off once you walk outside, claiming in the winter; the devices responsible will turn off once the weather is good ...). Smart energy is classified as the third global share of IoT projects, with 13 % and the strong adopters of smart energy are south and north America [18].

- **Cars** : Safe roads more lives saved, the connected cars allow a M2M (Machine to Machine) connectivity. That mean real time decisions to avoid accidents which is a Honda project. There are other companies working into those projects such as insurance companies. Most of IoT projects are focused on vehicle diagnostic and monitoring, and same as smart energy 13 % in global share of IoT projects.

- **Health care** : Connected health care, is at the same level or higher in importance from the previous axes. Even so connected health suffer a slow adoption in information technology. This technology offers a more data collection and continuous medical record from the smart sensors all over the patient body, that would eliminate the need of frequent sessions by making the Doctor in your device, and help patient to stay in medication . The benefits isn't for one specific type of the society; patients, caregivers, hospitals, populations, and policy makers [19].

- **Cities** : Smart Cities are a combination of smart devices; cars, trash bins, traffic lights, smart phones, smart buildings. All of those smart devices will work all together to provide a safe and more comfortable cities [20, 21].

IOT is a way to make our life more remotely and easier, and today it is integrated into the major axes in our lives.

1.4.2 Link between IOT and IA

To clarify the link between these two big domain of researches. We have to understand more about the detail of how the work is done. IOT work through five steps :

1. **Sense** : Sensing by smart sensors (mechanical, thermal, electrical, optical and chemicals) the environment (physical world). Sensors will gather the data from the physical world, so it should be many and small to build a wireless network.

2. **Transmit** : In this phase the sensors send the data. Nowadays sensors communication divided into short-range radio technology and LPWAN (Low Power Wide Area Network). Short-range radio technology (local communication methods) (e.g, IEEE 802.15.4, Zigbee, Bluetooth Low Energy ...). Another emerging technology **LPWAN** (Low Power Wide Area Network), sensors communication is to transmit data to another device or to the central data base.

3. **Store** : The data received and transmitted from the sensors is stored in central data base. To allow the processes (filter, analyse, aggregate, ...).

4. **Analyse** : This phase is the most important one, and the true value of any IoT service is determined. This is where artificial intelligence or more properly the subset of AI called

machine learning will provide a crucial role. Machine learning is a form of programming. Empowers a software agent with the ability to detect patterns in the data presented to it. So it can learn from these patterns in order to adjust the ways in which it then, analyzes that data.

5. **Act** : Acting isn't confined into physical form only, like deploying an ambulance to the site of an auto accident. It can be software acting, like sending a text message or delivering a report.

To consider an IoT worthable it need to be able to act, but acting isn't what all IoT about, the most important phase is the analyze, where AI take a place Defining the link between it and IoT.

1.4.3 Cloud: A ground for IOT

As a fact we can't deny that Cloud Computing is growing so fast. In the other hand so do IOT. IOT based on intelligent and analysing and acting so the question here why IOT need Cloud Computing ?, or the opposite. In reality IOT depend on small real things which most of times has a short storage and limited, that's one reason to go with and the other is, that those connected small things don't have a big computing or calculating capacities. In the other side we see that Cloud Computing has **Unlimited** Storage capacity, also processing power. All of Cloud Computing capacities will allow IOT and us to treat data in a better way in security and performance and enabling us to analyze and store all of those data. The emerging of those two powerful technologies will re-shape the world in the future, and will cause new capabilities and new paradigm [22].

1.4.4 Devices that make IoT

When it's about IoT devices, our minds go directly to imagine smart phones and tablets. Although smart phones and tablets have sensors, accelerometers and gyroscopes, also they are embedded devices with displays and keypad and can be connected to the internet. Therefore they fit every requirement of IoT devices ***Figure.1.7***. The suspicion was explained at a keynote event during Sept 2011s Mobile World Congress in Barcelona by Qualcomm Chairman and CEO Dr. Paul Jacobs. Paul Jacobs talked about how mobile technology could be used to connect non-phone and non-tablet devices called IoT devices and objects to the internet. In the future where everything will be connected on the web and mobile phones will serve as the hub, or the remote control, for the Internet of Things. From here we proceed to describe the characteristic of those smart devices, that makes the IoT: [15, 14]

- A set of system hardware and software Information and Communication Technology (ICT) resources.
- Dynamic component-oriented resource extensions and plugins (Plug and Play) of some hardware resources.
- Remote external service access and execution.
- Local, internal autonomous service execution.
- Access to specific external environments: human interaction, physical world interaction and distributed ICT / virtual computing interaction.
- Ubiquitous computing properties.

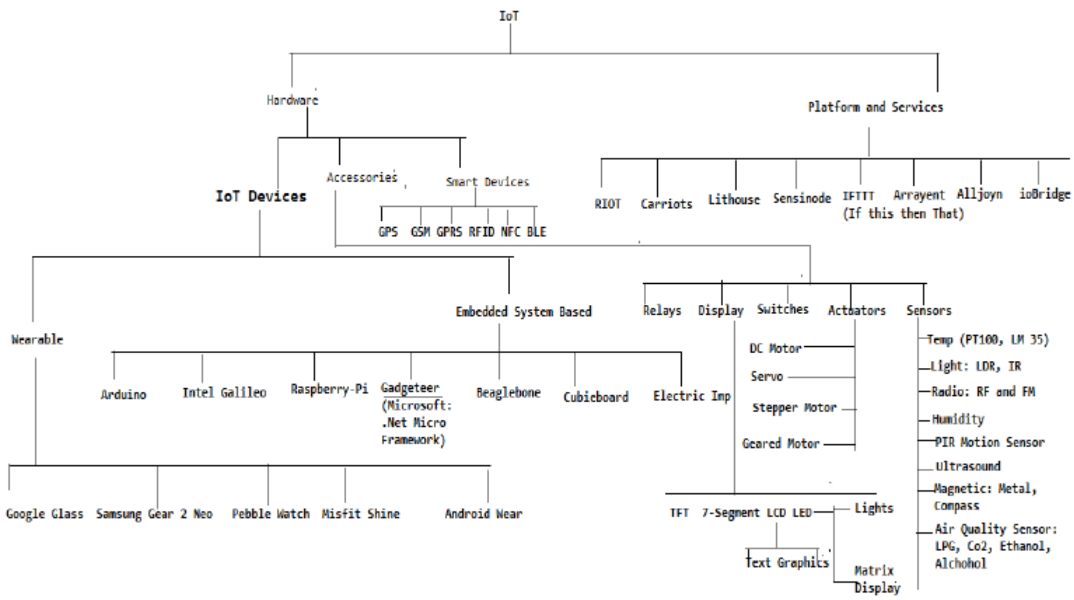


Figure 1.7: IoT devices [3]

1.4.5 IoT Platforms

IoT development can be divided into three platforms: Wearable, Embedded and Cloud that we are going to talk about next section. Developers can build apps for custom Wearable devices (Samsung Gear) or can often create their own platform using Embedded solution and then can develop app for that platform.

Wearable Platform

IoT most popular platform are getting day by day more used, because of big companies investment such as Google, Samsung.

Wearable devices getting more popular everyday, like android wear used for health care and Daily business and pebble watch work with both apple and android letting the user know they're heart beat, and make you know yourself better.

Embedded Platform

Embedded software solutions and services can be used to deliver feature rich IoT and connected devices that also meet the reliability, security and power consumption requirements of the demanding markets such as medical, industrial and automotive. Arduino is probably the best starting point, Raspberry Pi is one of the best and many other choices.

1.5 Watering systems: What do exist and what should exist ?

Before technology come up we as human had to make laws and systems to create an easier life, those systems kept changing and developing until our time being, and that bring us to our two mean questions: What do exist and what should exist ?. To answer the first question, the watering systems that do exist now some of them are manual but lot of them are automated, and in every system or service or technology the quantity doesn't matter, but the quality do. Which lead us to answer the second question, due to all what technology is offering today, we must have the most easier and simple and effective systems, We can evaluate it regarding to automatism, time scheduling, data analysis and most of certainly advises.

1.5.1 Existing manual watering system

Watering systems aren't something new; on opposite it's old as agriculture. Some existing systems are **Drip irrigation System**, **GreenWall irrigation** and **Sprinkler irrigation System *Figure.1.8***, and the last is classified : **fixed, periodic move, continuous move**. Has types like : **Center pivot, Traveler, Hand move, Side roll, Solid set**. ***Figure.1.9*** Those we mentioned are just a small sample and there a lot of other existing manual watering systems [4].

(a) Drip irrigation system (b) GreenWall irrigation (c) Spinkler irrigation system

Figure 1.8: Some manual irrigation systems [4]

1.5.2 Time scheduling : When?

One of the main problems that face automated irrigation systems isn't automation but it's time scheduling, the problems that is really important in our day, is time management. Time shouldn't be wasted on simply watering plants but instead it can be used to do more important things. At this point other problems related to when will rise, presuming that the system aware of the real time, and even so when to water day or night?, how many times?, what hour?. As a solution we suggest a system that enable the user to control remotely or use machine learning, and another solution is System Expert (SE), or we can think of an intelligent agent.

1.5.3 Time scheduling : How Long?

Eventually, another question pop-up it is **how long**, cause some plants don't need much water instead it can kill them. A system that takes advantages of machine learning, remotely control SE or intelligent agents can theoretically solve the problem.

Conclusion

In this chapter we had defined the main elements and concepts involved in this projects. Trying to clarify and offer a background, by discussing the different technologies involved.

(a) Center pivot

(b) Traveler

(c) Hand move

(d) Side roll

(e) Solid set

Figure 1.9: Sprinkler irrigation types [4]

Problems we talked about are just a piece of the big picture, but we did focus on the important ones. Many solutions and systems were provided either manual or automated, but each still miss something.

Next chapter we will provide a better look into existing systems. Going through they're architectures and how they works, depending on the criteria points.

CHAPTER 2

LITERATURE REVIEW

Introduction

The existing of automated irrigation systems is necessary. Because of, firstly, the importance of a better agriculture and food raising, also eliminating the human errors that kills plants because each of those plants have watering and climate conditions. Secondly, reducing the hard and tiring work that farm workers do, workers or farmers will not be obliged to check in every plant in big farms. Also time saving, the farmer existence in the farm won't be necessary anymore. Finally exploiting the technology to provide for better purpose.

In this chapter we will discuss some related works, trying to demonstrate the points that others failed to cover and the strong points in those systems. Showing the differences between those systems, and what will our system include eventually.

2.1 Related Works

Various researches and many automated irrigation systems were developed. They adopted different ways to determine the soil condition, each of those researches worked to have an accurate soil water sensing. Another thing was discussed is power sources for the sensors, those sources may have a long life but not auto generated, or not environmentally friendly. Besides, the technology for making a network through the sensors and designing a control system, the network with a less limitation and of course less energy consumption.

2.1.1 Systems Classification

Numerous systems were built, from basic ones to more technologically advanced ones. All trying to achieve one big goal;a better irrigation system. Those systems can be classified according to some criteria into :

1. The used Technology
2. The Embedded Processors
3. Incorporated Sensors
4. The Adopted System

Based on those qualities we can evaluate the system performance.

The used Technology

Technology used in irrigation systems are such as IoT and Cloud Computing also the internet and others like GSM and GPRS, those technologies had different specialties, some serves the communication side other handle data storage. Other technologies were used are, water soil measurement and thermal sensing, considered as a way to detect the plants needs of water and other stuff. Each of those technologies serves a propose.

The Embedded Processors

Diverse number of embedded processors used in irrigation systems. An embedded processor is; a microprocessor that is used in an embedded system and it's designed for handling its needs.

These needs are represented in less power requirements, and those processors are usually very small and draw less power. Embedded processors are even divergent on the basis of their clock speed, voltages and storage size. Usually, embedded processors have a storage capacity [23].

Embedded processors divided into two main types :

- Ordinary microprocessor
- micro-controller

Those two types have been always confused withe each other. They have common characteristics but yet have big differences. The vital different is that Micro-controllers have more peripherals on the chip, it has everything built in it ***Figure.2.1*** , besides contains the circuit of a microprocessor, and has built in; RAM, ROM, I/O devices, timer and counters. But Microprocessor depend on other chips in many functions, also it contains; general purpose register, stack pointer, program counter, timing and control circuit in addition to interrupt circuit. Microprocessor is suited to processing information in computer systems, instead micro-controller is suited to control I/O devices. Plus it has many bit manipulation instructions, on the opposite for microprocessor it has one or two bit manipulation instructions, and it has less number of malfunctioned pins which is on counter for the micro-controller. In the first hand we have; a flexible design,access time for memory and I/O, and expensive. In the other hand micro-controller have; less flexible design, require less hardware, less access times for built-in memory and I/O devices, and cheap. Micro-controllers are generally considered more useful as they require less support circuitry than microprocessors.

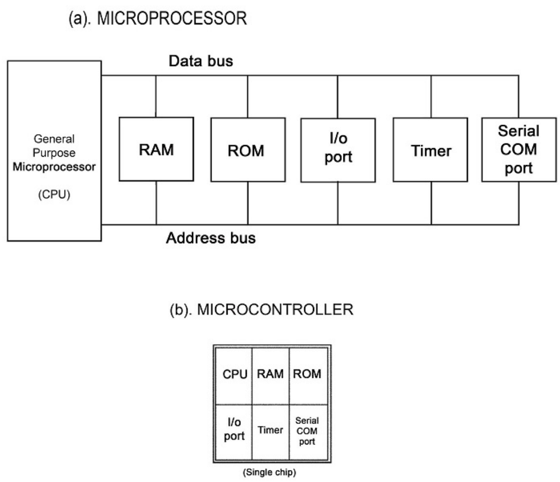


Figure 2.1: Microprocessor block and Micro-controller block [23]

Embedded processors can be programmed to do a specific intended work. Consequently, the CPU architecture have to be varied. The most common among those architectures types are RISC (Reduced Instruction Set Computer) and non-RISC like CISC (Complex Instruction Set Computer).

Many applications today uses embedded processors without us noticing. They are used in portable devices such as digital cameras, digital watches, GPS units. Also can be founded in large systems like controlling power plants, traffic lights. Other domains where embedded processors are applied just as Telecommunications systems uses it in telephone switches and mobile phones, and consumer electronics in printers and video games consoles, DVD players.

This diversity allowed the ability to combine more than one processor in one system. Some Embedded processors such; Arduino, Raspberry pi, node mcu ... ect.

Incorporated Sensors

Sensors can be defined as a devices that detect the changes in the physical world. Those changes or inputs can be light, motion, pressure, heat or moister. The output generated usually signaled and converted so human can read it. There are so many types of sensors. Those types can be classified as analog sensors and digital sensors. But, there are a few types of sensors such as temperature sensors, IR sensors, ultrasonic sensors, pressure sensors, proximity sensors and touch sensors that are frequently used in most of the electronics applications.

- Analog sensors : produce continuous analog output signal; practical examples : accelerometers, pressure sensors, light sensors, sound sensors, temperature sensors
- Digital sensors : Electronic sensors or electrochemical sensors in which data conversion and data transmission takes place digitally are called digital sensors. These digital sensors are replacing analog sensors as they are capable of overcoming the drawbacks of analog sensors

The adopted System

In general, irrigation system can be divided into two main categories: The first Low-tech principals which considered for the no automated systems and need no power supply. The second, High-tech principles which concerned automated systems. There is more than one High-tech automation system: Time Based system, Volume Based system, Open Loop system, Closed Loop system, Real Time feedback and Computer Based irrigation control system [24].

- Time Based System: The irrigation is based on time clock or timers, which are integral parts. The timer apply the necessary quantity of water at the right time. So the timer in order to be effective must be well programmed, avoiding under or over irrigation cases [25].
- Volume Based System: The irrigation is based on amount of water, and not time or duration. The system have to calculate, so that pre-set amount of water be enough for all the irrigation.
- Open Loop System: It takes the two specifications from the previous system. The system is based on operator that decides the amount of water to be used and the time of the event. It usually come with a clock or a timer that is used to start the irrigation, open loop system don't just work with time schedule, it works also with water volume [25].
- Closed Loop System: That system depends on an operator, and the operator develop a strategy. To generate that strategy the system a continuous feedback from one or more sensors. As a result of those data from the sensors the operator take decisions and action. The continuity of the feedback and action make a closed loop, which the operator considered as a strategy. That controller in the closed loop system require data

acquisition of the environment parameters (soil moisture, temperature, etc) and system parameters(flow, pressure, etc) [25].

- Real Time feedback: This whole system stands on the dynamic plant and the plant root zone demands. Offering the ability to the plant to determine the degree of the irrigation and the time. That system requires numerous sensors: temperature sensors, humidity sensors, rain sensors etc. Those sensors supply the controller to operate [25].

- Computer Based Irrigation Control System: Usually computer based system not specified into irrigation only and it's a combination of hardware and software. therefore it perform as a supervisor in irrigation and other related practices such as fertigation and maintenance. The system in general can be divided into two categories: interactive systems and fully automatic systems [25, 23].

2.1.2 Irrigation systems objectives

The main objectives of the new generation of automatic agriculture systems are :

- Remote monitoring: it is known in many domains as in health and security and irrigation. Remote monitoring is a standard specification that facilitates monitoring, through the use of remote devices known as monitors such as (smart phones, tablets , pcs, etc.). Those devices display the status from the sensors.

- Remote control: Depends on many different technologies (wifi, RFID, BLE, etc). Allowing us to control machines or large complex facilities from remote devices .

- Information transfer: The output that the sensors generate, has to be stored and transferred to the controller and to the monitor if it's founded. Nowadays systems look for the best and fast ways of information transfer and storage.

- Communication: The communication side is the most important to make all the previous enabled. The sensors controller is one side of the communication. depend on wired and no-wired, which is a low distance area and can use technologies such as Zigbee, Z-Wave, RFID. The other side is sensors to monitor and also monitor to controller, which is a long distance area, the technologies that can be used are GSM and LoRA or GPS and GPRS.

2.2 Literature Survey

In this section we are going to mention and classify existing systems, trying to present for each the methodology, advantage, disadvantage, approach and the title of the article [24]. ***[51]***

These works are classified based on their chronological order: **Table.2.1**

Begin of Table					
No	Article	Approach	Methodology	Advantages	Disadvantage
1	Online Farming Based On Embedded Systems and Wireless Sensor Networks [5] **[23]**	Soil water measurement	Proposed system uses IP cameras and wireless network to remotely monitor farm. System consists of three modules. Front end uses various sensors to monitor soil parameters. Management module control irrigation and monitoring module monitors and controls the system using internet.	• Easy to apply in practice. • Provides remote monitoring and control.	• Selecting the position that is representative of root zone is difficult. • Not accurate as direct measure.
2	IOT Based Automatic Drip Irrigation System [6] **[18]**	Soil water measurement	Proposed citrus crop water Management using moisture, temperature and humidity sensor. Sensor values are used to decide and control the amount of water required to supply through valves and then to drippers.	• Easy to applyin practice. • Less Expensive.	• Not accurate as direct measure.

Continuation of Table 2.1					
No	Article	Approach	Methodology	Advantages	Disadvantage
3	An Automatic Irrigation System using ZigBee in Wireless Sensor Network [26] **[24]**	Soil water measurement	Proposed system uses various sensors to monitor soil parameters. System consists of three nodes in which node 1 and 2 are sensing nodes and node 3 is receiving node. Sensing information is transmitted from receiving node to microcontroller and PC. Received value is compared with threshold value of crop to automate irrigation.	• Easy to applyin practice. • Less Expensive.	• Selecting the positionthat is representative of root zone is difficult. • Not accurate as direct measure.
4	Precision Agriculture Applications using Wireless Moisture Sensor Network [27] **[21]**	Soil water measurement	Proposed Green House Management System (GHMS) using wireless technology. Based on the sensor values GHMS automates the irrigation. Proposed system was tested on 100 chili plants.	• Easy to applyin practice. • Less Expensive.	• Not accurate as direct measure.

Continuation of Table 2.1					
No	Article	Approach	Methodology	Advantages	Disadvantage
5	Automatic Irrigation System Using Wireless Sensor Networks [28] **[20]**	Soil water measurement	Proposed automatic irrigation system using WSN. Moisture and temperature sensor values are used to control irrigation.	• Easy to applyin practice. • Less Expensive.	• Selecting the positionthat is representative of root zone is difficult. • Not accurate as direct measure.
6	Automatic Control of Irrigation System in Paddy Using WSN [29] **[19]**	Soil water measurement	The proposed system uses moisture sensor and water level sensor to automate irrigation. Moisture sensor value is compared with threshold value to start irrigation. Water level sensor value is used to stop irrigation. Time taken for irrigation is sent as an SMS to farmers mobile phone	• Easy to applyin practice. • Less Expensive. • Provides remote monitoring using mobile phone	• Selecting the positionthat is representative of root zone is difficult. • Not accurate as direct measure.

Continuation of Table 2.1					
No	Article	Approach	Methodology	Advantages	Disadvantage
7	Smart Drip Irrigation System for Sustainable Agriculture [30] **[22]**	Soil water measurement	GSM and ARM processor are used to design fully automated drip irrigation system. Proposed system monitors soil parameters using different sensors. Based on received sensor value ARM9 processor will take necessary action such as opening and closing the valve. Depending upon measured PH value suggestion is given to farmer through phone to add various chemicals.	• Easy to applyin practice. • Less Expensive. • Provides remote monitoring using mobile phone	• Selecting the positionthat is representative of root zone is difficult. • Not accurate as direct measure.
End of Table					

Table 2.1: Similar systems.

2.3 Advantages and limitations

Any developed system has its limitations and advantages.

2.3.1 Advantages

- The elimination of the manual hard and tiring operations, like opening and closing valves.
- The possibility to optimize the irrigation processes, and to change its frequency.
- Exploitation and adoption of new technologies, also the adoption of complex and difficult irrigation systems.
- Increasing the efficiency of water uses, and optimizing the water consumption.
- No specific time to start the system and the irrigation.

- Time scheduling so specific.

2.3.2 Irrigation systems limitations

- The system can be very expensive
- Not adaptable for the big-scale systems, and complex systems.
- power consumption and electricity need.
- sensors errors rate.

Conclusion

In this chapter we had an overview on related works. By going through systems classification, and the objectives that every irrigation system is trying to achieve.

In this same chapter we classified some similar systems. Representing their methods, approaches, advantages and disadvantages.

In the next chapter, we will present our system. The method and the approach we used, trying to highlight the software and hardware side of this project.

CHAPTER 3

SYSTEM ARCHITECTURE

Introduction

A good architecture leads to a good system. Its advantage appears in the control of the actions or in the automation. Each system can measured from the its architecture and the idea that defines how it work.

In this chapter we will work through the process of designing step by step to achieve the final system. Showing the flow chart and block diagram, then passing to the advantages and limitations of similar systems en general.

3.1 Architecture

To identify the system and the way it works, we are going to go through The general Architecture. ***Figure.3.1***

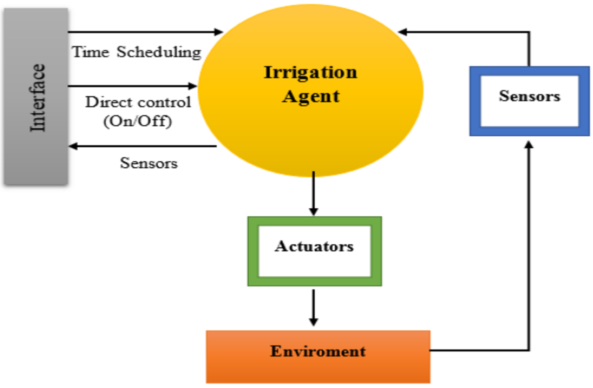


Figure 3.1: general Architecture

3.1.1 Interface

The interface is the way that user interact with the agent. The user interact in two ways. Either by allowing time scheduling represented in start time and end time. Secondly is Or direct on/off control in real time. The only interaction that the agent makes with the user is updating the sensors status.

3.1.2 Environment

The environment is the space where the agent updates its status from the sensors and interacts with that space using its actuators.

The goal of the agent is to make that space well irrigated. Based on the variables which are, the user interaction and the updated status of the environment from the sensors.

3.1.3 Sensors and Actuators

The sensors are the direct equipment that makes the exploration of the environment by the agent possible.

The actuators are the direct equipment that makes the interaction of the agent in the environment possible.

3.1.4 Irrigation Agent

The agent interact with environment by its sensors which is clarified in the previous section. The information obtained transmitted to lists and compared with the database and rule base, allowing the agent to create a situation each situation drive to a rule that rule transmitted into action.

The action got executed by the actuators, influencing the environment making the sensors gathering a new updated information which send us back to the same process.

3.2 Agent controller

3.2.1 Flow Chart

The proposed algorithm is shown as a diagram in **Figure.3.2**. After we start the system, the sensors which are placed in the root of the plants, transfer the temperature, moisture of the soil and if it is raining or not in a particular area. Those values transferred to the agent, so it decides to irrigate or not by going through the steps showing in **Figure.3.2**. Then update the status in the cloud so it be updated automatically in the user monitor. The other available irrigation option in the system is by time scheduling.

perception

This step is the first step made by the agent, which is generated by the sensors. The perception get temperature and humidity also soil moister creating the interaction between the environment. Those information transferred to the next step Data Processing.

Data Processing

Once the perception is done, the information be processed with the data gathered from the database. The processing is by changing all the information into proper boolean (0 and 1)**Figure.3.3**. Those value get transmitted to be stored in a array more explained in the next step.

Algorithm 1: High Temperature

Result: Get Temperature
if *temp* > *30* **then**
  return 1;
else
  return 0;
end

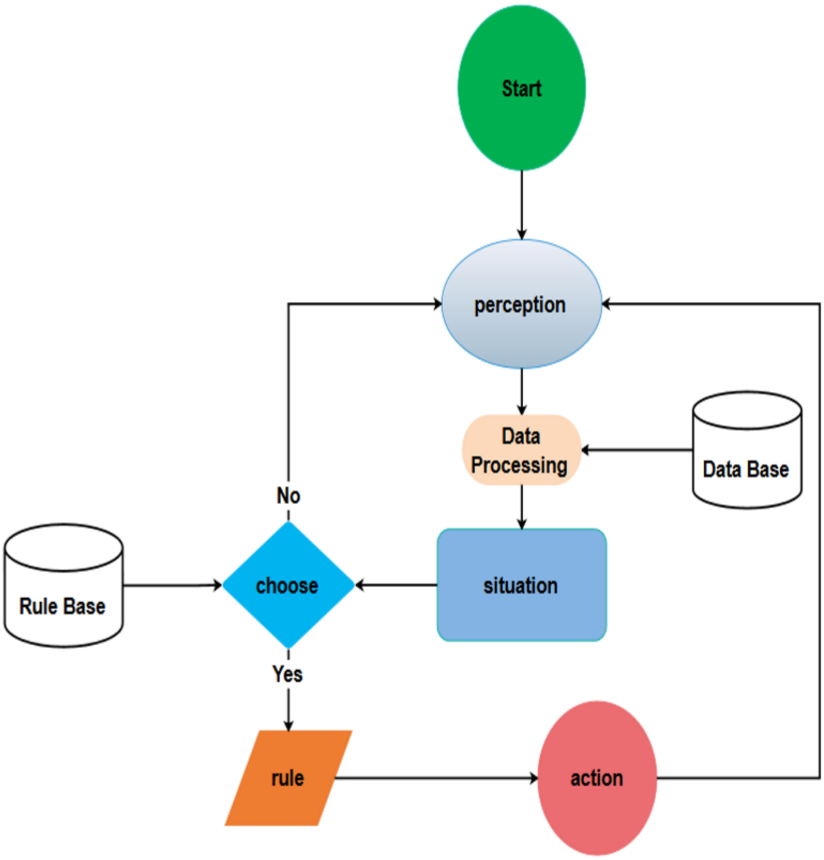


Figure 3.2: Flow Chart Diagram

Algorithm 2: Low Temperature

Result: Get Temperature

if *temp* < 30 **then**
  return 1;
else
  return 0;
end

Situation

The gathered information transformed to boolean data as shown in**Figure.3.3**. Those values stored into an array that has the size of the number of sensors of the system**Figure.3.3**. The situation is an array combined by the information of the environment and the database data. Each element can take one value even 0 or 1.

The Database contain the month of irrigation and the day, hour, and minute. Also it contain how long the irrigation based on that date, the database we are using handle potatoes irrigation scheduling. The next table explain more each element of the list.

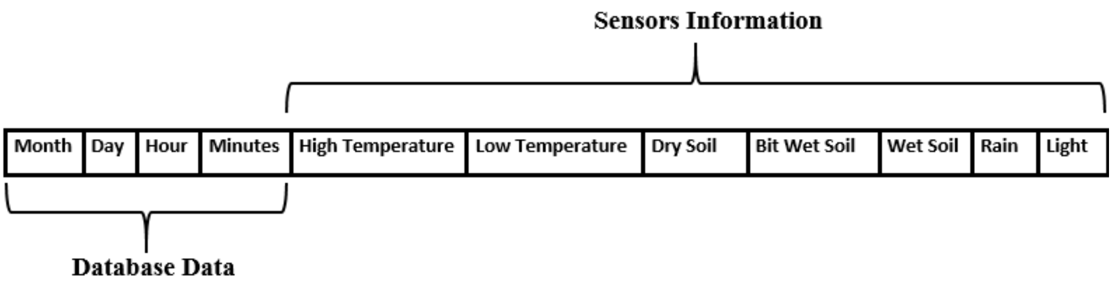


Figure 3.3: Situation List

Element	Explanation	Source
Month	The first element and sign to the month of irrigation	Data base
Day	This element sign to the day of irrigation	Data base
Hour	sign to the hour the irrigation should be	Data base
Minutes	sign to the minute the irrigation should be	Data base
High Temperature	symbolize the temperature of the environment when its high	Temperature sensor
Low Temperature	symbolize the temperature of the environment when its low	Temperature sensor
Dry soil	sign to the soil moister in environment	Soil sensor
Bit wet soil	sign to the soil moister in environment	Soil sensor
Wet soil	sign to the soil moister in environment	Soil sensor
Rain	sign to the rain in environment(raining or not)	Rain sensor
Light	symbolize the night and the day in environment	Light sensor

Table 3.1: Situation elements

Rule Choosing

This process has two inputs the situation that was generated before as a result of **perception** and **Database**. The other input is Rule base, the work that step do is to compare the situation array of zeros and ones with rule base giving one output which is the rule chosen if it does exist. If the rule does not exist the system return to the first step which is the perception.

Rule Base

It's a matrix contain number of rules, each rule fire an action. The rules are generated once the system start, those rules cover all possibilities of situations. Have the ability to accept more rules the only limitation is the memory of the agent.

Rule

The rule is an array of 11 element simulating the situation array **Figure.3.4**. The size of the array depend on the size of the situation array and the sensors provided in the system. The rule is a result of the situation and rule base, but that array resultant give also an output which is the action.

M	D	H	N	High Temp	Low Temp	DS	BWS	WS	R	L

Figure 3.4: Rule List

Action

Reaching that step mean that all the previous steps were executed successfully and each gave a result. The chosen rule fire an action each action after the execution return the agent to the first station which is the perception. The action also is an array with the same size **Figure.3.5**(10 element) each element define a condition was fulfilled in the situation and the rule array. The next Figure shows an example of how those combined steps produce an action. **Figure.3.5**

800 m	600 m	400 m	200 m	100 m	50 m	20 m	10 m	5 m

Figure 3.5: Action List

The first list is the situation. The first four values indicate to month, day, hour, Minute of irrigation. The next two values indicates the temperature (1) is high temperature requiring the other value next to it to be. Going to the rest of the list three next value refers to the soil status, dry soil (1) obliging the next two values to be (0), the other two one for rain the other for light representing day and night. The situation to rule go through rule base and the choose step.

The rule list has the same properties as the situation. As explained before each rule has an action. The list shown in the figure ***Figure.3.5*** the first value take (1) meaning 800 minute of irrigation, the next take (0) which indicate to 600 m, and so on. the result is 800m + 400m + 50m. The next table shows some rules and some resulting actions taken from real situation on the system.

Situation	Rule	Action
1—1—1—1—1—0—1—0—0—0—0	1—1—1—1—1—0—1—0—0—0—0	54738710 ms
1—1—1—1—1—0—0—1—0—0—0	1—1—1—1—1—0—0—1—0—0—0	42803520 ms
1—1—1—1—1—0—0—0—1—0—0	1—1—1—1—1—0—0—0—1—0—0	39600000 ms

Table 3.2: Situation-Rule-Action

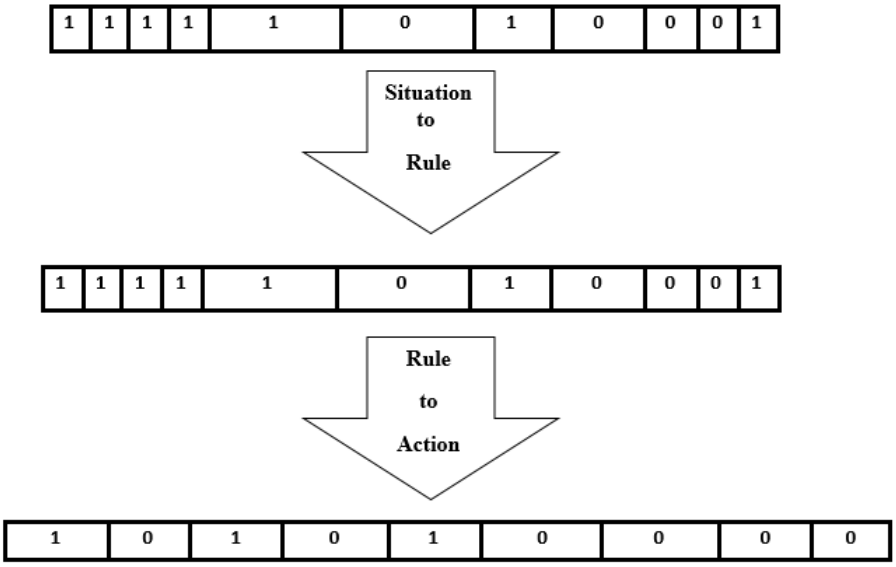


Figure 3.6: Action List

3.2.2 Block Diagram

The ***figure.3.7*** shows the structure of the system. In the first side, the input side containing the sensors, that transfer the data and information to the controller. The output in the other side divided into two: First via the phone through the cloud. Second via the LCD and actions via water pomp.

The output via phone go through the cloud, needs internet connection to show the statics and real time sensors status. The LCD shows also statics and sensors status, but with no need to internet connection. The other output is in a form of action, which is turning on or off the water pomp.

3.2.3 Sensors

Electronic parts that detect temperature and humidity, that detect also soil moisture and the rain.

3.2.4 Micro-controller

The mother board that manipulates and control input/output signals.

3.2.5 Input/Output

Inputs are the feedback or the signals that come from the sensors to the mother board. Outputs are the operation that the mother board is going to take based on the input. Also the results that are shown in the LCD.

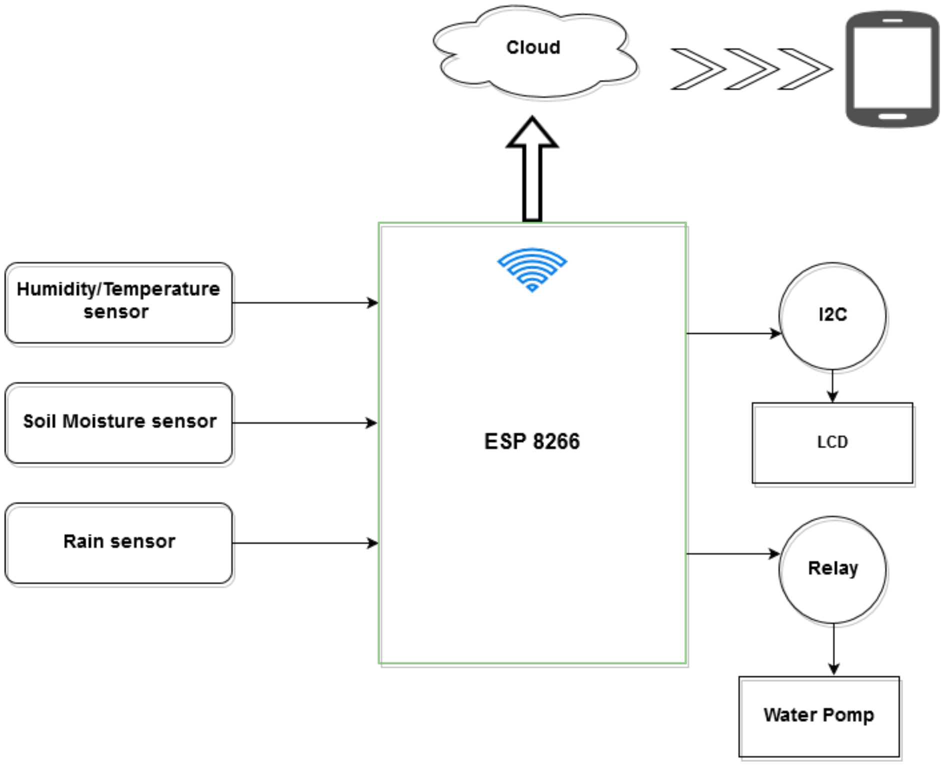


Figure 3.7: Block Diagram

3.2.6 Action

The action to turn on or off the irrigation.

3.3 Hypothesis

3.3.1 Irrigation

The system is designed to be able to lunch automatically the irrigation by three ways:

- **Soil moisture**: This way depends on the soil moisture sensor, whenever the water in the soil is low the sensor send the value to the controller to start irrigation.
- **Temperature**: The second way, that the system depends on temperature, the sensor sends the value to the controller and when the temperature get high enough, the controller starts the irrigation.
- **Time scheduling**: The third way to start irrigation automatically. Depends on time setting, the starting time of irrigation and the duration.
- **Database**: The most important one is the the time of irrigation that affects directly on the automatism of the system so that no action will be fired automatically if it is not matched with database conditions
- **Rule base**: The last factor is the rulebase that contain an ensemble of stats that got compared with the situation of the agent to make an action.

3.3.2 Real time System

The system works in real time in data sharing and sensors status, real time remote controlling and monitoring.

- **Real time data sharing**: All the data the controller receives have to be uploaded to the cloud and to the phone application, in the same moment.
- **Real time remote controlling**: The system beside the automation, can be controlled from the distance via Wifi. The control has to be in real time meaning in the exact moment.
- **Real time remote monitoring**: The information generated from the sensors and the status of those sensors, has to be showed in the phone also the status of the water pomp. All in real time and in the same moment.

3.4 Obstacles of the study

The major difficulties we faced are super low budget and time constraint. Which affected negatively in our project and system, dropping the possibility to program an expert system. The low budget did not allow us to buy the materials so we have a bigger range of connection and that is a typical wifi range: 46m indoors, 92m outdoors. The system supposed to be based on GSM rather than wifi. Which obliged us to remotely control and monitor the system only via the internet. Even trying to make the system work with ANN we found a lack of Dataset and models which made it impossible to work with a deep learning method.

Conclusion

In this chapter we saw the architecture of the system and its various components, the flow chart and block diagram, also the general architecture and each step of it. Clearer picture of models and sensors.

We also discussed the assumptions and hypothesis of the study without forgetting the limitations and the causes of those limitations.

In the next chapter we will discuss the implementation and more deep stuff about the system and its functionality.

CHAPTER 4

EXPERIMENT AND RESULT

Introduction

In this chapter we are going to define the objectives and requirements of the project. Besides of that we are going to define and explain most of the hardware and software we are using in this project. Along with the implementation and the algorithms, uses, results and the discussion.

Objectives

Realize a system capable of :

- auto control
- Remote monitoring
- Remote control
- Time scheduling

Requirements

The minimum requirements for this project are :

- Knowledge in C programming language.
- Nodemcu Kit or arduino Kit.
- Internet connection.
- Knowledge in Android.
- ESP8266 or other type of wifi shields.
- Cloud to store data and retrieve it.
- Knowledge in electronics.

4.1 Platform

To realize any system choosing the right platform is a must, and for our system we chose for the software part :

4.1.1 Arduino IDE

Definition

Arduino besides the boards, it provides the software to develop it. Which is the Arduino Integrated Development Environment - or Arduino Software(IDE), makes it easy to write code and upload it to the board. It runs on Windows, Mac OS X, and Linux. The environment is written in Java and based on Processing and other open-source software.**(For More Details view Annex 2)**

Why Arduino IDE

- Arduino is to easy to use
- flexible for advanced users
- Runs on Mac, Windows, and Linux
- Open source and extensible open source
- Simple, clear programming environment

4.1.2 MIT app Inventor

Definition

App Inventor allow us to develop applications for Android phones using a web browser and either a connected phone or emulator. The App Inventor servers store your work and help you keep track of your projects.**(For More Details view Annex 2)**

4.1.3 Why MIT app Inventor

- Simplicity
- Ease of setup
- Ease of programming

4.1.4 Hardware

The system is implemented on Acer TravelMate with the following specification :

- Cpu : intel(R) Core(TM) i5-3210M CPU @2.50GHz
- RAM : 4 GB
- Exploitation System : Windows 7 Ultimate Service Pack 1

The system run in a NodeMCU (Node MicroController Unit) ESP8266 WiFi System-on-a-Chip (SOC) with the next features:

- Processor: L106 32-bit RISC microprocessor core based on the Tensilica Xtensa Diamond Standard 106Micro running at 80 MHz
- Memory:
  - 32 KiB instruction RAM
  - 32 KiB instruction cache RAM
  - 80 KiB user data RAM
  - 16 KiB ETS system data RAM
- IEEE 802.11 b/g/n Wi-Fi
- 16 GPIO (General Perpose Input-Output) pins

Why Esp8266

- Low price
- Integrated Wifi module
- Integrated small processor
- Tiny size
- Supported by the Arduino community

4.2 Models & Sensors

Before going any further we will explain the models and sensors that we used. Models and sensors are linked to each other:

4.2.1 Humidity/Temperature

This sensor used to measure digital temperature and humidity DHT11, which is a calibrated measurement unit that measures both temperature and humidity. ***Figure.4.1***

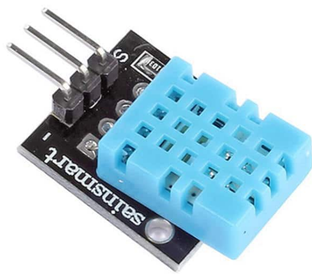


Figure 4.1: Humidity/Temperature sensor

4.2.2 Soil Moisture

It's a simple sensor to measure the availability of water in the soil, and it only needs to provided by power. ***Figure.4.2***

4.2.3 Rain sensor

This sensor, is analog liquid sensor, depends on changing the voltage coming out from it with the change of how much the liquid floods it. ***Figure.4.3***

4.2.4 Relay

Relay is an electronic switch that is used to control high voltage loads, for example 220 volts through a low-voltage 5 volt signal generated by a control panel such as the ESP8266.

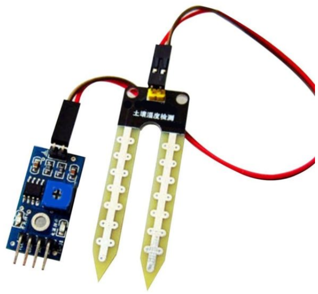


Figure 4.2: Soil moisture sensor

Figure 4.3: Rain sensor

It is usually used to control the lighting of a room or a hall, as well as in operating fans or refrigerators cooling unit if the temperature go high, or operating a water pump. ***Figure.4.4***

Figure 4.4: Electronic quadruple relay

4.2.5 I2C

Screen operator, reduces the number of legs used by the controller(ESP8266) to operate character display from 6 to 2 feet, allowing the connection of more sensors, motors and other features without the need to add a second panel.***Figure.4.5***

4.2.6 LCD

The 2004A module is designed to display letters, numbers, and symbols in a bitmap way. It can display 4 lines each containing 20 characters. These screens support data reception either

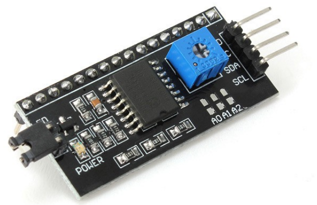


Figure 4.5: I2C

4-bit or 8-bit. It generate the characters through the memory, that has a built-in forms of characters and draw in the framework of a dot matrix size 7 * 5.

Those screens do not support the display of characters in Arabic and in the case of the need for the Arabic language must be used graphic screens instead of letters and numbers screen. ***Figure.4.6***

Figure 4.6: LCD 2004A

4.2.7 Light Sensor

light dependent resistance (LDR). Is simply a variable resistance depending on its value according to light. ***Figure.4.7***

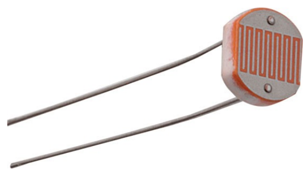


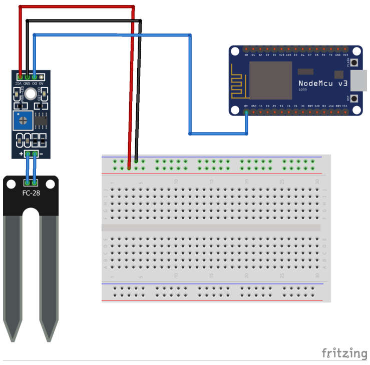
Figure 4.7: LDR

4.3 Implementation

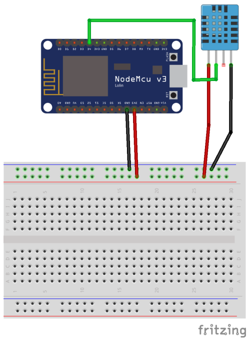
After going through the platform and defining it, now we will explain the implementation of the system. Our system contains two main parts : Automatic and manual.

4.3.1 Wiring

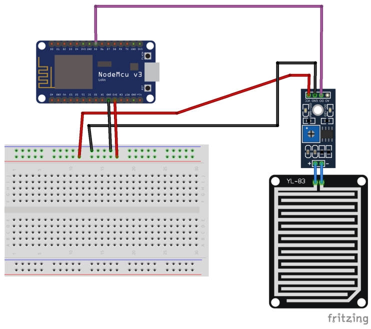
Passing to how the hardware parts are wired, the next figures will show how each sensor is wired. In those figures the wire with red color expresses the (+) and the black expresses the (-). ***Figure.4.8*** (a) shows soil moister sensor wiring, and (b) the wiring of the DHT 11, (c)rain sensor and (d) I2C LCD wiring. All of those models were made by fritzing software.



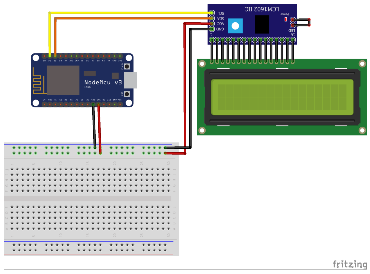
(a) Soil moister



(b) DHT 11



(c) Rain sensor



(d) I2C LCD

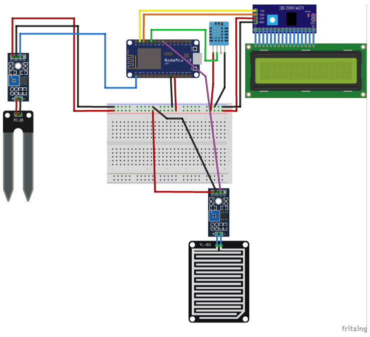
Figure 4.8: Parts wiring

The ***Figure.4.9*** shows all the parts wired up together, and the schematic of the wiring.

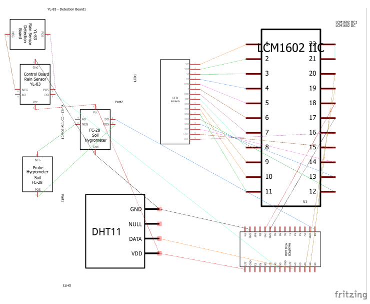
4.3.2 Automatic

The system contains some parts which are always automatic. one of these parts is perception. The perception is transmitting the data from the sensors to the system and then the system transfer it to the cloud, and the cloud send it to the mobile application and then show it to the user. The sensors are Temperature and humidity, soil moister, rain and light. Data transmitting of some sensors is shown in ***Figure.4.10, Figure.4.11, Figure.4.12***

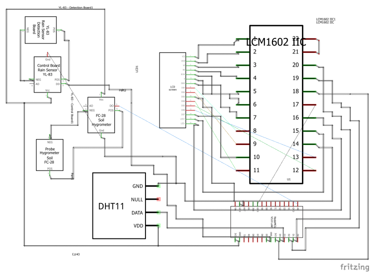
The ***Figure.4.10*** contains in (b) the declaration of the sensor with the name dht2 and Type is DHT using the library declared on (a), (c) shows the variable temperature and humidity taking



(a) System wiring



(b) System wiring schematic



(c) System wring clarified schematic

Figure 4.9: System wiring

```
#include "DHT.h"
```

(a) Library

```
DHT dht2(2,DHT11);
```

(b) Declaration

```
temperature = dht2.readTemperature();
humidity= dht2.readHumidity();

  if(!isnan(temperature) and !isnan(humidity)){
  bTemperature = temperature;
  bhumidity = humidity;}
```

(c) Data reading

Figure 4.10: Temperature and humidity sensor

the values of the sensor, then putting those values to into the new variables bTemperature and bhumidity if the sensor didn't read a nan value.

```
soilm=analogRead(A0);
```

Figure 4.11: Soil moister

Soil moister data transmitting is shown in the ***Figure.4.11*** reading the value from the pin A0 and putting it in variable soilm. The pin A0 is responsible for analog read and we chose this pin because the sensor return an analog variable from 1024 to 0.

Rain sensor is Unplugged with pin 14 which is shown in ***Figure.4.12*** (a) and declared as an input. Then raindetc variable take the value from the sensor (b). This value is digital that mean it is whether 0 or 1.

After the system get all the values from its sensors. The system have to transfer it to the cloud. The first step to do that is connecting to the internet, as shown next :

```
pinMode(14, INPUT);
```

(a) Declaration

```
raindetc=digitalRead(14);
```

(b)

Figure 4.12: Rain sensor

Firstly we have to use these libraries in Arduino IDE, **# include <ESP8266WiFi.h>, # include < ESP8266HTTPClient.h>**. The rest is shown in **Algorithm.1**

Algorithm 3: Connecting To Wifi

```
Result: Get Wifi Connection
WiFi.begin(SSID,Pasword);
while WiFi.status()! = WELL − CONNECTED do
    delay ← 300;
    write(..);
end
write(Connected);
write(WiFi.localIP());
```

The algorithm above shows how the system connect to wifi network. WIFI.Begin() is a predefined function take two variables the first is SSID and the second is the password of the network. After that we go to the while loop that take the condition if wifi status is not well connected then wait 3 seconds and write .. if the condition is not acquired then break the loop and write connected and local ip. All of these process happens in the setup procedure.

To use this next algorithm in a program in arduino IDE you need to declare this next library **# include <ThingSpeak.h>**

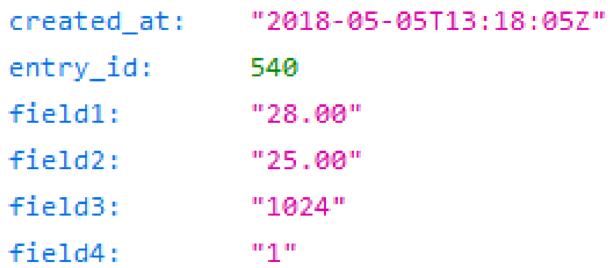
Algorithm 4: To Cloud

```
Result: Send Data to the cloud
String ThingspeakAddress, writeAPIKey, request-string;
ThingSpeak.begin(client);
if client.connect("api.thingspeak.com", 80) then
    writeAPIKey ← key;
    request − string ←
      thingSpeakAddress + key + Temperature + humidity + soilm + raindetc;
    http.begin(request − string);
    Callhttp;
end
```

This algorithm describe the process of sending data from the system to the cloud. this process need 4 string variables (ThingspeakAddress, writeAPIKey, request-string). ThingspeakAddress take the value "http://api.
thingspeak.com/update?" this value usually do not change. The second value is writeAPIKey this value change by the change of the user or the change of the channel, and it is responsible for making the writing in the cloud secured. The other one is request-string which contain the whole request, and it's combined of thing speak address and the key, the last part is the values

of the sensors it comes like that : ThingspeakAddress& writeAPIKey& field1=temperature& field2=humidity& ....field n = value n. This process happens on the loop procedure and every 30 second it is repeated.

Now that all the values are uploaded to the cloud the next and last step of this part is showing it the phone application. To download those values and show it in app we need first of all the ThingSpeakAddress and ReadAPIKey then the cloud return the results as a JSON list. ***Figure.4.13***



created_at: "2018-05-05T13:18:05Z"
entry_id: 540
field1: "28.00"
field2: "25.00"
field3: "1024"
field4: "1"

Figure 4.13: JSON List

Algorithm 5: Cloud to phone App

Result: Download Data from the cloud
String ThingspeakAddress, readAPIKey, results;
Float temperature, humidity, soil, rain, light;
List request;
Timer clock;
if *clock* **then**
    $Web \leftarrow ThingspeakAddress + readAPIKey + results$;
    Call Web;
end
if *Web.gettext()* **then**
    $request \leftarrow web.response$;
    $temperature \leftarrow request.segment("field1", 2)$;
    $humidity \leftarrow request.segment("field2", 2)$;
    $soil \leftarrow request.segment("field3", 4)$;
    $rain \leftarrow request.segment("field4", 1)$;
    $light \leftarrow request.segment("field5", 1)$;
end
Display(temperature,humidity,soil,rain,light);

As shown above to download the values from the cloud we need the variables (String ThingspeakAddress, readAPIKey, results ...Float temperature, humidity, soil, rain, light.......List request......Timer clock). ThingSpeakAddress as explained before take the value https://api.thingspeak.com/channels/457187/feeds.json?, the next string take readAPIKey which is changeable by the change of the channel or the user same as the write key, the last string is result which take a number "1", the number define how much results you want to get from that field. For example : 1 means the last result, 2 mean the last and the lasts result and so on.

Each of the float variables take its value from the request list segmentation. List request take the response from the cloud which come as an JSON list, the list get segmented by field names and the length of the value. After all of that is done the values get displayed on the phone screen. This process happens in a loop that replay every 10 seconds and that is the value of the clock.

The explained part above is an automatic processor that the user can't stop or interfere on its work. Also we have another part that enable the user to decide whether letting the system completely automatic or stop it.

The button manual is responsible if the system to be automatic or manual, and each time the button is clicked it get disabled and a counter of 30 second is lunched to return it enable again. Once the system is set automatic it start to irrigate automatically based on the next variables we are going to explain.

The automation of our system is represented in a reactive agent that take its situation and compare it with the RuleBase, the RuleBase is composed of different rules. Each time the comparison of situation and RuleBase produces a match rule to the situation, the agent offers an action that is suitable to the situation.

One of this variables that the irrigation is based on is the situation list. This list get filled with the information from two sources, First the database the other is the sensors it was more detailed in **Chapter 3**. The information filling is done by **Algorithm 4**.

Algorithm 6: Situation

```
Result: Get a situation
List Situation;
Integer i=0;
while i < 11 do
    Situation ← 0;
end
i ← 0;
if i == 0 then
    if ismonth() then
        Situation ← 1;
    end
end
i++;
if i == 1 then
    if isday() then
        Situation ← 1;
    end
end
i++;
.......;
..........;
if i == 9 then
    if israining() then
        Situation ← 1;
    end
end
i++;
if i == 10 then
    if islight() then
        Situation ← 1;
    end
end
```

The second variable that the agent depend on is RuleBase. The RuleBase is a set of rules that is generated in the start of the system in the setup procedure**Algorithm 7**.

Algorithm 7: Rule Base

```
Result: Get a Rule Base
List Rule, RuleBase;
Integer i=0,j=0;
if j == 0 then
    while i < 11 do
        Rule ← 0;
    end
    Rule[0] ← 1 Rule[1] ← 1 Rule[2] ← 1;
    Rule[3] ← 1 Rule[4] ← 1 Rule[6] ← 1;
    j++;
end
if j == 1 then
    while i < 11 do
        Rule ← 0;
    end
    Rule[0] ← 1 Rule[1] ← 1 Rule[2] ← 1;
    Rule[3] ← 1 Rule[4] ← 1 Rule[7] ← 1;
    j++;
end
............;
............;
if j == 22 then
    while i < 11 do
        Rule ← 0;
    end
    Rule[0] ← 1 Rule[1] ← 1 Rule[2] ← 1;
    Rule[3] ← 1 Rule[5] ← 1 Rule[7] ← 1;
    Rule[10] ← 1;
    j++;
end
if j == 23 then
    while i < 11 do
        Rule ← 0;
    end
    Rule[0] ← 1 Rule[1] ← 1 Rule[2] ← 1;
    Rule[3] ← 1 Rule[5] ← 1 Rule[8] ← 1;
    Rule[10] ← 1;
    j++;
end
```

Those rules that makes up the RuleBase is taken for real life situation shown in **Table 4.1**.

In the **Table 4.1** we shown some rules and the action generated from it. But the action comes as a result of the comparison step when we search for a match for the situation in the RuleBase. The algorithm of that step **Algorithm 8**

Rule	Action
if ismonth() and isday() and ishour and isminute and temperaturehigh() and soildry() and !israin() and atnight()	912 minute
if ismonth() and isday() and ishour and isminute and temperaturehigh() and soilbitwet() and !israin() and atnight()	713 ms
if ismonth() and isday() and ishour and isminute and temperaturehigh() and soildry() and israin() and atday()	no action
if ismonth() and isday() and ishour and isminute and temperaturehigh() and soilbitwet() and israin() and atday()	no action

Table 4.1: Rule-Action

Algorithm 8: Comparing rule and situation

```
Result: chose rule
List situation, compare,rulebase;
Integer i=0,j=0,chosenrule=-1;
while j < 24 do
    while i < 11 do
        compare[i] ← rulebase[i][j];
    end
    if array-comp(situation,compare,11,11) then
        chosenrule ← j;
        break;
    end
end
```

In the algorithm 8 we have three lists one for situation and another one for rulebase and the last that take the value of rule "j". The compare list take in every iteration the value of the rule in position j in the rule base and be compared with the situation in array-comp. Array-comp is a predefined function that take the value of two lists and they're size then return true if they are matched. If the function return false we continue searching, but once it return true then the variable chosenrule takes the value "j" which is the rule position and then break the loop.

After we chose the position we drop it on the action-base to return the suitable choice. As shown in **Algorithm 9**

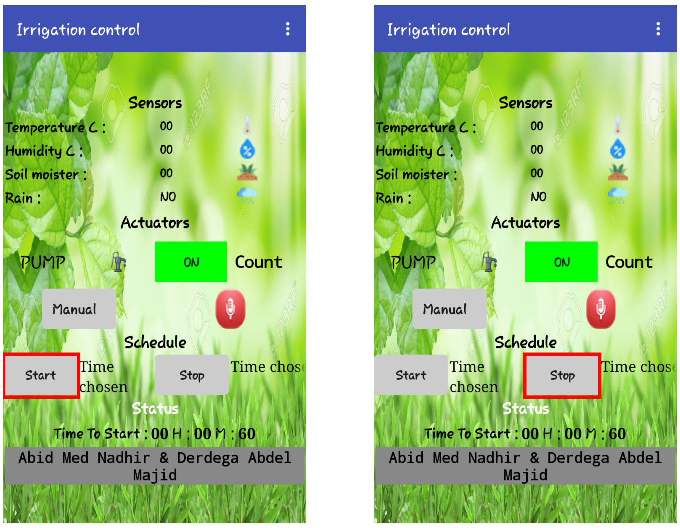
Algorithm 9: Action

```
Result: Do Action
List Situation;
Integer chosenrule=-1;
if chosenrule == 0 then
    Irrigate 912 minute;
end
if i == 1 then
    Irrigate 713 minute;
end
.......;
..........;
```

The generated action is a time of how long to irrigate and it is a conclusion of all the last steps. Beginning from generating RuleBase (RB) to gathering the information from sensors to form a situation and then compare it with RB to get the suitable action. All of this steps can be turned off by the user as mentioned before by a button, leading us to the other part of the system.

4.3.3 Manual

This part of the system is when the user can do whatever he wants from the start time of irrigation to the end time of irrigation meaning that the user can control how long the irrigation can be. ***Figure.4.14***



(a) Start irrigation (b) Stop irrigation

Figure 4.14: Manual part

When choosing a time of start the system calculates the remaining time to start compared with phone time and shows it in the to the user, and then send the start time to the cloud and the agent download it from the cloud ***Figure.4.15***. Also choosing the end time of irrigation gets the same treatment is uploading it to the cloud. Then the system calculates the difference between the end and the start showing the duration of the irrigation.

```
starthour = (ThingSpeak.readIntField(465589,1,"Y7R5G6XYC9WQKMAV"));
startminute = (ThingSpeak.readIntField(465589,2,"Y7R5G6XYC9WQKMAV"));
endhour = (ThingSpeak.readIntField(465592,1,"O69VU4ASQKDT92NS"));
endminute = (ThingSpeak.readIntField(465592,2,"O69VU4ASQKDT92NS"));
```

Figure 4.15: The Agent Download the start and end time from the cloud

4.4 Uses

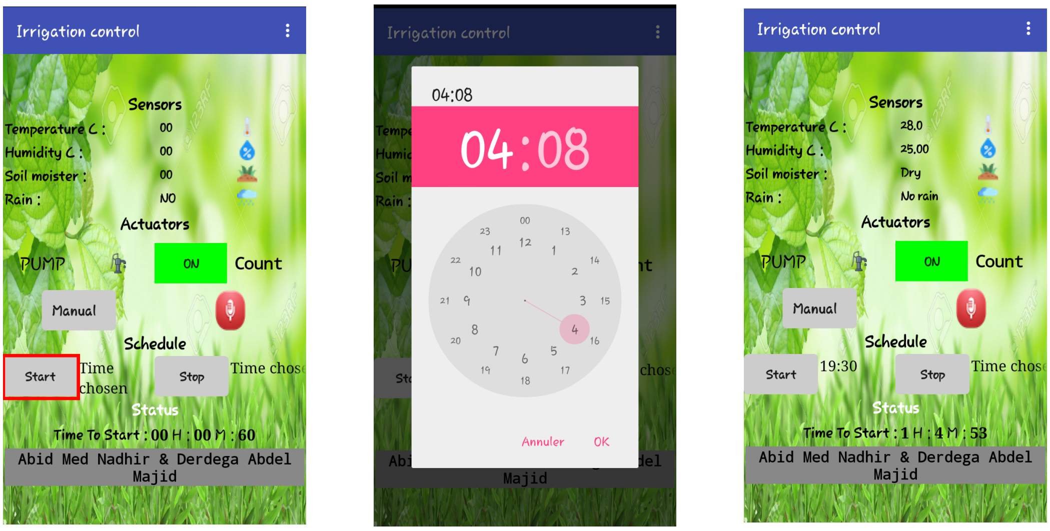
Our system is easy to use and simple. All the needs are power plugin and the internet connection. In uses we have two parts. The first is the automatic and it is more easy and needs no human interaction. The second is the manual which is based on human interaction and need some configuration.

4.4.1 Automatic

The automatic processing is on by default and as mentioned before it do all the work from perception to calculating ending with execution. The only thing that the system need in this stage is the internet connection and power supply. If the user wanted to disable the automatic part there is a button to switch off between the manual and a automatic as mentioned before.

4.4.2 Manual

This part is not 100% manual because the perception part is always on and it's automatic. The manual is the start time of irrigation and the end time, that is chosen by the user. The work is demonstrated in ***Figure.4.16***.



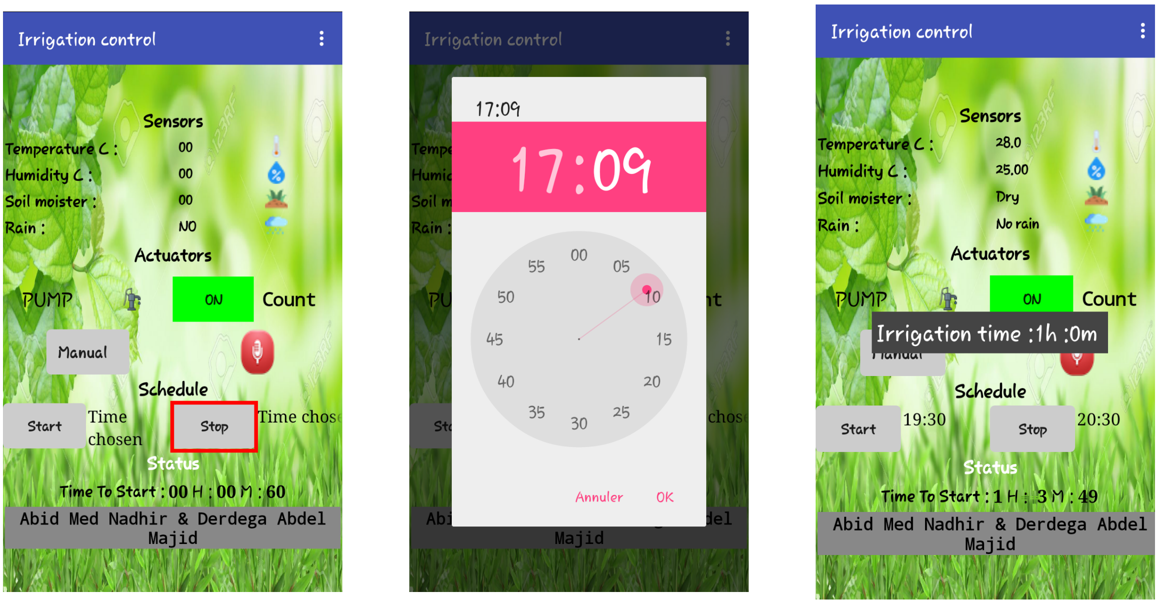
(a) Start irrigation (b) Start time (c) Selected Time

Figure 4.16: Manual part

In the ***figure.4.16*** in (a) the user clicks on the button start and a clock shows up (b) to chose the time, after that he clicks ok to confirm the chosen time. After that the time chosen appears on the label besides the button as shown in ***Figure.4.16* (c)**.

The label in the bottom of the screen with the title "time to start" shows the count down between the chosen time and the phone time. ***Figure.4.16 (c)***.

To choose the end time you need to click in the end button as shown in ***Figure.4.17* (a)**. Then a window with a clock pops up giving you the right to chose a time also shown in ***Figure.4.17* (b)** .



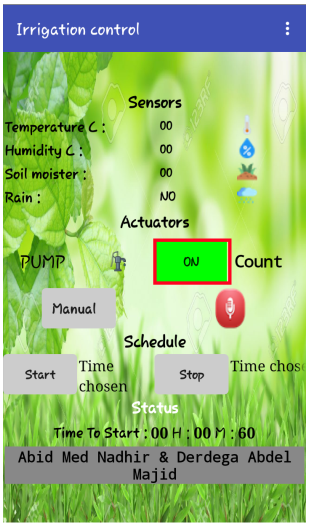
(a) Stop irrigation (b) Stop time (c) End Time

Figure 4.17: Manual part

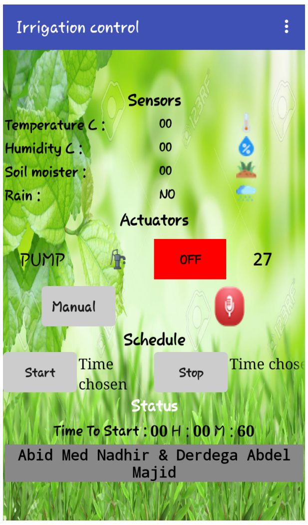
After the confirmation with the ok button a notifier shows up with the duration of the irrigation selected by the user. The label "time chosen" shows the chosen time.

Another function that the user can do it manually is turning on and off the pump without time scheduling ***Figure.4.18***

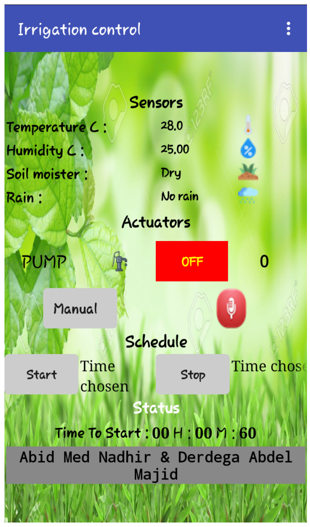
Once the button clicked it turn the pump on and lunch a counter of 30 second, in those 30 seconds the button will be disabled once it end the button will return enabled and the text will turn yellow. The same thing happen in off situation. ***Figure.4.18***



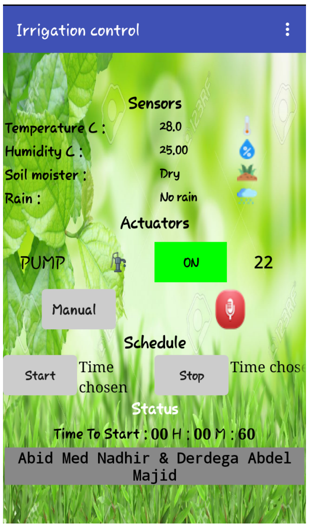
(a) On/Off button



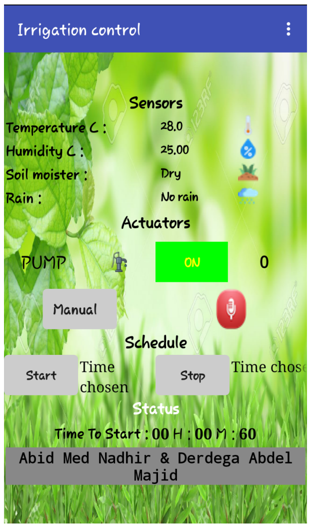
(b) ON clicked



(c) Timer done



(d) OFF clicked

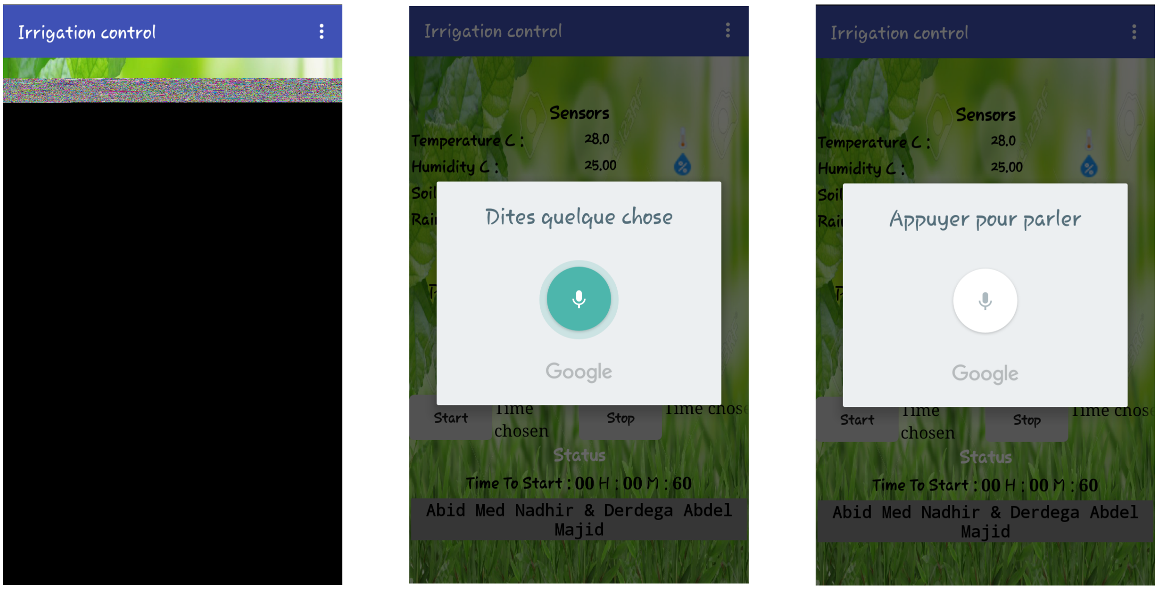


(e) Timer done

Figure 4.18: On/Off situations

In the same subject which is the manual things that the user can do. Now we are talking about the same function and it is setting the pump ON/OFF but in another way a more creative way and it is with voice recognizer.***Figure.4.19***.

The next figures shows how the voice recognizer works and what windows pops up and the steps.



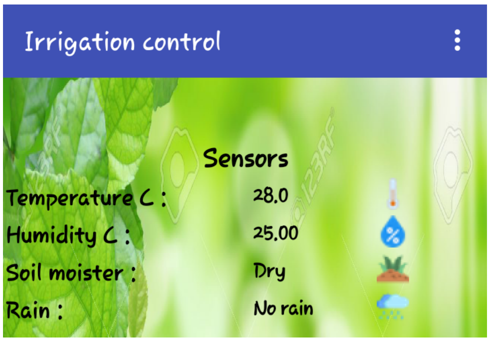
(a) interacting (b) Talking (c) After Talking

Figure 4.19: Speech Recognizer

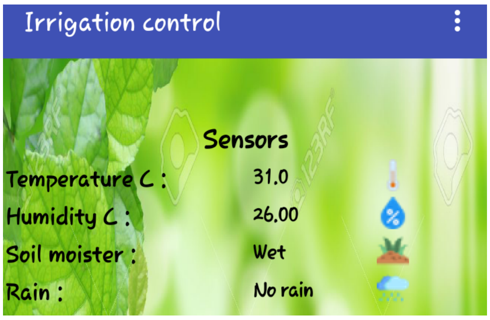
4.5 Result

The desired objectives were to implement an automated system that provide a real time monitoring and control and do provide also a time scheduling.

The real time monitoring is presented in the sensors block in the app interface as shown in ***Figure.4.16*** , and the real time control was shown in the sections above with time scheduling.



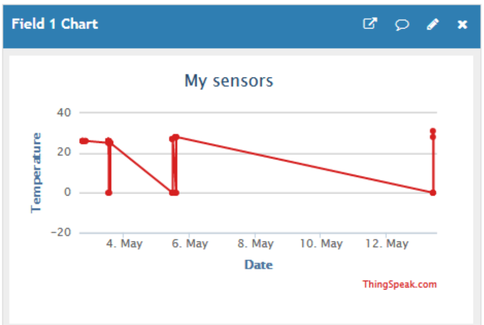
(a) Before



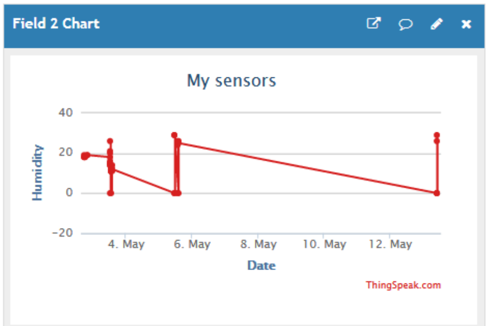
(b) After

Figure 4.20: Real Time situation

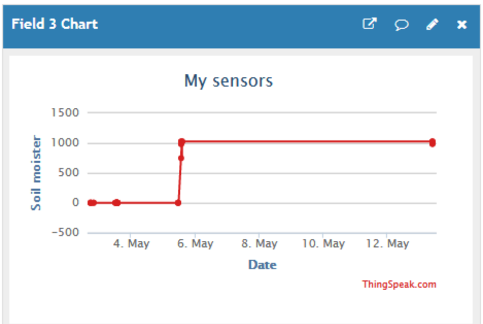
The values shwon are uploaded to the cloud and represented there as graphs and charts, each of those values is represented in a field as shown in ***Figure 4.17***



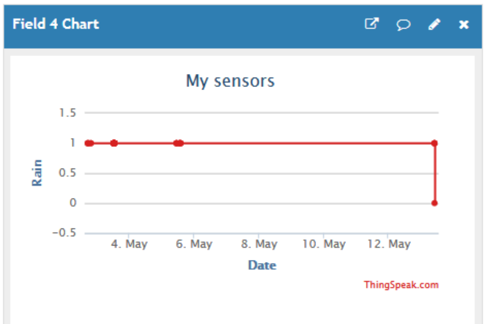
(a) Temperature



(b) Humidity



(c) Soil



(d) Rain

Figure 4.21: Cloud Charts

Those values change every 30 seconds from the system and the phone app download from the cloud every 30 seconds.

All the previous explanation serves the software part. Now for the hardware part we got also a satisfied results. We implemented a system with different sensors, and the capacity to power on and off an irrigation pump based on the weather condition, and predefined times in a database. ***Figure.4.18***

(a) Outside (b) Inside

Figure 4.22: Hardware

4.6 Discussion

We recorded a good results and many advantages, among them :

- Real time monitoring
- Real time control
- The ability to change from manual to automatic
- Time scheduling
- Mobility
- Low price
- Adaptability
- precision in time
- precision in the environment variables
- Water saving

A lot of other advantages, with all of these good results and advantages. This does not prevent the system to have some inconvenient that was a result of time limitation a camera placed on leaves to detect the diseases of the plant. After the disease is detect the needed antidote will be injected in the water applied. The most negative thing that we faced was the miss of a dataset that belong to the region.

Conclusion

In this chapter we discussed the platform used and the reason of choosing it. After that we did go through the implementation with both of it's part automatic and manual. Describing the algorithms we worked with.

Passing by the uses to the result ending with discussion, clarifying some of the advantages and inconveniences of the implemented system.

GENERAL CONCLUSION

In this thesis we tried to solve irrigation problem using the latest technologies like IoT and Cloud Computing. The objective of this system is to reduce the hard work of the farmer. The other objective is to reduce water consumption and to achieve a better yield.

The system uses an agent that has a rule base and acts based on different information. One of those information is the Database that contains the dates of the irrigation. The other is the environment perception, temperature, humidity and soil moister. After the intelligent agent gets the information, the data processing step that produces a situation. Then the situation is compared with Rule Base to produce a rule, and then the rule results an action.

The system implementation was made with Arduino IDE and MIT APP INVENTOR for the android application and NodeMcu ESP 8266 and set of sensors. It can be divided into two parts the android application which does most of the remote work, and the other part that do most of the automatic work, and an automatic part.

We achieved a system capable of :

- Remote control
- Remote Monitoring
- Automatic irrigation
- Time scheduling
- Real time monitoring

What distinguishes our system from other systems is the low cost and automation plus the knowledge of months and days of irrigation. An other advantage is the simplicity and performance in real time.

Walking through the different chapters we described and clarified the system architecture, faced limitations, some hypothesis and achieved objectives.

PERSPECTIVES

In the future work, we will try to strengthen our system with a better dataset and apply a deep learning method, because it was the big obstruction that prevented us from creating a better system along with time constraint. We will add a camera that detect the diseases from the leaves of the plant along with the type of the plant and based on the information that the system gather from the camera it injects the antidote without the need of any human interaction.

BIBLIOGRAPHY

[1] *http://www.pit.ac.in/file/elearning/CS6659-ARTIFICIAL%20INTELLIGENCE.pdf*. 28/1/2018.

[2] Stuart J.Russell and Peter Norvig. *Artificial Intelligence A Modern Approach Third Edition*. Pearson Higher Education, 1995.

[3] Anitha K. *AUTOMATIC IRRIGATION SYSTEM*. Pearson Higher Education, 2016.

[4] G. L. Wolflin. *SOME HISTORY OF EARLIER AUTOMATED IRRIGATION*. Pearson Higher Education, 1968.

[5] K.Sathish Kannan et al. *Online Farming Based On Embedded systems and Wireless Sensor Networks*. 2013.

[6] Ameya Bhale et al. *IoT Based Automatic Drip Irrigation System*. 2014.

[7] John McCarthy and Patrick J. Hayes. *SOME PHILOSOPHICAL PROBLEMS FROM THE STANDPOINT OF ARTIFICIAL INTELLIGENCE*. 1969.

[8] Chris Smith & Brian McGuire. *The History of Artificial Intelligence*. 2006.

[9] Ting Huang Chris Smith, Brian McGuire. The history of artificial intelligence. 2016.

[10] Wooldridge Michael and Nicholas R.Jennings. *Agent Theories, Architectures and Languages: A Survey*.

[11] *http://www.ee.mcgill.ca/ belmarc/agent-faq.html*. 28/1/2018.

[12] *https://people.eecs.berkeley.edu/ russell*. 30/01/2018.

[13] Stuart J. Russell and Peter Norvig. *Artificial Intelligence A Modern Approach Third Edition*. 2010.

[14] Siddharth Tripathi Somayya Madakam, R.Ramaswamy. Internet of things (iot): A literature review. *Journal of Computer and Communications*, 3, 2015.

[15] Ovidiu Vermesan and Peter Fries. *Internet of Things From Research and Innovation to Market Deployment*. River Publishers, 2014.

[16] Tomas Robles and Ramon Alcarria. *An IoT based reference architecture for smart water management processes*. Pearson Higher Education, 2008.

[17] Jim Buchan Michelle Selinger, Ana Sepulveda. Education and the internet of everything. *Cisco*, 2013.

[18] Antonio F. Skarmeta M.Victoria Moreno, Benito beda. How can we tackle energy efficiency in iot based smart buildings? *www.mdpi.com/journal/sensors*, 14, 2014.

[19] DELL. Connected health. *Dell.com/healthcare*, 2016.

[20] Jiong Jina and Jayavardhana Gubbib. *An Information Framework of Creating Smart City through Internet of Things.* Pearson Higher Education, 2013.

[21] Rolland Vida. Iot in smart cities. *IEEE Sensors Council, IEEE Communications Society*, 2016.

[22] Valerio Persico Alessio Botta, Walter de Donato. On the integration of cloud computing and internet of things.

[23] Danish Inamdar1 and Akash Patil2. *Automated Drip Irrigation System based on Embedded System and GSM Network.* Pearson Higher Education, 2016.

[24] Rekha B Venkatapur1 and S Nikitha2. *Review on Closed Loop Automated Irrigation System.* Pearson Higher Education, 2017.

[25] Joaquin Gutirrez and Juan Francisco. *DESIGN and IMPLEMENTATION OF AUTOMATIC IRRIGATION SYSTEM USING WIRELESS SENSOR NETWORK and ZIGBEE MODULE.* Pearson Higher Education, 2014.

[26] Pravina B et al. *An Automatic Irrigation System using ZigBee in Wireless Sensor Network.* 2015.

[27] Ibrahim Mat et al. *Precision Agriculture Applications using Wireless Moisture Sensor Network.* 2015.

[28] Gaurav Soni et al. *Automatic Irrigation System Using Wireless Sensor Networks.* 2015.

[29] A.Sathya et al. *Automatic Control of Irrigation System in Paddy Using WSN.* 2016.

[30] Kaviyanand et al. *Smart Drip Irrigation System for Sustainable Agriculture.* 2016.

[31] J.W. Rittinghouse and J.F. Ransome. *Cloud Computing: Implementation, Management, and Security.* CRC Press, 2017.

[32] Torry Harris. Cloud computing an overview.

[33] Ben Kepes. Understanding the cloud computing stack saas, paas, iaas. 2011.

[34] Eugene Gorelik. Cloud computing models. 2013.

[35] Grance Timothy Peter Mell. The nist definition of cloud. 2011.

[36] Bheemashankar Babanna Kumbar, Basavaraj Galagi. Smart irrigation system using internet of things. 2016.

[37] Shikha1 and Vibha2. *Automated Irrigation System using ZigBee - GSM.* Pearson Higher Education, 2016.

[38] M. Kranthi Kumar and K. Srenivasa Ravi. *Automation of Irrigation System based on Wi-Fi Technology and IOT.* Pearson Higher Education, 2016.

[39] AHMED ABDULLAH ALI SHAREEF1 and S. MANASSEH2. *Design and Development Automatic Irrigation System using Wireless Sensor Network and Raspberry Pi.* Pearson Higher Education, 2016.

[40] ARCHANA P and PRIYA R. *DESIGN AND IMPLEMENTATION OF AUTOMATIC PLANT WATERING SYSTEM.* Pearson Higher Education, 2016.

[41] Pramod V Deshmukh and Prashant R Deshmukh. *Architecture for Automated Irrigation System.* Pearson Higher Education, 2016.

[42] C. Kvien and C. Bednarz. *A real-time wireless smart sensor array for scheduling irrigation.* Pearson Higher Education, 2007.

[43] Mahir Dursun* and Semih Ozden. *A wireless application of drip irrigation automation supported by soil moisture sensors.* Pearson Higher Education, 2011.

[44] Yuthika Shekhar and Ekta Dagur. *Intelligent IoT Based Automated Irrigation System.* Pearson Higher Education, 2017.

[45] S.G.Manoj Guru and P.Naveen. *SMART IRRIGATION SYSTEM USING ARDUINO.* Pearson Higher Education, 2017.

[46] IG Amend. *Irrigation System Design.* Pearson Higher Education, 2005.

[47] Mr. Babanna Kumbar and Mr. Basavaraj Galagi. *SMART IRRIGATION SYSTEM USING INTERNET OF THINGS.* Pearson Higher Education, 2016.

[48] Animesh Mathur and Ajinkya Fotedar. *Automated Irrigation System using MSP430.* Pearson Higher Education.

[49] Constantinos Marios and Sotiris Nikoletseas. *A Smart System for Garden Watering using Wireless Sensor Networks.* Pearson Higher Education, 2011.