College of Engineering and Computer Science

EEE 193B/CPE 191
Senior Design Project II

End of Project Documentation

Team 5
Team Members: **Raj Bhatt, Thinh 'Jay' Nguyen, Gabriela Estrada, Xiomara Valdivia**
Instructors: **Douglas Thomas, Russ Tatro**
Date: **April 27, 2020**

TABLE OF CONTENTS

TABLE OF FIGURES

TABLE OF TABLES

Elevator Pitch

Our Smart Greenhouse is a user-friendly, automated, portable, greenhouse system that allows all to easily grow and control aspects of growing crops.

Executive Summary

Our team and our Smart Greenhouse design take a deeper look at climate change which is causing a decrease in the availability of agricultural land. Climate change has resulted in an increase in extreme weather events such as heatwaves, droughts, wildfires, and abnormal rainfall. These problems have affected many individuals' and communities' food security around the world, and it has brought forth concern for future generations. Also, due to an exponential increase in the population growth in recent years, cities have drastically expanded resulting in effects on the agriculture sector and farming land. An engineered solution is needed, and our Smart Greenhouse can make a positive impact on the effects caused by climate change and it can also provide food security for future generations.

To create a positive impact on climate change and food security we have developed a user-friendly and automated greenhouse that is portable. By making it a portable, automated, greenhouse we can reach users who have limited space. Therefore, individuals can help make a positive impact on the planet by using our greenhouse and have access to organic food in various agricultural conditions. To achieve high success in growing plants, our greenhouse was developed to allow for tight environmental constraints which have helped us achieve great results for growing different types of plants and it also functions to support the growth of these plants. The components which make the greenhouse smart include automation for watering the plants, automation for the temperature and humidity inside the system automated, and automation for light control. All these automated and additional manual controls are delivered to the user with a user-friendly mobile phone application that can be downloaded from the user's phone app store.

We achieved the best organization in implementing the greenhouse by dividing the entire project into three main features which were hardware, software, and structure. The hardware feature consists of sensors, power management, circuitry, and more. The goal of this feature is to ensure every hardware component works well in the system with enough power supplied. Jay primarily completed this feature with additional assistance from Raj on the implementation of a couple circuits. The software feature was divided into two main components, programming the Raspberry Pi and creating the graphical user interface (GUI). Gabriela focused on programming the Raspberry Pi which incorporated reading the sensor data and implementing the automated features. Xiomara focused on the GUI mobile phone application which consisted of the design of the GUI, displaying the data, and the manual user control for a few of the hardware components. This feature allows the user to access data from sensors, monitor the greenhouse remotely and automate the greenhouse wirelessly. Raj built the structure which incorporates the greenhouse foundation, wood framing, the outside tarp for coverage, irrigation system, and a few other components. The detailed information about our societal problem, each feature, component, and our project implementation are discussed in this report.

***Abstract* — Climate change has created numerous negative effects on all inhabitants of the planet by affecting the land around the world. The agriculture industry has severely suffered from the effects of climate change and it has introduced the significant problem of food security around the globe. Our solution for this problem is a Smart Greenhouse that is user-friendly, automated, and portable that should essentially provide individuals access to nutritional food nearby. The Smart Greenhouse comprises three main elements which are the structure, hardware, and software. The entire system is automated with features such as auto irrigation, temperature control, auto lighting system, and a graphical user interface (GUI). Each automation feature was implemented with different sensors, hardware parts, and the software to make the entire system efficient. The GUI provides the users with the environmental conditions inside the greenhouse, and other than the automation, it also provides the ability to control the features manually. With this design, our team wants to make an impact towards the food security issue by providing access to individuals to grow their own food in an efficient way and with ease. The process from start to finish and all aspects associated with the Smart Greenhouse will be discussed in depth including the societal problem, design idea, funding, work breakdown structure, risk assessment, deployable prototype status as well as the marketability forecast. The Smart Greenhouse project is a complete product that comes with efficiency and reliability.**

Keyword Index* — *Smart Greenhouse, hardware, Raspberry Pi, automation, structure, software, GUI, sensor

I. Introduction

Agricultural degradation has been caused by the effects of climate change such as heatwaves, droughts, rainfall pattern changes, increased water sea levels, and more. Agriculture is highly reliant on temperature and environment and these effects on weather and climate have affected the growth rate and health of crops that feed communities. These effects have impacted accessibility to food, known as food security, on the entire planet but have a more significant impact on the countries that do not have proper economies set up. Food security is an important discussion currently at hand for the world population. Food security is defined as the ability to access adequate food that provides enough nutrition for a healthy life [2]. The food security around the globe has been affected by multiple factors in past years. Inhabitants of developing and third-world countries do not have high food security and they can and are affected by additional issues such as the "limitation of suitable land, the deterioration of land quality, and environmental degradation" [3]. As these effects of climate change begin to worsen, we can also expect food security to worsen as well. In addition, it is estimated that the demand for food will rise by 70 to 100 percent by 2050 since we can expect the population around the globe to grow [4].

The decrease of food security for the population of Earth has resulted in a solution that must be researched and implemented to reduce starvation, avoid loss of lives and health problems that have been caused thus far. By developing a Smart Greenhouse, our team will be able to positively help communities that have very low food security and those who struggle to grow crops on their land. Our design is an automated system that behaves as a small-scale portable greenhouse. This greenhouse will be easily accessible and user-friendly so that an individual with little to no knowledge will be able to easily use the system and grow their own fruits or vegetables nearby.

This deployable prototype consists of three major features: hardware, software, and the structure. All these components have consisted of working countless hours in the Fall 2019 and Spring 2020 semester. The hardware consists of a microcontroller, a relay switch to control power to components, sensors, power management, Analog to Digital Converters, voltage regulators, and more. The software feature involved programming the Raspberry Pi in Python to read the sensor data and to autonomously control the hardware devices for the crops to flourish. The software also included the Graphical User Interface (GUI) implemented as a phone application that allows the user to view the statistics of their greenhouse environment and control certain aspects of their greenhouse system. The structure consists of various components such as the wood framing, the portability aspect, and the electrical box and wiring inside of the structure itself.

For this Smart Greenhouse design, we had to consider how our team was planning to purchase all the components needed to complete the design. This consisted of the team splitting the costs of the entire project. We also surpassed various project milestones as we reached our goal at the end of the semester. To organize and plan the entirety of the project, we created a Work Breakdown Structure (WBS) to assign all our team members tasks and features so that our deployable prototype would be completed on time. Additionally, we assessed the various risks that could occur while working on this project. Lastly, we analyze the marketability of our Smart Greenhouse product and discuss in this report the changes that would be needed to realistically sell and advertise the product itself.

II. Societal Problem

During Fall 2019, our societal problem was primarily focused on the effects of global warming. Our team mainly focused on the science behind global warming and what was causing it. The main cause of global warming has been the human expansion of the "greenhouse effect". The "greenhouse effect" is caused by greenhouse gases that are trapped in the atmosphere and they "absorb some of the outgoing radiation of Earth and re-radiate it back towards the surface" [5]. Global warming, being a societal problem that the world is facing today, was too broad of an issue for our project. We had the intention of helping make an impact towards global warming, but we understood later, in Spring 2020, that we could narrow our societal problem so that we could directly make an impact.

After conducting more research in Spring 2020 on the topic of global warming, agriculture, and food security in correlation to our project, the Smart Greenhouse, our team had a better understanding as to how our design tackles the problem. The topic we better understand now is climate change, agriculture, and food security. We have learned that our design is less directly related to global warming, although it is still a part of the problem. With this in mind, we have determined that the core issue, which our technical design is set to help make a difference, is the detrimental effects that agriculture is facing and how those effects are causing a greater concern for food security. Climate change has been a leading factor that has caused a vast impact on the agriculture sector. As stated by the United States Environmental Protection Agency, "climate change can disrupt food availability, reduce access to food, and affect food quality. Projected increases in temperatures, changes in precipitation patterns, changes in extreme weather events, and reductions in water availability may all result in reduced agricultural productivity" [6]. The effects that agriculture have been linearly affecting food security. Due to the decrease of land quality and ability to grow crops, the food security also has decreased. We provide in detail how both agriculture and food security are directly affected.

The agriculture sector has been primarily affected due to the extreme weather challenges caused by climate change. Climate change has many detrimental effects to individuals, society, animals, and the land that produces our crops. It encompasses a big part of everyday life. The Intergovernmental Panel on Climate Change (IPCC), a group that assesses the science related to climate change, found that "risks associated with extreme weather events continue to increase as the global mean temperature rises" [7]. The IPCC group has done several assessments across the globe to see how climate change has impacted individual regions which cause a negative impact to the agriculture of these regions. Through these assessments they have found that Europe is currently experiencing an increasing number of natural hazards, "notably extended periods of high temperatures, droughts, and extreme rainfalls" [7] which are causing losses in human lives and environmental damage. In Malaysia, the sea level is rising at an extremely fast rate and through studies that IPCC conducted, they "found that in the 21st century, the sea level is expected to rise from 18 to 66 cm" [7]. In the United States, snowmelt has been an integral part of the regional hydrologic cycle, but with climate change there has been changes "in the level of winter snowfall and snowmelt and the extent of growing season rainfall amounts and intensities" [8] due to the decrease of temperatures during the winter. These extreme fluctuations between heat, rainfall, and snowmelt has caused detrimental effects to the land of these specific regions. Due to the drastic changes, the land does not have a stable climate to prosper and allow for crops to grow sufficiently.

While there have been studies of specific regions, there are also studies of the extreme weather changes that have generally been changing for the past several years. As mentioned previously, these extreme weather events include heat waves, droughts, floods, and sea levels rising to name a few. The Center for Climate and Energy Solutions has recorded these weather extreme changes for the last several years and have demonstrated how climate change is truly impacting the world and the environment fully. To begin, heat waves are an increasing concern because hot days are becoming hotter and much more frequent. There has been a study conducted and it shows “that the annual number of days with a heat index above 100 degrees Fahrenheit will double, and days with a heat index above 105 Fahrenheits will triple” [9]. As shown in Fig. 1, it shows the projected change in the number of days that will be above 90 degrees Fahrenheit in the mid-21st century. It is alarming to see almost the entire United States being impacted by this increase in temperature. With extreme high temperatures, crops that need to grow in colder weather are not able to fully prosper because the temperatures are not in the climate profile that they require. Crops such as lettuce, radishes, and broccoli, to name a few are directly getting impacted and these subsectors of agriculture run a risk of completely losing the ability of being able to grow.

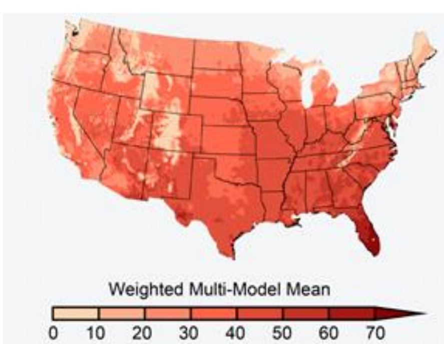


Fig. 1. Project Temperature Change [9]

Droughts have also been lasting longer due to the rise of temperatures. There have been record highs of droughts in recent years with the most recent year being 2012. As shown in Fig. 2, there are still many areas that are abnormally dry demonstrated by the yellow highlights on the map. Through these daily records, scientists have been able to predict that areas that are already dry will become drier. Crops of course need the resources from water to be able to effectively grow and produce properly. With the loss of water resources, agriculture begins to face many limitations.

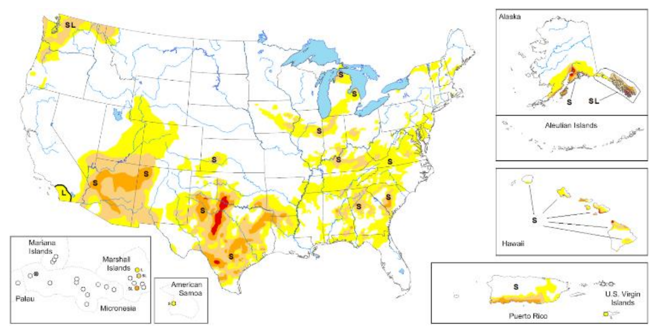


Fig. 2. United States Drought Monitor [9]

Due to hot, drier conditions, the risk for wildfires has greatly increased. Some areas are more affected than others, but as the temperature increases, the scope of areas getting affected gets wider. As studies have shown, for most of the western side of the United States, there is a projection that “an average annual 1 degree Celsius temperature increase would increase the median burned area per year as much as 600 percent in some types of forests” [9]. As shown in Fig. 3, it shows the percent increase in median annual area burned with a 1-degree Celsius increase in global average temperature. As shown, the western part of the US is truly impacted as more areas are highlighted in red. This in hand, directly affects the land because it deteriorates the soil due to the decrease of nutrients. With the loss of nutrients, crops need to overcompensate with what they do have, and in turn it becomes insufficient for a fully established crop.

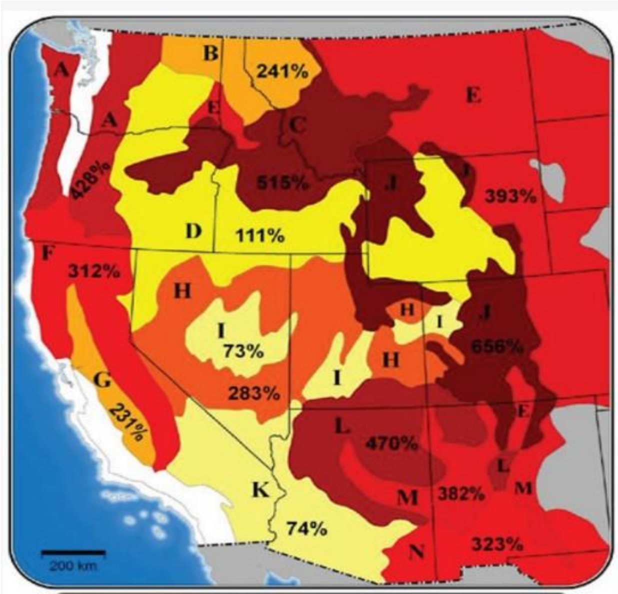


Fig. 3 Percent Increase for Area Burned [9]

Climate change has caused a great deal in precipitation rates. Since the air is becoming warmer, it can hold more water vapor causing more intense rainfalls. As shown in Fig. 4, it demonstrates the increase of heavy precipitation across the United States. This heavy precipitation has been a concern due to the increased levels of flooding. Flooding has become more prevalent and it brings tremendous downsides to communities that are not well equipped for this type of extreme weather. The increase of precipitation also correlates to the sea level rise which greatly impacts the increase of flooding as well. Studies have demonstrated that "flood frequencies will increase dramatically in many coastal areas by 2050" [5]. Similarly, to how having no water greatly impacts crops, having too much water can also cause the same level of impact if not more. With heavy precipitation rates, crops run the risk of getting "drowned" or destroyed. They also can get washed out and ultimately become a dead plant due to the excessive amount of water.

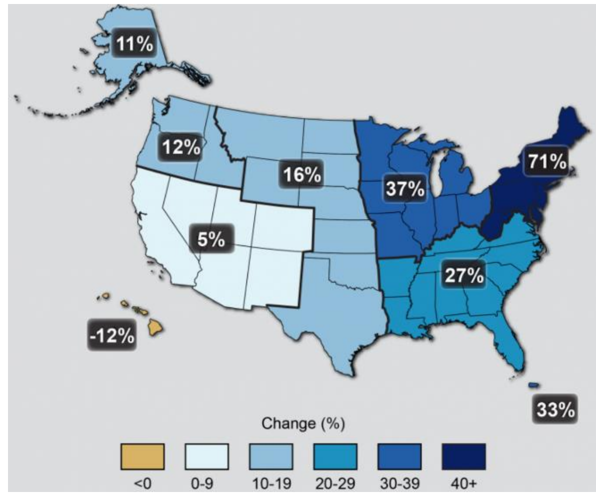


Fig. 4. Changes in Heavy Precipitation [9]

As mentioned amongst each extreme weather event, agriculture is greatly impacted in some form due to each individual natural disaster. Agriculture is highly reliant on specific weather conditions and with the increase of temperatures, there continues to be many difficulties set forth for this sector. While moderate warming and more carbon dioxide in the atmosphere may increase global food production, "severe temperature increases, changing precipitation rates, and other climate change stressors will have the opposite effect" [10]. As shown in Fig. 5, it demonstrates how the weather extremes have impacted crops of corn throughout the years. Each low peak was caused by weather extremes such as droughts and floods. This only demonstrates the impact on one type of crop, but a similar impact is seen on various, if not most, types of crops.

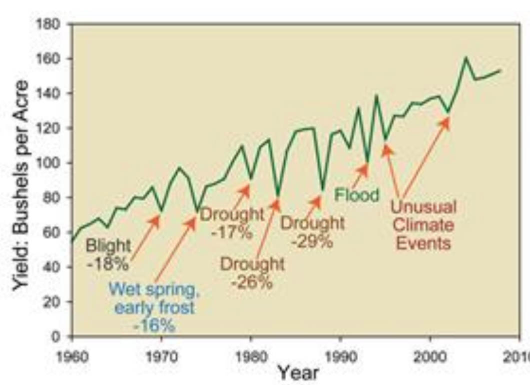


Fig. 5. Corn Yield Impacted by Climate Change [11]

There are various types of specific climate change stressors that truly impact agriculture. The leading concern is the rising temperatures and increased climate variability. Differences in temperatures have different effects on different regional areas. For example, in "semiarid and tropical areas, a temperature increase as small as 1-2 degrees Fahrenheit can lower crop fields" [10]. The increase in heat stressors can also cause an abundance of damaged crops in many regions. The change in precipitation rates is another climate change stressor, as mentioned previously. Periods of heavy rain can be just as harmful as periods of severe drought to crops. In open-air farming methods, crops that are exposed to heavy rain can cause "both soil washout and physical damage to the crops" [12]. If current trends continue, "droughts and irregular rainfall could reduce rain-fed agricultural yields by up to 50% by the year 2020" [10].

Soil vulnerability and increase of pests and crop diseases are also concerns that have been affecting agriculture due to the increasing temperatures. Climate change is affecting the overall quality of soil. Because the weather is getting warmer, it is causing the soil to become drier which affects further the crop yield. Having dry soil can then cause erosion which can be detrimental to crops causing them to die off. The increase of pests and crop diseases have correlated to the increase of warmer temperatures. Having warmer temperatures even in the winter "allows for pests to reproduce multiple times per year, which intensifies the threat they pose to agriculture" [10] not only that but the warmer weather also "encourages the growth of mold, fungi, and bacteria that are harmful to plants" [10]. Fig. 6 demonstrates the increase of crop yield loss due to increase of pests and crop diseases throughout the years for three different types of crops which are wheat, rice, and maize. The purple increase shows the crop yield losses due to insect metabolic activity and the orange and green increases show crop yield loss due to bacteria growth. In open-air environments, crops are exposed to these pests and diseases because they are not protected under these drastic climate changes.

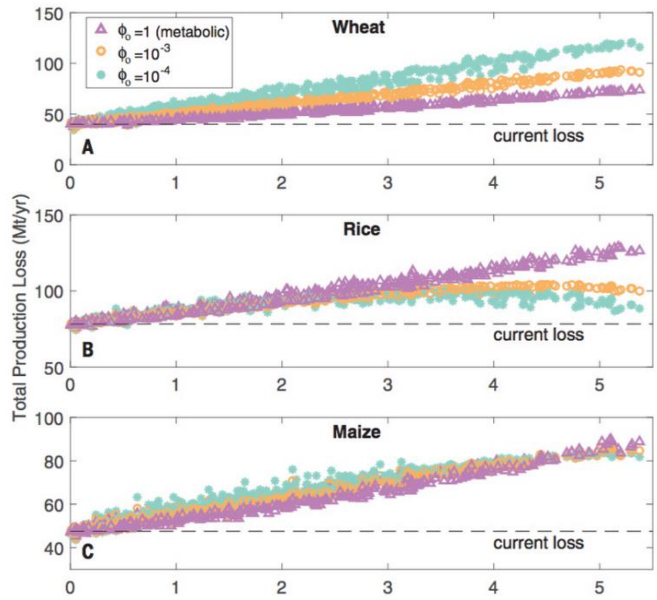


Fig. 6. Crop Yield Losses [14]

Lastly, one of the final climate change stressors that greatly affect agriculture are elevated atmospheric carbon dioxide levels. In general, plants do benefit from higher amounts of carbon dioxide since it helps stimulate photosynthesis, however, when not controlled, especially in open-air environments, it could cause negative effects to crops. The increase of carbon dioxide is better impacted under controlled environments for plants, since it can be managed how much carbon dioxide is used.

With today's climate change impact, "the degradation of land and water resources as a result of climate change can lead to a growing food deficit for the growing population of the planet, which will negatively affect food security" [13]. Not only are temperatures increasing at a fast rate, but so is the population and with that comes the demand for food. The negative impacts on agriculture have been having a big impact on food security, especially on countries that are not as highly developed. This impact on agriculture has greatly peaked the prices on food causing an impact on access to food as well. According to the USDA, in 2017, the rate of food insecurity is 26% of the population worldwide, which means almost 2 billion people are food insecure [16]. Countries such as the United States, which is highly developed, is also beginning to face the effects of loss in food security. The USDA has run multiple surveys through the recent years to gather the statistics of families that are undergoing food insecurities. There are multiple statistical points from the United States that demonstrated how alarmingly fast the concern for food insecurity is increasing. To begin, about “three-fourths [of the survey pool] who had very low food security recurred in 3 or more months of the year” [15]. Secondly, “for about one-fourth of food-insecure households and one-third of those with very low food security, the occurrence was frequent or chronic” [15]. Third, they found that “on average, households that were food insecure at some time during the year were food insecure in 7 months during the year” [15]. Lastly, they found “on average, households with very low food security at some time during the year experienced it in 7 months during the year and in 1 to 7 days in each of those months” [15]. More and more households are facing these challenges and are having more difficulty sustaining themselves with nutritious food.

Climate change is more “likely to affect food security at the global, regional, and local level” [6]. Not only does it increase food prices, but it also can interrupt food delivery which becomes a bigger concern when people do not even have the option to purchase food. Having limits on food delivery and transportation, brings in the issue of safety and quality getting negatively affected. The United States highly relies on water transportation for their crops, and with a weather extreme event “if it affects the waterways, there are few, if any, alternate pathways for transport” [6]. Overall, “impacts to the global food supply concern the United States because food shortages can cause humanitarian crises and national security concerns” [6].

We want to provide a useful solution to assist the problem that crops are getting negatively impacted and it is hindering the ability for people to have access to food due to the increase of food prices. We want to develop a solution that tackles both aspects of this problem by providing a better way of growing food under a controlled environment and having easy access to food from your own home. Also, we want to make it manageable for people of all ages to be able to take care of their crops with absolutely no skills or experience required. We believe that by solving the issue from an individual standpoint, it can make a bigger impact to society. It would provide better access to food, instead of relying on others to provide food access. To implement these factors into our solution, we want to build an automated, portable, and greenhouse-like system that will allow anyone to grow their plants from their backyard while having a user interface to control all aspects of plant growing. By having individuals grow their own crops in an efficient and sustainable way, it leads towards the right direction of making a change towards agriculture impact and food security.

III. Design Idea

To achieve promising results and have a positive effect on communities that are facing challenges with food security, we implemented a system that can provide ideal conditions to grow crops with no need of having knowledge about growing crops. To achieve this, our design idea consists of making a portable, automated, and user-friendly greenhouse. The greenhouse consists of multiple sensors such as temperature sensor, humidity sensor, moisture sensor, light level sensor, and water level sensor that provides cohesive information about the environment in the greenhouse. It consists of different hardware components as well such as fans, heater, irrigation system and lighting system to allow for better control of the environment. With this system, it also has a user-friendly interface that allows users to be able to remotely make changes into their greenhouse. We have named this design idea the Smart Greenhouse. Within this design we were able to divide it into three specific subdivisions which were the hardware, the software (Raspberry Pi and GUI), and the structure. We had an initial design idea in Fall 2019 but made improvements in Spring 2020. Below we will discuss each specific feature set and what each entailed during both semesters.

A. Fall 2019

Fall 2019 consisted of our initial prototype for our project. We were very optimistic about the ideas we had for our project and we were aiming for high expectations. The first semester consisted of working on the base of each feature set. For the structure, we wanted to make sure the framing was done. For the hardware, we wanted to ensure we would have enough power across all the components. For the Raspberry Pi, we wanted to confirm the sensors were working properly. For the GUI, we wanted to design the displays of each screen. All of this would allow us to ramp up during the second semester and focus on the technicalities of each feature set. Below we describe in depth, what was completed for the design idea of each feature set during Fall 2019.

1) Hardware

Certain components of the hardware had to be completed and researched to start coding the components in the Fall 2019 semester. Although Gabriela's main feature was the software, she also worked on the hardware feature to get the software feature working correctly since working with the microcontroller and programming it involved knowledge of how the sensors work. The sensors were very pertinent to our project because they detected the environmental aspects inside the greenhouse for the plants inside to thrive. For the microcontroller, Gabriela decided to use a Raspberry Pi B+ due to the fact that a team member already had one so this would not impact our budget as much. Gabriela first had to get acquainted with the wiring and the types of sensors to be used to detect the environment inside of the greenhouse. The different sensors that Gabriela read data from were a temperature and humidity sensor, a soil moisture sensor, and a light level sensor.

The temperature and humidity sensor used for the Smart Greenhouse was an Adafruit's DHT11 sensor, as seen in Fig. 7. This sensor uses a capacitive humidity sensor and a thermistor to measure the surrounding air and returns a digital signal. It operates at 3.3 to 5 volts and has a 2.5 mA max current while requesting data. Data can only be gathered from this sensor every 2 seconds. Its range is 20-80% humidity readings with 5% accuracy and is 0-50°C temperature readings ±2°C accuracy. It can handle no more than a 1 Hz sampling rate (once every second) [17].

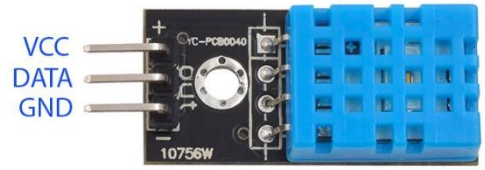


Fig. 7. Adafruit DHT11 Sensor [17]

Another sensor Gabriela wired and configured in the Fall 2019 semester was a YL-69 soil moisture sensor which works as a hygrometer as seen in Fig. 8. This sensor consists of two key pieces, the electronic board and the probes that detect the water content in the soil.

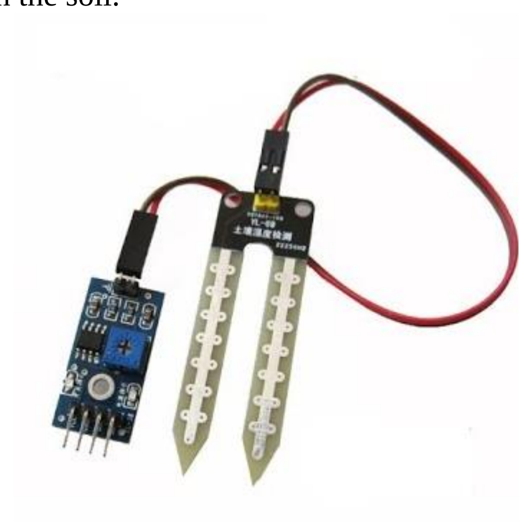


Fig. 8. YL-69 Sensor [18]

The soil moisture sensor has a built-in potentiometer for the sensitivity of the digital output (DO), a power light emitting diode (LED), , and a digital output LED, as seen in Fig. 9. The two probes allow current to pass through the soil and it checks the resistances value to measure the moisture. When there is more water, the soil will conduct more electricity therefore there is less resistance. Dry soil cannot conduct electricity well so when there is less water the soil conducts less electricity causing more resistance, so the moisture level is lower [18]. The operating voltage for this sensor is 3.3 to 5 volts and it has a dual output mode of DO or analog if more accuracy is needed. Additionally, in the Fall semester, Gabriela tested the digital and analog options for the sensor. The DO signal can be LOW

(0) or HIGH (1) depending on the water content and if the predefined threshold value was exceeded, the module outputted LOW, otherwise, it outputted HIGH. However, in the end, we decided to use analog readings since it is more accurate. For this sensor's analog readings, we received values from 0 to 1023.

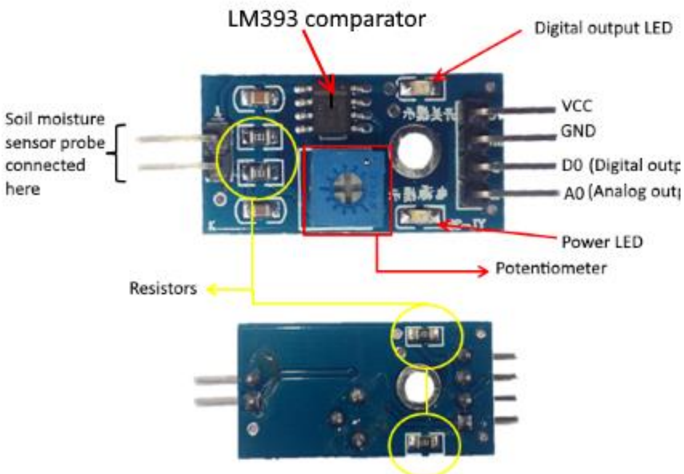


Fig. 9. YL-69 Sensor Components [18]

The last sensor Gabriela configured and wired was a DROK 200033 light level sensor as seen Fig. 10. This sensor has a direct current (DC)operating voltage of 3.3 to 5 volts and detects light intensity in a directional manner. This sensor can detect brightness and ambient light intensity. It also includes an adjustable intensity for the digital output that can be changed by using the potentiometer on the sensor. It has an output format of DO or analog, similar to the soil moisture sensor and also has an output of 0 to 1023.

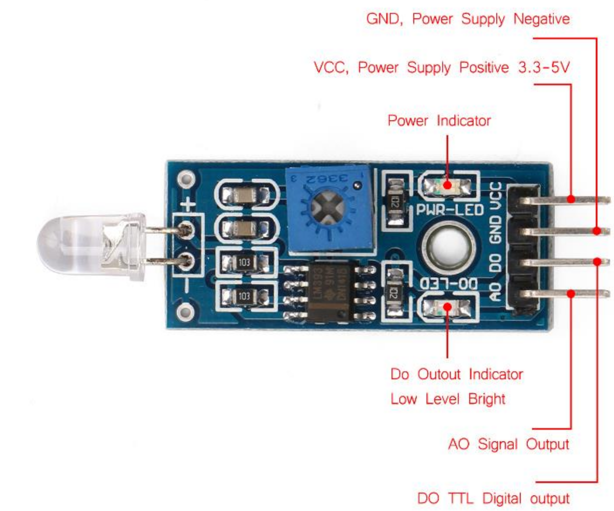


Fig. 10. DROK 200033 Light Level Sensor [19]

Since we wanted as much accuracy as possible, we configured the soil moisture sensor and the light level sensor to output analog data. However, the Raspberry Pi cannot take in analog readings, so we had to convert the analog data back to digital. Gabriela and Raj decided to implement an MCP3008 chip as an Analog-to-Digital Converter (ADC), as seen in Fig. 11. and Fig. 12. To implement an ADC and make it more efficient than a breadboard, Raj soldered a Printed Circuit Board (PCB) with the ADC. Raj implemented an 8-channel ADC by using only one MCP3008 chip. During the Fall semester, we only needed 4-channels to test the components and digital values.

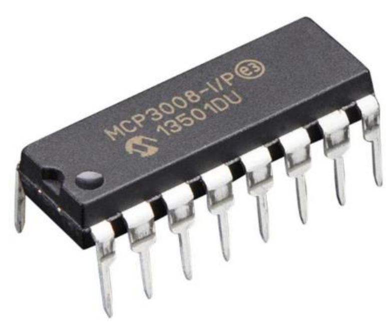


Fig. 11. MCP 3008 ADC [20]

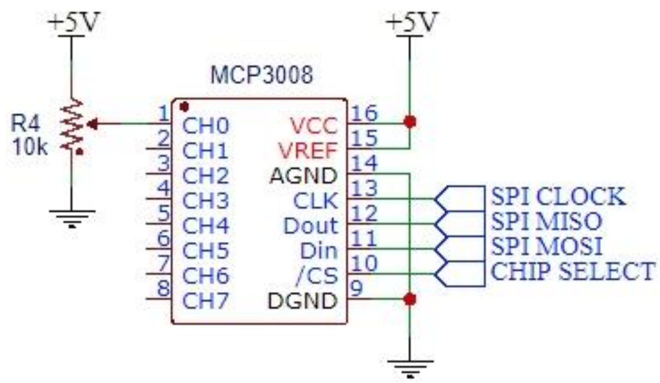


Fig. 12. MCP 3008 ADC Schematic [21]

Another hardware component that Gabriela set up for the hardware was a liquid crystal display (LCD) to output the sensor readings to show the user the data. We used a Sunfounder 20x4 LCD screen as shown in Fig. 13.

Fig. 13. SunFounder 20x4 LCD screen [22]

The pin layout for the LCD screen can also be seen in Fig. 14. For wiring this LCD screen, we had the option of using 4-bit mode or 8-bit mode. We decided to wire it according to 4-bit mode because it required fewer data pins to be used. However, the 4-bit mode can generate latency, but we did not notice any issues and we were satisfied with having to use only 4 GPIO pins.

PIN NO	Symbol	Fuction
1	VSS	GND
2	VDD	+5V
3	V0	Contrast adjustment
4	RS	H/L Register select signal
5	R/W	H/L Read/Write signal
6	E	H/L Enable signal
7	DB0	H/L Data bus line
8	DB1	H/L Data bus line
9	DB2	H/L Data bus line
10	DB3	H/L Data bus line
11	DB4	H/L Data bus line
12	DB5	H/L Data bus line
13	DB6	H/L Data bus line
14	DB7	H/L Data bus line
15	A	+4.2V for LED
16	K	Power supply for BKL(0V)

Fig. 14. LCD pin layout [11]

In conclusion for the Raspberry Pi hardware, Gabriela became acquainted with the wiring and the configurations of the sensors. The different sensors used were a temperature and humidity sensor, a soil moisture sensor, and a light level sensor. In addition, since some of these sensors required analog data to be outputted to be more accurate, Gabriela and Raj set up an ADC for the Raspberry Pi. Another component that was also connected to the Raspberry Pi was a 20x4 LCD screen to output the readings from the sensor. The culmination of these hardware components can be seen in Fig. 15. and the schematic with more details can be seen in Fig. B1.

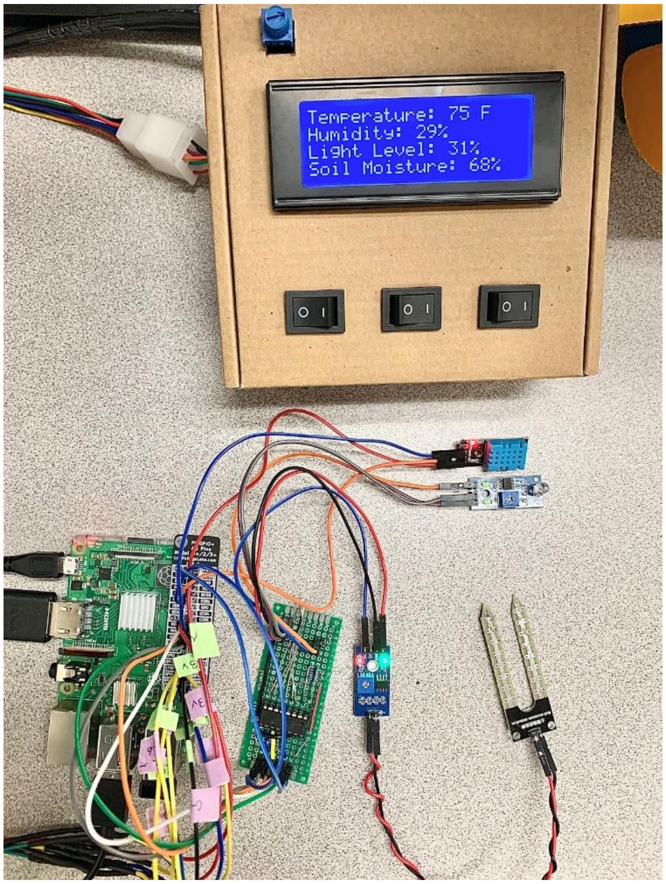


Fig. 15. Fall 2019 LCD [1]

In addition to the sensors and the ADC converter, Jay was responsible for the majority of the hardware components involved in the first prototype including the ventilation fans, growing LED strips, a water pump, a step-down voltage regulator, and a power supply.

The greenhouse's ventilation system is a set of 2 fans mounted on the structure with one facing-in the greenhouse as for air intake and one is facing-out as for air exhaust. Each fan is 120 x 120 x 25mm as seen in Fig. 16. It requires an input voltage of 12 V DC and draws the current of 0.25 A, which implies that the power consumption is 3 W. It has the rated revolutions per minute (RPM) of 1600 at maximum power [24]. The ventilation system was to help control the temperature as well as the humidity of the greenhouse.

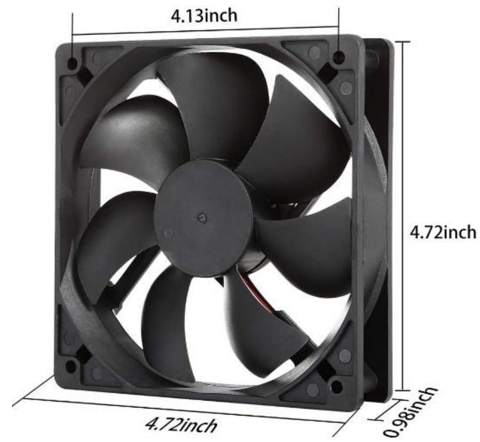


Fig. 16. 12V DC Ventilation Fans [24]

We used specific LED strips for plant growing because light is a very important aspect when growing crops. Plants need photosynthesis (a process of capturing light and transforming it to energy for the plant's activities). The LED growing strips come with a combination of red and blue light, which covers the wavelength range that is essential for photosynthesis. This LED light strip is a full spectrum surface mount device (SMD) 5050 LED with the red to blue ratio is 4:1 as shown in Fig. 17., and the beam angle is at 120 degrees. This light combination also promotes plant's growth and reproduction. The blue light has a wavelength range of 440nm to 460nm, which increases plant's growth, and the red light has a wavelength range of 640nm to 660nm, which not only helps plant grow but also promotes flowering [25]. There are two LED growing light strips inside the greenhouse. Each one is 35 inches long and is operating on the voltage of 12V DC with the rated current of 1.5A, so the total power consumed by the LED strips is 36W.

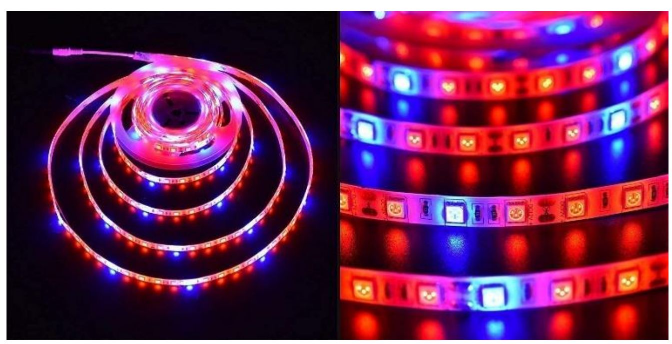


Fig. 17. LED Growing Lights [25]

The water pump we are using is from Mountain_Ark. It is 2.1in*1.9in*1.6in as shown in Fig. 18., and it has a weight of 220g. Even though it is a small pump, it can pump water with a rate of 63 gallon an hour or about one gallon a minute, which is more than enough for our Smart Greenhouse. One advantage of using this water pump is that it needs to be submerged inside the water tank, which will help lower the noise when the pump is running. The water pump is rated at the voltage of 12V DC with the power consumption of 3.6W [26].

Fig. 18. Mountain_Ark Water Pump [26]

As mentioned previously, we used Raspberry Pi as our microcontroller for this project. Raspberry Pi is a single board computer developed by the Raspberry Pi Foundation in the United Kingdom. The model we used for this project is shown in Fig. 19. below.

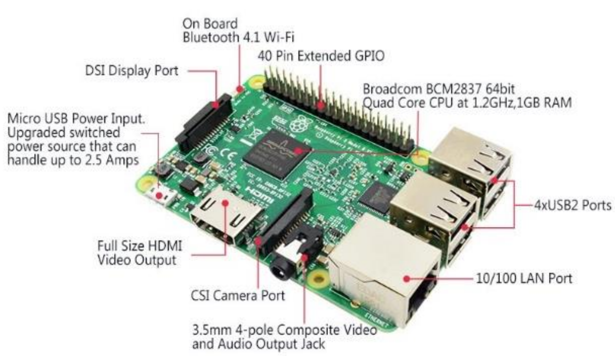


Fig. 19. Raspberry Pi 3B+ [27]

The Raspberry Pi 3B+ comes with the SoC Broadcom BCM2837BO and quad-core Cortex-A53 64-bit 1.4 GHz processor. It also has 1Gb of Random-Access Memory (RAM) that is capable of handling numerous activities. The Raspberry Pi has a SD card slot that can be used like a hard drive on a computer, a full-size HDMI, 4 USB 2.0 ports, an ethernet port, a CSI camera port, a DSI display port, and 40-pin General Purpose Input Output (GPIO) header [27]. With the number of GPIO pins provided, we can have the whole system working on one Raspberry Pi including reading sensors, controlling automation, and connecting with the GUI app. One significant advantage the Raspberry Pi offers is that it has built-in 2.4GHz and 5GHz IEEE 802.11.b/g/n/ac wireless LAN [27]. This enables the Raspberry Pi to go on the internet without additional hardware.

The power supply is not the most important component of a project, but our project could not have been completed without a power supply. In order to determine the power supply needed, Jay calculated the power consumption of the system. The power consumption of the system is determined by adding the rated power on every component involved. The rated power the components are given by the manufacturers. The Raspberry Pi 3B+ requires a power source of 5V 2.5A, meaning that its maximum power usage is 12.5W [27]. Each fan works at the voltage of 12V and draws the current of 250mA [24], so they consume the power of 6W . Each LED strip's rated values are 12V 1.5A, so they draw 36W of power [25]. Also at the voltage of 12V, each water pump draws 0.3A which results in a power of 3.6W [26]. The power consumption table is created as follow:

TABLE I
FALL 2019 PROTOTYPE POWER CALCULATION

Parts	**Current drawn (each)**	**Operating Voltage**	**Power consumption (Watts)**
Raspberry Pi 3B+	2.5A	5V	12.5
Fans (2)	0.25A	12V	6
LED strip (2)	1.5A	12V	36

Water pump	0.3A	12V	3.6
		Total	58.1

The total power consumption for the Fall 2019 prototype was 58.1W. In Fall 2019, we did not decide to use a heater, and we thought to use 2 separate Raspberry Pi's for the final prototype, so the total power consumption of the system was predicted to be around 71W. Jay decided to use the power supply with the rated value of 12V 3A 96W (Fig. 20.) with the dimensions of 190mmx110mmx55mm from Signcompolex. This power supply is simply a power transformer that converts the input voltage of 100-240V Alternating Current (AC) from the outlet to the output voltage of 12V DC with 8A maximum current [28]. Here in the US, the output value is 110V AC which is within the input range of the power supply.The power supply works well as expected with the output voltage measured to be constant at 12.02 V.

Fig. 20. 12V 96W Power Supply [28]

Besides the 12V components, there are others that are rated at 5V such as the Raspberry Pi. Since we already have a 12V power supply, we used the DROK buck converter as shown on Fig. 21.

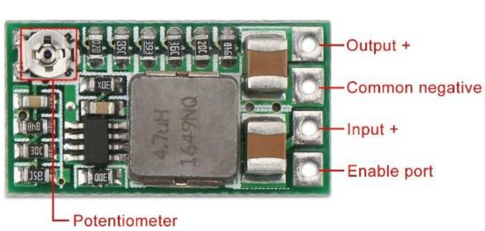


Fig. 21. DROK 5V 3A Buck Converter [29]

An advantage of this buck converter is that it is very small, so it can easily fit on a circuit board with other components. It also has the maximum efficiency of 97.5% according to the manufacturer [29]. High efficiency is very good because it implies less power loss. The input voltage ranges from 4.5V to 24V DC. The output voltage is adjustable by turning the potentiometer with the range of 0.8V to 17V DC. This buck converter also has fixed output of 1.8V, 2.5V, 3.3V, 5V, 9V, and 12V that we can choose on the back side. Jay chose the 5V output by soldering over its label accordingly to the given instruction, and the output voltage was tested to be 5.05V. One thing to note is that at 5V output, this buck converter can only supply a maximum current of 3A, so the load cannot pull more than 15W of power. The operating temperature is from -40℃ to +85℃ or -40℉ to +185℉ with the temperature rising range at full load of 40℃ [29]. From testing, this buck converter outputs a constant voltage at 5.09V in both conditions: with load and no load. The buck converter does not come with connectors, so to implement it to the circuit, Jay soldered 5V supply from the buck converter to the Raspberry Pi's micro USB cable.

2) Software

There were not many advances for the Raspberry Pi software side of the project in the Fall 2019 due to the fact that most of the hardware had to be ordered, completed, and wired correctly in order to test the Python code for all of the pieces of hardware. However, during this semester Gabriela was able to wire and code the Python for the reading and output of the sensor readings for the temperature and humidity sensor, the soil moisture sensor, and the light level sensor.

To read data from the DHT11 temperature and humidity sensor Gabriela used Adafruit's Adafruit_Python_DHT library. This library allowed for Gabriela to call various functions to read the data for the temperature and humidity sensor. For the soil moisture sensor and the light level, Gabriela used the Adafruit_Python_MCP3008 library. Since she was reading data that was converted by an ADC, the library was useful in interpreting the values to output from 0 to 1023. For the light sensor, Gabriela also used the MCP3008 library. A small pseudocode snippet of testing the light sensor can be seen in Fig. C1.

Gabriela also used the NumPy library during the Fall semester. NumPy is a library for Python that supports large multi-dimensional arrays and also supports high level mathematical functions. She used the NumPy library to take averages for the soil moisture sensor since she realized it was not very accurate. In addition, she used the library to convert readings into percentages since the soil moisture sensor and the light level sensor output values from 0 to 1023.

The last library Gabriela used was the Adafruit_Python_CharLCD library. This library allowed Gabriela to easily call functions to output the sensor data onto the LCD screen, as seen in Fig. 15. above.

During Fall 2019, the GUI was designed to be a website. We had initially decided to do a website because it would allow users to have access to their Smart Greenhouse from any type of device. Xiomara oversaw designing the GUI. Over the course of the first semester, she primarily focused on designing the layout of each page. She used HTML and CSS to do the design. The design of the website consisted of multiple pages that would allow users to be able to manually control different components of the Smart Greenhouse remotely, it would allow the users to automate the Smart Greenhouse accordingly to a plant profile, and the website would be an overall educational platform that users could learn about different plants. During the first semester, each page was primarily designed as a layout. While having a more basic format, it was going to be the base for the second semester when it would start getting connected to the Raspberry Pi and get improved overall.

The home page, as shown in Fig. 22., was going to be a welcome page for users to be informed about what the Smart Greenhouse was and how it worked. It also was going to provide statistics at a glance of each sensor reading including the temperature, humidity, soil moisture, and water level.

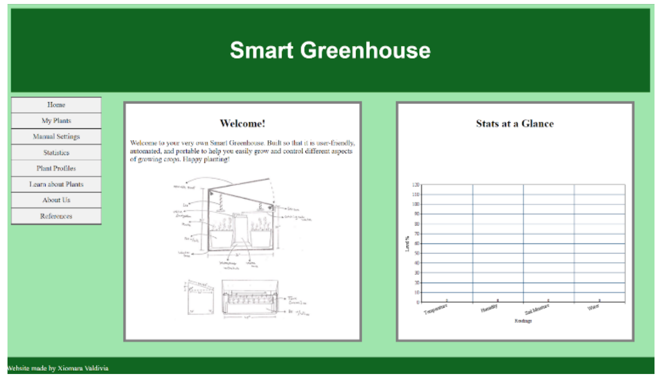


Fig. 22. Home Page [1]

The second page, which was called "My Plants", as shown in Fig. 23., was going to consist of the plants that the user actively was growing in their Smart Greenhouse. It would also provide information about the ideal conditions of each plant which would be considered as plant profiles. Ideally, we had planned that users would be able to automate their Smart Greenhouse accordingly to the plant that they would want. We of course reevaluated this in the second semester.

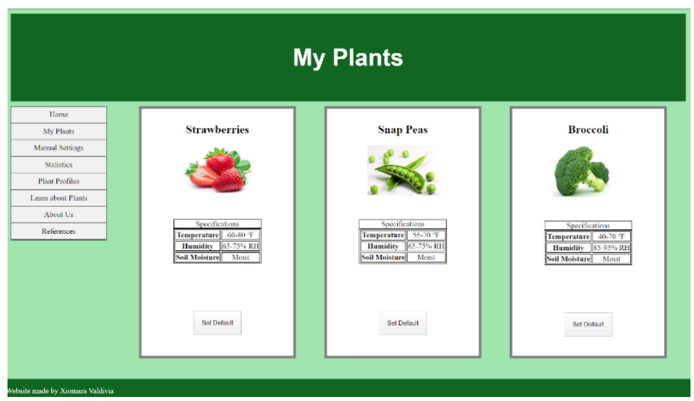


Fig. 23. Plants Page [1]

The third page, as shown in Fig. 24., consisted of the manual user controls. Here, users would be able to turn on and off the different hardware components such as the fans, lights, and water pump. Not only that but the user would be able to also set the temperature settings, the humidity settings, and the moisture settings of their preference. This is another feature that was reconsidered in the second semester.

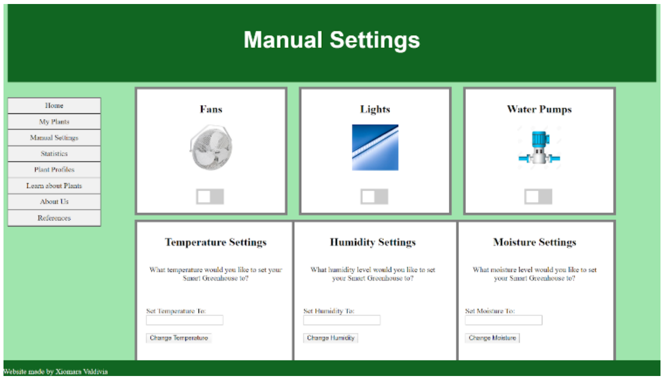


Fig. 24. Manual User Control [1]

The fourth page was going to consist of all the statistics that we would be keeping track of for the Smart Greenhouse. Ideally, it was going to consist of all the sensor readings, and it would provide the user with information on what would be needed for their plants. This was one of the layouts Xiomara did not complete as we were not actively receiving any data from the Raspberry Pi.

The fifth page, as shown in Fig. 25., consisted of other plants that users could add to their Smart Greenhouse if they were to add a new plant. Like the second page that had "My Plants", this page also had the plant profiles that provided the information of each individual plant's needs.

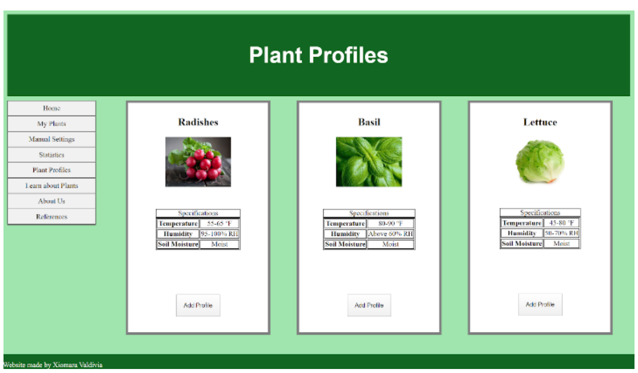


Fig. 25. Plant Profiles [1]

The sixth page, as shown in Fig. 26., was designed for educational purposes. It would provide facts about different plants and provide the needed living conditions. We believed that although our Smart Greenhouse was designed in a way that our users did not need to have too much knowledge about growing crops, it would still be good to provide the opportunity to be informed about crops for those who would want to learn about crops.

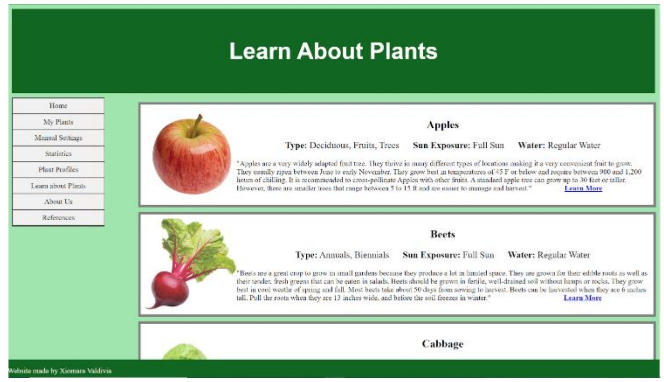


Fig. 26. Educational Page [1]

The last two pages of our website consisted of information about our team members and references to all our pictures. Overall, we did have a big plan for our GUI during the first semester. The term "user friendly" really geared the design of the website as we wanted it to allow users to have as much ease, but still have options to control the Smart Greenhouse to their liking. The design aspect of the GUI was much more manageable than the actual performance we were expecting. Xiomara did focus most of Fall 2019 on the design of the GUI with the connections to the Raspberry Pi in mind, however we decided making a drastic change to the GUI the second semester which we will discuss in the section below.

3) Structure

For the structure implementation, Raj divided the construction for the structure into two parts. For the Fall 2019 semester, the plan was to complete the basic framing and the base of the Greenhouse, so it was ready for the prototype showcase. This allowed Raj to plan the placement of all the components in the Spring semester. The entire greenhouse structure was made with wood as it is easy to cut and alter the size with only using a hand saw. By using the wood for the structure, Raj was also able to bring down the total weight for the greenhouse. All the wood used for the entire greenhouse structure was bought from Home Depot. Fig. 27. shows the wood used for the walls and the roof of the greenhouse structure and Fig. 28. shows the wood used for the base of the greenhouse structure.

Fig. 27. Framing Wood [30]

Fig. 28. Base wood [30]

Fig. 29. Desk wood [30]

Fig. 30 Greenhouse Wheels [30]

Fig. 29. shows the wood which was placed over the base frame to increase the stability of the structure and it also has a place to install all the pots, electric box, and the water tank. All the wood used in the framing is pressure treated which increases the quality of the wood. To be able to move the greenhouse structure around easily for future updates and for the prototype showcase, wheels were also installed during this semester. Fig. 30. shows the wheels used for the greenhouse. Each wheel is rated for 90 lbs. Thus, this allows the greenhouse to have a total weight of close to 360 lbs. This gave us a high margin for weight as we knew the structure would get heavy with the containers filled with soil, the water tank holding around 3 gallons of water, and the weight of the structure itself. Fig. 31. shows the basic frame design which Raj completed in the Fall semester.

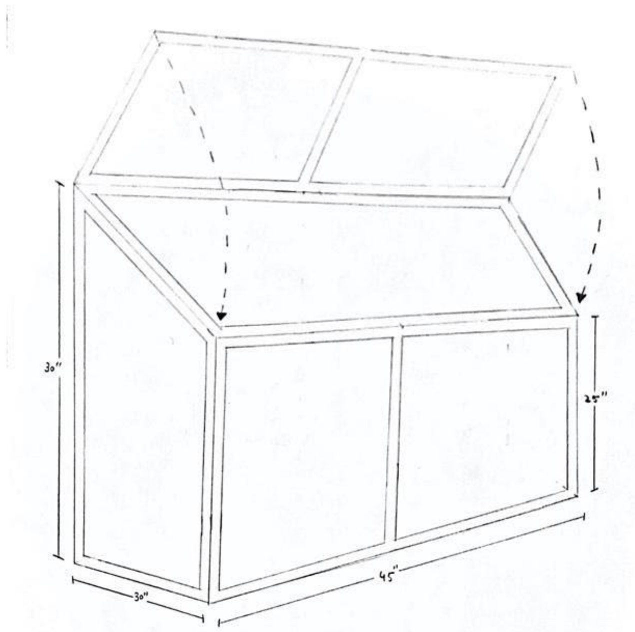


Fig. 31. Basic frame design [1]

The other part of the structure which Raj worked on was basic irrigation system implementation for testing. To complete this part, Raj laid out a simple water loop that pumped out water from a water tank and cycled back to the tank again. This way he was able to check the irrigation system connections and Jay was able to check the basic hardware implementation for the irrigation system.

B. Spring 2020

Spring 2020 was a semester of reflection, improvements, and completion. Before moving on with what we were working on during Fall 2019, our team made sure we were on the right track and made sure to improve on anything that was not sufficient to our project. We did the most extensive and important work during this semester. Each component needed to be up to par with our measurable metrics. The structure needed to be fully built and with great area management to allow hardware components to fit inside. The hardware needed to be efficiently wired. The automation needed to be completed. The GUI needed to be properly functioning. These tasks were very important for the completion of our project. Like the sections above, we will discuss what was completed for each feature set according to the design idea in Spring 2020.

1) Hardware

For the Spring 2020 prototype, we used the same temperature and humidity sensor, soil moisture sensors, Raspberry Pi, ventilation fans, water pump from the Fall 2019 prototype. For soil moisture sensors, since we implement 2 large and 1 small plant pots inside the greenhouse, we use 7 soil moisture sensors in total- 3 for each large pot, and 1 for the small pot. In addition to that, there are many new pieces of hardware involved in the Spring 2020 prototype including new light sensor, water level sensor, heater, electrical box cooling fans, 8-channel relay, ADC converter unit, humidifier, new power supply, and power supply output circuit.

One item that Gabriela had noticed in initial testing of the automation software was that the light sensor could not detect the LED plant grow lights when they were on. After research and discussion with the team, we had realized that this was due to the LEDs being a purple/blue hue and not white/yellow light that Gabriela had tested with previously. Due to this problem, we immediately had to order an Adafruit TSL2591 High Dynamic Range Digital Light Sensor, as seen in Fig. 32, that is a lot more sensitive to different ranges and colors of light. This sensor approximates human eye response and has a wide dynamic range of 1 to 600,000 Counts and has a Lux range of 188u Lux to 88,000 Lux input measurements. This sensor operates at 3.3 to 5 volts, like our other sensors. However, this sensor works by using the I2C pins on the Raspberry Pi.

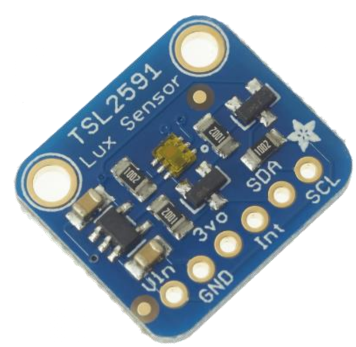


Fig. 32. TSL2591 Sensor [31]

To measure the water level in the tank, we use the ultrasonic sensor HC-SR04 shown in Fig. 33. HR-SC04 sensor’s operating voltage is 5V DC and it requires 15mA of current. It has a measure angle of 15 degree and the ranging distance from 2cm to 4m [32].

Fig. 33 Ultrasonic sensor [32]

Ultrasonic sensors measure distance from the sensor to the water surface by the principle of time of fly [33]. With the known speed of sound, the distance can be measured by the time of sending and time of receiving the ultrasonic wave which is illustrated in Fig. 34. From there, the water level can be calculated by subtracting the total distance from the sensor to the bottom by the distance measured from the sensor to the water surface.

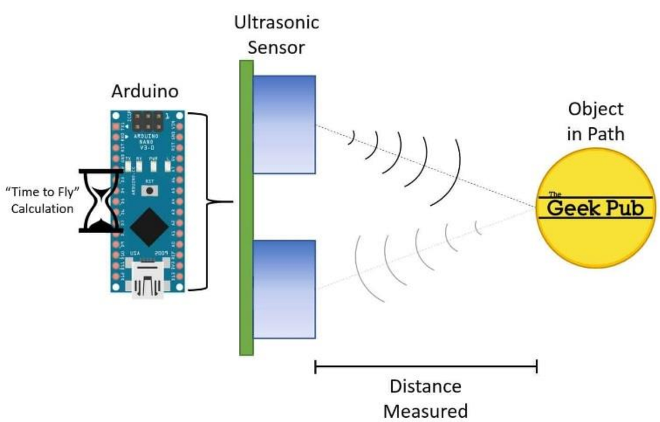


Fig. 34. Ultrasonic Sensor Working Principle [33]

The advantages of using an ultrasonic sensor is that it is very cheap and easy to program. The disadvantage is that the distance measured is not very accurate to the true value, so Jay created an estimation software algorithm for this sensor for better reading, which will be explained in detail in the software section.

For the Spring 2020 semester, Raj implemented a 16-channel ADC by using two MCP3008 chips. Changes in design and components were made after learning from the testing done in the Fall semester with an 8-channel ADC module. Jay and Raj implemented 3 PCB layer modules. These 3 layers hold the ADC module, analog components part and terminal connectors for Raspberry Pi to connect and communicate with the entire module. Fig. 35. shows a brief design implementation and Fig. 36. shows the 3 layered PCB design

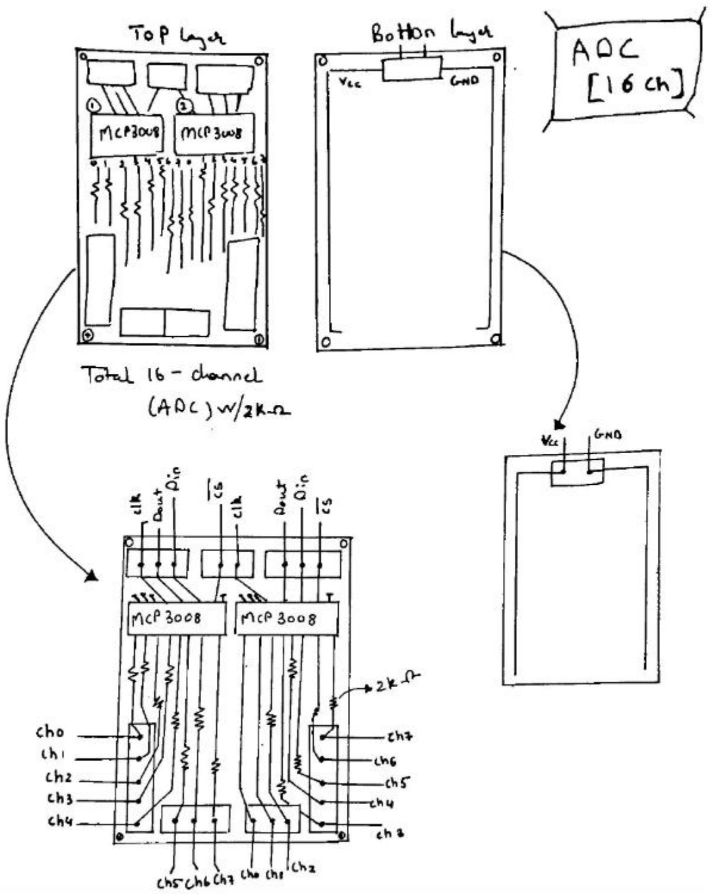


Fig. 35. 16-channel ADC [1]

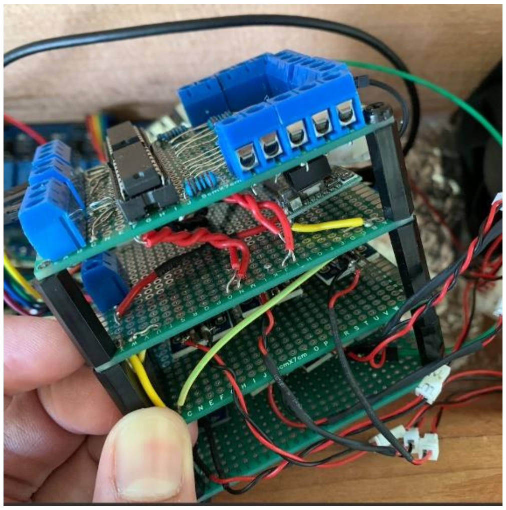


Fig. 36. 3 Layered PCB Design [1]

We decided to add a heater into our greenhouse so we can have a better temperature control feature. Since our power supply is outputting the voltage of 12V, Jay decided to use a 12V heater seen in Fig. 37., and it ended up to be a car heater with the rated values of 12V 100W with the size of 60mmx60mmx42mm [34]. Even though it is a low power heater, it can heat up an isolated space like our greenhouse very quickly. The heater is mounted to the frame of the greenhouse with its input air side facing outside and its output air pointing inside the greenhouse.

Fig. 37. 100W Heater [34]

In order to control and adjust the humidity in the greenhouse, we used a combination of the ventilation fans (to reduce humidity) and the humidifier to increase the humidity. The humidifier used is a USB humidifier from Aroma2go shown in Fig. 38 [35]. below:

Fig. 38. Humidifier [35]

The humidifier is an ultrasonic mist maker that has a piezo atomizer disc. This disc creates high frequency soundwaves on the atomizing surface, and the water particles are broken into fine mist of very small size droplets that expand into the surrounding to increase the humidity [36].

To keep the electrical components from heating up and prolonging their lifetime, we needed to have good air flow in the electrical box. For this reason, we use 2 cooling fans from GDSTime on two sides of the box. The cooling fan shown in Fig. 39. has the dimension of 80x80x25mm.

Fig. 39. GDSTime Cooling Fans [37]

These fans have the operating voltage range of 5.5V to 3.8V with the rated voltage of 12V DC and the rated current of 0.2mA so the maximum power consumption for each on is about 2.4W. The noise produced is rather low at 31 dBA with the rated speed of 2800 RPM [37].

A very important hardware component that determines our automation feature is the relays. The relay used is the SunFounder 8-channel relay shown in Fig. 40. Each relay needs a supply of 5V and 15-20mA of current to operate, and it supports control of up to 30V DC or 250V AC both at 10A maximum [38]. It comes with standard GPIO male connectors that can be controlled directly from the microcontroller.

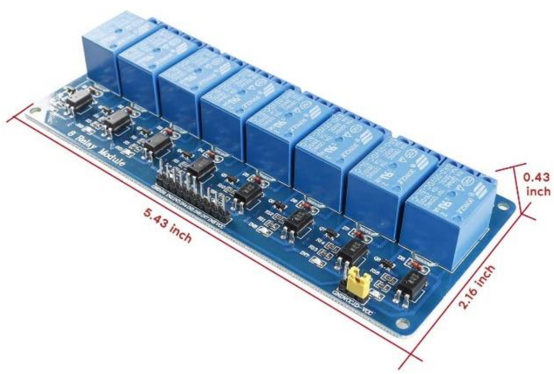


Fig. 40. SunFounder 8-channel Relay [38]

Each relay has 4 important parts: the electromagnet, the armature, the spring, and the electrical contacts as shown in Fig. 41. [39]. The electromagnet consists of an iron core wrapped by a coil of wires. When it is energized, it becomes a magnet. The armature is a movable magnetic strip. The spring is to pull the armature and disconnect the circuit. The electrical contacts include NO (Normally open- connected when the relay is activated), COM (connects to V+), and NC (Normally close- connected when the relay is not activated) ports.

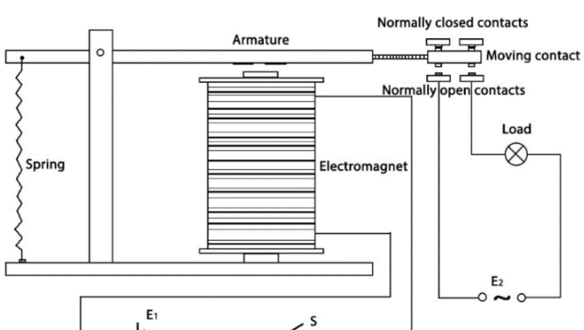


Fig. 41. Relay Diagram [39]

This 8-channel relay is active low, meaning that a LOW signal from the microcontroller will activate the relay. When the relay is activated, the electromagnet becomes a magnet that attracts the armature and the moving contact to connect the NO contact resulting in the load being energized. On the other hand, when the relay is not activated, the electromagnet does not attract the armature, and the spring will pull the armature back to the NC position and break the connection of the NO contact [38]. One thing to note about the relay when using Raspberry Pi to control is that the GPIO from Raspberry Pi is on 3.3V when it is HIGH while the relay is working on a 5V logic. To solve this problem, the relay should be supplied with an external 5V source to the JD-VCC pin, and the VCC pin is connected to the 3.3V pin on the Raspberry Pi.

For the Spring 2020 prototype, we added several hardware components as mentioned above. The power consumption of the system was calculated as in Table II.

TABLE III
POWER CONSUMPTION

Parts	**Maximum Current drawn (each)**	**Operating Voltage (each)**	**Power consumption (Watts)**
Raspberry Pi 3B+	2.5A	5V	12.5
Ventilation Fans (2)	250mA	12V	6
LED strip (2)	1500mA	12V	36
Water pump	300mA	12V	3.6
Heater	8.33A	12V	100
Relay (8 channels)	160mA	5V	0.8
Electrical box fans (2)	0.2A	12V	4.8
		Total	163.7

Initially, our team planned to use 2 heaters, so the power consumption of the system was about 263.7W. For power supply to be in good working condition and long lifetime, the system power consumption should be less than 80% of the supply's rated values. Jay picked the Menzo 360W power supply (shown in Fig. 42. below), so the system consumes about 75% of the supply's rated wattage. However, since the heater works really well to heat up the greenhouse, we decided to use only one heater, so the power consumption drops down to 163.7W as calculated in the table. We kept the same power supply because lower consumption compared to the supply power is even better for the lifetime of it.

Fig. 42. Menzo 12V 360W Power Supply [40]

The default output voltage of the power supply is 12V DC, but we can adjust the built-in potentiometer- a variable resistor, to adjust the output voltage level at will. This power supply takes input voltage of 110V AC from the outlet and transforms it to the desired DC voltage. There are 3 terminals for V+ and 3 terminals for V- as shown in Fig. 43. The rated values of this power supply is 12V 30A and 360W [40]. The output voltage of the power supply is measured to be constant at 12.09V when the system is running.

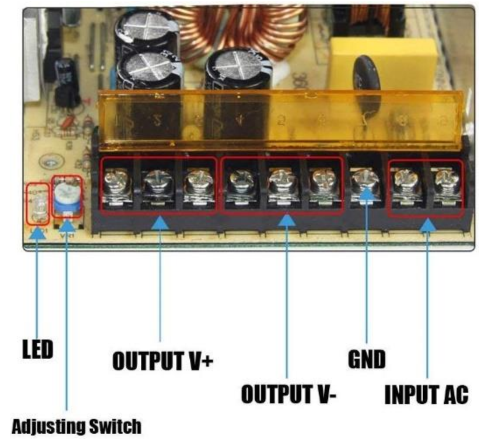


Fig. 43. Power supply's Terminals [40]

Jay designed and soldered the power supply distribution circuit board for the system to make the wiring easier. The board contains a 12V rail with 6 output terminals, 2 separate 5V rails with 2 terminals each, and a rail of ground with 8 terminals (Fig. 44.). The 12V terminals are connected to the relays and the 12V components: fans, water pumps, heater, and LED strip. There are two different 5V rails because each rail is powered by a voltage regulator, which only outputs 5V 3A maximum. One 5V rail supplies the Raspberry Pis, and the other one supplies other 5V components such as the 8-channels relay, the ADC converter circuit, and the humidifier. Each buck converter in the circuit was adjusted to output a voltage measured to be 5.15V, which is sufficient for the 5V components.

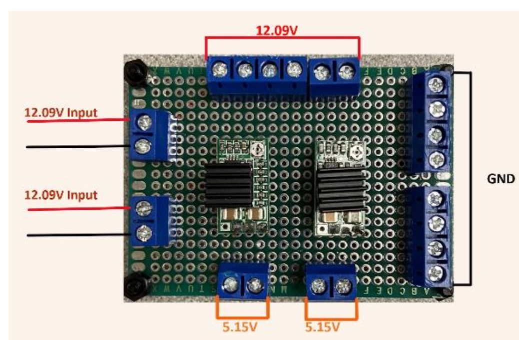


Fig. 44. Power Distribution Board [1]

2) Software

The Raspberry Pi side of the project for the Spring 2020 primarily consisted of using newer libraries, thoroughly testing each sensor, and fully implementing the automation features of the Smart

Greenhouse since much of the hardware was configured at this point. Gabriela also used more Python 3 than 2.7 this semester so that we were using the most recent version of Python. To read data from the temperature and humidity sensor during this semester Gabriela used Adafruit's Blinka (CircuitPython) DHT library. Circuit Python consists of Python core Application Programming Interfaces (API) with a growing list of over 150 device libraries and drivers that work with it [41]. CircuitPython is based on MicroPython. MicroPython is an open-source programming language interpreter that is compatible with Python 3 and optimized to run on microcontrollers instead of using C or C++, which was perfect for the implementation we are using for our Smart Greenhouse design. This library allows us to get data from the sensor by simply calling any of the functions in the library. The libraries Gabriela used in the Fall 2019 semester were deprecated so Gabriela decided to use CircuitPython since Adafruit still supports it.

Another update Gabriela did was to read data from the soil moisture sensor by using the CircuitPython MCP3xxx library. This library allowed us to use this ADC with any CircuitPython microcontroller board or any computer that contains GPIOs and Python. To read this data Gabriela created the Serial Peripheral Interface (SPI) bus, the Chip Select (CS) and the MCP object. From there Gabriela created the analog input to the channel and printed the data that is being read. Gabriela also decided to use this library for the light level sensor since it works in the same manner as the soil moisture sensor.

Gabriela also conducted various tests on the hardware components by reading data. The first part Gabriela tested on the Raspberry Pi was the sensor readings and sensor accuracy. The first sensor tested was the temperature and humidity sensor, which is an Adafruit DHT11 sensor. For this test, Gabriela took readings for 10 minutes, and sampled the data every 15 seconds. The data for the temperature and humidity can be seen in Fig. 45. and Fig. 46. The various temperature readings ranged from 71.6 °F to 73.4 °F. Gabriela was able to compare the readings to a digital thermostat that included a reading for the humidity. The actual reading from this device was 71.8 °F. Thus, giving a percent error of 0.28% to 2.23%. This falls in the range of our measurable metrics of 10% error. The various relative humidity (RH) readings ranged from 38 to 41% RH when the actual reading from the device was 40% RH. Thus, resulting in a percent error of 2.5% to 5%, also fitting in the acceptable range for our measurable metrics. "Relative humidity is the ratio of the partial pressure of water vapor to the equilibrium vapor pressure of water at a given temperature" [42].

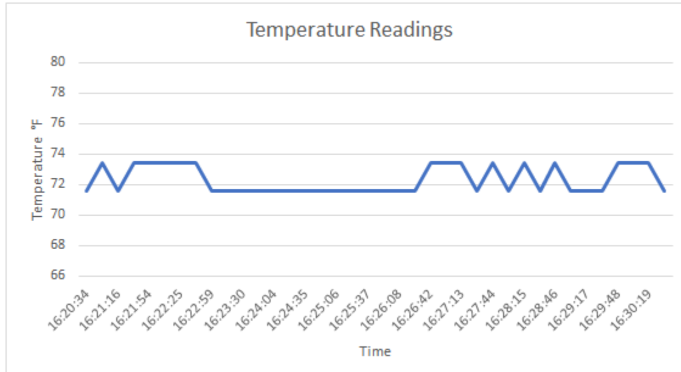


Fig. 45. Temperature Readings [1]

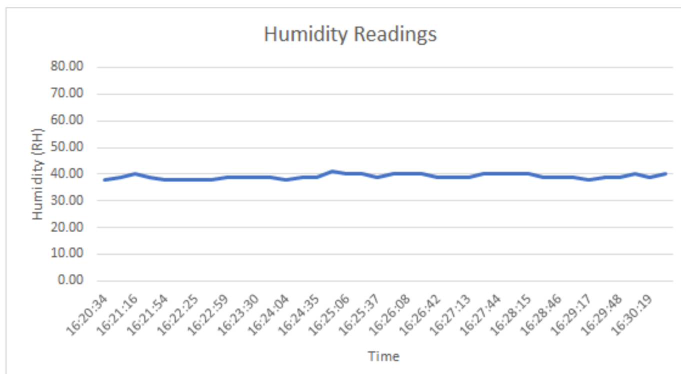


Fig. 46. Humidity Readings [1]

The second sensor tested was the soil moisture sensor, which is a YL-69 sensor. For this test Gabriela also took readings for 10 minutes and sampled the data every 15 seconds. The data for the soil moisture in dry and wet soil conditions can be seen in Fig. 47. and Fig. 48. Note that the soil moisture contains two probes that are used to measure the volumetric content of water. These probes allow current to pass through the soil and it gets the resistance value to measure the moisture value. When there is water, the soil will conduct more electricity which will result in less resistance.

Therefore, the soil moisture level will be higher. Dry soil does not conduct electricity well so when there is less water, the soil moisture level will be lower. Also, the analog readings are dependent on our source voltage. [18] The first test seen in Fig. 47. for the soil moisture sensor was soil in a completely dry condition. The various soil moisture readings ranged from 998 to 1002. The max readings for the soil moisture sensor is 1023 so this value seems accurate since 1023 means the dryest value possible.

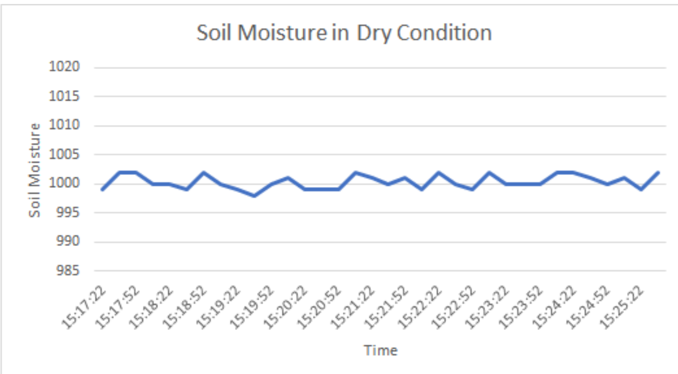


Fig. 47. Soil Moisture in Dry Condition [1]

The second test for the soil moisture sensor seen in Fig. 48. was soil in a wet condition when the soil has been just watered. The various soil moisture readings ranged from 279 to 349. The max readings for the soil moisture sensor is 0 so this value seems accurate since the value is closer to 0, meaning there is more water content in the soil. It can also be seen that the values continue to decrease with time which can be attributed to the water seeping into the soil since it had just been freshly watered.

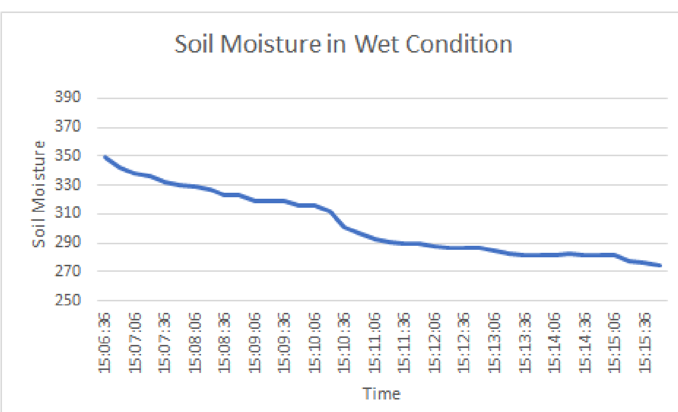


Fig. 48. Soil Moisture in Wet Condition [1]

The third sensor tested was the soil moisture sensor, which is the DROK 20033 Light Level sensor. For this test Gabriela also took readings for 10 minutes and sampled the data every 15 seconds. The data for the light level in direct light and average light conditions can be seen in Fig. 49. and Fig. 50. The first test seen in Fig. 49. is for the light level in very direct light. In order to simulate very bright light, our personal iPhone flashlight setting was used. The various light level readings ranged from 62 to 64. The minimum readings for the light level sensor is 1023 so this value seems accurate since 0 means the brightest value possible.

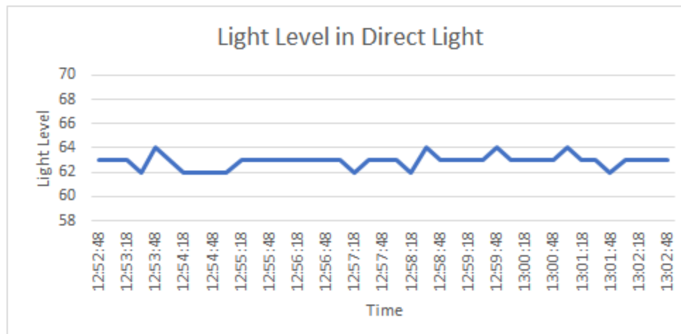


Fig. 49. Light Level in Direct Light [1]

The second test for the light level sensor seen in Fig. 50. is for the light level in average light. This test was set up in the senior design room with all of the lights on. The various light level readings ranged from 941 to 954. The maximum readings for the light level sensor is 1023 so this value seems accurate since the higher the value, the darker the room is.

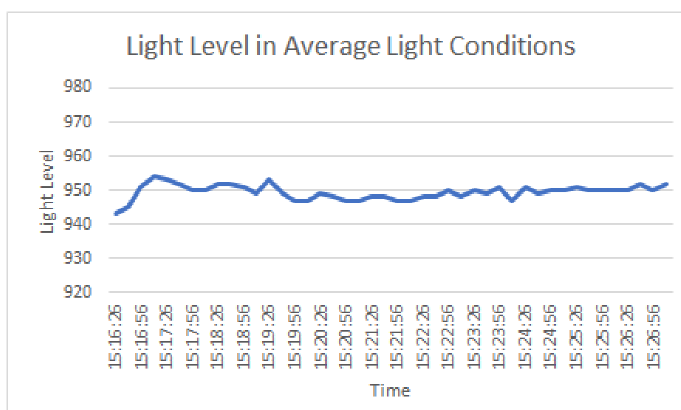


Fig. 50. Light Level in Average Light Conditions [1]

Jay worked on testing and adjusting the water level sensor. As mentioned above, we used ultrasonic sensor HC-SR04 as our water level sensor. The disadvantage is that the distance measured is not accurate to the true value. A solution for this problem is to come up with a function between the readings and the true values so that we can determine the true distance based on the readings. Jay first conducted a list of data points to determine the accuracy of the sensor. It turned out that this sensor does not work for distances lower than 4cm. Also, at some point the error could ramp up to 10%. A plot of the data points was done using Excel and Jay found that there is a non-linear relationship between the sensor readings and the true values. To solve this, Jay divided the data points into 3 sets, and conducted a function for each set as shown in Fig. 51.

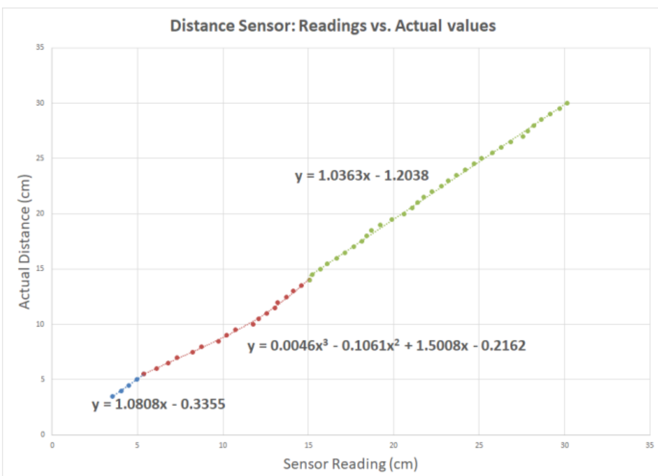


Fig. 51 Water Level Sensors Readings Calibration [1]

In the program, the readings are also divided into 3 intervals with the associated functions. Depending on which interval the reading falls into, a specific function is applied to output a value that is closer to the true value. By doing this, the accuracy of the sensor was improved significantly with the error of 5% maximum.

Gabriela also tested a few other pieces of hardware in order to get all the components prepared for the automation. Raj had worked on a 16 channel ADC in order to set up the various soil sensors that would be used for the multiple plant containers, so Gabriela tested every single channel for the ADC. Gabriela also tested that the relay switches Jay set up were properly working with turning on and off the plant LEDs, the humidifier, the fans, the heater, and the water pump.

The most important task for the Raspberry Pi side of the software was the automation. In order to test the automation, we had to see the greenhouse environment in different conditions and check if the hardware acted accordingly. Gabriela implemented the automation by creating universal settings in her Python code. These settings allowed us to simply change the GPIO pins of different items. These settings also let us easily adjust the timings for certain components to stay on for. This made it easier on both the hardware and software side so there were not multiple lines that needed to be changed if there was a hardware update since a variable was assigned that pin. A code snippet of the settings can be seen in Fig. C2.

To get the automation properly working according to the environmental parameters, Gabriela used multiple if statements that can be seen in Fig. C3. The if statements essentially detected if different thresholds were met for the temperature and humidity and it made hardware components turn on or off accordingly. A similar method was used for detecting the light level and for detecting the soil moisture. However, we used multiple soil moisture sensors for the various plant containers we had so Gabriela averaged the soil moisture sensor data since they were all watered together. A logic flow chart of the piece of code used to detect the soil moisture can be seen in Fig. C4. Overall, the implementation of the automation was rather straightforward and primarily relied on the current environmental parameters and the thresholds for each item such as temperature, humidity, light level, and the soil moisture.

Spring 2020 consisted of reevaluating the GUI and determining if making a website was the correct choice. Having a website would require an added networking aspect, which would include purchasing a router. Since we did not want to have this added into our budget, we decided that doing a website would not be an ideal game plan for our project. With this in mind, we did transition to making an application. Xiomara had to refocus her design and find the best method to make an application with only one semester worth of time.

During winter break, Xiomara did her research on the options available to make an application and decided on using the Blynk API to create it. Blynk is a "hardware-agnostic IoT platform with white-label mobile apps, private clouds, device management, data analytics, and machine learning" [43]. As shown in Fig. 52., Blynk is an application that can be downloaded on both iPhones and Androids. Xiomara downloaded the iPhone version since she has an iPhone. It connects to a Blynk server that we access with Blynk libraries. With these Blynk libraries, we can connect our application to our Raspberry Pi and use Python to program everything together. The ease of using Blynk truly made a positive impact as it helped remove the stress of any time constraints.

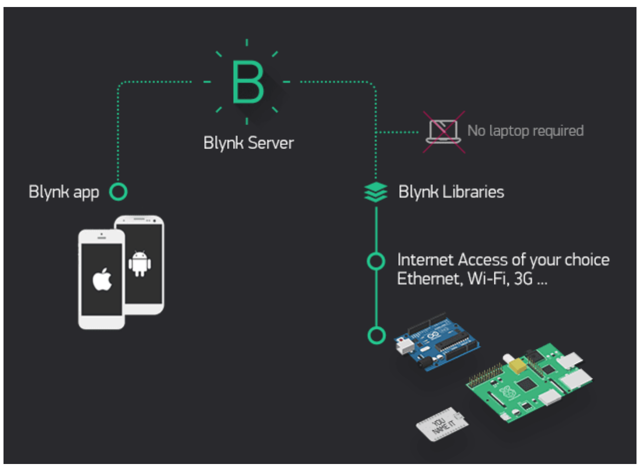


Fig. 52. How Blynk Works [44]

With only having one semester to work on an application from scratch, we also had to reevaluate the design of the GUI and only include the important aspects that would allow us to achieve the purpose of the Smart Greenhouse. We decided that the important aspects that we should include into our application would be the readings of each sensor, the manual user control to allow turning off and on hardware components, and getting statistics of overall readings to have a gauge on how the Smart Greenhouse is performing over time.

As shown in Fig. 53., the main page of the application consisted of the data display. This page received the data of the temperature sensor, the humidity sensor, the moisture sensor, the light level sensor, and the water level sensor. This consisted of all the sensors we used. Its purpose is to provide the user with real time sensor readings. Anytime there was an update in the Smart Greenhouse, it would be reflected on this page.

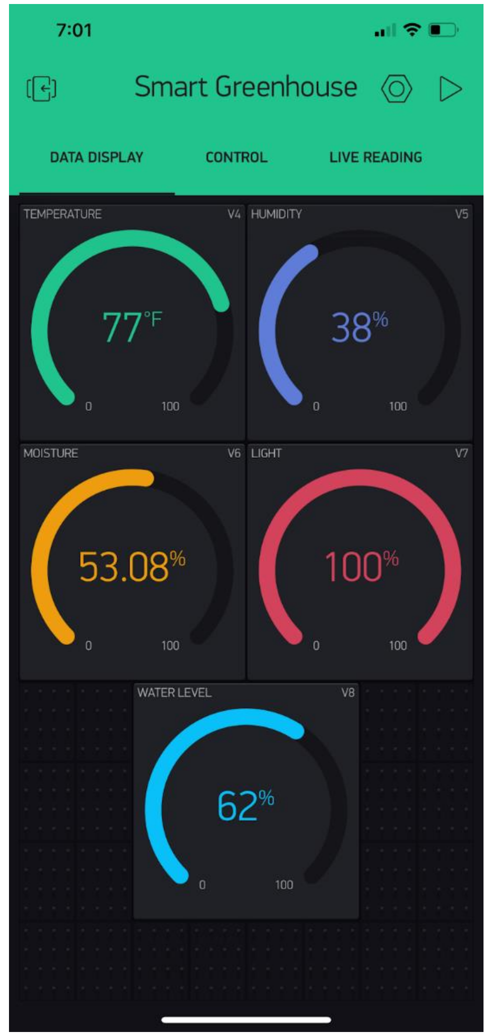


Fig. 53. Data Display [1]

The second page, as shown in Fig. 54., consisted of all the buttons that would allow the user to turn off and on each hardware component inside the Smart Greenhouse. The buttons were connected to the fans, LEDs, humidifier, heater, and the water pump. This would provide ease to the user because they would not need to physically go to the Smart Greenhouse to enable or disable these hardware components. The connections of the buttons on the application were connected with virtual pins, which Xiomara then connected to the GPIO pins in the python code. The virtual pins did not require a specific order, and they did not need to match the specific GPIO pin number and that is why a wider range was used for the connections.

Fig. 54. Manual User Control [1]

The last page of the application, which consists of Fig. 55. and Fig. 56., was where the live readings of the sensors were being graphed to provide the performance of the Smart Greenhouse over time. Since there were five sensor readings, Xiomara included two separate graphs to evenly disperse the graphs. The live readings can be displayed from one hour and up to three months. This would be useful for the users because it would provide a good gauge on how the environment of the Smart Greenhouse is doing. It could provide details if the temperature is too high, or if the water level has been consistently low, just to name a few examples.

Fig. 55. First Graph of Live Readings [1]

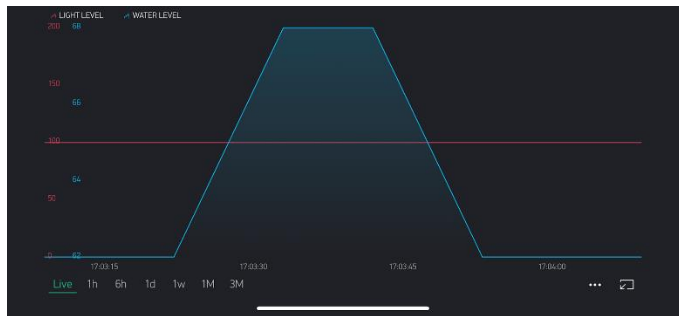


Fig. 56. Second Graph of Live Readings [1]

Between Fall 2019 and Spring 2020, there was a really big change in the expectations of the GUI. In Fall 2019, we did have a bigger plan for the GUI which would include much more for the user, but in Spring 2020, we realized that it was more important to just include the concrete aspects that would help demonstrate the purpose of the GUI. Although there was a setback of losing one full semester when we decided to completely change the GUI, we still overcame that obstacle and were able to successfully provide a GUI that was indeed user friendly.

3) Structure

For this semester, Raj started working on strength and stability for the structure. From the testing that Raj did during the 2020 winter break, he found out the structure was moving and bending side to side when the roof was opened and closed. To fix these issues and to also add more strength to the walls of the greenhouse, Raj used an ‘L’ bracket on each joint and edge. Using the “L” brackets, Raj was able to get the best result when installing the outside tarp because it did not leave any unwanted stretch marks in the tarp. Fig. 57. shows the ‘L’ brackets that Raj used to secure the walls and the roof.

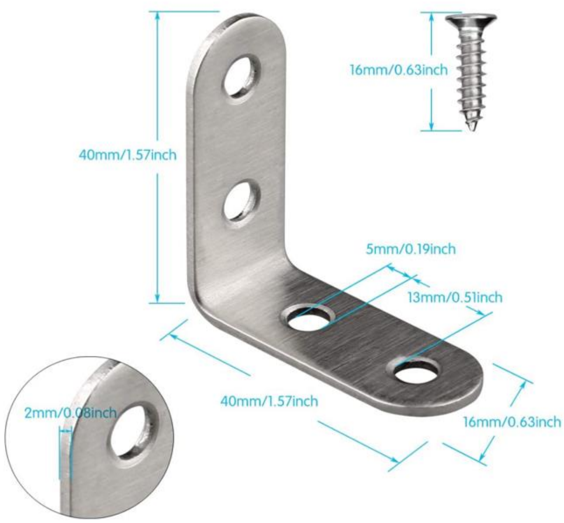


Fig. 57. ‘L’ Brackets [45]

The next part which Raj worked on was upgrading the irrigation system. As we only wanted to use one water pump to supply water to all the pots, Raj had to connect them in parallel. This was achieved by using a 1-to-4 irrigation convert. Fig. 58. shows the component which was used to connect multiple channels of water from just one water pump.

Fig. 58. 4 Port Irrigation Head [30]

By using this, the greenhouse used less water by supplying a perfect amount of water to all the plants. To supply the water to each plant, an irrigation bubbler was placed by each plant. Fig. 59. Show the component which allowed Raj to supply water to each plant. The components shown in Fig. 58. and Fig. 59. allowed Raj to connect each channel in parallel and each irrigation head in series for the plants.

Fig. 59. Bubbler irrigation head [30]

To maintain the internal temperature and humidity of the greenhouse Raj installed the outside tarp with the help of Jay and Gabriela. For the tarp, we used plastic Poly-film vinyl which was 8 mm thick. This thickness was perfect as it prevented outside temperature to affect the internal temperature of the greenhouse. Fig. 60. shows the vinyl which was used for the greenhouse and Fig. 61. shows how the vinyl looks on the greenhouse.

Fig. 60. Poly-Film vinyl [45]


Fig. 61. Greenhouse Vinyl installation [1]

After these components were completed, the hardware components were implemented and ready for the installation on the greenhouse, Raj with the help of Jay was able to install and secure the components such as fan, heater and LED onto the greenhouse. To house all the electrical components such as Raspberry Pi, power supply, ADC, and relays, Raj constructed a wood box that secured all the components. This box was made waterproof by sealing all the corners with All-purpose silicon. This silicon was also used in waterproofing the greenhouse structure and hardware components. To further waterproof the components and their wires, Raj installed cable runners to hide and waterproof the wires which connect between the components and the electrical box.

Lastly, and the most important part of the project, was selecting the plants we wanted to grow. We wanted to demonstrate the capacity and capability of the greenhouse. We chose strawberries and sugar pod peas. By selecting them we can show that two different kinds of plants are able to grow in a small area. Fig. 61. shows the completed greenhouse.

IV. Funding

The funding cost for our project was paid for by our group members only. Initially, we did reach out to the Environment Science department and the Biology department with a request to use the greenhouses on the Sacramento State campus for developing our project. If the department had allowed us, we would have saved money on structure parts. This would have helped us invest more in better hardware components. As a team, we decided in the initial stage what total amount we wanted to spend. The maximum we wanted to spend was $600 as a group. By the end of Spring 2020, our total cost of the entire greenhouse was $708.48 due to having to repurchase a few components. To make funding and dividing the amount easily, we had one of our team members order all the components and items for the project. Raj ordered all the items and kept an excel sheet with the component's price and link if we needed to order the same component again. Appendix E has the list of all the components used for this project.

V. Project Milestones

Project milestones were very important to our project as it geared us through our timeline and provided a guide to what important components needed to be completed for a successful deployable prototype. During the first semester we had very generic project milestones that consisted of completing the initial laboratory prototype for Fall 2019, completing the feature sets which are structure, hardware, and software, and lastly, to finish the deployable prototype for Spring 2020. When reflecting to our work, we realized we could break it down even further for more specific project milestones. We were able to compile project milestones according to each feature set as described below.

A. Hardware

An important project milestone within the hardware feature set was the power management. Power management was key because we had a decent amount of hardware components that we included into our Smart Greenhouse. Without successful power management, there could have been components that could not function properly and sufficiently due to the lack of power supply or in the worst case, components could be permanently damaged. Having good power management not only helped us prevent potential failure but also played a key factor to ensure the efficiency of the system.

A second project milestone for hardware was the completion of the wiring. All connections from the power supply to the power distribution board, from the Raspberry Pi to the relay, from the relay to other hardware components, and from the sensors to the ADC converters had to be secured by soldering as well as waterproofing. The wiring needed to be done correctly and efficiently because there were a lot of connections and wired lines required for the project. It also needed to be organized with name tags to allow for cohesiveness and no confusion. One wrong connection could have led to a failure in one feature and it could have damaged one or multiple components. Having the wiring completed successfully was an important factor that contributed to the completion of the project.

B. *Software*

A project milestone within the software feature set was the successful reading of the sensors. Having a successful reading of the sensors from the beginning allowed us to use the readings to test our other software features such as the automation and the manual user control. Having accurate sensor readings was also key to ensuring we were on the right track with our software feature set. Not only was the sensor reading needed from the Raspberry Pi, but it also allowed us to see that our GUI was successfully connected to the Raspberry Pi since we were getting the readings on the GUI.

A second project milestone within the software was the manual user control. Having the manual user control completed, confirmed the completion of one of our main design idea components which was the user-friendly aspect. Having this milestone finalized ensured that we had the ability to remotely make changes on the Smart Greenhouse.

Lastly, a third project milestone for the software was the completion of the automation. The automation was crucial to our design idea as well. Having this completed allowed for the Smart Greenhouse to be automated under a controlled environment. With the automation complete, being that it was one of the final things for our timeline, it demonstrated how exactly our system worked under controlled conditions.

C. *Structure*

One important project milestone within the structure feature set was the completion of adding the tarp to the structure. Adding the tarp signified that every hardware component needed for the project was successfully installed inside. Also, adding the tarp would allow for further testing of the conditions for the Smart Greenhouse that would be used for the automation. Without the tarp, we would not have been able to have a controlled environment within the Smart Greenhouse which would be insufficient for our project measurable metrics.

Another project milestone within the structure feature was the waterproofing. Since all our components, including hardware, wiring, and water tank, were all going to be compact and right next to each other, the waterproofing needed to be done successfully. With the waterproofing completed, we could manage all our wiring and hardware components without any fear of water damage.

VI. Work Breakdown Structure

There are multiple features that have been implemented to complete the Smart Greenhouse and this required our team to assign various components to different members starting in Fall 2019. The features of our design are the hardware, software, and structure. A quick overview of the team members assigned to each feature of this project can be seen in Table III and the total hours each member has worked can be seen in Table IV. The work had been divided in a way that emphasizes a logical approach of organizing the items needed to be completed for the project, known as the Work Breakdown Structure (WBS). This has allowed us to manage the design of the project by confirming that we have delivered results and completed all tasks that are required in the project. This has also ensured our tasks have been completed in a specified time frame, as seen in the project schedule in Appendix F. Some obstacles that we have come across are the complexity of the project, and the requirements for measurable metrics. By having used a “top down” approach, we outlined Features, Tasks, and Activities for our design, which can be seen in Table V in addition to the total hours worked for each feature.

There have been many milestones as discussed in the previous project milestones section. To reiterate, some of our major project milestones were completing the initial laboratory prototype for the Fall 2019 semester, completing the structure for the greenhouse system, completing the hardware, completing the software, and finishing the deployable prototype for the Spring 2020 semester.

TABLE IIIII
WORK DISTRIBUTION

Feature	Assigned Team Member
Hardware	Thinh 'Jay' Nguyen
Structure and Hardware Assistance	Raj Bhatt
Software: Raspberry Pi	Gabriela Estrada
Software: GUI	Xiomara Valdivia

TABLE IVV
TEAM MEMBER TOTAL HOURS WORKED

Team Member	Fall 2019 Semester	Spring 2020 Semester	Total Hours
Thinh 'Jay' Nguyen	40	136.41	176.41
Raj Bhatt	77	110	187
Gabriela Estrada	53.5	110.01	163.54
Xiomara Valdivia	40	96.56	136.51

TABLE V
FEATURE TOTAL HOURS WORKED

Feature	Assigned Team Member	Hours Worked	Total Hours
Hardware	Raj Bhatt	39	153
	Thinh 'Jay' Nguyen	114	
	Gabriela Estrada	27	
Software	Gabriela Estrada	88	177
	Xiomara Valdivia	89	
Structure	Raj Bhatt	90	90

A. *Hardware*

The first feature of our design involved the hardware needed to execute our solution. Jay and Raj were primarily involved in this feature with some additional help from Gabriela. The first task involved the sensors. The first activity involved researching into the type and model for sensors. The next activity was to purchase the sensors. The third activity was to connect the sensors to the microcontroller which Gabriela completed. This was attempted to be done in a timely manner since the information for the pins used would be implemented in the software side of the design. The last activity of this task was to test the sensors to confirm that they were working correctly.

The next task for the hardware was the power management of the system, which Jay completed. Since we were using multiple devices and sensors, we confirmed that there was enough power to run all of the components. The activities include researching into the total power consumption of the system, researching if a power supply is needed and purchasing it. Other activities also included testing the power supply and setting up the components to use the power supply.

The last task for the hardware was to build any circuits necessary for the sensors and devices that would be used. The first activity was to research relays and any voltage regulators that would be needed. After making purchases for these items, Jay and Raj set up Printed Circuit Boards (PCBs). A few other tasks that were related were finalizing the Laboratory Prototype, completing the final product for the power management, finalizing the circuit board, connecting everything to the Raspberry Pi, and soldering the ADC circuit.

B. *Software*

The second feature of our Smart Greenhouse system is the software. There are also multiple team members who worked on this feature. The first task for this feature includes reading the sensor data from the Raspberry Pi, which Gabriela worked on. The language for programming the Raspberry Pi was Python 2.7 and Python 3. The first activity was to research into various implementations that can be used for reading information from the sensors. This helped to decide which Python libraries to use.

The second activity was to program reading data. The last activity for this task was to troubleshoot and test that the Raspberry Pi was reading data from the sensors.

The next task was to set up the communication with the GUI. The communication involved tasks such as researching the implementation, programming the Raspberry Pi to correctly automate and control the system according to user input, and testing the communication, which Gabriela also worked on in partnership with Xiomara.

Another task Gabriela worked on was creating the manual control from the microcontroller side for the different aspects of the greenhouse system. The activities involved researching how to implement this from the GUI at the end of the design. This also involved communicating with Jay who worked on hardware since Gabriela had to program the relays. The second activity was to program the microcontroller to control the system according to user input from the GUI. The third activity was to troubleshoot and test the manual control from both the microcontroller end and the application.

The last task Gabriela worked on was automating the system. This task involved activities such as researching the environmental parameters for the crops that we would be growing and setting those environmental parameters in the code. In addition, other activities were to program the microcontroller and to troubleshoot and test automation.

The next task was to create the GUI application, which Xiomara worked on. This activity involved researching how to implement the application. The next activity was to design the GUI by creating a sketch of the format.

The next task Xiomara continued to work on involves outputting the sensor data onto the application. The activities in this task included researching how this would be implemented, programming the GUI, and testing that the information would be outputted onto the screen correctly.

Another task for software was creating the automation feature from the application. The activities that would be involved for Xiomara were researching how to display the environmental parameters to be chosen, programming detecting the user input, and programming the GUI.

The last task that Xiomara worked on was the manual control of features that users will select. The activities for this task included researching into the implementation, programming the user input, and testing and troubleshooting the application.

C. Structure

The last feature of our design is the structure of the greenhouse shown in Fig. 62. The member who was assigned to this feature was Raj. The first task that was completed was the wood framing. Our planned wood framing consists of the sides, top, and bottom of the structure. The first activity involved was researching various materials (wood, metal, etc.) and also creating a rough sketch for the system. Researching the type of material was important because this accounted for the budget of our design. The sketch helped us determine rough measurements and sizing for the greenhouse. One thing that was put into consideration was that the size must be portable so it must be able to fit in doors for transporting. Also, after researching and creating a rough drawing of the greenhouse, the second activity was to determine the exact measurements for the framing. The third activity was to purchase all the components needed for the wood framing such as the wood, nails, hand saw, etc. The fourth and final activity for this task was to assemble the entire framing for the system. This included assembling the framing to incorporate ventilation fans (to decrease humidity), assembling the structure in a way that the lighting will be in the appropriate location for the plants, and assembling the structure in a way that the irrigation system will be routed correctly.

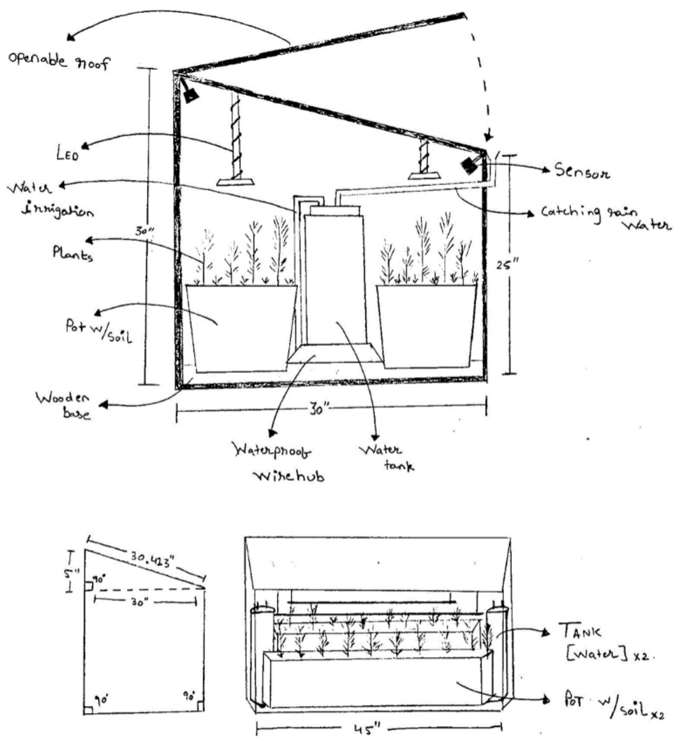


Fig. 62. Conceptual: front & side view of the Smart Greenhouse [1]

The second task for this feature was to make the structure portable. We initially planned to use a cart for the portability aspect of the design. However, we decided to install lockable wheels into the wood foundation instead, but this still required the same type of activities. The first activity correlating to this task was researching and measuring the foundation and wheel sizing that was needed. The second activity was to purchase the foundation and wheels and the third activity involved installing the greenhouse structure on top of this cart. This was an important activity because the greenhouse must be secured safely on top of the cart for transportation.

The third task was to install the outside, tarp-like material that will be transparent and waterproof. The first activity for this task was to research the type of material that would be used for the exterior of the greenhouse system. We decided to use a common material used in greenhouses, such as polycarbonate. The second activity was to purchase the material and the third activity was to install the material onto the wood framing of the structure. The material needed to be installed correctly to ensure there was insulation within the system.

The fourth task of the structure was to confirm there is waterproofing throughout the system. This task also consisted of an activity that required research, measuring the material to be used, and purchasing the material. We decided to use silicon for insulation for any open slits and an electrical box for the hardware components. The last activity for this task involved installing all the waterproofing and checking that it is indeed waterproof. This activity was very important because we have wires and the microcontroller within the system, so we had to confirm that the system is completely waterproof and safe from leakages.

The next task was to insert the crops containers into the structure. Once the structure was finalized, we prepared for planting the crops. This task also involved similar activities such as researching the measurements and type of container or pot needed and purchasing the containers. These were also needed to be installed into the system in a way that the plants would be secure and did not fall over, possibly damaging the plant.

The final task was to manage all the components inside the structure. This also involved research and installation of the components. Also, we had to wire the wires in an efficient and safe method and decide where components, such as the microcontroller would be placed. After breaking down the features of the structure we were able to organize the timeline for it.

D. *Leaders and Course Assignments*

The team leaders were Gabriela, Xiomara, Thing, and Raj, in this order. It can be observed in our WBS table from Appendix E that we included our course assignments that overlapped with the different aspects of the features in our design. The course assignments from Fall and Spring semester are also noted.

VII. Risk Assessment

With implementing this level of product, there are risk factors which are involved while developing. Each part of the project has a different risk factor such as physical risk factors, hardware malfunctioning, and unsecured program and network implementation. All these factors affect the project in different ways. As the project relies on each part to work properly to function at its finest, remediation plans are needed to recover for big impact. Some external problems such as natural disasters or a pandemic are also considered for our risk factors, which may also upset our project schedule. All this will help us maintain the progress of the project during the implementation period.

As the Smart Greenhouse' main parts are hardware, software, and structure. each part has a different risk factor. For structure, risk factor includes getting physically hurt while building the structure, structure getting damaged while transporting and not meeting the place needs for the project. For the hardware aspect of our project, the risk factor includes device malfunction, new parts not working as expected and not able to find parts which are needed for the project. All this can affect the project progress extensively. That said, software is a big part of our project as it includes the GUI and the automation of our project. The risk factor and impact level are high as the main implementation and feature of our project will not work.

All these risk factors affected the project progress in different ways depending on the part of the project. It also shows each team member responsibly if the part they are working on is affected due to any factors. This gives us a better way of approaching a problem if we ran into any problems. The structure and networking component will be handled by Raj, the hardware component will be handled by Jay, the microcontroller part of the software will be handled by Gabriela, and the GUI part of the software will be handled by Xiomara. With these tasks at hand, we wanted to make sure nothing goes wrong. Thus, risk assessment is done, and remedial plans are created.

We took every possible risk into account including external and internal risks. Besides the external risks, internal risks consist of risk factors in the process of making the Smart Greenhouse Project itself. The Smart Greenhouse is divided into three main parts: structure, hardware, and software. Each part has multiple tasks that carry specific risks.

A. Hardware Risk Factors

The hardware risk factors with likelihood level and impact level is described in Table VII below.

TABLE VI
HARDWARE RISK FACTORS

	Potential Consequences				
Likelihood	**Minimum/ No Impact (1)**	**Can be tolerated (2)**	**Limited impact (3)**	**May jeopardize project (4)**	**Will jeopardize project (5)**
Near certainly (0.9)					
Highly Likely (0.7)					
Likely (0.5)					Power supply and buck converter outputting wrong voltage.
Low Likelihood (0.3)		Relay malfunction		Getting burnt or electrocuted when building circuit. Networking hardware malfunction.	Wrong connections / Loose connections in the circuit.
Not Likely (0.1)					

Sensors are very important since they are the devices to show how well we can control the environment in our Smart Greenhouse. There are several issues that might occur with sensors. First, choosing the type of sensors is essential. There are many different models of a sensor that do the same job with different efficiency. Choosing one that does not fit the job requirement can result in a poor performance. Secondly, the sensors we bought might be defective from the manufacturer. It will cost us extra time trying to troubleshoot to make it work. Sensors breaking is another issue that can occur during experimenting and handling. Besides, one common issue that many sensors have is that they give inaccurate readings due to improper mounting. For instance, a soil moisture sensor needs to be oriented sufficiently downward into the soil in order to give accurate measurements. Each sensor has specific working conditions that have to be met. To overcome these issues, it is best to test each sensor with a simple circuit and troubleshoot until we get good signals. In the case of broken and defective sensors, we will have to buy new ones. The impact of this risk is level 3, and the likelihood is level 4.

Power management is a significant task of the project. The Smart Greenhouse will be powered from the 110V AC power outlet. Our hardware components are rated at 12V (fans, grow lights, water pumps, heater) and 5V (Raspberry Pi 3B+, relay, humidifier, and sensors). We use a power adapter that transforms 110V AC into 12V DC 30A for our system. There might be chances that the transformer does not output the correct value meaning it might supply a voltage lower or higher than 12V. For a lower voltage supplied, the impact is small because the 12V components will then not work efficiently due to the low power supply. On the other hand, if the transformer outputs a voltage higher than 12V, it will make the 12V rated component working harder than they can and eventually collapse. In the case of 5V rated components like the Raspberry Pi and the sensors, we use buck converters to convert the 12V DC power supply into 5V 3A. Similar to the power transformer, the buck converter might not output the wanted voltage. The components will not work properly under the low voltage supply condition, and they will burn if supplied with a higher voltage than their rated value. Another issue with power supply is that the system draws more current than it can supply. This can also make the system to work inefficiently. Since power supply can affect the whole system negatively, this issue can jeopardize our project. The impact is at level 5 and the likelihood is considered at level 3. To mitigate this the power supply problem, we will re-evaluate power consumption and troubleshoot or purchase new power supply.

Another important component of the hardware is relays. We cannot control the features without relays since the Raspberry Pi GPIO pins only output a 3.3V, which is not enough voltage to run other components such as fans or water pumps. The relays, which can be triggered by the Raspberry Pi signal, are connected between the 12V power supply and the load (fans, grow lights, water pumps). There is a risk that the relays are not working properly resulting in poor performance of the controlling features. The impact of this issue is minimal at level 2, and the likelihood is also level 2.

The process of building circuits on PCB boards can have several risks. Jay had to use the solder to implement wires and components onto the PCB board. The soldering iron tip gets very hot in order to melt the leads, and it is easy to get burnt during this process. Another type of physical injury might occur is getting electrocuted if the power supply does not work properly as mentioned above. To mitigate, protective tools are required. The impact of this problem is level 4, and its likelihood is level 2.

There are two other issues with the circuit that can severely affect our project are wrong connections and loose connections. Wrong connection not only makes the system perform poorly but can also destroy components if it forms a short-circuit. Short-circuit causes the current to increase to a rather high value and result in fire or explosion. This can potentially destroy multiple components of the systems and jeopardize our project. A loose connection in the circuit, on the other hand, creates an open circuit. When there is an open circuit, the system cannot fully work because some components might not have connections to the supply. It might have a lower impact on the well-being of components, but it will cost time to troubleshoot the system. The likelihood of circuit issues is at level 2 and the impact is at level 5. Our mitigation plan is to first troubleshoot the circuit, then replace the bad parts, and rebuild the circuit.

B. *Software Risk Factors*

The software risk factors with likelihood level and impact level are shown in Table VIII below.

TABLE VII
SOFTWARE RISK FACTORS

		Potential Consequences				
		Minimum/ no Impact (1)	**Can be tolerated (2)**	**Limited impact (3)**	**May jeopardize project (4)**	**Will jeopardize project (5)**
Likelihood	**Near certainly (0.9)**					
	Highly Likely (0.7)					
	Likely (0.5)					Inaccuracy in reading and transferring sensor data. Automation feature failing User input is not detected GUI not responding RPi and website lose communication.
	Low Likelihood (0.3)		Bugs occurs when improving the software	Incorrect website link	Network software malfunction	IP issues, no connection
	Not Likely (0.1)					

Since software risk factors do not relate to any physical components, the mitigation plan for every issue is to troubleshoot and debug.

Reading and outputting data from sensors from the Raspberry Pi can potentially have issues. Other than inaccurate readings, the data from the sensors are in digital or analog depending on each sensor. For accuracy in controlling, we mostly want to deal with analog data from the sensors. For each specific sensor, there are specific codes associated with it, and there might be issues in programming to get the right readings. The impact of this issue is critical (level 5) since sensors are directly related to the environment we want to control, and the likelihood is at level 3.

The automation feature is a significant component of our project. It is rather hard to control the hardware (fans, water pumps, grow lights) correctly with minimal errors. The risk of hardware over-performing or under-performing is expected to occur. Like automation, the manual control feature of the project might not work due to user-input not detected. Programming is sensitive to changes, one small change in the code may alter the performance of multiple components. It takes time to troubleshoot and debug the codes, which can potentially delay our progress. The impact of these risks is level 5, and they are likely to occur.

Accuracy is very important in controlling and changing the environment. Bugs might occur during the process of improving the accuracy of modules. This risk factor has low likelihood (level 2) and a lower impact to our project (level 2).

In order to provide the user with a GUI, the Raspberry Pi needs to communicate correctly with the website in real time. There are chances that the microcontroller and the website do not communicate with each other. Since Xiomara and Gabriela are working on this part, they will have to collaborate to troubleshoot this issue. The impact is level 5 and the likelihood is level 3.

Since we offer the user with GUI, and the website is accessed wirelessly through IP address. This might be a problem since each network will result in a different IP address. This issue is rather critical because if the GUI does not work, there is nothing to show that we can change the environment in our Smart Greenhouse. The impact level is 5 and the likelihood level is 2. Xiomara will collaborate with Raj on troubleshooting if there are network problems.

The GUI will be responsible for displaying sensor data and receiving user input for manual control as well as automation features of the system.

There are several risks associated with this feature. Firstly, the sensor data transferred from the microcontroller to the website might be inaccurate since the format is different when extracting data from sensors from the microcontroller and when sending data from the microcontroller to the website. Information might be missing in the process. Secondly, since we will implement the GUI with sets of pre-defined parameters for automation, there might be errors in setting up these parameters that make the automation features perform poorly. Lastly, the most important component of GUI is user input. We want to provide the user with both automation and the freedom to manually adjust the environment in the greenhouse. The user input might not work properly resulting in inefficient control. The impact of these issues is significant at level 5 and the likelihood is at level 3.

C. Structure Risk Factors

The risk factors of the structure with likelihood level and impact level can be shown in Table VI below.

TABLE VIII
STRUCTURE RISK FACTORS

Likelihood	Potential Consequences: Minimum/ no Impact (1)	Can be tolerated (2)	Limited impact (3)	May jeopardize project (4)	Will jeopardize project (5)
Near certainly (0.9)					
Highly Likely (0.7)				Damage hardware component	Irrigation system leaking, pumping issues
Likely (0.5)					Wood framing broken/ unstable. Waterproofing fail. Plant container breaks, plants dies.
Low Likelihood (0.3)				Wheels under the foundation break. Outside Tarp ripped.	Physical injuries while building structure.
Not Likely (0.1)			Inaccurate measurements.		

When building the structure using tools such as a hammer, drill, and saw, it is very common to run into accidents. The risk of this task is physical injuries such as getting cut by the saw or fingers getting slammed by the hammer. The injuries could be anywhere from minor to severe depending on the accident. If this happened, the progress would be slowed down because the structure contains all other features. To mitigate, the person in charge of this part would be assisted by other members. The likelihood of this risk is level 2 with the impact of level 5.

The wood framing of the structure is very important to our design. The framing is built using screws, nuts, and bolts. Because our design is portable, we will move it from place to place, and movement and vibration might cause the connections of the structure to be loosened. Therefore, there is a risk of broken or unstable structure. Worst case scenario would be that the framing collapses with other components inside. Since we only have 4 weeks left until our prototype presentation, the impact of this risk will be very critical to our progress. Our mitigation plan would be to fix or to rebuild the frame. The impact is significant at level 5 and the likelihood level is 2.

The cart foundation is another crucial part of the structure. We have wheels attached to the foundation so that we can move our design around, if the design turns out too heavy, the wheels might not be able to bear the weight and break. Even if everything stays intact when the wheels break, we will not move the design anywhere. The wheels breaking might then result in the whole structure collapsing and damage our hardware inside. Like the wood framing, this risk will jeopardize the project. To mitigate this, we will have to buy and reinstall new wheels with higher weight capacity. The likelihood is low at level 2, and the impact level is 4.

For the outside tarp of the greenhouse, there is a risk of buying the wrong material or the tarp rips due to an accident while we are working on the design. The covering tarp must be transparent as well as strong enough. This is an important component of the structure because without a good covering tarp, we will not have the greenhouse environment isolated from the outside environment. We would not be able to show any control or change the environment inside the greenhouse without the isolation. To mitigate from the ripped tarp, we will have to buy and install better material for the tarp. The likelihood level is 2 and the impact level is 4.

One significant component of the structure that will severely affect the project if it goes wrong is waterproofing. Waterproofing is one of the most important tasks of the structure. Since the controlling system including circuit, power supply, microcontroller, and other devices are inside the greenhouse, we need to secure them from water, or the system will be destroyed otherwise. The risk of having bad waterproofing can cost us significantly, and result in destroying many components and jeopardizing the project. Our mitigation plan for this is to check the whole system for errors and reinstall new components that are affected.

Plant containers might also cause problems with our project. There are chances that the container is cracked or broken and allow water to escape. It then creates two problems. The first problem will be water leakage causing the water pump to keep running, and there will be a lot of water going to be wasted. The second problem is when the container cannot contain water, it might not provide enough water for the crops, and eventually kills the plant. The plant dying shows that our project has failed. To mitigate this, we must buy new containers and plants to grow all over again. Since the well-being of the plants is the parameter to determine the success of our project, this risk is rated to be crucial at level 5, and the likelihood is level 3.

Irrigation system might not work as expected as I might have water leakage and pumping issues. Water leakage can be caused by the loose connections between the pumps, water tank, and water pipes. Ultimately, it will end up wasting water and providing too much or too little water to the plants, which decreases the plants growth rate or kills the plants in the worst case. The impact of this risk is at level 5 with the likelihood of 4. This risk is mostly expected and to mitigate it, there will be a lot of troubleshooting, fixing, or installing new parts.

During installation of hardware components such as fans, grow lights and water pipes, the mounting might become loose, resulting in dropping and damaging the parts. This risk does not have a heavy impact on our project since it only affects components individually. Our mitigation plan is to fix the components if possible or else we will purchase and replace new parts. The likelihood is level 4, and the impact is rated at level 3.

From the prototype in the first semester, we must improve our structure for the final product by implementing more features and components. Raj will have to design a layout for the final product. The area and locations of each feature must be specific and measured correctly. The design must be based on the measurements of the structure components as well as the hardware components. There are chances that the measurements are inaccurate causing the components to not fit in the structure. We might also have too many components and they cannot all be organized inside the greenhouse. The impact of this risk is somewhat significant since it will take time to resize and redesign the structure to mitigate this problem. However, the likelihood of this risk is minimal at level 1 because we have carefully measured and come up with reasonable measurements beforehand. The impact is considered at level 3.

D. External Risk Factors

The external risks are factors from outside of the project and we could not control or prevent such as extreme weathers or the COVID-19 pandemic quarantine. External risks would result in school being shut down, or the team cannot meet up to work on the project. One specific example is the Camp Fire on November 8, 2018 that affected the air quality of a large area including campus. Ultimately, the campus had to close for a week, and no student would have access to any room on campus. Another risk might be strong winds that can also cause campus to be close. On the recent weekend, we had strong winds for a couple days causing a lot of broken tree branches. It would be dangerous for both students and staff to go to school with this weather condition. Campus would have been closed if it continued the weekdays. For any reason, if the campus were to close for a long period, it would be extremely hard for the team to meet up and work on our project since we have always met at school. Another strong example of an external risk factor is the COVID-19 pandemic that not only causes the school to shut down but also prevents any face to face meeting in a period of months. Any pandemic situation would significantly delay us from getting our project done before the due date. The likelihood of this risk is at level 1 since it does not happen often in our area. However, it has a high impact on our progress, so the impact level is rated at level 4.

Through this deep analysis of each work package, we have identified the risks, reported the likelihood and impact of each risk, and provided mitigation plans to be prepared for each situation. Our rankings for each risk have allowed us to determine what is at most risk, and what is at least risk. Having this planned out, we were not only prepared for a mitigation plan for any risk that had happened but also were aware to completely avoid as much risk as possible. Through this process, we can see that even small work packages can make a big impact on the project.

VIII. Design Philosophy

Our project is a solution to the growing problem of agriculture degradation, climate change, and food security due to the growing population and global warming. Our design philosophy for our project was to have a solution that had a positive impact on all the world's environmental problems. Our team knew four or even a hundred users of our product would not create an impact which will create a positive on our societal problem. Thus, we wanted to develop a product that anyone can have access to and be able to buy it. To keep it accessible, user-friendly, and simple with having positive results, during the designing, implementation, and execution stage of our project we kept small details in mind.

We brainstormed multiple different design variations of the greenhouse structure. The design we finalized and implemented was done as we saw more plus points to the design. The reason the greenhouse is taller on one side and shorter on the other is because one side the user can grow taller plants which will be in the back of the greenhouse. Thus, leaving the front of the greenhouse open for smaller plants and also users can also access the components such as the water tank and electrical was more space. The roof of the greenhouse opens from the side which is smaller in height. The width and length were also kept in mind while designing the greenhouse. The width is just 2 inches under the normal size door open. This allows the user to live in an apartment to be able to buy our product and arrange it on the balcony or their house patio. We also placed three pots, two large and one small which allows the user to choose which plant they want to grow more.

From the hardware perspective, we wanted to use the best components but also keep the overall price of the product affordable. We used the best components we found online, and which was compatible with our electrical box design. We implemented our ADC which helped us in many ways. We were able to make a smaller PCB design with 16-channel input. This allowed us to switch channels if any channels are not working. All the hardware components are also replaceable as we installed wire connectors to each component. Thus, this keeps the system downtime at minimal.

Similarly, the software implementation was key in making our project user-friendly and simple. We wanted to use an application which is downloadable for any phone users and able to get started in a few easy steps. The layout of the app home screen, displaying values, and switch control are designed in a way that is easy to read and operate. This small designing layout makes the user interface simple and clean.

All these small details bind our entire project together and achieve results we wanted to create a positive impact on the problem of food security.

IX. Deployable Prototype Status

The Smart Greenhouse and its features have been completed as of Spring 2020. The structure was finished with implementing outside tarp, plant pots, water tank, electrical box with the LED growing light strips, fans, heater, and humidifier in appropriate locations. The electrical box was built inside the greenhouse to store the power supply, power distribution board, 8-channel relay, Raspberry Pi, and the ADC converter unit. All pin connectors are made to be inside the electrical box for better management. The components outside the box were connected by soldering and the connections are waterproof. For the power management aspect, the system has been tested to run for a long period of time and there has been no signs of failing or overheating on any component. The power supply has a built-in cooling fan that turns on at a certain temperature, but it has never been on because the temperature in the electrical box always stays low thanks to the ventilation system of the box. Similar results were found with the buck converter. The power management and connections were working well with no issues.

The automation aspect of the Smart Greenhouse has been done according to the measurable metric. From testing, the sensor readings have a percentage error at 5% maximum as explained above. The error is well below our value for error stating in the measurable metric of 10%. The system can control the temperature, the humidity, the irrigation, and the LED growing light strips automatically based on the sensor readings. Also, the system can control the temperature by 5°F increments and the humidity by 10% RH according to the measurable metrics.

The Blynk GUI was also completed. The GUI can be accessed through the app Blynk from a smartphone or tablet. On the GUI, the readings from sensors are shown to provide the user with the status of the greenhouse: temperature and humidity, light intensity, soil moisture level, and water level in the tank. Besides, the GUI also has a separate tab for manual control features. In this tab, the user can control the features such as LED strips, fans, or water pump manually just by pressing on the associated virtual buttons.

We chose strawberry and sugar pod pea plants that require a similar level of light intensity, humidity, temperature as well as soil moisture level. Four strawberry plants and three pea plants have been growing in the Smart Greenhouse for over a month. They have been growing well with no sign of failure. The strawberry plants have been blooming and we are expecting some strawberries soon. The sugar pod pea plant needs to grow bigger and we expect them to produce peas around June or July.

X. Marketability Forecast

From our previous market research, our team understands that there are a few items that would need to be refined in order to achieve an actual marketable device. Many end-users want a completely revised and finished product and since this project was created under a college-student budget, this means that there is room for improvement for all the features incorporated in this project.

Our team does expect that the Smart Greenhouse market is expected to significantly grow according to research reports conducted by market research companies. The market for Smart Greenhouses is expected to grow "from USD 1.3 billion in 2018 to USD 2.3 billion by 2023, at a Compound Annual Growth Rate (CAGR) of 12.6%", which can also be seen in Fig. 63. [46]. These market research companies state that driving factors in the market is the current population growth and government incentives regarding lighting technologies. In direct relation to our soil-based smart greenhouse approach, "the non-hydroponic segment is expected to grow at the highest growth rate during the forecast period" [46]. One modification to our project based on this information that could impact the sale of our design is that we could make the system eco-friendlier by modifying lighting and power systems. Therefore, this would make the product more desirable for governments to purpose. Another change that we could make to our project to favor the market based on this research is to focus on our non-hydroponic approach to make our greenhouse more competitive. For example, we could add hardware and software features to control and check pH levels in the soil. To continue, we could create an even larger market by advertising that any type of plant can be grown versus solely advertising growing fruits and vegetables.

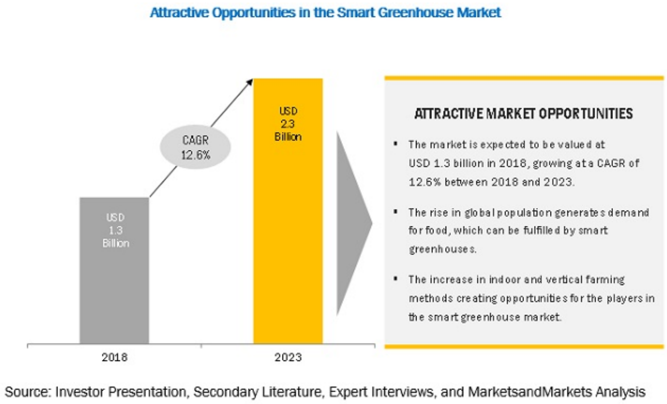


Fig. 63. Opportunities in the Smart Greenhouse Market [46]

We have found that in the Smart Greenhouse market, Europe accounts for the largest market size, following the Asia-Pacific region, as seen in Fig. 64. Europe has been known to be leading in using advanced techniques regarding greenhouses. These locations are expected to drive the demand for indoor controlled-environment agriculture in the future [46]. We could further improve our design by including different European languages in our GUI to favor a wider range of customers since GUI text is only in the English language.

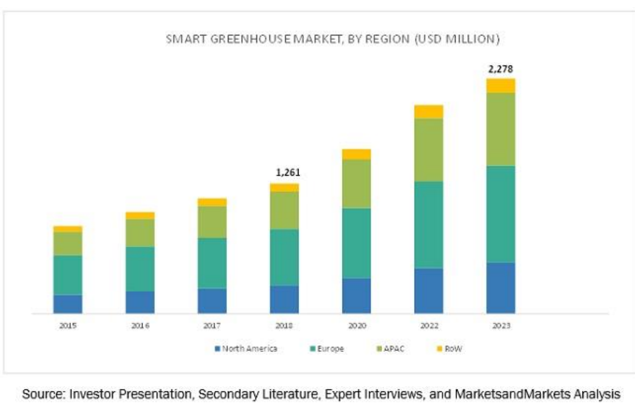


Fig. 64. Smart Greenhouse Market by Region [46]

In Fig. 65. it can also be observed that other areas such as Scandinavia, the Netherlands, Germany, and the United Kingdom have the top market share for smart greenhouse systems in Europe in 2017. The top 4 technologies for smart greenhouses were LED grow lights, heating, ventilation, and air conditioning (HVAC) systems, material handling, and irrigation systems. If our design were to focus on these top 4 technologies and improve and directly implement them into our current design, we could see a great profit.

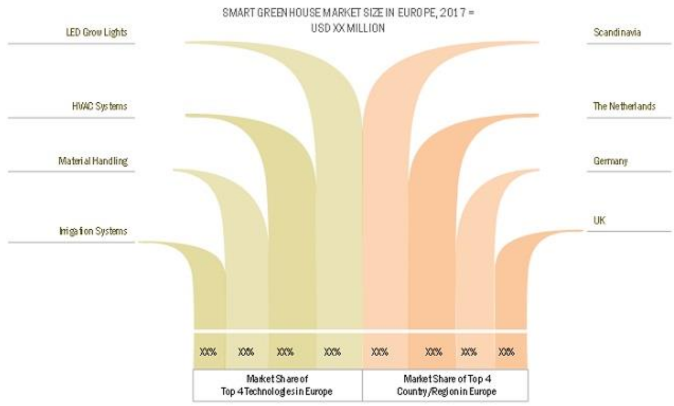


Fig. 65. Smart Greenhouse Market Size in Europe [46]

Many factors affect the cost of a greenhouse. First, is the size of the greenhouse. “Small greenhouses cost between $100 and $250, and larger buildings cost up to $25,000 for a complete setup with plumbing and electricity” [47]. The national median cost for greenhouses is approximately $16,283. The minimum cost is $800 while the maximum cost is $35,000 as seen in Fig. 66. The reported average cost to build a professionally installed greenhouse is estimated to be $13, 893 [47]. Another factor regarding the cost is the materials that are used to construct the greenhouse. The three main components that are needed are the frame, floor, and sides, which all have different costs according to what material is used. Another modification to our design that could increase market profitability is being able to advertise using different types of materials in order to allow the customer to decide how much the total cost will be according to their budget since we know the cost can vary greatly. We could also allow users to choose if they wanted additional features such as HVAC systems.

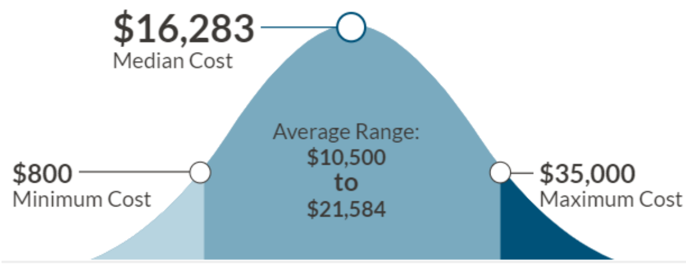


Fig. 66. National Cost for Building a Greenhouse [47]

Other modifications to our features that could be changed are optimizing our code for it to be at a professional level, using higher quality sensors to get more accurate readings, using industrial ventilation and lighting techniques, and using a possibly more advanced microcontroller. Since our team was on a limited budget and not sponsored, there are many opportunities for us to improve our system.

There is clearly market demand for the automated greenhouse system. There are similar products in the market, but our Smart Greenhouse provides both automation with GUI and portable aspects where other products only provide one or the other. Even though it costs a lot to make, the market is growing. It is promising to be lower than the average cost people would spend for a greenhouse while we still can make profit.

XI. Conclusion

Climate change significantly affects the agriculture industry around the globe. With the loss of agriculture and the growing population of the world, food security has become a serious problem to many individuals. The Smart Greenhouse is a great solution to solve this problem. The goal of this project is that individuals can easily grow crops with no physical work or agriculture experience required. The growing process is taken care of by the automated system including irrigation, lighting, ventilation, temperature, and humidity control features. The project is also implemented with a GUI that not only displays data but also allows the user to control the greenhouse features. The Smart Greenhouse consists of three major factors: structure, hardware, and software. Each team member specializes in different aspects of the project. The Smart Greenhouse was completed following the work breakdown structure and the project milestones.

We have considered all potential sources of failures that each work package may potentially come across. There are potential risks for every aspect of the structure, hardware, and software. With the risk assessment, we were more aware and more prepared to avoid the risks as well as mitigate the problems that have occurred.

We created the Smart Greenhouse with the total cost of $708.48. Several tests on structure, hardware, and software have been done throughout the entire course of Spring 2020 to make sure all features perform as expected. From the result of testing, some parts have shown to work well, some have been replaced, and there have been a lot of modifications to our circuits and programs to better the system.

The Smart Greenhouse was completed and has been working efficiently with no issues. The complete product comes with the structure that has an isolated environment with everything implemented inside. The electronics components are secured inside an electrical box, and the sensors and other hardware components are built in appropriate places. The automation feature can control the environment accurately to our measurable metrics with the error of 5%. The Smart Greenhouse has shown to be working well according to the result of growing strawberry and sugar pea plants.

References

[1] Bhatt, Raj; Estrada, Gabriela; Nguyen, Thinh ‘Jay’; Valdivia, Xiomara. Produced by one or more members of CSUS 2019 - 2020 Senior Design Team Five.

[2] “Food Security.” *Ifpri.org*, www.ifpri.org/topic/food-security.

[3] Pantanahiran, Wichai. “The Impact of Climate Change on Food Security and Agriculture Production in the Pak Phanang River Basin, Thailand.” *2017 6th International Conference on Agro-Geoinformatics*, 21 Sept. 2017. *IEEE Xplore*, ieeexplore.ieee.org/document/8047028.

[4] “Food Security.” *USDA*, www.usda.gov/topics/food-and-nutrition/food-security.

[5] Shahzad, Umair. “Global Warming: Causes, Effects and Solutions.” ResearchGate, Durreesamin Journal, Aug. 2015, www.researchgate.net/publication/316691239_Global_Warming_Causes_Effects_and_Solutions

[6] “Climate Impacts on Agriculture and Food Supply.” *EPA*, Environmental Protection Agency, 6 Oct. 2016, 19january2017snapshot.epa.gov/climate-impacts/climate-impacts-agriculture-and-food-supply_.html.

[7] Faggian, Paola, and Riccardo Bonanno. “Raising Awareness on Climate-Change Related Hazards That Might Impact Electric Infrastructures.” *2018 AEIT International Annual Conference*, 17 Dec. 2018. *IEEE Xplore*, ieeexplore.ieee.org/document/8577318.

[8] Elgaali, Elgaali, and Zeyad Tarawneh. “Effects of Climate Change on River Flow Stochastic Approach.” *2019 Advances in Science and Engineering Technology International Conferences*, 16 May 2019. *IEEE Xplore*, ieeexplore.ieee.org/document/8714549.

[9] “Extreme Weather and Climate Change.” *Center for Climate and Energy Solutions*, 2019, www.c2es.org/content/extreme-weather-and-climate-change/.

[10] Biek, Megan A., et al. “Affordable Greenhouses: A Tool to Increase Farmers' Adaptive Capacity to Climate Change.” 2015 IEEE Global Humanitarian Technology Conference, 3 Dec. 2015. IEEE Xplore, ieeexplore.ieee.org/document/7343991.

[11] “Climate Impacts on Agriculture and Food Supply.” United States Environmental Protection Agency, 19 Jan. 2017, snapshot.epa.gov/climate-impacts/climate-impacts-agriculture-and-food-supply_.html.

[12] Elgaali, Elgaali, and Zeyad Tarawneh. “Effects of Climate Change on River Flow Stochastic Approach.” *2019 Advances in Science and Engineering Technology International Conferences*, 16 May 2019. *IEEE Xplore*, ieeexplore.ieee.org/document/8714549.

[13] Zhemukhov, Ruslan Sh., et al. “Modeling of Crop Productivity of Agricultural Crops with Climate Change at Regional Level and Issues of Disposal of Pollutants with Irrigated Land.” 2017 International Conference "Quality Management,Transport and Information Security, Information Technologies" , 30 Oct. 2017. IEEE Xplore, ieeexplore.ieee.org/document/8085897.

[14] Dunne, Daisy. “Rise in Insect Pests Under Climate Change to Hit Crop Yields, Study Says.” Carbon Brief Clear on Climate, www.carbonbrief.org/rise-in-insect-pests-under-climate-change-to-hit-crop-yields-study-says.

[15] “Frequency of Food Insecurity.” *USDA ERS - Frequency of Food Insecurity*, www.ers.usda.gov/topics/food-nutrition-assistance/food-security-in-the-us/frequency-of-food-insecurity/.

[16] Smith, Michael. “Who Are the World’s Food Insecure?”. USDA ERS .https://www.ers.usda.gov/amber-waves/2019/june/who-are-the-world-s-food-insecure-identifying-the-risk-factors-of-food-insecurity-around-the-world/

[17] Adafruit Industries. “DHT11 Basic Temperature-Humidity Sensor Extras.” Adafruit Industries Blog RSS, www.adafruit.com/product/386.

[18] “Acoptex.com.” *Electronics and RoboTechnics*, acoptex.com/project/179/basics-project-036a-soil-moisture-sensor-yl-69-fc-28-or-hl-69-at-acoptexcom/#sthash.t2GiP23A.dpbs.

[19] “Arduino Raspberry Pi, 5PCS Soil Moisture Sensor Module and 5 PCS Temperature Humidity Sensor with 20PIN Female to Female.” Amazon,

www.amazon.com/Kuman-Raspberry-Moisture-Temperature-Humidity/dp/B071NX3234/ref=sr_1_10?keywords=temperature and humidity sensor raspberry pi&qid=1570841376&sr=8-10.

[20]"MCP3008 - 8-Channel 10-Bit ADC With SPI Interface." *The Pi Hut*, thepihut.com/products/adafruit-mcp3008-8-channel-10-bit-adc-with-spi-interface.

[21] "MCP3008-I/P 8-Channel 10-Bit A/D Converter." *ProtoSupplies*, protosupplies.com/product/mcp3008-i-p-8-channel-10-bit-a-d-converter/.

[22] "SunFounder LCD2004 Module with 3.3V Backlight." *Amazon*, Goettsche Partners, 2011, www.amazon.com/gp/product/B071W8SW9R/ref=ppx_yo_dt_b_asin_title_o01_s00?ie=UTF8&psc=1.

[23] Gus. "Raspberry Pi LCD Tutorial." *Pi My Life Up*, 18 Sept. 2019, pimylifeup.com/raspberry-pi-lcd-16x2/.

[24] "DC 12V Cooling Fan," *Amazon*, 2011. [Online]. Available: https://www.amazon.com/gp/product/B07D493BDX/ref=ppx_yo_dt_b_asin_title_o00_s00?ie=UTF8&th=1.

[25] "LED Grow Light," *Amazon*, 2011. [Online]. Available: https://www.amazon.com/gp/product/B07R8TM9PW/ref=ppx_yo_dt_b_asin_title_o01_s00?ie=UTF8&psc=1.

[26] "MOUNTAIN_ARK 12V Water Pump," *Amazon*, 2011. [Online]. Available: https://www.amazon.com/gp/product/B073XMP2VX/ref=ppx_yo_dt_b_asin_title_o01_s01?ie=UTF8&psc=1.

[27] "Raspberry Pi 3 Model B," *Raspberry Pi*. [Online]. Available: https://www.raspberrypi.org/products/raspberry-pi-3-model-b-plus/.

[28] "Signcomplex 12V 8A Power Supply," *Amazon*, 2011. [Online]. Available: https://www.amazon.com/gp/product/B075R3RW6J/ref=ppx_yo_dt_b_asin_title_o01_s00?ie=UTF8&psc=1.

[29] "DROK 5V 3A Buck Converter," *Amazon*, 2011. [Online]. Available: https://www.amazon.com/gp/product/B0758ZTS61/ref=ppx_yo_dt_b_asin_title_o01_s00?ie=UTF8&psc=1.

[30] Homedepot.com

[31] Adafruit Industries. "Adafruit TSL2591 High Dynamic Range Digital Light Sensor." Adafruit Industries Blog RSS, www.adafruit.com/product/1980.

[32] "Ultrasonic sensor- HC-SR04," *SparkFun Electronics*. [Online]. Available: https://www.sparkfun.com/products/15569.

[33] "Ultrasonic Sensor Tutorial," *The Geek Pub*, 15-Dec-2019. [Online]. Available: https://www.thegeekpub.com/235425/arduino-ultrasonic-sensor-tutorial/.

[34] "100W Car Heater," *Amazon*, 2011. [Online]. Available: https://www.amazon.com/gp/product/B07MDN81C3/ref=ppx_yo_dt_b_asin_title_o00_s00?ie=UTF8&psc=1.

[35] "Aroma2go Diffuser," *Aroma2go*. [Online]. Available: http://aroma2go.com/aroma-diffusers/arrival-compact-silent-essential-oil-diffuser-w-led-changing-lights.html.

[36] "Ultrasonic Mist Maker," *ElectroSchematics.com*, 24-Jun-2019. [Online]. Available: https://www.electroschematics.com/ultrasonic-mist-maker/.

[37] Kuklovskai͡a Elizaveta, "GDSTime Cooling Fans," *Amazon*. [Online]. Available: https://www.amazon.com/dp/B082HJ7BLL/ref=psd

c_11036291_t3_B07D493BDX?th=1. [Accessed: 13-Apr-2020].

[38] “8 Channel 5V Relay Module,” *Wiki*. [Online]. Available: http://wiki.sunfounder.cc/index.php?title=8_Channel_5V_Relay_Module.

[39] “Lesson 4 Relay Module,” *SunFounder*. [Online]. Available: https://www.sunfounder.com/learn/sensor-kit-v2-0-for-b/lesson-4-relay-module-sensor-kit-v2-0-for-b.html.

[40] “Menzo 12V 360W Power Supply,” *Amazon*, 2011. [Online]. Available: https://www.amazon.com/gp/product/B06VWV5YCH/ref=ppx_yo_dt_b_asin_title_o00_s01?ie=UTF8&psc=1.

[41] Adafruit. “Adafruit/Circuitpython.” GitHub, 26 Feb. 2020, github.com/adafruit/circuitpython.

[42] Chandler, Nathan. “What Is Relative Humidity and How Does It Affect How I Feel Outside?” *HowStuffWorks Science*, HowStuffWorks, 27 Jan. 2020, science.howstuffworks.com/nature/climate-weather/atmospheric/question651.htm.

[43] “Blynk.” *Blynk*, blynk.io/.

[44] “Raspberry Pi and Blynk: How to Use Blynk App with Raspberry Pi.” *Thetips4you*, 29 Dec. 2019, www.thetips4you.com/raspberry-pi-and-blynk-how-to-use-blynk-app-with-raspberry-pi/.

[45] Amazon.com

[46] *Smart Greenhouse Market by Type (Hydroponic and Non-Hydroponic), Technology (HVAC, LED Grow Light, Irrigation System, Material Handling, Valves, Control System, and Sensor & Camera), and Geography - Global Forecast to 2023.* MarketsandMarkets Research Private Ltd., 2019.

[46] “How Much Does It Cost To Build A Greenhouse?” *2020 Greenhouse Building Cost | Build Your Own Greenhouse*, www.improvenet.com/r/costs-and-prices/greenhouse-building-cost-estimator.

GLOSSARY

Climate change - The changes in the Earth’s climate due to varying weather patterns for long periods of time.

Food security - The measure of food availability and individuals’ ability to access it.

Agricultural sector - Comprises establishments that grow crops, raise animals, harvest fish, and other animals on farms, ranches, or natural habitats.

Greenhouse effect - Radioactive gases in Earth’s atmosphere trap warmth to the surface of the planet.

Carbon dioxide - A gas that is produced by burning carbon, organic compounds, and respiration.

Microcontroller - A computer on a compact integrated circuit consists of processor, memory, and I/O peripherals.

Hygrometer - An instrument used to measure the amount of humidity and water in a atmosphere

Python library - A collection of functions and methods allowing a person to perform calls when writing code

Relative humidity - The ratio of the partial pressure of water vapor to the equilibrium vapor pressure of water at a given temperature

DO - Digital Output. A signal with a value at a specified interval of time.

Printed Circuit Board (PCB) - A board that mechanically supports and electrically connects electronic components.

Raspberry Pi - A low-cost single-board computer that enables people to learn computing and programming computer languages such as Java, C, and Python.

Python - An object oriented, high level, general purpose programming language.

HTML - Hypertext Markup Language. HTML is a standard markup language that provides the structure of a Web page.

CSS - Cascading Style Sheet, manages the presentation of HTML elements on screen, paper, or in other media.

User interface - UI. Everything designed to support the interaction between human and computer.

Graphical User Interface - GUI. A type of user interface that provides graphical elements to make it easier for humans to interact with computers.

Automation - A process where little to no human assistance is needed to complete a task

Analog-to-digital converter - ADC. A device to convert analog signal to digital data.

pH - A scale to specify how acidic a solution is

Heating, ventilation, and air conditioning systems - HVAC. Technology that controls indoor environments that affect temperature and air quality

Relays - Relays open and close circuit electromechanically or electronically. They are connected to a microcontroller to control them.

Lux - An International System of Unit measurement of illuminance

I^2C - A serial protocol to connect devices such as microcontrollers

Compound Annual Growth Rate (CAGR) – The rate of return for an investment to grow.

Hydroponic - A method of growing plants without soil and using a water solvent with nutrients instead

Short-circuit - This is a circuit which allows the current to spike up very large causing damage to circuit components.

Relays - Relays open and close circuits based on the signal voltage from the microcontroller allowing the microcontroller to control high voltage components.

Soldering iron - It is used to connect two or multiple wires together by applying heat to melt the solder onto the connection.

Power management - Planning and supplying power to the system correctly and efficiently.

Mitigation plans - Process of development and implementing plans to reduce risk. This includes identifying risk, evaluating risk, and creating fallback plans.

Printed Circuit Boards (PCBs) - A board that mechanically supports and electrically connects electronic components.

Power Supply - an electrical device that delivers electric power to electrical loads.

Buck Converter - DC to DC power converter - is a device to step-down the voltage from its input to the output.

Ultrasonic Sensor - is to measure distance based on sending and receiving ultrasonic signals.

Appendix A
User Manual

Materials Included

- Fully built greenhouse with wood framing, lockable wheels, and insulated tarp
- 1 Light Level Sensor
- 1 Temperature/Humidity Sensor
- 1 Water Level Sensor
- 7 Soil Moisture Sensors
- 2 LED Strips
- 2 Fans
- 1 Heater
- 1 Humidifier
- 1 3-gallon Water Tank
- 1 Water Pump
- 7 Drip Irrigation Bubblers
- 2 Large Plant Containers
- 1 Small Plant Container
- 1 Electrical Box
- 12V 360W Power Supply

Steps for Setup

1. Place Smart Greenhouse in preferred location (Inside or Outside)
2. Connect Smart Greenhouse to power
3. Open the roof top for access into the Smart Greenhouse
4. Plant preferred plants into plant containers
5. Download the Smart Greenhouse Application
6. Connect Smart Greenhouse to same Wi-Fi network as device that has the Smart Greenhouse Application
7. Happy Growing!

How to Operate Smart Greenhouse

1. To get started, confirm all the components are plugged in by checking the wired connections. This can be done by opening the roof of the greenhouse and opening the electrical box cover. Plug in the main power supply in the electrical box to a wall socket.
2. To secure the greenhouse in one place and position, twist the key lock of each wheel. We recommend locking the opposite corner wheels.
3. Add water to refill the water tank.
4. Add soil in the plant containers and plant seeds or a new plant into the soil.
5. For plant pruning, access the pots by opening the greenhouse roof. Find picture directions in appendix D\
6. Install the Smart Greenhouse Application (see steps below).

Using Smart Greenhouse Application

1. Download Smart Greenhouse from the App Store.

Fig. A1. Smart Greenhouse Application [1]

2. When downloading the application, it will have 3 pages consisting of the Data Display, Control, and Live Reading.

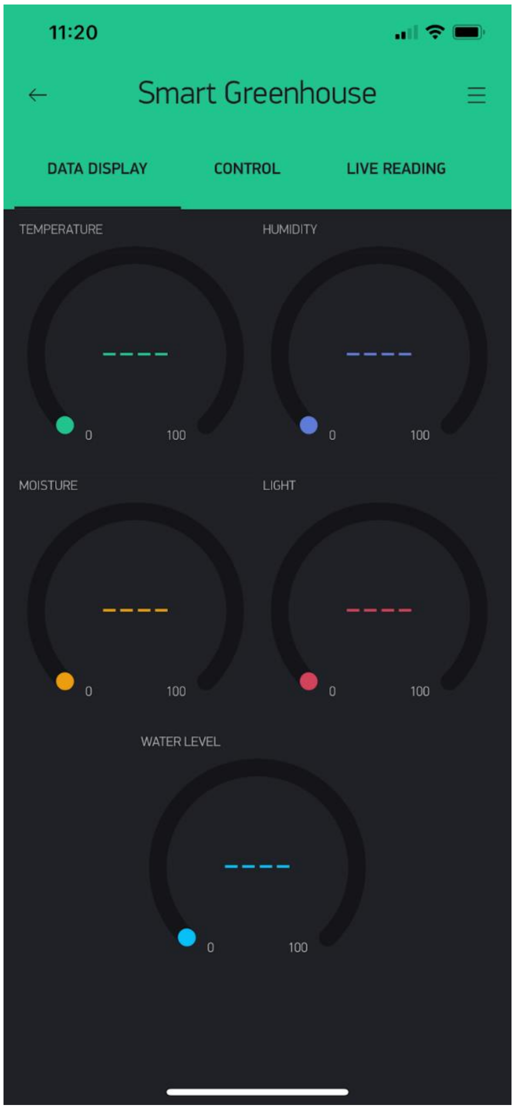


Fig. A2. Application Tabs [1]

3. The Data Display will initially have no values. It has 5 gauges for all the sensors in the Smart Greenhouse. It includes the data of the temperature sensor, humidity sensor, moisture sensor, light level sensor, and water level sensor. When the Smart Greenhouse is connected to power and the sensors begin reading data, it will be displayed automatically on this screen. It is live, so anytime the sensors get a new reading it will be reflected on this screen. It is on a spectrum of 0 °F - 100 °F for the temperature, and 0% - 100% for all the other sensors.

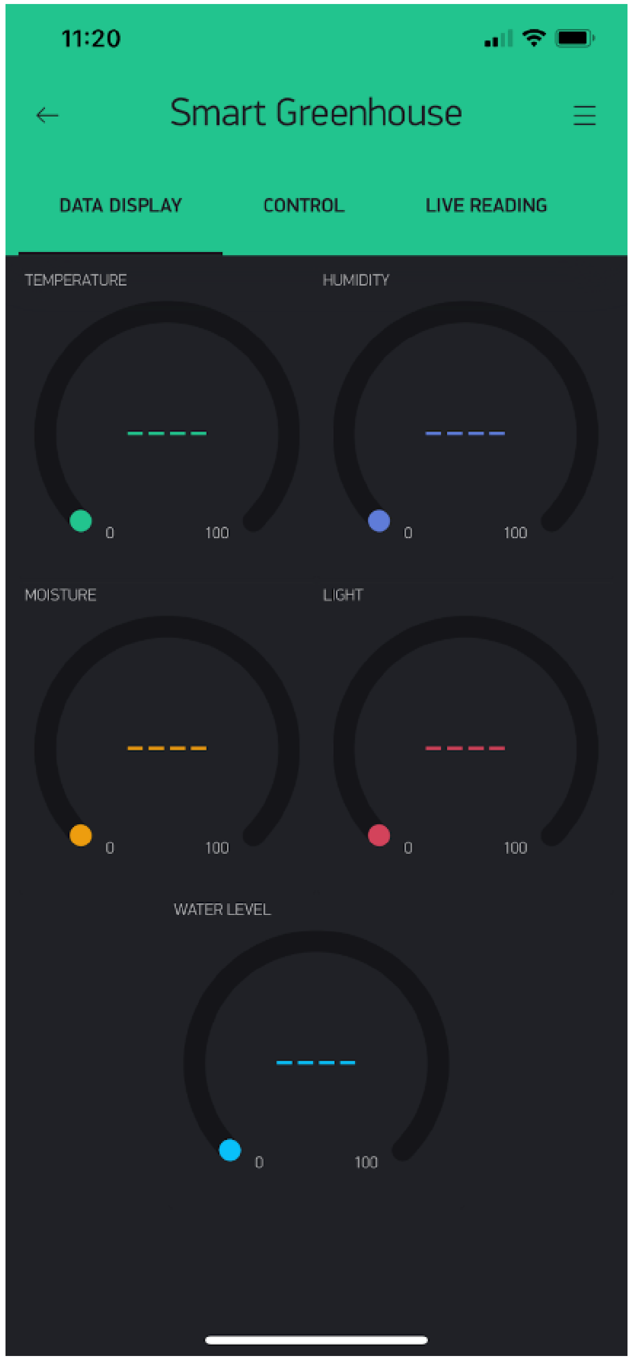


Fig. A3. Data Display Tab [1]

4. The Control page has all the buttons to manually turn on and off the hardware components. Initially, the buttons will be set to off. The hardware components that can remotely be changed include the fans, LEDs, humidifier, heater, and the water pump. There is no need to physically go to the Smart Greenhouse to turn it off or on. The gauges in the Data Display page will be a great guide to see how the Smart Greenhouse is doing, and if needed, you will be able to turn things on or off on your own time, remotely.

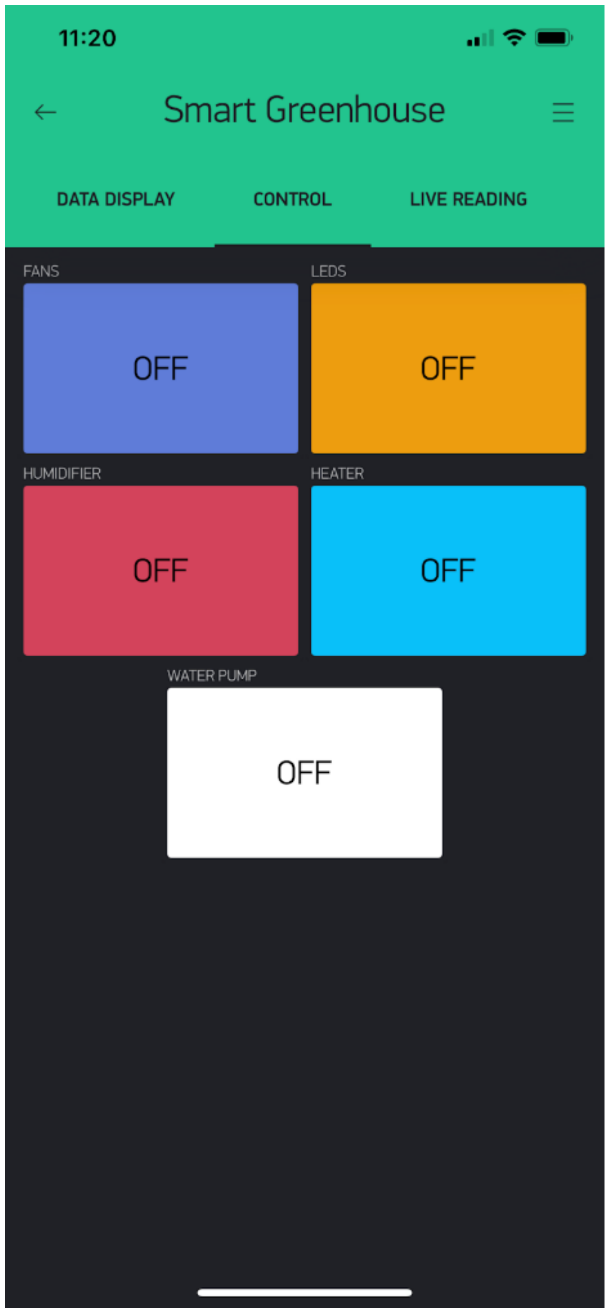


Fig. A4. Control Tab [1]

5. The Live Readings screen will provide the sensor readings over time. Since the Data Display page only shows the sensor reading live at that instant moment, the Live Readings screen will provide in detail how the Smart Greenhouse is doing. It will initially have no data, but as soon as the Smart Greenhouse is connected to power it will have live readings and it will keep track of the data between 1 hour and up to 3 months. You also have the option to export your Live Readings to an excel sheet, if you would like to see the statistics of your Smart Greenhouse.

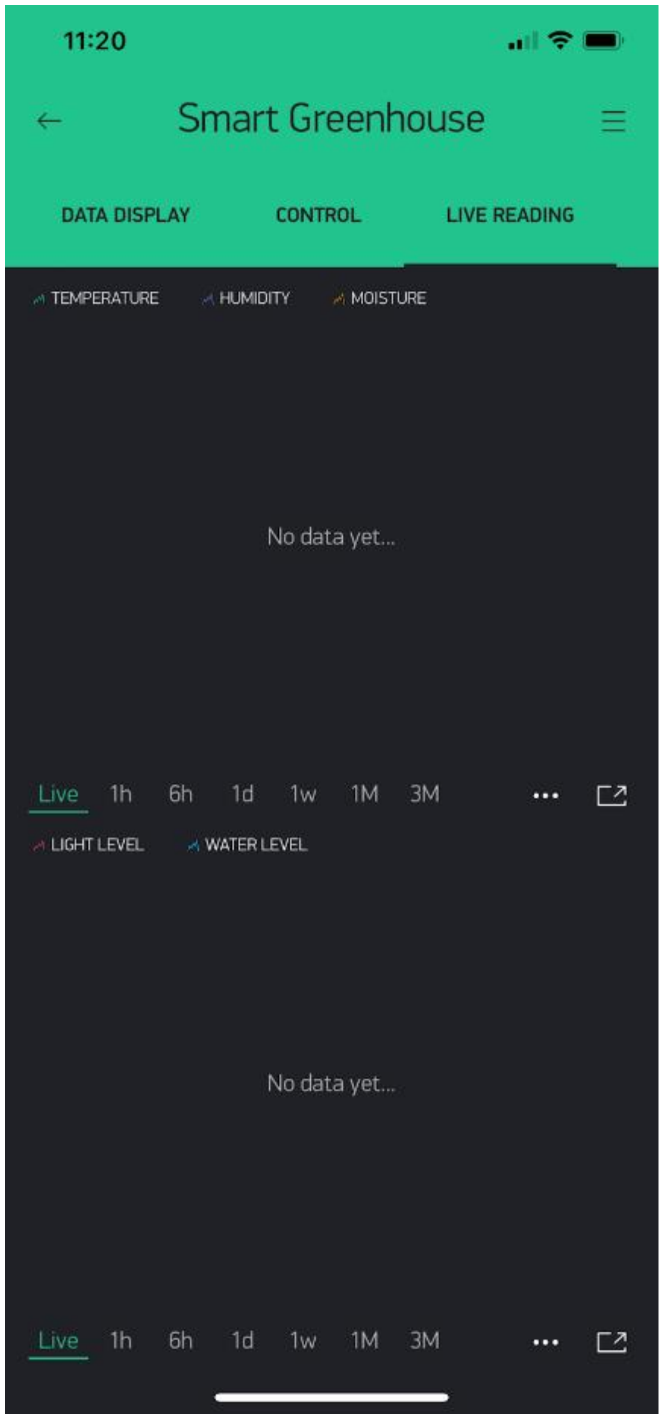


Fig. A5. Live Reading Tab [1]

Specifications of Greenhouse

TABLE A1
SPECIFICATIONS OF GREENHOUSE

Product Name	Smart Greenhouse
Size Dimensions	30’’ W x 45’’ L x 30’’ H
Max Weight Supported	360 lbs
Model Version	1.0.a
Voltage Input	110 V AC
Frame Material	Wood
Tarp Material	Heavy duty vinyl film

Specifications of Sensors/Hardware in Greenhouse

TABLE A2
SPECIFICATIONS OF SENSORS/HARDWARE

Sensor/Hardware	Specific Name/Type
Temperature/Humidity Sensor	Adafruit DHT11 Sensor
Soil Moisture Sensor	YL-69 Sensor
Light Level Sensor	Adafruit TSL2591 High Dynamic Range Digital Light Sensor
Water Level Sensor	Ultrasonic HC-SR04
Power Supply	MENZO 12V 30A DC Universal Regulated Switching Power Supply
Heater	100W 12V Energy Saving PTC Fan Air Heater
Ventilation Fans	Strong Quiet 12025 Fan 120x120x25mm 12cm 120mm DC 12V 1600 RPM
Electrical Box Cooling Fans	GDS Time 80x80x25mm DC 12V 2600RPM
LED Growing Light	Full Spectrum, 6.56ft LED Grow Light Strip
Water Pump	MOUNTAIN_ARK 63 GPH Submersible Water Pump DC 12V 3.6W 9.8ft Lift
Humidifier	Aroma2go USB Humidifier

HARDWARE

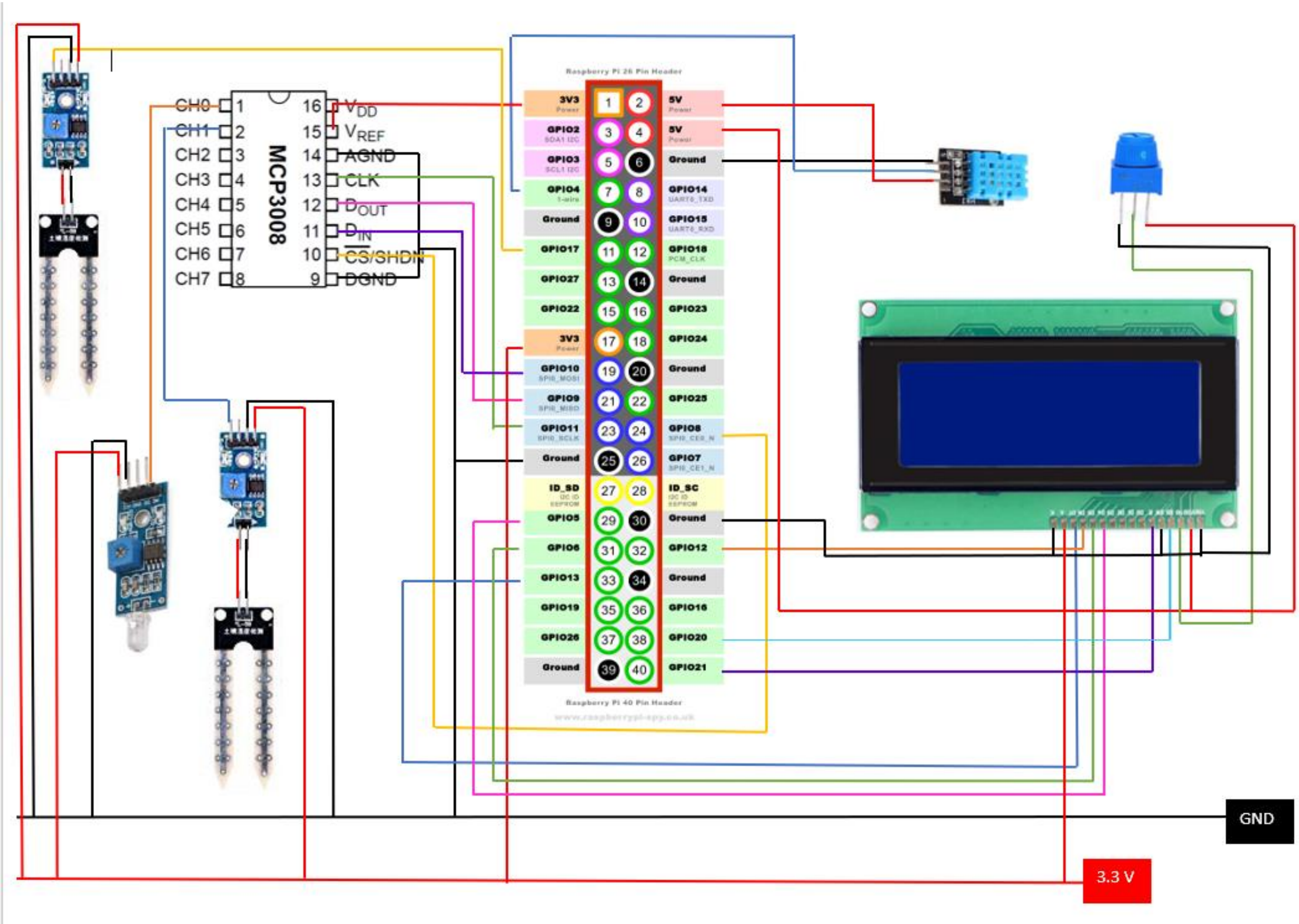


Fig. B1. Fall 2019 Raspberry Pi Connections Schematic [1]

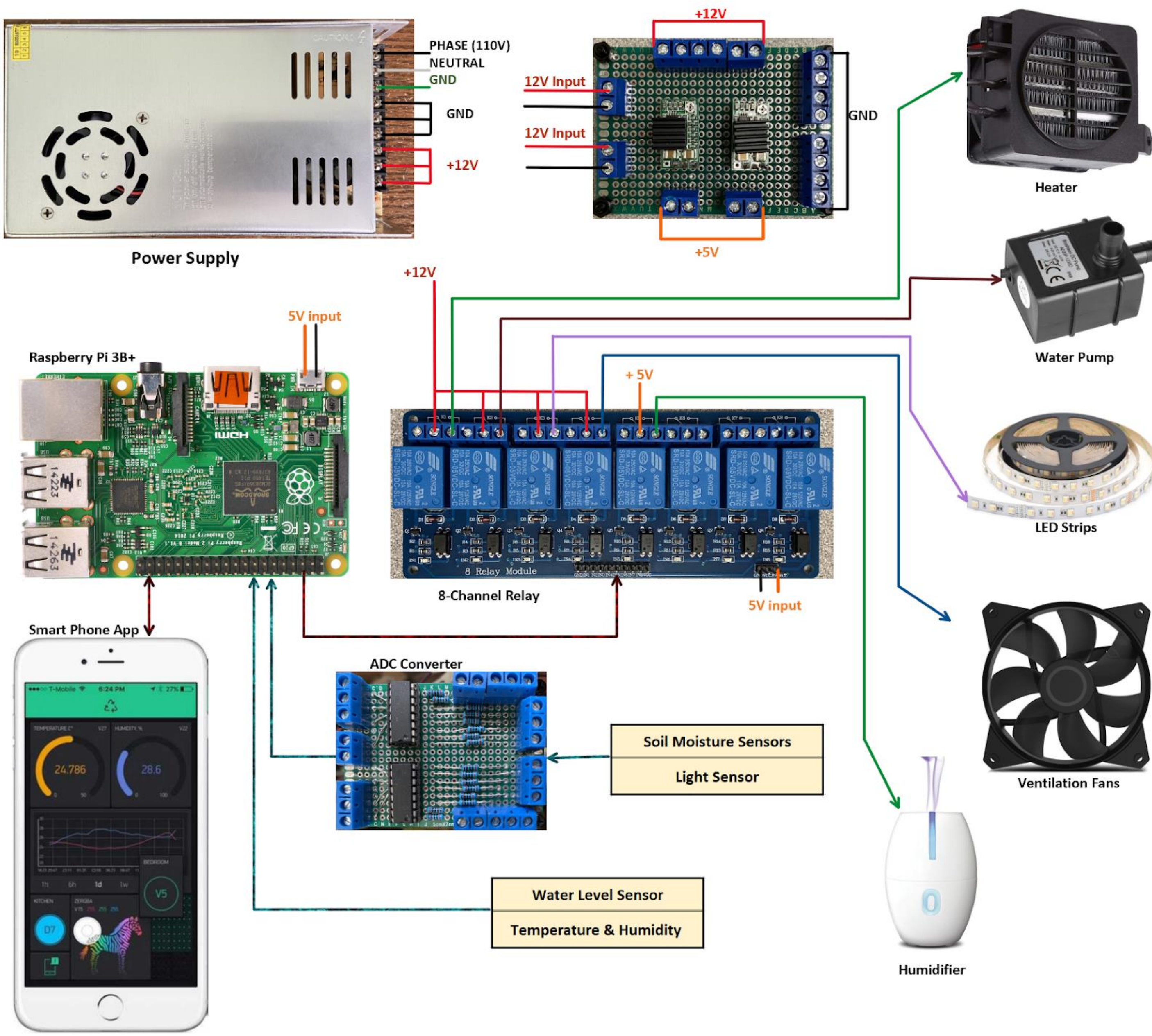


Fig. B2. Spring 2020 Prototype Hardware Connections [1]

Appendix C
Software

```
This program reads the light sensor data from
the light sensorn by reading data from the MCP3008 ADC

import Adafruit GPIO SPI library
import Adafruit MCP3008 library
import time library

initialize the SPI port
initialize the SPI device
initialize the MCP ADC chip

while True:
    read the light sensor from channel 7 of the MCP3008 ADC chip
    print the data read
    sleep for 3 seconds
```

Fig. C1. Pseudocode for Reading and Outputting Light Sensor Data [1]

```
####################################################################
##################### CUSTOMIZEABLE SETTINGS #####################
####################################################################
SETTINGS = {
    "LIGHT_CHANNEL":             7,    # of MCP3008
    "LIGHT_THRESHOLD":          60,    # Light threshold, below this value, the LEDs will turn on
    "FAN_GPIO":                  5,    # GPIO Number (BCM) for relay
    "FAN_TIME":                 20,    # Seconds, how long the fan should be on
    "HEATER_GPIO":              26,    # GPIO Number (BCM) for relay
    "HEATER_TIME":              10,    # Seconds, how long the heater should be on
    "HUMIDIFIER_GPIO":          19,    # GPIO Number (BCM) for relay
    "HUMIDIFIER_TIME":          10,    # Seconds, how long the humidifier should be on
    "LED_GPIO":                 6,     # GPIO Number (BCM) for relay
    "TEMP_HIGH_THRESHOLD":      30,    # in Celsius (~85F). Above this value, the fan will turn on
    "TEMP_LOW_THRESHOLD":       -1,    # in Celsius (~30F). Below this value, the heater will turn on
    "HUMIDITY_HIGH_THRESHOLD":  60.0,  # in RH, Below this value, the fan will turn on
    "HUMIDITY_LOW_THRESHOLD":   40,    # in RH, Below this value, the humidifier will turn on
    "MOISTURE_THRESHOLD":      500,    # Below this value, the water pump will turn on
    "WATERING_TIME":            10,    # Seconds, how long the pump should be turned on
    "WATER_PUMP_GPIO":          13,    # GPIO Number (BCM) for relay
```

Fig. C2. Automation Settings [1]

```
This function gets the temperature and humidity data from
our temperature and humidity sensors and turns on the corresponding
hardware components to automatically regulate the temperature and humditity.

function getTempHumidity():
    read the temperature (in Celsius) and humidity (in Relative Humidity)0
    calculate the the temperature in Fahrenheit

    if the temperature exceeds the highest threshold for the temperature
        set up the GPIO channel as output and send a LOW signal to turn on the fan from the relay
        leave the fans on for a designated amount of time
        turn the fans off

    else if the temperature is less than the lowest threshold for temperature
        set up the GPIO channel as output and send a LOW signal to turn on the heater from the relay
        leave the heater on for a designated amount of time
        turn the heater off

    if the humidity exceeds the highest threshold for the humidity
        set up the GPIO channel as output and send a LOW signal to turn on the fans from the relay
        leave the fans on for a designated amount of time
        turn the heater off

    if the humidity is less than the lowest threshold for the humidity
        set up the GPIO channel as output and send a LOW signal to turn on the humidifier from the relay
        leave the humidifier on for a designated amount of time
        turn the humidifier off

    return
```

Fig. C3. Pseudocode for Temperature and Humidity Automation [1]

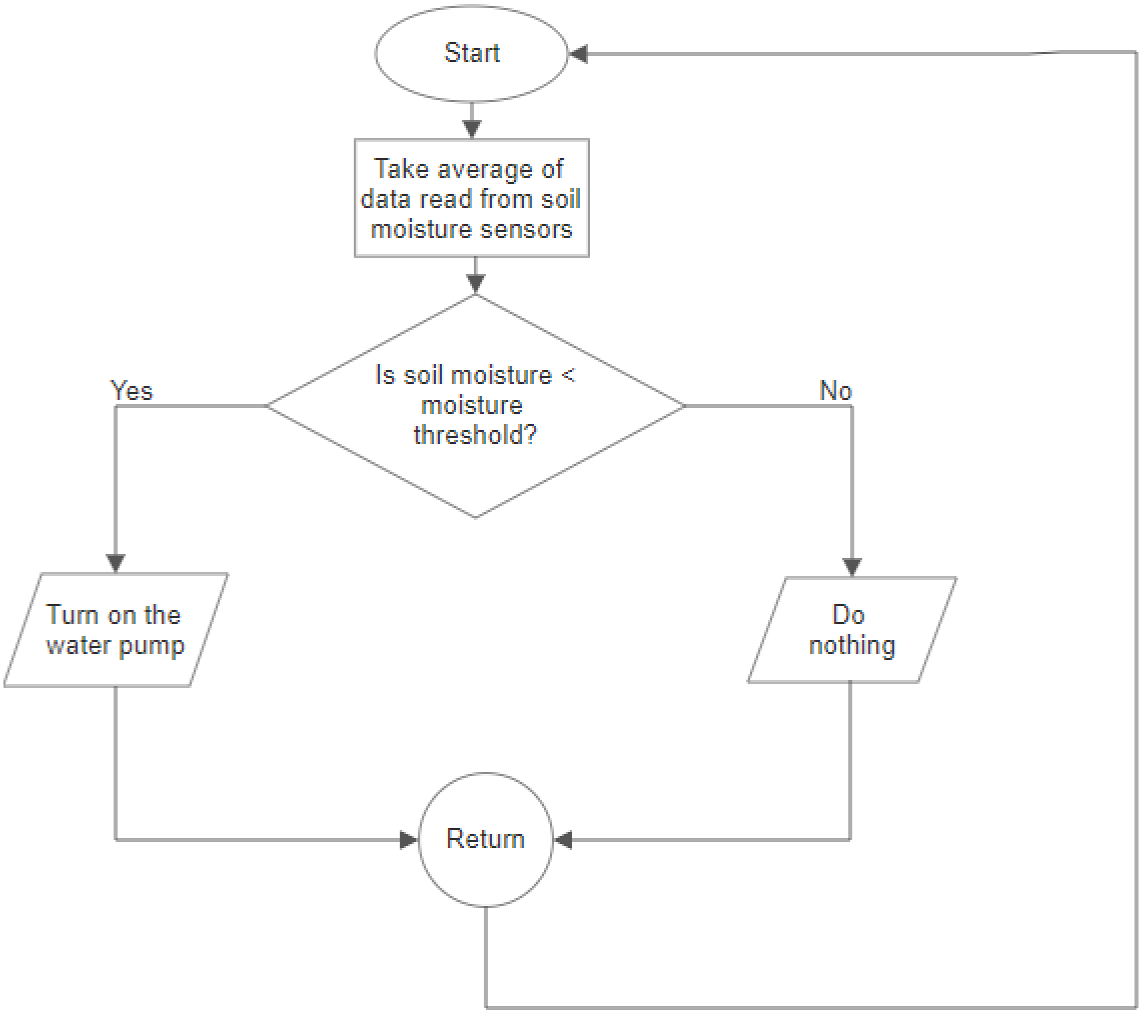


Fig. C4. Logic flowchart for automated soil moisture function [1]

APPENDIX D
STRUCTURE

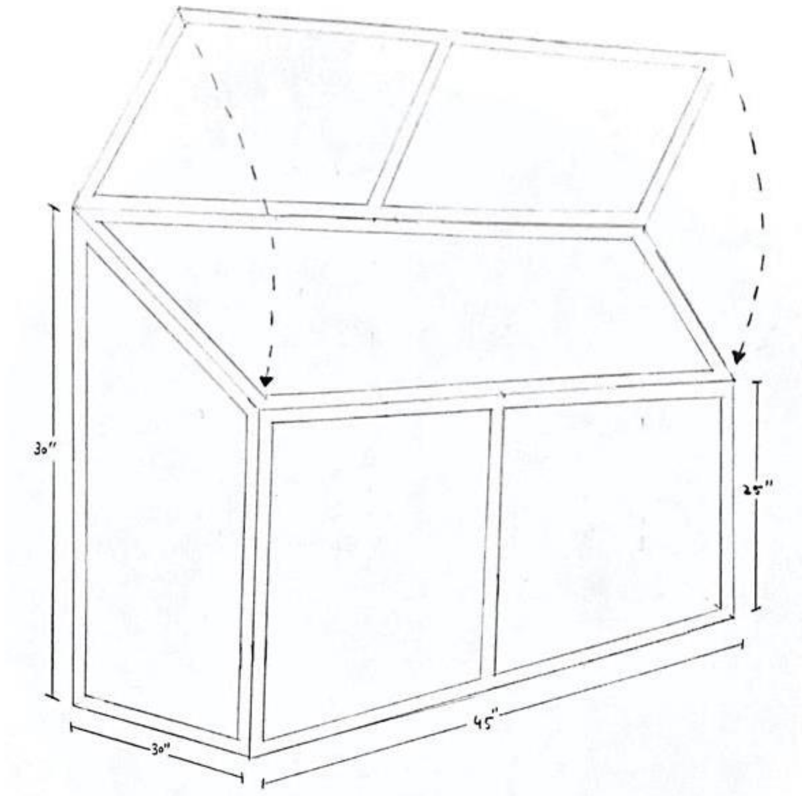


Fig. D1. How to Open/Close Smart Greenhouse [1]

Appendix E
Funding

TABLE E1
COMPONENTS PURCHASED

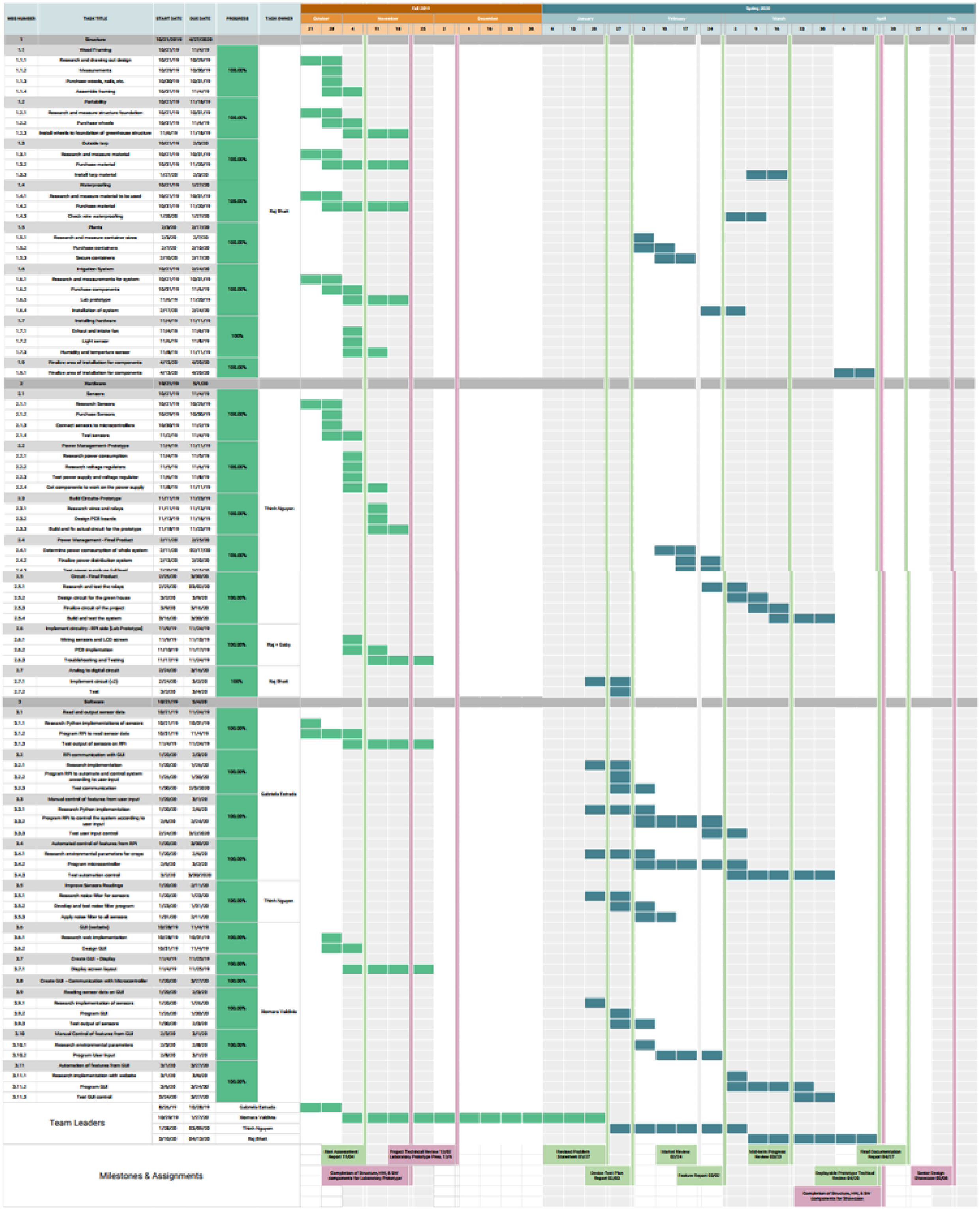
Part Name	**Cost Per Unit**	**Units Bought**	**Total Cost**
Raspberry Pi	$42.89	1	$42.89
Humidity, temperature, & soil moisture sensor pack	$16.99	1	$16.99
Photosensitive Sensor Module	$5.89	1	$5.89
Fans	$11.99	1	$11.99
Plant Growing Light	$12.99	1	$12.99
Storage Kit	$12.25	1	$12.25
Wires	$12.99	1	$12.99
Heat Shrink Tubes Wire Wrap	$7.89	1	$7.89
Wire Connectors Housing Terminal	$15.93	1	$15.93
Pump Control Valves	$7.28	1	$7.28
Water Pump	$17.99	1	$17.99
Mini Voltage Reducer	$8.99	1	$8.99
5V 8 Channel Relay Shield Module	$8.79	1	$8.79
12V Power Supply	$20.98	1	$20.98
LCD screen for display	$7.69	1	$7.69
GPIO Reference Board	$5.99	1	$5.99
2x2x8ft	$2.28	6	$13.68
2x3x8ft	$1.98	2	$3.96
screw 8 x - 1/4	$2.30	1	$2.30
Hinges	$2.98	2	$5.96
3" Screws	$5.97	1	$5.97
Cart wheel	$3.98	4	$15.92
4 port irrigation hub	$3.92	1	$3.92
Screws 8x1	$2.30	1	$2.30

Pot 36"	$10.97	3	$32.91
Hinges (for circuit box)	$2.18	1	$2.18
Metal plate	$2.88	1	$2.88
Pot 24"	$7.97	1	$7.97
Red wood 8'	$5.00	2	$10.00
20' Tubing 3/8	$6.62	1	$6.62
20' tubing 1/4	$3.95	1	$3.95
Mister	$2.97	1	$2.97
Connect 3/8 to 1/2	$4.41	2	$8.82
Valve 3/8 to 3/8	$9.94	1	$9.94
Plumbing tape	$1.62	1	$1.62
1/2"connector	$0.42	1	$0.42
1/2" Riser	$1.63	1	$1.63
T Join	$0.42	1	$0.42
Flowmeter Water Flow Counter Meter	$7.99	1	$7.99
L brackets 24 pcs	$13.49	1	$13.49
Switches 15pcs	$6.99	1	$6.99
Copper tape (heat spread)	$5.88	1	$5.88
Cable cover	$12.59	1	$12.59
Waterproofing silicone	$4.89	1	$4.89
12v dc power 360W and 30A	$18.99	1	$18.99
Heater (new)	$18.88	1	$20.65
Outside trap	$23.47	1	$23.47
Foam closer - Airtight	$1.98	1	$1.98
Irrigation sprinklers (4/pk) total 8 heads	$4.47	2	$8.94
16-gauge wire	$6.50	1	$6.50
2 pin connectors	$7.60	1	$7.60
Heatsink	$5.99	1	$5.99
PCB mount terminal	$7.99	1	$7.99
RPi	$42.99	1	$42.99

Temp. and Humidity meter	$8.59	1	$8.59
Soil moisture sensors (5 pieces total)	$7.49	1	$7.49
Green Acres Nursery & Supply (total)	$13.50	1	$13.50
Home depot (total)	$6.00	1	$6.00
Mist maker	$7.99	1	$7.99
Fan 50mm (2 pack)	$6.59	1	$6.59
clear silicone	$4.47	1	$4.47
Amazon order 6.2	$6.51	1	$6.51
Amazon order 6.3	$13.48	1	$13.48
Vent grill (Home depot)	$4.74	1	$4.74
Tank (container store)	$5.83	1	$5.83
Parking ticket	$62.50	1	$62.50
		Total	$708.48

Appendix F
Project Timeline

TABLE F1
SMART GREENHOUSE PROJECT TIMELINE



Appendix G
Resumes

This page is intentionally left blank.

Raj Bhatt

EDUCATION

Computer Engineering
GPA 3.1

California State University, Sacramento Expected May 2020

Focus in Circuit designing with in-depth understanding of hardware and software.

- **Relevant Courses:** Computer Organization, Computer Logic design, Design of Microcomputer System, Computer Hardware design (Study of Intel & Motorola architecture), Data structure in Java & C++, Network Analysis (Study of Three Phasor Power, Sinusoid and Steady State), Cmos & VLSI design, and probability & Random signal.
- **Programming language:** C/C++, Java, Verilog, VDHL, PSpice, Assembly, TCL and Machine language.
- **Software:** Windows, Linux, iOS, Cadence and Putty
- **Hardware:** Oscilloscope, Digital Multi-Meter, Logic Analyzer, PCB, Breadboard, FPGAs, ADC/DCA and Raspberry Pi.
- **Foreign Languages:** Hindi (advance) and Gujarati (advance).
- **Extracurricular activity:** Member for Tau Beta Pi – Upsilon. Sacramento State Engineering honors society.

EXPERIENCE

Verification Engineer Intern

Microchip Technology Inc. January 2020 – Present

Providing design verification support to the department of the Architecture Co-Verification.

- Develop scripts to verify the design's different edge cases, PCIe lanes, and test schedules .
- Collect and analyze data for future modification for the pre silicon design.
- Working with different departments to troubleshoot problems and verifying design implementation and program simulation.

Network Engineer Intern

Sacramento Municipal Utility District – SMUD June 2019 – January 2020

Providing Network and Audit support to the department of Operational Technology – Networking.

- Maintaining and configuring servers for future use.
- Assist with internal and federal audits by providing information for network assets
- Maintaining evidence file for each change made to server and its individual components for future reference
- Maintaining and modifying the SolarWinds website as per the asset's configuration.

IT Support Specialist

California State University, Sacramento Nov 2017 – August 2019

Providing Technology and Web support to the departments which are affiliated with Student Affairs.

- Configure/upgrade computers for newly hired staff and departmental upgrades to meet required specs and needs.
- Troubleshoot hardware, network and printers' issues with external device issue which requires drivers and scripts files.
- Generate how-to guide of systematic approach to problems thus minimizing system/equipment down time.
- Oversee all department computers to keep them up to the stands with hardware and software.
- Order hardware and equipment at best price possible to meet the specs and department needs.

PROJECTS

Senior Design Project – Automated Greenhouse
Aug 2019 – Present

Our solution is to build a user-friendly, automated, portable, greenhouse system that allows all to easily grow and control aspects of growing crops.

- Designed and built the greenhouse structure with persistent measurement.
- Implementation circuit to fit is small space, and water resistant so it able to function in different climate condition.
- Implementation 16 channel Analog to Digital circuit (ADC) on PCB.

THINH ‘JAY’ NGUYEN

OBJECTIVE:

Actively seeking an internship in the areas of Hardware, Firmware, or Software Engineering.

EDUCATION:

Bachelor of Science, Electrical & Electronic Engineering Expected: May 2020
California State University, Sacramento, CA GPA: 3.98

Highlight Courses:

- Electronic 1 & 2: Op-amps, MOSFETS, BJTs, and OTAs
- Physical Electronic: Semiconductor physics
- Machine Vision: Digital Image Processing & Machine Vision
- Modern Communication system
- Robotics: Principles of robotics and design of robots
- Microprocessor – Computer Interfacing

WORK EXPERIENCE:

Math Tutor: Algebra, Calculus, and Differential Eqn
California State University, Sacramento, CA 08/2018 – 12/2018
Cosumnes River College, Sacramento, CA 08/2016 – 12/2016

- Give advice on studying skills and help students how to engage and solve problems.
- Collaborate with a team of faculty to develop a better program for students in need.

Receptionist
Profile Nails, Roseville, CA 05/2014 – 07/2018

- Handle phone calls, answer questions, arrange appointments, organize schedules.
- Take care of the supplies and restock as needed

SKILLS:

Programming: C, Python, JavaScript, x86 Assembly, HTML/CSS, PID Control, Machine Vision
Programs/Applications: Matlab, PSpice, Arduino IDE, Quartus Prime, Attollic TrueStudio, STM32CUBEMX, DOS, Windows (XP, Vista, 8.1, 10), Linux (Debian), VMWare, OpenCV

RELATED PROJECTS:

Senior Design Project

- *Smart Greenhouse:* Collaborated with a team of 4 to develop a smart greenhouse system that can grow crops automatically with no human physical work. The Smart Greenhouse was built with fully automated features with a GUI that provided the environment data and allowed the user to adjust the environment as will. Directly working with power management, circuit design of the project, and sensors accuracy.

Computer Interfacing Projects

- *Solar Tracker with Automation*: Cooperated with two group members to design and build a solar tracker with a website data streaming capability. The solar tracker would turn to the direction of most light intensity and report data to a webpage.
- *Serial Communication and Data Streaming:* Developed a program to perform serial communication between Raspberry Pi 3 and Parallax Propeller 2. Sensor Data collected from the Propeller would be sent to the RPi continuously; then the RPi would update them to a web page in real time using JavaScript.

Robotics Project

- *Maze Solver:* Developed an automated robot that can solve a maze utilizing concepts of PID controller and machine learning. Responsibility includes designing, building, and programming the maze solver using Arduino IDE and Python. PID controller assisted the robot to perform motions such as moving forward or turning left and right accurately. PID controller was also used to keep the robot to stay in between the walls. Machine learning helped the robot to navigate and move to the end point.

Hardware Design

- *Feedback Amplifier System:* Designed and built a Feedback Amplifier System using an Operational Transconductance Amplifier that met a required design specifications of input & output resistance, voltage gain, and voltage swing capability.

Machine Vision Project

- *Face Recognition:* Programmed on the Raspberry Pi using Python and OpenCV to detect faces using Haar Cascade Classifier and recognize the person in a real-time camera recording.

AWARDS/CLUBS:

- Dean's Honor List
- MESA Engineering Program (MEP)
- American Indian Science & Engineering Society (AISES)

REFERENCES:

- Mariano Contreras, P.E., Senior Electrical Engineer – (609) 606 2826

Gabriela Estrada

Education

California State University, Sacramento *Expected Graduation: May 2020*
Major: Computer Engineering Major GPA: 3.2/4.0

Work Experience

Intel Corporation, Graphics Product Support Engineer Intern December 2016 – Present

- Assist in analyzing then resolving external customer issues by diagnosing the root cause.
- Configure and set up systems for issue-replication and troubleshooting.
- Prioritize and manage several open customer cases simultaneously.
- Collaborate with customers, support team, and development teams to define, evaluate, and improve graphics products.
- Author technical white papers, support articles, and other documents that provide customers with technical specifications regarding graphics and graphics drivers.

Skills

Software: Waveform, OrCAD PSpice, Cadence, Multisim, MS-DOS, Arduino IDE, Python IDE, Eclipse IDE, Visual Studio Code, Android Studio, Microsoft Office
Hardware: Arduino, Raspberry Pi, Parallax Propeller Board, ChipKit Max32, Analog Discovery, ADC/DAC, FPGA, sensors, oscilloscopes
Programming Languages: Python, C, Verilog, VHDL, Java, Assembly, HTML, CSS

Relevant Courses

Systems Programming in UNIX	Programming Concepts and Methodology
Data Structures and Algorithm Analysis	Operating System Principles
Computer Networks and Internets	Advanced Logic Design
Electronics I	Network Analysis
Introduction to Computer Architecture	Computer Hardware System Design
Advanced Computer Organization	Computer Interfacing

Extracurricular Activities

Member, Tau Beta Pi California Upsilon Honors Engineering Society 2019 – Present
Member, Institute of Electrical and Electronics Engineers (IEE) 2018 – 2019
Member, Society of Hispanic Professional Engineers (SHPE) 2015 – 2019
Member, MESA Engineering Program (MEP) 2015 – Present
Member, California State University Louis and Stokes Minority Participation Program (CSU-LSAMP) 2015 – Present

Projects

Senior Design Project – Smart Greenhouse Fall 2019 – Spring 2020

- Implement/troubleshoot software in Python to read and output sensor data from a Raspberry Pi.
- Program Raspberry Pi to autonomously control the environment inside of the system.
- Design 8-channel ADC schematic and implement onto PCB using solder.
- Wire and troubleshoot hardware (sensors and microcontroller).

Remote Controlled Guard Bot Spring 2018

- Collaborated in a team to develop a bot that detects movement utilizing a Raspberry Pi.
- Controlled robot through a website taking advantage of TCP protocol for communication.
- Assisted with the design of the robot and coding the motors for the robot using Python.

Xiomara Valdivia

Education

Undergraduate at California State University, Sacramento *Expected Graduation: May 2020*
Major: Computer Engineering GPA: 3.58/4.0

Work Experience

Technical Support Engineer, PowerSchool HQ *April 20, 2020 - Present*

- New Hire
- Currently in training

AppleCare Home Technical Advisor, Apple *July 2, 2018 - April 7, 2020*

- Troubleshoot hardware and software issues regarding macOS, iOS, iPadOS, and watchOS devices
- Simultaneously log customer and issue information while researching databases for issue resolution.
- Use Bomgar to remotely support customers
- Over 95% customer satisfaction rate

Admissions and Outreach Assistant, CSUS *Sep. 25, 2017 - June 8, 2018*

- Collected and generated data graphs to convey our statistics from events, personal contact with businesses to collect donations, conduct presentations for recruitment, and assist in the planning committee.

First Year Experience Peer Mentor, CSUS *Aug. 21, 2017 - Dec. 8, 2017*

- Provided advising for graduation and general education requirements for 20 first year students, led classroom presentations, hosted office hours to assist students, and responded to students in crisis.

Residential Advisor for SMASH, UC Davis *June 19, 2017 - July 31, 2017*

- Summer Math and Science Honors Academy
- Assisted 60 students academically and through residential life, made daily reports of incidents and student behavior, point of contact between parents and students, and mentored in a hackathon.

Skills

Software: Windows, macOS, iOS, iPadOS, watchOS, Microsoft Office, Blynk API
Programming Languages: HTML, HTML 5, CSS, Java, C, Python, SQL
Hardware: Raspberry Pi, Arduino, FPGA, Oscilloscopes
Languages: Fluent in both Spanish and English

Highlights

Academic and Honors Achievements/Conferences Attended:

- Member of the Tau Beta Pi California Upsilon Engineering Honor Society
- Recipient of the National Action Council for Minorities in Engineering Scholarship
- Member of the Golden Key International Honour Society
- Selected to attend the Student Leadership Conference for MESA
- Attended Regional Leadership District Conference (RLDC) at Oregon State University in Spring 2016
- Attended Hispanic Engineer National Achievement Awards Conference (HENAAC) in Fall 2016
- Dean's Honor List - Spring 2016, Spring/Fall 2017, Spring/Fall 2018, Spring/Fall 2019