Self-reliant Mushroom Farm System

CHIRAPAT SUNINHONG
PHUPIRAT RAMMASITH
THANAWIN KOJIMA WACHIRATONGKUM

**A PROJECT SUBMITTED IN PARTIAL
FULFILLMENT OF THE REQUIREMENT FOR THE DEGREE
BACHELOR OF ENGINEERING IN COMPUTER ENGINEERING**

**FACULTY OF ENGINEERING & INTERNATIONAL COLLEGE
MAHIDOL UNIVERSITY**
2019

SELF-RELIANT MUSHROOM FARM SYSTEM

CHIRAPAT SUNINHONG
PHUPIRAT RAMMASITH
THANAWIN KOJIMA WACHIRATONGKUM

A PROJECT SUBMITTED IN PARTIAL
FULFILLMENT OF THE REQUIREMENT FOR THE DEGREE
BACHELOR OF ENGINEERING IN COMPUTER ENGINEERING

FACULTY OF ENGINEERING & INTERNATIONAL COLLEGE
MAHIDOL UNIVERSITY
2019

Computer Engineering Project

entitled

SELF-RELIANT MUSHROOM FARM SYSTEM

____
Mr.Phupirat Rammasith
Researcher

____
Mr.Chirapat Suninhong
Researcher

____
Mr.Thanawin K.Wachiratongkum
Researcher

____
Lalita Narupiyakul, Ph.D.
Ph.D. (Computer Science)
Ph.D. (Electrical and Computer Engineering)
Advisor

Thesis

entitled

SELF-RELIANT MUSHROOM FARM SYSTEM

was submitted to the Faculty of Engineering & International College,
Mahidol University
For the degree of Bachelor of Engineering (Computer Engineering)

On
April 10, 2020

………………………………..
Lalita Narupiyakul, Ph.Ds.
Ph.D. (Computer Science)
Ph.D. (Electrical and Computer Engineering)
Chair Committee

……………………………
Vasin Suttichaya, Ph.D.
Ph.D. (Computer Science)
Committee

………………………………..
Asst.Prof. Konglit Hunchangsith, Ph.D.
Ph.D. (Information Technology)
Committee

BIOGRAPHY

NAME	**Mr.Phupirat Rammasith**
DATE OF BIRTH	**13 July 1994**
BIRTHPLACE	**Bangkok**
EDUCATION BACKGROUND	**Kasintorn Saint Peter, 1999-2011** **High school education**
MOBILE PHONE	**+66 – 83 – 035 1311**
E-MAIL	**phupiratrammasith@gmail.com**

NAME	**Mr.Chirapat Suninhong**
DATE OF BIRTH	**24 February 1998**
BIRTHPLACE	**Chiang Mai**
EDUCATION BACKGROUND	**Cambridge College (Thailand), 2014-2016** **High school education**
MOBILE PHONE	**+66 – 63- 4463850**
E-MAIL	**chirapat@me.com**

NAME	**Mr.Thanawin Kojima Wachiratongkum**
DATE OF BIRTH	**17 June 1997**
BIRTHPLACE	**Bangkok**
EDUCATION BACKGROUND	**Assumption College Thonburi, 2003-2015** **High school education**
MOBILE PHONE	**+66 – 81 – 136 4343**
E-MAIL	**thanawin.putter1@gmail.com**

ACKNOWLEDGEMENT

We would like to express our deep gratitude to Dr.Lalita Narupiyakul, Dr.Vasin Suttichayaa and Dr.Konglit Hunchangsith, our research supervisors, for their patient guidance, enthusiastic encouragement and useful critiques of this research work. We would also like to thank Mr.Prasong, for his advice and assistance in mushroom farming. Our grateful thanks are also extended to Mr.Jetthana Rammasith for his help in model design and ensuring that everything is going according to the schedule.

We would also like to extend our thanks to the technicians of the Department of Computer Engineering for their help in offering us the resources throughout our experiment.

Finally, we wish to thank our parents for their support and encouragement throughout our study.

Phupirat Rammasith
Chirapat Suninhong
Thanawin K.Wachiratongkum

Self-reliant Mushroom Farm System

STUDENTS Mr. Phupirat Rammasith ICCI 5881103
Mr. Chirapat Suninhong ICCI 5881018
Mr. Thanawin K.Wachiratongkum ICCI 5980103
DEGREE Bachelor of Engineering (Computer Engineering)
PROJECT ADVISOR Lalita Narupiyakul, Ph.Ds.
DATE OF GRADUATION March 12, 2020

ABSTRACT

The Embedded System is viewed to be the heart of the study of computer engineering. The team is extremely passionate to make use of the knowledge that our team has gained throughout the program into the creation of this project. Our project focuses on mushrooms, as they are one of the most consumed and nutritious foods in Thailand. The mushroom farming processes are extremely tedious and it requires a lot of farmer's attention to maintaining the perfect environment for mushrooms constantly. The objective of this project is to create a system that is self-reliant, while the farmers can monitor the system from afar via our controllable web system.

KEYWORDS: Mushroom farm/ IoT devices/ Smart farm.

38 pages

CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF FIGURES (cont.)

LIST OF FIGURES (cont.)

Figure **Page**

CHAPTER I
INTRODUCTION

1.1 Background and Motivation

Thailand is well known for being "the kitchen of the world". As for a very long time, the agricultural industry has contributed highly to the GDP of the country. However, the shocking fact is that investing and shares have significantly decreased in the last few decades. According to The Global Economy (2019) [1], GDP shares from agriculture in Thailand is decreased from 37.5% to 9.8% in the last 50 years, even though it serves as the country's backbone [2]. The reasons, that cause the deterioration in the agricultural industry, are new employment opportunities, a complication in monitoring and climate change. These drive the newer generations to shift from staying in the homeland area to work in the city [3].

In order to define the word "agriculture" in this project, the term agriculture is divided into 2 scales, which are macro-level farming and micro-level farming. Macro-level farming is defined as the agricultural industry, which requires a lot of employees and largely contributes to the economy. This involves a business that requires big farms and factories to function. Micro-level farming, defined in this project, refers to the individual farming activities in the household. It is popular when the family grows plants in the house for decorating purposes or consuming purposes nevertheless, both scales require a lot of contributions and time. This project is dedicated to research about mushroom farming, which is one of the most consumed agricultural goods in Thailand. A big proportion of farmers find them beneficial as the main source of lower to medium farmer's income is generated from harvesting mushrooms.

This project adapts the idea of hydroponics farming since mushroom farms require a lot of attention. Minor changes in the environment can heavily affect the cultivation of the mushroom. The factors which are temperature, humidity, and microparticles in the air, dust and spore can lower the quality of the yield. The closed-system farm is more suitable for mushrooms because it is practical for farmers to maximize the outcome with relative ease.

1.2 Objectives

The purpose of this project is to develop a model that will make mushroom farming easier without reducing the quality of the mushroom. While a web application is developed to allow users to remotely monitor the system. Additionally, this farm can detect changes in the environment. It can report the users in real-time data and allow users to control the farm no matter where they are because the system is also connected to the internet. Our farm is designed as a closed system, so it is efficient for mushrooms. Despite the fact that each species needs to be treated differently, the system can provide the best circumstances for mushrooms. This project is advised by the professional mushroom farmers during the research period.

The project was divided into four stages: research, experiment, implementation, and finalization. In the first stage, the previous research and models are reviewed to solve many problems. The RaspberryPi microcontroller is embedded in our system with the sensors namely: temperature sensor and humidity sensor. The system connects to the server on the internet to control the embedded system. The second stage is implementation. In this stage, the ventilators and pumps are implemented. The farm is able to maintain its closed system. It is also necessary to design the system such that the wiring and the operating of the system are waterproof.

Lastly, a web application is created. The application is easy to use and responsive to any device. Upon this stage, the flow of connectivity between the web application, the database and the system should be consistent. This stage is also used as spare time for fixing bugs, improving the system if the previous stages are overdue.

1.3 Scope:

- The closed system mushroom farming
- Website to monitor and control the threshold of the humidity and temperature

1.4 Expected Result:

- The system can control the humidity and temperature under the satisfaction range
- The website will be able to display the temperature and humidity in a real-time manner
- Users can modify the temperature and humidity threshold for controlling the system on the website

1.5 Timeframe

Table 1.1 Time schedule plan of the project

Timeframe	2018	2019										2020
	Dec	Jan	Feb	Mar	April	May	June	July	Aug	Oct	Nov	Dec	Jan
Research													
Design													
Adjustment						x			x				
Experiment													
Evaluation													

From table 1.1, timeframes are divided into 5 slots. In December and January 2018, the team will mainly focus on researching. Mushroom species, hardware, and the software are analyzed to find the most suitable option for the project. The rough design of the project is built during February to April 2019. From May 2019 until August 2019 are taken as to make adjustments in the project to achieve the best result. During this period, issues that are found will be fixed. Hardware, software, and models are finalized and ready to be experimented. For the next step, the model is tested to see how mushrooms will grow under different circumstances. Lastly, the experimental results are evaluated and discussed to conclude the result of the experiment.

CHAPTER II
LITERATURE REVIEWS

2.1 Model/Mushroom Review

The project mainly focuses on mushroom farming, due to time and the scale factors of the project. Mushrooms, which have a very short life span and harvesting interval, would be perfect fit. The species of the mushroom that are used in the project is "Oyster Mushroom".

Figure 2.1 The traditional way of growing Oyster Mushroom

Figure 2.1 shows the traditional way of growing Oyster Mushrooms, which is one of the most common types of cultivated mushrooms in the world. It also contributes a lot to Thailand's economy and is consumed widely. The species of mushroom grows naturally on and near trees in temperate and subtropical forests around the world. They are also grown commercially in many countries. They can be dried and are typically eaten cooked [4]. According to Watson, the mushrooms are widely consumed because of their mild, savory flavor complemented with their delicate texture. The mushrooms typically have broad, thin, oysters- or fan-shaped caps and are white, gray, or tan, with gills lining the underside. The caps are sometimes frilly-edged and can be found in clusters of small mushrooms or individually as larger mushrooms.

Although it is not very complicated for mushrooms to be grown, it requires an extreme amount of attention on the farm. Firstly, the environment has to be controlled at 27 Celsius, while the humidity must be at 80% [5]. Contrary to popular belief that mushrooms need to grow in a dark environment, the sunlight is preferable as long as the humidity is high. Secondly, if the environment is suitable for the mushroom, a nice white growth of mycelium will take over the entire mushroom bag. In this stage, if there is green mold, it is most likely contaminated and it will not grow. Mycelium looks like white, furry cobwebs, and it is expected to see it develop within two to three weeks [6].

Figure 2.2 Full grown mushrooms

Figure 2.2 shows the perfectly grown mushrooms that this system aims to achieve. The system is extremely useful to the mushroom farming industry because it helps prevent the farm from these common problems: contamination, temperature and insufficiency moisture in air. The reason is that farmers usually leave the system exposed to the environment so they can come and control the system manually, while with this closed system implemented, the system can be constantly supervised and monitored without the risk of human error. Mushrooms are also hypersensitive. The external influence on the system could interfere with these values and make the controlled factors fluctuate.

2.2 Hardware Review

The central processing unit of this system is Raspberry Pi3 shown in figure 2.3. This specific microcontroller is an open source electronics platform which is the most popular and made the connection between hardware and software very easy. It reads input signals from sensors such as light sensors and finger buttons and it connects to the internet and controls the embedded system.

Figure 2.3 Raspberry Pi Board

The main sensors that are initially planned to be put in the embedded system are temperature sensors and humidity sensors. It is very essential as specific humidity and temperature play a very vital role in farming. It measures and calculates the perfect amount of water that will be sprayed over the mushroom bags.

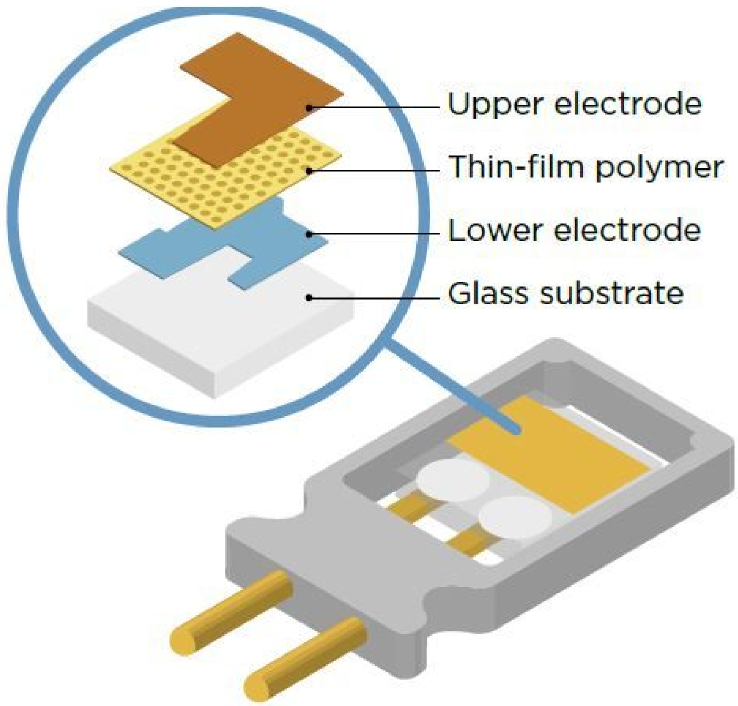

Figure 2.4 The components of the humidity sensor

According to EFY (2018) [7] , a humidity sensor or is professionally known as a hygrometer which is shown in figure 2.4 will be able to measure and report both air temperature and level of moisture. This sensor works by measuring the relative humidity in the air between two electrodes between the strips of metal oxide. If the relative humidity in the air is changed, then the electrical capacity in the metal oxide will change too. Therefore, this particular sensor is extremely useful in measuring the humidity in the air. Despite the fact that DHT-11 is widely used, the sensor chosen in this project is AM2320. Due to the fact that the data sent and read by AM2320 models are more stable and consistent.

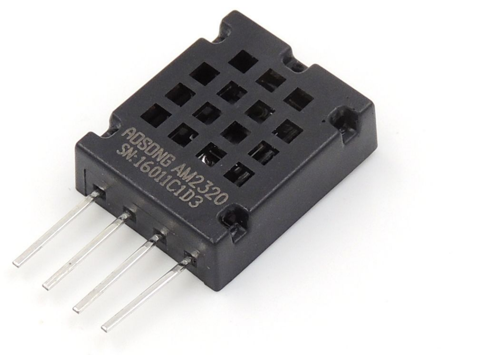

Figure 2.5 Humidity and temperature sensor model AM2320 (Aosong, 2014.) [8]

The initial sensor used in this project is shown in Figure2.5. However, this farming system is designed in such a way that the water is sprung throughout the model. The AM2320 model is not waterproof. Measuring the temperature directly from them will not be accurate. AM2320 has an exposed metal circuit. This can lead to the short circuit or create an error if the water spray directly hits the sensors. For this specific reason, the temperature sensor model, DS18B20 which is shown in figure 2.6 is used. It is waterproof and very durable, which is appropriate for the project. This can be used as the substitute for measuring temperature from AM2320 as mentioned earlier.

Figure 2.6 Temperature sensor, model DS18B20 (Maxim Integrated, 2007) [9]

2.3 Software Review

The main tools used on the website are Bootstrap, HTML, PHP Netpie and CanvasJS. According to the Bootstrap website [10], it is the world's most popular framework for building responsiveness on the webpage. Moreover, it is used because it is free and can easily create a responsive design website suitable for this project. Netpie is chosen to be used because it is free and easy to implement according to the Netpie's document [11], Netpie takes care of the tedious work, so the user can focus on the IoT applications. Additionally, HTML and PHP are used together to create an easy, secured and protected authentication in the website.

CHAPTER III

METHODOLOGY

3.1 Overview

The scope of this project focuses on the cultivation of mushrooms specifically because they have a very short harvesting period and the suitable environmental factors are easy to control. Mushrooms are additionally considered a good option because they are highly nutritious and are fairly cheap to cultivate.

Figure 3.1 The traditional mushroom farm model.

Figure 3.1 shows the traditional way of mushroom farming. The model suggested closed system cultivation of mushrooms which is proven to be more effective. With all of the factors considered, adapting the traditional mushroom farming model with a closed environment is the best model for this experiment.

For the precision of the experiment, the information was collected from the real conventional mushroom farm in Pathumthani as shown in figure 3.2. The information was provided by the farmer who is experienced in farming the mushroom for more than 10 years for the typical process of treating, planting and growing the mushroom effectively. Mushrooms that have different yielding periods are purchased for the experiment to see whether the model is effective for all the range of growing.

Figure 3.2 The mushroom farm in Pathumthani

The information gathered from the professional mushrooms farmers are concluded and listed below

1) Mushrooms need to be watered at least once a day
2) The water must be foggy so that it is able to reach the mushrooms thoroughly.
3) The water must be sprayed on top or at the back of the mushroom bags. Watering on the mushroom directly is forbidden because it is likely to bruise the texture of the mushrooms.
4) The ideal temperature should be below 30 °C for the best result, however in a tropical country like Thailand, this condition is almost impossible to achieve.
5) The air moisture is controlled to be around 80% in order to keep the best condition and prevent the unharvested mushrooms from drying out.

3.1.1 Experiment Preparation:

Figure 3.3 Mushroom bags

Mushroom spores are stored with sawdust in plastic bags as shown in figure 3.3. The mushroom bags were divided into 2 different groups, which are the controlled group and normal group. The life period of the mushroom will be equally divided, 4 for each. The controlled group will be planted under our prepared environment, that controls by our model and records to show the difference of the experimentation

Figure 3.4 Mushroom bags being divided into groups

Figure 3.4 shows a part of the experiment where mushroom bags are divided into 4 groups. Each group consists of 8 bags and they are classified into 4 batches according to their expected harvesting time. The experimental time is 3 days, 5 days, 7 days and 15 days for each Black Hungary oyster mushroom batches.

3.2 Model Design

Figure 3.5 shows the model that is used in a study temperature and humidity control system in the mushroom greenhouse from Rajamangala University of Technology Suvarnnabhumi National Conference [RUSNC] in 2015 [12] , in which the project focuses on developing a further implementation of IoT and embedded systems from this model.

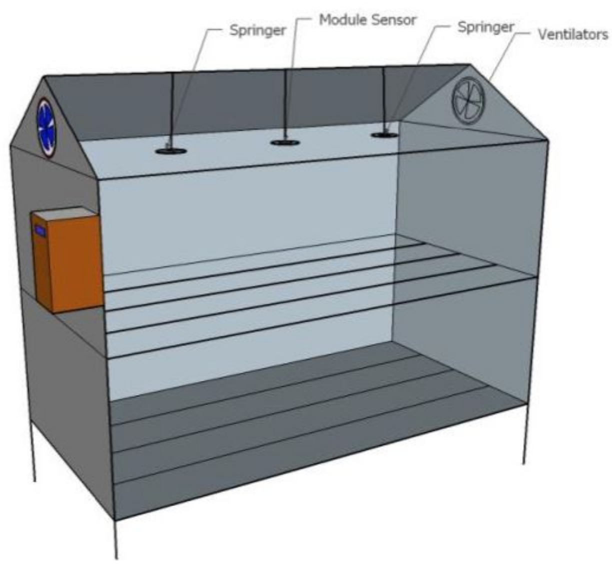

Figure 3.5 The RUSNC mushroom cottage farm model

Figure 3.5 shows the prototype model designed by RUSNC. In this model, the springers, module sensors, and ventilators are placed on top of the model. This system is used and adapted into a new model used in this experiment. The dimension of this model size dimensions is 1.70 * 1.50 * 1.20 m. The expectation capacity of the model is approximately 40 bags of mushrooms.

3.3 Hardware Design

Table 3.1 List of devices

No.	Device Name	Amount
1	Raspberry Pi 3 Model B+	1
2	AM2320 (Temperature and Humidity sensor)	4
3	DS18B20(Temperature sensor)	1
4	Relay 5 volts (4 channels)	1
5	Water springer	4
6	Brushless 12V DC Pump Model: QR30E	4
7	Ventilators	2
8	TCA9548A(I2C multiplexer)	1

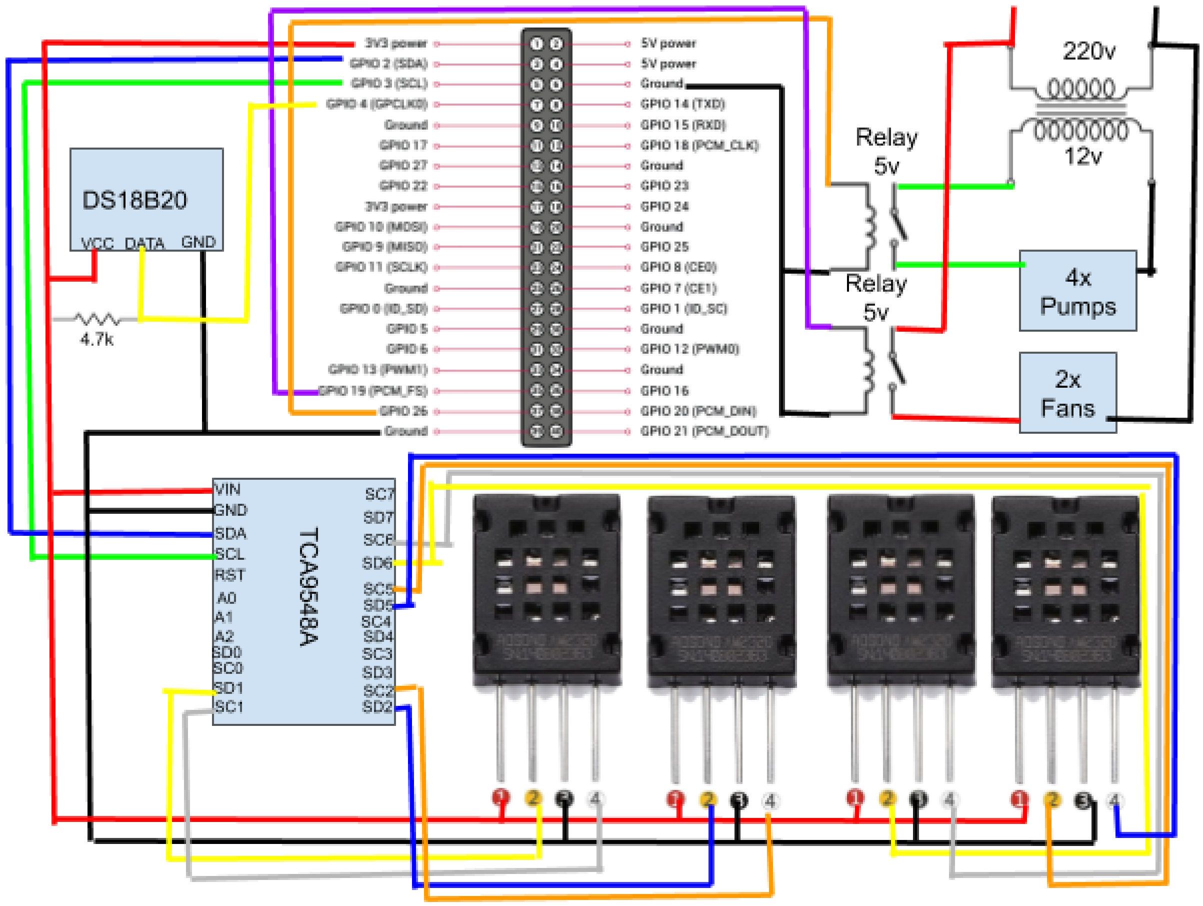

Figure 3.6 Design of the circuit

The design circuit of the model used in this experiment is illustrated in figure 3.6. Two relays are added to the circuit to control the ventilators, 4 sensors are connected to the TCA9548A Multiplexer which is connected to the Raspberry Pi.

Table 3.2 Process control system

Temperature (°C)	Humidity(%)	Ventilators	Pumps
Less than 28	Less than 80	Not working	Working
Less than 28	More than 80	Working	Not working
More than 28	Less than 80	Working	Working
More than 28	More than 80	Working	Not working

After consulting with the local mushroom farmers, the most suitable conditions to grow the mushroom are concluded to create table 3.2, where the algorithm used to control the ventilators and pumps is listed. The ideal temperature for the mushrooms is at 28°C and the ideal humidity is at 80%. If the temperature and humidity are lesser than the ideal threshold the ventilator will not work but the pumps will work to increase humidity. If the temperature is below 28°C and the humidity is too high, then only the ventilators will work. Moreover, when the temperature is higher than 28°C and the humidity is lesser than 80% both the ventilators and the pumps will work to adjust the environment accordingly. Lastly, if both the temperature and humidity are higher than the threshold then only the ventilator will work.

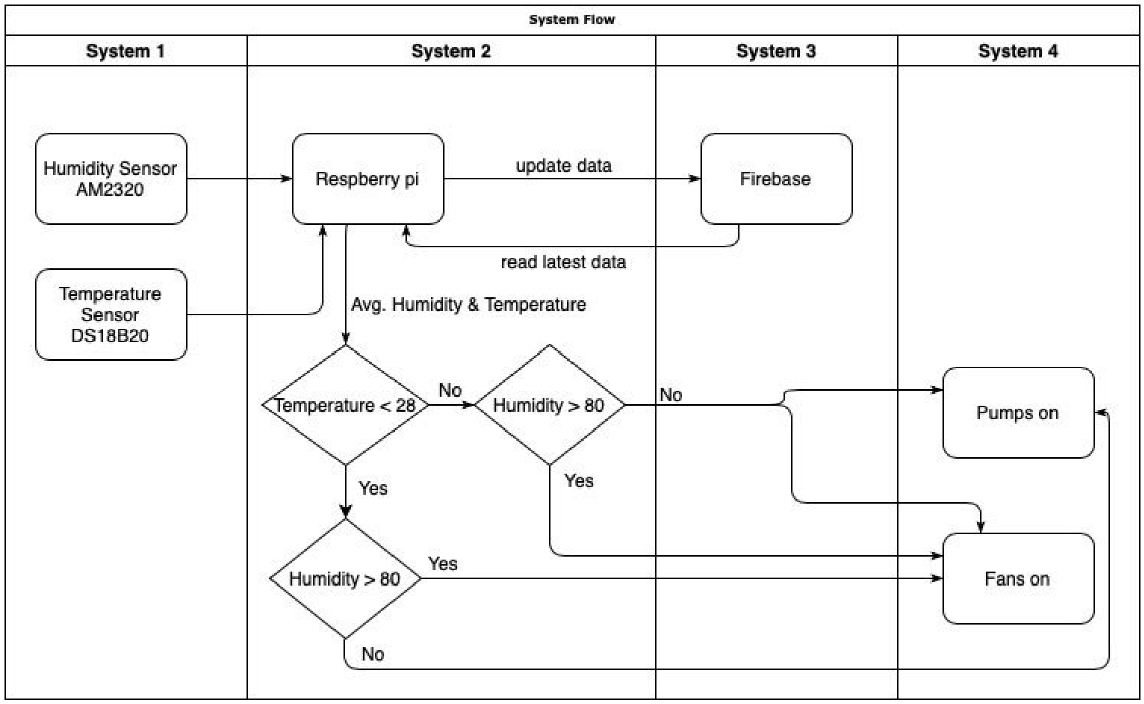

Figure 3.7 System flow diagram

Figure 3.7 explains the system flow of this project. This system is derived from table 3.2 and works according to the threshold set in table 3.2.

Figure 3.8 Picture of temperature displayed on thermometer

The additional thermometer is also connected to the system and its display is placed outside the system as shown in figure 3.8. This allows the team to be able to check the accuracy of the other sensors inside the system. It also helps us to verify the correctness of sensor information stored in the database.

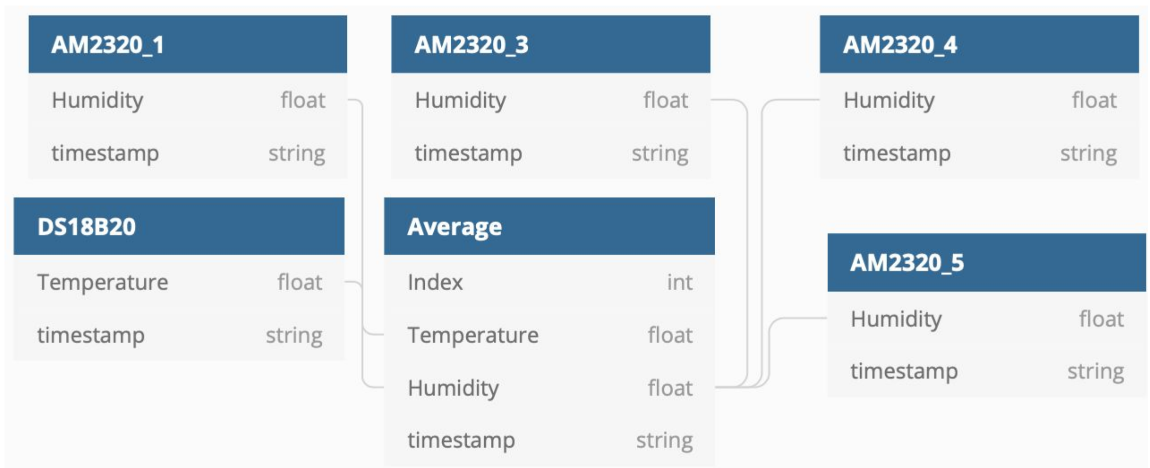

Figure 3.9 Data structure of the system

Figure 3.9 shows the data types that are sent out and stored from each sensor. There are five sensors in total where four are humidity sensors (AM2320) and one is the temperature sensor (DS18B20). All of these sensors send values to the firebase with a timestamp. Then, all humidity values are used to calculate the average value and save to the average branch which stores the average temperature and humidity from all the AM2320 and DS18B20 in the system. This average branch is used to plot the graph in the website to illustrate the trend to the user. When the average humidity value and temperature value are saved to the average branch, the index will be incremented. This index is used in the mean calculation, if there is no value in this branch, the index value is set to be equal to one so it does not affect the mean value. However, if there are already some values in this branch, the index will continue from the latest value.

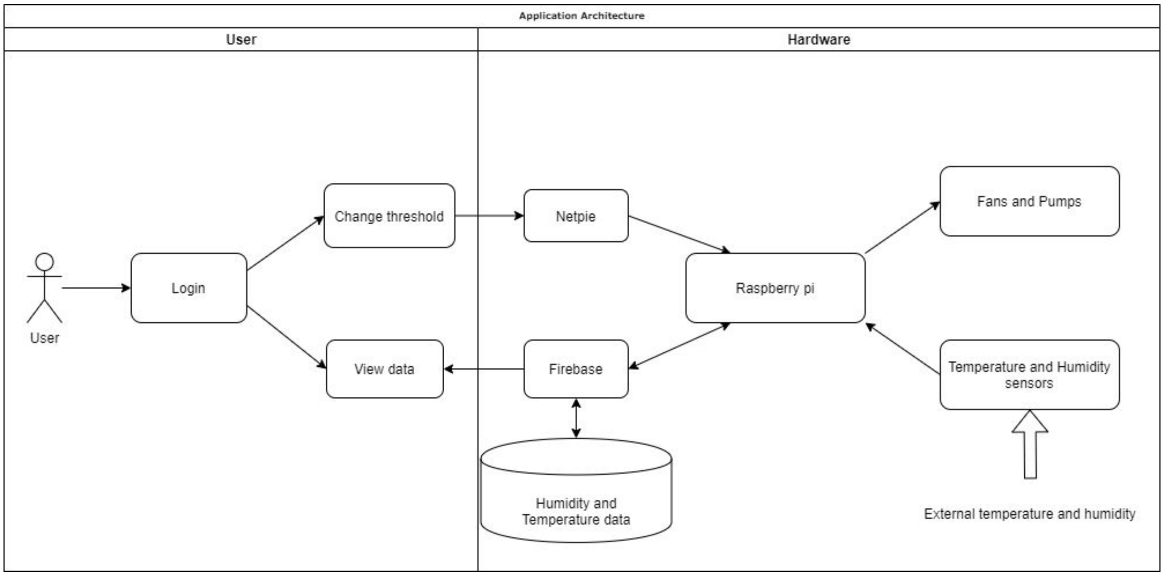

Figure 3.10 Application architecture of the system

Figure 3.10 shows the application architecture of the system from the users and the hardware sides. The user is required to go through the Login session to be able to change the threshold value or view the data. If the user changes the threshold, the new value will be sent to Raspberry Pi through Netpie. If the user chooses to view the data, the data is retrieved from firebase to be plotted and shown for the users. The diagram also shows the input from the temperature and humidity sensors values to Raspberry Pi and Output to the ventilators and pumps.

Figure 3.11 The interior design of the model

Figure 3.11 shows the interior view of the model where mushroom bags are stacked on top of each other. The sensors inside of the model are wired to the Raspberry Pi which is placed on the board outside of the model, as shown in figure 3.12.

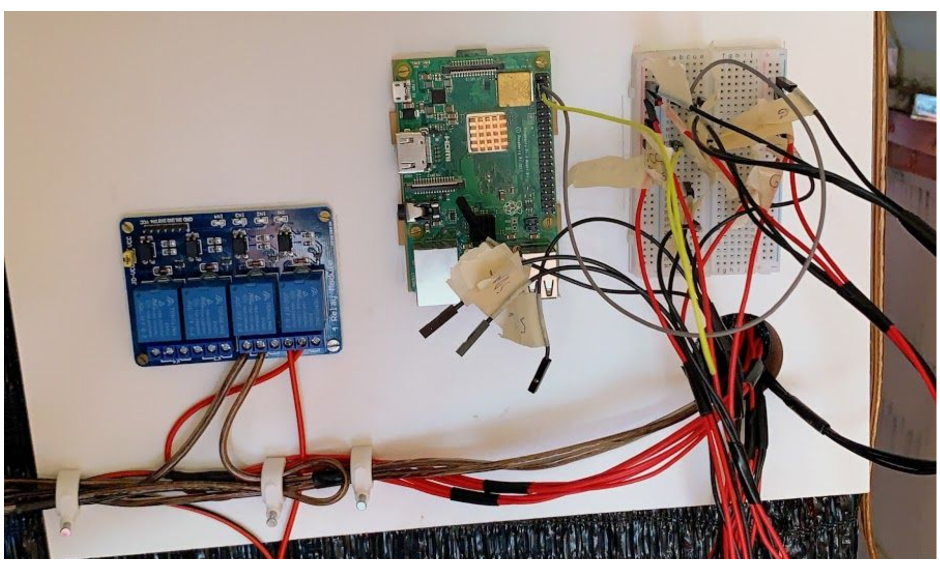

Figure 3.12 The circuit outside of the model

3.3.1 Obstacles

Previously, the bus' addresses of all the sensors were allocated using I2C address adjustment under the I2C library. However, the latest version of Raspbian OS causes the additional I2C buses to be no longer operational. This is fixed by using a new module ofI2C multiplexer, which eases the process of I2C bus' address allocation. Consequently, the implementation of several libraries in the program is no longer needed, since all the functions(required by temperature, humidity, and multiplexer) are included in the AdaFruit library alone. The image of the I2C multiplexer chip is shown in figure 3.12.

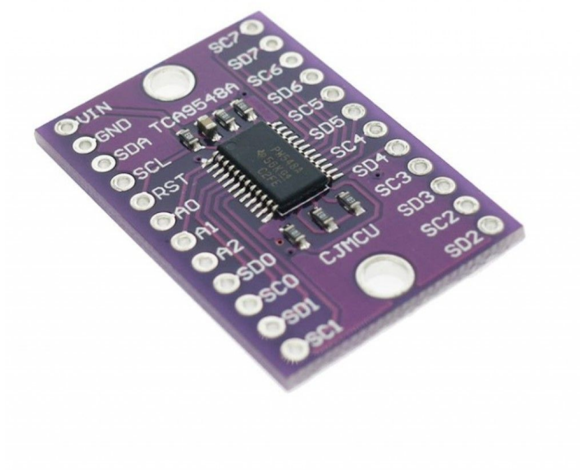

Figure 3.12 The I2C multiplexer chip used to substitute the I2C allocation.

3.4 Website Design

Bootstrap, which is an open-source framework, is used to help the design of the home page. While HTML and PHP are used to create a secured and protected login session. Additionally, the website is connected to Raspberry Pi through Netpie, which is a Network Platform for the Internet of Everything. The advantage of using Netpie platform is that it is a free service, highly scalable and commercial ready

Graph is then plotted using CanvasJs while the data is being pulled out of the firebase in real-time to be plotted. The user can zoom and view the graph in a specific range from the big graph, or go into the 'daily chart' tab to choose a specific date to view the data.

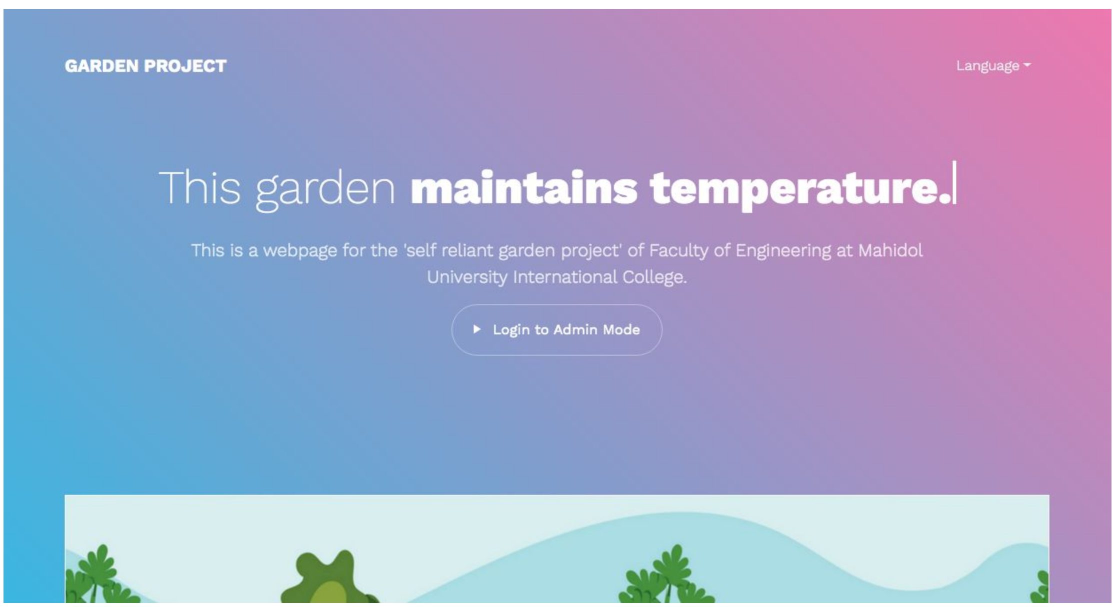

Figure 3.13 the homepage of the website

Figure 3.13 shows the design of the homepage using tools like Bootstrap and CSS in HTML. If the user clicks on 'Login to Admin Mode' and the correct username and password are authenticated. The user is redirected to the overview page as shown in figure

3.14. This page will allow the user to set a new threshold value which will be sent to RaspberryPi through Netpie as explained in figure 3.10.

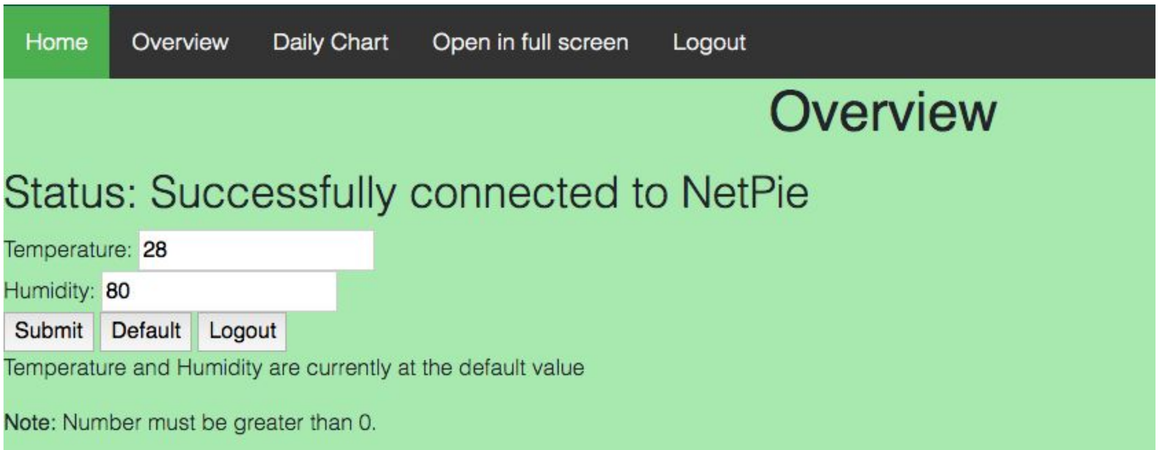

Figure 3.14 The overview page where the user can send new threshold to Raspberry Pi

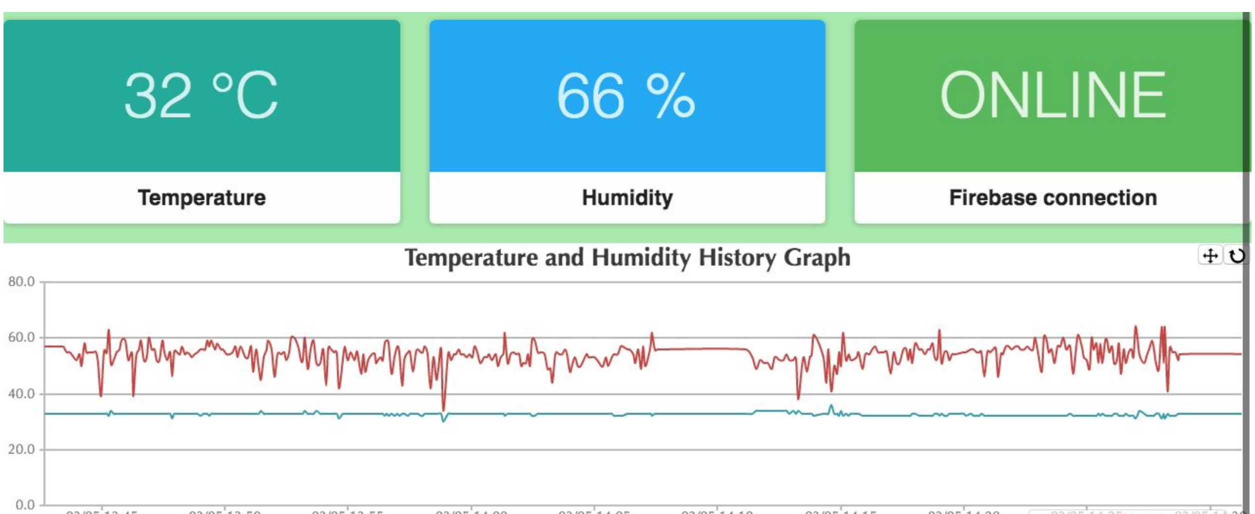

Figure 3.15 The latest data entry from firebase and overall graph in overview page

Figure 3.15 shows an example of the humidity and temperature graphs where the data is retrieved from the 'average node' in firebase. The temperature and humidity boxes on top of the page show the data pulled from the latest node in firebase.

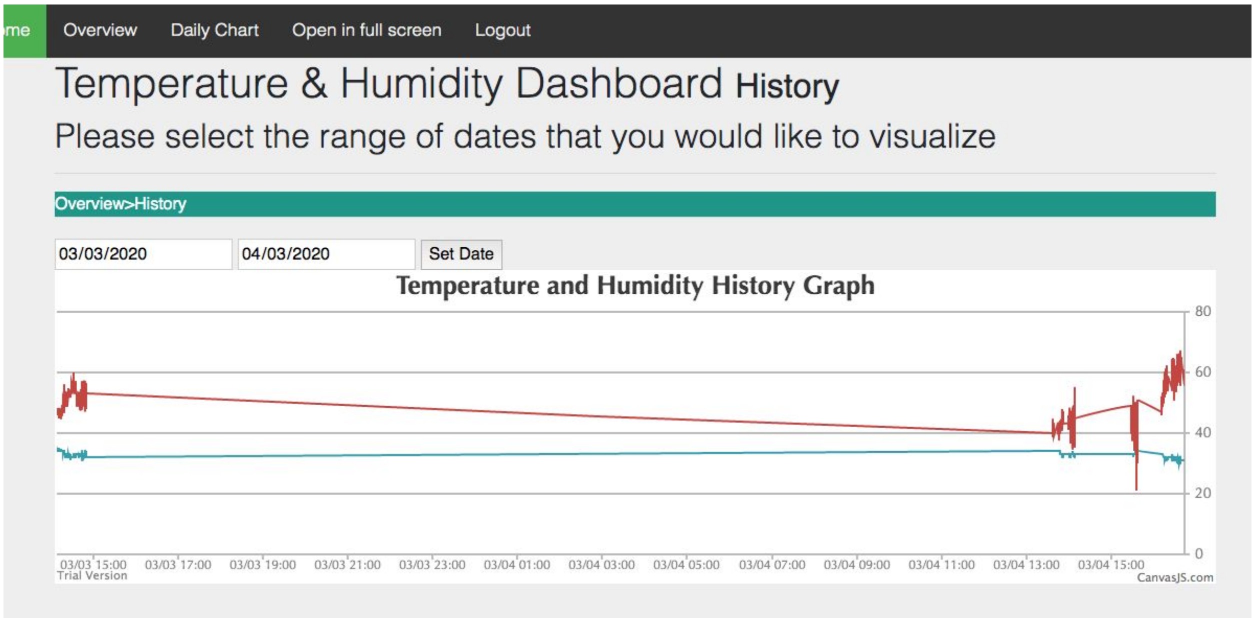

Figure 3.16 the Daily Chart page of selected dates interval on the graph

If the user would like to view the data from a specific interval of dates, the user can do so in the 'Daily Chart' tab as shown in figure 3.16. In this particular figure, the graph is drawn from the date of 03/03/2020 to 04/03/2020 because that is what the user has selected.

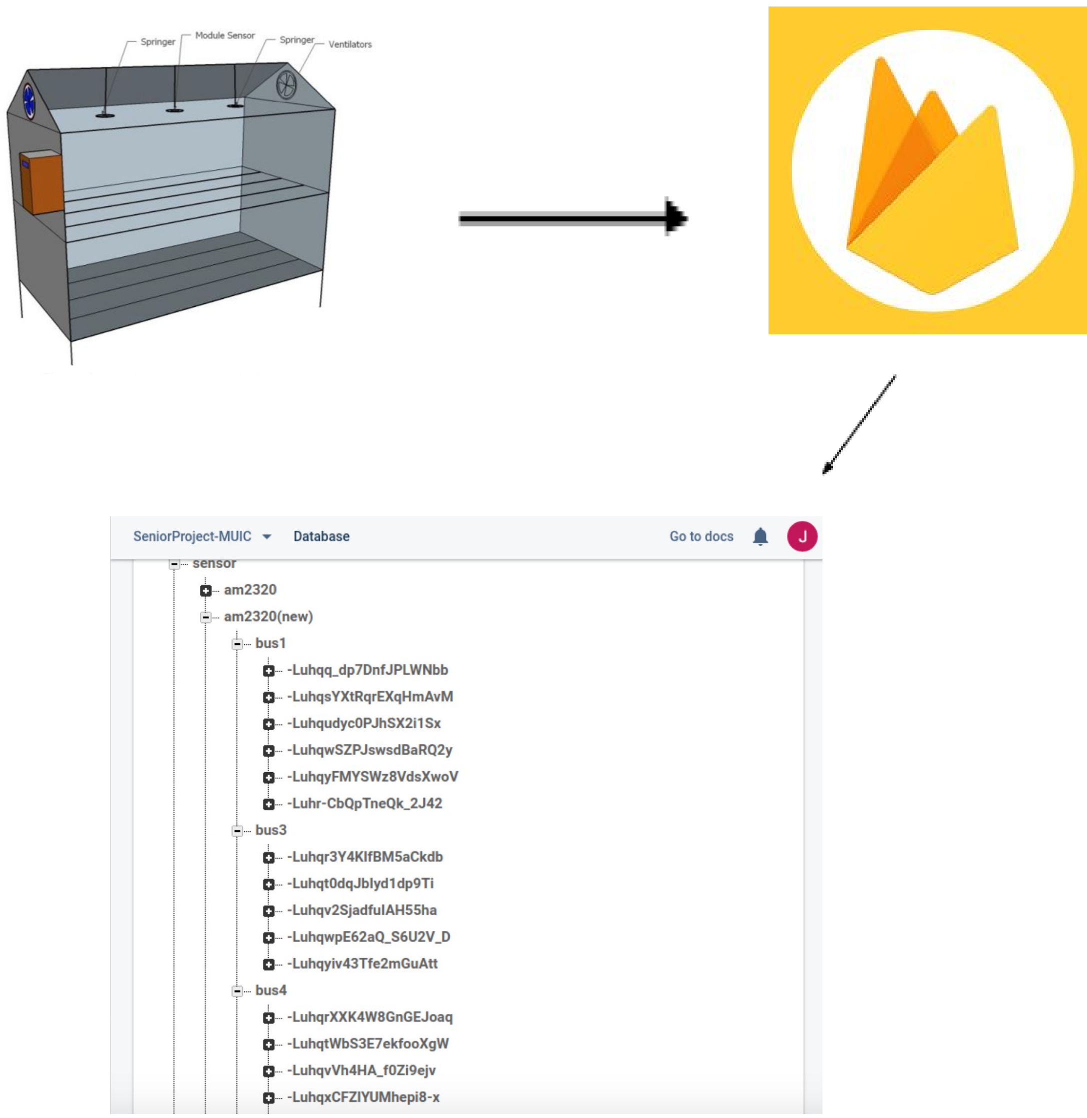

Figure 3.17 The overview structure of the system

Figure 3.17 illustrates that all the data from the sensors are sent out from the raspberry pi to the firebase and the sensor inputs are stored in branches that have several buses as child nodes.

CHAPTER IV

EXPERIMENTAL DESIGN AND RESULTS

4.1 The criteria used for determining the experiment:

1) Weight of the mushroom bags
2) Moisture (80 for the controlled group)
3) Temperature (under 30 °C for the controlled group)

4.2 Record:

Our assumption is that mushrooms, planted under our environment, can grow more effectively when compared to the typical method. Thus, for the experiment, the weight of mushrooms bags is recorded daily before watering. Due to the various weights of mushroom bags, three simple mushroom bags are collected from each group and calculated the weight difference in percentage.

Table 4.1: For the first data set, used the sample of 7 days expected growth

Days	Normal group (Weight in grams)			Control group (Weight in grams)		
	754	852	746	806	783	791
15/01/2020	752	840	772	823	771	794
16/01/2020	751	861	732	814	750	743
17/01/2020	762	868	762	823	772	748
18/01/2020	778	865	801	882	789	768
19/01/2020	792	866	823	895	788	779
20/01/2020	802	-	821	899	-	778
21/01/2020	805	-	-	-	-	-

Table 4.1 shows one of the data sets recorded during the experiment. The weight of the mushroom bags was daily measured in order to find the rate of growth. If the growth rate of the mushrooms stopped then the measurement will be stopped as well, resulting in the dash lines as shown in Table 4.1.

Figure 4.1 The mushrooms harvested from the experiment

The average weight of the mushroom from uncontrolled group and control group is 60.8/ 59.7 (in grams) respectively

I. Controlled group 82.3 SD 1.8

II. Uncontrolled group 81.2 SD 3.5

From the result above, the SD value of the uncontrolled group is almost doubled from the one from the controlled group. Additionally, the average weight of the outcome is also lower than the controlled group. This indicates that the yield from the system has a higher quality and easier to control.

4.3 Software result:

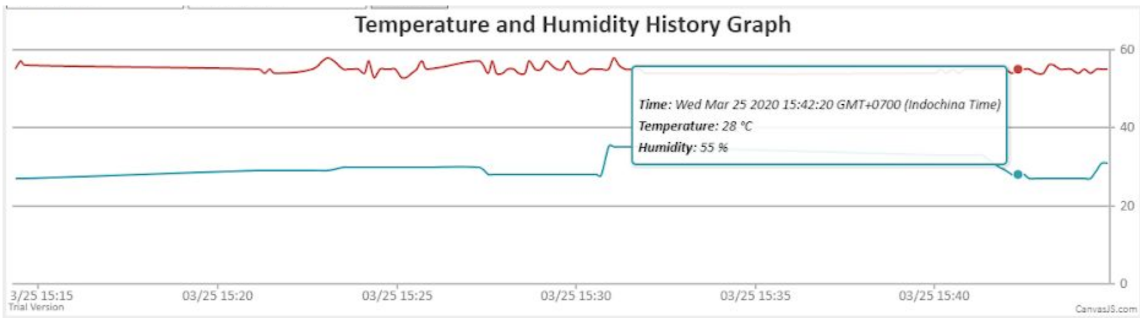

Figure 4.2 The graph displayed rate of changes for humidity and temperature

Figure 4.2 shows the interval of changing in temperature and humidity in every 5 mins. It shows that the temperature has decreased significantly within 2-3 minutes, which is considered as a satisfactory result. For the humidity, it did not increase at the same rate of temperature. The system can slowly increase the humidity to meet the threshold as what it was set by the system.

4.4 Hardware result:

Table 4.2: The comparison of parameters

Time (In minutes)	Internal Temp(°C)/Humidity (%)	External Temp(°C)/Humidity (%)
0	33 °C/ 40%	33 °C/ 40%
2	29 °C / 50%	33 °C/ 40%
5	28* °C/ 60%	33 °C/ 40%
10	28 °C/ 80%	33 °C/ 40%

***Appropriate value**

CHAPTER V

EXPERIMENT DISCUSSION

5.1 Discussion

The experimental results from table 4.1 is used to demonstrate the approximated results of the mushroom grow under controlled and uncontrolled environments. All the factors mentioned earlier, humidity and temperature are used to differentiate both groups for consideration. After 3 batches have been measured to find the average values and calculate for the standard deviation. All the results are used to calculate the mean and standard deviation values. The result suggested that the SD value of the uncontrolled group is almost doubled from the one from the controlled group. This indicates that the yield from the system has a higher quality and easier to control.

5.2 Evaluation

There are various factors that play an important part in the yielding of the mushrooms. This specific species of mushroom requires the control of the humidity to be at 80% and temperature of around 27 °C. The complete system achieves that based on a few adjustments on the system, like a change in sensor type for instance. The slight alterations in the model are needed, throughout the experiment period. For example, an extra layer of the canvas was added on top of the previous layer in order for the internal temperature to be more stable. There are also hardware-related problems, like loose wire and input errors from sensors.

During the experimental period, the model was also required to move to a new location. Another factor that points out is the air quality, which affected the quality of the mushroom significantly. When the model is moved to the location where all the air has poor quality and dust, the quality of the mushrooms lessens as shown in figure 5.2.

When mushrooms are constantly exposed to dust, the weight slightly drops and the appearance is not as clear. There will be noticeable white mold forming in the middle of the mushroom. When compared to mushrooms from the dust-free environment in figure 5.1, which is so much clearer in appearances.

Figure 5.1 Dust-free environment

Figure 5.2 Dusty environment

In terms of commercials, the white mold mushroom will be considered as a low-quality product. The farmer cannot sell them to the shop and usually throw them away. Therefore, this factor is needed to be seriously considered if the project will be put into real use for commercial purposes.

Table 5.1: Comparison between the automated system and traditional growing method

	Automated system	**Traditional**
Temperature	Always be controlled under 28°C	Depends on the environment
Humidity	Always be controlled above 80%	Daily watering
Yield	Average weight of 60.8 grams	Average weight of 59.7 grams
Monitoring	Via website	In-person

5.3 Limitations

I. All electronic devices must be carefully implemented to ensure that humidity and water will not affect the performance of the system.

II. Temperature and humidity factors must be manually set.

III. Despite the fact that the system can constantly maintain its stability, the users must constantly check the yields of the system.

IV. The current design only supports one specific kind of mushrooms at a time.

V. The wires in this system are soldered together to prevent loosening, therefore, if a sensor is broken then all the wires will have to be changed.

CHAPTER VI
CONCLUSION

This project encourages us to integrate the knowledge gained from various classes. This includes everything from engineering electronics, embedded systems, microprocessors and controllers, web programming and report writing classes. Additional knowledge is also obtained from researching from several sources including surveying at an actual farm and garden. The project has made progress based on trials and errors throughout the project. The models in the project are constantly modified and changed. At first, the project was designed to create a self-reliant system with image processing that is designed to look after plants like tomato and chili. Then the idea shifts from that to this current project. The decision to eliminate the image processing system because it is unnecessary and would take a longer time to develop. The focus of the project also shifts to plant gardening and mushroom farming specifically. The suggestions from advisors who are extremely experienced in everything were very helpful, as well as being able to receive advice and guidance from a real-life mushroom farmer throughout the project. It is said that this project would be a tremendous help to an actual farmer if they decide to implement the system inside their farms. This is due to the fact that it can make everything so much easier to be controlled and do not require a human to go on the farm and interfere with the controlled factors. Additionally, a closed system garden would be much more suitable for the mushrooms to grow at a constant rate. Hopefully, the project can further be implemented for real use, and improve the quality and quantity of agricultural yield of Thailand overall.

REFERENCES

1. Thailand GDP share of agriculture [Internet]. TheGlobalEconomy. 2019 [cited 16 March 2019]. Available from: https://www.theglobaleconomy.com/Thailand/Share_of_agriculture/
2. “Thailand GDP From Agriculture 1993-2019 Data: 2020-2022 Forecast: Historical,” *Thailand GDP From Agriculture | 1993-2019 Data | 2020-2022 Forecast | Historical*. [Online]. Available: https://tradingeconomics.com/thailand/gdp-from-agriculture. [Accessed: 07-Apr-2020].
3. Poapongsakorn N. Rapid growth of selected Asian economies [Internet]. Fao. 2006 [cited 17 March 2019]. Available from: http://www.fao.org/3/ag089e/AG089E04.htm
4. Watson, M. (2020). *What Are Oyster Mushrooms?*. [online] The Spruce Eats. Available at: https://www.thespruceeats.com/what-are-oyster-mushrooms-4172003 [Accessed 8 Mar. 2020].
5. A. Sinclair, “How To Grow Oyster Mushrooms,” *Oak Hill Gardens*, 12-Feb-2018. [Internet]. Available: https://www.oakhillgardens.com/blog/how-to-grow-oyster-mushrooms. [Accessed: 23-Dec-2019].
6. S. Dulal, “Oyster Mushroom Farming - an Overview,” Unpublished, 2019, doi: 10.13140/RG.2.2.28189.87526
7. Humidity sensor [Internet]. Electronics For You. 2018 [cited 16 March 2019]. Available from: https://electronicsforu.com/resources/electronics-components/humidity-se

nsor-basic-usage-parameter

8. Digital temperature and humidity sensor [Internet]. Aosong [image] 2014. [cited 23 march 2020] Available from: https://datasheetspdf.com/pdf/952504/Aosong/AM2320/1
9. DS18B20 Programmable resolution 1-wire digital thermometer [Internet]. Maxim Integrated. 2019 [cited 23 March 2020]. Available from: https://datasheets.maximintegrated.com/en/ds/DS18B20.pdf
10. M. Otto and J. Thornton, “Bootstrap,” · *The most popular HTML, CSS, and JS library in the world.* [Online]. Available: https://getbootstrap.com/. [Accessed: 08-Apr-2020].
11. An official guide to netpie [Internet]. Nectec. 2017. [cited 12 March 2019] Available from: http://203.159.154.241/innogoth/wp-content/uploads/2017/09/NETPIE-WS_v23.pdf
12. S. Boonyung, S. Santi, (2015) Temperature and humidity control system in Mushroom greenhouse.[online] RUSNC. Available at: http://www.rdi.rmutsb.ac.th/2011/digipro/002/proceedings/1-สาขาวิศวกรรมศาสตร์และ อุตสาหกรรมวิจัย/24.59111067-176-183.pdf [Accessed: 28-Sep-2019]

APPENDICES

APPENDIX A

Raspberry Pi setup

A.1 Download softwares

The user must download Raspbian Buster for desktop at https://www.raspberrypi.org/downloads/raspbian/ and download BalenaEtcher software to flash the Raspbian to micro SD card at https://www.balena.io/etcher/

A.2 How to flash the raspbian and power up the raspberry pi

On the left panel, the user has to select the zip file that was downloaded. On the center, the user has to select the micro SD card and then click flash. After flashing is finished, the user inserts the micro SD card to the raspberry pi and powers it up. After powering up the raspberry pi, installation will run automatically.

A.3 Setting up raspberry pi

Once the raspbian installation is completed, the screen from figure A1 will pop up.

Figure A1. Rasbian completed installation screen

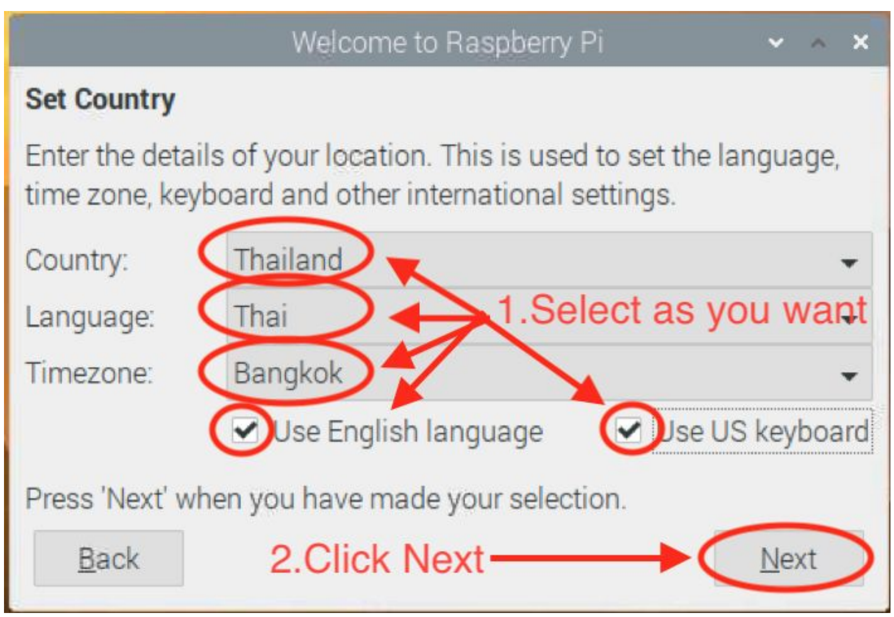

Figure A2. Rasbian initial setting screen

A.4 Setting up country and timezone

The user should set up the country, timezone, and language according to the user's location as shown in figure A2.

A.5 Setting up the username and password

The default username and password of raspberry pi is “pi” “raspberry”. The user can manually set up their own username and password as shown in figure A3. These will be used for remote accessing.

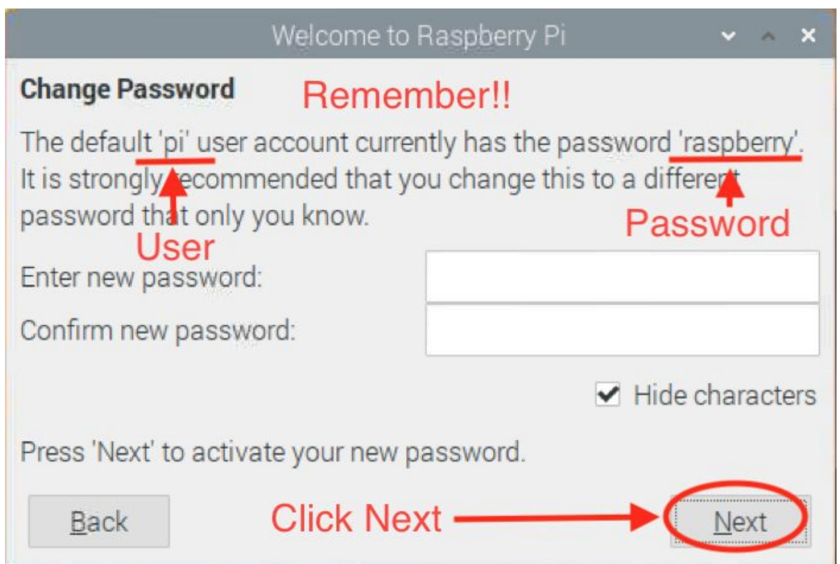

Figure A3. Raspberry username/ password

A.6 Setting up the screen

Screen setup for the black border is needed. The user should check in the box, which is shown in figure A4. This will make the remote accessing from other computers more visible due to adjusting the width and height of the raspberry pi screen.

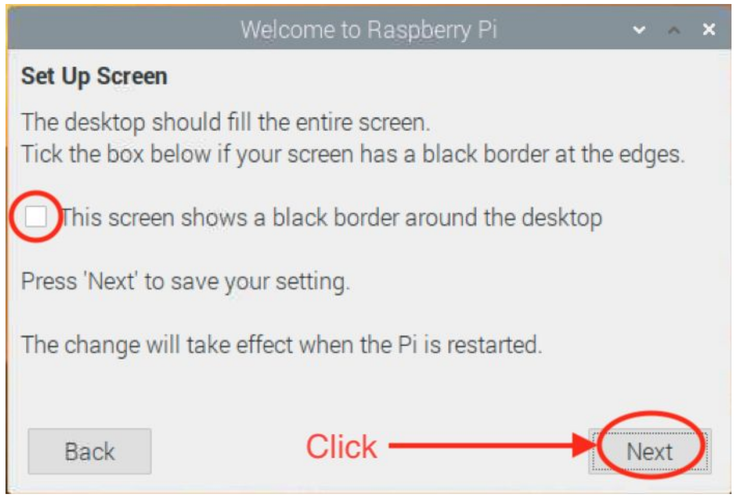

Figure A4.The setup screen to enabling your screen to display black border

A.7 Setting up network

From figure A5, the user can select the wifi network according to user preference. This network should be the same for raspberry pi and computer, used for remote control.

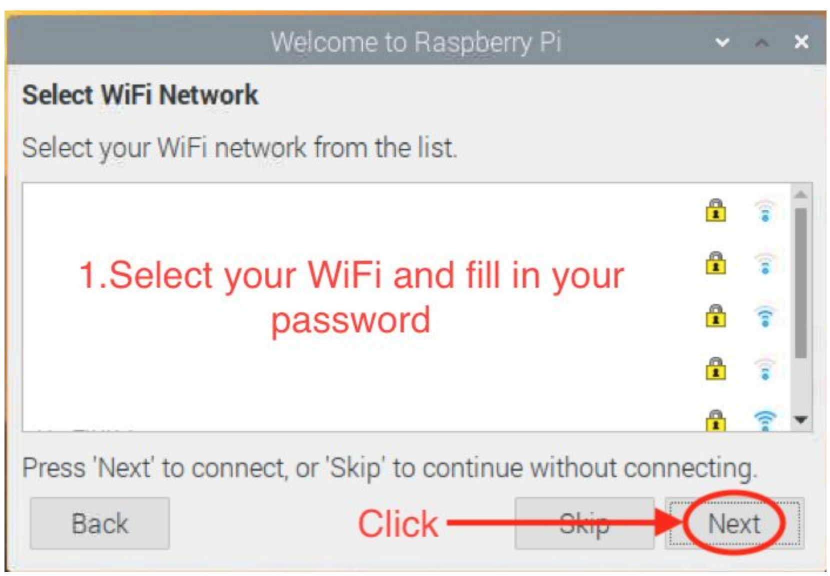

Figure A5. Wifi setting

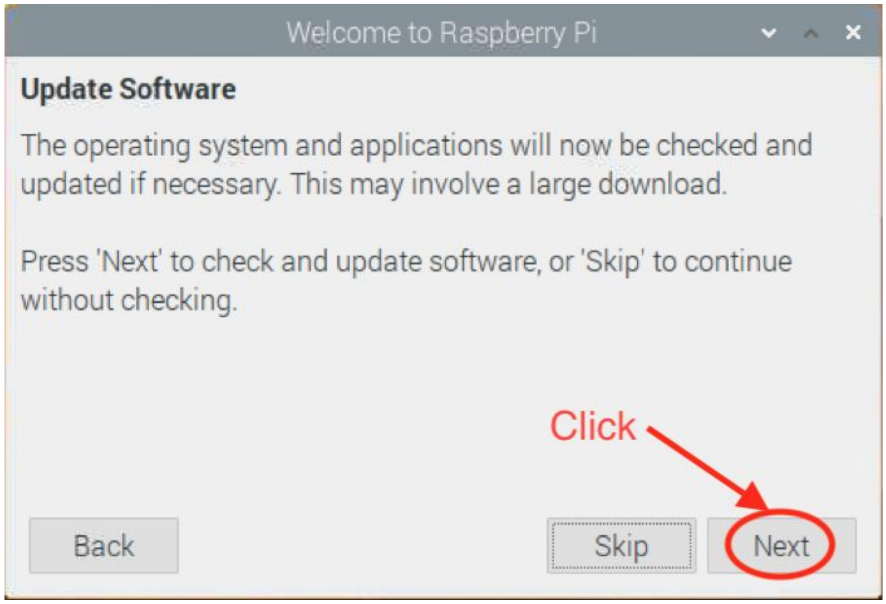

Figure A6. Updating raspbian's software screen

A.8 Checking for update

After wifi network selection is completed, the user should update the raspbian's software to the latest version, shown in figure A6. This step will take a few minutes to complete. Once the updating is finished. Raspberry pi will pop up the screen asking for the restart as shown in figure A7. The user should click on the restart button to finish the updating process.

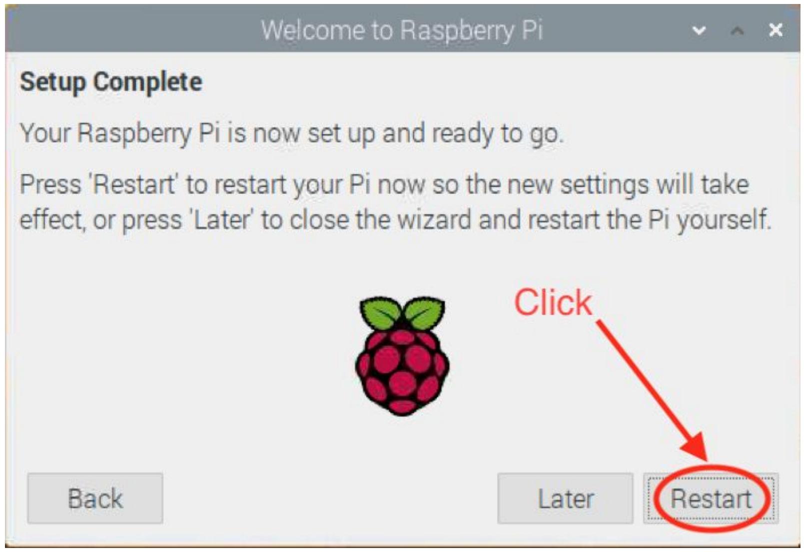

Figure A7. Restart permission screen after completed the update

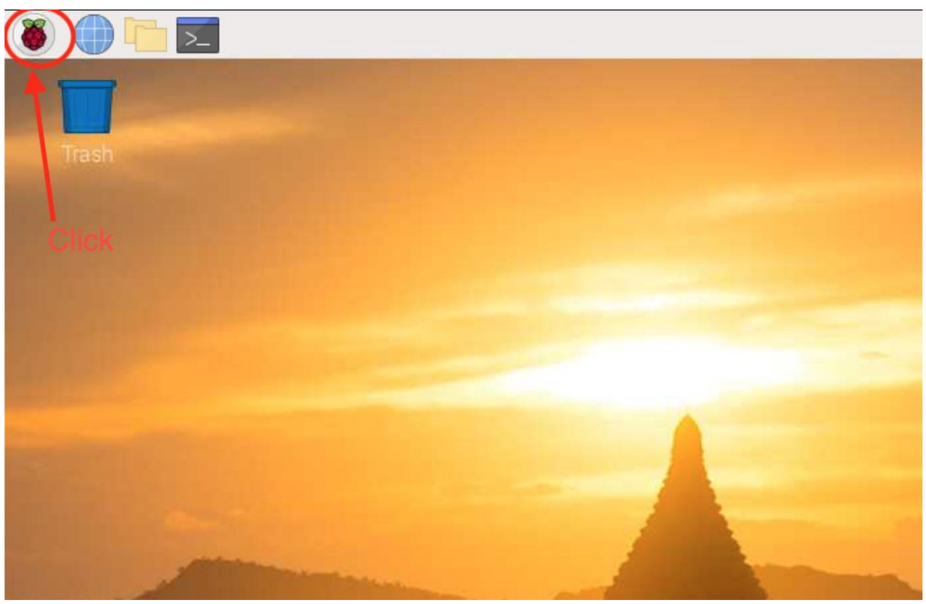

Figure A8. Raspberry start icon

A.9 Raspberry pi configuration

After the Raspberry pi finished reboot, the user has to click on the Raspberry start icon, which is located on the top left of the screen. The user selects "Raspberry Pi Configuration", which is located in the Preferences tab, shown in figure A8 and A9.

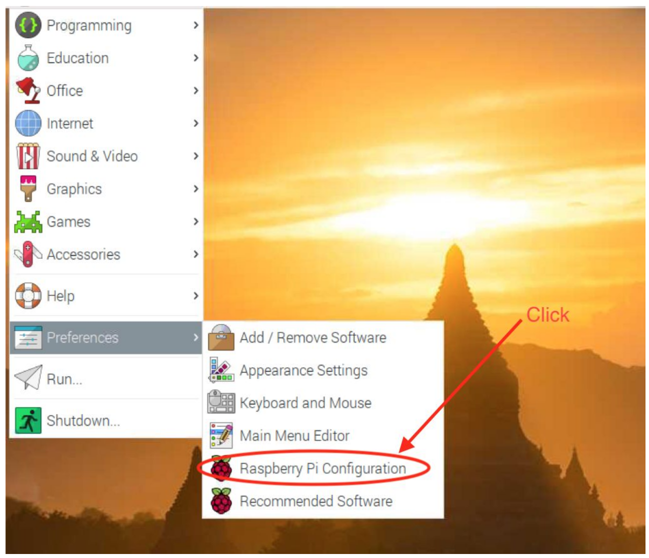

Figure A9. Raspberry Configuration

A.10 Necessary interfaces

After the user selects "Raspberry Pi Configuration", the interface tap will appear. The user must check all the boxes, which is shown in figure A10. After clicking the ok button, Raspberry Pi will ask the user for permission to reboot the system, shown in figure A11.

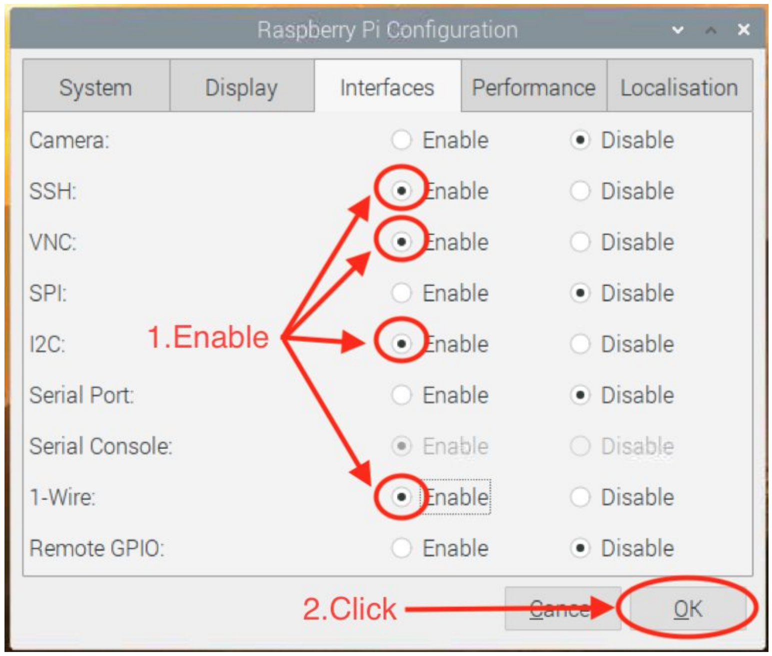

Figure A10. Raspberry's ports configuration

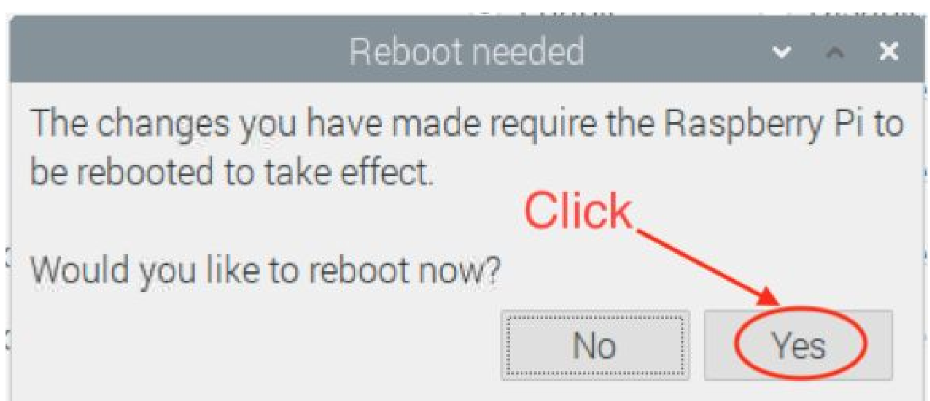

Figure A11. The pop-up screen after ports configuration

A.11 Installing necessary libraries

After the raspberry pi is rebooted, the following commands must be typed to install the drivers for sensors.

```
sudo apt-get update

sudo apt-get upgrade

sudo apt-get dist-upgrade

sudo apt-get install i2c-tools

sudo apt-get install python3-smbus

sudo pip3 install —upgrade setuptools

sudo pip3 install RPI.GPIO

sudo pip3 install adafruit-blinka

sudo pip3 install adafruit-io

sudo pip3 install adafruit-circuitpython-tca9548a

sudo pip3 install adafruit-circuitpython-am2320

sudo pip3 install w1thermsensor

sudo pip3 install requests==2.2.1

sudo pip3 install python-firebase

sudo pip3 install microgear
```

A.12 Changing symbols in python files

Due to the version of python, some symbols in the files need to be changed.

cd /usr/local/lib/python3.7/dist-packages/firebase/

sudo mv async.py async_.py

sudo nano __init__.py —> .async —> .async_ —> Ctrl X -> y -> enter

sudo nano firebase.py —> .async —> .async_ —> Ctrl X -> y -> enter

sudo reboot

After all the steps are completed, Raspberry Pi is ready to install the file to operate the system.

APPENDIX B

FIREBASE SETUP

B.1 Adding new Firebase project

Firstly, the user navigates the browser to console.firebase.google.com and clicks "Add project" like figure B1.

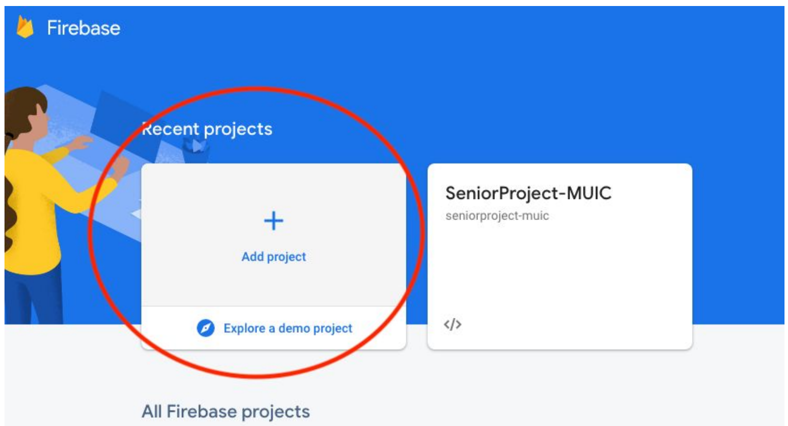

Figure B1. Add new project on Firebase

B.2 Naming the project

When a new project is added to Firebase, the user is redirected to rename a project which is shown in figure B2.

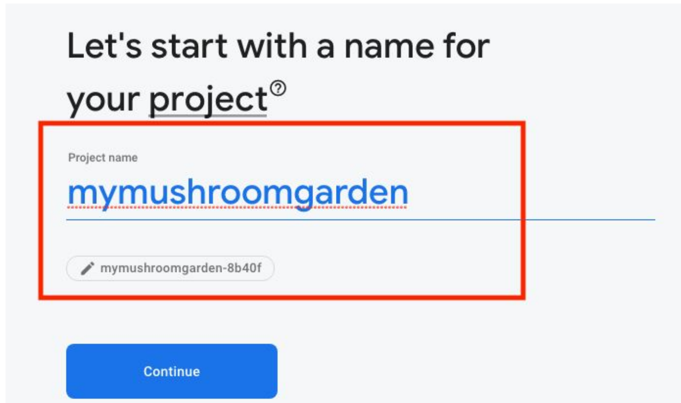

Figure B2. Name the project on Firebase

B.3 Creating new Firebase project

After naming the project, the user is redirected to Google Analytics page. It is highly recommended that the user turn it off because it is unnecessary. After the option is toggled then, the user clicks the 'Create Project' button as shown in figure B3.

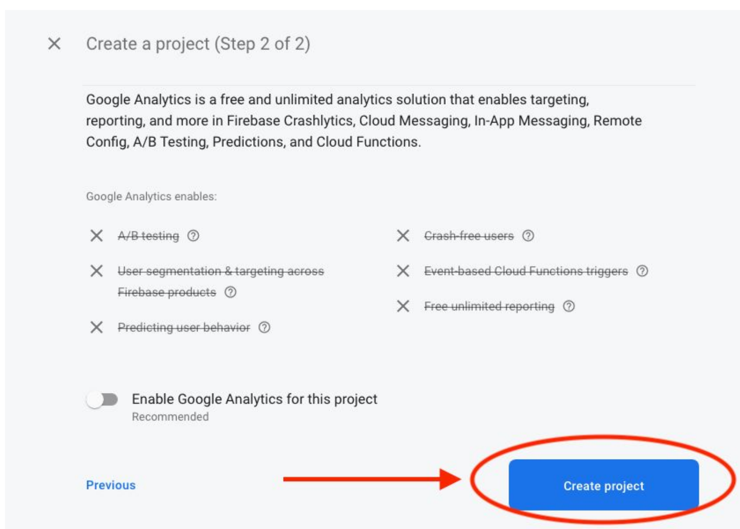

Figure B3. Create new project on Firebase

B.4 Navigating to the Database

The user can find the database page by navigating from the left panel as shown in figure B4.

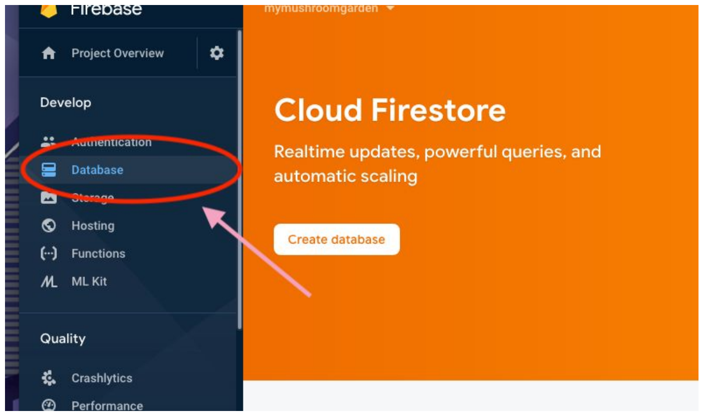

Figure B4. Navigation to database

B.5 Creating realtime database

Next step is creating a realtime Database. This type of database is suitable because data is synced across all clients in real time. The data also remains available when your app goes offline. To do this, the user has to scroll down to find the 'Realtime Database' banner as shown in figure B5.

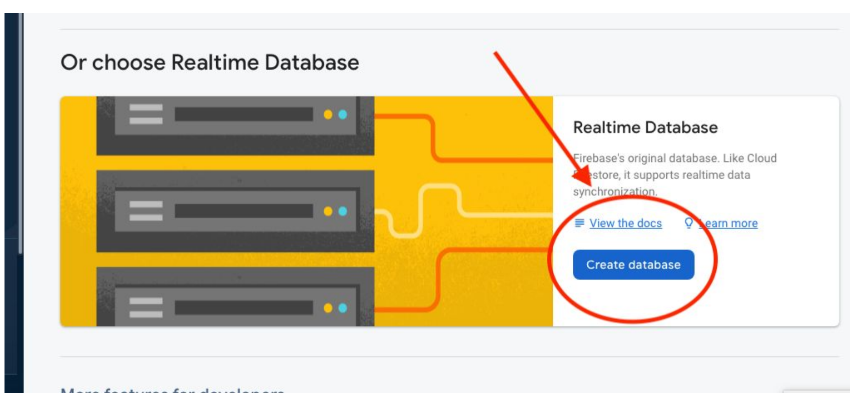

Figure B5. Create realtime database

B.6 Database security rules

Choose ‘Start in test mode’ then click enable to start the project as shown in figure B6.

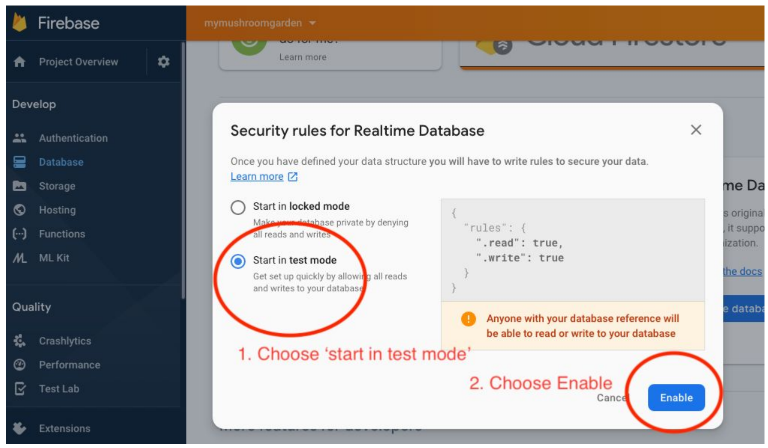

Figure B6. Adjust security rule

B.7 Project setting

Configuration for the project can be found by navigating to ‘project settings’ as shown in figure B7.

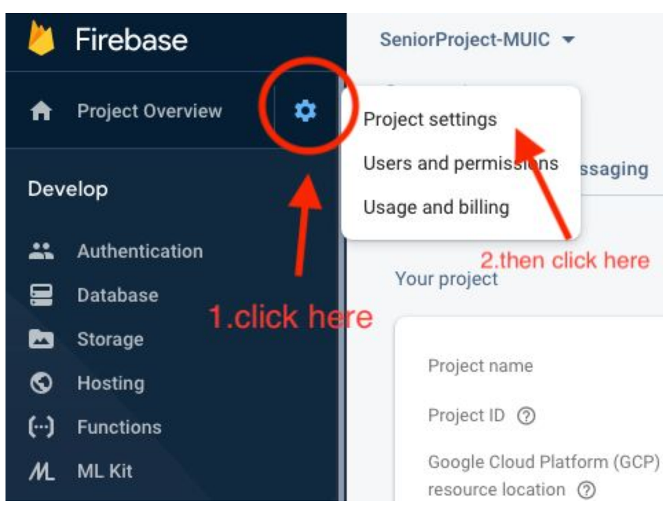

Figure B7. Navigation to project settings

B.8 Finding configuration key

The user has to scroll down to find ‘Firebase SDK snippet’ as shown in Figure B8. Then select the ‘Config’ radio button. The Firebase configuration can be found here.

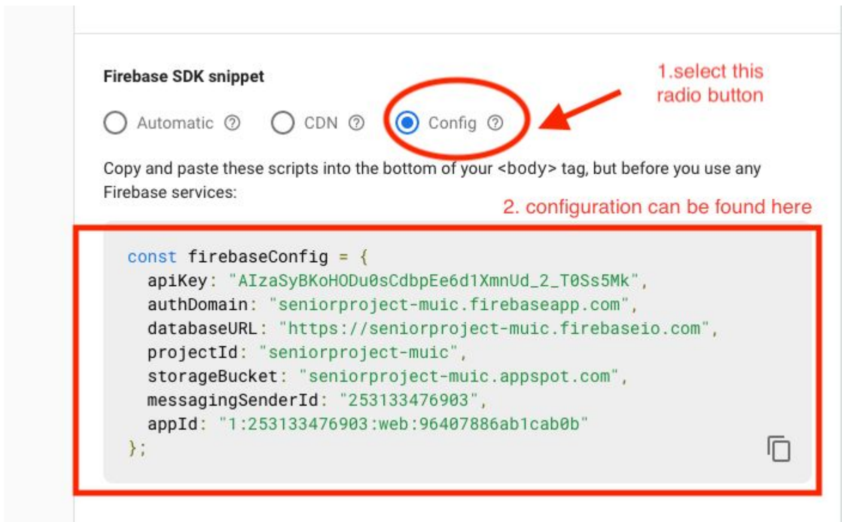

Figure B8. Firebase configuration

APPENDIX C
NETPIE SETUP

C.1 Signing up for an account

If the user already has an account, the user should click the “LOGIN” button. If the user does not have an account, the user should click at the “SIGN UP FREE” button as shown in figure C1.

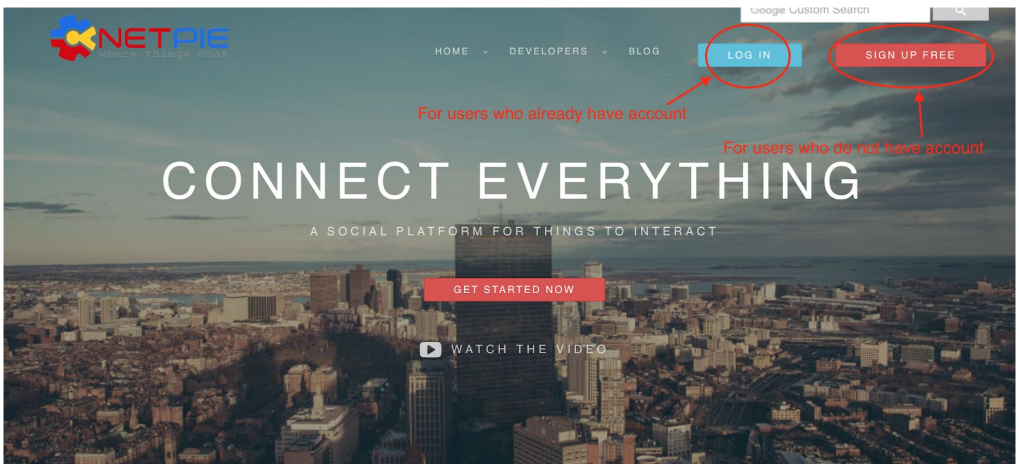

Figure C1. Creating an account on Netpie

C.2 Going to application page

After the user is logged in to the account, the user clicks on “RESOURCES” and then clicks on “APPLICATIONS” as shown in figure C2 to go to the application page.

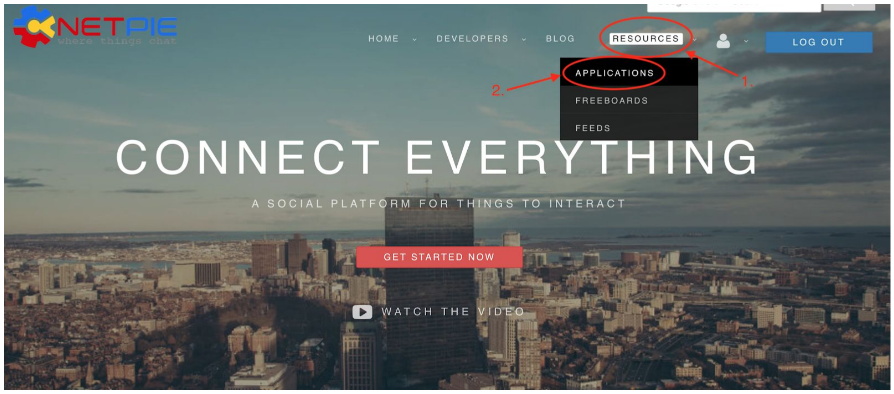

Figure C2. Going to the application page on Netpie

C.3 Creating an application ID

User clicks on the plus sign on the right side of the application box as shown in figure C3 in order to create a new application ID. After that, the box will pop up to allow the user to name a new application ID which is shown in figure C4. After the user creates the new application ID, it will appear at the application box. If it does not appear in the box, the user should refresh the page one time.

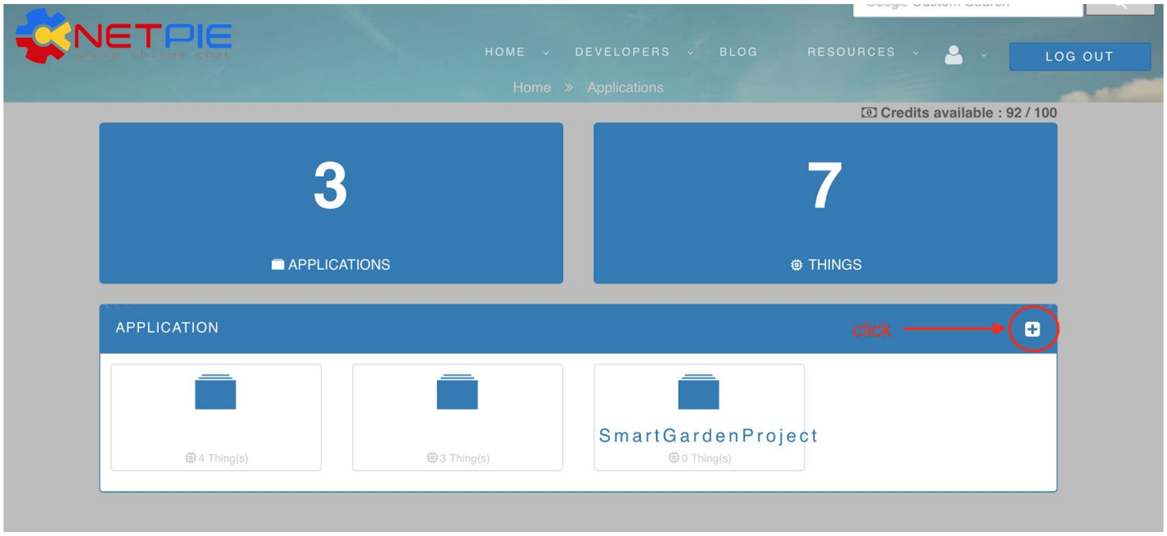

Figure C3. Creating new application ID

Figure C4. Naming an application ID on Netpie

C.4 Creating the device key

After user' application ID appears, the user has to click on the application ID in order to create the device key and the session key as shown in figure C5. In this page, the user will see the plus sign on the right side of the application key box as shown in figure C6. The user has to click that button, and then the box will appear to let the user name the key and select device key as shown in figure C7. After the user creates the device key, it will be found at the application key box as illustrated in figure C8.

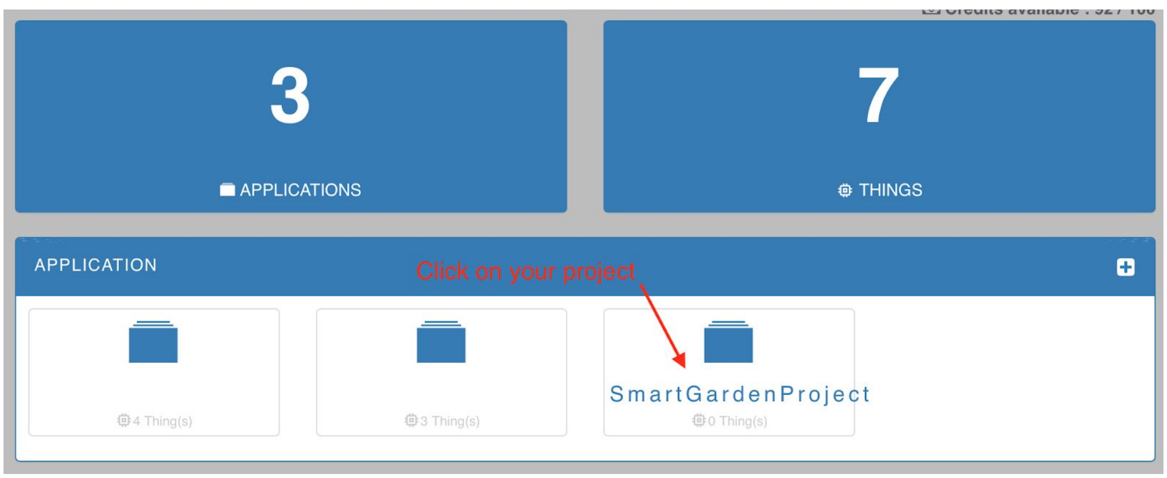

Figure C5. Going to the key page

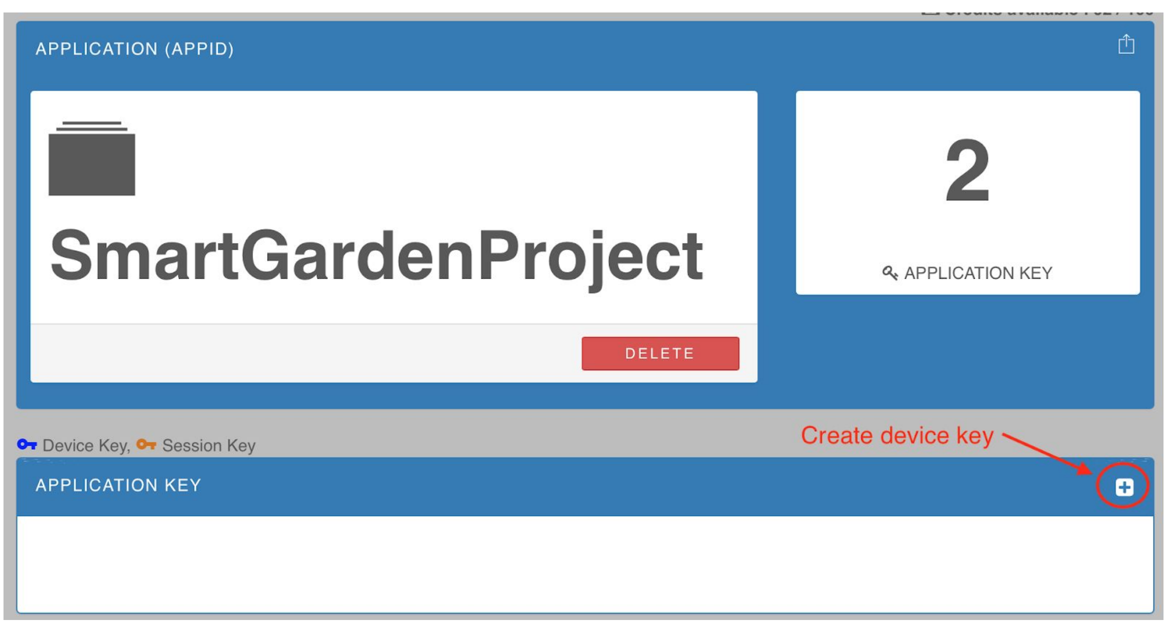

Figure C6. Creating a device key

Figure C7. Naming a device key

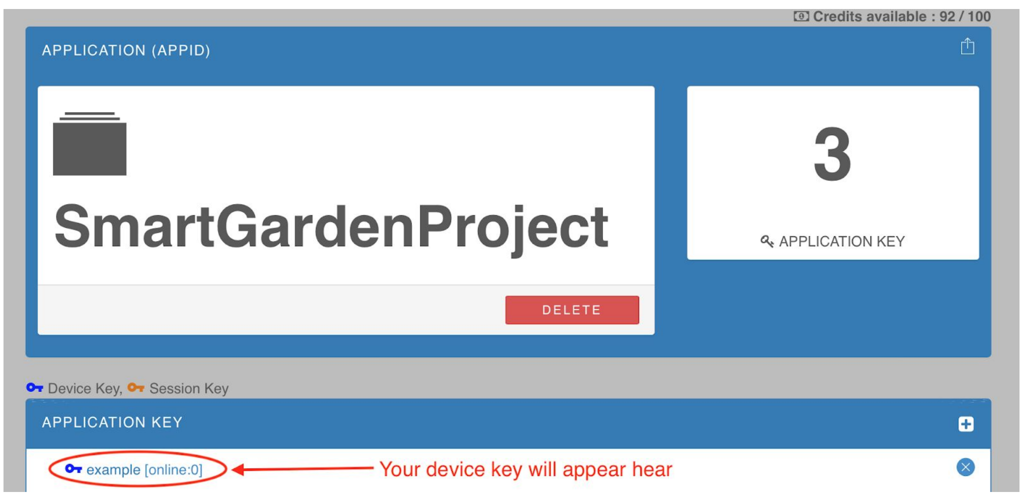

Figure C8. Location of the device key

C.5 Creating the session key

To create the session key, repeating the same procedure as process C.4 as shown in figure C9, but this time choose the session key as shown in figure C10. After the session key is created, it can be found in the in application key box below the device key as shown in figure C11.

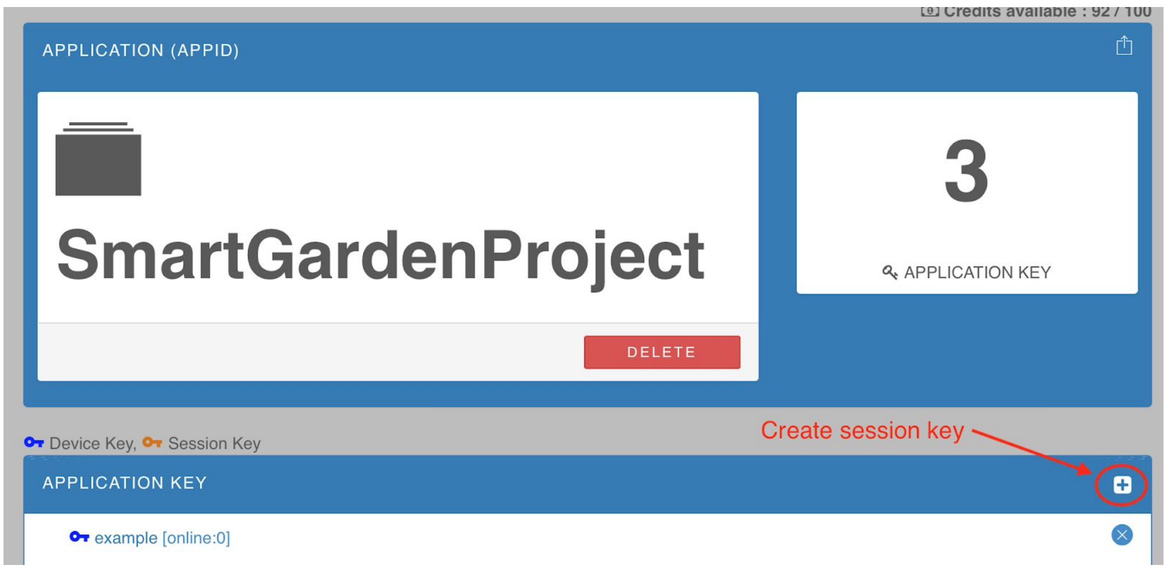

Figure C9. Creating the session key

Figure C10. Naming the session key

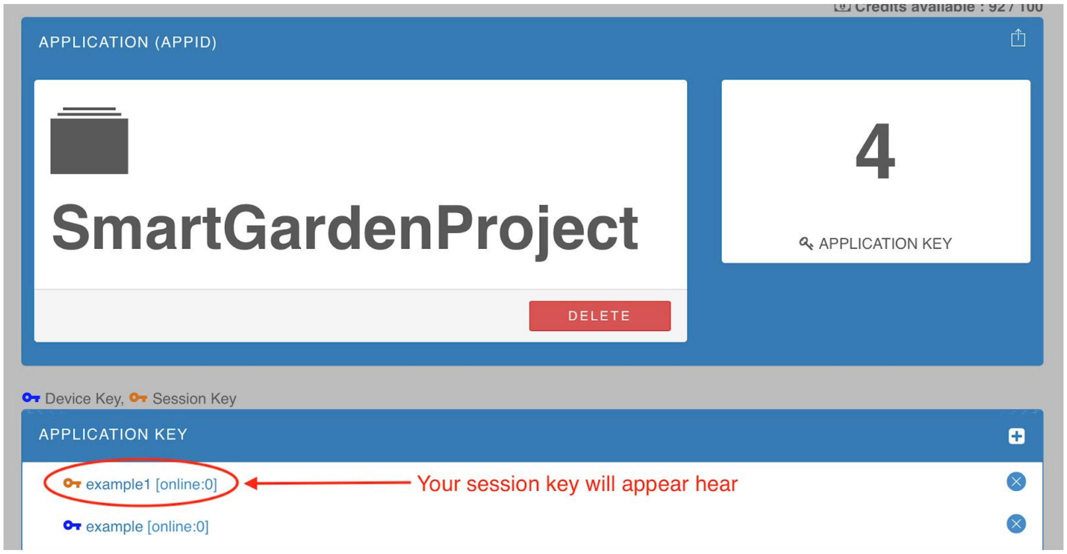

Figure C11. Location of the session key

C.6 Viewing the keys

After both keys are created, the user has to click on device key and session key one by one to see the keys that are hidden inside as shown in figure C12. Then, the box will pop up like what is shown in figure C13.

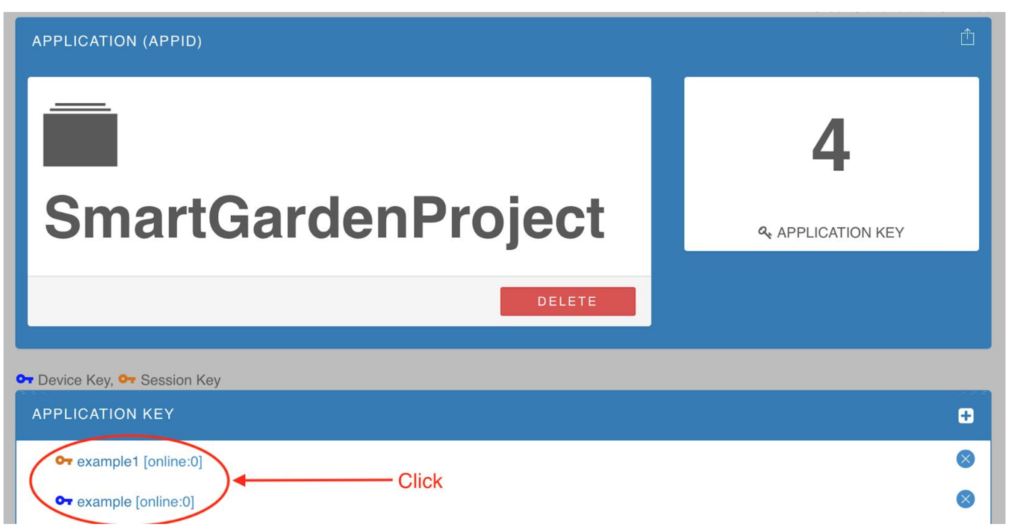

Figure C12. Seeing the keys

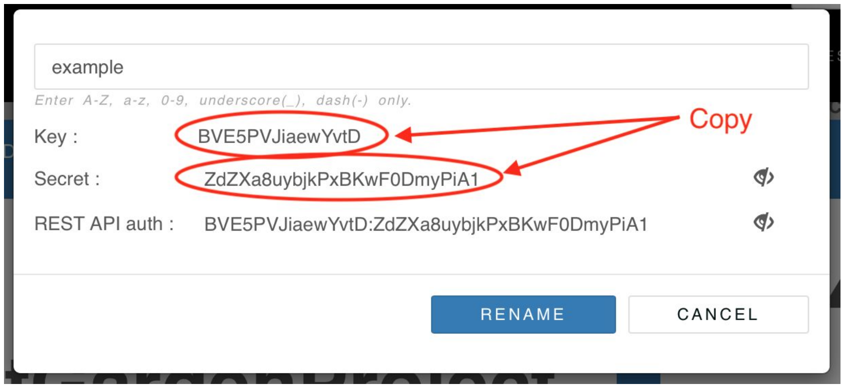

Figure C13. Example of the key

C.7 Applying the key

After the user gets both keys, the user has to rewrite the device key in the python as shown in figure C14. Also, the user has to rewrite the session key in the html file as shown in figure C15.

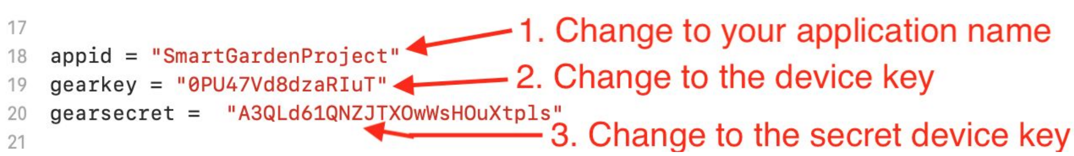

Figure C14. Add the device key in python file

Figure C15. Adding the session key in html file

APPENDIX D

Website Manual

D.1 Home page

The user goes to the home page and clicks ‘Login to Admin Mode’ on the home page as shown in figure D1.

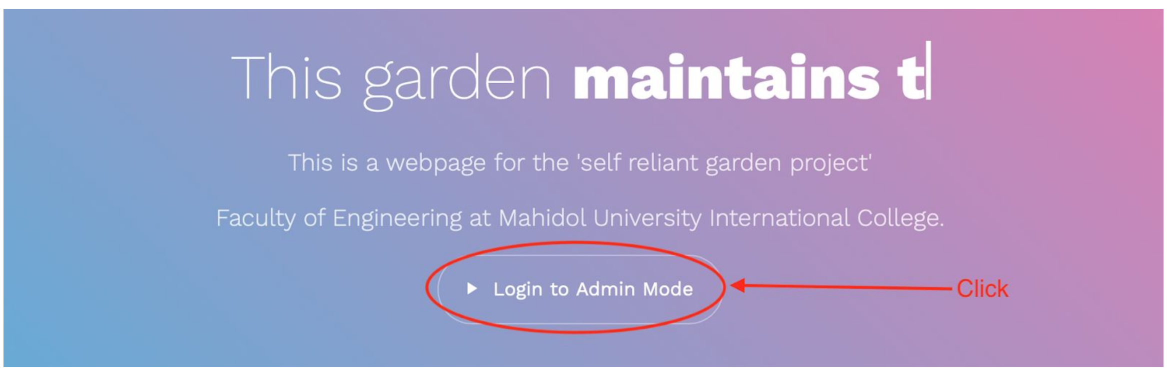

Figure D1. Login button from the website

D.2 Logging in

The user types in username: putter and password: putter (default setting) then click submit as shown in figure D2.

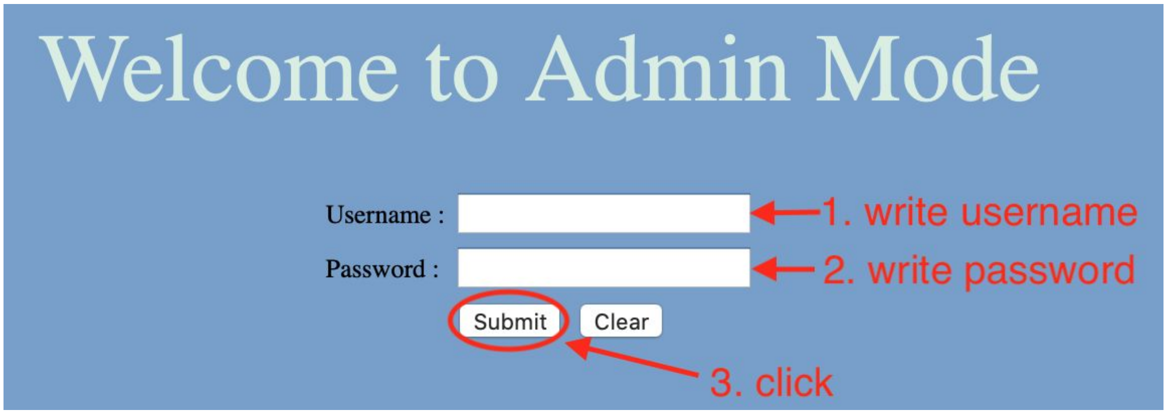

Figure D2. Username and Password input from the website

D.3 Changing threshold and viewing graph

After the user is authenticated page, the user can input preferred temperature and humidity then click submit to send data to Netpie as shown in figure D3. Moreover, the user can see the graph of temperature and humidity on this page.

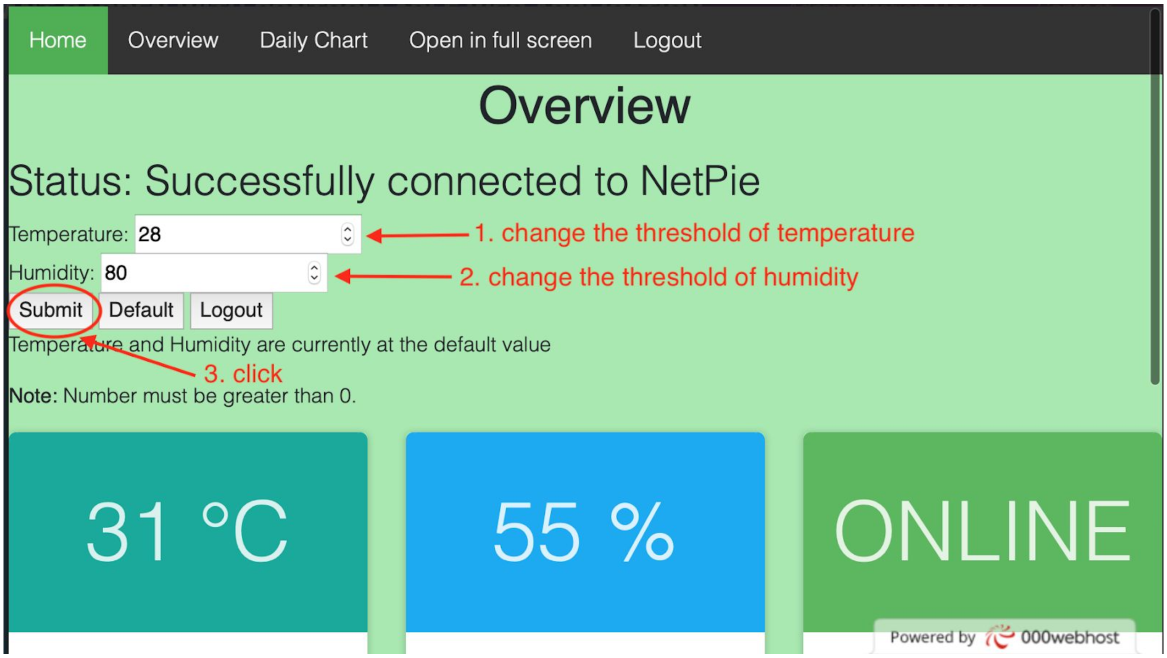

Figure D3. Temperature and humidity inputs from the website

APPENDIX E

Hardware specifications

Raspberry pi

1. Quad Core 1.2GHz Broadcom BCM2837 64bit CPU
2. 1GB RAM
3. BCM43438 wireless LAN and Bluetooth Low Energy (BLE) on board
4. 100 Base Ethernet
5. 40-pin extended GPIO
6. 4 USB 2 ports
7. 4 Pole stereo output and composite video port
8. Full size HDMI
9. CSI camera port for connecting a Raspberry Pi camera
10. DSI display port for connecting a Raspberry Pi touchscreen display
11. Micro SD port for loading your operating system and storing data
12. Upgraded switched Micro USB power source up to 2.5A

Am2320

1. Supply Voltage: 3.1 to 5.5V
2. Operating Range and Accuracy (Temperature): -40 to 80 C; +/-0.5 C
3. Operating Range and Accuracy (Humidity): 0-99% RH; +/-3% RH
4. Output Signal: I2C via single-bus

5. Dimensions (excluding pins): 15.0mm (0.59") length x 12.1mm (0.47") width x 4.5mm (0.17") thick

TCA9548A

1. Operating power-supply voltage range of 1.65 V to 5.5 V
2. 0- to 400-kHz Clock frequency
3. 1-to-8 Bidirectional translating switches
4. I2C Bus and SMBus compatible
5. Active-low reset input
6. Three address pins, allowing up to eight TCA9548A devices on the I2C bus

DS18B20

1. Usable temperature range: -55 to 125°C (-67°F to +257°F)
2. Uses 1-Wire interface- requires only one digital pin for communication
3. Multiple sensors can share one pin
4. ±0.5°C Accuracy from -10°C to +85°C
5. Query time is less than 750ms
6. Usable with 3.0V to 5.5V power/data

Relay 5v

1. 5V 4-Channel Relay interface board
2. High-current AC250V/10A, DC30V/10A relay
3. Status LED
4. Water Pump DC 12V 4.2W 240L / H
5. Pump material: ABS
6. Temperature resistance range: 0 to 60 degrees Celsius
7. Conditions of use: Continuously

8. Power consumption: 4.2W
9. Received voltage: 12V DC
10. Flow rate: 240L/H
11. Lift: 300cm / 9.8ft
12. The use: Submersible pump
13. Drive mode: Electromagnet
14. Transportation media: Fresh water pump
15. Working life: 20000 hours