ECE 445

SENIOR DESIGN PROJECT

DESIGN DOCUMENT

Desktop-sized Environment-controlled Greenhouse

Team #2

ZHIMIN WANG
(zhiminw2@illinois.edu)
ZE YANG (zeyang2@illinois.edu)
TAORAN LI (taoranl2@illinois.edu)
HAOYU QIU (haoyuq2@illinois.edu)

TA: Qi Wang

Sponsor: Professor Wee-Liat Ong

March 20, 2023

Contents

1 Introduction

A greenhouse (also called a glasshouse) is a structure with walls and a roof made chiefly of transparent material, such as glass, in which plants requiring regulated climatic conditions are grown [1]. These structures range in size from small sheds to industrial-sized buildings, according to various purposes. Although greenhouse technology has shown its value in different uses, people who live in apartments in urban areas have not enjoyed such techniques yet. There are still several obstacles and gaps that remain to be solved for such kinds of products.

1.1 Background and Problem

Greenhouse plays a significant role in modern agriculture, especially in densely populated areas such as eastern China. Greenhouse environmental conditions have proven efficient and essential for crop yield, pest prevention, and energy saving [2]. As a productive system, the large-scale and medium-scale greenhouses allow us to respond to the growing global demand for fresh and healthy crops throughout the year, which is widely applied in agricultural production [3]. Traditionally, small-scale greenhouses are used in agricultural experiments. Researchers cultivate their plants in a modular environment-controlled greenhouse, to gather data on the state of crop growth in a highly specified and optimized environment.

In most cases, these greenhouses are not intended for ordinary consumers. Even those greenhouse products marked for home use, are mainly for those customers who live in houses with gardens. It is not likely for a user who lived in an apartment to enjoy a greenhouse for ornamental use. In fact, several obstacles remain to be solved for a customer greenhouse product. First, traditional greenhouses are usually too large in their size. The most common glasshouse size for growers is 8 to 10 feet wide, while the large greenhouses for sale may range from 12 to 20 feet in width [4]. Secondly, the greenhouse industry's current practices require considerable energy to power lighting to maintain plant growth on overcast days[5]. Excessive energy consumption also makes it not applicable for household use. Considering the above two factors, it is very inconvenient for the customers to install and carry away. The main cause for the large size and the high energy consumption is that the greenhouse environment is not easily controlled [3]. As its climate parameters are interrelated, a specific amount of sensors should be installed in a comparably small model, which is the third problem. Another problem is that current products on the market do not have fully-featured applets to support the use of the greenhouse. Since customers nowadays are more likely to be accustomed to using their phones to control devices in their homes [6], there is still much room for improvement in this area.

1.2 Overview of the Solution

In an attempt to solve the problems mentioned above, we plan to design a desktop-size environment-controlled greenhouse that can be used for ordinary customers. To reduce its size and energy consumption, only the necessary components would be kept in the product. The product is a cuboid space with an environment-controlling system. All the advanced control functions will be implemented through the applet on the mobile phone. The whole product's dimension is strictly controlled to be the desktop size. The energy consumption should be similar to general household appliances.

1.3 Components List and Visual Aid

1.3.1 Components List

1. There is a main planting cube. The model should be able to hold a fully functional environment-controlling system.
2. The environment-controlling system includes:
   (a) Multiple adjustable LEDs to provide demanded light.
   (b) A temperature-controlling system that can maintain the temperature.
   (c) A humidity-controlling system that can maintain the humidity.
3. A filter for the input and out gas, which should filter the harmful particles.
4. A fan can input and output the air according to the demands.
5. A camera that can monitor the plants: OV2640/OV5640 (have not determined)
6. A 5V relay for the water vent.
7. Several environment detectors should be installed, including:
   (a) Temperature: DHT11/DHT22[7]
   (b) Humidity: DHT11/DHT22[7]
   (c) Illumination detector.
   (d) Air quality.
8. A mobile phone app that can receive the data and adjust the settings.
9. A cloud server to transfer the data from mobile phone applications and ESP32 board.

1.3.2 Visual Aid

As shown in Fig. 1, the whole model consists of three parts: a main growing space with a water tank and a control board, a cloud server, and a mobile phone. The detailed 3D model version of the growing space is shown in Fig. 2.

The graphic flat design for the upper level of the model is shown in Fig. 3. Its 3D model version is shown in Fig. 4. The graphic design for the lower levels is shown in Fig. 5 and Fig. 6

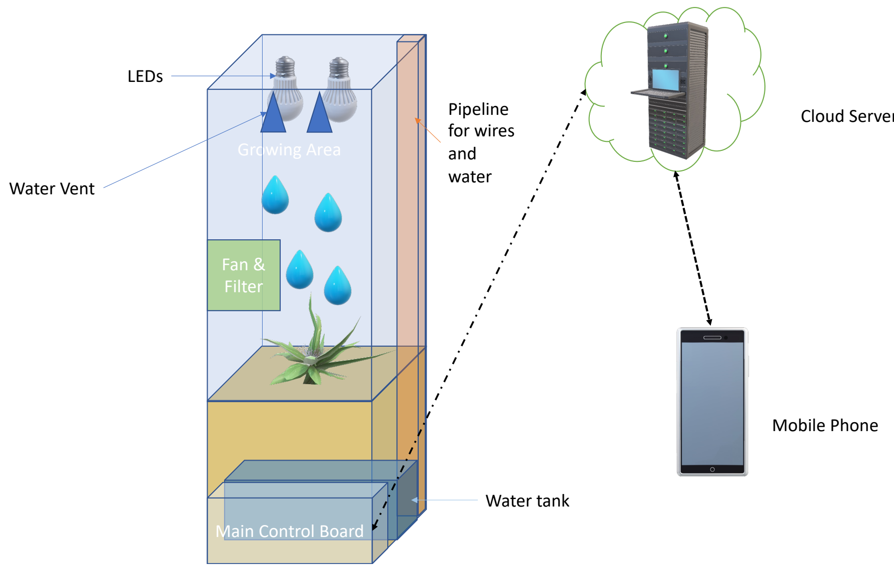

Figure 1: Overview of the model

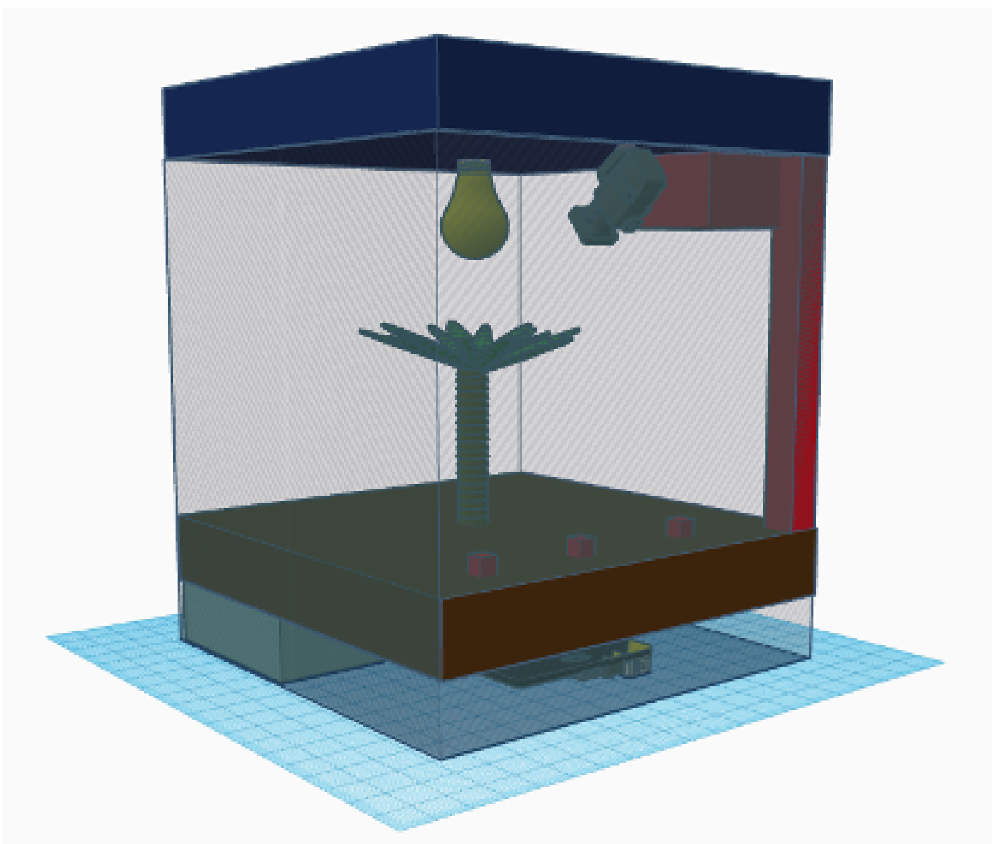

Figure 2: 3D Model draft

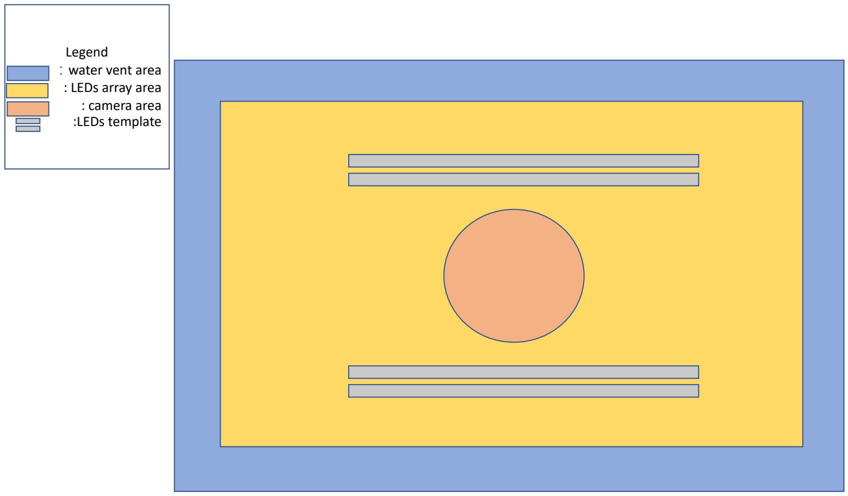

Figure 3: Model's upper platform design

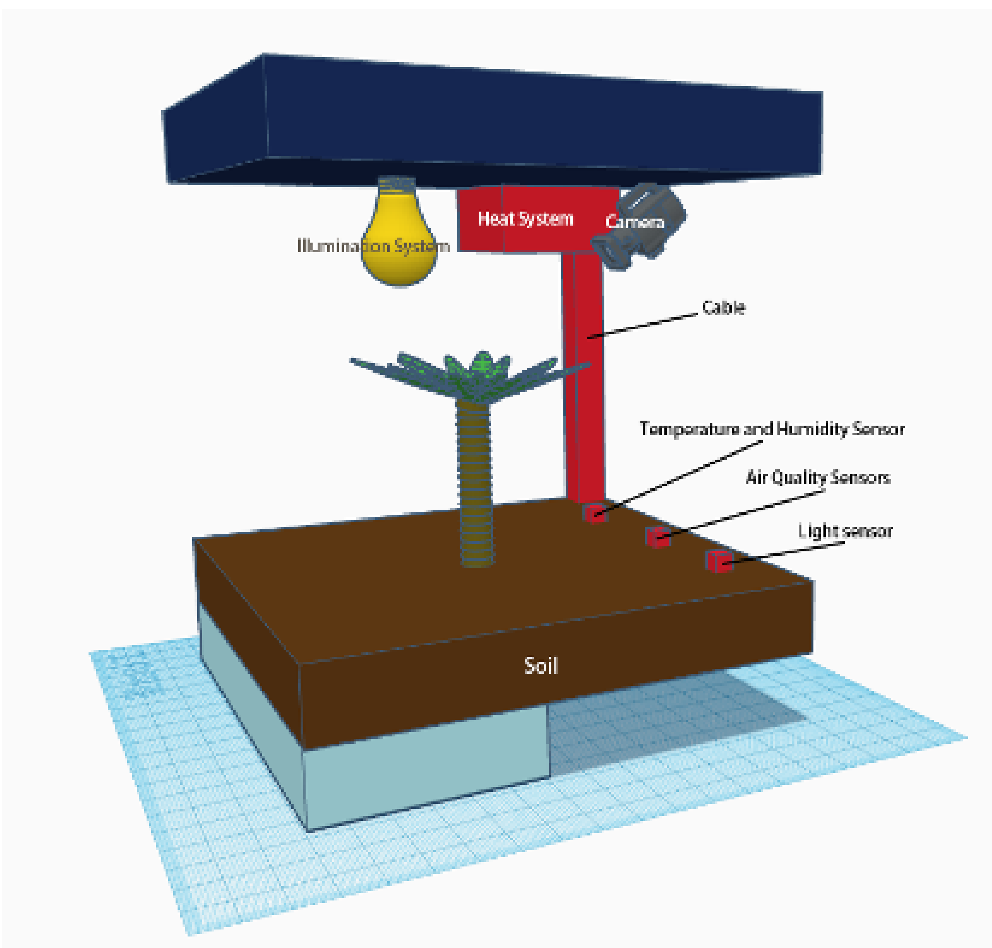

Figure 4: Model's upper platform design (3D version)

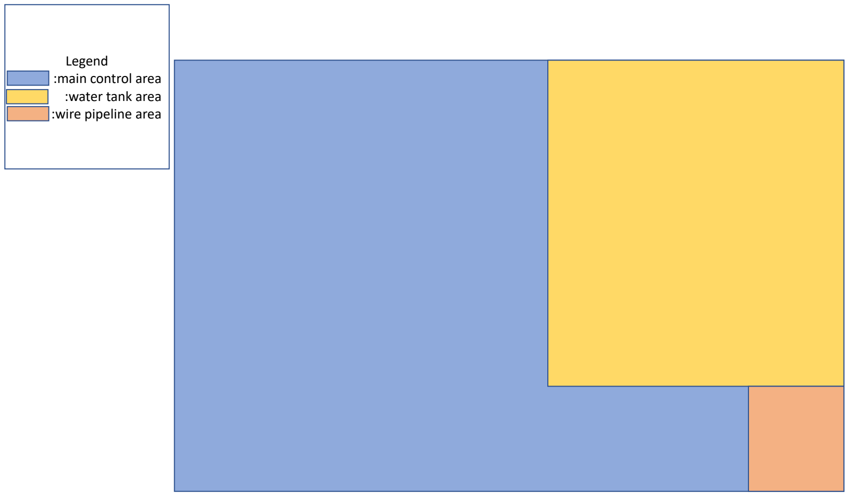

Figure 5: Model's lower platform design

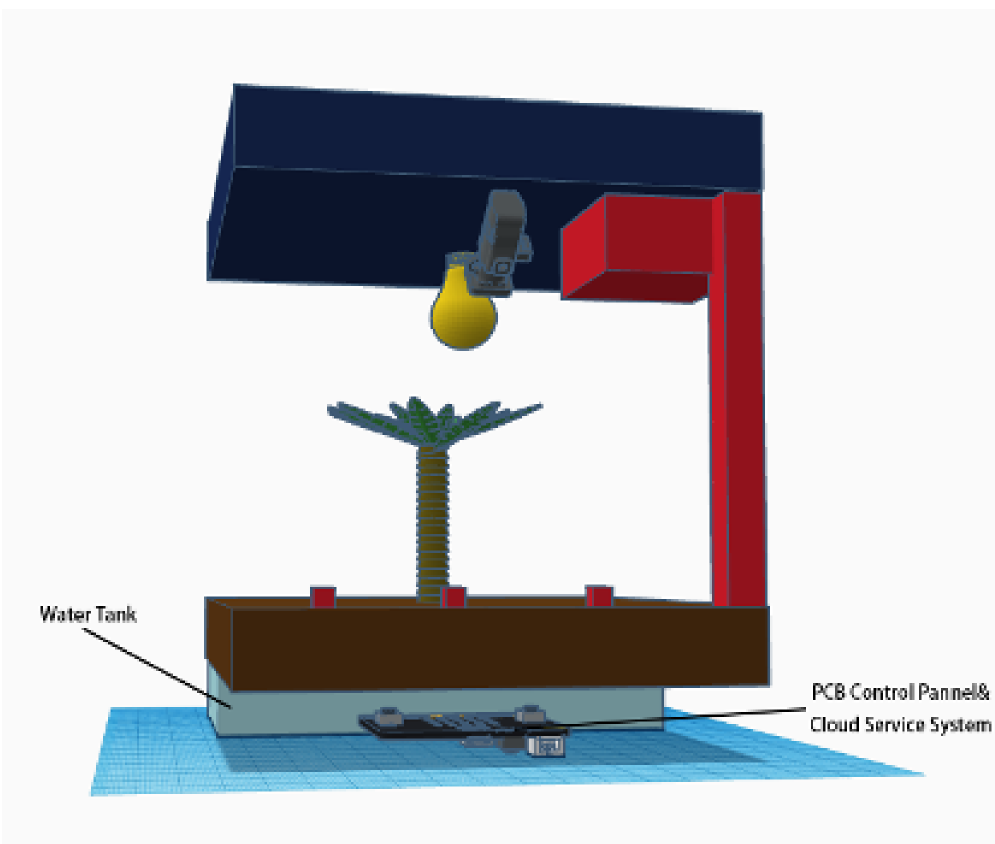

Figure 6: Model's lower platform design (3D version)

1.4 High-Level Requirement List

1. The whole model's size should be controlled under 40cm (width) * 40cm (length) * 90cm (height).
2. The temperature controlling system should be able to control the temperature in the main planting area from 18° to 24°C (64°F – 75°F) [8].
3. The humidity controlling system should be able to adjust the humidity in the growing space from 70% to 90%, as it is a relative humidity setpoint for most plants [8].

2 Design and Requirement

2.1 Block Diagram

The design's block diagram overview is shown in Fig. 7. In order to decrease the size of the accessories in the model, leaving more space for the growing area, only the necessary sensors and a chip microcontroller would be left in the model. There will be a default environmental setting for the model to maintain the temperature and humidity. The information collected by the sensors will be packaged and sent to a cloud server. The server will then process these data, sending them to the mobile phone. The users can view this information directly on their mobile phones, and adjust the environmental settings.

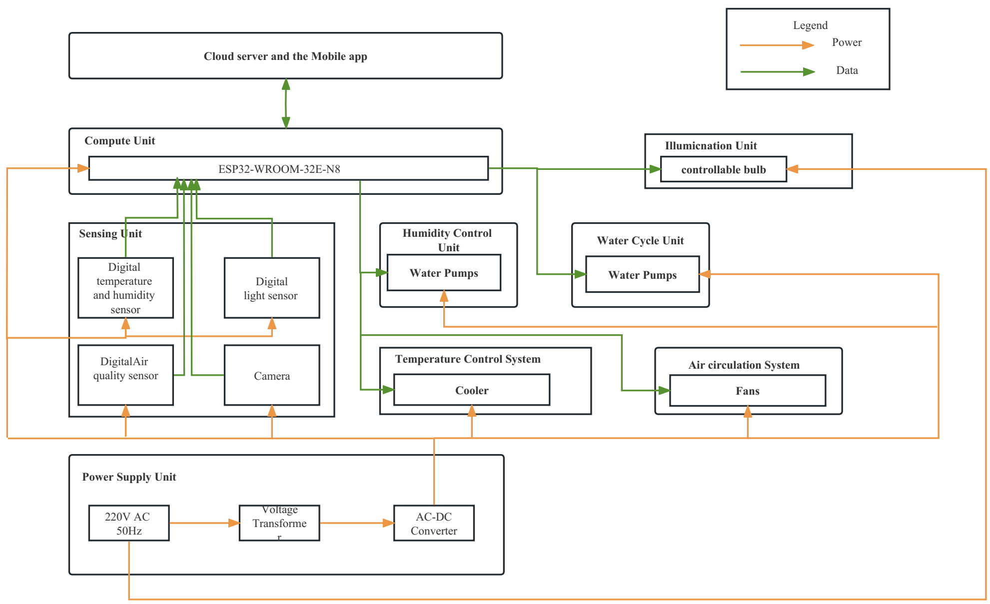

Figure 7: Overview of block diagram

2.2 Subsystem Overview

2.2.1 Model Shell

For the model shell, we will use special materials to decorate the surrounding area for people to enjoy and receive daylight to reduce the energy consumption of plant light. In 3D printing, we use materials such as fully transparent PLA or fully transparent photosensitive resin to ensure the overall light transmission performance. Also, we can either use the acrylic sheet to assemble the device to ensure light transmission. A double layer of material is employed to maintain good insulation or moisture retention. The growing space in the model shell is the central part of this tabletop greenhouse landscape. Environmental factors such as temperature, humidity, and this part need to be controlled by the program we designed. This desktop landscape facility is soil cultivation, and it can detect whether various environmental conditions are suitable through the mobile app. The requirements and verification can be found in Table 1.

Requirement	**Verification**
1. Provide enough space to integrate the various control elements and a good plant-growing environment. 2. The shell of the model should have good light transmission (plastic or glass) and should be able to achieve excellent heat preservation and moisture retention.	1. The model shell is designed according to the ordered and assembled console, cables, etc. The dimensions are scaled according to the size of the various consoles at the time of assembly. At the same time, we will ensure that the plants have at least 40cm*40cm*80cm of growing space. 2. With a temperature difference of 5 °C between the inside and outside, the device was left to stand for one hour. Use a temperature and humidity sensor to detect, without the working of the temperature control system, the change in temperature should be controlled to within 3 °C. Meanwhile, with a humidity of 80%, the humidity decline should be controlled under 10%.

Table 1: Model Shell

2.2.2 Illumination System

The light inside the growing space is provided by side-by-side LED devices. Since plants need different light conditions at different growth stages, we need to control the wavelength, intensity, and duration of light in the growing space by changing the light of LEDs. It is worth mentioning that the light environment outside the greenhouse is also under our consideration, such as the difference between day and night light, to maintain the stability of the light environment in the growing space. These light conditions are measured by a photoresist and portable optical wavelength meter and reflected in the mobile app. A light sensor, shown in Fig. 8, would be used to detect these parameters. The requirements and verification can be found in Table 2.

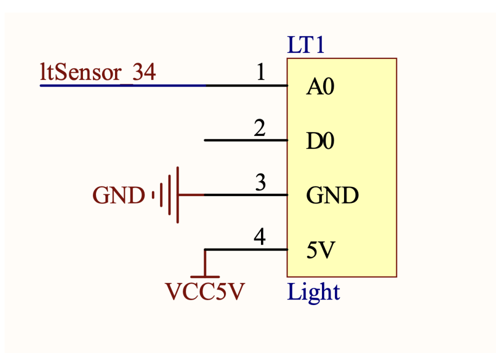

Figure 8: Illumination Sensor

Requirement	Verification
1. In general, plants need more blue light for growth space and then need red and orange light for production space [9]. The LEDs should be able to provide blue light wavelength at 400-500nm, green light spectrum from 500-600nm, red light at 600-700nm, and far-red radiation from 700-800nm meeting the requirements for greenhouse planting [10]. 2. The heat produced by the LED light should not affect the growth of the plant. 3. In the case the light intensity in the environment is too high that the LED array is unable to adjust to an appropriate condition. The user should manually adjust the light condition of the environment. 4. Report the data of the current light condition to the mobile app. The user could monitor the condition and adjust it manually.	1. Turn on the power supply of the illumination system systems, the system could detect the current light condition with the LT1 light sensors and adjust to the ideal light by the LEDs. 2. We place a green plate at different heights in the growth space to test the temperature increment. The increased temperature should not be higher than 2 °C. When the increased temperature is at 2 °C, we mark this height as the upper limit of the plant's height. The plant should not be taller than this height. 3. We provide a curtain that could cover the growth space. If the light is too bright in the environment the LED array cannot adjust. The user would be warned by the mobile app. In this case, the user should close the curtain or place the greenhouse in a darker place. 4. The real-time data of light conditions will be fed to the mobile app through the light sensors placed in the growth space. The user can monitor the current light condition and adjust the target LED light intensity, wavelength, and duration manually.

Table 2: Illumination System

2.2.3 Air Circulation System

The plants we grow need a good air circulation to grow, which we need to provide in the growing space. Different kinds of plants have different efficiency in filtering the air quality, while their abilities on absorbing the specific harmful particles in the air are various. For example, aloe vera is a good formaldehyde absorber, which can absorb 90% of formaldehyde contained in 1 cubic meter of air but may have lower efficiency on other particles. We want to achieve air filtration and circulation using a filter and fan, timed to switch on and off through the app's control program. We will also air composition testing instrument, using MQ-135 shown in Fig. 10 and Fig. 9, and reflect the data in real-time on the cell phone app, which can remotely control the switch of the air circulation system. The requirements and verification can be found in Table 3.

Requirement	Verification
1. The system should be able to adjust its exhaust airflow according to the plant the user grow in the growing space and the information received from the air quality detector. 2. Report the data of the current air quality to the mobile app. The user can adjust it manually through the switch.	1. Connecting the air circulation system systems to esp32 L5 relays, the system could measure the current air quality with an air quality sensor and filter the air in the growth space. Also, we will record the ideal air quality for commonly grown plants in the mobile app. 2. The real-time data on air quality will be fed to the mobile app through the air sensor placed in the growth space. The user can monitor the current air quality and can also manually start a filter process.

Table 3: Air Circulation System

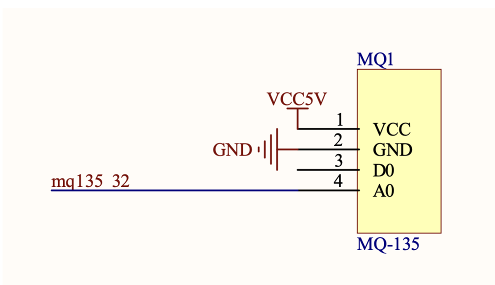

Figure 9: Schematics of MQ-135

Figure 10: MQ-135

2.2.4 Temperature Control System

This system controls the temperature in the growing space to meet the different needs of different plants and different growth stages of plants. We achieve temperature control through a small air conditioner that works with the air circulation system to raise or lower the temperature. At the same time, the temperature in the growing space will also be measured by a digital thermometer for real-time transmission to the cell phone app. The requirements and verification can be found in Table 4.

Requirement	Verification
1. Maintain a temperature range of around 18º-24ºC (64ºF – 75ºF) inside the growing space. as the temperatures outside of this range would normally lead to slower or halted growth and suboptimal plant quality [8]. 2. Report the temperature condition to the mobile app. When the temperature is too high or too low, a pop-up window will warn on the mobile app.	1. Turn on the power supply of the refrigeration systems, which could be used to control the temperature. In one hour, without manually controlling, the temperature should be controlled around targeted temperature within $\pm 3°C$. 2. The real-time data of temperature will be fed to the mobile app through the capacitive temperature and humidity sensors DHT11 built into the refrigeration systems and the cloud system, and the user can adjust it through the app. Detect and control the system frequently at different times during the twenty-four hours.

Table 4: Temperature Control System

2.2.5 Humidity Control System

This system is used to provide plants with the right humidity environment for growth by adjusting the supply of water. We use a moisture Sensor to detect the humidity condition of the air around plants. Underneath the device, we are equipped with a water tank, which can be watered regularly through the mobile app's control or manually through a moisture sensor that detects the moisture condition. DHT11, shown in Fig. 11, can detect both humidity and temperature. The requirements and verification can be found in Table 5.

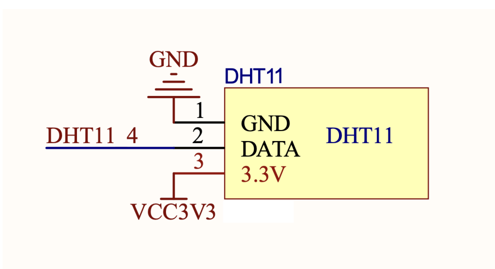

Figure 11: Schematics of DHT11

Requirement	Verification
1. Maintain humidity around 80%, which is a relative humidity setpoint for most plants. At this level, growth rates are highest for common greenhouse plants. At higher or lower humidity levels, plant physiological processes may slow down, leading to slower growth and lower quality output [8].	1. Turn on the power supply of the humidity systems, which could be used to raise or lower humidity. During the twenty-four hours, the temperature should be maintained within 70-90% degrees to meet the requirement. 2. The real-time data of humidity will be fed to the mobile app through the capacitive temperature and humidity sensors DHT22 built into the humidity systems and the cloud system, and the user can adjust it manually. Detect and control the system frequently at different times during the twenty-four hours.

Table 5: Humidity Control System

2.2.6 Real-time Monitoring System

The detection system can detect the growing space in real-time, and this is detected by the installed camera. We planned to use OV2640, which is shown in Fig. 12 and Fig. 13, as the camera. This module can produce an image array capable of operating at up to 15 frames per second (fps) in UXGA resolution with complete user control over image quality, formatting, and output data transfer [11]. At the same time, we can monitor the situation inside through the mobile app. The requirements and verification can be found in Table 6.

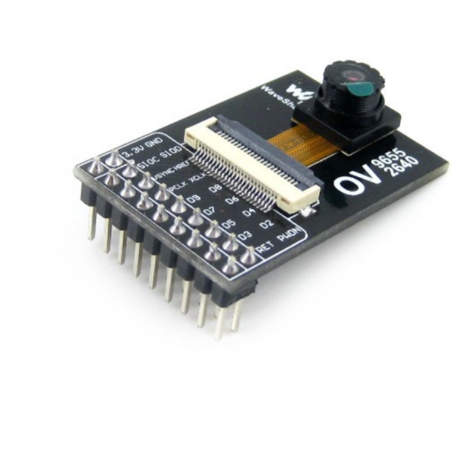

Figure 12: OV2640

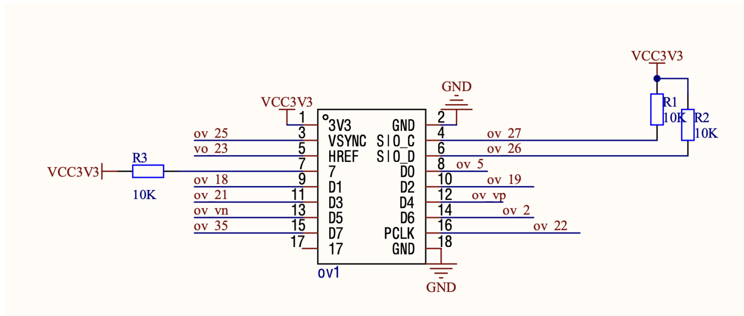

Figure 13: Schematic of OV2640

Requirement	Verification
1. To save energy, the default status for the camera module should be off. It should be manually controlled by the mobile phone. When the user turns it on, it will send the live video to the mobile app.	1. Connect the real-time monitoring system systems to esp32 L5 relays, the system would monitor internal space with an OV2640 camera which is a low voltage CMOS image sensor that provides the full functionality of a single-chip UXGA (1632x1232) camera and image processor in a small footprint package. The camera would be connected to the server for the transportation of live video. It is fully controlled by the user of the mobile app. When the user turns the real-time monitoring system on, he or she can monitor the situation inside the growing space with the mobile app.

Table 6: Real-time Monitoring System

Main Control System

The main control system is composed of an ESP32[12] (in Subsection 2.2.7), a Tencent Cloud Server (in Subsection 2.2.8), and a mobile phone app (in Subsection 2.2.9). All the sensors' information should be first sent to ESP32 to process and packaging. These data should then be pushed into the cloud server. The cloud server will set up a log for them. The mobile phone app can receive the information from the server and feed it back to the user. At the same time, we can also remotely control the light, temperature, humidity, and air condition in the device through the mobile app and can set the timer operation or threshold operation. The mobile app can also give suggested ranges of various parameters for the user's reference based on the information in the database and network according to the plant type and growth stage selected by the user.

ESP32 should be able to handle multiple sensors' information. The server should be able to store users' information and a database for planting. A backup for the sensors should also be included as a part of the database on the server. The essential features of the mobile phone app are like a remote controller. All the sensors' switchers are integrated into the app.

2.2.7 ESP32

We use ESP32-WROOM-32E-N8, shown in Fig. 14 and Fig. 15, as the control center of the whole system. Digital temperature and humidity sensor (DHT11), camera (OV2640), light sensor, and air quality sensor (MQ-135) will collect environment data and send it to the control center for further processing (shipping to the server and activating parts according to the signal from server) and display on the screen of Greenhouse. The requirements and verification can be found in Table 7.

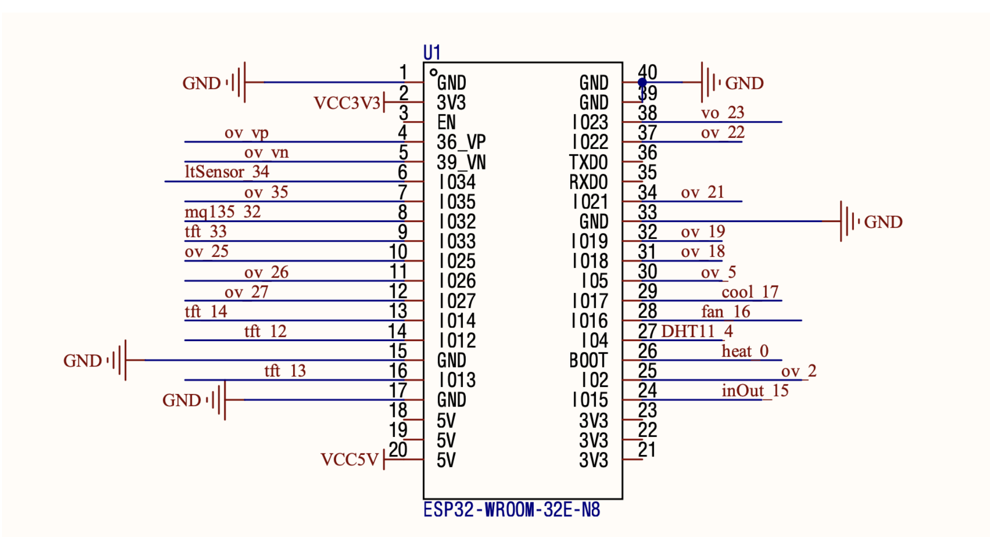

Figure 14: Schematics of ESP32

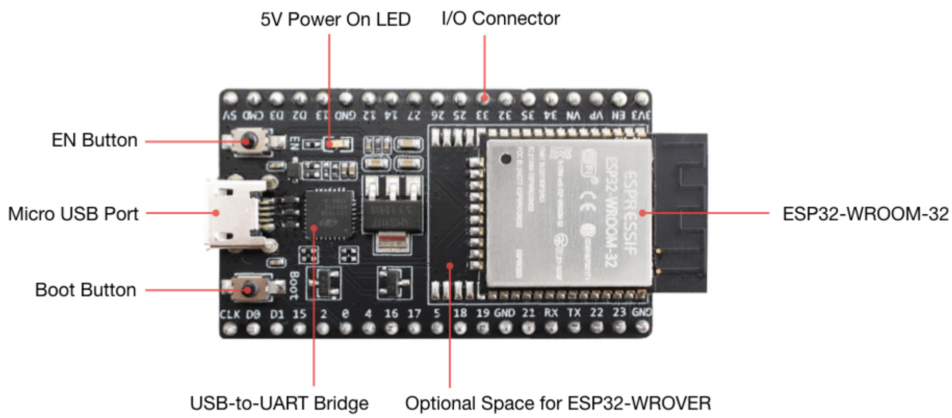

Figure 15: ESP32

Requirement	**Verification**
1. The microcontroller needs to receive multiple data inputs, process all the data, and produce multiple data outputs. We need many pins to connect all the parts. 2. The data processing part needs to process various data relatively quickly to ensure our system can work properly.	1. We would plan a connection between the microcontroller and the other electrical components that can make a good distribution of each of the necessary pins. We will connect all sensors to the microcontroller to see if the sensor can work well. For example, when We put the lighter in front of the sensor and burn it for a few seconds, causing the air quality to deteriorate the MQ135 air quality sensor should detect the air quality change correctly. 2. We need to optimize the code as much as possible to speed up the processing. When the environmental factors change, i.e., the temperature increases, the changed value should be displayed on the screen in 5 seconds.

Table 7: ESP32

2.2.8 Cloud Server System

This system will interact with ESP32 and the smartphone app for data transmission. It will work like a bridge connecting ESP32 and the smartphone app, letting the user activate parts on ESP32. The cloud server system is a 2-core Tencent cloud server. The requirements and verification can be found in Table 8.

Requirement	Verification
1. The system must receive from and transfer data to the phone and esp32. We need several ports for the phone app and esp32. 2. The data processing part should be done relatively quickly to ensure our system can work properly.	1. We would plan and distribute the server ports for the phone app and esp32. For the test, we should connect the phone, the server, and the esp32. And the data can be synced among these three. 2. We need to optimize the code as much as possible to speed up the processing. When the environmental factors change, all the data should be synced among these 3 platforms in 5 seconds.

Table 8: Tencent Cloud Server

2.2.9 Smartphone App System

The key feature of this system is used to provide remote monitoring. The users can check the environment status of the model, including temperature, humidity, air quality, and so on. If there's time, we will further develop the app to control parts in the greenhouse, like changing the water flow and opening the fan. Fig. 16 shows the progress of how the mobile phone interacts with the remote microcontroller through the cloud server. The requirements and verification can be found in Table 9.

Requirement	Verification
1. The cloud server system must receive from and transfer data to the phone and esp32. We need several network ports set for the phone app and esp32. 2. The data processing part should be done relatively quickly to ensure our system can work properly.	1. We would plan and distribute the server ports for the phone app and ESP32. We should connect these three parts for the test and see if the phone app can receive the information from ESP32. ESP32 should follow the switcher's status on the phone (change the status of the switchers on the ESP32 in further developments). 2. We need to optimize the code as much as possible to speed up the processing. All data should be synced in 5 seconds and phone control operations must not have significant delays, like several minutes.

Table 9: Smartphone App

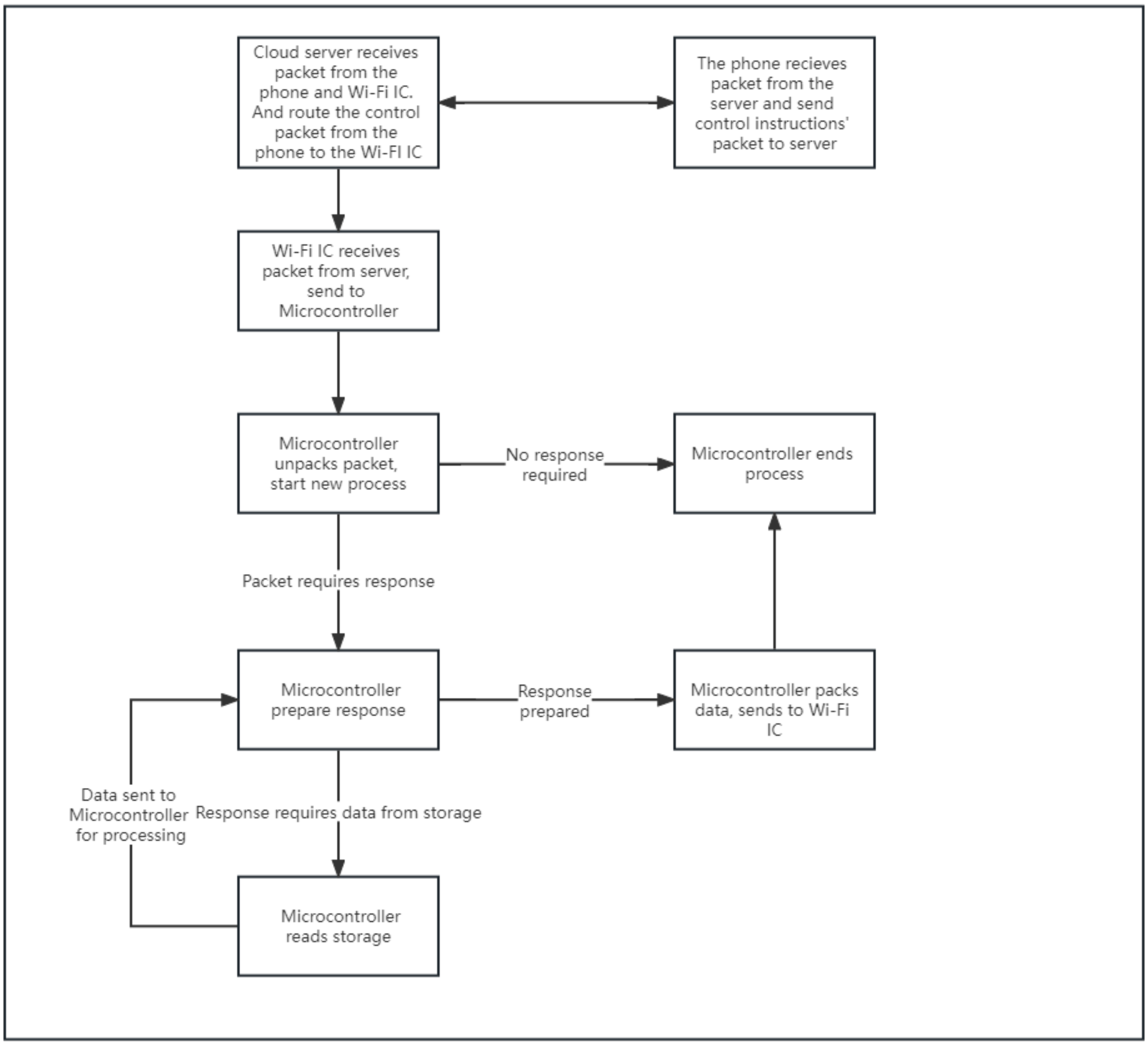

Figure 16: Mobile Phone Control Algorithm

2.3 Board Layout

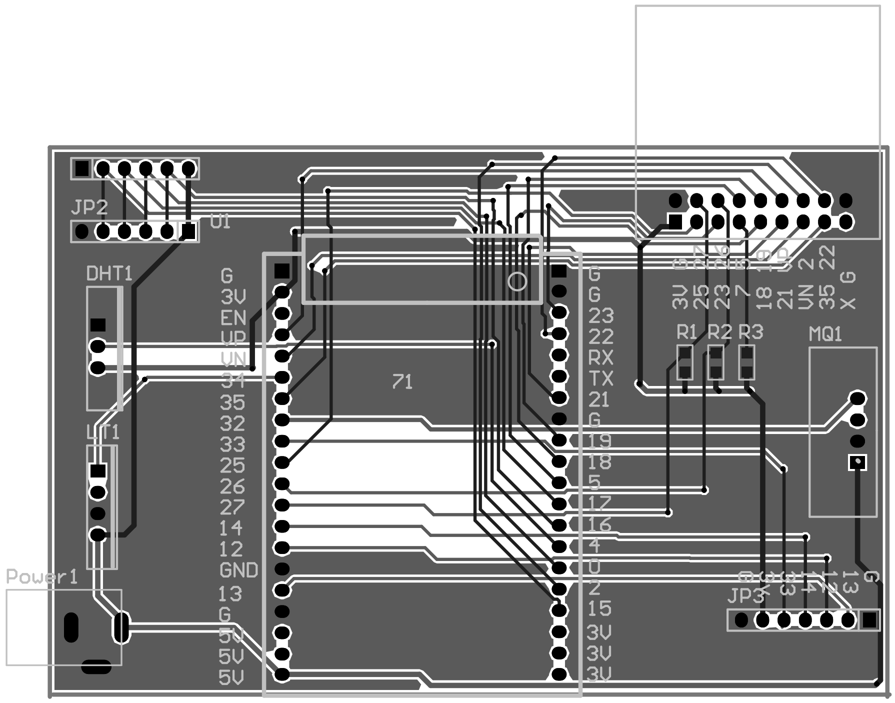

Figure 17: Final PCB Layout

2.4 Tolerance Analysis

Light is an important environmental factor influencing plant growth, development, and secondary metabolism. All plants have their own optimal light intensity (illuminance) ranges for growth [13]. To meet the requirements for illumination, one important factor that we need to take into consideration is photosynthetic photon flux density (PPFD). PPFD can be regarded as the unit of photosynthetic active radiation (PAR). PAR light is the wavelengths of light within the visible range of 400 to 700 nanometers (nm) that drive photosynthesis. It defines the type of light needed to support photosynthesis [14].

For different plants, as shown in Fig. 18, the PAR requirements are different. To achieve the required types of light, given an illuminance value (lux or footcandles), we need to calculate the PPFD in micromoles per second per square meter ($\mu mol \cdot s^{-1} \cdot m^{-2}$) for the given light source.

According to CIE 1931 luminous efficiency function $V(\lambda)$ [16], we can then get the total luminous flux in a source of light:

$$\Phi_v = 683.002(lm/W) \cdot \int_0^{+\infty} V(\lambda)(\lambda)\Phi_{e,\lambda}(\lambda)d\lambda, \tag{1}$$

where Φ_v is the luminous flux in lumens, $\Phi_{e,\lambda}$ is the spectral radiant flux in watts per

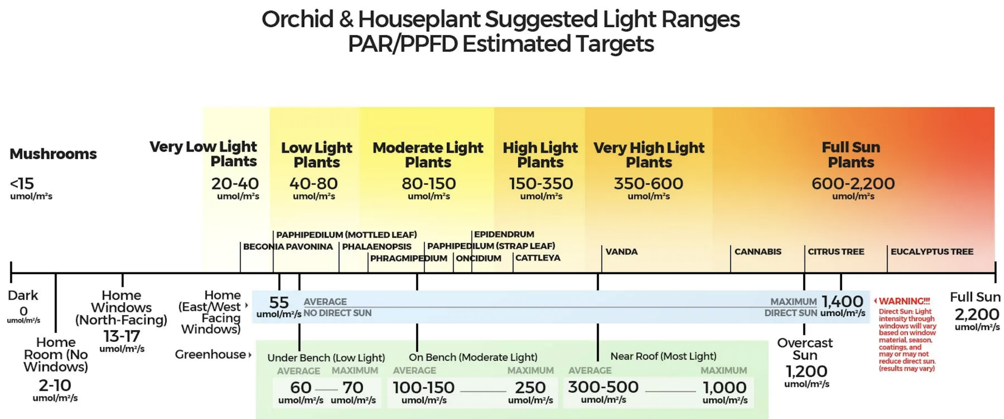

Figure 18: Suggested light ranges[15]

nanometre, $V(\lambda)$ is the CIE 1931 luminous efficiency function at wavelength λ in nanometres.

According to eq. 1, if the source light is across the relevant wavelengths (400-700 nm), then for each lumen, we can calculate the spectral radiant flux $\Phi(\lambda)$ for plants in watts per nanometer:

$$\Phi(\lambda)/lumen = [W_{rel}(\lambda)]/[683.002(lm/W) \cdot \sum_{400}^{700} V(\lambda)(\lambda)\Phi_{e,\lambda}(\lambda)d\lambda], \tag{2}$$

where $W_{rel}(\lambda)$ is the relative spectral power distribution.

We can get the photosynthetic photon flux (PPF):

$$PPF/nm = 10^{-3} \cdot [\lambda\Phi(\lambda)]/(N_a hc), \tag{3}$$

where Na is Avogadro's constant, h is Planck's constant, and c is the speed of the light.

Thus, we get the PPF equals to $8.359 * 10^{-3} * \sum_{400}^{700}[\lambda\Phi(\lambda)\Delta\lambda]$. Once the wavelength of the LEDs has been decided, within the PPF calculated, we can then calculate the PPFD for the given light source.

The distance between the light and the plants will affect the PPF and PPFD heavily, as the area covered by the light is increasing when the distance is increasing.

As a rough calculation, assuming the PAR provided by the LEDs is 150 PPF, we can have an equation:

$$PPFD = \frac{PPF}{r \cdot x^2}, \tag{4}$$

where r is the conversion factor depending on different LED products. If r is 14.6 [5000K white light LED (Philips Luxeon Rebel LXW8-PW50) [17]], we can get Fig. 19, where the y-axis represents the PPFD, and the x-axis represents the distance between the light source and the plants.

Thus, the LEDs should provide at least 250 PPF, while the distance should be controlled under 0.46 m.

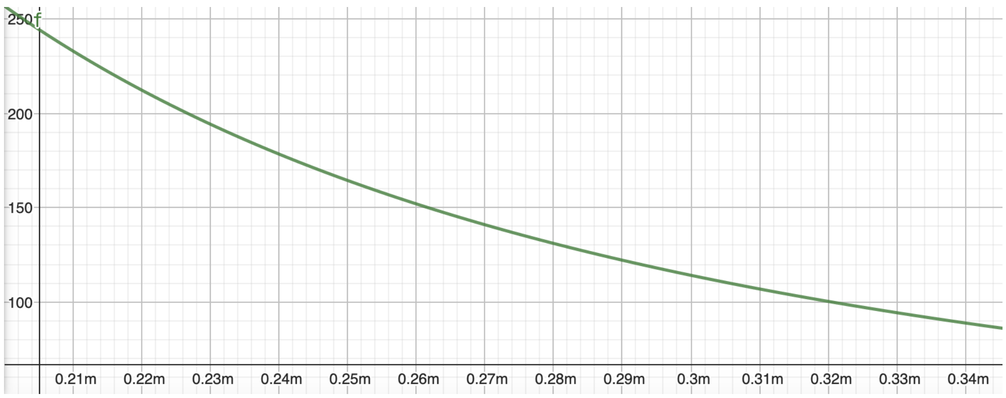

Figure 19: PPFD on different distances (LEDs: 150 PPF)

3 Costs and Schedule

3.1 Cost Analysis

1. **Labor**

   According to UIUC's Computer Engineering average salary ($105,325), the fixed development costs are estimated to be $52/hour, 20 hours/week for everyone.

$$weeklycost = \frac{\$52}{hour} \cdot \frac{20hours}{week} \cdot 14weeks \cdot 4 \cdot 2.5 = \$145,600 \tag{5}$$

   (a) Zhimin Wang: $36,400

   (b) Ze Yang: $36,400

   (c) Haoyu Qiu: $36,400

   (d) Taoran Li: $36,400

2. **Parts** [1]

Part	**Cost**
ESP32-WROOM- 32E-N8 (Taobao)	¥21.8
DHT11 (Taobao)	¥7.9
DHT22 (Taobao)	¥25.8
MQ-135 (Taobao)	¥9.76
OV2640 (Taobao)	¥21.5
OV5640 (Taobao)	¥158+¥5 (delivery)
Manufacture of PCB (Taobao)	¥200+¥5 (delivery)
2-core Tencent Cloud Server (Tencent)	¥30
Acrylic shell	¥1200
Total cost of parts	¥1706.56

Table 10: Part cost

3. Total cost estimated: ¥1706.56 + $ 145,600

[1] Noting: Costs spend as of 4/19/2023.

3.2 Schedule

WEEK	Zhimin Wang	Ze Yang	Taoran Li	Haoyu Qiu
2/13/2023	Form the team.	Form the team.	Form the team.	Form the team.
2/20/2023	Request for Approval (RFA) and contract writing.	Modified RFA.	/	/
2/27/2023	Writing all the remaining part of the proposal.	Proposal's subsystem description.	Proposal's subsystem description.	Proposal's subsystem description.
3/6/2023	Keep writing the proposal; draft model design; contact modules' manufacturor	Contact modules' manufacturer; Proposal's subsystem description	Proposal's subsystem description; block diagram draft	Proposal's subsystem description; block diagram draft
3/13/2023	Ordering the modules; Design Document (DD) writing; weekly meeting ppt modification; PCB designing	DD subsystem parts: esp32, cloud server, mobile app; weekly meeting ppt drafting; PCB designing	DD subsystem parts: model shell, temperature, humidity	DD subsystem parts: illumination, air circulation camera
3/20/2023	Keeps working on DD; delivery should be arrived by this week	DD subsystem parts: esp32, cloud server, mobile app;	DD subsystem parts: model shell, temperature, humidity	DD subsystem parts: illumination, air circulation camera
3/27/2023	Cloud Server environment testing.	Android Applet developing.	Model shell design for PCB.	Sensor testing and enhance features.
4/3/2023	Connect the cloud server to the ESP32; feature developing	Connect Applet to the cloud server; feature developing	Model analyzing and designing.	Connect the ESP32 to the server.
4/10/2023	Connect the cloud server to the applet; feature developing	Help test the communication among esp32, server, and applet; feature developing	Model analyzing and model producing.	Model analyzing and model producing.
4/17/2023	Finish the connection among esp32, server, and applet; feature testing	Finish the connection among esp32, server, and applet; feature testing	Model producing; installing hardware.	Model producing; installing hardware.
4/24/2023	Final testing; Final report writing.	Final testing; Final report writing.	Final testing; Final report writing.	Final testing; Final report writing.
5/1/2023	Final testing; Final report writing.	Final testing; Final report writing.	Final testing; Final report writing.	Final testing; Final report writing.
5/8/2023	Mock Demo; Final report	Mock Demo; Final report	Mock Demo; Final report	Mock Demo; Final report
5/15/2023	Enhance the performance; Final report.	Enhance the performance; Final report.	Enhance the performance; Final report.	Enhance the performance; Final report.
5/22/2023	Demo	Demo	Demo	Demo

4 Ethics and Safety

According to IEEE codes of ethics [18], there are a few potential safety and ethics concerns we need to take into consideration during the development process.

The first safety-related is that it must not hurt people, which is corresponding to the IEEE Code of Ethics, #1:" to hold paramount the safety, health, and welfare of the public" [18]. Traditionally, the main concern on house-use electrical products is the leaking of electricity. Considering that our product incorporates many precision electronic components, while the working space is quite wet, moisture could do damage to the nodes, causing a short circuit. Since the ESP32's working space is near the water tank, a closed waterproof design is needed for the work area. Meanwhile, the ESP32 board should never be charged over a safe voltage (3.0 to 3.6 V) [19]. The temperature should also be controlled between –40 and 85 °C [19].

Despite the risk in the circuit, there is also a risk that users may use this product for drug-related plant planting, which is against the IEEE Code of Ethics, #1:" to hold paramount the safety, health, and welfare of the public" [18]. We do not want to see our product being used to feed drug abuse. Since cannabis is one of the few plants which flourish under more than 950 to 1,200 photosynthetic photon flux density (PPFD) [20], our product should not choose the LEDs that support this working power.

Another ethics-related problem is that since our product has a monitoring system, the data collected by the camera may include the user's privacy. According to the IEEE Code of Ethics, #1: "to protect the privacy of others, and to disclose promptly factors that might endanger the public or the environment" [18], we need to inform the users of the risk using it. On the other hand, these data should be carefully ciphered and saved on the server with necessary security protection. A leaking of information may lead to a severe problem.

Defects may always exist in our product, and sometimes we may accidentally overlook them due to various reasons. It is essential for us of being humble and ready to accept advice. This is also an implementation of the IEEE Code of Ethics, #5: "to seek, accept, and offer honest criticism of technical work, to acknowledge and correct errors" [18].

References

[1] "Greenhouse." [Online]. Available: https://www.oed.com/view/Entry/81202?redirectedFrom=greenhouse#eid

[2] K. Li, Z. Sha, W. Xue, X. Chen, H. Mao, and G. Tan, "A fast modeling and optimization scheme for greenhouse environmental system using proper orthogonal decomposition and multi-objective genetic algorithm," *Computers and Electronics in Agriculture*, vol. 168, p. 105096, 2020. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0168169919309172

[3] A. Rizwan, A. N. Khan, R. Ahmad, and D. H. Kim, "Optimal environment control mechanism based on ocf connectivity for efficient energy consumption in greenhouse," *IEEE Internet of Things Journal*, 2022.

[4] H. Botanic, "Greenhouse buying guide: Size, location, type." [Online]. Available: https://hartley-botanic.com/guides/what-size-greenhouse-i-need-us/

[5] R. T. Watson, M.-C. Boudreau, and M. W. van Iersel, "Simulation of greenhouse energy use: An application of energy informatics - energy informatics," Aug 2018. [Online]. Available: https://energyinformatics.springeropen.com/articles/10.1007/s42162-018-0005-7

[6] R. Scott, "Consumers are using phones to control smart home devices," Apr 2018. [Online]. Available: https://www.etcentric.org/consumers-are-using-phones-to-control-smart-home-devices/

[7] A. Industries, "Dht22 temperature-humidity sensor + extras." [Online]. Available: https://www.adafruit.com/product/385

[8] DevOps, "Optimal humidity and temperature for greenhouse growing - drygair," Apr 2022. [Online]. Available: https://drygair.com/blog/optimal-humidity-temperature-greenhouse/#:~:text=Most%20common%20greenhouse%20crops%20require,most%20greenhouses%20around%20the%20world.

[9] N. Vercelletto, "What to know about greenhouse lighting," May 2021. [Online]. Available: https://www.happysprout.com/outdoor-living/greenhouse-grow-lights-lighting-requirements/

[10] "Led grow light spectrum guide," Sep 2021. [Online]. Available: https://lightsciencetech.com/led-grow-light-spectrum-guide/

[11] [Online]. Available: https://www.uctronics.com/download/cam_module/OV2640DS.pdf

[12] "Esp32," Mar 2023. [Online]. Available: https://en.wikipedia.org/wiki/ESP32

[13] J. Pan and B. Guo, "Effects of light intensity on the growth, photosynthetic characteristics, and flavonoid content of epimedium pseudowushanense b.l.guo,"

Nov 2016. [Online]. Available: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6273461/

[14] "About par, ppf, and ppfd," Feb 2022. [Online]. Available: https://fluence.science/science-articles/horticulture-lighting-metrics/?_gl=1%2A1bpe2df%2A_up%2AMQ..%2A_ga%2AMTg3MTE5NDc0OC4xNjc5NDIxOTMz%2A_ga_LEDTVYF9SN%2AMTY3OTQyMTkzMi4xLjAuMTY3OTQyMTkzMi4wLjAuMA..#tab-par

[15] Dustin, "Light recommendations: Ppfd (par) for orchids and houseplants," Apr 2021. [Online]. Available: https://herebutnot.com/light-recommendations-ppfd-par-for-orchids-and-houseplants/

[16] "Luminous efficiency function," Dec 2022. [Online]. Available: https://en.wikipedia.org/wiki/Luminous_efficiency_function

[17] [Online]. Available: https://docs.agi32.com/AGi32/Content/adding_calculation_points/PPFD_Concepts.htm#:~:text=The%20mathematical%20basis%20for%20the%20calculation%20of%20PPFD%3A&text=%CE%BB%20%3D%20wavelength%20in%20meters.&text=Given%20an%20illuminance%20value%20(lux,for%20the%20given%20light%20source.

[18] IEEE. (2016) "IEEE Code of Ethics". [Online]. Available: https://www.ieee.org/about/corporate/governance/p7-8.html

[19] [Online]. Available: https://www.espressif.com/sites/default/files/documentation/esp32-wroom-32e_esp32-wroom-32ue_datasheet_en.pdf

[20] "What to know about lighting cannabis from a former greenhouse cultivator," Dec 2021. [Online]. Available: https://gpnmag.com/article/what-to-know-about-lighting-cannabis-from-a-former-greenhouse-cultivator/#:~:text=Cannabis%20is%20one%20of%20the,daily%20light%20integral%20(DLI).