UNIVERSITI
MALAYSIA
UniMAP PERLIS

EKT360- GROUP DESIGN PROJECT
Session 2020/2021 Semester II
Group 6

Controlling Temperature and Humidity for Oyster Mushroom Cultivation System

NAME	MATRIC NUMBER	PROGRAM
SAM JUN AN	181021172	RK20
NUR FIKRY BIN NORZURI	181021136	RK20
MUHAMMAD SHUKRI BIN MAT ZAHID	181021115	RK20
MOHAMED ABDULRAQEEB MOHMD	181020040-5	RK20

SUPERVISOR: Dr. SALINA BT MOHD ASI

TABLE OF CONTENTS

ABSTRACT

Control systems become the most important factor in automation systems. One of the control system methods/models is fuzzy logic modelling. Fuzzy logic is a modern control system that can produce decisions like human logic. Oyster Mushrooms cultivation is a business with many advantages such as growth that does not depend on weather and season; cultivation does not require extensive land and special tools. However, business needs to be optimized so the quantity and quality of production increase. One way to optimize the quantity and quality of production is to keep the temperature and humidity in optimum conditions because temperature humidity is an important factor in the growth of Oyster mushrooms. So far to keep the temperature and humidity, entrepreneurs use the traditional way sprayed with water granules to the Oyster mushrooms, but this is not optimal because humidity can not be maintained at all times. In addition, entrepreneurs must pay for spraying services. The solution to the problem is to create an automatic humidity regulator with a Fuzzy logic control system.

ABSTRAK

Sistem kawalan menjadi faktor terpenting dalam sistem automasi. Salah satu kaedah / model sistem kawalan adalah pemodelan logik kabur. Logik kabur adalah sistem kawalan moden yang dapat menghasilkan keputusan seperti logik manusia. Penanaman Cendawan Tiram adalah perniagaan yang mempunyai banyak kelebihan seperti, pertumbuhan tidak bergantung pada cuaca dan musim, penanaman tidak memerlukan tanah yang luas dan alat khas. Namun, perniagaan perlu dioptimalkan sehingga kuantiti dan kualiti pengeluaran meningkat. Salah satu cara untuk mengoptimumkan kuantiti dan kualiti pengeluaran adalah menjaga suhu dan kelembapan dalam keadaan optimum kerana suhu-kelembapan adalah faktor penting dalam pertumbuhan jamur Tiram. Sejauh ini untuk menjaga suhu dan kelembapan, pengusaha menggunakan cara tradisional yang disembur dengan butiran air ke cendawan tiram, tetapi ini tidak optimum kerana kelembapan tidak dapat dijaga sepanjang masa. Di samping itu, pengusaha mesti membayar perkhidmatan penyemburan. Penyelesaian untuk masalah ini adalah dengan membuat pengatur kelembapan automatik dengan sistem kawalan logik kabur.

ACKNOWLEDGEMENT

First, we would like to thank our university, University Malaysia Perlis (UniMAP) and Faculty of Electronic Engineering Technology (FTKEN), to use our knowledge we had learned throughout these few years of study in university. We want to express our deep and sincere gratitude to our project supervisor, Dr Salina bt Mohd Asi, to allow us to do the project and provide invaluable guidance throughout this project. Besides that, thanks to every group member for giving the best teamwork and cooperation for finishing this project smoothly.

APPROVAL AND DECLARATION

This project report titled Project Title was prepared and submitted by Sam Jun An (181021172), Nur Fikry Bin Norzuri(181021136), Muhammad Shukri Bin Mat Zahid(181021115), Mohamed Abdulraqeeb Mohmd(181020040-5) and has been found satisfactory in terms of scope, quality and presentation as partial fulfillment of the requirement for the Bachelor of Engineering (Computer Engineering) in University Malaysia Perlis (UniMAP).

Checked and Approved by

(Dr. SALINA BT MOHD ASI)
Project Supervisor

(Mdm SABARINA BINTI ISMAIL)
Project Supervisor

Faculty of Electronic Engineering Technology

University Malaysia Perlis

July 2021

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

1.1 Background

Pleurotus ostreatus, sometimes known as oyster mushrooms, is one of the most widely grown mushroom species on the planet. Pearl oyster mushrooms and tree oyster mushrooms are other names for them. The mushrooms may grow naturally on and around trees in temperate and subtropical woods worldwide, and they're also farmed commercially in several nations. Oyster mushrooms are used in various cuisines, although they are particularly popular in Chinese, Japanese, and Korean cuisines. They may be dried and are most commonly consumed cooked. One cup of raw oyster mushrooms has 28 calories, 3 grams of protein, 5 grams of carbs, and no fat. 1 They're high in riboflavin, niacin, phosphorus, potassium, and copper, among other vitamins and minerals. Oyster mushrooms are an excellent nutritional choice for vegetarians and vegans since they are high in protein and other nutrients. Oyster mushrooms are prized for their delicate texture and mild, savoury flavour throughout the world. White, grey, or brown caps with gills lining the bottom characterize the mushrooms with broad, thin, oyster- or fan-shaped caps. The caps have frilly edges and can be found in tiny mushrooms or as single bigger mushrooms.

Oyster mushrooms are more expensive than white button mushrooms, but not as much as rarer mushrooms like morels, and they require little preparation because they may be used whole or sliced. They are also used to build mycelium furniture and a variety of other items. Nowadays, oyster mushrooms have mass production due to low demands and high supply, making the price cheaper and cheaper. However, mushroom farming productivity is still regarded as poor in reality. This is because mushroom growing requires the right temperature and humidity, which is between 22 to 28 ° C, and humidity is between 60% to 80% to provide the best results[11]. While the process of controlling temperature and humidity to achieve ideal mushroom cultivation is currently done in a traditional manner using a manual sprayer, this makes it more difficult for farmers to achieve optimal results, and it takes extra energy to secure the temperature and humidity of the mushroom house during the dry season. On this basis, it is proposed to produce automated equipment to manage temperature and humidity in oyster mushroom farmhouses.

Figure 1.0: Oyster Mushroom Cultivation Farm

Fuzzy logic is a computing technique based on "degrees of truth" rather than the traditional "true or false" (1 or 0) Boolean logic that the modern computer is built on. In the 1960s, Lotfi Zadeh of the University of California at Berkeley proposed fuzzy logic for the first time. Zadeh was working on a computer programme that could interpret natural language. Like most other aspects of life and the world, natural language is difficult to convert into absolute terms such as 0 and 1. Whether everything can be described in binary terms, in the end, is a philosophical topic worth exploring. Still, in reality, most of the data we'd like to give a computer is somewhere in between, as are computing outcomes. It may be helpful to think of fuzzy logic as the true nature of thinking, binary or Boolean, logic as a subset of it.

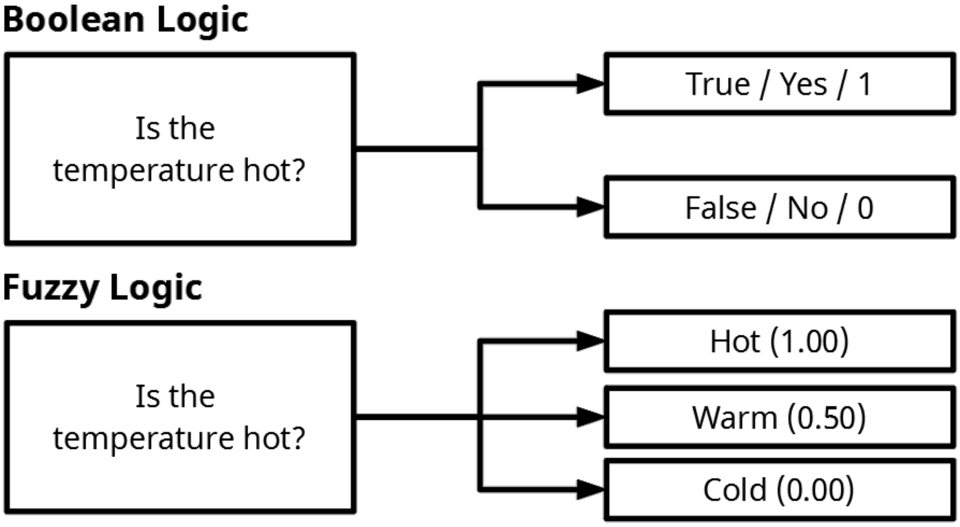


Figure 1.1: Difference between Boolean and Fuzzy Logic

Fuzzy logic is an excellent technique for controlling the humidity and temperature of the oyster mushroom farm. It is because oyster mushrooms are very sensitive to the surrounding humidity and temperature. The surrounding humidity and temperature is not in the suitable range, the mushroom will die instantly. So, a much more sensitive logic which is Fuzzy Logic, is used rather than Boolean Logic.

1.2 Problem Statement

The oyster mushroom is the third most popular cultivated fungus. China, the world's largest producer of oysters, provides approximately 85% of the world's total production of about a million tonnes. Korea, Japan, Italy, Taiwan, Thailand, and the Philippines are among the countries that produce oyster mushrooms. Due to insufficient domestic demand, India's current crop yield is barely around 1500 tonnes. Another impediment is that export demand orders are large and can only be met if a relationship between producers, cooperatives, and exporters is established. Because of the fierce competition, this industry requires optimization to excel and generate the full product. Cultivation is one of the areas where improvements can be made.

Temperature and humidity in the mushroom house are crucial factors in mushroom cultivation. Mushrooms will grow at their best condition if the humidity and temperature are just proper. Unfortunately, many farmers ignore the humidity and temperature in their mushroom houses. However, not the most common cause of crop loss or poor harvest, a lack of monitoring of surrounding temperature and humidity is one of the most common causes. When it comes to humidity control, Farmers typically make the mushroom house humid by spraying the soil with water only in the morning and afternoon. The humidity level is not tested to see if it is adequate. Because many farmers are overworked, they must hire labourers to spray water. The effect of low humidity is that it takes longer to harvest, the frequency with which each medium is harvested drops, and the mushroom may die. About the farmhouse's surrounding temperature, most of the farmhouse's location is built around a country where the temperature is suitable for cultivating oyster mushrooms. But we need to ensure that any country will be able to produce mushrooms, so by just controlling the temperature of the farmhouse, we might be able to achieve that. As a result, we will need a means to keep the mushroom house's temperature and humidity at an appropriate level that can be monitored in real-time.

1.3 Objectives

- To design a temperature and humidity control system by implementing fuzzy logic.

- To design a website and database that allow users to monitor the farmhouse's temperature and humidity.

- Produce automated equipment to manage temperature and humidity in oyster mushroom farmhouses.

1.4 Scope of Project

This project's scope is to explore the coding and learning the use of Python Fuzzy library that have been coded by using Python. We need to implement and modify the code that satisfies the requirements of this project title. Besides that, to allow a user-friendly way for users to retrieve the inputs and outputs of the system, PHP coding is implemented. The purpose of PHP is to enable users to access the data wirelessly.

This project only involves software, but it is distributed into two parts. The primary system is a fuzzy temperature and humidity control system, and the second part is web page design and database access.

CHAPTER 2

METHODOLOGY

2.1 Design Complexity

The entire system is split into two parts. The first part is the temperature and humidity control system which is coded with Python 3. Then for the second part is the website and database, which allow the user to view and store current temperature, humidity, and the output of the system into the database.

Figure 2.0 shows the overall block diagram of the temperature and humidity control system and connected with the database and website. The temperature and humidity control system are coded in Python 3 language and using the library of scikit fuzzy. Meanwhile, the website and database are coded in PHP.

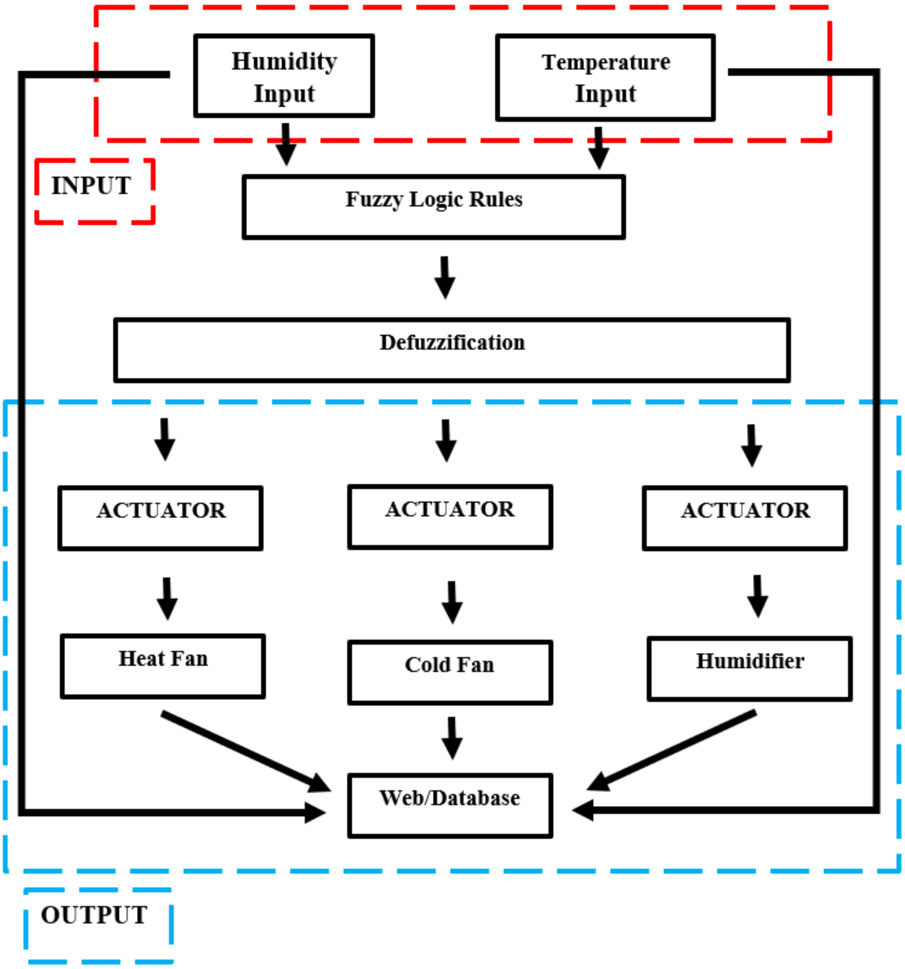


Figure 2.0: Block Diagram of Fuzzy Logic System

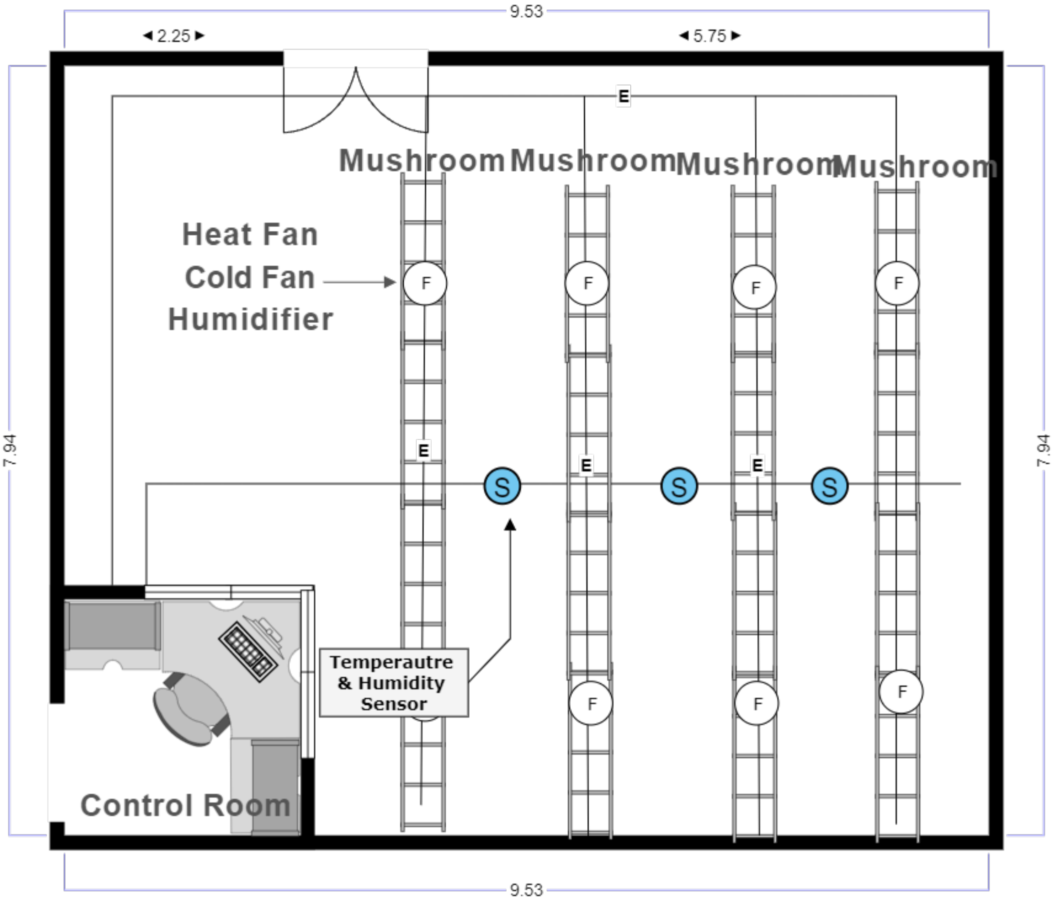


Figure 2.1 Floor Layout of the farmhouse

Figure above is the Floor Layout of the farmhouse, the purpose of this is to ensure the system can work in a suitable environment. From the figure we can see that the green bar is the mushroom plant which we now assuming we have 4 rows. Then for the red line is the heat, cold fan and humidifier. For the blue line is the temperature and humidifier sensor which is used to collect the input data for the system. Input will be sent back to the control room and the output generated will be send out from the control room. The reason why of the control room is in another room rather than the same room with the mushroom farm, it is because the farm normally will stay at the high humidity, the water particle will damage the component of the computer easily. Besides that, all the components should be having water resist feature, it can save a lot of cost on replacing the component.

2.1.1 Software Description

2.1.1.1 PART I -Temperature and Humidity Control System

<u>2.1.1.1.1 Fuzzy Logic (Memberships)</u>

In the universe of discourse, a membership function for a fuzzy set A. Each element of X is mapped to a value between 0 and 1 in the formula A:X [0,1]. A fuzzy set can be represented graphically using membership functions. The x-axis depicts the discourse universe, whereas the y axis shows the degrees of membership in the [0,1] interval. Membership functions are built using simple functions. Because we are defining fuzzy ideas, utilizing more sophisticated functions will not help us get any closer to our goal.

A lower limit a, an upper limit d, a lower support limit b, and an upper support limit c create a trapezoidal function, where a b c d.

$$\mu_A(x) = \begin{cases} 0, & (x < a) \; or \; (x > d) \\ \frac{x-a}{b-a}, & b \le x \le c \\ 1, & c \le x \le d \end{cases} \tag{2.1}$$

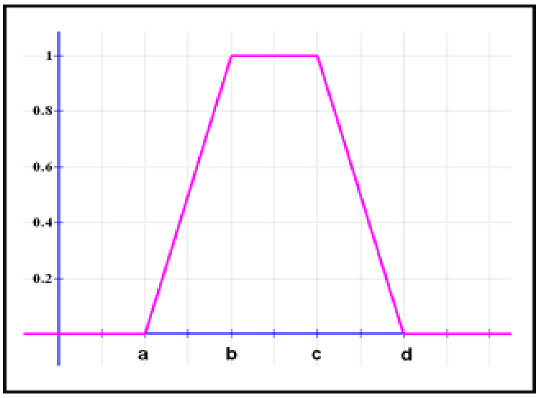


Figure 2.2: Trapezoidal function graph

There are two special cases of a trapezoidal function, which are called R-functions and L-functions:

R-functions: with parameters a = b = $-\infty$

$$\mu_A(x) = \begin{cases} 0, \; x < d \\ \frac{d-x}{d-c}, \; c \le x \le d \\ 1, \; x < c \end{cases} \tag{2.2}$$

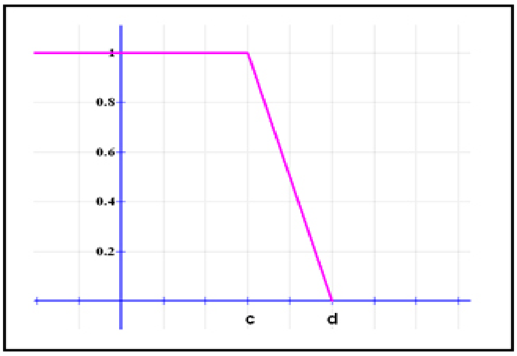


Figure 2.3: R-Trapezoidal function graph

L-Functions: with parameters c = d = + ∞

$$\mu_A(x) = \begin{cases} 0, \; x < a \\ \frac{x-a}{b-a}, \; a \leq x \leq b \\ 1, \; x > b \end{cases} \tag{2.3}$$

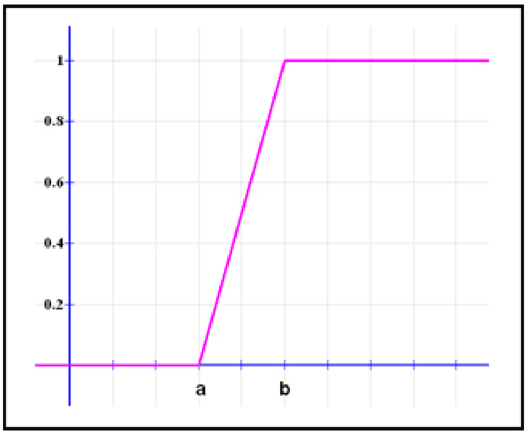


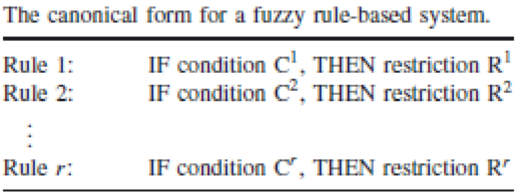
Figure 2.4: L-Trapezoidal function graph

2.1.1.1.2 Fuzzy Logic (Rule)

There are many ways to express information in the field of artificial intelligence (machine intelligence). Forming human knowledge into natural language phrases of this kind is perhaps the most prevalent technique to represent it.

IF premise (antecedent), THEN conclusion (consequent)

The rule-based form in the expression above is known as the IF-THEN form, and it is also known as the deductive form. It usually expresses an inference in which we can infer or deduce another fact termed a conclusion if we know a fact (premise, hypothesis, or antecedent) (consequent). This form of knowledge representation, characterized as *shallow knowledge*, is quite appropriate in the context of linguistics because it expresses human empirical and heuristic knowledge in our own language of communication.[12] Because they use linguistic variables as their antecedents and consequent, the fuzzy rule-based system is most helpful in modelling some complex systems that humans can observe; as described here, these linguistic variables can be naturally represented by fuzzy sets and logical connectives of these sets.



The canonical form for a fuzzy rule-based system.

Rule 1:	IF condition C^1, THEN restriction R^1
Rule 2:	IF condition C^2, THEN restriction R^2
⋮	
Rule r:	IF condition C^r, THEN restriction R^r

Figure 2.5: Canonical form for a fuzzy rule-based system[12]

Any compound rule structure can be dissected and reduced to several simple canonical rules utilizing the basic properties and operations established for fuzzy sets, as shown in Figure. These rules are based on natural language models and representations, which are based on fuzzy sets and fuzzy logic in turn. A collection of restrictions on the output based on particular input conditions expresses the fuzzy level of comprehending and describing a complicated system (Figure). Fuzzy sets and relations are commonly used to model restrictions. Linguistic connectives like "and," "or" and "otherwise" are widely used to connect these limitation assertions. The constraints R1, R2..., Rr apply to the rules' output actions or consequences.

2.1.1.2 PART II - Website/Database

Designing Website using HTML, CSS, PHP, XAMPP and WinSCP.

HTML

HTML, or Hyper Text Markup Language, is the standard markup language for texts that are intended to be viewed on a web browser. Technologies such as Cascading Style Sheets (CSS) and programming languages like JavaScript can help. Web browsers receive HTML documents from a web server or locally stored files and convert them to multimedia web pages. HTML originally featured cues for the document's look and described the structure of a web page logically.

HTML elements are the components that make up HTML pages. Images and other objects, such as interactive forms, can be embedded in the produced page using HTML techniques. HTML allows you to create organised documents by indicating structural semantics for text elements like headers, paragraphs, lists, links, quotations, and other elements. Tags, which are written in angle brackets, separate HTML elements. Tags like <img /> and <input /> insert content into the page immediately. Other tags, such as <p>, surround and offer information about document text, and may comprise sub-elements such as other tags. The HTML tags are not displayed by browsers, but they are used to read the page's content.

HTML allows scripting languages like JavaScript to insert programmes that influence the behaviour and content of online pages. CSS determines the appearance and layout of material. Since 1997, the World Wide Web Consortium (W3C), which used to maintain the HTML standards and now maintains the CSS standards, has pushed the usage of CSS over explicit presentational HTML.[14]

PHP

PHP is a programming language. It's a pretty widely used language and can be embedded into HTML, which is the big draw for anyone interested in web development. HTML can be long, with lots of commands. PHP pages contain HTML with embedded code, making it much easier to manage.

PHP stands for Hypertext Pre-processor, and within itself, it is confusing. It's not as complicated as it seems though – if you have a basic understanding of HTML, PHP will be pretty easy to learn. It allows you to collect, process and utilize data to create output. It will enable you to interact with all your pages.

Three main areas where PHP used:

- Server-side scripting – the main field for PHP. If you are an amateur practising at home, this is the way to do it.
- Command-line scripting – ideal for scripts made using cron or Task Scheduler. Also great for simple text processing.
- Writing desktop applications – PHP is probably not the best language to use to create desktop applications. Still, for the advanced web developer, it provides you with many more options than its competitors.

XAMPP

There are many ways that XAMPP has been defined over the years. Some refer to it as an open-source software product – just like any other piece of software, XAMPP is a product you can install and run on your computer.

Others call it a local server or a web server solution – that is, in fact, the task that it has been designed to perform: acting as an actual web server but on a local machine.

Hence, all these definitions hold.

Now, let us break down the name itself. There are five components to unpack here:

X – this represents a cross, indicating cross-platform (compatible with multiple operating systems, such as Windows and Linux)

A – Apache HTTP Server, more commonly known as Apache.

M – MySQL (perhaps the most frequently used database among beginners; its latest version is called MariaDB, which is why people have now switched the M in XAMPP from MySQL to MariaDB)

P – PHP, a popular scripting language loved by both amateur and expert programmers.

P – Perl, a high-level programming language

Together, all these components form an integrated environment where developers can perform tasks such as

- Managing the server and associated port(s)
- Maintaining a database to store, fetch, or edit the user-related data for the website.
- Writing scripts for the website to allow data exchange between the site and the database.

WinSCP

WinSCP is an open-source FTP client for Microsoft Windows. Development began in 2000, and it is maintained by Martin Prikryl. In addition to its GUI interface, WinSCP offers batch file scripting and a command-line interface.

WinSCP offers everything you need to connect and transfer files to FTP server:

- Command-line options.
- Support for synchronizing files between folders on different systems.
- Transfer resume options for interrupted file transfers.
- Advanced filtering options to only see the files you need.

The Site Manager window appears when you first open WinSCP.

Connection information need to be added to the Site Manager to connect to FTP server:

- The protocol want to use to connect. ExaVault supports both FTP and SFTP connections. ExaVault does not support SCP or WebDav connections.
- Your ExaVault FTP server (host) name.
- Port number will change depending on the protocol you choose port 21 for FTP and port 22 for SFTP.
- Add your Username.
- Add your Password.

When all done, click the **Save** button to save the connection information for future connections or click **Login** to get connected immediately to server.

2.2 Design Flow and Process

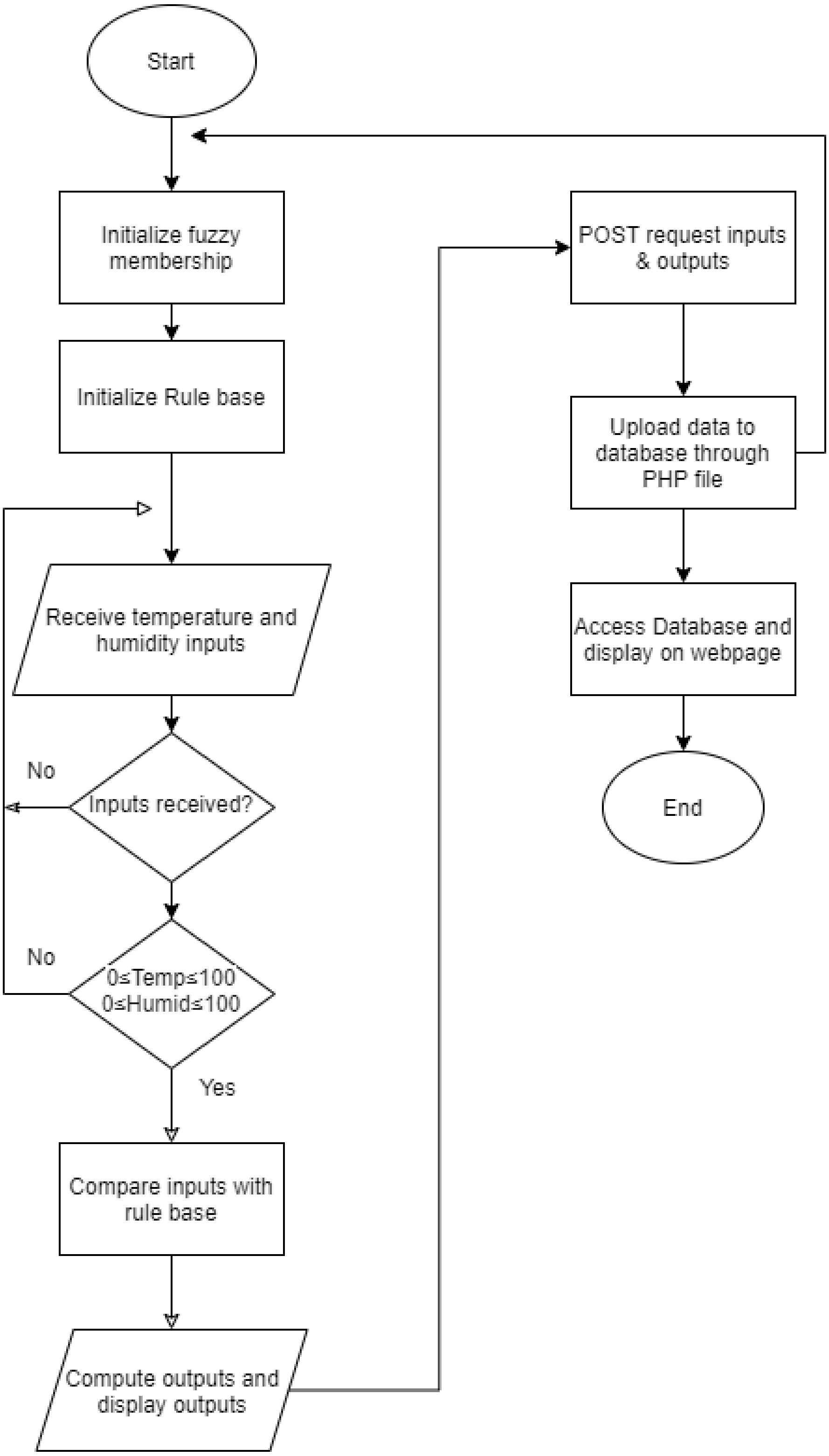

Start
Initialize fuzzy membership
Initialize Rule base
Receive temperature and humidity inputs
No
Inputs received?
No
0≤Temp≤100
0≤Humid≤100
Yes
Compare inputs with rule base
Compute outputs and display outputs
POST request inputs & outputs
Upload data to database through PHP file
Access Database and display on webpage
End

Figure 2.6: Design flow and process of fuzzy system

2.2.1 PART I -Temperature and Humidity Control System

2.2.1.1 Input Variables

2.2.1.1.1 Membership Functions and Ranges for Controlling the Temperature

The desired temperature is between 22 to 28℃ for ensuring the cultivation of healthy oyster mushrooms. The productivity of oyster mushroom cultivation in low-lying areas are still not optimal. This is due to the cultivation of oyster mushrooms needs ideal temperature and humidity (temperature 22-28 ° C with a humidity of 60% - 80%) [11]

Table 2.0: Fuzzy Membership Function for Current Temperature

MEMBERSHIP FUNCTIONS	TEMPERATURE RANGE (℃)
Very Cold	0 to 14
Cold	15 to 21
Normal	22 to 28
Hot	29 to 35
Very Hot	36 to 100

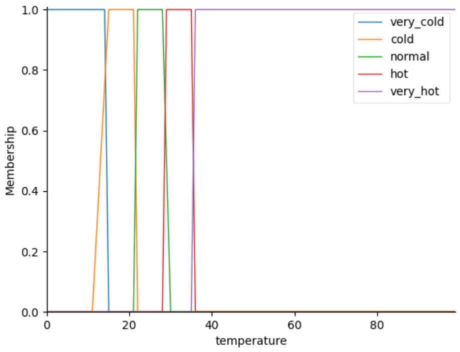
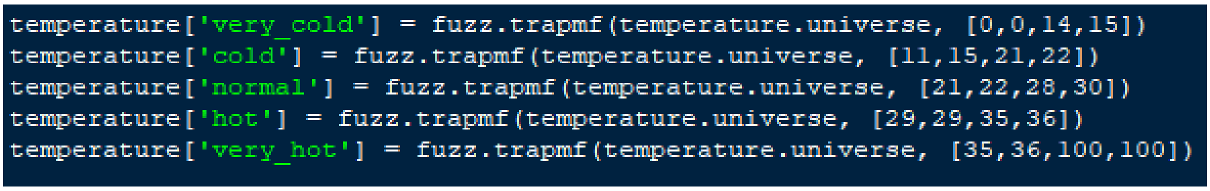


Figure 2.7: Plot of Fuzzy Membership for Temperature



```
temperature['very_cold'] = fuzz.trapmf(temperature.universe, [0,0,14,15])
temperature['cold'] = fuzz.trapmf(temperature.universe, [11,15,21,22])
temperature['normal'] = fuzz.trapmf(temperature.universe, [21,22,28,30])
temperature['hot'] = fuzz.trapmf(temperature.universe, [29,29,35,36])
temperature['very_hot'] = fuzz.trapmf(temperature.universe, [35,36,100,100])
```

Figure 2.8: Fuzzy Membership for Current Temperature Coded in Python

2.2.1.1.2 Membership Functions and Ranges for Controlling the Temperature

The desired humidity is between 60 to 80% for ensuring the cultivation of healthy oyster mushrooms,. Based on the article on Sietalab it stated that high and low humidity indicates > 80% and < 60% [12].

Table 2.1: Fuzzy Membership Function for Current Humidity

MEMBERSHIP FUNCTIONS	HUMIDITY RANGE (%)
Very Dry	0 to 39
Dry	40 to 59
Moist	60 to 80
Wet	81 to 100

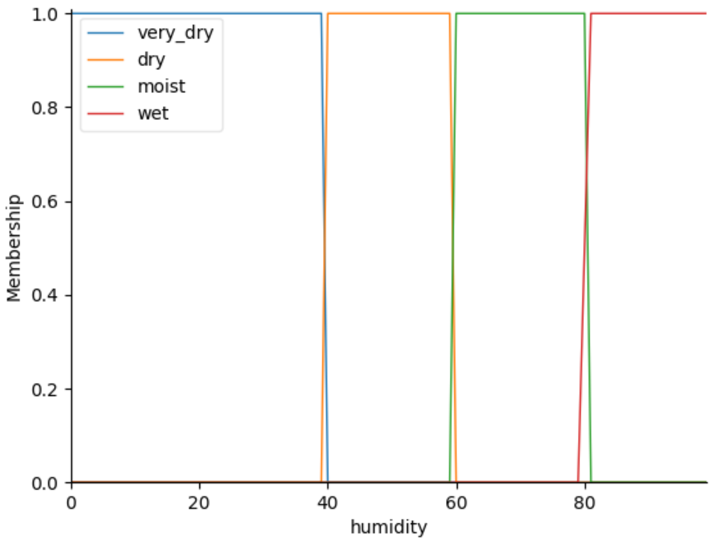
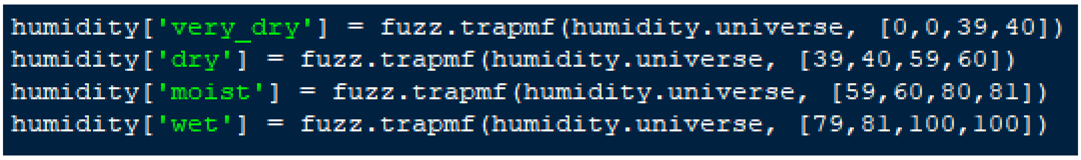


Figure 2.9: Plot of Fuzzy Membership for Humidity



```
humidity['very_dry'] = fuzz.trapmf(humidity.universe, [0,0,39,40])
humidity['dry'] = fuzz.trapmf(humidity.universe, [39,40,59,60])
humidity['moist'] = fuzz.trapmf(humidity.universe, [59,60,80,81])
humidity['wet'] = fuzz.trapmf(humidity.universe, [79,81,100,100])
```

Figure 2.10: Fuzzy Membership for Current Humidity Coded in Python

2.2.1.2 Output Variables

2.2.1.2.1 Membership Functions and Ranges for Controlling the Heat Fan Speed

The heat fan speed is controlled by the amount of power allowed by the system. When the delivered power is lower than the minimum requirement of the fan, the fan will stop operating.

Table 2.2: Fuzzy Membership Function for Heat Fan Speed

MEMBERSHIP FUNCTIONS	DUTY CYCLE (%)
Off	0 to 9
Slow	10 to 59
Fast	60 to 100

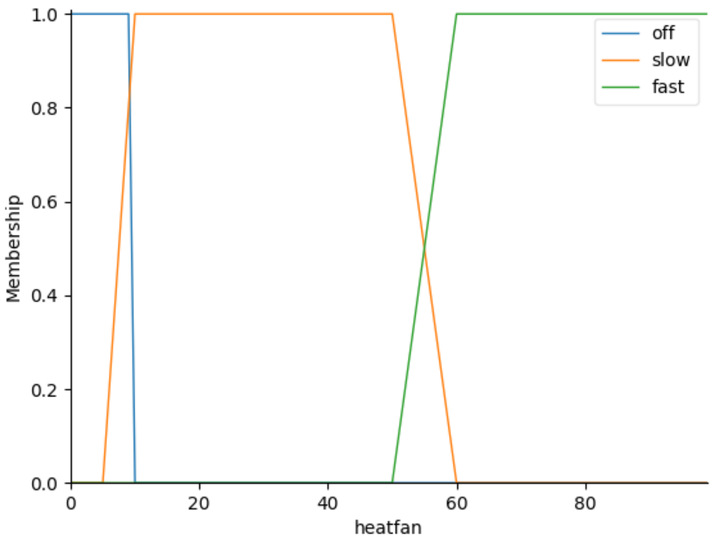
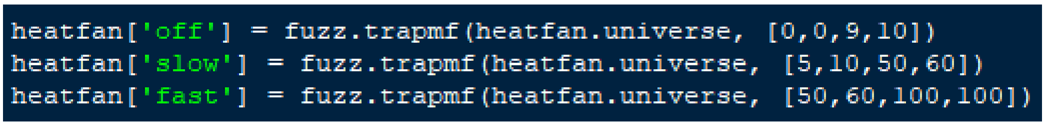


Figure 2.11: Plot of Fuzzy Membership for Heat Fan



```
heatfan['off'] = fuzz.trapmf(heatfan.universe, [0,0,9,10])
heatfan['slow'] = fuzz.trapmf(heatfan.universe, [5,10,50,60])
heatfan['fast'] = fuzz.trapmf(heatfan.universe, [50,60,100,100])
```

Figure 2.12: Fuzzy Membership for Heat Fan Speed Coded in Python

2.2.1.2.2 Membership Functions and Ranges for Controlling the Cold Fan Speed

The heat fan speed is controlled by the amount of power allowed by the system. When the delivered power is lower than the minimum requirement of the fan, the fan will stop operating.

Table 2.3: Fuzzy Membership Function for Cold Fan Speed

MEMBERSHIP FUNCTIONS	DUTY CYCLE (%)
Off	0 to 9
Slow	10 to 59
Fast	60 to 100

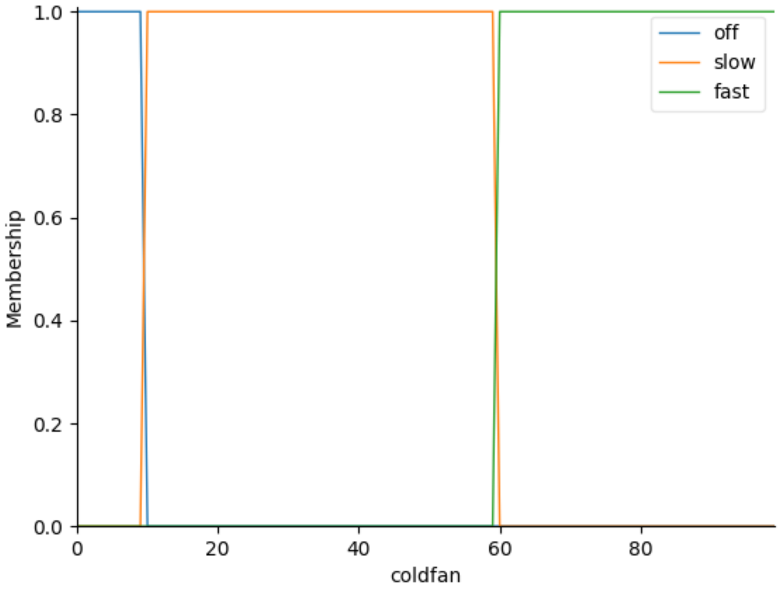
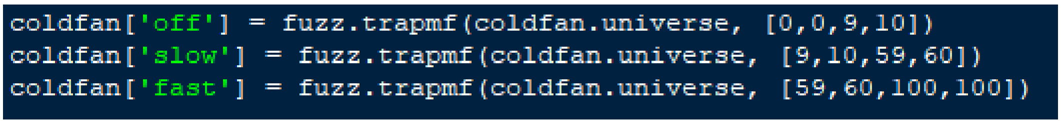


Figure 2.13: Plot of Fuzzy Membership for Cold Fan



```
coldfan['off'] = fuzz.trapmf(coldfan.universe, [0,0,9,10])
coldfan['slow'] = fuzz.trapmf(coldfan.universe, [9,10,59,60])
coldfan['fast'] = fuzz.trapmf(coldfan.universe, [59,60,100,100])
```

Figure 2.14: Fuzzy Membership for Cold Fan Speed Coded in Python

2.2.1.2.3 Membership Functions and Ranges for Controlling the Humidifier

A humidifier allows the soil to absorb the mist created by the humidifier. These moves will enable the mushroom to have enough humidity.

Table 2.4: Membership Function for Humidifier Control

MEMBERSHIP FUNCTIONS	DUTY CYCLE (%)
Off	0 to 9
Low	10 to 59
High	60 to 100

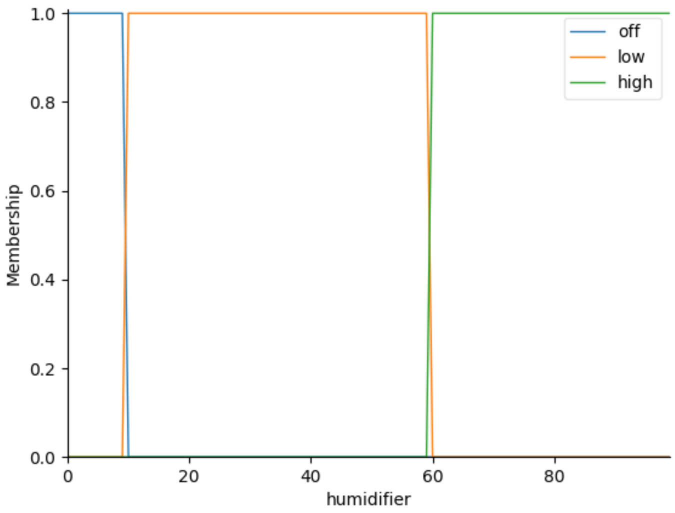
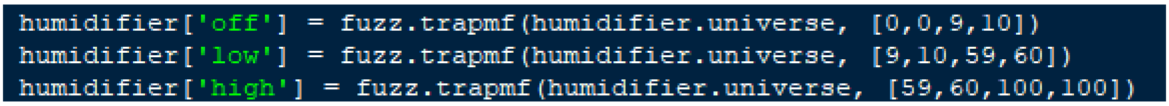


Figure 2.15: Plot of Fuzzy Membership for Humidifier



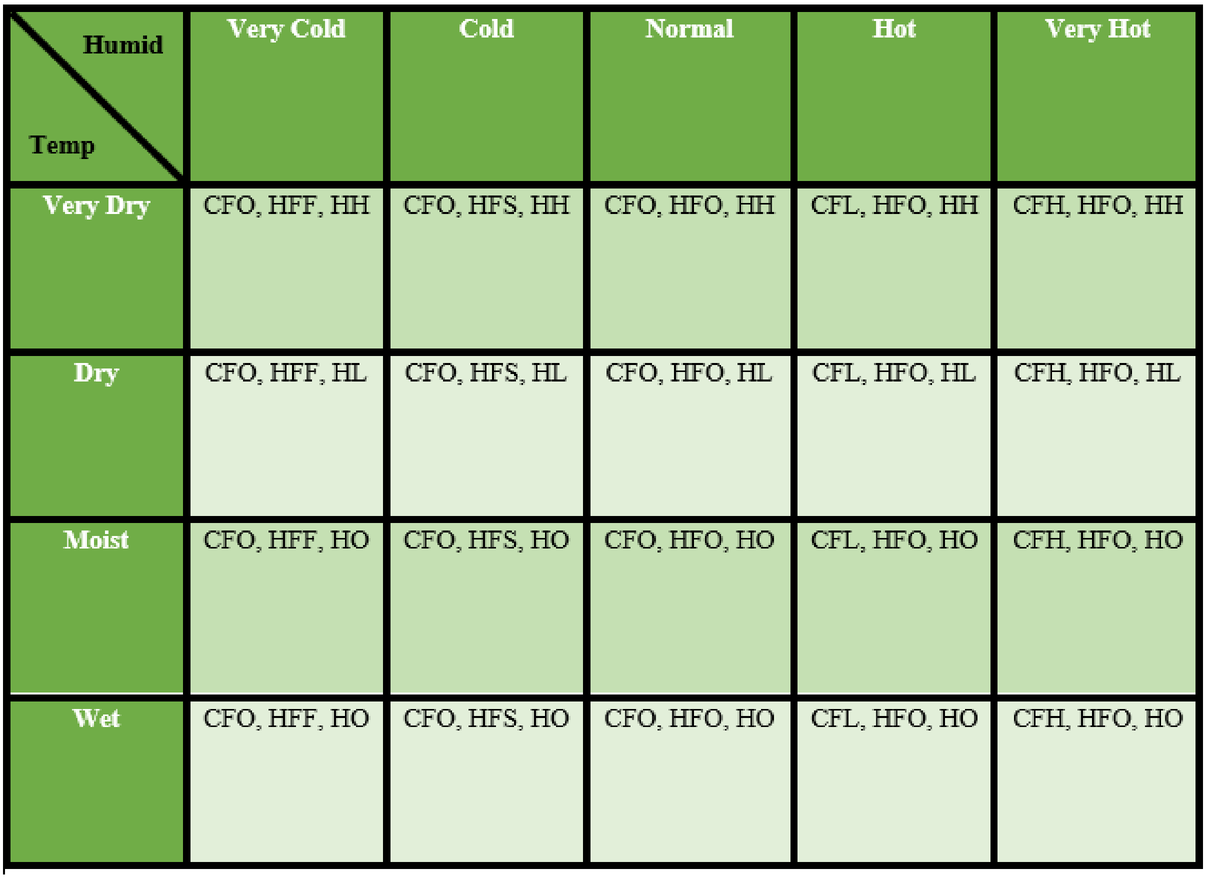
```
humidifier['off'] = fuzz.trapmf(humidifier.universe, [0,0,9,10])
humidifier['low'] = fuzz.trapmf(humidifier.universe, [9,10,59,60])
humidifier['high'] = fuzz.trapmf(humidifier.universe, [59,60,100,100])
```

Figure 2.16 Fuzzy Membership for Humidifier Coded in Python

2.2.1.3 Rules base

Rule base are formed by the fuzzy membership.

Table 2.5: Simplified Rule Base for Fuzzy Logic Controller



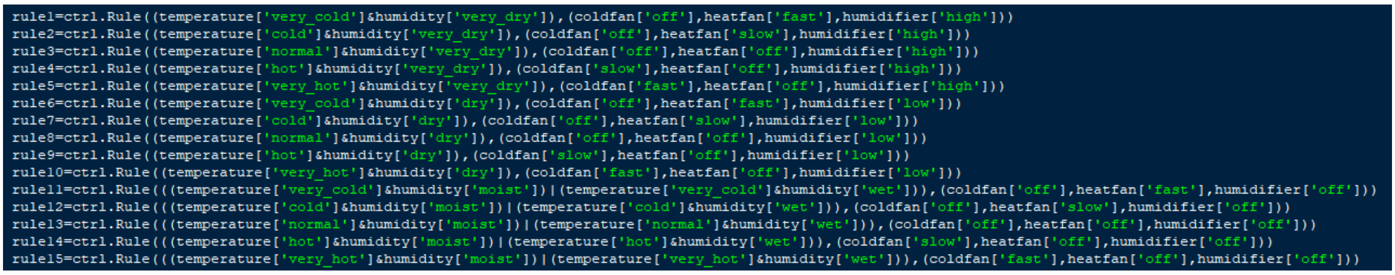
Humid Temp	Very Cold	Cold	Normal	Hot	Very Hot
Very Dry	CFO, HFF, HH	CFO, HFS, HH	CFO, HFO, HH	CFL, HFO, HH	CFH, HFO, HH
Dry	CFO, HFF, HL	CFO, HFS, HL	CFO, HFO, HL	CFL, HFO, HL	CFH, HFO, HL
Moist	CFO, HFF, HO	CFO, HFS, HO	CFO, HFO, HO	CFL, HFO, HO	CFH, HFO, HO
Wet	CFO, HFF, HO	CFO, HFS, HO	CFO, HFO, HO	CFL, HFO, HO	CFH, HFO, HO

Table 2.6: Indicator for Table

CFO = Cold Fan OFF	**HFO = Hot Fan OFF**	**HO = Humidifier OFF**
CFL = Cold Fan LOW	**HFO = Hot Fan LOW**	**HL = Humidifier LOW**
CFH = Cold Fan HIGH	**HFO = Hot Fan HIGH**	**HH = Humidifier HIGH**

Table 2.7: Rule Base for Fuzzy Logic Controller

RULE No.	TEMP	HUMIDITY	HEAT FAN	COLD FAN	HUMIDIFIER
1	Very Cold	Very Dry	Fast	Off	High
2	Cold	Very Dry	Slow	Off	High
3	Normal	Very Dry	Off	Off	High
4	Hot	Very Dry	Off	Slow	High
5	Very Hot	Very Dry	Off	Fast	High
6	Very Cold	Dry	Fast	Off	Low
7	Cold	Dry	Slow	Off	Low
8	Normal	Dry	Off	Off	Low
9	Hot	Dry	Off	Slow	Low
10	Very Hot	Dry	Off	Fast	Low
11	Very Cold Very Cold	Moist Wet	Fast	Off	Off
12	Cold Cold	Moist Wet	Slow	Off	Off
13	Normal Normal	Moist Wet	Off	Off	Off
14	Hot Hot	Moist Wet	Off	Slow	Off
15	Very Hot Very Hot	Moist Wet	Off	Fast	Off

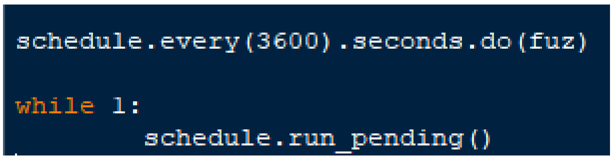


```
rule1=ctrl.Rule((temperature['very_cold']&humidity['very_dry']),(coldfan['off'],heatfan['fast'],humidifier['high']))
rule2=ctrl.Rule((temperature['cold']&humidity['very_dry']),(coldfan['off'],heatfan['slow'],humidifier['high']))
rule3=ctrl.Rule((temperature['normal']&humidity['very_dry']),(coldfan['off'],heatfan['off'],humidifier['high']))
rule4=ctrl.Rule((temperature['hot']&humidity['very_dry']),(coldfan['slow'],heatfan['off'],humidifier['high']))
rule5=ctrl.Rule((temperature['very_hot']&humidity['very_dry']),(coldfan['fast'],heatfan['off'],humidifier['high']))
rule6=ctrl.Rule((temperature['very_cold']&humidity['dry']),(coldfan['off'],heatfan['fast'],humidifier['low']))
rule7=ctrl.Rule((temperature['cold']&humidity['dry']),(coldfan['off'],heatfan['slow'],humidifier['low']))
rule8=ctrl.Rule((temperature['normal']&humidity['dry']),(coldfan['off'],heatfan['off'],humidifier['low']))
rule9=ctrl.Rule((temperature['hot']&humidity['dry']),(coldfan['slow'],heatfan['off'],humidifier['low']))
rule10=ctrl.Rule((temperature['very_hot']&humidity['dry']),(coldfan['fast'],heatfan['off'],humidifier['low']))
rule11=ctrl.Rule(((temperature['very_cold']&humidity['moist'])|(temperature['very_cold']&humidity['wet'])),(coldfan['off'],heatfan['fast'],humidifier['off']))
rule12=ctrl.Rule(((temperature['cold']&humidity['moist'])|(temperature['cold']&humidity['wet'])),(coldfan['off'],heatfan['slow'],humidifier['off']))
rule13=ctrl.Rule(((temperature['normal']&humidity['moist'])|(temperature['normal']&humidity['wet'])),(coldfan['off'],heatfan['off'],humidifier['off']))
rule14=ctrl.Rule(((temperature['hot']&humidity['moist'])|(temperature['hot']&humidity['wet'])),(coldfan['slow'],heatfan['off'],humidifier['off']))
rule15=ctrl.Rule(((temperature['very_hot']&humidity['moist'])|(temperature['very_hot']&humidity['wet'])),(coldfan['fast'],heatfan['off'],humidifier['off']))
```

Figure 2.17: Rule Base Coded in Python

2.2.1.4 Timer

A temperature and humidity control system for the cultivation of oyster mushrooms might be required to track and control within specific periods. The working concept is like a refrigerator or an air conditioner. A timer must be set to ensure the system keeps on looping to ensure the surrounding temperature is kept at the desired temperature and humidity value. So, in Python, we have implemented the timer using the schedule and time library to ensure the temperature and humidity control system will work every 1 hour.



```
schedule.every(3600).seconds.do(fuz)

while 1:
        schedule.run_pending()
```

Figure 2.18: Timer

This line of code will allow the system to call the main function every 3600 seconds, which is equal to 1 hour. So, the system will be kept in a loop until the user decides to terminate the system.

2.2.1.5 Output Explanation

The output of the heat, cold fan and the humidity is generated by referring the duty cycle. The working method is based on Pulse Width Modulation (PWM). In the area of electrical engineering, PWM (Pulse Width Modulation) or modulation with the width of an impulse is a common phrase. It has a wide range of applications, including telecommunications, audio equipment, and servo motors, among others. The use of PWM in voltage regulation is intriguing to us as hobbyists. In a short, PWM works like a switch that cycles on and off repeatedly, adjusting the amount of power the fan or pump motor receives. The PWM system for regulating fans and pumps works with the motor, which receives either +12V (full power) or 0V (no power) (no power). Look at the figure 2.18 to get a better idea of how this works.

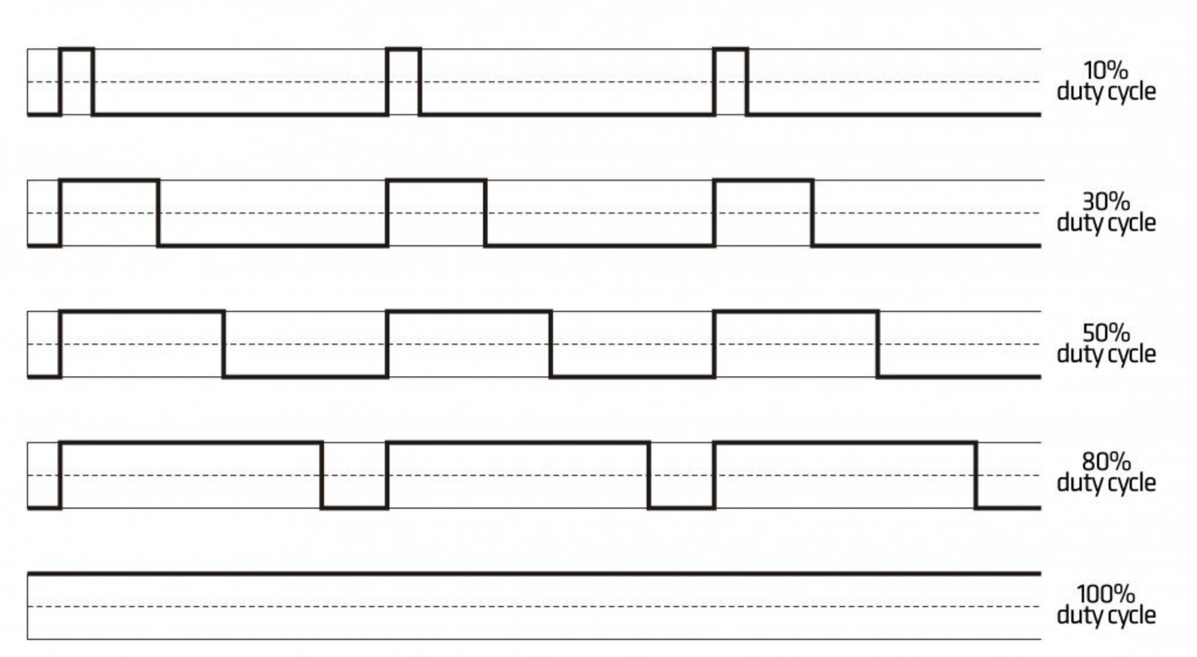
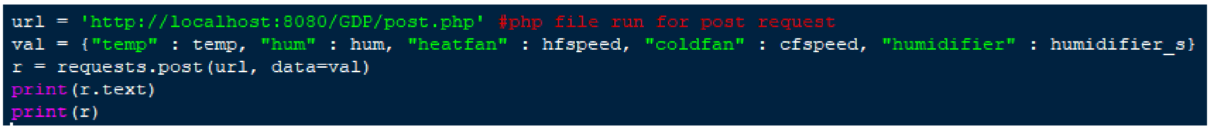


Figure 2.19: Duty Cycle Graph [13]

A 10% duty cycle means that the motor will spin slowly and only get a few power impulses over a period of time, whereas a 100% duty cycle means that the fan/pump will run at full speed all of the time. In our project the range of the duty cycle is based on our own analysis, the actual value will need to have some modifications based on the real behaviour of the hardware specification.

2.2.2 PART II - Website/ Database

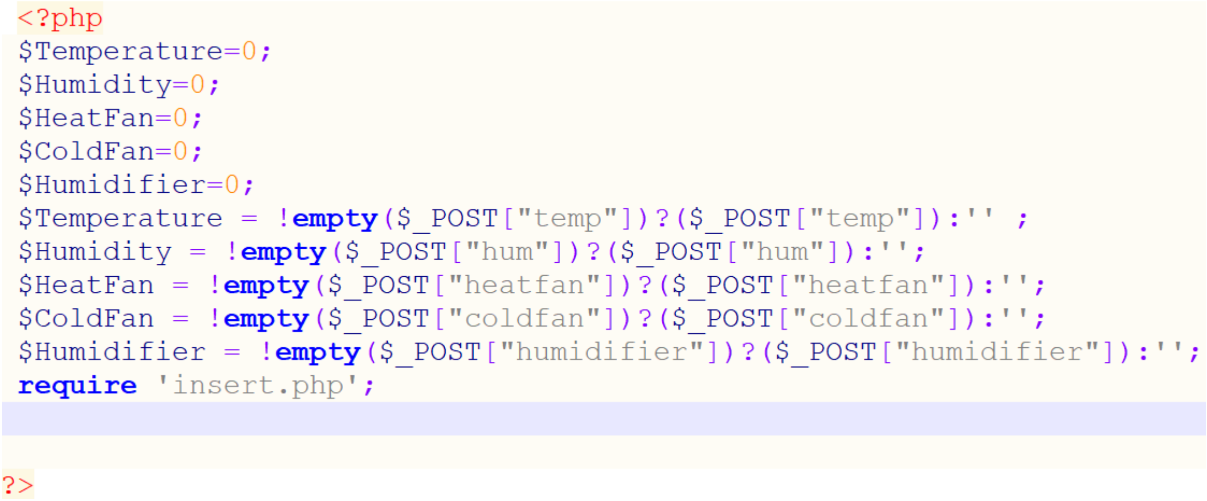
The user might need to record the change of temperature and humidity while the system is operating, so we need to transfer all the input and output to the database to allow the user to track back the data. To transfer the data to the web, we have used the POST request in Python which allows data transfer to the web with URL insertion. Below is the coding of the POST request.



```
url = 'http://localhost:8080/GDP/post.php' #php file run for post request
val = {"temp" : temp, "hum" : hum, "heatfan" : hfspeed, "coldfan" : cfspeed, "humidifier" : humidifier_s}
r = requests.post(url, data=val)
print(r.text)
print(r)
```

Figure 2.20: POST request

After the POST request from Python, we have to retrieve the data with a POST request from PHP.

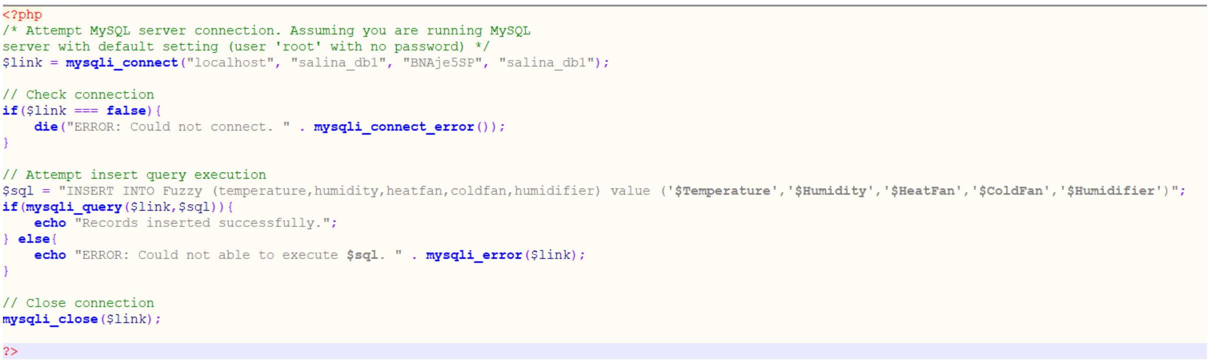


```
<?php
$Temperature=0;
$Humidity=0;
$HeatFan=0;
$ColdFan=0;
$Humidifier=0;
$Temperature = !empty($_POST["temp"])?($_POST["temp"]):''  ;
$Humidity = !empty($_POST["hum"])?($_POST["hum"]):'';
$HeatFan = !empty($_POST["heatfan"])?($_POST["heatfan"]):'';
$ColdFan = !empty($_POST["coldfan"])?($_POST["coldfan"]):'';
$Humidifier = !empty($_POST["humidifier"])?($_POST["humidifier"]):'';
require 'insert.php';


?>
```

Figure 2.21: POST request from PHP

Then we need to transfer the retrieved data and insert it into the database by doing the following code.



```
<?php
/* Attempt MySQL server connection. Assuming you are running MySQL
server with default setting (user 'root' with no password) */
$link = mysqli_connect("localhost", "salina_db1", "BNAje5SP", "salina_db1");

// Check connection
if($link === false){
    die("ERROR: Could not connect. " . mysqli_connect_error());
}

// Attempt insert query execution
$sql = "INSERT INTO Fuzzy (temperature,humidity,heatfan,coldfan,humidifier) value ('$Temperature','$Humidity','$HeatFan','$ColdFan','$Humidifier')";
if(mysqli_query($link,$sql)){
    echo "Records inserted successfully.";
} else{
    echo "ERROR: Could not able to execute $sql. " . mysqli_error($link);
}

// Close connection
mysqli_close($link);

?>
```

Figure 2.22: Database access

First, we need to create a connection with the database. The first line of code is to connect to the database by inserting the corrected password, username, and database name. To ensure the database can access, so for the second line of the code is its purpose. When it fails to connect, it will display "Could not connect". Then the following code inserts the data according to the database column. When the column could not be found, it will display "ERROR: Could not be able to execute", but when it is successful in inserting, it will display "Records inserted successfully."

Then we created a website for letting user to view the data stored in the database. To design the website we have implemented HTML, CSS, and PHP. Besides of showing data from database, we have also provided some extra features for a better experience of using the website.

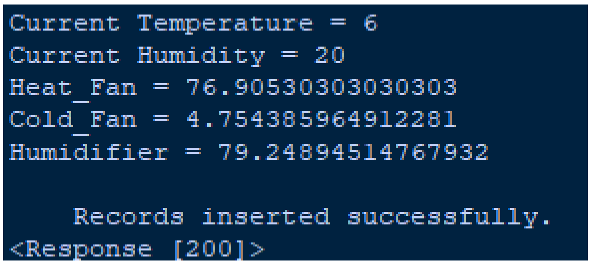
CHAPTER 3

RESULTS AND DISCUSSION

3.1 Results

3.1.1 PART I -Temperature and Humidity Control System

3.1.1.1 Rule 1



```
Current Temperature = 6
Current Humidity = 20
Heat_Fan = 76.90530303030303
Cold_Fan = 4.754385964912281
Humidifier = 79.24894514767932

    Records inserted successfully.
<Response [200]>
```

Figure 3.0: Rule 1

Table 3.0: Rule 1

RULE No.	TEMP	HUMIDITY	HEAT FAN	COLD FAN	HUMIDIFIER
1	Very Cold	Very Dry	Fast	Off	High

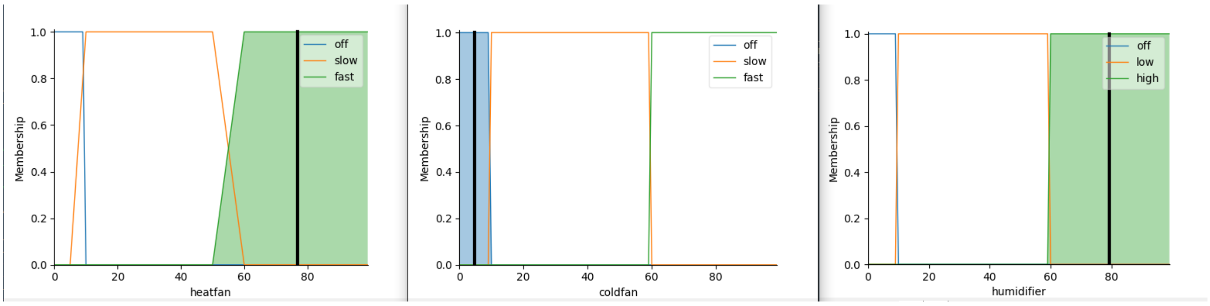
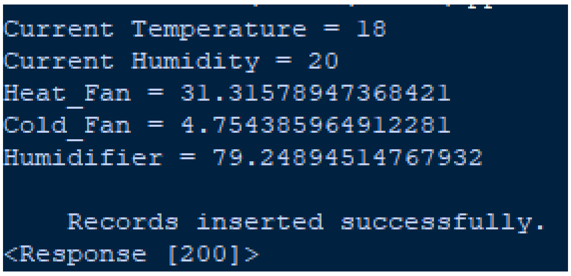


Figure 3.1: Graph for Rule 1 output

3.1.1.2 Rule 2



```
Current Temperature = 18
Current Humidity = 20
Heat_Fan = 31.31578947368421
Cold_Fan = 4.754385964912281
Humidifier = 79.24894514767932

    Records inserted successfully.
<Response [200]>
```

Figure 3.2: Rule 2

Table 3.1: Rule 2

RULE No.	TEMP	HUMIDITY	HEAT FAN	COLD FAN	HUMIDIFIER
2	Cold	Very Dry	Slow	Off	High

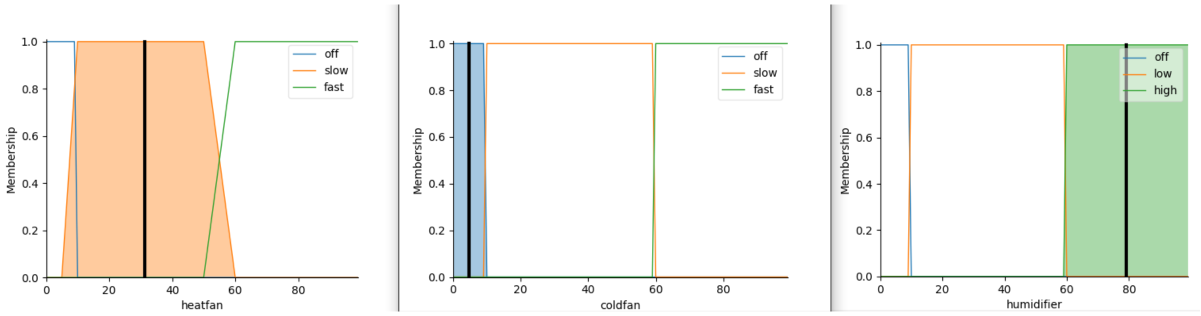
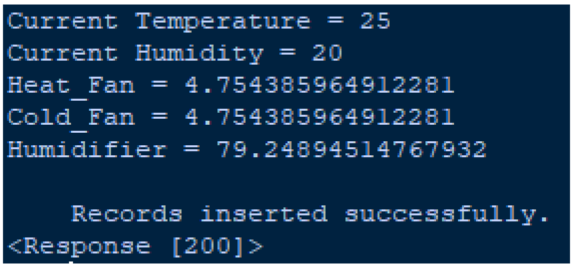


Figure 3.3: Graph for Rule 2 output

3.1.1.3 Rule 3



```
Current Temperature = 25
Current Humidity = 20
Heat_Fan = 4.754385964912281
Cold_Fan = 4.754385964912281
Humidifier = 79.24894514767932

    Records inserted successfully.
<Response [200]>
```

Figure 3.4: Rule 3

Table 3.2: Rule 3

RULE No.	TEMP	HUMIDITY	HEAT FAN	COLD FAN	HUMIDIFIER
3	Normal	Very Dry	Off	Off	High

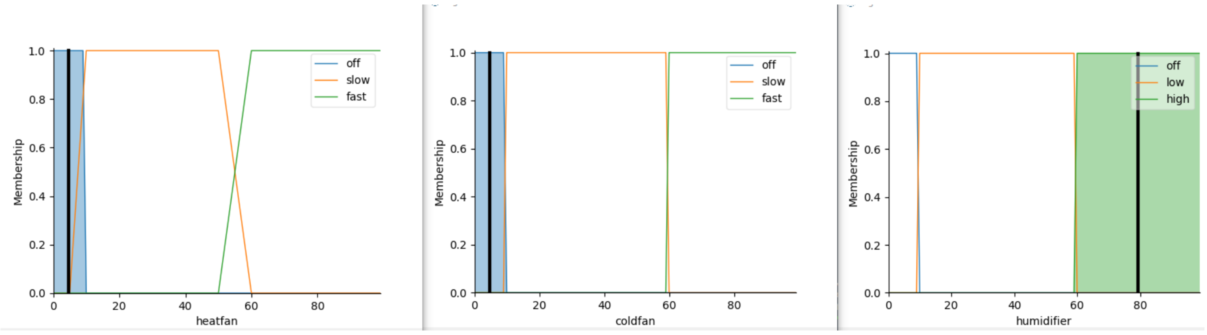
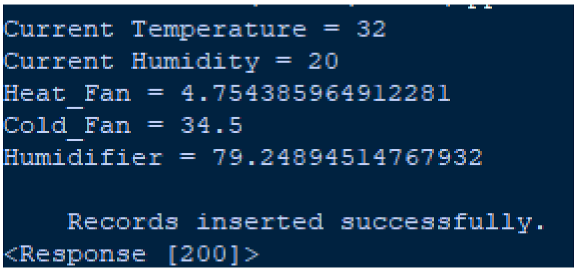


Figure 3.5: Graph for Rule 3 output

3.1.1.4 Rule 4



```
Current Temperature = 32
Current Humidity = 20
Heat_Fan = 4.754385964912281
Cold_Fan = 34.5
Humidifier = 79.24894514767932

    Records inserted successfully.
<Response [200]>
```

Figure 3.6: Rule 4

Table 3.3: Rule 4

RULE No.	TEMP	HUMIDITY	HEAT FAN	COLD FAN	HUMIDIFIER
4	Hot	Very Dry	Off	Slow	High

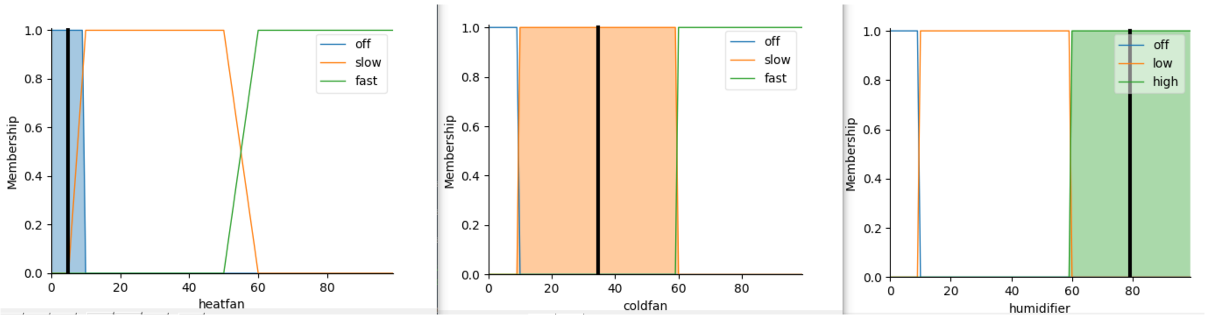
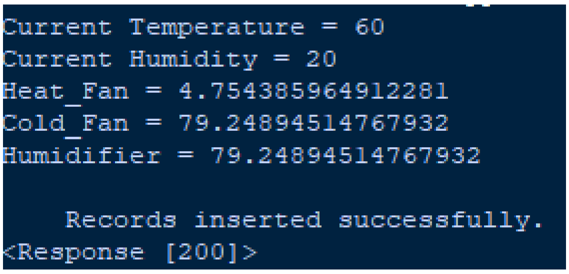


Figure 3.7: Graph for Rule 4 output

3.1.1.5 Rule 5



```
Current Temperature = 60
Current Humidity = 20
Heat_Fan = 4.754385964912281
Cold_Fan = 79.24894514767932
Humidifier = 79.24894514767932

    Records inserted successfully.
<Response [200]>
```

Figure 3.8: Rule 5

Table 3.4: Rule 5

RULE No.	TEMP	HUMIDITY	HEAT FAN	COLD FAN	HUMIDIFIER
5	Very Hot	Very Dry	Off	Fast	High

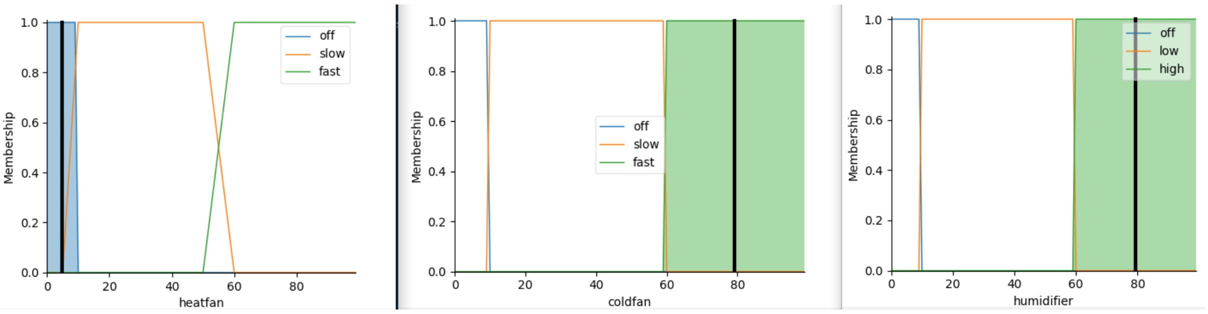
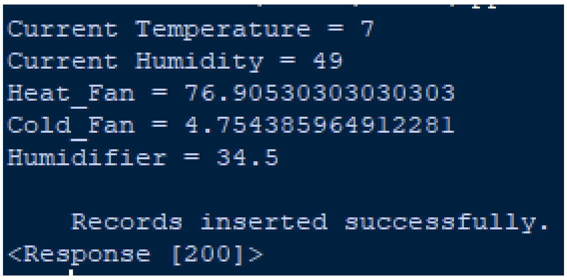


Figure 3.9: Graph for Rule 5 output

3.1.1.6 Rule 6



```
Current Temperature = 7
Current Humidity = 49
Heat_Fan = 76.90530303030303
Cold_Fan = 4.754385964912281
Humidifier = 34.5

    Records inserted successfully.
<Response [200]>
```

Figure 3.10: Rule 6

Table 3.5: Rule 6

RULE No.	TEMP	HUMIDITY	HEAT FAN	COLD FAN	HUMIDIFIER
6	Very Cold	Dry	Fast	Off	Low

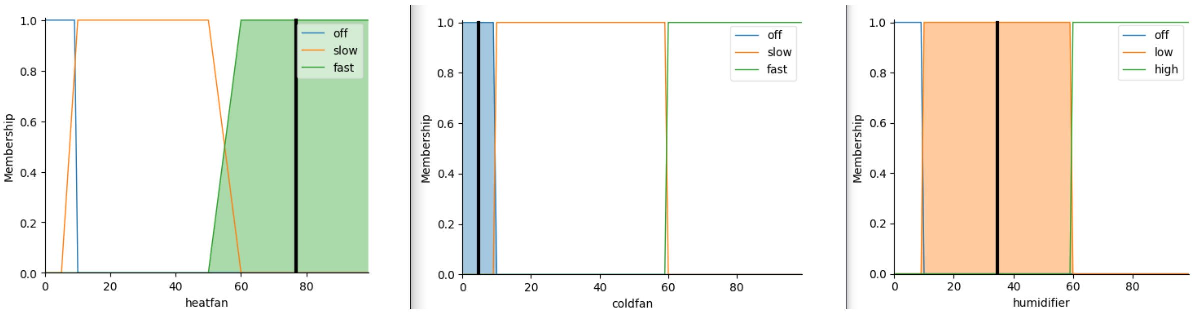
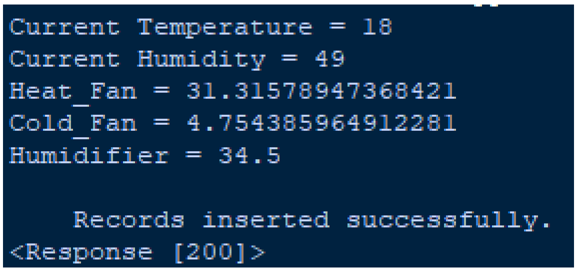


Figure 3.11: Graph for Rule 6 output

3.1.1.7 Rule 7



```
Current Temperature = 18
Current Humidity = 49
Heat_Fan = 31.31578947368421
Cold_Fan = 4.754385964912281
Humidifier = 34.5

    Records inserted successfully.
<Response [200]>
```

Figure 3.12: Rule 7

Table 3.6: Rule 7

RULE No.	TEMP	HUMIDITY	HEAT FAN	COLD FAN	HUMIDIFIER
7	Cold	Dry	Slow	Off	Low

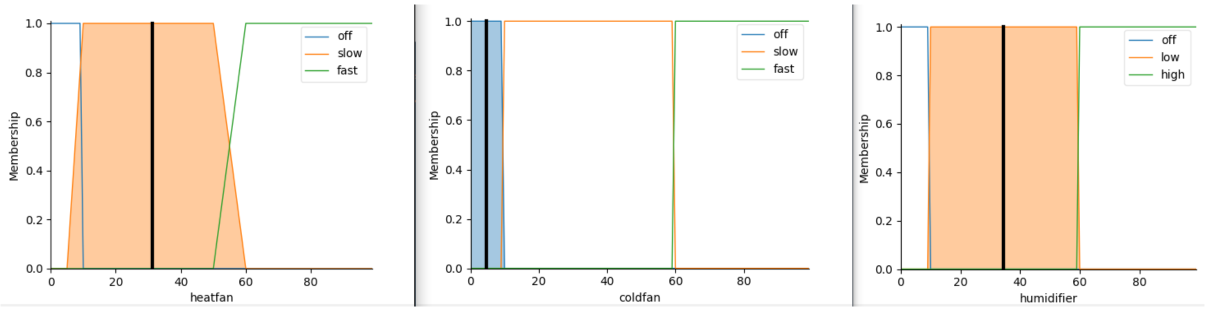
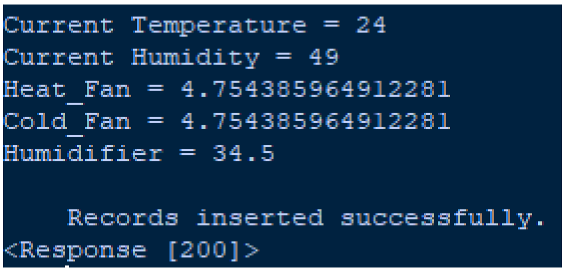


Figure 3.13: Graph for Rule 7 output

3.1.1.8 Rule 8



```
Current Temperature = 24
Current Humidity = 49
Heat_Fan = 4.754385964912281
Cold_Fan = 4.754385964912281
Humidifier = 34.5

    Records inserted successfully.
<Response [200]>
```

Figure 3.14: Rule 8

Table 3.7: Rule 8

RULE No.	TEMP	HUMIDITY	HEAT FAN	COLD FAN	HUMIDIFIER
8	Normal	Dry	Off	Off	Low

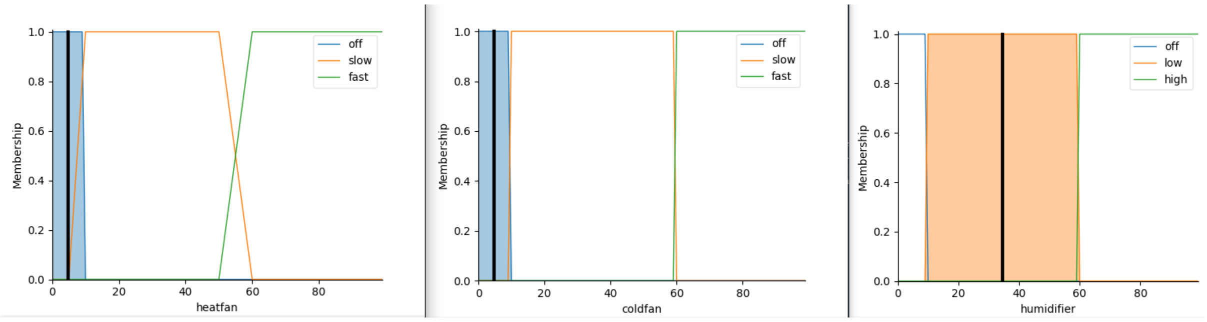
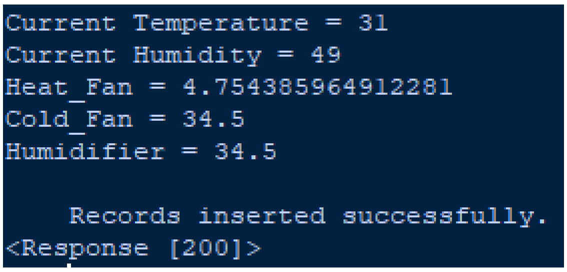


Figure 3.15: Graph for Rule 8 output

3.1.1.9 Rule 9



```
Current Temperature = 31
Current Humidity = 49
Heat_Fan = 4.754385964912281
Cold_Fan = 34.5
Humidifier = 34.5

    Records inserted successfully.
<Response [200]>
```

Figure 3.16: Rule 9

Table 3.8: Rule 9

RULE No.	TEMP	HUMIDITY	HEAT FAN	COLD FAN	HUMIDIFIER
9	Hot	Dry	Off	Slow	Low

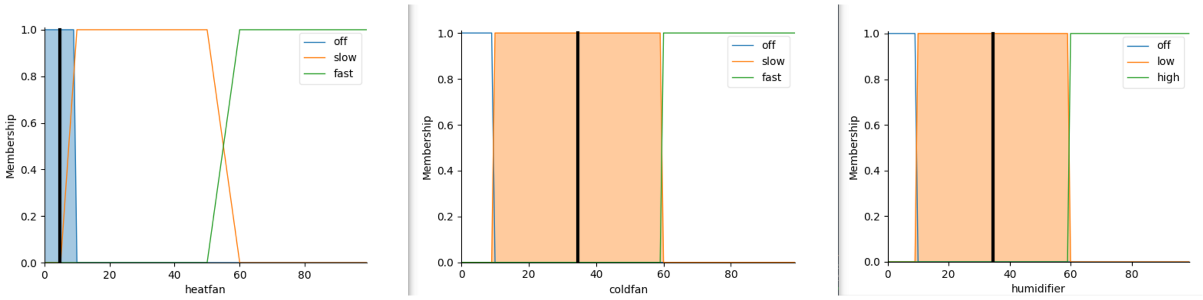
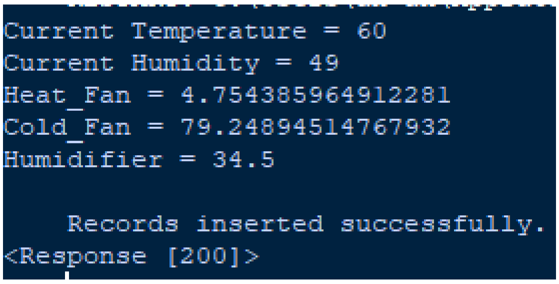


Figure 3.17: Graph for Rule 9 output

3.1.1.10 Rule 10



```
Current Temperature = 60
Current Humidity = 49
Heat_Fan = 4.754385964912281
Cold_Fan = 79.24894514767932
Humidifier = 34.5

    Records inserted successfully.
<Response [200]>
```

Figure 3.18: Rule 10

Table 3.9: Rule 10

RULE No.	TEMP	HUMIDITY	HEAT FAN	COLD FAN	HUMIDIFIER
10	Very Hot	Dry	Off	Fast	Low

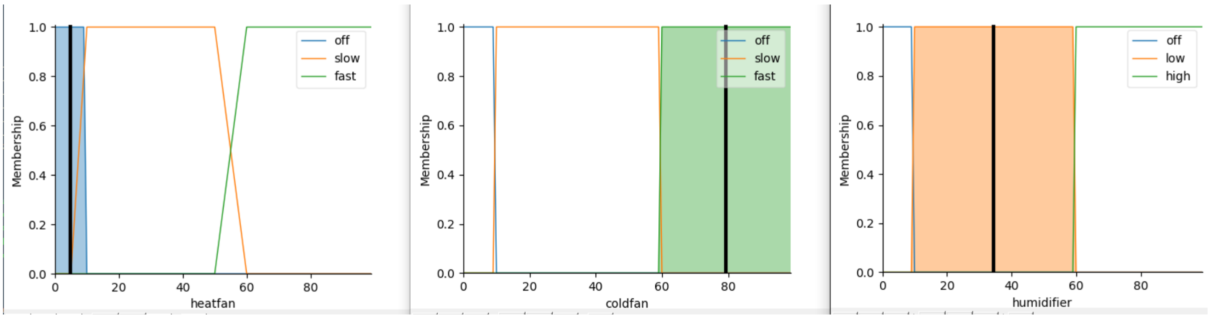
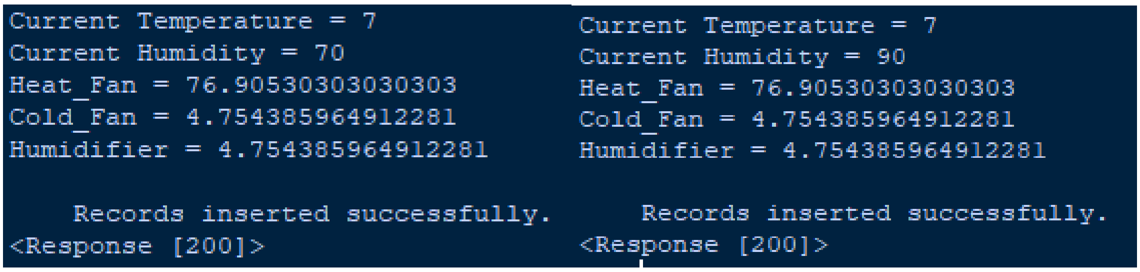


Figure 3.19: Graph for Rule 10 output

3.1.1.11 Rule 11



```
Current Temperature = 7
Current Humidity = 70
Heat_Fan = 76.90530303030303
Cold_Fan = 4.754385964912281
Humidifier = 4.754385964912281

    Records inserted successfully.
<Response [200]>
```

```
Current Temperature = 7
Current Humidity = 90
Heat_Fan = 76.90530303030303
Cold_Fan = 4.754385964912281
Humidifier = 4.754385964912281

    Records inserted successfully.
<Response [200]>
```

Figure 3.20: Rule 11

Table 3.10: Rule 11

RULE No.	TEMP	HUMIDITY	HEAT FAN	COLD FAN	HUMIDIFIER
11	Very Cold Very Cold	Moist Wet	Fast	Off	Off

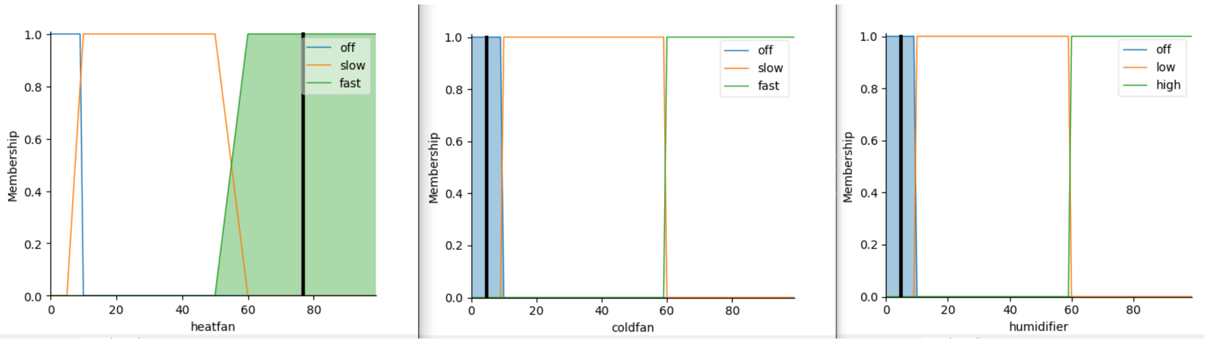
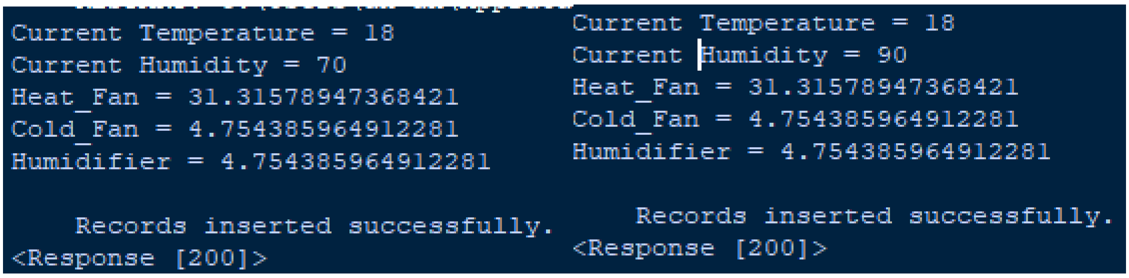


Figure 3.21: Graph for Rule 11 output

3.1.1.12 Rule 12



```
Current Temperature = 18
Current Humidity = 70
Heat_Fan = 31.31578947368421
Cold_Fan = 4.754385964912281
Humidifier = 4.754385964912281

    Records inserted successfully.
<Response [200]>
```

```
Current Temperature = 18
Current Humidity = 90
Heat_Fan = 31.31578947368421
Cold_Fan = 4.754385964912281
Humidifier = 4.754385964912281

    Records inserted successfully.
<Response [200]>
```

Figure 3.22: Rule 12

Table 3.11: Rule 12

RULE No.	TEMP	HUMIDITY	HEAT FAN	COLD FAN	HUMIDIFIER
12	Cold Cold	Moist Wet	Slow	Off	Off

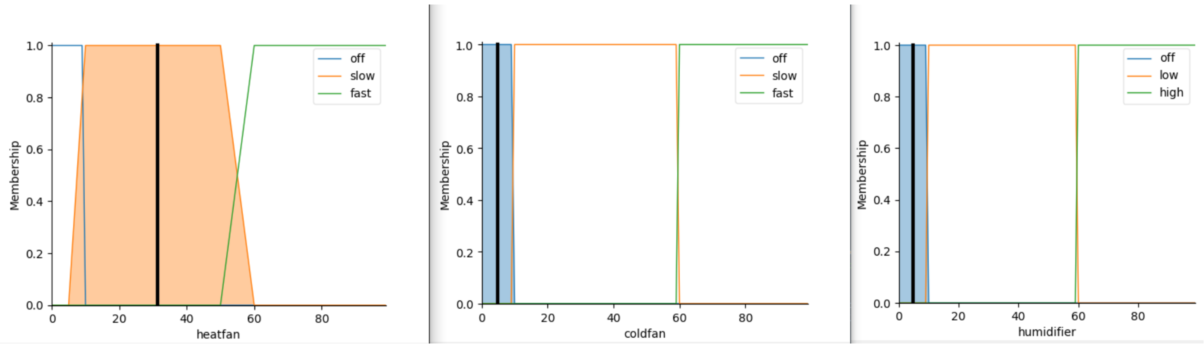
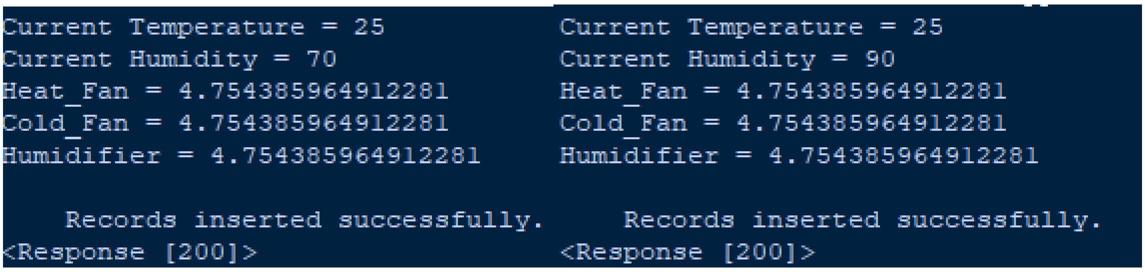


Figure 3.23: Graph for Rule 12 output

3.1.1.12 Rule 13



```
Current Temperature = 25
Current Humidity = 70
Heat_Fan = 4.754385964912281
Cold_Fan = 4.754385964912281
Humidifier = 4.754385964912281

    Records inserted successfully.
<Response [200]>
```

```
Current Temperature = 25
Current Humidity = 90
Heat_Fan = 4.754385964912281
Cold_Fan = 4.754385964912281
Humidifier = 4.754385964912281

    Records inserted successfully.
<Response [200]>
```

Figure 3.24: Rule 13

Table 3.12: Rule 12

RULE No.	TEMP	HUMIDITY	HEAT FAN	COLD FAN	HUMIDIFIER
13	Normal Normal	Moist Wet	Off	Off	Off

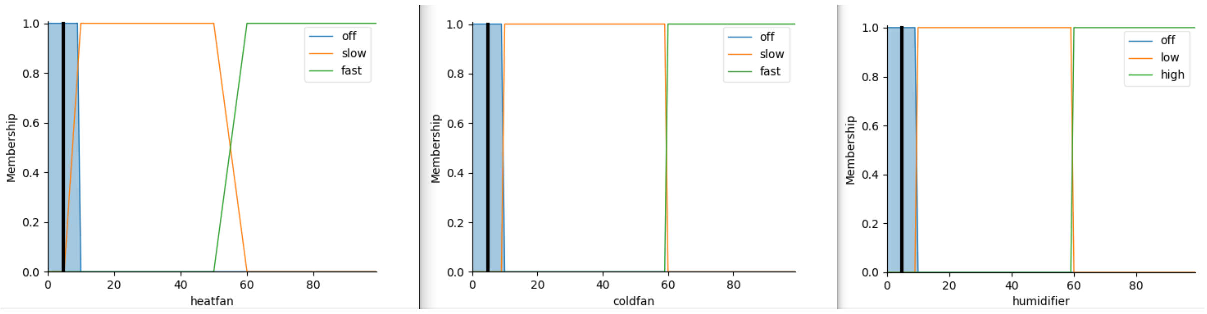
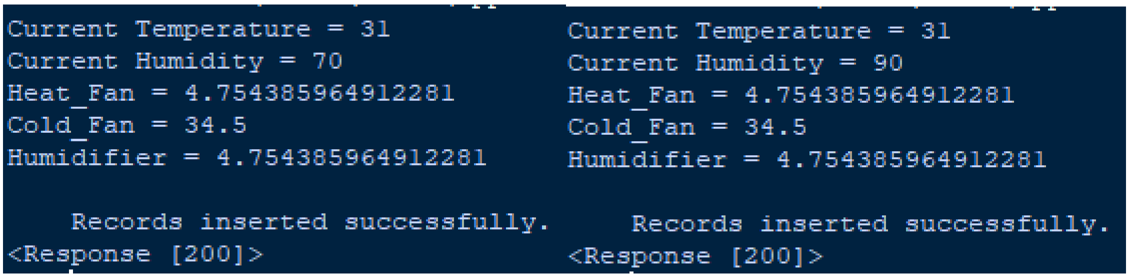


Figure 3.25: Graph for Rule 13 output

3.1.1.14 Rule 14



```
Current Temperature = 31
Current Humidity = 70
Heat_Fan = 4.754385964912281
Cold_Fan = 34.5
Humidifier = 4.754385964912281

    Records inserted successfully.
<Response [200]>
```

```
Current Temperature = 31
Current Humidity = 90
Heat_Fan = 4.754385964912281
Cold_Fan = 34.5
Humidifier = 4.754385964912281

    Records inserted successfully.
<Response [200]>
```

Figure 3.26: Rule 14

Table 3.13: Rule 14

RULE No.	TEMP	HUMIDITY	HEAT FAN	COLD FAN	HUMIDIFIER
14	Hot Hot	Moist Wet	Off	Slow	Off

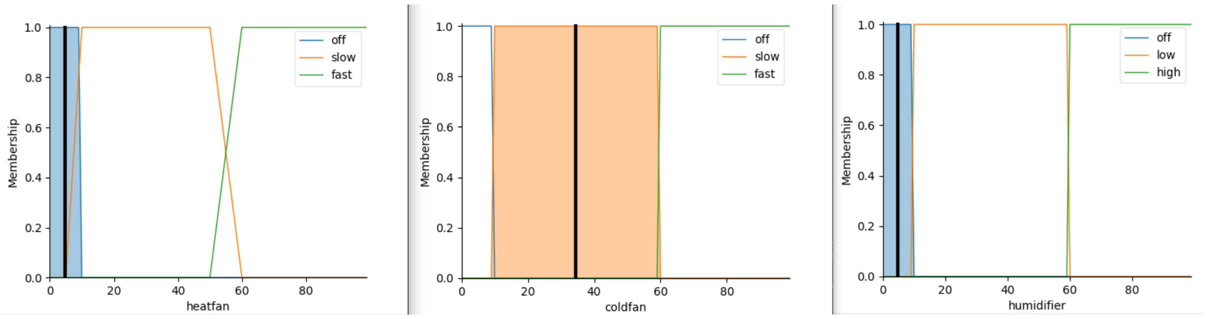
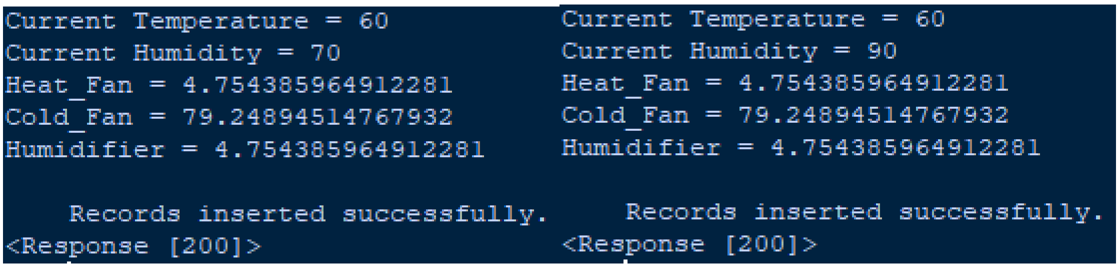


Figure 3.27: Graph for Rule 14 output

3.1.1.15 Rule 15



```
Current Temperature = 60
Current Humidity = 70
Heat_Fan = 4.754385964912281
Cold_Fan = 79.24894514767932
Humidifier = 4.754385964912281

    Records inserted successfully.
<Response [200]>
```

```
Current Temperature = 60
Current Humidity = 90
Heat_Fan = 4.754385964912281
Cold_Fan = 79.24894514767932
Humidifier = 4.754385964912281

    Records inserted successfully.
<Response [200]>
```

Figure 3.28: Rule 15

Table 3.14: Rule 15

RULE No.	TEMP	HUMIDITY	HEAT FAN	COLD FAN	HUMIDIFIER
15	Very Hot Very Hot	Moist Wet	Off	Fast	Off

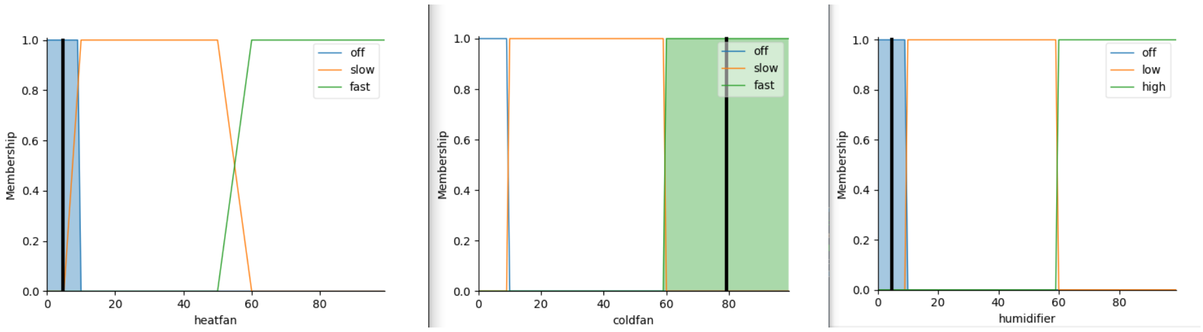


Figure 3.29: Graph for Rule 15 output

3.1.2 PART II - Website/ Database

3.1.2.1 Home Page

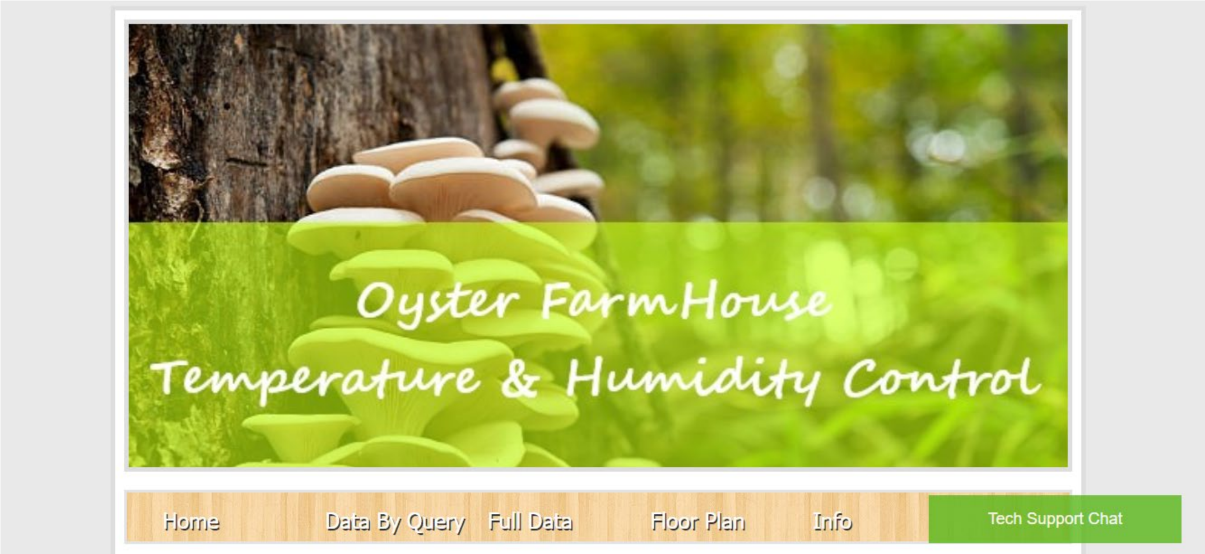
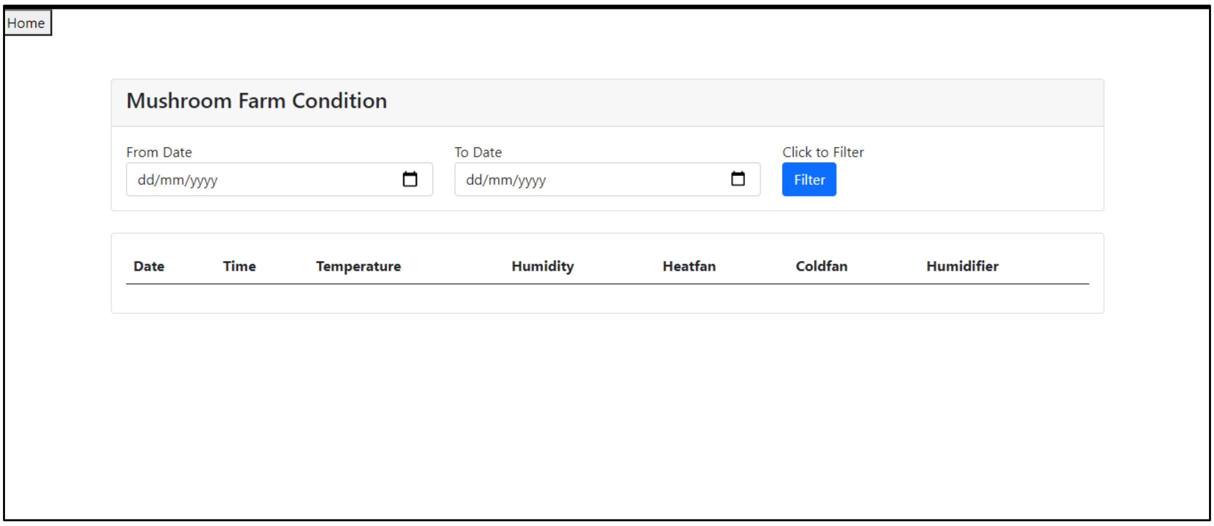


Figure 3.30 Home Page

The main purpose for creating the website is to allow the user to monitor the farmhouse easier. In our website, we allow user to access to the database. We have prepared either to access the database by selecting date between date and just display the full data stored in the database. The query by date method of accessing the database can sort out which is the date the user wants to access, since the amount of data is huge. Then on the website we have the floor plan which is included the floor layout of the farmhouse about where the sensor and control room located. About the info tab, it is a guideline for user to know the condition of the farmhouse after accessing the data from the database. Lastly, we have included the tech support chat, which is allowing user to make a contact with the technician when there is any issue regarding the system or about the sensor.

3.1.2.2 Access Data by Query



Mushroom Farm Condition

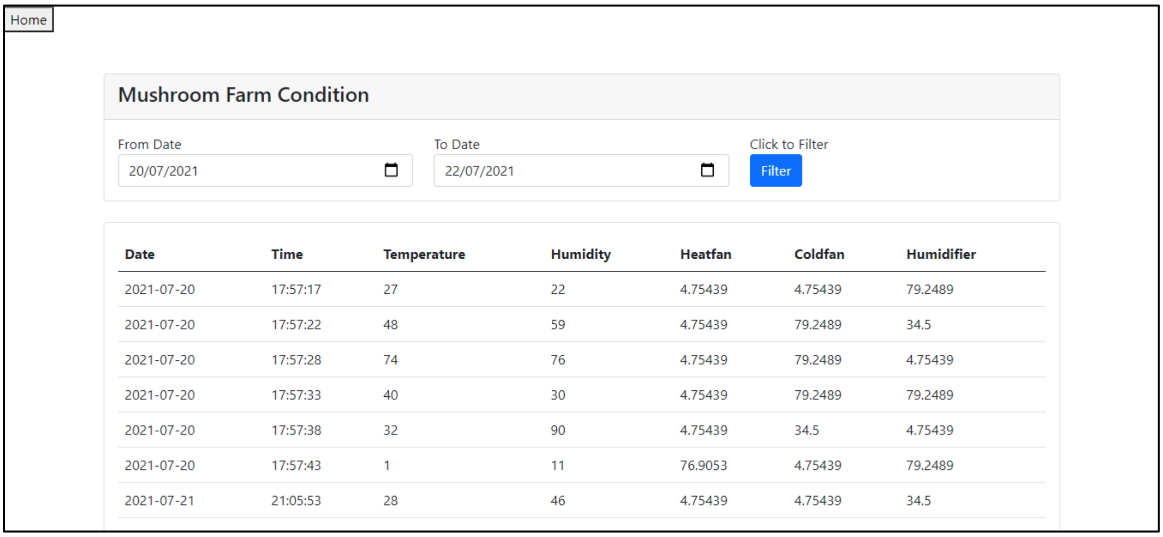
From Date: dd/mm/yyyy

To Date: dd/mm/yyyy

Date	Time	Temperature	Humidity	Heatfan	Coldfan	Humidifier

Figure 3.31: Access Database by Query

From the figure 3.31 can see that we can input from date and to date, then it will only show the data from the database just only between two dates inserted.



Mushroom Farm Condition

From Date: 20/07/2021

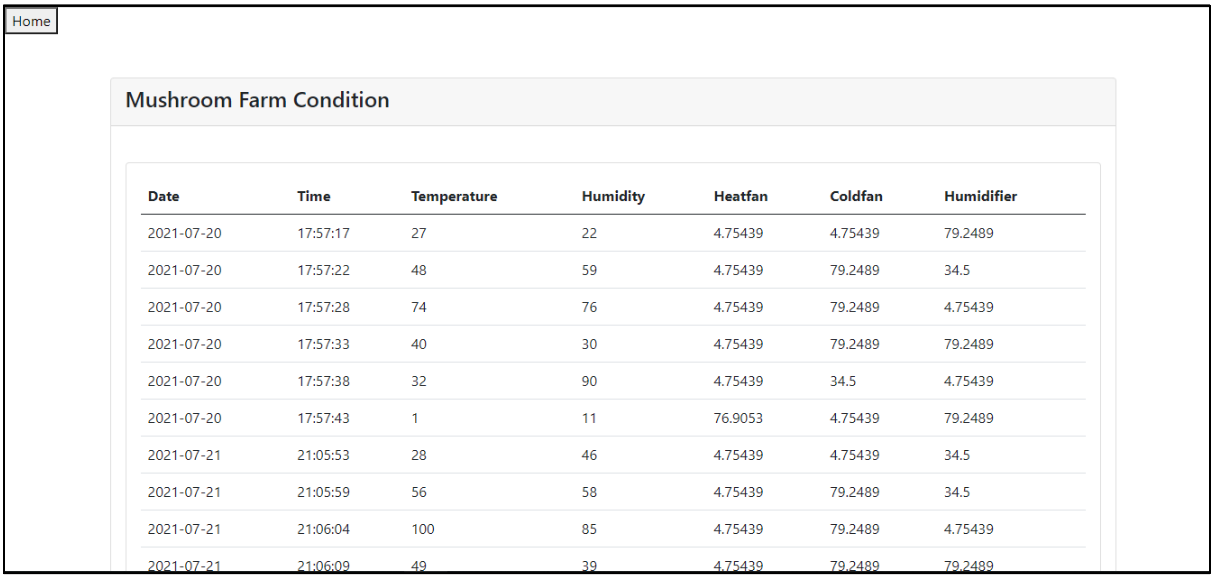
To Date: 22/07/2021

Date	Time	Temperature	Humidity	Heatfan	Coldfan	Humidifier
2021-07-20	17:57:17	27	22	4.75439	4.75439	79.2489
2021-07-20	17:57:22	48	59	4.75439	79.2489	34.5
2021-07-20	17:57:28	74	76	4.75439	79.2489	4.75439
2021-07-20	17:57:33	40	30	4.75439	79.2489	79.2489
2021-07-20	17:57:38	32	90	4.75439	34.5	4.75439
2021-07-20	17:57:43	1	11	76.9053	4.75439	79.2489
2021-07-21	21:05:53	28	46	4.75439	4.75439	34.5

Figure 3.32: Data between two dates

Figure 3.32 is showing the data from the database just because the input of from date and to date is 20/7/2021 and 22/7/2021, so the data will only show between both dates. The data will start showing when the filter button is pressed after inserting the date. For returning to the home webpage, just press the home button on the top left corner.

3.1.2.3 Access Full Data



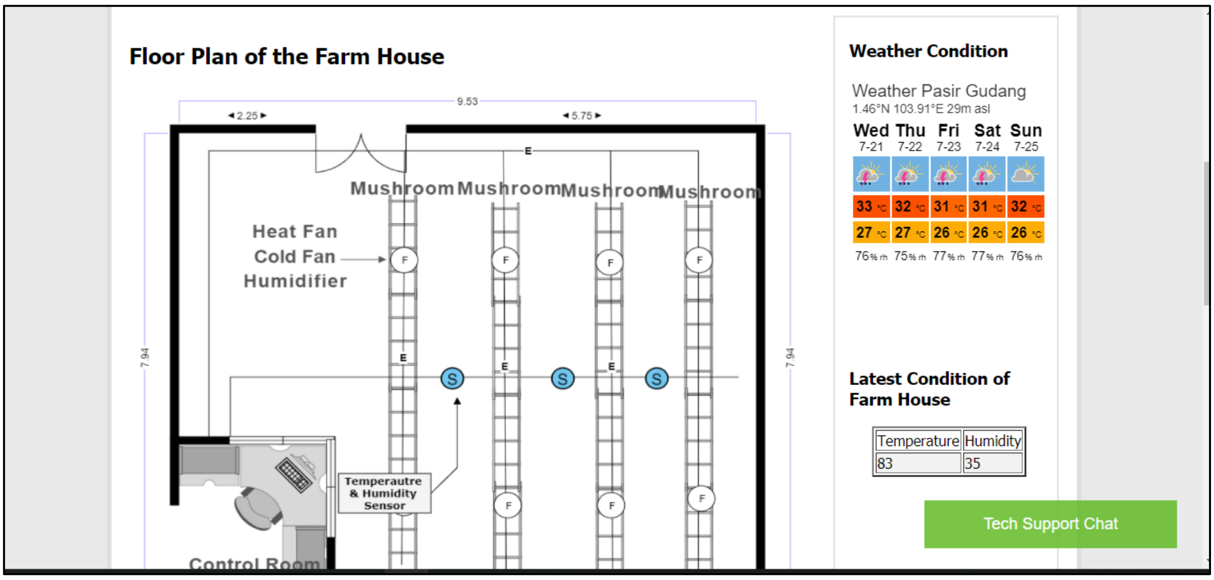
Mushroom Farm Condition

Date	Time	Temperature	Humidity	Heatfan	Coldfan	Humidifier
2021-07-20	17:57:17	27	22	4.75439	4.75439	79.2489
2021-07-20	17:57:22	48	59	4.75439	79.2489	34.5
2021-07-20	17:57:28	74	76	4.75439	79.2489	4.75439
2021-07-20	17:57:33	40	30	4.75439	79.2489	79.2489
2021-07-20	17:57:38	32	90	4.75439	34.5	4.75439
2021-07-20	17:57:43	1	11	76.9053	4.75439	79.2489
2021-07-21	21:05:53	28	46	4.75439	4.75439	34.5
2021-07-21	21:05:59	56	58	4.75439	79.2489	34.5
2021-07-21	21:06:04	100	85	4.75439	79.2489	4.75439
2021-07-21	21:06:09	49	39	4.75439	79.2489	79.2489

Figure 3.33 Access Full Data

Figure 3.32 shows the full data which retrieve from the database, which is good for user when wanted to trace the data without knowing the specific period.

3.1.2.4 Main Page



Weather Condition

Weather Pasir Gudang
1.46°N 103.91°E 29m asl

Wed 7-21	Thu 7-22	Fri 7-23	Sat 7-24	Sun 7-25
33 °C	32 °C	31 °C	31 °C	32 °C
27 °C	27 °C	26 °C	26 °C	26 °C
76% rh	75% rh	77% rh	77% rh	76% rh

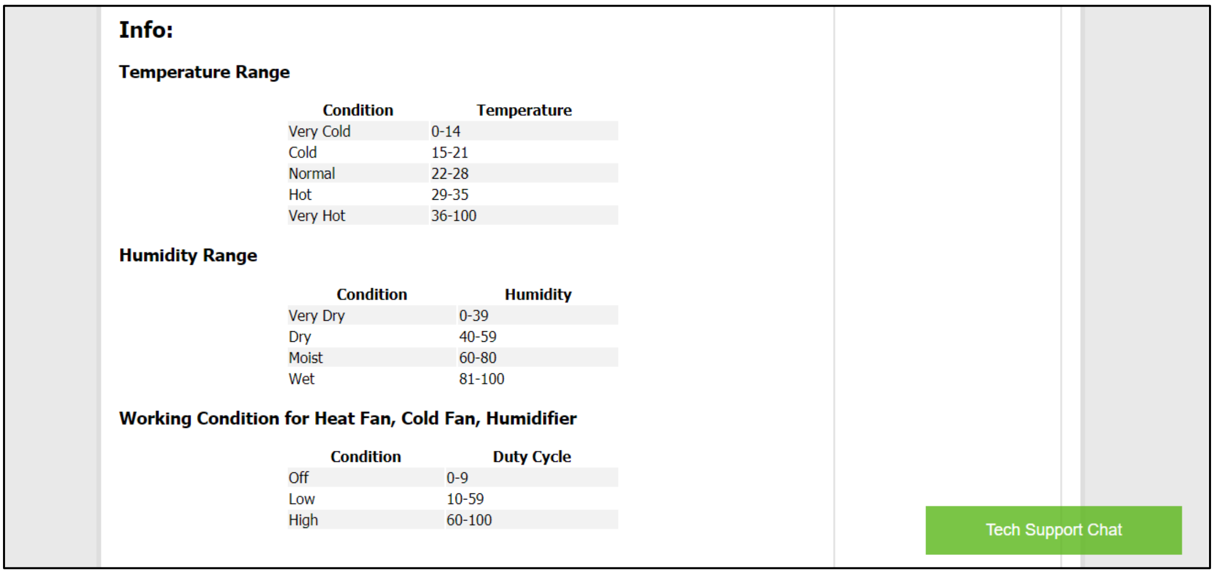
Latest Condition of Farm House

Temperature	Humidity
83	35

Figure 3.34 Main Page

This is the main page of the website, it has the floor plan of the farm house which is allowing the user to know about the situation of the farm house. On the right side we have the weather conditions which is showing 5 days weather conditions. From the table we can gather the minimum and maximum temperature and also the humidity of the weather. Below of the table, user can always get the latest temperature and the humidity in the farmhouse.

3.1.2.5 Info Tab



Info:

Temperature Range

Condition	Temperature
Very Cold	0-14
Cold	15-21
Normal	22-28
Hot	29-35
Very Hot	36-100

Humidity Range

Condition	Humidity
Very Dry	0-39
Dry	40-59
Moist	60-80
Wet	81-100

Working Condition for Heat Fan, Cold Fan, Humidifier

Condition	Duty Cycle
Off	0-9
Low	10-59
High	60-100

Figure 3.35 Info Tab

The info tab included the information about the temperature and humidity range and also the working condition of the heat, cold fan and the humidifier. The reason of these is included is because user will be able to refer what is the condition when accessing the data from the database. If there is any abnormal condition, the user able to make any adjustment quicker.

3.1.2.6 Tech Support Chat Feature

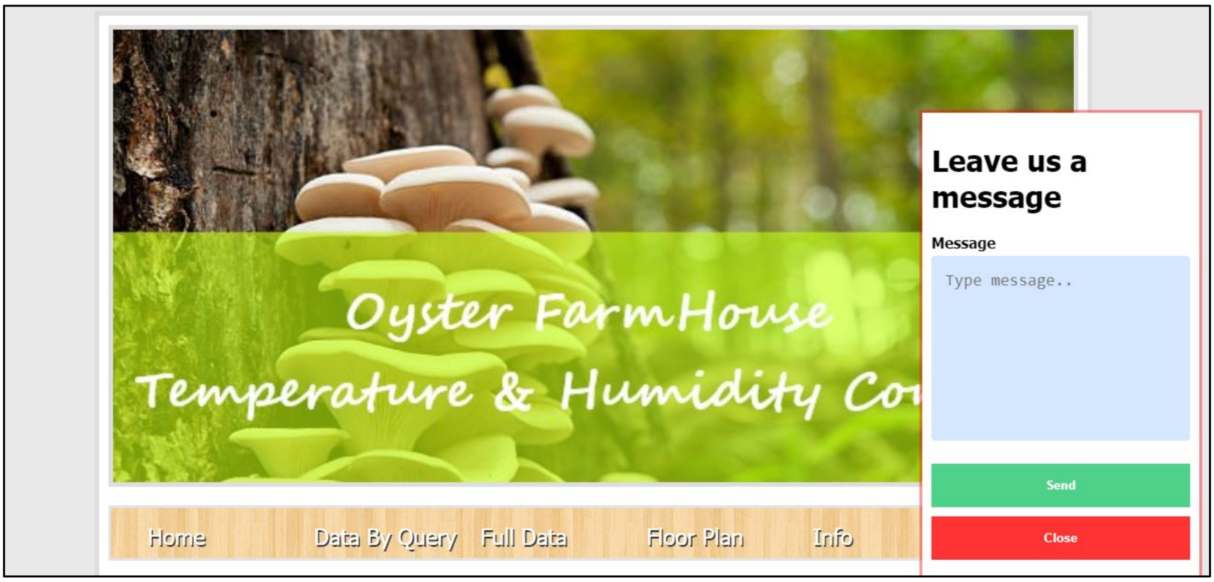
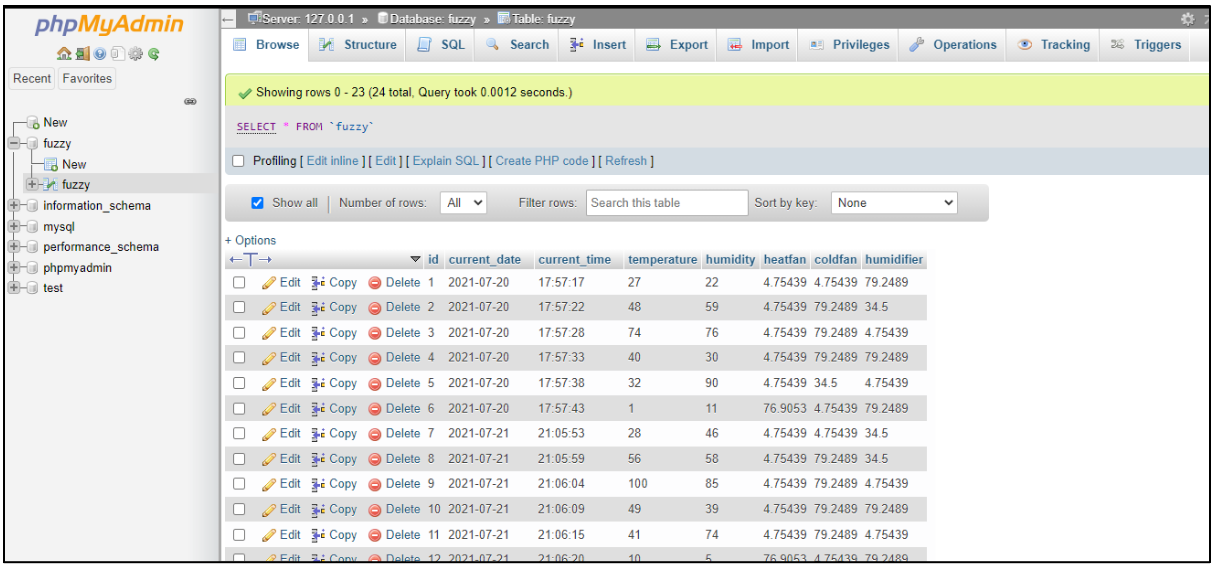


Figure 3.36 Tech Support Chat

This feature is a pop out chat box which allow the user to make a contact to the technician when there is any issue happened on the system and make some repairs on the problem immediately.

3.1.2.7 The Database



✓ Showing rows 0 - 23 (24 total, Query took 0.0012 seconds.)

```
SELECT * FROM `fuzzy`
```

	id	current_date	current_time	temperature	humidity	heatfan	coldfan	humidifier
□ Edit Copy Delete	1	2021-07-20	17:57:17	27	22	4.75439	4.75439	79.2489
□ Edit Copy Delete	2	2021-07-20	17:57:22	48	59	4.75439	79.2489	34.5
□ Edit Copy Delete	3	2021-07-20	17:57:28	74	76	4.75439	79.2489	4.75439
□ Edit Copy Delete	4	2021-07-20	17:57:33	40	30	4.75439	79.2489	79.2489
□ Edit Copy Delete	5	2021-07-20	17:57:38	32	90	4.75439	34.5	4.75439
□ Edit Copy Delete	6	2021-07-20	17:57:43	1	11	76.9053	4.75439	79.2489
□ Edit Copy Delete	7	2021-07-21	21:05:53	28	46	4.75439	4.75439	34.5
□ Edit Copy Delete	8	2021-07-21	21:05:59	56	58	4.75439	79.2489	34.5
□ Edit Copy Delete	9	2021-07-21	21:06:04	100	85	4.75439	79.2489	4.75439
□ Edit Copy Delete	10	2021-07-21	21:06:09	49	39	4.75439	79.2489	79.2489
□ Edit Copy Delete	11	2021-07-21	21:06:15	41	74	4.75439	79.2489	4.75439
□ Edit Copy Delete	12	2021-07-21	21:06:20	10	5	76.9053	4.75439	79.2489

Figure 3.37: phpMyAdmin Database

The figure 3.36 shows about the database from the phpMyAdmin, we have total of 7 types of data stored in the database. We have the record of the time and date when the system is running, it allows user to track back the data based on certain period. Besides that, we have stored the input and output of the system, which is temperature, humidity, heat fan speed, cold fan speed and humidifier. All data is being inserted automatically from Python to PHP, which saved a lot of time if inserting manually.

3.2 Discussion

The central concept of the system is when the Temperature and Humidity Control System is attached with a humidity and temperature sensor, the data collected by the sensor will be the input of the system. The input will determine the value of the output. The input will be processed by the fuzzy system and produce the output. The output is in numerical form, so to ensure the fan and humidifier will work properly, a comparator should be used. The comparator will be able to make the fan and humidifier work correctly.

During the project execution, there are few issues encountered. One of the issues is at the beginning of the system development, we are using the triangular membership function. Still, then the input and output of the fuzzy system are not that accurate, so we replaced it with the Trapezoidal membership function. After the replacement, we can generate proper output based on the required temperature and humidity.

Another issue we face is about the output of the Temperature and Humidity Control System. At first, we are proposed only to make one normal fan as the temperature control. Still, after considering the weather in Malaysia, we should make it two cold and heat fans to make sure that the temperature can increase and decrease according to the unstable weather conditions.

The original plan of the project is to create the system with raspberry pi. Still, due to the Movement Control Order (MCO) carried out in Malaysia, we cannot meet up and work on the project offline, so we decided to make it fully software based.

During creating a website, we faced some difficulties as this is our first time creating a website. This is the way we are getting familiar with PHP, HTML, CSS, and SQL languages. There are so many errors during the process of writing the code of the website, but we manage to make a correction from all the errors and build the website.

CHAPTER 4

PROJECT DESIGN CONSIDERATION

4.1 Health Consideration

As the project contribute to the production of healthy product and have good values pf diet. Oyster mushroom is high in protein, vitamins, minerals, fibre, and other antioxidants such as selenium, which assist in protecting body cells from damage that can lead to chronic diseases and boost the immune system. Oyster mushroom has a low-calorie count, is fat-free, cholesterol-free, gluten-free, and sodium-free. Oyster mushroom consumption appears to lower the chances of obesity, diabetes, cancer, and heart disease and boost the body's immunity system. Since we can benefit from consuming oyster mushrooms, but due to the sky-rocket price of oyster mushrooms, not everyone can afford it. The rising price is because the cost of cultivation of oyster mushrooms is expensive. After all, as a traditional farmhouse, many workers need to be hired to monitor the mushroom. So, implementing the temperature and humidity control system in the cultivation of oyster mushrooms will cut down costs. So, everyone can afford to consume oyster mushrooms that bring a lot of health benefits to humans.

Here is the nutritional content of 1 cup (86 grams) of raw *P.* ostreatus oyster mushrooms

- **Calories:** 28
- **Carbs:** 5 grams
- **Protein:** 3 grams
- **Fat:** <1 gram
- **Fibre:** 2 grams
- **Niacin:** 27% of the Daily Value (DV)
- **Pantothenic acid (vitamin B5):** 22% of the DV
- **Folate:** 8% of the DV
- **Choline:** 8% of the DV
- **Potassium:** 8% of the DV
- **Iron:** 6% of the DV
- **Phosphorus:** 8% of the DV
- **Zinc:** 6% of the DV

Besides from the benefits of the mushroom, the project does not involve any chemical so during the installation of the project, it does not bring any health issue to the user. The Environmental Protection Agency (EPA) has created informational summaries on several substances to outline how individuals might be exposed to them and how their health might be affected. The summaries also explain how chemicals end up in the environment, who controls them, and who to contact for further information.

Hazardous substances can be ingested in a variety of ways, including:

- Water

  Drinking contaminated groundwater or surface water or inadvertently ingesting it while swimming or showering can lead to exposure. Direct skin contact happens during activities such as swimming and showering and is another route of exposure. Visit the MDNR, CDC, and EPA websites for more information on water and harmful substances.
- Soil, sediment, or dust are all examples of soil.

  People can be exposed to harmful chemicals in soil, silt, or dust if they swallow it, breathe it in, or come into contact with it directly on their skin. Children are particularly vulnerable to these types of exposure. Children tend to have frequent hand-to-mouth contact and put non-food things into their mouths during their regular activities. Visit the EPA's website for more information about hazardous substances.
- Air

  When humans breathe in dangerous chemical vapours or air that has been contaminated by hazardous chemicals or dust, they are exposed. The MDNR, CDC, and EPA all provide websites with more information regarding pollutants in the air.
- Food

  Hazardous chemicals can be ingested, exposing people to them. When food comes into contact with harmful chemicals, it might become contaminated. It can also happen at a lower food chain level, such as consuming a poisonous oyster mushroom.

4.2 Safety Consideration

Our automatic system controls the output based on the input to prevent workers' injuries during the working time. In a traditional farmhouse, many workers need to work together. It is hard to avoid accidents that happen in the farmhouse. Since the oyster mushroom farmhouse must be in a confined space to make sure the mushroom is in good condition, when too many workers stay in the confined space for too long, it might cause suffocation problems for the worker. There is no need for workers to manage the farmhouse manually when the system is implemented. Everything can be managed through the computer.

4.3 Environmental Consideration

The temperature and humidity control system are operating on schedule. In terms of reducing energy consumption, it saves a lot of power as it is not running the whole time. Besides that, the fuzzy system will determine when the fan and humidifier need to stop operating when it reaches the suitable temperature and humidity level. A fuzzy system can also control the power of the fan and humidifier, so it can avoid the fan and humidifier using 100% power all the time when operating. Reducing energy use benefits the environment by lowering power plant emissions. Most power plants use coal, crude oil, or other fossil fuels to generate electricity. Although this kind of energy creation is relatively inexpensive, it comes at a cost to our planet: carbon dioxide, sulphur dioxide, and nitrogen oxides are just a few of the by-products of traditional power generation.

The humidifier needs a lot of water to operate. The traditional way of moist the oyster mushroom will waste excess water because humans are unable to determine the humidity of the farmhouse without the help of technology. So, using a fuzzy system to control the humidifier will conserve a lot of water. Conserving water can reduce the impact of drought and water scarcity. Our supply remains stable even though our need for fresh water sources is constantly increasing due to population and industry growth. Even though water eventually returns to Earth via the water cycle, it does not necessarily do so in the same location or the same quantity or quality. We can better safeguard ourselves against future droughts by lowering the amount of water we use.

By having the temperature and humidity control system working on a schedule and controlling the power of the fan and humidifier, we can prevent the unnecessary sound produced by the fan and humidifier because sound pollution is one of the factors why some farmhouses will get reported by the nearby residents frequently. It is not only harmful to humans but also animals. Sound pollution has a huge environmental impact and causes considerable ecological damage, according to the National Park Service (NPS) in the United States. According to experts, sound pollution can disrupt breeding cycles and rearing and hasten the extinction of some species.

4.4 Cultural and Benefit to the Society

The farmers can currently control temperature and humidity to achieve optimal mushroom cultivation using a manual sprayer, this makes it more difficult for farmers to achieve optimal results and requires additional energy to maintain the mushroom house's temperature and humidity during the dry season. If we controlled the temperature and humidity perfectly, it would be easy and beneficial for the people. We would have a big chance to get lots of quantities and the farmers to produce more amounts in less time and less effort.

This project can be beneficial in providing lots of quantities to society. People need the oyster mushroom because of the proteins and nutritional components that are in the mushroom. But because of a poor cultivation technique being used nowadays, it caused low production of oyster mushrooms, so the price is getting higher. So to make sure everyone can afford to buy the oyster mushroom in the market, a good cultivation technique can ensure a high supply of oyster mushrooms in a high demand society.

CHAPTER 5

PROJECT MANAGEMENT AND COSTING

5.1 Project Gantt Chart

Table 5.0: Gantt Chart

Activities	**Week 1**	**Week 2**	**Week 3**	**Week 4**	**Week 5**	**Sem Break**	**Week 6**	**Week 7**	**Week 8**	**Week 9**	**Week 10**	**Week 11**	**Week 12**	**Week 13**	**Week 14**	**Week 15**
	5/4-12/4	**12/4-19/4**	**19/4-26/4**	**26/4-3/5**	**3/5-10/5**	**10/5-17/5**	**17/5-24/5**	**24/5-31/5**	**31/5-7/6**	**7/6-14/6**	**14/6-21/6**	**21/6-28/6**	**28/6-5/7**	**5/7-12/7**	**12/7-19/7**	**19/7-26/7**
Task 1-*Project Planning*	▨	▨	▨													
Task 1.1- **Study parameter humidity(Shukri)* **Study parameter temperature(Fikry)* **Study type of sensor(Raqeeb)* **Study fuzzy logic(Sam)*		▨														
Task 1.2- **Review Fuzzy/Python (Sam&Raqeeb)* **Review HTML,CSS,PHP,* *bootstrap (Fikry&Shukri)*			▨													
Task 2- ***Hardware Procurement***				▨												
Task 2.1- **Identify type of hardware will be use*				▨												
Task 2.2- **Purchasing hardware (Supervisor help us purchase the hardware)*				▨												
Task 3- ***Development***				▨												
Task 3.1- *-Control and monitoring* *-Web/Apps interface*				▨												
Project Management Evaluation by supervisor					▨											
Task 4- *-Improvement on development*									▨	▨	▨					

Project report chapter 1-2											■	■	■		
Project report chapter 1-4											■	■	■		
Project report final														■	
Report Submission															■
Presentation															■

5.2 Project Costing

Throughout the project life cycle, project cost management is estimating, planning, and controlling expenses to stay within the agreed budget. As a result, project cost management is one of the most important pillars of project management, and it applies to every industry, including manufacturing, retail, technology, construction, etc. It aids in creating a financial baseline against which project managers may compare the present status of project costs and, if necessary, realign the project's course.

Table 5.1 Project Costing

No.	Item	Quantity	Price (RM)
1	Codecademy Pro monthly subscription	1	66.45
		Total	66.45

The purpose of Codecademy is to learn and practice Python and PHP within a short period. Python (pandas-Python library, Beautiful Soup-Python Library), Java, Go, JavaScript (jQuery, AngularJS, React.js), Ruby (Ruby on Rails-Ruby framework), SQL, C++, C#, Swift, and Sass, as well as markup languages HTML and CSS, are among the programming languages offered by Codecademy.

5.3 Project Allocation

Task allocation is a working process organization in which responsibilities and workloads for a single task are spread among individuals and organizational units who each conduct their sections or portions of the ordinary activity.

Table 5.2: Task allocation for report

Task	Student's Name			
	Sam	Fikry	Shukri	Raqeeb
Abstract & Abstrak	🟦			🟦
Acknowledgement	🟦			
Approval and declaration	🟦			
Table of contents	🟦	🟦	🟦	🟦
List of tables and figures	🟦	🟦	🟦	🟦
References	🟦	🟦	🟦	🟦
Appendix	🟦			
Chapter 1: Introduction				
1.1 Background	🟦			
1.2 Problem Statement				🟦
1.3 Objective	🟦			
1.4 Scope of Project	🟦			
Chapter 2: Methodology				
2.1 Design Complexity	🟦			
2.1.1 Software Description				
2.1.1.1 PART I-Temperature and Humidity Control System	🟦			
2.1.1.2 PART II-Website/Database		🟦		
2.2 Design Flow and Process	🟦	🟦		

2.2.1 PART I-Temperature and Humidity Control System	🟦			
2.2.2 PART II-Website/Database	🟦	🟦	🟦	
Chapter 3: Results and Discussion				
3.1.1 PART I-Temperature and Humidity Control System	🟦			
3.1.2 PART II-Website/Database			🟦	
3.2 Discussion	🟦		🟦	
Chapter 4: Project Design Consideration				
4.1 Health Consideration	🟦			🟦
4.2 Safety Consideration	🟦			🟦
4.2 Environment Consideration	🟦			🟦
4.4 Cultural and Benefit to the Society	🟦			🟦
Chapter 5: Project Management And Costing				
5.1 Project Gantt Chart	🟦	🟦	🟦	🟦
5.2 Project Costing	🟦			
5.3 Task Allocation	🟦	🟦	🟦	🟦
5.4 Logbook Allocation	🟦	🟦	🟦	🟦
Chapter 5: Conclusion				
Conclusion	🟦			

5.4 Logbook Allocation

Project logbooks are used to keep track of our daily activities from the moment we start the project (with an introduction statement stating what the project is about) until the effort is completed (including the final results, did the project meet the core objectives, etc.). The supervisor of the project suggests logbook allocation.

Table 5.3 Logbook Allocation

Logbook week	**Student's name**			
	Sam	**Shukri**	**Raqeeb**	**Firky**
1	🟦			
2	🟦			
3		🟦		
4			🟦	
5				🟦
6	🟦			
7			🟦	
8		🟦		
9	🟦			
10				🟦
11			🟦	
12		🟦		
13				🟦
14	🟦			

CHAPTER 6

CONCLUSION

This project has developed a system used to control Temperature and Humidity for Oyster Mushroom Cultivation System. The main complexity of the system is the implementation of fuzzy logic. This allows the output to be determined by the input, which is the temperature and humidity of the surrounding. Besides that, a user-friendly interface on web pages is developed to allow users to monitor and record the input and output change. This system can bring a lot of benefits to humans and less harm to the environment.

REFERENCES

[1] "Mamdani's fuzzy inference method," *eMathTeacher: Mamdani's fuzzy inference method - Membership functions*. [Online]. Available: http://www.dma.fi.upm.es/recursos/aplicaciones/logica_borrosa/web/fuzzy_inferencia/funpert_en.htm. [Accessed: 22-Jun-2021].

[2]“skfuzzy," *skfuzzy 0.2 docs - skfuzzy v0.2 docs*. [Online]. Available: https://pythonhosted.org/scikit-fuzzy/. [Accessed: 22-Jun-2021].

[3]M. Watson, "What Are Oyster Mushrooms?," *The Spruce Eats*. [Online]. Available: https://www.thespruceeats.com/what-are-oyster-mushrooms-4172003. [Accessed: 22-Jun-2021].

[4]*Task Allocation*. [Online]. Available: http://www.taskmanagementguide.com/solutions/task-management-solution/task-allocation.PHP. [Accessed: 05-Jul-2021].

[5]“Price Charts,” *Tridge*. [Online]. Available: https://www.tridge.com/prices/chart?from=2021-01-06&to=2021-07-06&period=w¤cy&unit=kg&annotations=insight,article&entries=40876287&isSaveModalOpen=false&isEntrySearchModalOpen=false. [Accessed: 06-Jul-2021].

[6]“How Does Saving Energy Help The Environment," *Save On Energy Blog*, 07-Aug-2018. [Online]. Available: https://www.saveonenergy.com/learning-center/energy-saving-tips/how-does-saving-energy-help-the-environment/. [Accessed: 06-Jul-2021].

[7]M. Evans, "5 Reasons We Should Care About Saving Water," *The Balance Small Business*. [Online]. Available: https://www.thebalancesmb.com/conservation-efforts-why-should-we-save-water-3157877. [Accessed: 06-Jul-2021].

[8]G. P. Cikarge and F. Arifin, "Oyster Mushrooms Humidity Control Based On Fuzzy Logic By Using Arduino ATMega238 Microcontroller," *Journal of Physics: Conference Series*, vol. 1140, p. 012002, 2018.

[9]I. Corporativa, "Noise pollution: how to reduce the impact of an invisible threat?," *Iberdrola*. [Online]. Available: https://www.iberdrola.com/environment/what-is-noise-pollution-causes-effects-solutions. [Accessed: 08-Jul-2021].

[10] "Health Effects from Chemical Exposure," *Health Effects from Chemical Exposure | Hazardous Substances and Sites | Health & Senior Services*. [Online]. Available: https://health.mo.gov/living/environment/hazsubstancesites/healtheffects.php. [Accessed: 11-Jul-2021].

[11] R. Y. Adhitya et al., "Comparison methods of Fuzzy Logic Control and Feed Forward Neural Network in automatic operating temperature and humidity control system (Oyster Mushroom Farm House) using microcontroller," 2016 International Symposium on Electronics and Smart Devices (ISESD), 2016, pp. 168-173, doi: 10.1109/ISESD.2016.7886713.

[12] T. Ross, *Fuzzy Logic with Engineering Applications, 3rd Edition*. John Wiley & Sons, 2010.

[13] A. Gobor, MethanolMan, Chris, and A. G. says:M. 3, "What is PWM and how does it work?," *ekwb.com*, 04-Oct-2019. [Online]. Available: https://www.ekwb.com/blog/what-is-pwm-and-how-does-it-work/. [Accessed: 21-Jul-2021].

[14] "4 Conformance: requirements and recommendations," *Conformance: requirements and recommendations*. [Online]. Available: https://www.w3.org/TR/REC-html40-971218/conform.html#deprecated. [Accessed: 21-Jul-2021].

APPENDICES

Code for the temperature and humidity control system in Python:

```
import numpy as np
import skfuzzy as fuzz
from skfuzzy import control as ctrl
import requests
import schedule
import time
import random


def fuz():
    randomlist =[]
    temperature = ctrl.Antecedent(np.arange(0,100,1),
'temperature')
    humidity = ctrl.Antecedent(np.arange(0,100,1), 'humidity')
    heatfan = ctrl.Consequent(np.arange(0,100,1), 'heatfan')
    coldfan = ctrl.Consequent(np.arange(0,100,1), 'coldfan')
    humidifier = ctrl.Consequent(np.arange(0,100,1),
'humidifier')


###############################################################
############## MEMBERSHIP
    temperature['very_cold'] =
fuzz.trapmf(temperature.universe, [0,0,14,15])
    temperature['cold'] = fuzz.trapmf(temperature.universe,
[11,15,21,22])
    temperature['normal'] = fuzz.trapmf(temperature.universe,
[21,22,28,30])
    temperature['hot'] = fuzz.trapmf(temperature.universe,
[29,29,35,36])
    temperature['very_hot'] =
fuzz.trapmf(temperature.universe, [35,36,100,100])


    humidity['very_dry'] = fuzz.trapmf(humidity.universe,
[0,0,39,40])
    humidity['dry'] = fuzz.trapmf(humidity.universe,
[39,40,59,60])
    humidity['moist'] = fuzz.trapmf(humidity.universe,
[59,60,80,81])
    humidity['wet'] = fuzz.trapmf(humidity.universe,
[79,81,100,100])


    heatfan['off'] = fuzz.trapmf(heatfan.universe, [0,0,9,10])
    heatfan['slow'] = fuzz.trapmf(heatfan.universe,
[5,10,50,60])
```

```
    heatfan['fast'] = fuzz.trapmf(heatfan.universe,
[50,60,100,100])


    coldfan['off'] = fuzz.trapmf(coldfan.universe, [0,0,9,10])
    coldfan['slow'] = fuzz.trapmf(coldfan.universe,
[9,10,59,60])
    coldfan['fast'] = fuzz.trapmf(coldfan.universe,
[59,60,100,100])


    humidifier['off'] = fuzz.trapmf(humidifier.universe,
[0,0,9,10])
    humidifier['low'] = fuzz.trapmf(humidifier.universe,
[9,10,59,60])
    humidifier['high'] = fuzz.trapmf(humidifier.universe,
[59,60,100,100])



rule1=ctrl.Rule((temperature['very_cold']&humidity['very_dry']
),(coldfan['off'],heatfan['fast'],humidifier['high']))

rule2=ctrl.Rule((temperature['cold']&humidity['very_dry']),(co
ldfan['off'],heatfan['slow'],humidifier['high']))

rule3=ctrl.Rule((temperature['normal']&humidity['very_dry']),(
coldfan['off'],heatfan['off'],humidifier['high']))

rule4=ctrl.Rule((temperature['hot']&humidity['very_dry']),(col
dfan['slow'],heatfan['off'],humidifier['high']))

rule5=ctrl.Rule((temperature['very_hot']&humidity['very_dry'])
,(coldfan['fast'],heatfan['off'],humidifier['high']))

rule6=ctrl.Rule((temperature['very_cold']&humidity['dry']),(co
ldfan['off'],heatfan['fast'],humidifier['low']))

rule7=ctrl.Rule((temperature['cold']&humidity['dry']),(coldfan
['off'],heatfan['slow'],humidifier['low']))

rule8=ctrl.Rule((temperature['normal']&humidity['dry']),(coldf
an['off'],heatfan['off'],humidifier['low']))

rule9=ctrl.Rule((temperature['hot']&humidity['dry']),(coldfan[
'slow'],heatfan['off'],humidifier['low']))

rule10=ctrl.Rule((temperature['very_hot']&humidity['dry']),(co
ldfan['fast'],heatfan['off'],humidifier['low']))

rule11=ctrl.Rule(((temperature['very_cold']&humidity['moist'])
|(temperature['very_cold']&humidity['wet'])),(coldfan['off'],h
eatfan['fast'],humidifier['off']))

rule12=ctrl.Rule(((temperature['cold']&humidity['moist'])|(tem
perature['cold']&humidity['wet'])),(coldfan['off'],heatfan['sl
ow'],humidifier['off']))

rule13=ctrl.Rule(((temperature['normal']&humidity['moist'])|(t
emperature['normal']&humidity['wet'])),(coldfan['off'],heatfan
['off'],humidifier['off']))

rule14=ctrl.Rule(((temperature['hot']&humidity['moist'])|(temp
erature['hot']&humidity['wet'])),(coldfan['slow'],heatfan['off
'],humidifier['off']))

rule15=ctrl.Rule(((temperature['very_hot']&humidity['moist'])|
(temperature['very_hot']&humidity['wet'])),(coldfan['fast'],he
atfan['off'],humidifier['off']))
    output_f_ctrl = ctrl.ControlSystem([rule1, rule2, rule3,
rule4, rule5, rule6, rule7, rule8, rule9, rule10, rule11,
rule12, rule13, rule14, rule15])
    output_f = ctrl.ControlSystemSimulation(output_f_ctrl)


    for i in range(2): #random number generator
      n = random.randint(0, 100)
      randomlist.append(n)
    temp = randomlist[0]
    hum = randomlist[1]

    temp= 60
    hum= 90
    output_f.input['temperature'] = temp
    output_f.input['humidity'] = hum
    print("Current Temperature =",temp)
    print("Current Humidity =",hum)
    output_f.compute()
    cfspeed=output_f.output['coldfan']
    hfspeed=output_f.output['heatfan']
    humidifier_s=output_f.output['humidifier']
    print("Heat_Fan =",hfspeed)
    print("Cold_Fan =",cfspeed)
    print("Humidifier =",humidifier_s)
    coldfan.view(sim=output_f)
    heatfan.view(sim=output_f)
    humidifier.view(sim=output_f)


    url = 'http://localhost:8080/GDP/post.PHP' #PHP file run
for post request
```

```
    val = {"temp" : temp, "hum" : hum, "heatfan" : hfspeed,
"coldfan" : cfspeed, "humidifier" : humidifier_s}
    r = requests.post(url, data=val)
    print(r.text)
    print(r)


schedule.every(3600).seconds.do(fuz)

while 1:
        schedule.run_pending()
```

Code for POST request on PHP:

```
    <?PHP
     $Temperature=0;
     $Humidity=0;
     $HeatFan=0;
     $ColdFan=0;
     $Humidifier=0;
    $Temperature
= !empty($_POST["temp"])?($_POST["temp"]):'' ;
    $Humidity = !empty($_POST["hum"])?($_POST["hum"]):'';
     $HeatFan
= !empty($_POST["heatfan"])?($_POST["heatfan"]):'';
     $ColdFan
= !empty($_POST["coldfan"])?($_POST["coldfan"]):'';
     $Humidifier
= !empty($_POST["humidifier"])?($_POST["humidifier"]):'';
     require 'insert.PHP';
   ?>
```

Code for insert data into database on PHP:

```
<?PHP
/* Attempt MySQL server connection. Assuming you are running
MySQL
server with default setting (user 'root' with no password) */
$link = mysqli_connect("localhost", "root", "", "fuzzy");

// Check connection
if($link === false){
    die("ERROR: Could not connect. " .
mysqli_connect_error());
}

// Attempt insert query execution
$sql = "INSERT INTO fuzzy
(temperature,humidity,heatfan,coldfan,humidifier) value
('$Temperature','$Humidity','$HeatFan','$ColdFan','$Humidifier
')";
if(mysqli_query($link,$sql)){
    echo "Records inserted successfully.";
} else{
    echo "ERROR: Could not able to execute $sql. " .
mysqli_error($link);
}

// Close connection
mysqli_close($link);
?>
```

Code for the website:

```
<!DOCTYPE html>
<html>
    <head>
        <meta http-equiv="Content-Type" content="text/html;
charset=UTF-8">
        <title><?php echo $title; ?></title>
        <link rel="stylesheet" type="text/css"
href="Styles/Stylesheet.css" />
    </head>
    <body>

        <div id="wrapper">
            <div id="banner">
            </div>
            <nav id="navigation">
                <ul id="nav">
                    <li><a href="index.php">Home</a></li>
                    <li><a href="query.php">Data By
Query</a></li>
                    <li><a href="test.php">Full Data</a></li>
                    <li><a href="#floorplan">Floor
Plan</a></li>
                    <li><a href="#info">Info</a></li>
                </ul>
            </nav>

           <div id="content_area">
               <div id="floorplan">
               <h2> Floor Plan of the Farm House </h2>
               <img src="Images/Floor_Plan.jpg" alt="Floor
Plan" width ="780" height="600">
               <p></p>
               </div>
               <div id="info">
               <h2> Info: </h2>
               <h3> Temperature Range </h3>
               <table class="center">
  <tr>
    <th>Condition</th>
    <th>Temperature</th>
  </tr>
  <tr>
    <td>Very Cold</td>
    <td>0-14</td>
  </tr>
  <tr>
    <td>Cold</td>
    <td>15-21</td>
  </tr>
    <tr>
    <td>Normal</td>
    <td>22-28</td>
  </tr>
      <tr>
    <td>Hot</td>
    <td>29-35</td>
  </tr>
      <tr>
    <td>Very Hot</td>
    <td>36-100</td>
  </tr>
</table>
               <h3> Humidity Range </h3>
               <table class="center">
  <tr>
    <th>Condition</th>
    <th>Humidity</th>
  </tr>
  <tr>
    <td>Very Dry</td>
    <td>0-39</td>
  </tr>
  <tr>
    <td>Dry</td>
    <td>40-59</td>
  </tr>
    <tr>
    <td>Moist</td>
    <td>60-80</td>
  </tr>
      <tr>
    <td>Wet</td>
    <td>81-100</td>
  </tr>
</table>

               <h3> Working Condition for Heat Fan, Cold Fan,
Humidifier </h3>
               <table class="center">
  <tr>
    <th>Condition</th>
    <th>Duty Cycle</th>
  </tr>
  <tr>
    <td>Off</td>
    <td>0-9</td>
  </tr>
  <tr>
    <td>Low</td>
    <td>10-59</td>
  </tr>
```

```
    <tr>
    <td>High</td>
    <td>60-100</td>
  </tr>
</table>
               </div>
           </div>

            <div id="sidebar">
             <h3> Weather Condition </h3>
<iframe
src="https://www.meteoblue.com/en/weather/widget/daily?geoloc=
detect&days=5&tempunit=CELSIUS&windunit=KILOMETER_PER_HOUR&pre
cipunit=MILLIMETER&coloured=coloured&pictoicon=0&pictoicon=1&m
axtemperature=0&maxtemperature=1&mintemperature=0&mintemperatu
re=1&windspeed=0&windgust=0&winddirection=0&uv=0&humidity=0&hu
midity=1&precipitation=0&precipitationprobability=0&spot=0&pre
ssure=0&layout=light"  frameborder="0" scrolling="NO"
allowtransparency="true" sandbox="allow-same-origin allow-
scripts allow-popups allow-popups-to-escape-sandbox"
style="width: 220px; height: 300px"></iframe><div><!-- DO NOT
REMOVE THIS LINK --><a
href="https://www.meteoblue.com/en/weather/week/index?utm_sour
ce=weather_widget&utm_medium=linkus&utm_content=daily&utm_camp
aign=Weather%2BWidget" target="_blank"></a></div>
               <h3>Latest Condition of Farm House</h3>

<table border="2" class="center">
  <tr>
    <td>Temperature</td>
    <td>Humidity</td>
  </tr>

<?php
$link = mysqli_connect("localhost", "root", "", "fuzzy");

// Check connection
if($link === false){
    die("ERROR: Could not connect. " .
mysqli_connect_error());
}
$records = mysqli_query($link,"SELECT * FROM fuzzy ORDER BY id
DESC LIMIT 1"); // fetch data from database

while($data = mysqli_fetch_array($records))
{
?>
  <tr>
    <td><?php echo $data['temperature']; ?></td>
    <td><?php echo $data['humidity']; ?></td>
  </tr>
```

```
<?php
}
?>
</table>

<?php mysqli_close($link); // Close connection ?>

            </div>

<button class="openChatBtn" onclick="openForm()">Tech Support
Chat</button>
<div class="openChat">
<form method="post">
<h1>Leave us a message</h1>
<label for="msg"><b>Message</b></label>
<textarea placeholder="Type message.." name="msg"
required></textarea>
<button type="submit" class="btn">Send</button>
<button type="button" class="btn close" onclick="closeForm()">
Close
</button>
</form>
</div>
<script>

document .querySelector(".openChatBtn") .addEventListener("cli
ck", openForm);
   document.querySelector(".close").addEventListener("click",
closeForm);
   function openForm() {
      document.querySelector(".openChat").style.display =
"block";
   }
   function closeForm() {
      document.querySelector(".openChat").style.display =
"none";
   }
</script>
            <footer>
                <p>All rights reserved</p>
            </footer>
        </div>
    </body>
</html>

<?php

if(isset($_POST['msg']))
{
$data=$_POST['msg'];
$fp = fopen('form.txt', 'a');
```

```
fwrite($fp, $data);
fclose($fp);
}
?>
```

Code For Website (CSS)

```
body
{
    font-family: lucida grande ,tahoma,verdana,arial,sans-
serif;
    background-color: #e9e9e9;
}

body p
{
    font-size: 0.8em;
    line-height: 1.28;
}

#wrapper
{
    width: 1080px;
    background-color: white;
    margin: 0 auto;
    padding: 10px;
    border: 5px solid #dedede;
}

#banner
{
    background-image: url(../Images/header.jpg);
    background-repeat: no-repeat;
    background-size: cover;
    border: 5px solid #dedede;
    height: 500px;
}

#content_area
{
    float: left;
    width: 750px;
    margin: 20px 0 20px 0;
    padding: 10px;
}

#sidebar
{
    float: right;
    width: 250px;
    height: 400px;
    margin: 20px 10px ;
    padding: 6px 8px 1600px 16px ;
    border: 2px solid #E3E3E3;
}

footer
{
    clear: both;
    width: auto;
    height: 40px;
    padding: 10px;
    border: 3px solid #E3E3E3;
    text-align: center;
    color: #fff;
    text-shadow: 0.1em 0.1em #333;
    background-image: url(../Images/bar_background.png);
}

#navigation
{
    height: 60px;
    border: 3px solid #E3E3E3;
    margin-top: 20px;
    text-shadow: 0.1em 0.1em #333;
    background-image: url(../Images/bar_background.png);
}

#nav
{
    list-style: none;
}

#nav ul
{
    margin: 0;
    padding: 0;
    width: auto;
    display: none;
}

#nav li
{
    font-size: 24px;
    float: left;
    position: relative;
    width: 180px;
    height: 50px;
}

#nav a:link, nav a:active, nav a:visited
{
    display: block;
    color: #fff;
    text-decoration: none;
}

#nav a:hover
{
    color: lightblue;
}

.imgLeft
{
    float: left;
    width: 240px;
    height: 150px;
    margin: 0px 10px 10px 0;
    padding: 10px;
}

.imgRight
{
    float: right;
    width: 200px;
    height: 250px;
    margin: 0px 0 10px 10px;
    padding: 10px;
}


/* Style inputs with type="text", select elements and
textareas */
input[type=text], select, textarea {
  width: 100%; /* Full width */
  padding: 12px; /* Some padding */
  border: 1px solid #ccc; /* Gray border */
  border-radius: 4px; /* Rounded borders */
  box-sizing: border-box; /* Make sure that padding and width
stays in place */
  margin-top: 6px; /* Add a top margin */
  margin-bottom: 16px; /* Bottom margin */
  resize: vertical /* Allow the user to vertically resize the
textarea (not horizontally) */
}

/* Style the submit button with a specific background color
etc */
input[type=submit] {
  background-color: #04AA6D;
  color: white;
  padding: 12px 20px;
  border: none;
  border-radius: 4px;
  cursor: pointer;
}

/* When moving the mouse over the submit button, add a darker
green color */
input[type=submit]:hover {
  background-color: #45a049;
}

/* Add a background color and some padding around the form */
.container {
  border-radius: 5px;
  background-color: #f2f2f2;
  padding: 20px;
}

table {
  width: 50%;
}

tr:nth-child(even) {
  background-color: #f2f2f2;
}

.center {
  margin-left: auto;
  margin-right: auto;
}

* {
      box-sizing: border-box;
   }
   .openChatBtn {
      background-color: #59B618;
      color: white;
      padding: 16px 20px;
      border: none;
      font-weight: 500;
      font-size: 18px;
      cursor: pointer;
      opacity: 0.8;
      position: fixed;
      bottom: 23px;
      right: 28px;
      width: 280px;
   }
   .openChat {
      display: none;
      position: fixed;
      bottom: 0;
      right: 15px;
      border: 3px solid #ff08086b;
      z-index: 9;
   }
   form {
      max-width: 300px;
      padding: 10px;
      background-color: white;
   }
   form textarea {
      width: 100%;
      font-size: 18px;
      padding: 15px;
      margin: 5px 0 22px 0;
      border: none;
      font-weight: 500;
      background: #d5e7ff;
      color: rgb(0, 0, 0);
      resize: none;
      min-height: 200px;
   }
   form textarea:focus {
      background-color: rgb(219, 255, 252);
      outline: none;
   }
   form .btn {
      background-color: rgb(34, 197, 107);
      color: white;
      padding: 16px 20px;
      font-weight: bold;
      border: none;
      cursor: pointer;
      width: 100%;
      margin-bottom: 10px;
      opacity: 0.8;
   }
   form .close {
      background-color: red;
   }
   form .btn:hover, .openChatBtn:hover {
      opacity: 1;
   }
```