UNIVERSITY of RWANDA

COLLEGE OF SCIENCE AND TECHNOLOGY

Website: www.aceiot.ur.ac.rw
Mail: aceiot@ur.ac.rw

"A FUZZY INFERENCE MODEL FOR IOT SHIITAKE MUSHROOM FARM MONITORING AND CONTROL"

Name: Jean De La Croix Ntivuguruzwa

Reference Number: 220014132

College of Science and Technology

African Center of Excellence in Internet of Things

Master of Science in Internet of Things – Embedded Computing Systems

2019 - 2021

Submission Date: December 2021

UNIVERSITY of RWANDA

COLLEGE OF SCIENCE AND TECHNOLOGY

ACEIoT

Website: www.aceiot.ur.ac.rw
Mail: aceiot@ur.ac.rw

Thesis Title:

"A Fuzzy Inference Model for IoT Shiitake Mushroom Farm Monitoring and Control"

By

Name: Jean De La Croix Ntivuguruzwa

Reference Number: 220014132

A dissertation submitted in partial fulfilment of the requirements for the degree of Master of Science in Internet of Things with specialization in Embedded Computing Systems in College of Science and Technology, University of Rwanda.

Supervisor : Dr. Didacienne MUKANYILIGIRA

Co-Supervisor: Dr. Louis SIBOMANA

Declaration

I hereby declare that all information in this document is original and has never been presented in any University or other Institutions of Higher Learning. I also declare that, as required by rules I have fully cited and referenced all material and results that are not original to this work.

Student Name and Number

Jean De La Croix Ntivuguruzwa
220014132

Signature:

Date: December 27, 2021

Bonafide Certificate

This is to certify that the dissertation report work titled "**A Fuzzy Inference Model for IoT Shiitake Mushroom Farm Monitoring and Control**" is a record of original work done by Mr. Jean De La Croix Ntivuguruzwa a post-graduate student in MSc in the Internet of Things with a specialization in Embedded Computing Systems(ECS) at the University of Rwanda, College of Science and Technology in African Center of Excellence in the Internet of Things (ACEIoT), We certify that the work reported does not form a part of any other research project.

Supervisors,

Dr. Didacienne MUKANYILIGIRA	**Dr. Louis SIBOMANA**
Signature:	**Signature:**
Date: December 27, 2021	**Date: December 27, 2021**

Head of Masters and Training ACEIoT,

Dr. James Rwigema

Signature:

Date:

Acknowledgments

First of all, I would like to thank the Almighty God, who gave me the grace to accomplish this work. I would like to express my sincere and special thanks to my supervisors Dr. Didacienne Mukanyiligira, and Dr. Louis Sibomana, for their great guidance and supervision in the achievement of this work from which I gained invaluable knowledge. Secondly, I wish to thank the African Center of Excellence in the Internet of Things, College of Science and Technology, University of Rwanda. I also wish to thank my family members who have done great sacrifices by providing me with all the necessary means to have a good education. Thirdly, I thank my classmates, especially those who were together in groups in all academic activities for their direct and indirect contribution to the achievement of my objectives. Lastly, my appreciation goes to the lectures on the Internet of Things for the good job done during the 2 years of our courses. My acknowledgment goes to everyone who supported me in my everyday life to reach this accomplishment.

May the Almighty God bless you all!

Abstract

Mushrooms are considered as one of the main sources of both medicinal and culinary products. Growing mushrooms by watering have been taken into consideration with attention as an efficient way of producing the desired harvest in them. Shiitake mushroom (Lentinula edodes) is the most important culinary medicinal mushroom which ranks at number two in terms of total mushroom production in the world only next to button mushroom. Shiitake is beneficial for soothing bronchial inflammation and regulating urine incontinence as well as for reducing chronic high cholesterol, therefore shiitake mushrooms need to be scientifically increased in the harvest. However, the production in mushrooms is less because of traditional methods such as manual greenhouse for mushrooms, mushrooms' cultivation on tree logs, fixed time-based watering method, that are implemented with manual approaches, and thus present a gap in preciseness to manage vital parameters of the mushrooms.

Internet of things (IoT) based smart farming system is an agricultural solution that is built for monitoring the crops growing medium. This involves IoT-based technologies to enable farmers to optimize the usage of agricultural resources such as water, electricity, fertilizers, and others to enhance crop productivity.

Within the previous researches, the use of sensors in mushrooms cultivation has not sufficiently provided optimized harvest. Thus, there is still a need to improve the existing methods of mushrooms cultivation by using a real-time surveillance system of the mushrooms' farm vital needs quantity variation to support in decision making.

This research aims at introducing an automated system with intelligence based on fuzzy logic to maintain mushrooms substrate suitable for shiitake mushrooms and to save the quantity of water used in mushroom farming.

Within this research, by using fuzzy logic we avail an IoT-based system for shiitake mushrooms farming that keeps the farm's air humidity, substrate moisture, and humidity to a wanted level for the sake of improved harvest in shiitake mushrooms.

Keywords: Mushrooms farming, Shiitake mushroom, Fuzzy logic, Internet of Things (IoT).

List of acronyms

ACEIoT	:	African Centre of Excellence in the Internet of Things
API	:	Application Programming Interface
DHT	:	Digital Humidity and Temperature
ESP	:	Espressif
FL	:	Fuzzy Logic
GUI	:	Graphical User Interface
IDE	:	Integrated Development Environment
IoT	:	Internet of Things
MCU	:	Microcontroller Unit
PA	:	Precision Agriculture
PF	:	Precision Farming
SA	:	Smart Agriculture
WSN	:	Wireless Sensor Network

List of Figures

Figure 1: 12V DC water pump [15]..11
Figure 2: Relay Pinout [17] 11
Figure 3:NodeMCU ESP8266 Pinout [23]..12
Figure 4: DHT11 Sensor Pinout [24]. 12
Figure 5: Capacitive soil moisture sensor V1.2 Pinout [24]...13
Figure 6: Photoresistor Sensor (LDR) [26] 13
Figure 7: PIR (passive infrared) sensor Pinout [26]. 14
Figure 8: Research Approach 15
Figure 9: System development model adapted from [22] 16
Figure 10: Layers of the proposed system adapted from [22]..17
Figure 11: Water pump automated control flowchart 18
Figure 12: Fuzzy Inference system flowchart 19
Figure 13: Components of the Proposed system 20
Figure 14: Fuzzy logic designer for Mushroom monitoring 22
Figure 15:Fuzzy Rules viewer 24
Figure 16:Temperature input for fuzzy design 25
Figure 17: Soil moisture input for Fuzzy design 26
Figure 18: Humidity input for Fuzzy design 27
Figure 19: Fuzzy output variable-watering time 28
Figure 20: On-device performance specifications..29
Figure 21: Temperature and humidity Serial Output...31
Figure 22: Soil moisture Serial Output...32
Figure 23: PIR Serial output..33
Figure 24: Output of the LDR 34
Figure 25: Thingspeak visualization...35
Figure 26: Sensors fusion Serial Output..36
Figure 27: Shiitake Mushroom farm monitored with IoT 38
Figure 28: Shiitake Mushroom farm based on time-interval watering.....................................39

List of Tables

Table 1: Comparison between IoT platforms [23] --30
Table 2: Comparison of the harvest between traditional and IoT based means ------------------------------37

Table of contents

Declaration i
Bonafide Certificate ii
Acknowledgments iii
Abstract iv
List of acronyms v
List of Figures vi
List of Tables vii
Table of contents viii
CHAPTER ONE: GENERAL INTRODUCTION 1
1.1 Overview and Background 1
1.2 Problem statement 3
1.3 Objectives of the research 4
1.4 Hypotheses of the research 4
1.5 Scope of the research 4
1.6 Benefits of the research 4
1.7 Contribution of the research 5
1.8 Organization of the research report 5
CHAPTER TWO: LITERATURE REVIEW 6
2.1 Introduction 6
2.2 Review of related works 6
2.3 Summary on gap identification and our scientific contribution 9
2.3.1 Weaknesses identified 9
2.3.2 Scientific contribution 10
2.4 Sensors and tools 10
2.4.1 Water pump (12V DC water pump) 10
2.4.2 Relay 11
2.4.3 Microcontroller 11
2.4.4 Temperature sensor (DHT11) 12
2.4.5 Moisture sensor (Capacitive soil moisture sensor) 13
2.4.6 Light sensor (Photoresistor Sensor (LDR)) 13
2.4.7 Proximity sensor (PIR (passive infrared sensor)) 14
CHAPTER THREE: METHODOLOGY 15

3.1 Introduction 15
3.2 Research and design approach 15
3.3 Scientific research approach 16
3.4 System Development 16
3.5 Components of the proposed system 17
3.5.1 Logical components 17
Source: Own drawing 18
3.5.2 Physical components 19
3.6 Hardware components selection 20
3.6.1 Microcontroller Selection 20
3.6.2 Sensing side components selection 21
3.6.3 Actuating side components selection 21
3.7 Software components selection 22
3.8 Cloud platform selection 29
3.9 Integrated Development Environment Selection 30
CHAPTER FOUR: SYSTEM RESULTS AND DISCUSSION 31
4.1 Introduction 31
4.2 Results of subsystems 31
4.3 Results of the system 36
4.4 Shiitake mushroom farms after 21 days 38
CHAPTER FIVE: CONCLUSION AND RECOMMENDATIONS 40
5.1 Conclusion 40
5.2 Recommendations 41
References 42
Appendix: Codes for the project Implementation 44

CHAPTER ONE: GENERAL INTRODUCTION

1.1 Overview and Background

Wireless Sensor Network (WSN) technology evolved significantly which made a rapid growth of the introduction of Internet of things (IoT) technology in various sectors including agricultural applications [1], [2]. Due to the pervasiveness and ubiquitous nature of IoT technology, it is enabled to be a favorable tool for agriculture automated with preciseness feature [3],[4]. Precision Farming (PF) and Precision Agriculture (PA), and Smart Agriculture (SA) are newly introduced terminologies for the advancement of agricultural technology even though their similarities are close but they are differently implemented [5].

In the field of PF, mushrooms farming which is also called fungi culture is defined as a process to produce mushrooms on your own either at home or in a small-scale farming substrate [6]. Even though mushrooms strictly impose a specific substrate in terms of humidity, temperature, a light intensity, producing a viable harvest can be achieved by taking critical mushroom breeding steps and by varying the types of seeds and various growing methods such as growing mushrooms on logs, growing mushrooms on spawn, and growing mushrooms on straw [6]. Differently to most agricultural species, mushrooms specifically, must be grown in a particular medium with a restriction to not being exposed to the light to produce the best possible harvest. Traditionally, mushrooms have been bred in the wild and picked from there [7].

The mushroom need has been increased due to their use in medicinal and culinary practices. Besides, mushrooms are considered to be one of the good cash crops. Farmers do not cultivate them at a large scale because they require a lot of manpower for monitoring the farms, and sometimes it becomes impossible to introduce some crucial types of mushrooms [8].

Because mushrooms are of capital importance for health (healthy nutrients and medicinal applications) as of [9], mushroom farming was chosen in this work to contribute to the improvement of the yield of mushrooms, to support the availability of medical ingredients and food.
Specifically, our research considers the farming of shiitake mushrooms, which can generate income, diversify farms, culinary and pharmaceutical enterprises, add value to medicinal products, create opportunities for farmers and researchers in medical products laboratories.

Shiitake mushroom cultivation originated in Japan thousands of years back, and people have been growing them on downed shii trees which is even the origin of its name [10]. Shiitake have not been introduced quickly in many countries, for instance, the U.S. started growing them in the early 1980s, and researches have not often focused on shiitake mushroom cultivation, to identify how to monitor the substrate, seasons, laying yards for this type of mushroom which is important in boosting human immune system [10].

In African countries, shiitake mushroom is cultivated but with less yield. Among those countries, we can cite South Africa, Egypt, Tunisia, and Rwanda some but very few farmers try shiitake types of mushrooms. Thus, the type of mushroom requires specific attention because of its various applications that are important for society in general. Shiitake is used medicinally for diseases involving depressed immune function including cancer, AIDS, environmental allergies, Candida infections, and frequent flu and colds. Shiitake is also beneficial for soothing bronchial inflammation and regulating urine incontinence as well as for reducing chronic high cholesterol. Lentinan (a cell-wall constituent extracted from the fruiting bodies of shiitake) is an immunomodulating agent which may be useful both as a general rejuvenate for older persons, as well as prophylactically to protect healthy, physically active young people from overwork and exhaustion [10].

This research focuses on how to maintain a favorable environment for shiitake mushroom farms by availing real-time surveillance of the farm temperature, light, air movement speed, soil humidity, and insect prevention to increase the yield. This system works on a real-time basis to identify any critical variation of the farm environment at any time to take action on addressing it on time. To analyze and decide on the data collected by sensors, we use a knowledge base (database) storing the standard data about the desired farm environment parameters for shiitake and by benchmarking the collected data with the desired ones, once a variation is out of the desired range a decision to re-establish them to the desired ones is taken. To make that possible, we use a fuzzy-inference-based model. Fuzzy Logic (FL) is based on an approach of imitating human decision-making by including all in-between alternatives of digital values YES and NO. This model is one of the most used algorithms for decision-making systems of artificial intelligence [11].

This research employs Fuzzy Logic to build a corresponding fuzzy inference system for surveillance of mushroom farm' vital conditions (temperature, light, and soil humidity) with real-time decision making on the action to maintain them in the desired range.

1.2 Problem statement

Mushroom growing by water used to monitor the farm has been considered as the only sustainable and reproducible approach to efficiently produce mushrooms for both culinary and medicinal use. Mushroom cultivation has gained increased attention because of economic, culinary and medicinal, industrial reasons [9]. However, the harvest of mushrooms remains with gaps because its potential success depends on how easy the preparation of a substrate for mushroom cultivation is and on the fact that mushroom farming can be easily practiced in any region regardless of the climate of the region [9]. Several methods in growing mushrooms get their significance in providing enough water to maintain the farming medium with desired parameters for an imperative medium to grow mushrooms and focus on an optimized way to utilize water [12].

Within this research, various works [7-8], [10-11], [13] have been reviewed and the main gap identified is that the harvest in mushrooms is still insufficient and water resource is not efficiently exploited for mushroom cultivation, and the research that is currently being done does not attain the satisfaction in mushroom yield. Shiitake mushrooms, which are used for both medicinal and diet purposes, have not been worked on by many researchers to avail them with sufficient harvest and there are gaps in using a technological approach to control the cultivation medium parameters on a real-time basis and manage water resource.

To alleviate the drawbacks above said, this research aims to design and implement a fuzzy inference-based system that will be a tool to maintain a favorable environment for shiitake mushrooms cultivation by supervising and keeping in the desired range of medium parameters (soil humidity, air humidity, temperature, and light) in a real-time basis. This will be done by applying the Internet of things (IoT) concept of wireless sensor network and a fuzzy inference system which is used to obtain a control system with imprecision inputs, many parameters, and good results similar to how humans perform decision making.

1.3 Objectives of the research

General objective

The main objective of this work is to develop an IoT-based shiitake mushroom cultivation system with a fuzzy inference model for a decision-making system to trigger actuators' activity. Fuzzy inference is a mathematical model to build a machine with intelligence [22].

Specific objectives

To achieve our general objective, the following specific objectives will be considered:

1. Design and implement a system to monitor the shiitake farm's air humidity and temperature to increase the harvest.
2. Design and implement a system to monitor the shiitake farm's substrate's moisture to increase the harvest.
3. Design and implement a system to identify movement and light levels in the shiitake farm to increase the harvest.
4. Develop a fuzzy logic algorithm to control the pump for water use optimization.

1.4 Hypotheses of the research

In this research, two main null hypotheses are taken to be considered in the evaluation of the proposed method. These are:

1. The use of the internet of things in shiitake mushroom cultivation increases the yield of shiitake mushrooms compared to traditional methods.
2. Using fuzzy logic in automating the control of the watering time saves water quantity used.

1.5 Scope of the research

This research focuses on identifying how IoT is used in improving the harvest of shiitake mushrooms by monitoring farming substrate vital parameters. A fuzzy inference system is used to process the data from sensors and make a decision on the actuator state.

1.6 Benefits of the research

The purpose of this thesis is to develop an IoT-based algorithm with a prototype to efficiently cultivate shiitake mushrooms regardless of the region and climate.

Mushroom farmers will get a system to help them introduce a new type of mushroom that is important in human immunity-boosting without complications with manual handling.

This research informs readers and scholars in information technology especially in the field of smart agriculture and precision farming by providing the basics for integrating technology in farming to increase the harvest of mushrooms.

1.7 Contribution of the research

The contribution of this thesis is to propose a method of mushroom cultivation based on the IoT and specifically the fuzzy inference system to control the mushroom farm parameters. The first contribution is using technology to increase the yield of shiitake mushrooms. The proposed method avails an IoT approach to result in a good and quality harvest. The second contribution is about the development of algorithms that will be used in the smart cultivation of sensitive plant species regardless of the regional climate with a wanted harvest.

1.8 Organization of the research report

This thesis report is organized into five chapters as follows:

Chapter one: General introduction, describes the background of the study, the problem statement, and the objectives of the study, scope of the study, the significance of the study, project interests, and the organization of the study.

Chapter two: presents basic theories and related literature review. It discussed related works in the fields of mushroom cultivation, shiitake mushrooms cultivation, and fuzzy logic used in automated systems and the rationale of this research after analyzing the existing works.

Chapter Three: describes the research methodology. It emphasizes the proposed methodology to use IoT for monitoring and enhancing decision-making in agricultural applications.

Chapter four: presents the system prototype design and results analysis, and illustrates the simulation results

Chapter five: encompasses the conclusions, and recommendations for future research.

CHAPTER TWO: LITERATURE REVIEW

2.1 Introduction

This chapter discusses related works and identifies limitations, and presents contributions to adding them for efficient cultivation of shiitake mushrooms using IoT. In this chapter, an overview of the background of this research area is discussed. Related work is presented in this chapter. Similarities and contrasting features within conferences and journal papers are also discussed. Some of the parameters, technologies and materials that were not considered by some previous researchers are highlighted and our scientific contributions are identified.

2.2 Review of related works

Previous researchers studied different ways of mushroom cultivation where soil parameters of the mushroom's substrate have not been recorded and analyzed further decision making but kept by water dispersion either manually or automatically. Watering the substrate is one of the top techniques to maintain the temperature and humidity of the cultivation house [14].

By focusing on oyster mushroom cultivation, the research in [14] worked on using fuzzy logic to monitor temperature and humidity in oyster mushroom form by detecting the temperature with one sensor DHT11 and the response to these parameters' variation is done by using a fan. It should be noted that [14] did not address issues of insects' prevention, soil (substrate) humidity, light, and wind speed in the mushroom cultivation, which are the very crucial parameters for mushroom growth, and thus the harvest was not improved sufficiently. Besides, an air fan only makes air available but doesn't do air conditioning that consists of entering cool air and getting out hot air for cultivation medium humidity monitoring.

In [15], a distributed system for supporting smart irrigation using IoT, soil moisture sensors have been used to monitor crop-field smartly and automate the field irrigation. The aim was to apply IoT technologies, sensor distribution, and automation in making a smart agricultural system to supervise the variation of soil moisture of an agricultural field. The system presents a feature of randomly switching ON or OFF a water sprinkler, which can lead to overwatering the field, hence a drawback of misuse and inundation in the field is likely to happen.

It is crucial to have an automated system that encompasses a decision-making system such as fuzzy inference, based on the soil parameters level of the substrate to control the water pump sprinkler.
The work in [16], designed a platform to manage water smartly is designed by using IoT to optimize water resources use in heterogeneous plots, where various types of crops are grown. The fact that this watering system is designed for different types of plants, it is not feasible for the types of plants that have different needs in water to be grown under the same system. Therefore, there is a need to emphasize one type of plant, study its specific soil parameters for its enhanced growth to get to a suitable system that contributes to improved harvest in that plant, hence our research focuses on one type of mushroom.

In [17], an IoT-based system was designed to monitor a greenhouse climate using sensors and a microcontroller. The research worked on improving the plant growth in a greenhouse by recording and reporting the greenhouse's humidity, temperature, and soil moisture, and these were considered as a main climatic factor of a greenhouse while some others like light intensity have also to be taken into consideration to keep the desired climate of a field. This work in [17] focused on the measurement of humidity, temperature, and soil moisture and on the collection of the data in them to report them on an android device to alert the farmer on their variation. Because this system lacks a self-decision-making system to restore the soil parameters to a needed level, and that it involves much manpower to re-establish these parameters to normal, it still needs to be completed by an improved system with a self-decision-making feature to lessen human intervention in greenhouse climatic factors management. Therefore, our research focuses on designing a system to supervise and control a farm for mushroom cultivation using a fuzzy inference system for decision making to automatically take action on any significant variation of the farm's climatic factors.

Moreover, in [18], a survey on IoT-based mushroom monitoring systems was focused on, where the sensors are placed in specific places of the growing area to record data of the farm. This survey identifies those basic parameters to be taken into consideration are notably the temperature, humidity, and gas content.
However, there are other parameters with capital importance to the cultivation of mushrooms such as light intensity, and wind direction, and speed in the farm. Within the system proposed at [18], an intelligent application was designed to check the status of the mushroom's farm by the user who is connected with the control unit through a server.

Thus, this system lacks automation in taking action on a variety of field parameters because any actuation is triggered by a user through a server. Therefore, our research avails a system to handle this autonomy drawback in taking action on the parameters variation by introducing the Mamdani fuzzy inference system which is an effective tool for handling events that are not restricted to only the value of a given time point but also include all values within certain time intervals.

The work in [19], designed a wireless sensor network system to monitor the soil moisture, and the presence of microbial using soil moisture sensors and biosensors. Raspberry Pi is used as a central microcontroller and other different ESP8266 nodes are used to communicate for real-time data management. Within this system, a website is introduced to act as an interface for users to interact with the system and also to act as an automation tool. Though this research [19] presents a needed key feature to detect any appearance of microbial and/or insects, it doesn't cover all the needed features to effectively control the climate and security of the mushroom's farm such as temperature, farm's humidity, and other vital parameters for shiitake mushroom growing. Thus, our research improves this system of the sensor by adding an artificial intelligence-based feature of fuzzy logic to support automated decision-making based on the variation of the parameters of the mushroom's farm.

A system to observe white mushrooms was implemented in [20] where they used a robot to reduce human responsibility on the mushroom plant by automating white button mushroom cultivation and observing the product space set standing. Within this research, specific farm parameters focused on are namely temperature, humidity, carbonic acid gas concentration, and light-weight intensity. Besides, the fact that this system in [20] is costly, there are some other mushroom farm parameters with capital importance that have not been considered such as insects' presence detection, which will be considered in our work.

Moreover, an intelligent mushroom monitoring system was presented in [21] where the researchers considered the temperature, humidity, CO_2 of the farm as the main soil parameters to work on to increase the yield of the mushroom's cultivation. The work [21]e developed a system based on the IoT with sensors to measure and monitor the soil parameters. The data collected are benchmarked to the threshold values of each parameter to trigger the actuation process. To get to an optimum growth of the mushrooms, some other parameters should be considered to optimize the climate conditions of the mushroom farm.

To address the drawbacks identified in all of the previous works consulted our research targets to designing and implementing a fuzzy inference-based system to autonomously monitor the climate conditions of shiitake mushroom's farm by considering as main parameters the farm's humidity and temperature, the substrate's moisture, the presence of insects, the wind and light-weight intensity in the farm.

2.3 Summary on weaknesses identification and our scientific contribution

Based on related literature, mushrooms have been cultivated using different techniques such as time-based watering method, implementing a greenhouse for mushroom farming, and others working on cultivating Shiitake mushrooms as they have been identified in the literature review above. However, weaknesses have been identified, and through this research, a scientific contribution is made to increase the harvest of Shiitake mushrooms and make an efficient use of water.

2.3.1 Weaknesses identified

After reviewing the literature above and analyzing them, they have been classified into five categories based on what they did and their scientific contribution.

Firstly, a researcher in [15] worked on applying IoT in agriculture by designing a platform that manages irrigation water by automating water pump switching On and Off based on the variation in soil water level known as soil moisture. This work presents two main gaps namely addressing soil parameters that cannot be enough for crops growth and lacking a self-decision making such as fuzzy logic to prevent inundation and water misuses.

Secondly, the works in [16] and [19] developed distributed systems for smart irrigation by placing sensors in different specific locations of the field of cultivation. Specifically, the research in [19] worked on mushroom cultivation. Both these works present a common gap of not being specific on the type of plant or mushroom to work with their systems which lead to uncertainty that can prevent these works from being practiced.

Thirdly, the research in [17] worked on an improvement of a greenhouse plant's growth by using an approach to record the quantity in humidity, temperature, and soil moisture of the greenhouse and manually taking action by involving human intervention to maintain them at the desired level.

This work has as the main challenge, the lack of a self-deciding system to take action to any variation of the greenhouse plant vital parameters variations with the addition of addressing a few parameters such as wind direction and light intensity in the greenhouse. Fourthly, in [20] a work to cultivate mushrooms has been done where a robot was used to reduce human responsibility on caring for mushrooms by automating white button mushroom field parameters re-establishment. The parameters considered are soil temperature, humidity, carbonic acid gas concentration, and light-weight intensity. Within this work, two gaps have been identified namely the designed system is costly, and insect's prevention, an important parameter for mushroom growth, has been looked down and this reduces the harvest.

Finally, the work in [13] designed a system based on Fuzzy logic to monitor oyster mushroom farm's temperature and humidity. As the main gap identified in this work, all basic parameters to maintain a mushroom farm have not been considered which implies that this work cannot efficiently contribute to the harvest improvement in mushrooms.

2.3.2 Scientific contribution

Our research works on the development of a prototype for shiitake mushroom farm monitoring and controlling. Shiitake mushrooms are a type of mushroom that is mostly used in medicinal applications to boost immunity and sometimes used as a dietetic ingredient. Shiitake mushrooms' cultivation has been looked down upon in previous many works about mushroom cultivation because they are useful in both medicinal uses, we decided to work on them in this research project. This research contributes to shiitake mushroom harvest improvements as well as easing its introduction in different regions of the world by availing a prototype to be used in cultivating them regardless of the region's climate.

2.4 Sensors and tools

2.4.1 Water pump (12V DC water pump)

A water pump, which is a direct current motor, is an electronic machine that converts direct current into mechanical power. A 12V DC water pump is mostly applied in IoT systems including smart irrigation projects due to its small size and low-cost [15].

Figure 1: 12V DC water pump [15]

2.4.2 Relay

A relay is a device that becomes active automatically when an electrical current is passed. Relays are like switches operated by the current. Relays are used to protect the systems and control power distribution. Relays turn on the corresponding devices like sprayer, light, door, and water motor. A relay can be managed at low voltage like 3.3 volts that are got from the pins of the ESP8266 and allows it to control high voltages such as 12V, 24V, etc. [17].

Figure 2: Relay [17]

2.4.3 Microcontroller

The NodeMCU is a product of Expressive systems that integrates a system on chip ESP8266 with 32-bit low power controlling unit, a Wi-Fi module of 2.4GHz, a RAM of 5KB, an ADC (Analog to Digital Converter), a 17 General Purpose Input and Output pins (GPIO), and an event-controlled API for the network [23], an addition of the NodeMCU to be a low-cost microcontroller made it considered as a best microcontroller to be used in our research.

The bellow figure3 illustrates the used Microcontroller (NodeMCU esp8266)

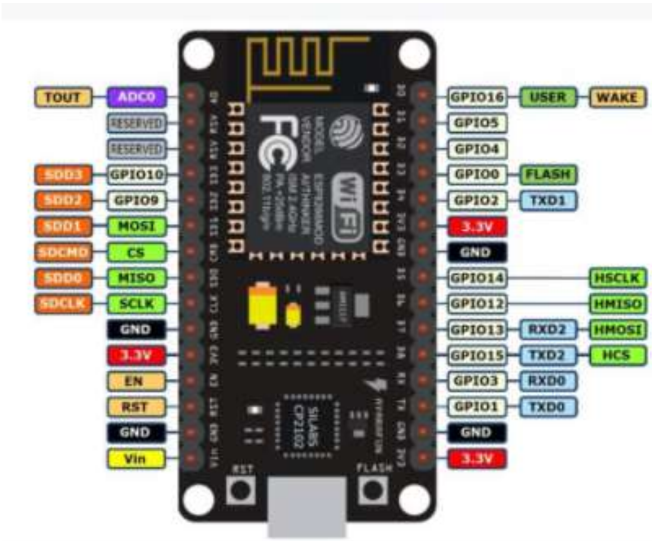

Figure 3:NodeMCU ESP8266 [23]

2.4.4 Temperature sensor (DHT11)

A temperature sensor is an electronic device that through an electrical signal gives the measurement of the temperature by converting the measured hotness and coldness detected from an object. A humidity sensor is an electronic device to measure the water vapor and convert the measured output into corresponding electrical signals. A humidity sensor is otherwise called a hygrometer because it combines both air moisture and air temperature [24].

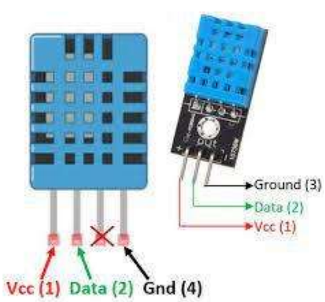

Figure 4: DHT11 Sensor [24].

2.4.5 Moisture sensor (Capacitive soil moisture sensor)

A moisture sensor is an electronic device used to measure the amount of water present. For a soil moisture sensor, it can be defined as a component to measure the volumetric water content in soil [24].

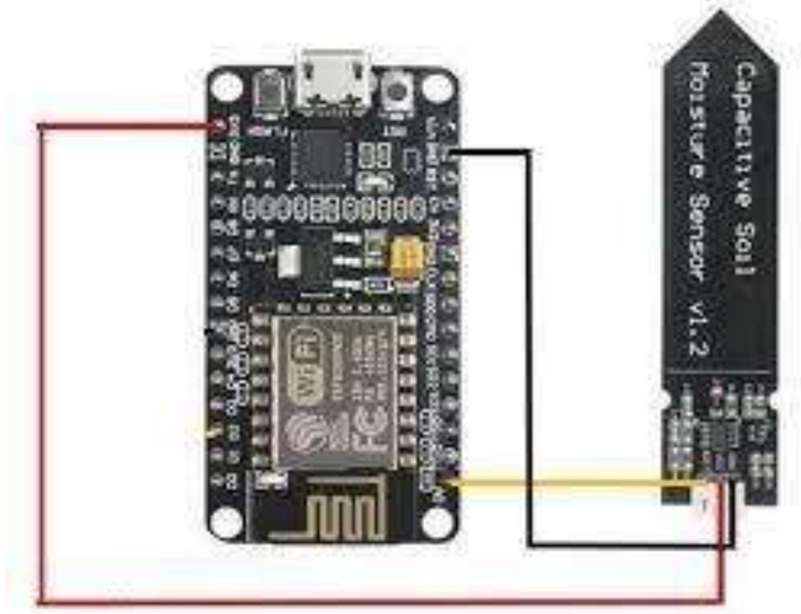

Figure 5: Capacitive soil moisture sensor V1.2 [24]

2.4.6 Light sensor (Photoresistor Sensor (LDR))

An electronic light sensor is a device that detects whether or not light or darkness is present. The light sensors are classified into different categories notably photodiodes, phototransistors, and photoresistors [26].

Figure 6: Photoresistor sensor (LDR) [26]

2.4.7 Proximity sensor (PIR (passive infrared sensor))

PIR sensors allow you to sense motion, almost always used to detect whether a human has moved in or out of the sensors range [26].

Figure 7: PIR (passive infrared) sensor [26].

CHAPTER THREE: METHODOLOGY

3.1 Introduction

The overall design of our proposed solution is presented in this chapter namely the hardware and software design. The methodological approaches identified encompass both physical designs which reflect hardware components selection and design and logical design which also reflects the software components selection and design.

3.2 Research and design approach

Figure 8 presents the overview of the steps involved in research and system development from the step of gathering the ideas to the final step of prototype and getting results.

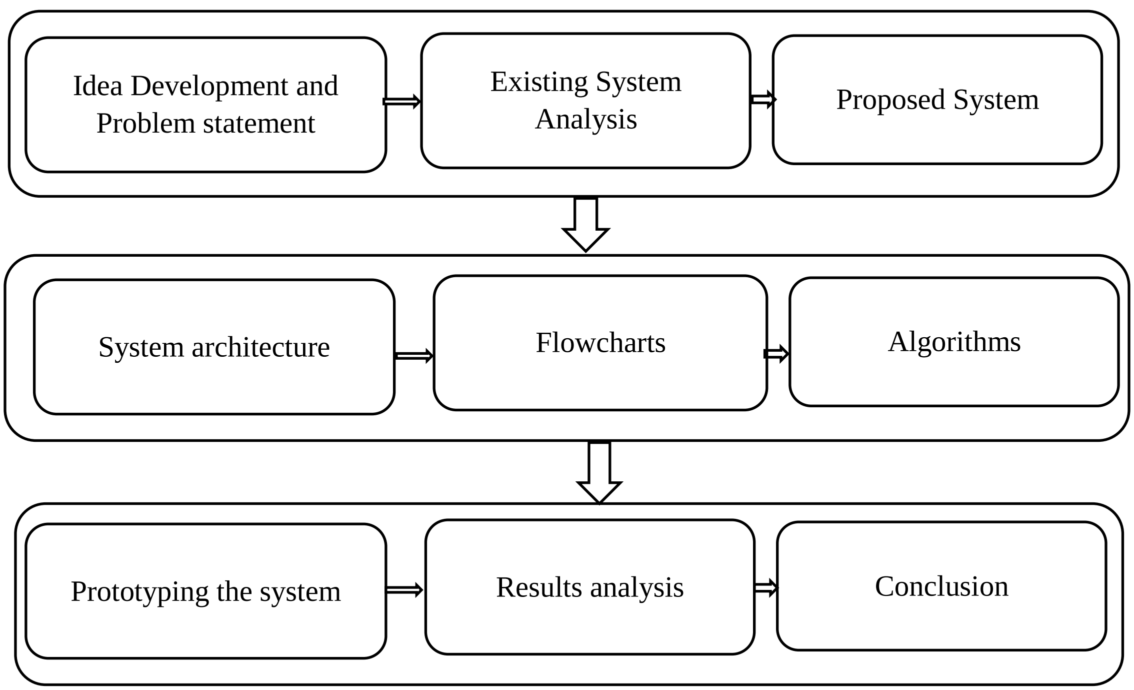

Figure 8: Research Approach Source: Own drawing

Firstly, the idea development and problem statement for this research departed from conducting a literature review to identify the problem and limitations of the existing works. After analysis of the existing system and identification of the gaps, we proposed a system to address the gaps. Secondly, the system architecture, flowcharts, and algorithms for the proposed system were made, and finally, we did system prototyping and analyzed the obtained results to conclude.

3.3 Scientific research approach

In this research, we have used qualitative research including observation of the existing mushroom farms, and document consultancy by reading journal papers. The goal was to investigate the existing system and to find its merits and weaknesses to find a way to improve it by applying technology.

3.4 System Development

In the context of research methodology, software development life cycle is a prototyping model with a system development method (SDM) in which a prototype is built, tested, and then reworked as necessary until an acceptable prototype is finally achieved from which the complete system or product can now be developed [22].

The following are the steps involved in the System Development Method used (Prototyping Model).

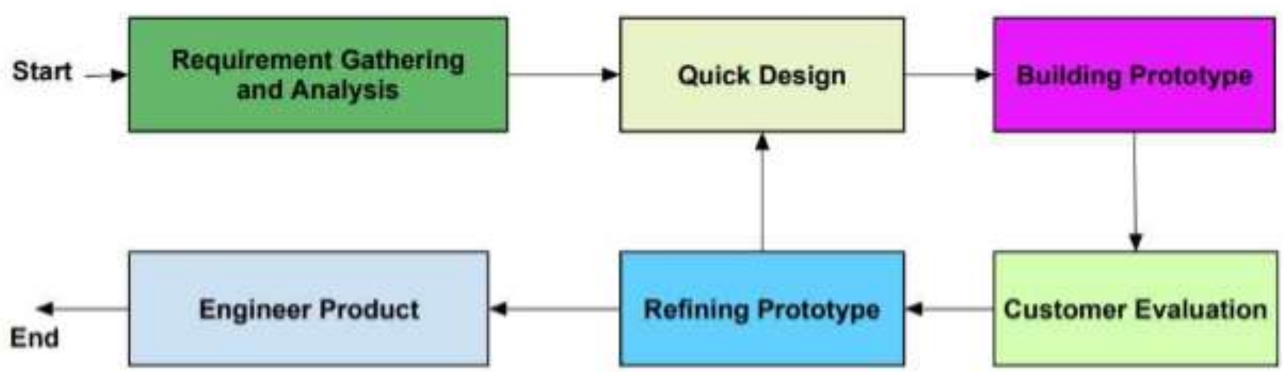

Figure 9: System development model adapted from [22]

For our research, we adopted two blocks of figure 9 above, namely requirements gathering and analysis and prototype building.

For the requirements gathering part, firstly we visited a mushroom farm called CP&T LTD at Ndera. CP&T LTD is a mushroom breeding company that works in collaboration with Rwanda Green Fund otherwise called FONERWA. FONERWA invests in public and private projects which have an aim to transformative change. At CP&T LTD we investigated how a mushroom farm is implemented from the seed's nursery to the farm harvest.

To understand the existing method of shiitake cultivation, we examined the conditions of the farm by observation, and to analyze the collected information, we read through various documents especially [10] that are about shiitake mushrooms cultivation.

For prototype building, we used IoT technology to build a prototype after getting to know the requirements for shiitake mushroom cultivation. The built prototype implements fuzzy logic to monitor the farm's vital parameters namely air humidity, temperature, and soil moisture in the substrate. We used a system with three main parts namely the sensing part (Sensors), controlling part (Microcontroller), and Actuating part (Water pump).

3.5 Components of the proposed system

3.5.1 Logical components

A. Layers of the proposed system

The proposed system has four layers arranged from perception layer to application layer as illustrated in figure 10.

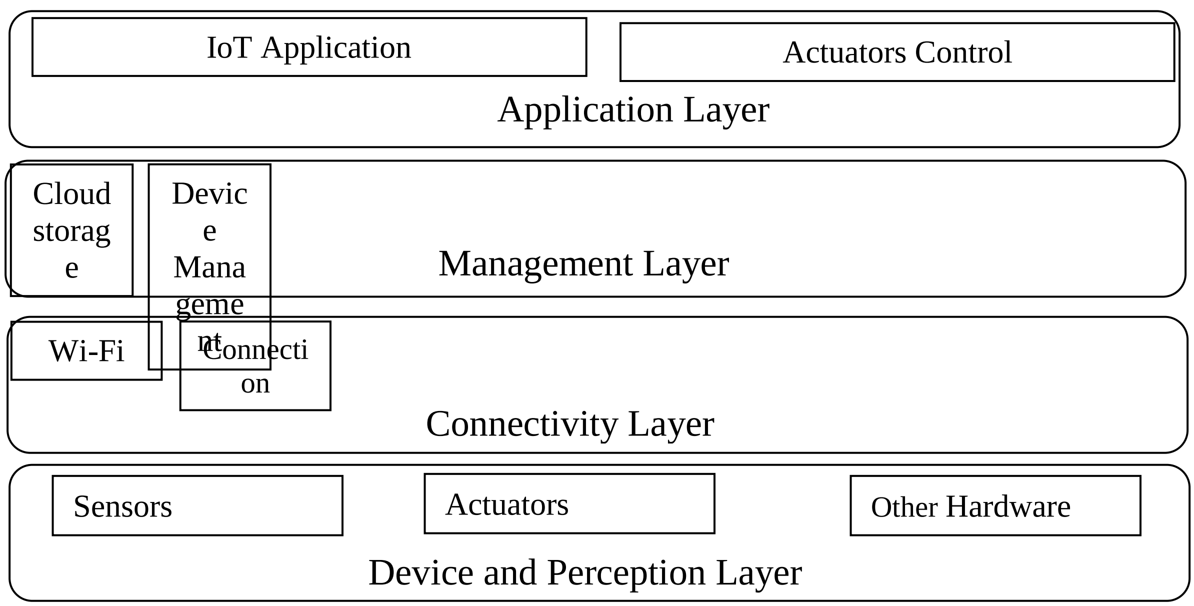

Figure 10: Layers of the proposed system adapted from [22]   Source: Own drawing

The purpose of these layers is to secure any communication and interactivity between the components of the system. All hardware devices of our self-powering automatic watering system of the mushrooms farm are placed in the device and perception layer, all the networking and communication components are placed in the connectivity and network layer, the management layer hosts all services such as cloud storage, device management, and storage, and application layer is for IoT application and actuation service.

B. Flowcharts Design of the proposed system

This system is made of an actuating task namely, pumping water for irrigation (figure 11). Based on the fact that the main flowchart is designed separately and put together the whole system.

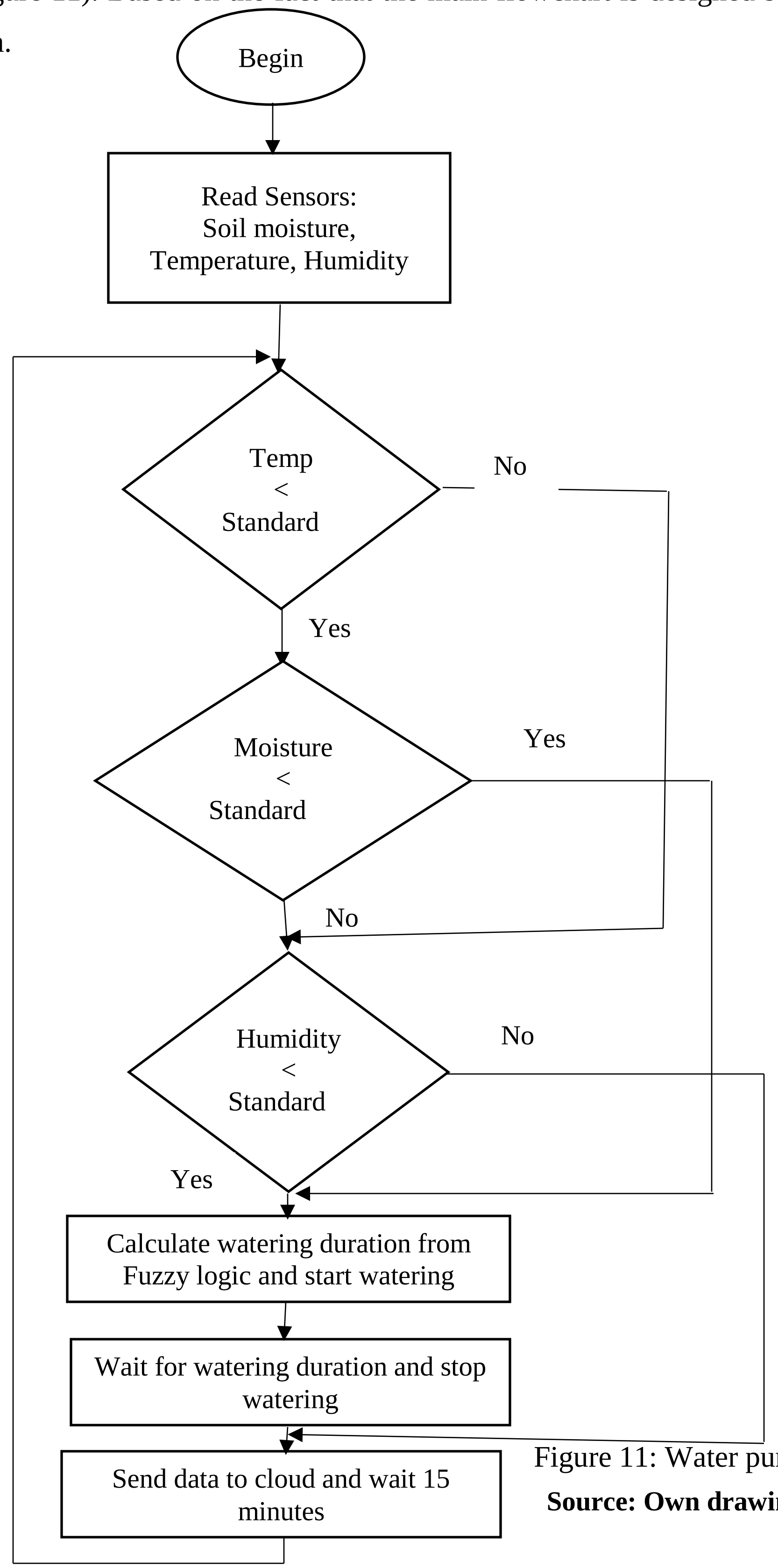

Figure 11: Water pump flowchart

Source: Own drawing

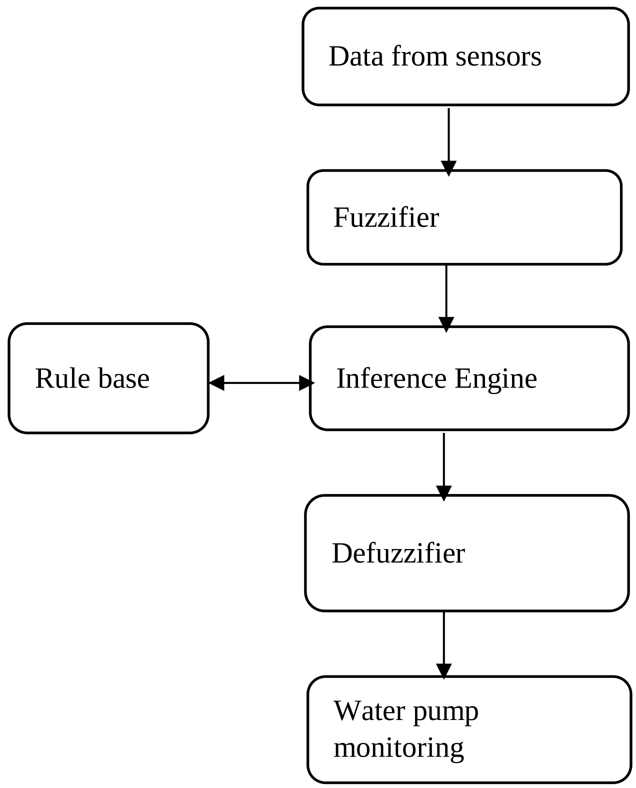

Figure 12: Fuzzy Inference system flowchart Source: Own drawing

3.5.2 Physical components

The design of the proposed system is based on the following Figure 13 with three main sides namely: A sensing side, a controlling, and actuating side. The sensing side is composed of a temperature & humidity sensor for the farm air temperature and humidity collection, a soil moisture sensor for the substrate's moisture sensing, a PIR to detect movements, and a light sensor to detect the light intensity of the farm. The controlling side is made of a NodeMCU ESP8266, Arduino UNO, and a cloud platform for data analytics, and an actuating module is made of one 12V DC water pump and one relay to make them controlled by a NodeMCU ESP8266.

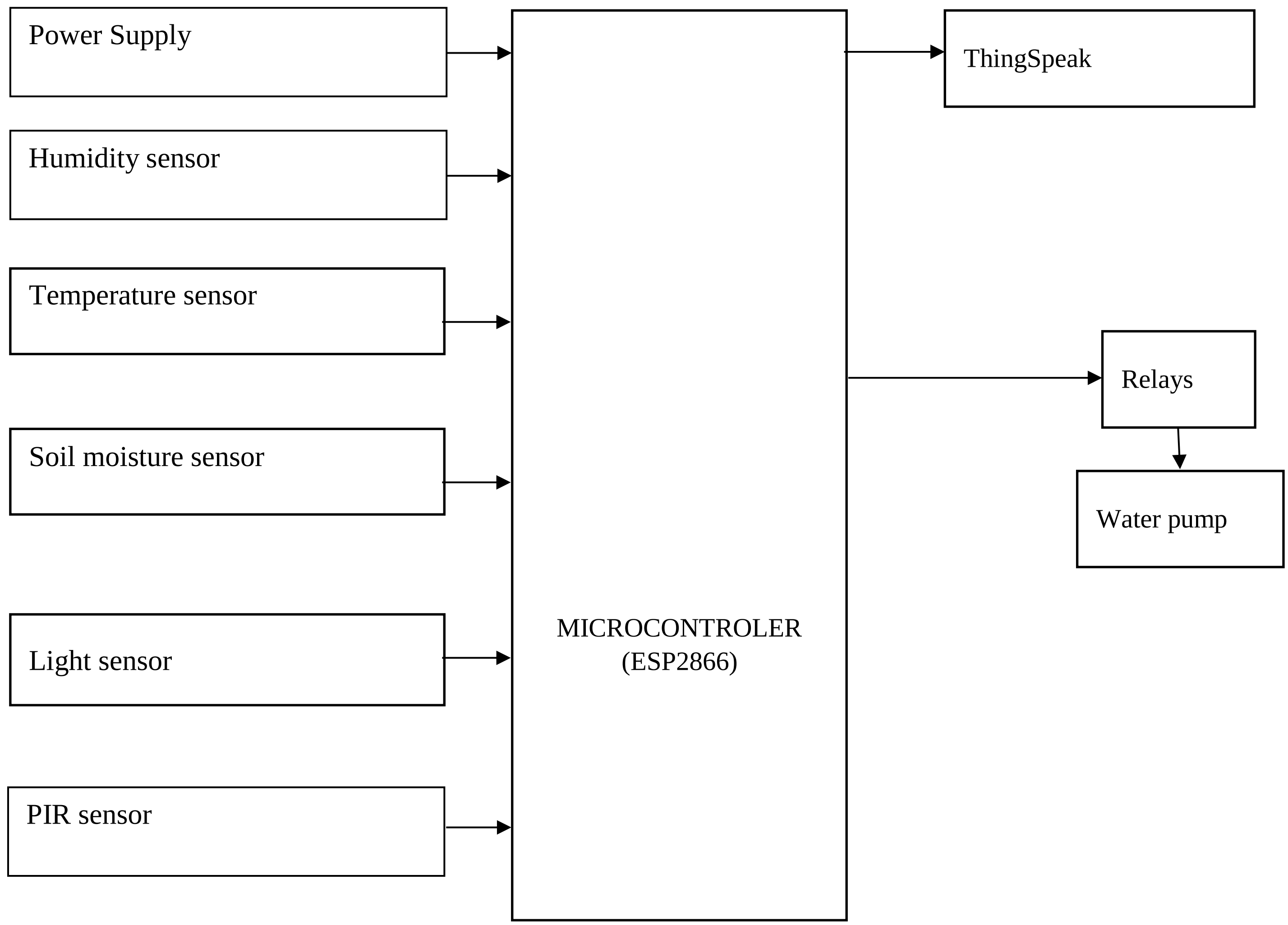

Figure 13: Components of the Proposed system Source: Own drawing

3.6 Hardware components selection

3.6.1 Microcontroller Selection

The NodeMCU microcontroller has been chosen because it has the advantage to operate at a current of 80 mA with a quiescent current of 5 to 11 mA delivered from power dissipation by its printed circuit [23].

3.6.2 Sensing side components selection

I. Humidity and Temperature Sensor

In this research, the air temperature needs to be measured to get the expression of the hotness and coldness of the soil timely. To measure the temperature and humidity, a DHT11 sensor is chosen based on its capacity to measure both at the same time, it is an ultra-low-cost device, and gets new data every two seconds (2 secs).

II. Soil Moisture Sensor

In this research, it is crucial to get the data about the quantity of water available in the soil to decide whether to spread water or not. Thus, we decided to use capacitive soil moisture sensor V1.2 which has as an advantage its ability to yield a reliable relationship between the output voltage and gravimetric water content.

III. Light sensor

For our project, the researchers have chosen to use a passive photoresistor (LDR – Light Dependent Resistor) for recording data about the brightness of light in the mushroom's substrate.

IV. Proximity Sensor (PIR)

PIR (passive infrared) sensor is chosen for our project because they are small, inexpensive, low-power, easy to use, and don't wear out and for a reason that they are commonly found in appliances and gadgets used in homes or businesses.

3.6.3 Actuating side components selection

I. Relay selection

In this research, we preferred to use a relay to make the devices with various voltages controlled using ESP8266 that uses 3.3 v.

II. Water pump selection

A 12 volts direct current motor water pump was chosen for our project considering its features to fit in and the fact that it is a cost-effective and efficient device.

3.7 Software components selection

Fuzzy logic is a tool capable of building a machine with intelligence. The fuzzy system is performed through three stages namely fuzzification, fuzzy processing, and defuzzification. Fuzzification is the first operation in a fuzzy system that works on input data, it is a process mapping a crisp input value of a system into a fuzzy input, that is to associate with a fuzzy set. Since each linguistic variable has values that can be described linguistically to a set of its values, we can determine the number of linguistic terms. Sometimes, the inputs are as features of classes, as in the Fuzzy classifier.

Mathematically, fuzzification can be defined as a mapping F of the crisp input domain I, with x into a set A of fuzzified input.

That is as; $F: I \rightarrow A$

In the context of our research, the crisp input domain I am a set of all the data collected from the sensors, and the fuzzified input are the ones that will be got from MatLab as processed sensors input.

Simulating this in MatLab, a fuzzy inference system, the main component of the proposed decision-making system, also based on a set of **if-then rules**, in addition to a set of membership functions, and fuzzy logic operators such as "and", "or" are developed referring to Mamdani fuzzy inference system.

To get figure 14 below, we supplied the MatLab fuzzy designer with three variables namely the temperature, the soil moisture, and the humidity.

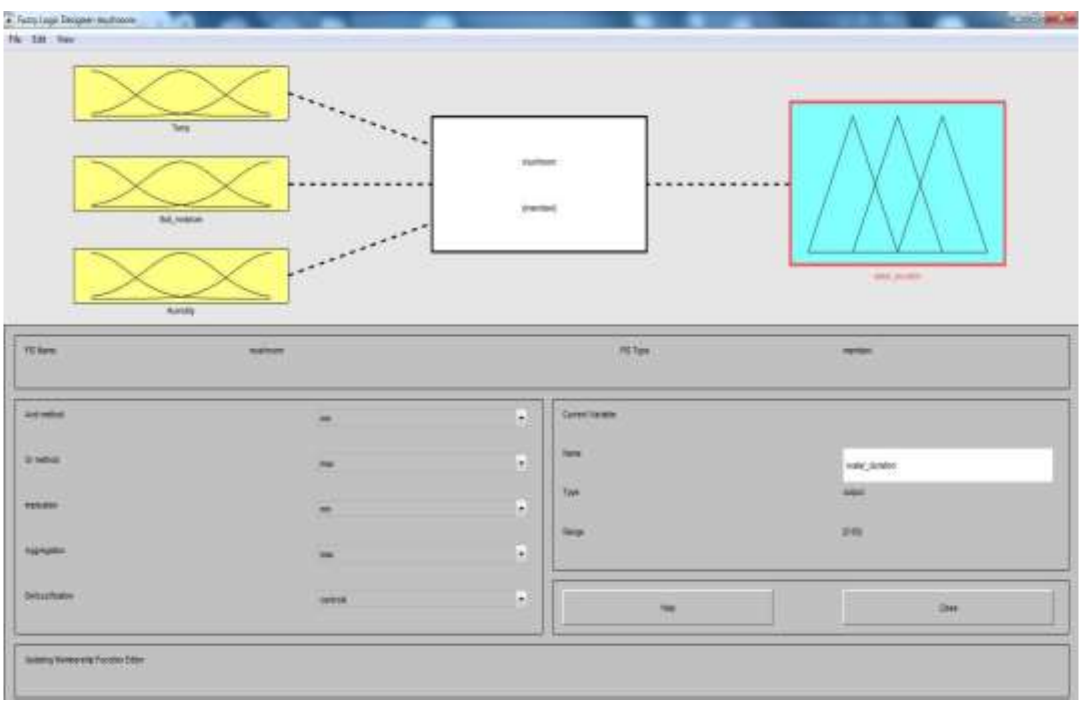

Figure 14: FL designer for Mushroom monitoring   Source: Interface from MatLab based on our input

Figure 14 above shows the fuzzy logic design as a black box that consists of the input block, rule inference engine, and an output block to help visualize the structure of the fuzzy inference system.

A. Fuzzy rules

These are rules used during processing to determine the output of the system. Based on the inputs, the rules will analyze the combination of various inputs and run inference using the engine to determine a suitable output to maintain moisture content.

Fuzzy processing consists of applying fuzzy rules on inputs with the purpose to provide output. A Fuzzy rule is a form that serves to describe, in linguistic terms (words) a qualitative relationship between two or more variables. That is, it joins the input variable(s) and its (variables) influence with the output variable(s) and these may be linguistic or scalar.

In this research, fuzzy logic is used in this module due to its bi-fold significance. First, fuzzy logic is a better approach for decision-making problems. Secondly, fuzzy logic is simple and easy to implement in IoT applications [22], and the following steps will be used to start from handling crisp input to crisp

For our research, we have 8 rules to monitor the water pump state based on the input values. These rules are set based on the threshold values that are the normal values of the Shiitake mushroom farm to grow efficiently. As of [10] the normal ranges for temperature, soil moisture, and humidity to grow shiitake mushroom are respectively 20^oC- 30^oC, 30%-65%, and 80%-90%, and if the two of them are satisfied, the farm is in a normal state.

Rule 1: if the temperature t is such that $20^oC \leq t \leq 30^oC$ and Soil moisture SM is such as $30\% \leq SM \leq 65\%$ and Air humidity H are such as $80\% \leq H \leq 90\%$, then switch off the water pump (Normal).

Rule 2: if the temperature t is such that $20^oC \leq t \leq 30^oC$ and Soil moisture SM is such as $30\% \leq SM \leq 65\%$ and Air humidity H are such as $H \leq 80$ % or, then switch off the water pump (Normal).

Rule 3: if the temperature t is such that $20^oC \leq t \leq 30^oC$ and Soil moisture SM is such as $SM \leq 30\%$ and Air humidity H are such as $80\% \leq H \leq 90\%$, then switch off the water pump (Normal).

Rule 4: if the temperature t is such that $30^oC \leq t$ and Soil moisture SM is such as $30\% \leq SM \leq 65\%$ and Air humidity H are such as $80\% \leq H \leq 90\%$, then switch off the water pump (Normal).

Rule 5: if the soil moisture SM is such as $30\% \leq SM \leq 65\%$ and temperature t are such that $30^oC \leq t$, and air humidity H is such as $H \leq 80\%$, then switch on the water pump (abnormal).

Rule 6: if the soil moisture SM is such as SM ≤ 30% and temperature t are such that 20°C ≤ t ≤ 30°C and air humidity H is such as H ≤ 80%, then switch on the water pump (abnormal).

Rule 7: if the soil moisture SM is such as SM ≤ 30% and temperature t are such that 30°C ≤ t and air humidity H is such as 80%≤ H ≤ 90%, then switch on the water pump (abnormal).

Rule 8: if the soil moisture SM is such as ≤ SM ≤ 30% and temperature t are such that 30°C≤ t, and air humidity H is such as H ≤ 80%, then switch on the water pump (abnormal).

Figure 15 below, shows a fuzzy interface of how the watering duration varies when the sensor inputs vary. Specifically, the rule viewer below reflects rule number 8 where the temperature is 94.6°C, soil moisture is 3.01%, and Humidity is 5.42 %. The pump state is watering and the watering duration is 51.8 seconds.

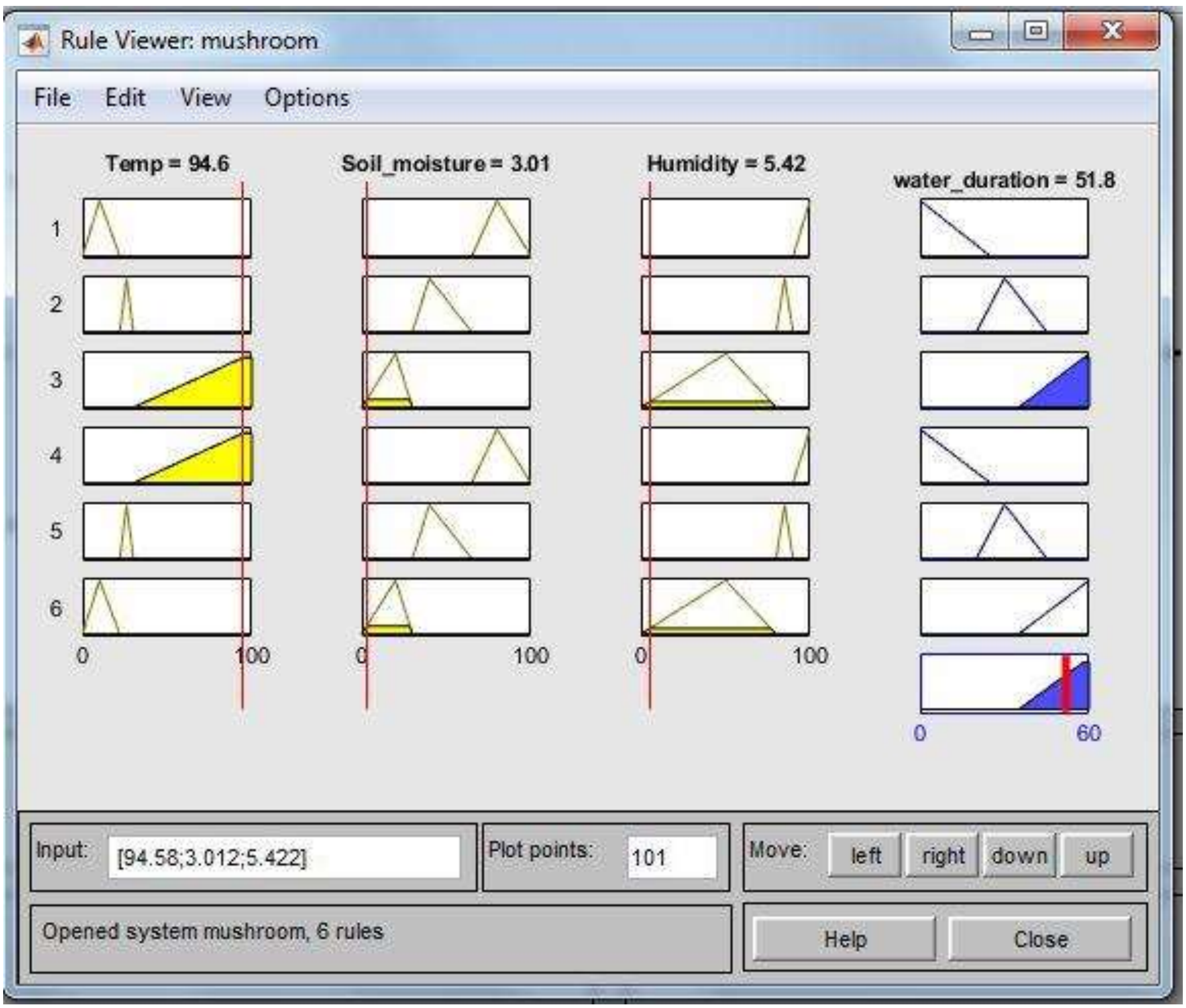

Figure 15:Fuzzy Rules viewer Source: Interface from MatLab based on our rule 8.

B. Fuzzy inputs (Data from sensors)

B.1 Input 1 for our fuzzy inference (Temperature)

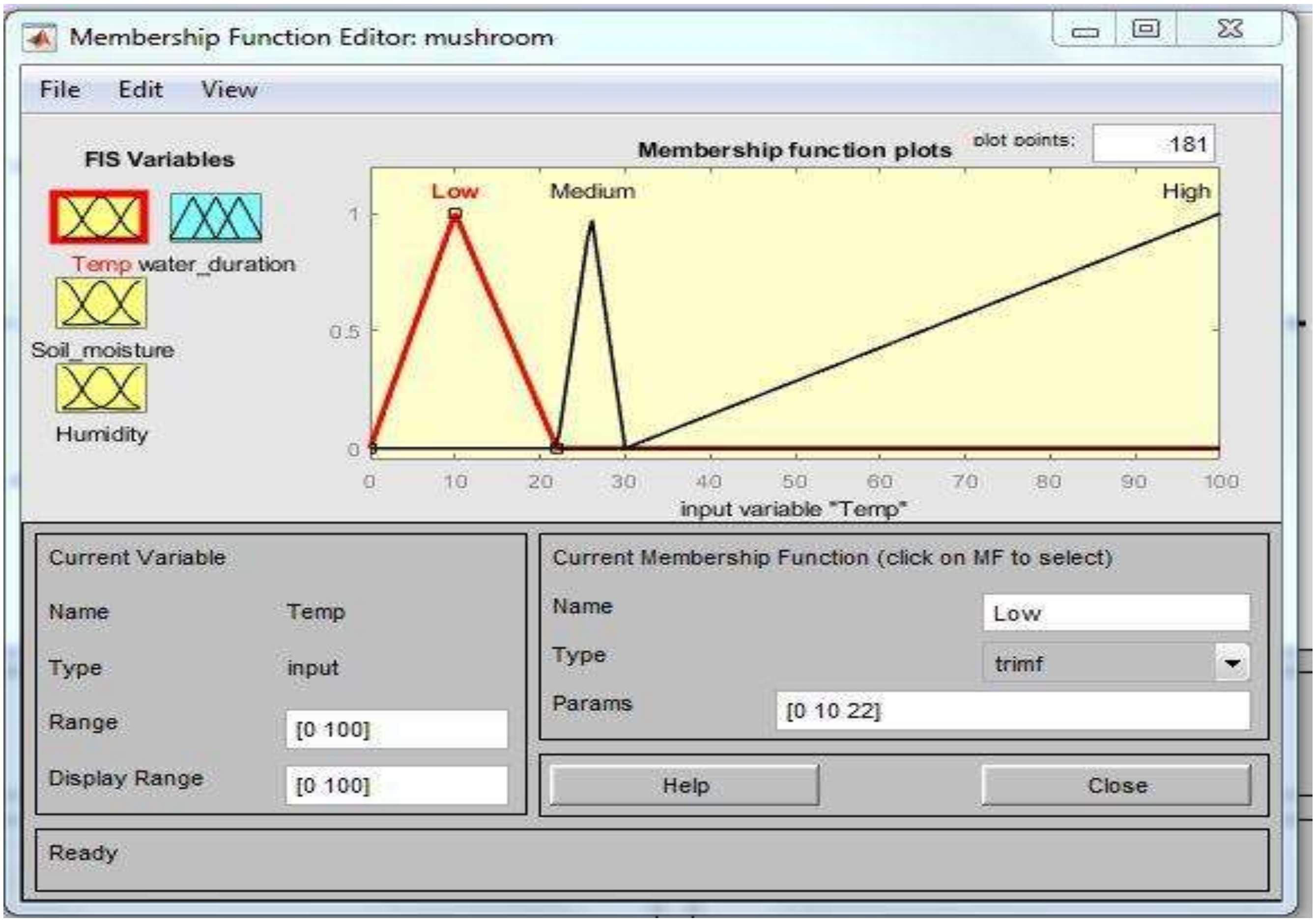

Figure 16:Temperature input for fuzzy design

Source: Interface from MatLab based on our supplied temperature range of 0 ^{o}C - 22 ^{o}C.

Figure 16 above is a fuzzy interface we get after supplying the FIS designer with a temperature range of 0 ^{o}C - 22 ^{o}C. This supplied range is not favorable to shiitake mushroom farm because, as of [10], shiitake mushroom farm's favorable temperature range is 20 ^{o}C - 30 ^{o}C.

As shown in Figure 16 above, the temperature range is from 0-100(degree Celsius): Based on a range of 20 ^{o}C - 30 ^{o}C as a valid range of temperature for shiitake mushroom growth as of [10], we consider the following criteria in creating input for temperature: Low-temperature range from 0-20, the medium is 20-30 and High is greater than 30.

B.2 Input 2 for our fuzzy inference (Soil moisture)

As of figure 17 below, the soil moisture range is from 0-100(%): Referring to [10], a range of 30% to 65% is considered to be a valid range of soil moisture for shiitake mushroom growth, therefore for our input as of figure 17 below, low soil moisture range is from 0%-30%, medium soil moisture range is 30%-65% and high soil moisture range is greater than 65%.

The figure below is an interface of MatLab FIS designer got by inputting a soil moisture range of 0%-30%, which is qualified as a low moisture level based on the standard range of 30%-65% of shiitake mushroom farm as of [10].

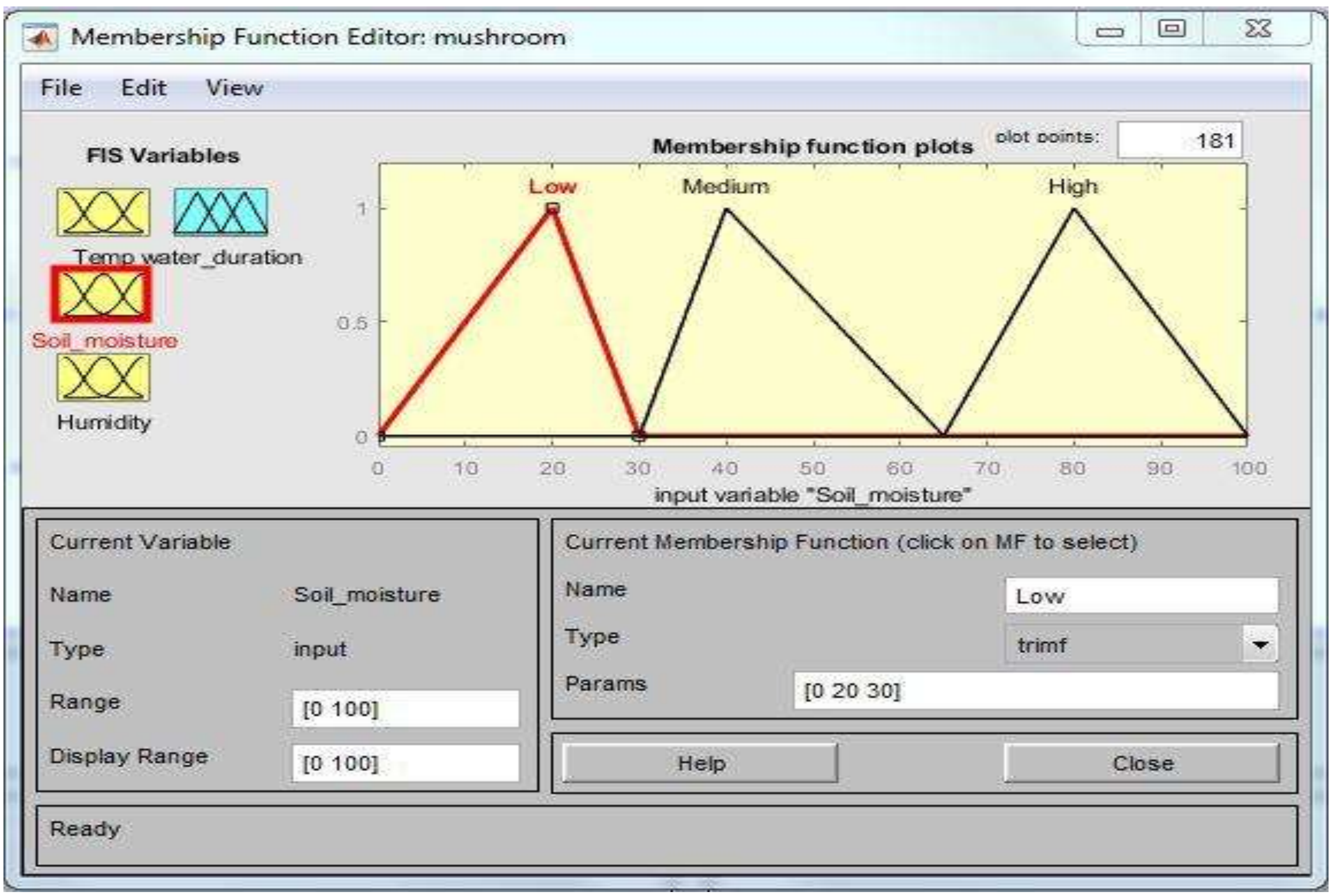

Figure 17: Soil moisture input for Fuzzy design

Source: Interface from MatLab based on our supplied soil moisture range of 0%-30%.

B.3 Input 3 for our fuzzy inference (Humidity)

As of figure 18 below, the air humidity range is from 0% to 100(%): Referring to [10], a range of 80% to 90% is considered to be a valid range of air humidity for shiitake mushroom growth, therefore for our input, low air humidity range is from 0%-80%, medium air humidity range is 80%-90% and high air humidity range is greater than 90%.

The figure below is a MatLab interface after supplying an air humidity range from 0%-80%, which is considered as a low air humidity for shiitake mushroom based on [10].

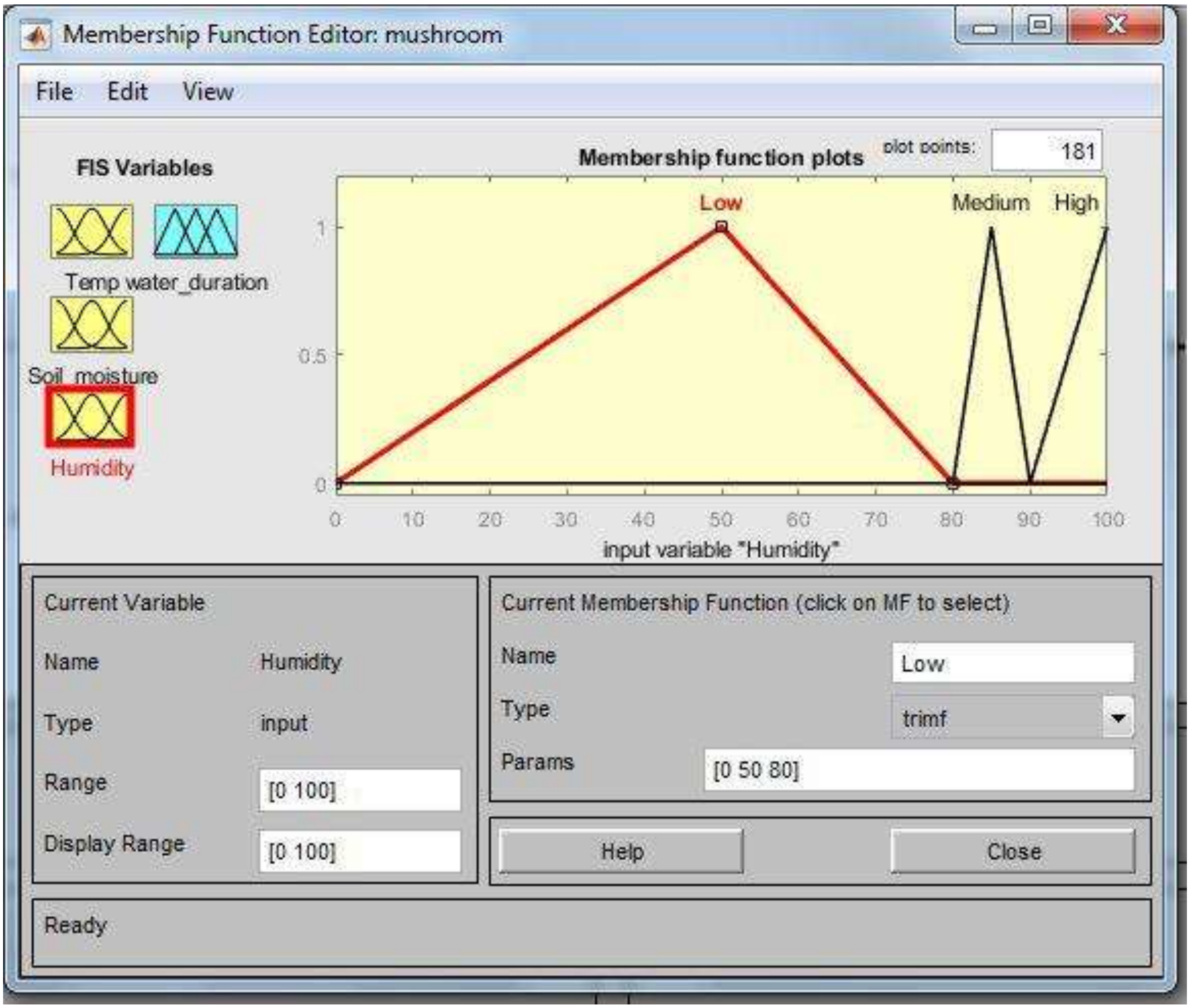

Figure 18: Humidity input for Fuzzy design

Source: Interface from MatLab based on our supplied-air humidity range of 0%-80%.

C. Membership functions of our fuzzy inference

The membership functions could then represent the amounts of tension, when choosing the input membership functions, the definition of terms may be different for each input.

For this research, our membership functions that represent the tension of our input data level are Low, Medium, and High. Low is used for input values that are below the range of the normal values, Medium is used for input values in the range of the normal values, and high is used for values above the range of normal values.

D. Fuzzy output (watering time)

Based on the volume of water per second from the water pump, the range of time the pump could be open is in the range of 0-60 seconds. Low watering time is 0-25, medium ranges from 20-45 while high ranges from 35-60.

Figure 19 below shows an interface from MatLab for a watering duration which here a low watering time because it is 25 seconds. A short watering time is beneficial to a farmer because the conservation of water is achieved.

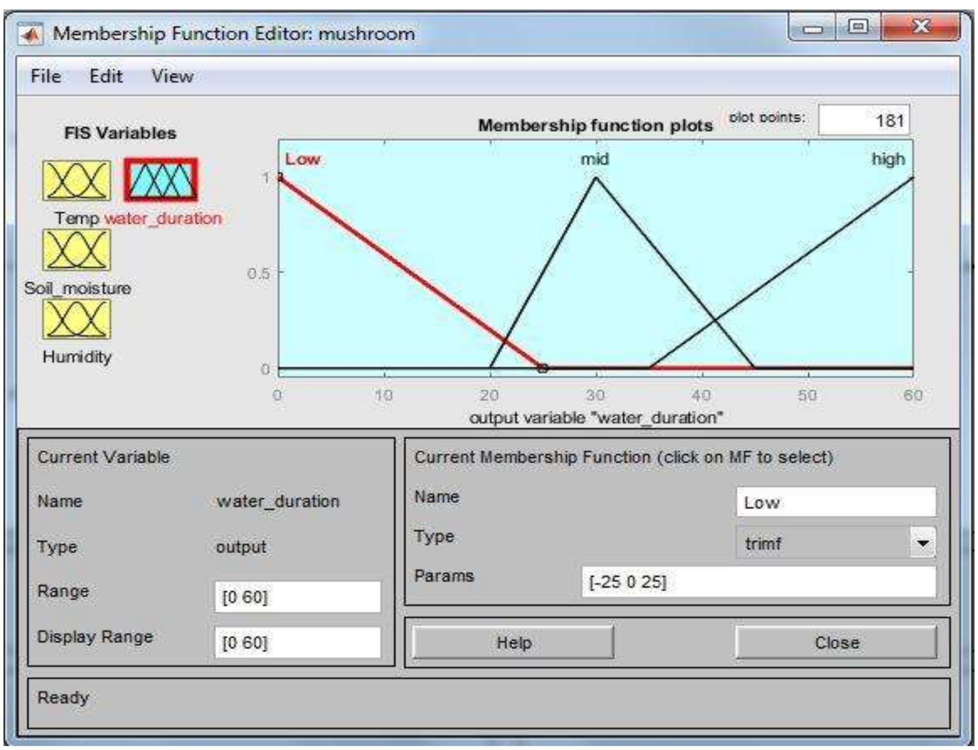

Figure 19: Fuzzy output variable-watering time Source: Interface from MatLab with a watering time of 25 seconds

E. Fuzzy output use for actuation (Being supplied to Arduino to run the pump)

It is well noted that the converted Matlab fuzzy file into Arduino code performs similarly to the MatLab simulations with similar outputs. Therefore, the overall created Fuzzy inference engine can acquire inputs directly from the sensor value and therefore determine watering periods automatically without intervention. Figure 20 below shows that the fuzzy engine uses 266108 bytes (25%) of program storage space. Global variables use 27288 bytes (33%) of dynamic memory, leaving 54632 bytes for local variables that will be for Arduino.

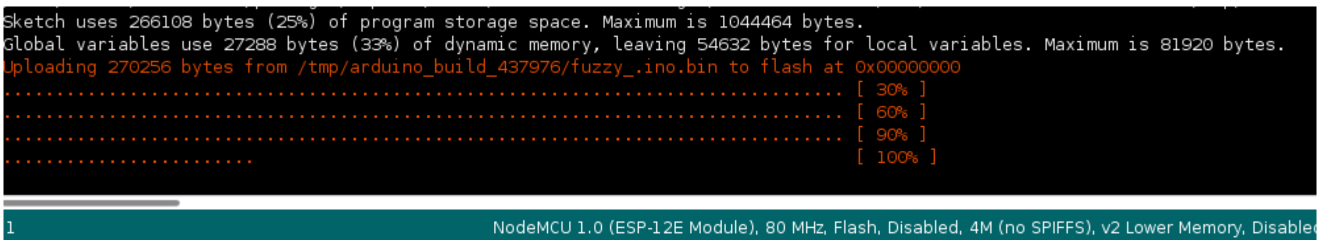

Figure 20: On-device performance specifications Source: Arduino application

3.8 Cloud platform selection

Cloud computing means storing and accessing data and programs over the internet instead of your computer’s hard drive. The cloud is also not about having dedicated network-attached storage hardware or server in residence, it’s as a service such that with all the various data stored on the computers in a cloud, data mining and analysis are necessary to access that information in an intelligent manner [23]. Thing speak is an open-source Internet of Things application and API to store and retrieve data from things using the HTTP and MQTT protocol over the internet. It’s an IoT analytics platform service that allows to aggregate, visualize, and analyze data streams in the cloud [23].

Within table 1 below, six platforms are compared by considering their number of channels, persistence days of the messages per year, and data retention. These platforms need a personal usage license freely delivered but with some restrictions [23]. Beebotte is a cloud platform that provides key building blocks to accelerate the development of the Internet of Things and real-time connected applications. Beebotte enables the transformation of any physical object or software application into a channel of digital resources. Beebotte is a cloud platform that offers infrastructure and connectivity with the help of REST, WebSockets, and MQTT [23].

DataGekko is a fully managed enterprise-grade metrics as a service solution, DataGekko can ingest data over MQTT with millisecond precision of data points with full resolution, this is an IoT Telemetry platform ready for the next generation of internet-connected devices that scales with you from hobby plans to enterprise systems [23].

IoTPlotter and Horavue are also good cloud platforms but ThingSpeak beats them in terms of persistent messages a day per year.

Platform	Number of channels	Persistent messages (day/year)	Data retention
Beebotte	Unlimited	5000/day	3 months
Iothook	3	4300/day	1 month
DataGekko	Unlimited	1440/day	7 days
IoTPlotter	1	5760/day	1 month
Horavue	10	1 GB/year	1 year
ThingSpeak	8	8200/day	1 year

Table 1: Comparison between IoT platforms [23]

For our project, the ThingSpeak API is chosen because it has three channels where each channel can store 8 variables and we have to store data about harvested energy, soil moisture, humidity, temperature, and real-time with an additional feature of best store and visualize data.

3.9 Integrated Development Environment Selection

The Arduino Integrated Development Environment (IDE) is a cross-platform application (for Windows, macOS, Linux) that is written in functions from C and C++.

It is used to write and upload programs to Arduino compatible boards, but also, with the help of third-party cores, other vendor development boards. In this research, Arduino 1.8 version is used.

CHAPTER FOUR: SYSTEM RESULTS AND DISCUSSION

4.1 Introduction

This chapter focuses on the acquired results. Its main parts are namely the results of the subsystems, the results of the whole systems, and the description of shiitake mushroom farm yield within 21 days.

4.2 Results of subsystems

A. Results of subsystem 1 (DHT11)

The DHT 11 as shown in figure 21 can capture Humidity and Temperature. Sometimes, due to climate variation, at room temperature the humidity values vary and can sometimes do not meet the expected humidity values recommended for Mushroom farming which are respectively 20 to 30 degrees Celsius, and 80% to 90% for temperature and air humidity. Therefore, for figure 21 below the temperature displayed is in acceptable range through the air humidity is out of range which implies the need to check the soil moisture to decide the water pump state. Hence, our system can visualize how temperature and humidity variations in the mushroom farm.

Figure 21: Temperature and humidity Serial Output   Source: Arduino based on our farm's data

B. Results of subsystem 2 (Soil moisture sensor)

Figure 22 below shows an increase in soil moisture percentage after dipping the sensor in water. Levels obtained reached 100%. With a steady increase of water over 1 week in an environment with limited high temperature. We were able to deduce that temperature factors can lead to soil moisture content being reduced to factors such as evaporation. Analog values are read tentatively from the soil moisture sensor and read on the Serial monitor.

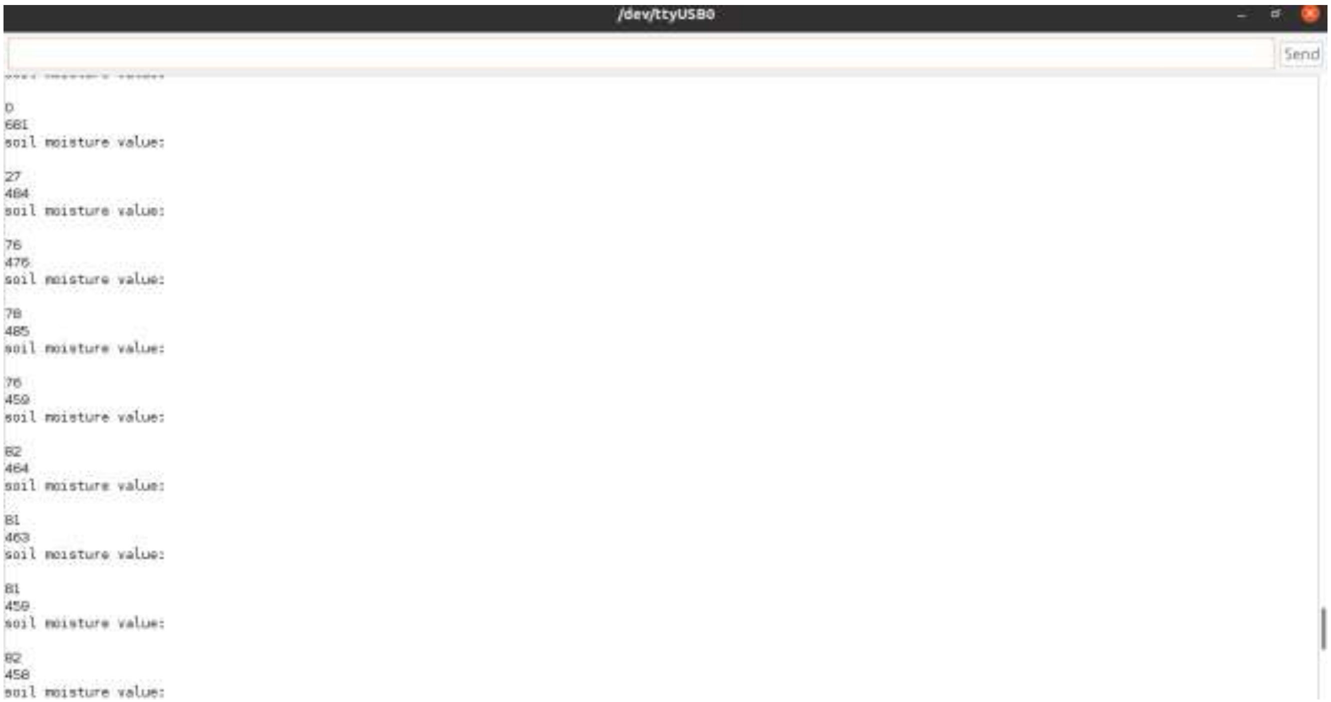

Figure 22: Soil moisture Serial Output Source: Arduino based on our farm's data

C. Results of subsystem 3 (PIR sensor)

As the pest is carried in the farm by external agents, the PIR is used in our project to detect the motion in the farm for the farmer to decide whether the pesticide can be spread in the farm or not depending on the cause of movement detection. Figure 23 below is about the results from the PIR sensor that show the movement detected in the farm

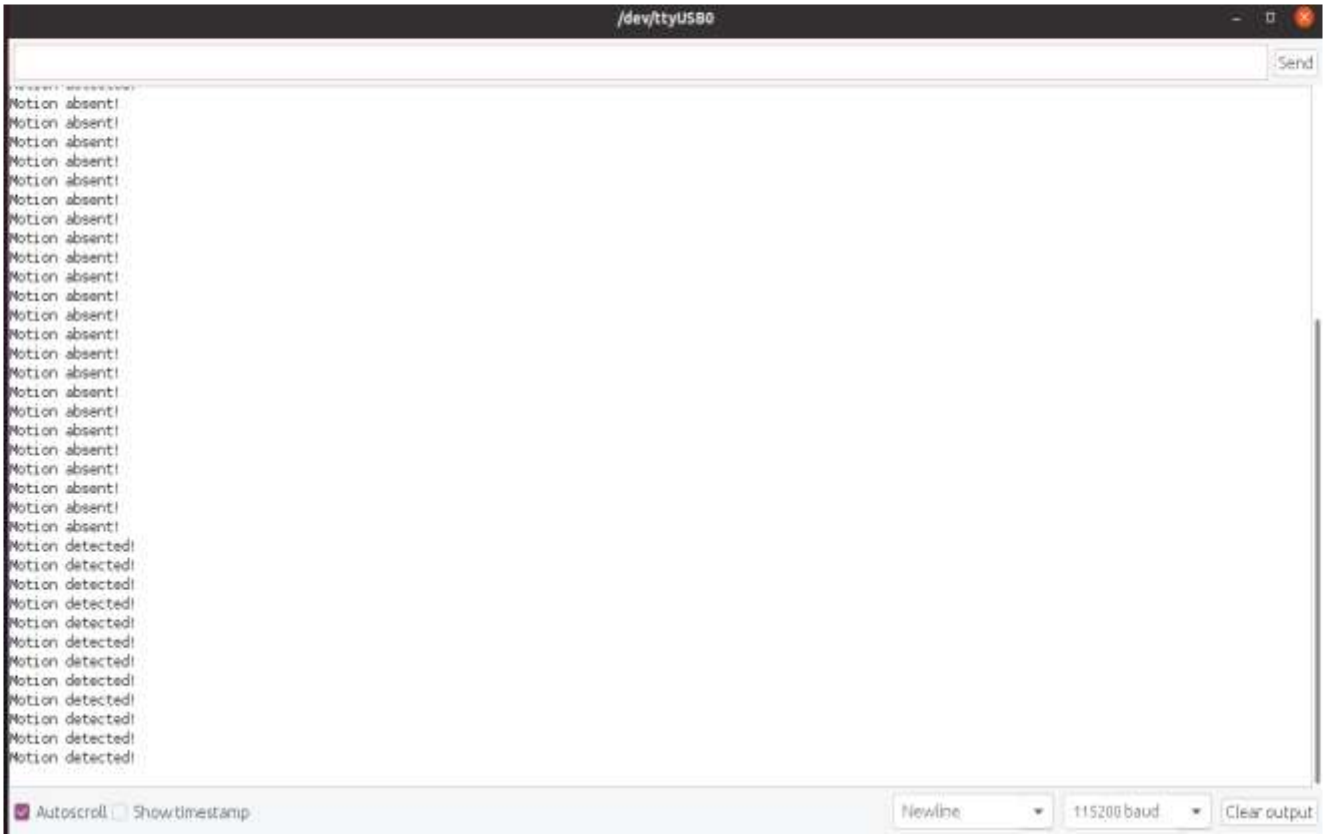

Figure 23: PIR Serial output Source: Arduino based on our farm's data

D. Results of subsystem 4 (LDR sensor)

As the mushroom quality depends on the light intensity in the farm, we also added a light sensor to yield the data about the light variation in the farm for the farmer to take action on that very important factor of mushroom growth.

Figure 24 below shows the yield data in our mushroom farm.

Figure 24: Output of the LDR Source: Arduino based on our farm's data

The serial output of the diagram represents the voltage values. In an averagely lit room, values range from 1- 2. When the LDR is illuminated, values range to a high of 4.67. When the LDR is covered to emulate a mushroom room, the value drops to a low 0.17. A bright room will give a reading in the range of 3-5 V. A dark room will give readings in the range of 0-1.5. Other values that do not fall in the criteria are considered average.

E. Results of subsystem 5 (Output of the cloud platform, Thingspeak)

Figure 25 below shows the data entries made over a while graph. The graph is a visualization in ThingSpeak that indicates the status of sensor values viewed continuously over a given period. The data from sensors sent to the cloud from different entries is different which explains the up/ down lines in the graphs.

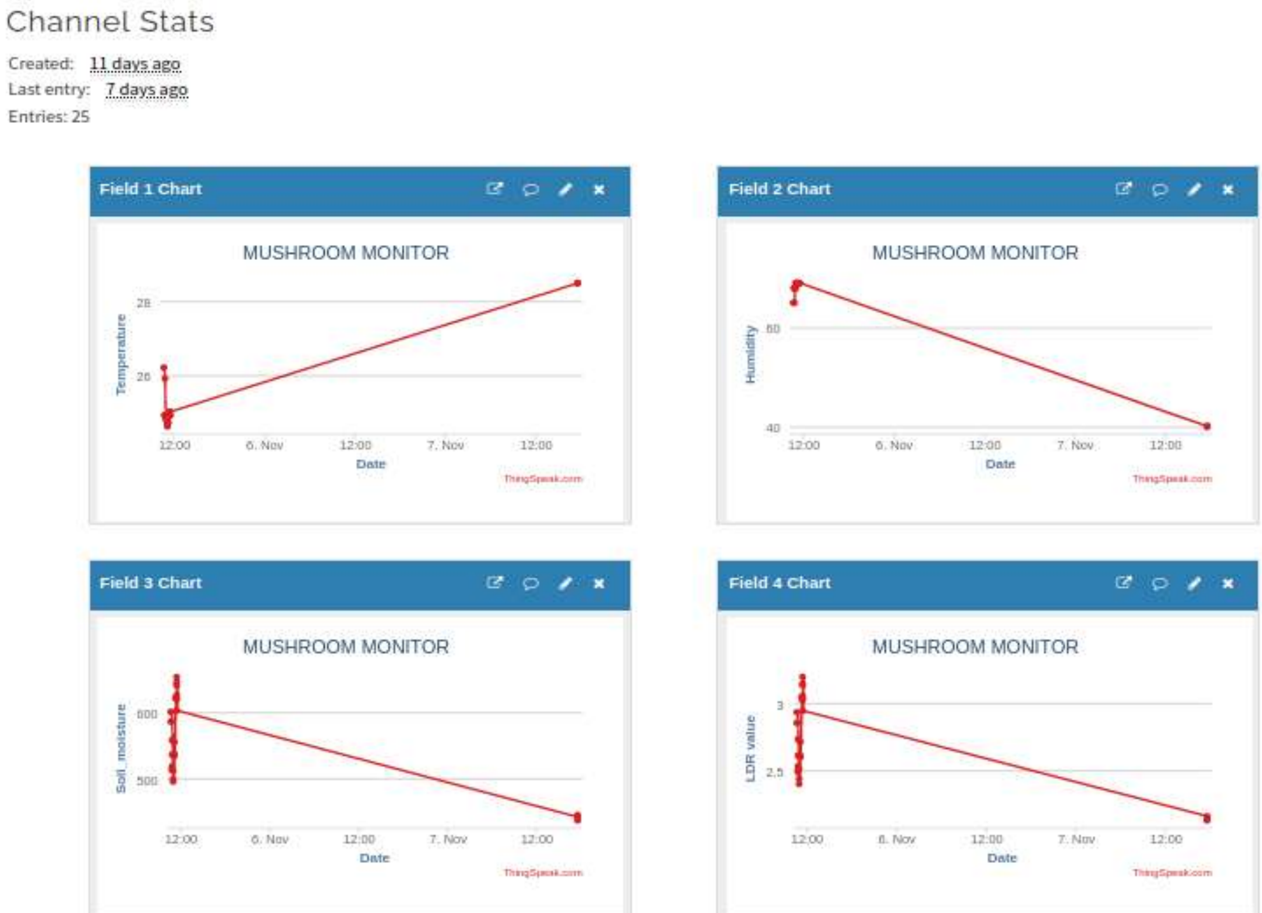

Figure 25: Thingspeak visualization Source: Thingspeak application

Figure 25 above, shows results displayed on the cloud platform. The variation displayed is as depicted from sensor values gathered at different environmental conditions. From the output, we can deduce that when the temperature is low the humidity and soil moisture content is high. Moisture content is therefore affected by the amount of heat in the environment. The LDR displays the amount of light that is not dependent on the other three variables.

These values show the amount of light we exposed the mushroom firm in the range of 0-5 V where 5 represents the highest light intensity and 0 shows lack of light.

4.3 Results of the system

The working of the code is based on the system which can convert sensor data into digital input. The parameters selected are fed in as inputs as sensor data. The soil moisture, humidity, and temperature are relevant inputs for the fuzzy inference system.

The system can capture these inputs and intelligently make a decision that will determine a watering period in second as the output. Further, these sensed values are sent to a cloud platform that can help us to visually see the behavior and trend over a while.

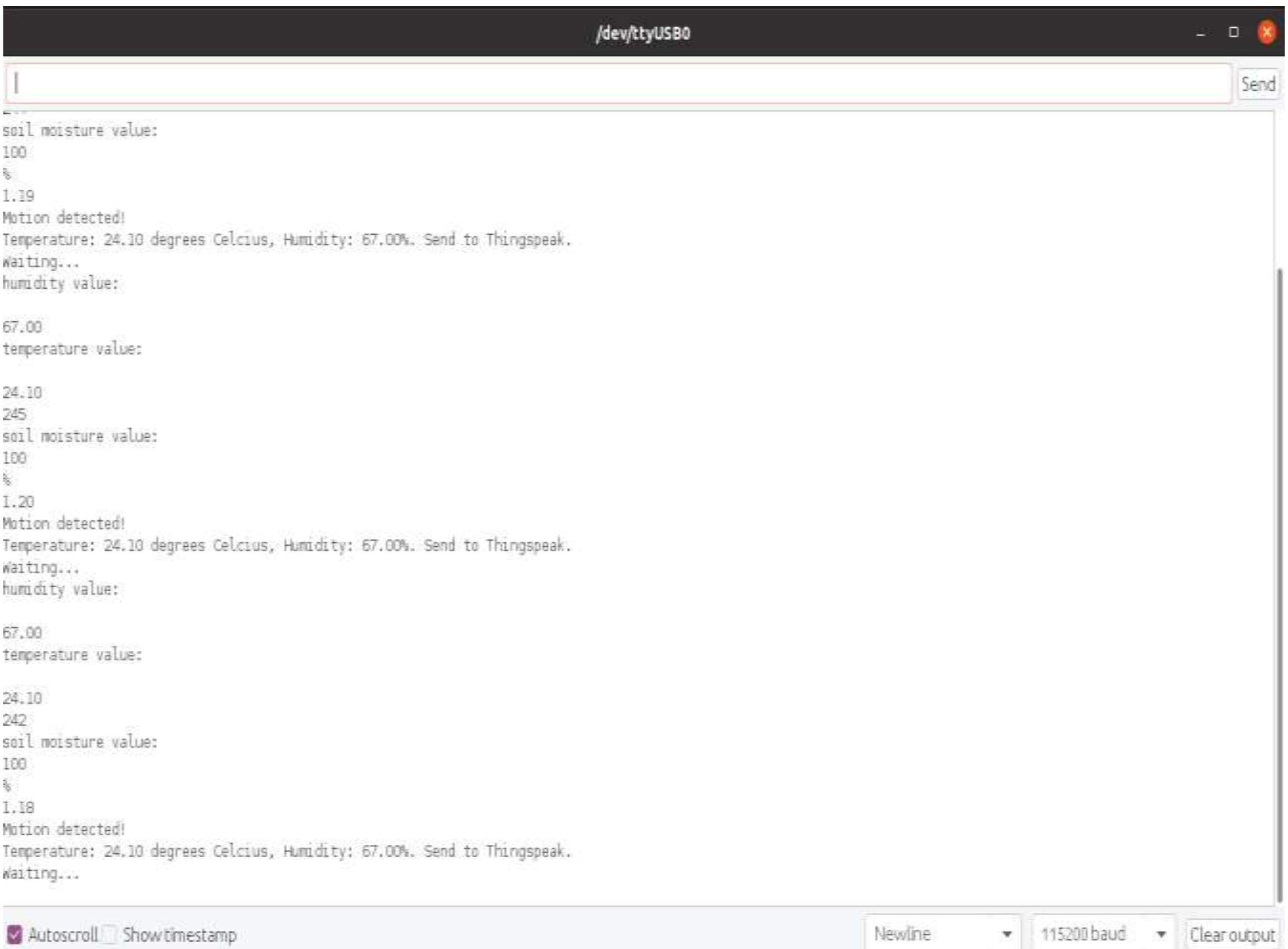

Figure 26: Sensors fusion Serial Output Source: Arduino based on our farm

The water pump achieves up to 1.6 L/min. This value can be used to calculate the volume of water dispensed by the overall system. As per figure 27, the watering duration is 30 seconds which means that 0.8 L was dispensed from the overall system.

The results achieved from this were as follows:

- Conservation of water. Based on the soil moisture content, whenever the system detected high amounts of water the pump remains closed. Tradition techniques that require visual aid to determine if the soil needs to be watered prove to be ineffective which leads to soil moisture going above 90% or under 80%. The use of an IoT system helps to regulate the soil moisture content by maintaining the soil moisture content at the recommended levels.

  Based on the duty scheduling the system can regulate environmental conditions up to 95% of the monitoring duration. This metric is measured by taking into account the number of times during the day the environmental conditions stayed within the recommended ranges against the total duration, that is,

  (Hours conditions stayed in range/No of hours in a day) *100
- Increase of harvest yield

Table showing number of mushrooms collected in a week measuring traditional means against an IoT-based system on two substrates of 10 seeds per each.

Week	Ready to Harvest Seeds (With traditional means)	Ready to Harvest Seeds (IoT based monitoring)
2	0	2
3	3	10
4	8	13
5	12	23

Table 2: Comparison of the harvest between traditional and IoT based means

Source: Own drawing based on farms surveillance data.

The table2 above shows the increase in output when compared with traditional means with IoT-based farm monitoring. Fuzzy logic in automating the control of the watering time saves water quantity because the water pump opens on a schedule based on data gotten from the field. Thanks to an intelligent system, we can regulate the volume of water especially when the conditions are favorable.

With the traditional fixed time watering system, the pump gets open always based on time intervals that can make much water used without being needed based on the farm parameters or climate.

Further as shown in figure 26 above, the combination of the subsystem outputs shows that the subsystem components can be integrated harmoniously. The sensor fusion shows accurate readings from various sensors. This is to prove that combining the sensors the values acquired are not a result of interference from other subsystem components.

This is useful to show that sensor combination can give an accurate view of the environment which can further be utilized as inputs for an intelligent processing fuzzy module. We can accurately monitor the exact values being transmitted to the cloud platform, as well as the start and the end of transmission.

4.4 Shiitake mushroom farms after 21 days

Within this research, two types of mushroom farms are experimentally implemented, where the two farms are equal in the number of seeds with the same type because they have been got from the same nursery. Two substrates are made with the same materials and components, but the difference resides in the technologies used to grow the seeds after plantation because one farm uses our fuzzy-based technology and another uses a watering system based on time interval watering.

Figures 27, and 28 below, show experimental prototypes of shiitake mushroom farms on the 21^{st} day after planting.

Figure 27: Shiitake Mushroom farm monitored with IoT Source: Picture was taken from our pilot farm

Figure 28: Shiitake Mushroom farm based on time-interval watering

Source: The picture was taken from our pilot farm

As seen in figures 27 and 28 above, the harvest after 21 days on the farm with technology is 100% because all the 10 seed bags in the farm have mushrooms with more than one mushroom on one bag but in the farm with time-intervals watering, only 3 seed bags have mushrooms which mean 30% of the total 10 seed bags farm have mushrooms.

From these practical results obtained, we can conclude that our approach of using IoT in shiitake mushroom cultivation is better when compared to the existing time interval watering-based approach.

CHAPTER FIVE: CONCLUSION AND RECOMMENDATIONS

In this chapter, we draw out conclusion based on the obtained results based on the consulted documents and proposed methodology. Then, we give suggestions for future research in smart agriculture.

5.1 Conclusion

In this thesis, an IoT fuzzy logic-based approach scheme was developed for shiitake mushrooms with an automated watering system to monitor on an automatic basis a mushroom farm.

Based on related literature examined, the currently used techniques in mushroom farming are mainly manual and showed that the harvest is not sufficient and water resource use is not optimized. Most of the types of mushrooms are important for medical and nutritional ingredients, but shiitake mushroom is the most needed type of mushroom because of its nature to fit too many medicinal drugs ingredients, and many nutritional ingredients and hence this specific type of mushroom deserves attention in increasing its harvest. Our research focuses on using IoT-based technology to increase harvest and water use optimization.
This research was done in two phases namely requirements gathering and analysis and prototype building.
To collect data about the requirements, we visited a mushroom farm called CP&T LTD at Ndera and observed the process of shiitake mushrooms cultivation as well as the conditions that this particular type of mushroom imposes for harvest improvement. In addition to the real data got from the field, we also consulted documents especially [10] that are about shiitake mushrooms cultivation.

Our prototype was built using IoT technology relying on the data about the requirements for shiitake mushroom cultivation. The built prototype implements fuzzy logic to monitor the farm's vital parameters namely air humidity, temperature, and soil moisture in the substrate.

The obtained results demonstrate that the use of IoT in shiitake mushroom cultivation increases the yield in shiitake mushrooms compared to traditional methods namely, fixed time watering because, with our internet of things-based approach, we can harvest more than one mushroom from all the 10 seeds at the 21^{st} day after seeds plantation while with the existing fixed time watering at the 21^{st} day we can only get the mushrooms on 3 seed bags.

Based on the results of our experiment, the farm without IoT was able to produce mushrooms on all of its 10 seed bags after 35 days while with the technology-based farm it was time to get the second harvest of the 10 seed bags. Further, using fuzzy logic in automating the control of the watering time, saves water quantity because the water pump opens based on data got from the field, which means that when the farm parameters are favorable, a pump can remain closed for more than two days which saves water. With the traditional fixed time watering system, the pump gets open always based on time intervals that can make much water used without being needed based on the farm parameters or climate.

5.2 Recommendations

Regarding the results, we recommend that this study should be implemented with many other types of mushrooms to improve the harvest of mushrooms because they are the crucial inputs for both nutritional and medicinal use and their harvest is still low.

In addition, this research can be improved by controlling other factors of the mushroom farm such as wind speed and direction, water type and temperature control, and also fuzzy type II can be used to handle with precision those suggested farm factors.

References

[1] Ramona Ojha, “Wireless Sensor Networks for Agriculture: The State-of-the-Art in Practice and Future Challenges,” *IEEE Transactions on Instrumentation and Measurement,* vol. 63, no. 1, 2015.

[2] P. P. Ray, “Internet of things for smart agriculture: Technologies, practices and future direction,” *Journal of Ambient Intelligence and Smart Environments,* vol. 9, p. 395, 2017.

[3] A. Kazy Noor, “Optimal Dynamic Routing Protocols for Agro-Sensor Communication in MANETs,” *International Conference on Wireless Communications, Vehicular Technology, Information Theory and Aerospace & Electronic Systems (VITAE),* vol. 7, no. 3, pp. 53-62, 2019.

[4] J. H. J. Balendonck, “Sensors and Wireless Sensor Networks for Irrigation Management under Deficit Conditions (FLOW-AID),” *International Conference on Agricultural Engineering.*, pp. 301-3012, August 2018.

[5] C. D. Moreno-Moreno, “Wireless Sensor Network for Sustainable Agriculture,” *Environment, Green Technology and Engineering International Conference (EGTEIC 2018), Caceres, Spain.*, vol. 50, pp. 1-14, 23 October 2018.

[6] W. C. Michael Crigler, Speciality mushroom cultivation in the northeast of the United States, Northeast: Northeast Sustainable Agriculture Research, 2019.

[7] S. Sierigk, Best Management Practices for Log-Based Shiitake Cultivation in the northeastern united states, Burlington, Vt.: Elsevier, 2013.

[8] J. d. Cruz-del, “Fuzzy Logic-based Controlled Environment for the Production of Oyster Mushroom,” *IEEE 11th International Conference on Humanoid, Nanotechnology, Information Technology, Communication and Control, Environment, and Management (HNICEM),* 2019.

[9] C. H. L. Mehreen Zeb, “Medicinal Properties and Bioactive Compounds from Wild Mushrooms Native to North America,” *Molecules,* vol. 26, no. 251, January 2021.

[10] L. S. Andrew Ravlikovsky, “Agroecological Aspects of Cultivation Shiitake Mushroom in Ukraine,” in *International Council on Technologies of Environmental Protection (ICTEP),* Starý Smokovec, Slovakia, 2019.

[11] Sina Faizollahzadeh Ardabili, “Modeling and comparison of fuzzy and on/off the controller in a mushroom growing hall,” *Elsevier Ltd,* vol. 90, p. 127–134, 2016.

[12] Yousef E., “Optimally Heterogeneous Irrigation for Precision Agriculture Using Wireless Sensor Networks,” *Arabian Journal for Science and Engineering,* vol. 44, p. 3183–3195, 2019.

[13] B. I. Setiawan, “Designing a temperature control system for mushroom cultivation,” *The Biological and Cultivation of Edible Mushrooms. Mc Graw Hill., New York.*, 2006.

[14] A. Najmurrokhman, “Design and Implementation of Temperature and Humidity Control System in Oyster Mushroom Cultivation using Fuzzy Logic Controller,” *International Conference on Computer, Control, Informatics and its Applications,* 2019.

[15] A. A. Ahmed, “A distributed system for supporting smart irrigation using Internet of Things technology,” *John Wiley & Sons Ltd.,* vol. 4, 2020.

[16] C. A. Kamienski, “Smart Water Management Platform Overview and Security challenges,” *Journal of Network and Computer Applications,* pp. 72-78, 2018.

[17] S. Mathew, “Monitoring and Controlling of the greenhouse using Microcontroller,” *International Journal of Pure and Applied Mathematics,* vol. 118, no. 20, pp. 585-590, 2018.

[18] Pravinthraja.S, “IoT Based Mushroom Monitoring System-A survey,” *International Journal of Recent Research Aspects,* vol. 5, no. 1, pp. 311-314, 2018.

[19] Sampada Singh, “Smart Mushroom Cultivation using IoT,” *International Journal of Engineering Research & Technology (IJERT),* 2020.

[20] A. Nalini, “Automatic mushroom growth monitoring using IoT,” *International journal of creative research thoughts (IJCRT),* vol. 8, no. 3, 2020.

[21] S.Pravinth Raja, “Intelligent Mushroom Monitoring System,” *International Journal of Engineering & Technology,* vol. 7, pp. 1238-1242, 2018.

[22] Tran Manh Tuan, “M-CFIS-R: Mamdani Complex Fuzzy Inference System with Rule Reduction Using Complex Fuzzy Measures in Granular Computing,” *Int. J. Fuzzy Log. Intell. Syst.,* vol. 8, no. 7, 2020.

[23] G. Verma, “A survey on hardware design issues in RF energy harvesting for wireless sensor networks (WSN),” *IEEE Trans. Pattern Anal. Mach. Intell.,* 2016.

[24] F. Jonard, “Ground-Based Soil Moisture Determination,” *IEEMA Engineer Infinite,* pp. 25-43, 2018.

[25] L. G. Pisana Placidi, “Characterization of Low-Cost Capacitive Soil Moisture Sensors for IoT Networks,” *Journal of Hydrology,* 2020.

[26] C. I. A. Lakhiar, “Monitoring and Control Systems in Agriculture Using Intelligent,” *Hindawi Journal of Sensors,* vol. 5, pp. 1-17, 2018.

[27] Debajit Saha, “Explosive sensing with insect-based biorobots,” *Journal of Biosensors and Bioelectronics,* 2020.

[28] M. S. V. Surve, “Arduino Based Pesticide Spraying Robot,” *International Research Journal of Engineering and Technology (IRJET),* vol. 06, no. 12, 2019.

Appendix: Codes for the Project Implementation

```
#include "ESP8266WiFi.h"
#include <ThingSpeak.h>
#include <SPI.h>
#include <Wire.h>
#include <DHT.h>
#include "fis_header.h"

// Number of inputs to the fuzzy inference system
const int fis_gcI = 3;
// Number of outputs to the fuzzy inference system
const int fis_gcO = 1;
// Number of rules to the fuzzy inference system
const int fis_gcR = 6;

FIS_TYPE g_fisInput[fis_gcI];
FIS_TYPE g_fisOutput[fis_gcO];

String apiKey = "GVKA8UPIEXB5XUBS";     //  Enter your Write API key from ThingSpeak

const char *ssid =  "Retbin Inc";     // replace with your wifi ssid and wpa2 key
const char *pass =  "kibuye@36";
const char* server = "api.thingspeak.com";
WiFiClient client;

#define DHTPIN D2
#define DHTTYPE DHT11
```

```
const int AirValue = 790;   //you need to replace this value with Value_1
const int WaterValue = 390;  //you need to replace this value with Value_2
const int SensorPin = A0;
int soilMoistureValue = 0;
int soilmoisturepercent=0;
const int LDRpin = A0;
const int RELAY_PIN = 7;

int pir_sensor = D5;
bool motion = "FALSE";

int Status = 12;  // Digital pin D6

int sensor = D5;  // Digital pin D7

void setup() {
  Serial.begin(115200);
  pinMode(RELAY_PIN, OUTPUT);
  pinMode(pir_sensor, INPUT);
  Serial.println("WiFi connected");
  dht.begin();
       Serial.println("Connecting to ");
       Serial.println(ssid);
```

```
        WiFi.begin(ssid, pass);

       while (WiFi.status() != WL_CONNECTED)
      {
             delay(500);
             Serial.print(".");
      }
       Serial.println("");
       Serial.println("WiFi connected");

  pinMode(sensor, INPUT);   // declare sensor as input
  pinMode(Status, OUTPUT);  // declare LED as output
}


void loop()
{
  //DHT code
  float h = dht.readHumidity();
  // Read temperature as Celsius (the default)
  float t = dht.readTemperature();
   Serial.println("humidity value:\n");
  Serial.println(h);
  Serial.println("temperature value:\n");
```

```
//soil moisture code
soilMoistureValue = analogRead(SensorPin);  //put Sensor insert into soil
Serial.println(soilMoistureValue);
soilmoisturepercent = map(soilMoistureValue, AirValue, WaterValue, 0, 100);
Serial.println("soil moisture value:\t");

if(soilmoisturepercent > 100)
{
  soilmoisturepercent=100;

}
else if(soilmoisturepercent <0)
{
  soilmoisturepercent=0;

}
else if(soilmoisturepercent >=0 && soilmoisturepercent <= 100)
{
  soilmoisturepercent=soilmoisturepercent;
}

Serial.println(soilmoisturepercent);
Serial.println("%");

//LDR
```

```
  float voltage = sensorValue * (5.0 / 1023.0);   // Convert the analog reading (which goes from 0 - 1023)

  Serial.println(voltage);   // print out the value you read

  long state = digitalRead(sensor);
    if(state == HIGH) {
      digitalWrite (Status, HIGH);
      Serial.println("Motion detected!");
      delay(1000);
    }
    else {
      digitalWrite (Status, LOW);
      Serial.println("Motion absent!");
      delay(1000);
      }
// Begin fuzzy inferencing
    // Read Input: Temp
    g_fisInput[0] = t;
    // Read Input: Soil_moisture
    g_fisInput[1] = soilMoistureValue;
    // Read Input: Humidity
    g_fisInput[2] = h;

    g_fisOutput[0] = 0;
```

```
    g_fisOutput[0] = 0;

    fis_evaluate();
    Serial.println("Watering period is:");
    Serial.println(g_fisOutput[0]);
    Serial.println("seconds");
    delay(5000);

  digitalWrite(RELAY_PIN, HIGH); // turn on pump 5 seconds
  delay(1000);
  digitalWrite(RELAY_PIN, LOW);  // turn off pump 5 seconds
  delay(g_fisOutput[0]*1000);

          send_to_cloud(h,t,soilMoistureValue, voltage);
          Serial.println("Waiting...");
          delay(5000);

}


void send_to_cloud(float h, float t, int soilMoistureValue, float voltage)
{
if (client.connect(server,80))   //   "184.106.153.149" or api.thingspeak.com
{

                                 String postStr = apiKey;
```

```
postStr += String(t);
postStr +="&field2=";
postStr += String(h);
postStr +="&field3=";
postStr += String(soilMoistureValue);
postStr +="&field4=";
postStr += String(voltage);
//postStr +="&field5=";
//postStr += String(motion);
postStr += "\r\n\r\n";

client.print("POST /update HTTP/1.1\n");
client.print("Host: api.thingspeak.com\n");
client.print("Connection: close\n");
client.print("X-THINGSPEAKAPIKEY: "+apiKey+"\n");
client.print("Content-Type: application/x-www-form-urlencoded\n");
client.print("Content-Length: ");
client.print(postStr.length());
client.print("\n\n");
client.print(postStr);

Serial.print("Temperature: ");
Serial.print(t);
Serial.print(" degrees Celcius, Humidity: ");
Serial.print(h);
Serial.println("%. Send to Thingspeak.");
```

```
        }
    client.stop();
}


//*******************************************************************
// Support functions for Fuzzy Inference System
//*******************************************************************
// Triangular Member Function
FIS_TYPE fis_trimf(FIS_TYPE x, FIS_TYPE* p)
{
    FIS_TYPE a = p[0], b = p[1], c = p[2];
    FIS_TYPE t1 = (x - a) / (b - a);
    FIS_TYPE t2 = (c - x) / (c - b);
    if ((a == b) && (b == c)) return (FIS_TYPE) (x == a);
    if (a == b) return (FIS_TYPE) (t2*(b <= x)*(x <= c));
    if (b == c) return (FIS_TYPE) (t1*(a <= x)*(x <= b));
    t1 = min(t1, t2);
    return (FIS_TYPE) _max(t1, 0);
}

FIS_TYPE fis_min(FIS_TYPE a, FIS_TYPE b)
{
    return min(a, b);
```

```
{
    return min(a, b);
}

FIS_TYPE fis_max(FIS_TYPE a, FIS_TYPE b)
{
    return max(a, b);
}

FIS_TYPE fis_array_operation(FIS_TYPE *array, int size, _FIS_ARR_OP pfnOp)
{
    int i;
    FIS_TYPE ret = 0;

    if (size == 0) return ret;
    if (size == 1) return array[0];

    ret = array[0];
    for (i = 1; i < size; i++)
    {
        ret = (*pfnOp)(ret, array[i]);
    }

    return ret;
}

//***********************************************************************
// Data for Fuzzy Inference System
//***********************************************************************
// Pointers to the implementations of member functions
_FIS_MF fis_gMF[] =
{
    fis_trimf
};

// Count of member function for each Input
int fis_gIMFCount[] = { 3, 3, 3 };

// Count of member function for each Output
int fis_gOMFCount[] = { 3 };

// Coefficients for the Input Member Functions
FIS_TYPE fis_gMFI0Coeff1[] = { 0, 10, 22 };
FIS_TYPE fis_gMFI0Coeff2[] = { 22, 26, 30 };
FIS_TYPE fis_gMFI0Coeff3[] = { 30, 100, 141.7 };
FIS_TYPE* fis_gMFI0Coeff[] = { fis_gMFI0Coeff1, fis_gMFI0Coeff2, fis_gMFI0Coeff3 };
FIS_TYPE fis_gMFI1Coeff1[] = { 0, 20, 30 };
FIS_TYPE fis_gMFI1Coeff2[] = { 30, 40, 65 };
FIS_TYPE fis_gMFI1Coeff3[] = { 65, 80, 100 };
FIS_TYPE* fis_gMFI1Coeff[] = { fis_gMFI1Coeff1, fis_gMFI1Coeff2, fis_gMFI1Coeff3 };
FIS_TYPE fis_gMFI2Coeff1[] = { 0, 50, 80 };
```

```
FIS_TYPE fis_gMFI2Coeff2[] = { 80, 85, 90 };
FIS_TYPE fis_gMFI2Coeff3[] = { 90, 100, 140 };
FIS_TYPE* fis_gMFI2Coeff[] = { fis_gMFI2Coeff1, fis_gMFI2Coeff2, fis_gMFI2Coeff3 };
FIS_TYPE** fis_gMFICoeff[] = { fis_gMFI0Coeff, fis_gMFI1Coeff, fis_gMFI2Coeff };

// Coefficients for the Output Member Functions
FIS_TYPE fis_gMFO0Coeff1[] = { -25, 0, 25 };
FIS_TYPE fis_gMFO0Coeff2[] = { 20, 30, 45 };
FIS_TYPE fis_gMFO0Coeff3[] = { 35, 60, 85 };
FIS_TYPE* fis_gMFO0Coeff[] = { fis_gMFO0Coeff1, fis_gMFO0Coeff2, fis_gMFO0Coeff3 };
FIS_TYPE** fis_gMFOCoeff[] = { fis_gMFO0Coeff };

// Input membership function set
int fis_gMFI0[] = { 0, 0, 0 };
int fis_gMFI1[] = { 0, 0, 0 };
int fis_gMFI2[] = { 0, 0, 0 };
int* fis_gMFI[] = { fis_gMFI0, fis_gMFI1, fis_gMFI2};

// Output membership function set
int fis_gMFO0[] = { 0, 0, 0 };
int* fis_gMFO[] = { fis_gMFO0};

// Rule Weights
FIS_TYPE fis_gRWeight[] = { 1, 1, 1, 1, 1, 1 };

// Rule Type
int fis_gRType[] = { 1, 1, 1, 1, 2, 1 };

// Rule Inputs
int fis_gRI0[] = { 3, 1, 1 };
int fis_gRI1[] = { 3, 2, 2 };
int fis_gRI2[] = { 1, 3, 3 };
int fis_gRI3[] = { 1, 1, 1 };
int fis_gRI4[] = { 1, 1, 1 };
int fis_gRI5[] = { 2, 2, 2 };
int* fis_gRI[] = { fis_gRI0, fis_gRI1, fis_gRI2, fis_gRI3, fis_gRI4, fis_gRI5 };

// Rule Outputs
int fis_gRO0[] = { 3 };
int fis_gRO1[] = { 2 };
int fis_gRO2[] = { 1 };
int fis_gRO3[] = { 1 };
int fis_gRO4[] = { 1 };
int fis_gRO5[] = { 1 };
int* fis_gRO[] = { fis_gRO0, fis_gRO1, fis_gRO2, fis_gRO3, fis_gRO4, fis_gRO5 };

// Input range Min
FIS_TYPE fis_gIMin[] = { 0, 0, 0 };

// Input range Max
FIS_TYPE fis_gIMax[] = { 100, 100, 100 };
```

```
// Output range Min
FIS_TYPE fis_gOMin[] = { 0 };

// Output range Max
FIS_TYPE fis_gOMax[] = { 60 };

//***********************************************************************
// Data dependent support functions for Fuzzy Inference System
//***********************************************************************
FIS_TYPE fis_MF_out(FIS_TYPE** fuzzyRuleSet, FIS_TYPE x, int o)
{
    FIS_TYPE mfOut;
    int r;

    for (r = 0; r < fis_gcR; ++r)
    {
        int index = fis_gRO[r][o];
        if (index > 0)
        {
            index = index - 1;
            mfOut = (fis_gMF[fis_gMFO[o][index]])(x, fis_gMFOCoeff[o][index]);
        }
        else if (index < 0)
        {
            index = -index - 1;
            mfOut = 1 - (fis_gMF[fis_gMFO[o][index]])(x, fis_gMFOCoeff[o][index]);
```

mfOut = 1 - (fis_gMF[fis_gMFO[o][index]])(x, fis_gMFOCoeff[o][index]);

```
        }
        else
        {
            mfOut = 0;
        }

        fuzzyRuleSet[0][r] = fis_min(mfOut, fuzzyRuleSet[1][r]);
    }
    return fis_array_operation(fuzzyRuleSet[0], fis_gcR, fis_max);
}

FIS_TYPE fis_defuzz_centroid(FIS_TYPE** fuzzyRuleSet, int o)
{
    FIS_TYPE step = (fis_gOMax[o] - fis_gOMin[o]) / (FIS_RESOLUSION - 1);
    FIS_TYPE area = 0;
    FIS_TYPE momentum = 0;
    FIS_TYPE dist, slice;
    int i;

    // calculate the area under the curve formed by the MF outputs
    for (i = 0; i < FIS_RESOLUSION; ++i){
        dist = fis_gOMin[o] + (step * i);
        slice = step * fis_MF_out(fuzzyRuleSet, dist, o);
        area += slice;
```

```
        momentum += slice*dist;
    }

    return ((area == 0) ? ((fis_gOMax[o] + fis_gOMin[o]) / 2) : (momentum / area));
}

//***********************************************************************
// Fuzzy Inference System
//***********************************************************************
void fis_evaluate()
{
    FIS_TYPE fuzzyInput0[] = { 0, 0, 0 };
    FIS_TYPE fuzzyInput1[] = { 0, 0, 0 };
    FIS_TYPE fuzzyInput2[] = { 0, 0, 0 };
    FIS_TYPE* fuzzyInput[fis_gcI] = { fuzzyInput0, fuzzyInput1, fuzzyInput2, };
    FIS_TYPE fuzzyOutput0[] = { 0, 0, 0 };
    FIS_TYPE* fuzzyOutput[fis_gcO] = { fuzzyOutput0, };
    FIS_TYPE fuzzyRules[fis_gcR] = { 0 };
    FIS_TYPE fuzzyFires[fis_gcR] = { 0 };
    FIS_TYPE* fuzzyRuleSet[] = { fuzzyRules, fuzzyFires };
    FIS_TYPE sW = 0;

    // Transforming input to fuzzy Input
    int i, j, r, o;
    for (i = 0; i < fis_gcI; ++i)
    {
        for (j = 0; j < fis_gIMFCount[i]; ++j)
        {
            fuzzyInput[i][j] =
                (fis_gMF[fis_gMFI[i][j]])(g_fisInput[i], fis_gMFICoeff[i][j]);
        }
    }

    int index = 0;
    for (r = 0; r < fis_gcR; ++r)
    {
        if (fis_gRType[r] == 1)
        {
            fuzzyFires[r] = FIS_MAX;
            for (i = 0; i < fis_gcI; ++i)
            {
                index = fis_gRI[r][i];
                if (index > 0)
                    fuzzyFires[r] = fis_min(fuzzyFires[r], fuzzyInput[i][index - 1]);
                else if (index < 0)
                    fuzzyFires[r] = fis_min(fuzzyFires[r], 1 - fuzzyInput[i][-index - 1]);
                else
                    fuzzyFires[r] = fis_min(fuzzyFires[r], 1);
            }
        }
```

```
            }
        }
        else
        {
            fuzzyFires[r] = FIS_MIN;
            for (i = 0; i < fis_gcI; ++i)
            {
                index = fis_gRI[r][i];
                if (index > 0)
                    fuzzyFires[r] = fis_max(fuzzyFires[r], fuzzyInput[i][index - 1]);
                else if (index < 0)
                    fuzzyFires[r] = fis_max(fuzzyFires[r], 1 - fuzzyInput[i][-index - 1]);
                else
                    fuzzyFires[r] = fis_max(fuzzyFires[r], 0);
            }
        }

        fuzzyFires[r] = fis_gRWeight[r] * fuzzyFires[r];
        sW += fuzzyFires[r];
    }

    if (sW == 0)
    {
        for (o = 0; o < fis_gcO; ++o)
        {
            g_fisOutput[o] = ((fis_gOMax[o] + fis_gOMin[o]) / 2);
```

```
    if (sW == 0)
    {
        for (o = 0; o < fis_gcO; ++o)
        {
            g_fisOutput[o] = ((fis_gOMax[o] + fis_gOMin[o]) / 2);
        }
    }
    else
    {
        for (o = 0; o < fis_gcO; ++o)
        {
            g_fisOutput[o] = fis_defuzz_centroid(fuzzyRuleSet, o);
        }
    }
}
```