P2 Project: Autonomous Plant System

Presented to: Prof. Saeed Gazor & Majid Pahlevani

(Person(s)/Company to whom the report is submitted – Prof Saeed Gazor, Prof Majid Pahlevani & TA. Hamed Alizadeh Moghaddam)

P2 Proposal Report

APSC 200/293 ECE Engr Design & Practice II/Eng Communications F22

Computer Engineering

Faculty of Engineering and Applied Science

Queen's University

Presented by Team #12, Section 208

Arta Namjoo, 20292265

Carter Moore, 20295192

Erhowvosere Otubu, 20293052

Tao Nichol, 20297380

Will Pritchard, 20293510

Date of Submission: 04-12-2022

Statement of Originality

Our signatures below attest that this submission is our original work.

Following professional engineering practice, we bear the burden of proof for original work. Therefore, we have read the Policy on Academic Integrity posted on the Faculty of Engineering and Applied Science website (http://engineering.queensu.ca/policy/Honesty.html) and confirm that this work is in accordance with the Policy.

Individual 1:

Signature:	Sere O	Date:	2022-09-16
Name:	Erhowvosere Otubu	ID #:	20293052

Individual 2:

Signature:	C. Moore	Date:	2022-09-16
Name:	Carter Moore	ID #:	20295192

Individual 3:

Signature:		Date:	2022-09-16
Name:	Will Pritchard	ID #:	20293510

Individual 3:

Signature:	Tao Nichol	Date:	2022-09-16
Name:	Tao Nichol	ID #:	20297380

Individual 5:

Signature:		Date:	2022-09-16
Name:	Arta Namjoo	ID #:	20292265

The table below shows the individual contributions made in the sections of this report.

	Content Identify here the sections of the report for which you contributed content.	**Write-up** Identify the sections of the report that you wrote or edited.	**Editing** Identify the sections of the report that you edited.
Erhowvosere Otubu	Section 1 (1.1, 1.2), Section 3 (3.1), Decision Harm and Radicalization	Section 1 (1.1, 1.2), Section 3 (3.1), Decision Harm and Radicalization	Entire document
Will Pritchard	Section 2 (2.4), Section 3, (3.1, 3.2, 3.3, 3.4), Conclusion	Section 2 (2.4), Section 3, (3.1, 3.2, 3.3, 3.4), Conclusion	Entire document
Tao Nichol	Section 2 (2.3), Section 6, Section 10	Section 2 (2.3), Section 6, Section 10	Entire document
Carter Moore	Section 1.2, Section 8.1, Section 7	Section 8.1, Section 7	Entire document
Arta Namjoo	2.2, 4, 5, 8.2	Executive Summary, 2.2, 4, 5, 8.2	Entire document

Executive Summary

This report discusses the development of an automated plant growth system to be used in areas of high population density and remote regions. Due to the characteristics of these areas, developing any form of agriculture can be very difficult. The report discusses the steps taken in developing the project, all considerations made, and the final design.

The primary stakeholders considered for this project are the users of the system. They have been specifically designated as residents of remote communities and highly populous cities. Other stakeholders considered for the purpose of this project include the manufacturers of the system components, the agricultural industry, and potentially the construction industry. Constraints and objectives provided by the stakeholders are explained in further detail within the report. Additional constraints on the project include the lack of certification to work with certain voltages. This prevented the inclusion of a light and heater mechanism. These functions were replaced with red and blue LED (Light-emitting diode) lights in the final prototype. The short time frame in which the project was to be completed acted as another constraint.

This project entails the use of varying sensors that determine when the system requires certain actions. Relevant background research on all necessary components has been described and explained in the report. This includes information regarding a hygrometer to determine soil moisture levels, a thermometer to determine biome temperature, a photoresistor to mediate plant light consumption, and an ultrasonic sensor to determine water quantity in the water pump tank.

The report further describes the project's economics. Financial needs for the initial design and prototyping are described, with a detailed rundown of all financial considerations made and all costs endured. Discussion of the potential benefits and drawbacks of practical applications have been included as well. Environmental considerations made primarily revolve around the benefits presented by the project. The primary downside determined is that of the materials necessary for the system components and the manufacturing. The report details the use of EDII and other ethical practices as they relate to the project. EDII and ethics considerations made for the project focus on the use of the system in remote areas, especially those in poor weather conditions with benefits being pointed to northern indigenous communities.

The final section of the report's main body describes the final design. Details on the process taken to arrive at the final point, including the use of planning tools have been included to aid in understanding the steps taken toward the final answer. Issues and resolutions encountered are explained; the foremost issue found with the project was the high voltage necessary to run certain components. As a result, many portions of the final presented prototype had to be altered. The final design is explained according to two categories: the circuitry and wiring of the system, and the programming of the system.

Contents

Table of Figures

1 Problem Definition

The following sections, Section 1.1 and Section 1.2, will elaborate on the problem definition and give an overview of background information regarding the autonomous plant system project and the problem that must be addressed.

1.1 Problem Statement

Gardening is one of the many hobbies people all around the world do as a form of relaxation and connecting with nature [1]. It is an activity that is good for both the mind as well as the body and can be enjoyed by people of all ages [1]. However, maintaining an indoor garden is difficult with the daily tasks and responsibilities people of urban societies, indigenous communities, and densely populated communities face on a day-to-day basis. With responsibilities to jobs, family, and oneself, there is not always time in our day to properly take care of our plants. Additionally, during seasons like winter where there could be days of extreme cold conditions, gardeners must put their hobby on hold. To make this process more efficient and allow gardeners to continue with their hobbies, it has been decided that an autonomous plant system will be designed that can maintain plants indoor when a person is not there to attend to it and allow the user to receive constant updates on the status of the system. The design solution will contain multiple Arduino compatible sensors that will work simultaneously with each other to create the ideal conditions for a plant to grow without the constant necessary supervision of the user. A prototype will be made for the design solution and the components must work simultaneously, and according to their function in the Arduino code.

1.2 Stakeholders

The stakeholders in this design process include the system users, the manufacturers of the electrical components, as well as the agricultural and construction industries. The stakeholders' needs are crucial to determining the scope and assessing the success of a solution. If the stakeholders' needs are delivered as expected, or the execution of the final design goes beyond expectations, it will indicate the success of the proposed solution. Otherwise, the proposed solution will be a failure. Table 1 briefly describes the needs ands and expectations from each stakeholder.

Table 1: Stakeholders for the autonomous plant system and their expectations

Stakeholders	What are they expecting?
System Users	An operable plant system that succeeds at growing plants indoor without the physical supervision of the user
Manufacturers	Arduino component manufacturers will want their product to operate exceptionally and meet all the conditions required (e.g., sensor functionality, data processing, communication through Wi-Fi)
Agricultural industry	The success of the project to see an increase in demand of products like seeds because of the shift towards home farming (more populated areas)
Construction industry	The success of the project to see an increase in sales due to the implementation of the project in newly designed buildings

The primary stakeholder of this project will be the system users. The system users will be the future gardeners that will be afforded the opportunity to continue their hobbies and passions for indoor gardening while enduring busy schedules that keep them physically away from their plants. The expected users would be residents of urban, indigenous, and other densely populated areas where the opportunities to farm are far more scarce than rural areas. People from these urban societies lack the space to grow plants and want to garden indoor but lack the time throughout the day to tend and take care of their plants. To satisfy their needs, the proposed solution must allow plants to grow indoors without the supervision of the user. Additionally, the plant system must communicate to the user the status of their plant(s) to keep the user up to date with things going on. The device must also be cost-effective, ensuring that it is affordable to the public. An expensive solution will deter many potential buyers who may other otherwise be unable or unwilling to invest in it. Furthermore, the solution must be user-friendly. This includes the minimization of language and technology barriers that may make it hard for some people to use.

Manufacturers of the system and their electrical products are also stakeholders of the project. The different manufacturers will want their respective products to operate according to their functionality and meet all the conditions required to indicate success of the project. If the manufacturers deliver fully functioning products that can be implemented into the Arduino Uno seamlessly and satisfy the user, they may see an increase in their revenue as well as customer satisfaction. However, if the manufacturer delivers a product that is incompatible with the project due to faults like broken or missing pieces, the project will fail due to missing components that otherwise would make it a well-rounded autonomous plant system. This would mean that customer satisfaction would decrease, as well as the trust of investors and customers for future interactions.

It is important to consider the two significant sectors that will be heavily impacted by the plant system solution, one of them being the agricultural industry. Since the solution allows for many new people in urban areas to grow their plants, including vegetables, and fruits, the agricultural industry could experience a new shift towards home farming instead of regular farming. This could be beneficial to certain parts of the industry, for example, companies that well seeds. According to the United Nations (UN), it is estimated that about 55% of people live in urban areas [2]. In accordance with the UN's research, since most people live in urban areas, if the plant system was to be implemented on a larger scale like a city, companies that sell seeds or other essentials for indoor farming would see a significant increase in sales because the majority population would have better opportunities than before. However, the farming industry could experience a decrease in revenue as people will no longer feel the need to do traditional farming outside when they can do it at home autonomously under ideal conditions for plant growth.

Lastly, the other section that will be heavily impacted if the solution is a success is the construction industry. If newly developed apartment buildings and houses account for the solution and dedicate spaces towards it, the layout of buildings could completely change to consider indoor gardening. Additionally, they could help sales in the industry due to the current lack of garden space in urban housing.

2 Electrical Components

The following sections will go in depth on the electrical components that the team needs in order to design a fully operational autonomous plant system that can meet the needs of the most, if not all, the stakeholders. The importance of each electrical component, their functionality, and how the team intends to use them will be elaborated further.

2.1 Ultrasonic Sensor

An ultrasonic sensor is an electronic device that measures the distance of a target object by emitting ultrasonic sound waves and converts the reflected sound into an electrical signal for the Arduino to analyze and present data [3]. Ultrasonic sensors have two main components: the transmitter (which emits the sound using piezoelectric crystals) and the receiver (which encounters the sound after it has travelled to and from the target) [3]. Figure 1 shows a diagram of how an ultrasonic sensor operates.

The sensor's transmitter emits sound waves, and they bounce back to the sensor's detectors when they encounter a target. Once the detector receives the reflected sound waves, it can detect the distance between the target and sensor. The ultrasonic sensor used in the project is the HC-SR04. It is said to have an accuracy of 0.3 cm (0.1 inches) [4]. To make the solution as autonomous as possible, the ultrasonic sensor is used to detect the water levels of the water storage for the plant system to determine when the user should refill the tank. The detection of water levels in a closed container is very important for autonomous plant systems. Additionally, it was important for the sensor to detect the water levels as accurately as possible so that the user has reliable information and knows whether to refill the tank or not.

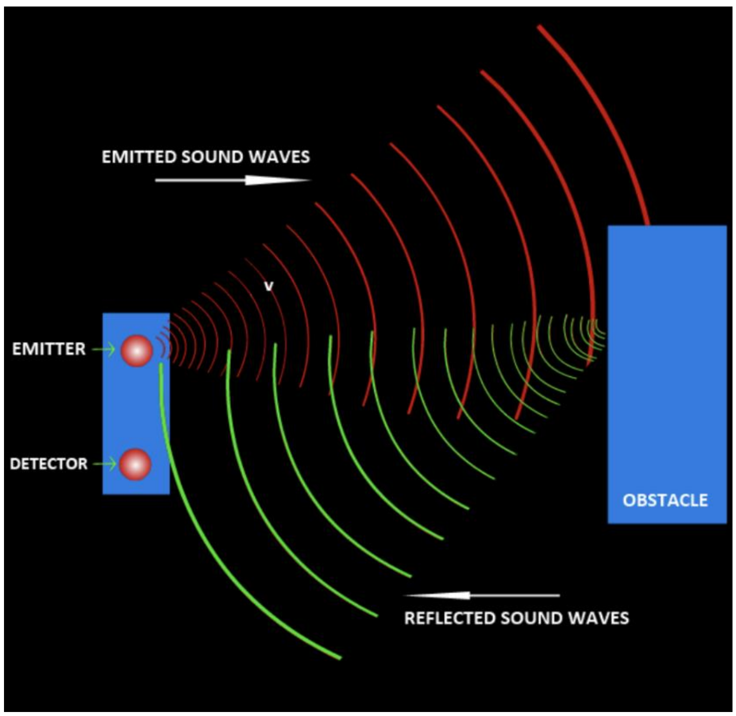

Figure 1: Ultrasonic sensor functionality diagram

2.2 Hygrometer

A hygrometer is an instrument used to measure humidity [5]. The sensor used is the Capacitive Soil Moisture Sensor v1.2 and it measures soil moisture levels by using capacitive sensing [6]. A capacitive sensor was chosen as it can sense a diverse number of materials like metal, liquid, plastic, and most importantly, soil [7]. As an essential sensor for an ideal autonomous plant system, the hygrometer controls the humidity and moisture levels within the soil. The hygrometer acts as the primary influencer in determining when the hydration system releases water into the plant and the soil habitats. When it is placed in the dry soil, the output is set as the baseline for determining when to water the soil. This allows for varying levels of soil moisture to be considered. When the sensor receives the input of how dry the soil is, it allows for a certain level to be reached before turning on the water pump using the Arduino code. As a result, different plant species could be grown based on moisture necessities.

It is important to consider the humidity of an area as it significantly influences the plant life that grows around it [8]. Humidity impacts plant growth by increasing or decreasing the water absorption in plants [8]. Increased water in plants results in more nutrients and minerals being absorbed from the soil by the plant through osmosis [9]. Thus, higher humidity increases the rate and size to which plants can grow [9]. Due to the necessity of certain plants and crops, as they relate to humidity, many species of plant are limited in habitats. For the design solution, wherein the goal is to create a successful method to grow a variety of plants, the influence of the humidity and soil moisture content on the plant must be governable. This is to provide the necessary criteria for the plants' growth and living conditions to be satisfied.

2.3 Thermometer

A thermometer is a device which measures temperature. The specific type of thermometer being an analog temperature sensor. The temperature is measured by having a voltage over the diode increase at a known rate [10]. By amplifying the change in voltage, it can create an analog signal mirroring the temperature [10]. This sensor will control the temperature of the plants environment by activating a heater when it is best for the plant.

The temperature of the environment heavily affects how well a plant can grow. Plants need on average a 10 – 15 degree Fahrenheit difference between nighttime and daytime temperatures [11]. Germination temperature also usually varies from a plants usual growing temperature. This usually has to do with the season in which the plants are regularly grown. The quality of a crop can be affected by the temperature: for example, high temperatures cause lettuce to go bitter [11]. Plants need the change in temperature so they can go through photosynthesis and respiration, both of which happen at two different extreme temperature conditions. To get to these, a way to control temperature is necessary.

The use of the sensor positively benefits the solution as it regulates all problems related to temperature. For whichever plant is being grown, there will be a temperature for every part of the day. If the temperature is not met, the system activates the heater and corrects the temperature to what is desired. The model of the sensor bought is the LMT86LPGM Temperature Sensor Analog by Texas

Instruments [12]. This was chosen because it is compatible with the Arduino Uno and has a trusted manufacturer [12].

2.4 Photo Resistor

A photo resistor is a light sensitive device that is most commonly used to detect the absence or presence of light [13]. The sensor used in the final design is the KY-018 Photoresistor that measures light intensity. The resistance decreases in the presence of light and increases in the absence of it [14]. The output is analog and determines the intensity of light [14]. The photo resistor used in the design solution controls the amount of light the plant is exposed to at any given time.

Photosynthesis is vital for plant growth. Photosynthesis is the process in which light is converted into chemical energy. This converted chemical energy is then used to fuel the cellular activities of the organism [15]. When working with light, both the intensity of the light being omitted, and the duration of exposure are important to consider. Light intensity is measured in lux, where lux is one lumen per square meter. Varying the lux that a given plant is exposed to will influence its rate of photosynthesis. As the lux increases, the rate of photosynthesis does as well [16]. Additionally, the average plant only needs 12-16 hours of light per day to facilitate healthy growth [16]. Plants need both periods of light and darkness to achieve the best results [16]. Both the light intensity and the duration of light emission must be monitored and variable to successfully grow a variety of plants.

The small size of the KY-018 allows for it to be easily incorporated into the design. The photo resistor works by sensing the amount of light exposure there was within the system. Every plant needs a varying amount of light for healthy growth. Each plant option has varying code to accommodate its needs. Depending on which plant the user has selected to grow, the photo resistor will track the duration of light exposure and will turn off the grow light once the plant had received enough light for optimal growth. Additionally, the photo resistor would turn the grow light back on after the plant had gone without light for the optimal amount of time.

3 Objectives, Design Criteria and Specifications

The following sections outline the project plan that has been followed throughout the duration of the P2 project. Additionally, the objectives of the P2 project are clearly outlined as well as the constraints and limitations that were faced.

3.1 Work Breakdown Structure & Deliverables

Table 4 found in the Appendix outlines all the tasks that are due throughout the semester. The chart clearly outlines when each deliverable is due, as well as a recommended start date for each task to ensure all projects are completed on time. Additionally, the Gantt chart has a column that tracks the progress of each task. This ensures that all team members are aware of what remains to be completed for any given deliverable. Finally, the chart includes a column specifying whether the task is a group or individual activity. The Gantt chart provides the team with a clear schedule to follow to ensure success throughout the semester.

3.2 Gantt Chart

Figure 6 found in the Appendix outlines all the tasks that are due throughout the semester. The chart clearly outlines when each deliverable is due, as well as a recommended start date for each task to ensure all projects are completed on time. Additionally, the Gantt chart has a column that tracks the progress of each task. This ensures that all team members are aware of what remains to be completed for any given deliverable. Finally, the chart includes a column specifying weather the task is a group or individual activity. The Gantt chart provides the team with a clear schedule to follow to ensure success throughout the semester.

3.3 Objectives

The main objective of the P2 project was to design and create a fully autonomous plant growing system. In order for this to be successful, all aspects of plant growth had to be considered. The factors influencing healthy and optimal plant growth consist of soil moisture levels, proper temperature, and light intensity [17]. All of these factors must be constantly monitored and variable in order for any given plant to grow in its optimal conditions. By considering the required conditions to effectively grow various plants, it was determined that four different sensors would be needed to create a system that is truly autonomous. The four sensors include the KY-018 Photoresistor, the LMT86LPGM thermometer, the HC-SR04 Ultrasonic Sensor, and the Capacitive Soil Moisture Sensor v1.2. By utilizing a detailed project plan, the team was able to build a functioning prototype in the given time frame. Designing the prototype was split up into four different phases, where each phase consisted of working on a different sensor. Each sensor was worked on individually to ensure that it was able to execute its task successfully. Once all the sensors were fully functional, having them all working simultaneously with one another became the next objective. Details on how this was achieved can be found in the final design section of the report. With limited time, designating days to work on the prototype as well as the reports were crucial to completing all aspects of P2 on time. Having a comprehensive project plan allowed for time to be designated efficiently. The Gantt Chart outlined specific meeting days as well as what tasks needed to be completed, allowing for all team members to have clear objectives in mind.

3.4 Constraints & Limitations

The constraints on the project became apparent while designing the prototype. The main constraint that was faced revolved around having insufficient power to supply all the electrical components of the autonomous plant system. Two electrical components intended to be implemented into the prototype included a heater and light source. The heater required 12 volts and the light source required 120 volts which is significantly above the voltage that a laptop can supply to the Arduino. To overcome that obstacle, a substitution was made for both the heater and light bulb with red and blue LEDs, respectively. In result, once any given plant needed or had enough light exposure, the blue LED would switch on or off indicating that the requirements of the plant were being fulfilled. Similarly for the heating system, the temperature sensor would switch the red LED on or off indicating whether the heater is on or off. In addition to having power supply issues, using a supplemental power source was not an option because certificates and qualifications would be required [18]

Another constraint on the project was time. Ideally, to demonstrate the capability of the prototype, a plant would have been grown fully autonomously. Unfortunately, there was no way to grow a plant in the given time. The average plant takes roughly 2 weeks to sprout [19]. Seeing as the prototype would have to be fully complete before plant growth could begin, it became apparent early on that growing a plant would not be feasible. As a result, ensuring that the prototype would be capable of functioning with a simulated plant and its ideal growth conditions was a top priority. Despite not having a physical plant, knowing the conditions the plant needed for optimal growth was enough to program the prototype and demonstrate its efficacy.

4 Project Economics

The overall financial costs of this project amounted to $100.80. The primary costs were a result of the materials for the project's success. A summary of all parts has been given in the Bill of Materials in Table 2 below.

Table 2: Bill of Materials of all electrical components and their estimated total cost

#	Description/name of part	Total Quantity	Est total cost
1	KY-018 Photo Resistor	1	5.49
2	Capacitive Soil Moisture Sensor v1.2 (Hygrometer)	2	9.88
3	Thermometer	3	5.56
4	LED Grow Lights	1	5.29
5	HC-SR04 Ultrasonic Sensor	1	11.99
6	Heater	1	35.54
7	Water Pump	3	15.65
8	Wall socket	1	2.64
9	Bulb socket	1	5.79
10	Tube	1	2.97

When considering the finances of the project, most of the cost influence occurred during the initial research of all components. When considering each sensor, due to the significant variety, and the relatively slight differences in varying sensors, a look at the cost and pre-determined reliability of components were what determined which parts were selected for the project. During this stage, the necessary certificates to work with the light and heater's voltages were unaccounted for. As a result, they could not be used in the final prototype, resulting in excess wasted costs of $49.62. For future goals in the project, a workaround to either include the lights and heaters or to completely remove them from the scope should be made. Potential costs because of this would entail the costs to become certified to work with higher voltages and the inclusion of a battery into the system to power these components. This would cost approximately $760 for each individual working with high voltages and wiring [20].

This project was intended for use in highly populous and remote areas. As a result, the system was designed with the intent to fit in small living areas and unused spaces. Due to these specific design criteria, the components for the system had to be small enough so that the entire system could fit in a small growth tube. As a result, fewer sensors of the same kind, such as the soil moisture sensor and light sensor, were necessary. This was accounted for and determined to be cheaper as opposed to using a larger system which would potentially require multiple sensors and devices working throughout the day.

When considering future costs for a project of this scale, a mind map was used. The mind map used briefly describes all economic factors that were considered because of this project. An image of the mind map has been included in Figure 2 below.

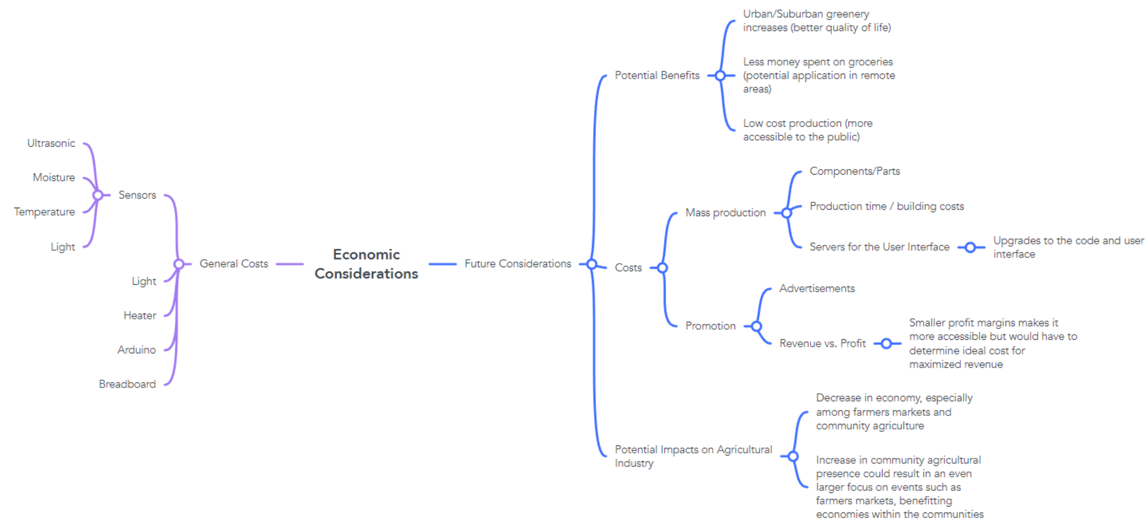

Figure 2: Mind map of the economic considerations made during the creation of the prototype

In terms of future costs for this project, it can be summarized into the cost for components, the cost to produce, and the cost to sell. On average the cost for mass production of the finalized design would approximately $117.68 per system [21]. The cost to produce was reduced as much as possible in consideration of future project economics and ethics as it would make it more readily accessible to the public for which it was intended. Additionally, when considering mass production, the decrease in the cost per unit if the components are bought in bulk must also be contemplated.

The system would prove most economically beneficial to residents in areas of high-density populations. This would include cities, urban areas, and possibly suburbs. Taking this into account, the average allowance for groceries in a month for an individual in Canada is approximately $217.07 [22]. The use of the systems provided in this project could reduce these costs by 10% to 22% on average [23]. Additional factors considered were the space necessary for the system to operate which would prove more difficult as the system increased in size. This proved ineffective for the purpose of the project, which is why the system was designed to work in any environment with minimal space required. As a result, the cost-effectiveness of the project dropped, as more systems would be needed to grow an equivalent number of crops to that of a larger, cheaper system.

Other considerations made were for the system's use in remote areas. Remote areas can be defined as locations that are hard to access. As a result, many remote areas with a population have significantly larger grocery and general shopping costs. Examples of this can be seen in Nunavut, wherein a 2.2-pound bag of grapes, which in Ontario grocery stores would cost approximately $6, costs almost $30 in Nunavut [24]. This results from the difficulty with which products are delivered to these areas. This project would prove useful in these areas as it would provide an outlet for the highly expensive costs of food, allowing the economy to flourish in other areas.

5 Environmental Considerations & Impact

As a plant growth system designed for remote and urban locations, the project assists the environment in multiple manners. The increase in indoor gardens in areas of high population, which tend to have a low level of greenery, will increase the overall quality of life. Indoor gardens require fewer resources as opposed to rural farming [25]. On average, urban and vertical farms consume 80 % to 99% less water than traditional farming methods [25]. Additionally, in areas of high emissions, such as cities, the common presence of indoor gardens can partially assist in clearing the air [26]. This could revitalize the presence of wildlife in cities.

In remote areas, the presence of indoor gardens, as presented by the report, would act to reduce the number of agricultural deliveries necessary to these areas. The method by which products are currently delivered to these areas is inefficient with copious emissions. By reducing the number of deliveries made, the carbon emissions made from these deliveries would drop as well.

Environmental drawbacks endured by the project include mining and manufacturing the system parts. The system requires many electrically intensive and specific qualities that are obtained from only certain materials. With a rapidly decreasing quantity of resources available in the world, this can prove negatively impactful on both the environment and society. Other environmental considerations to be made for the project include the potentially high level of energy consumption. Currently a miniature system such as the one presented for the project required approximately 200 volts to become fully functional. However, even with these environmental drawbacks, the use of indoor gardens to facilitate self-sufficient crop growth continues to prove more environmentally beneficial than the general agricultural industry [25].

6 EDII (Equity Diversity Inclusion Indigenization)

For the duration of the P2 project EDII was heavily considered. During the research phase all information gathered was double checked on other websites to get the clearest results and inclusion of multiple perspectives. For example, certain communities grow plants under different growing conditions to outcome different tastes, so change of conditions can be done over the website to allow for anyone to grow their desired plant to their liking [27]. The variety of plants that the system will have pre coded in will be very large so using the system is just as efficient for everyone and not a specific community. The website is also very user friendly as language and font can be easily changed, allowing for people who speak any language and people who have poor eyesight or are just irritated by smaller words to use the system with no problems. The website also includes pictures next to each plant to allow for younger kids to use the system and grow their own plants. EDII was also considered within the group itself, as it was made sure all opinions were valued equally and work loads were evenly spread out regardless of whom wanted to do more or less work. The system was made so it will be functional everywhere, regardless of the conditions the system is around. This was done so people in remote communities can effectively grow crops. Prices for groceries in places which are remote are often more expensive making the system even more viable in such areas.

Any groups of people which are found out to have a hard time navigating or using the system can be dealt with and the system can be updated to better provided for such groups of people.

7 Final Design

To create an environment that promotes plant growth, several sensors and devices are needed to work simultaneously. This includes a hygrometer, photoresistor, thermometer, and ultra-sonic sensor. These sensors are discussed thoroughly in Section 3: Electrical Components. By using these sensors, the environmental conditions can be monitored and manipulated so that a specific species of plants can grow. To manipulate the conditions, a heater and LED grow light were selected. These devices would theoretically be able to provide the plant with enough heat and light to grow efficiently. However, when designing the prototype, it became clear that these devices require relatively high voltage and current amounts to operate. These power requirements created some problems that were quickly resolved. Furthermore, a digital program created using the Arduino IDE (Integrated Development Environment) is used to control the output devices using data retrieved from the sensors. The program is also responsible for wirelessly communicating with the user in an easily operatable manner using the Wi-Fi capabilities of the Arduino Wi-Fi Rev 2 microcontroller. Figure 3 depicts a block diagram describing the function of the final design.

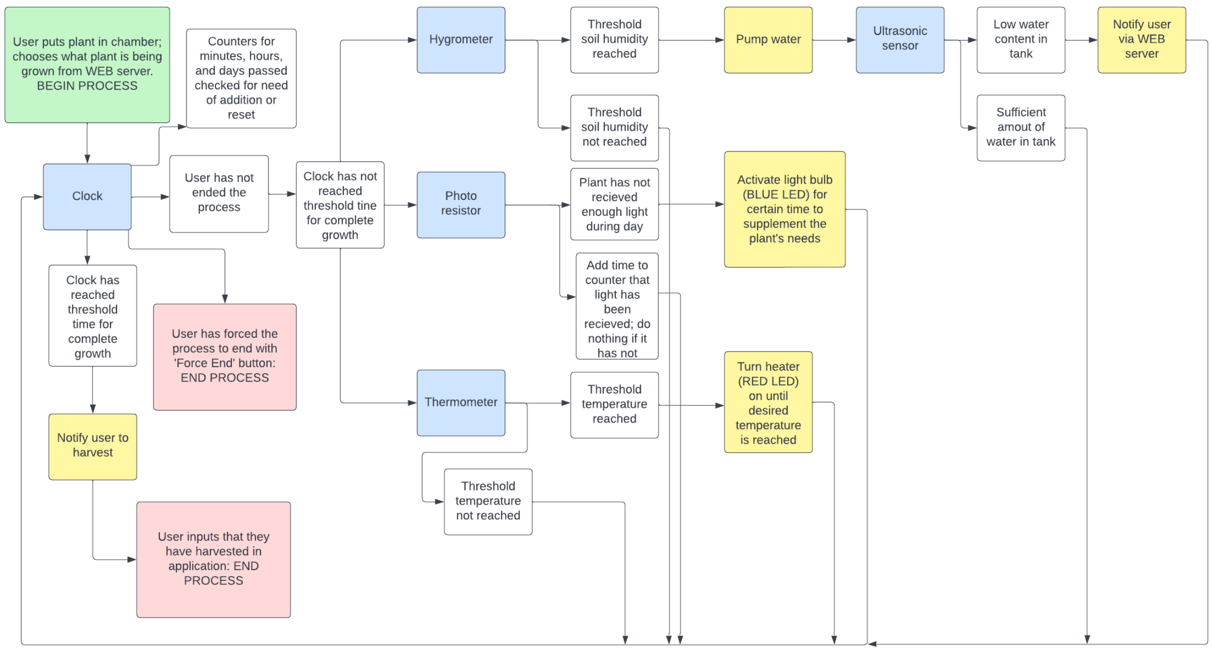

Figure 3: Block diagram showing the functionality of the final design

7.1 Issues Encountered with High Power Components

Working with high current and voltage components can introduce health and safety risks when designing a circuit. The heating unit chosen for the P2 project requires 12V, 10A of power and the light requires about 120V, 0.5A of power. If mistakes are made when working with devices that draw this much power, the health and safety of the operator is at risk. It is for this reason that the National Fire Protection Association (NFPA) has several codes and standards that apply to working with relatively high voltages. For example, "NFPA 70E ® 130.4(F)(1) states the qualified worker's hands must be insulated or the energized parts must be guarded before they can enter the restricted approach boundary [18]." Furthermore, NFPA 70E ® table 130.4 (D)(a) lists the "restricted approach distance" as "avoid contact." This means that personal protective equipment such as rubber gloves may be needed when working with devices such as the heater and grow light that were selected. Thus, without the proper knowledge and training, these devices could not be safely operated. This required a decision to be made between three options: completely discard the implementation of the heating and lighting system, use another means of representing the light and heater instead of using them, or get training to work with the desired devices. A "Pros and Cons" table was created to aid in comparing these options.

Table 3: Pros and Cons table to aid the decision for the issues with the required devices

Pros	Cons
Option 1: Discard the implementation of the devices	
• Creating the prototype would take less time as the complexity of implementing these devices is avoided • The health and safety of the team is preserved as the risks of working with high-power are completely avoided	• The system would not be able to grow plants as intended as heat and light are vital to a plants growth. • The complexity of the project would be greatly reduced, meaning that it would be basic and unsatisfactory.
Option 2: Represent the loads using a method that requires less power	
Pros	Cons
• The health and safety of the team is preserved as the risks of working with high-power are completely avoided • The system would not be able to grow a plant as intended, however it can be represented. This means that the system would work as intended if the new method was replaced with the intended devices. Thus, creating a system that would work in theory.	• The system would not be able to grow plants, only theoretically.
Option 3: Get the training that is required to work with the power that is needed	
Pros	Cons
• The system would be able to grow a plant as intended. This means that with an adequate design of the circuit and software, the design would be able to not just represent, but grow plants efficiently.	• The health and safety of the team is still risked, even with the required training. • The training would take time, postponing the rest of the project which could suffer due to less time spent on the design and implementation phases.

By creating this table, the options were easily assessed, and it became apparent that the best solution would be option two. Option two allows for the system to be represented in a way that still displays its full functionality. It also mitigates the risk of working at high voltage and would not require training to implement. Option one proved to fall outside of the scope of the project being too simple, and option three could put the operator at unnecessary risk. To represent the absent heater and light devices, LEDs that can be activated with low power are used in place of these devices. The other device, the water pump, operates at five volts, meaning that by using a relay module (a circuit that can control the flow of current to a device using a low voltage signal), it is able to be operated safely without any hazards.

7.2 Circuitry and Physical Connections

The circuit uses the Arduino Uno Wi-Fi REV 2 microcontroller as the energy and control source for the water pump, LEDs, and sensors. The microcontroller can receive and transmit analog as well as digital signals from the sensors, allowing for decisions about the pump and LEDs to be made using the readings. These decisions are made by programming the microcontroller in an Integrated Development Environment (IDE). Figure 4 depicts the circuit that was designed for the P2 project. The components are all clearly labeled, where the pins labeled "D#" on the Arduino are digital pins and pins labeled "A#" are analog pins. Figure 5 depicts the real implementation of the schematic in Figure 4.

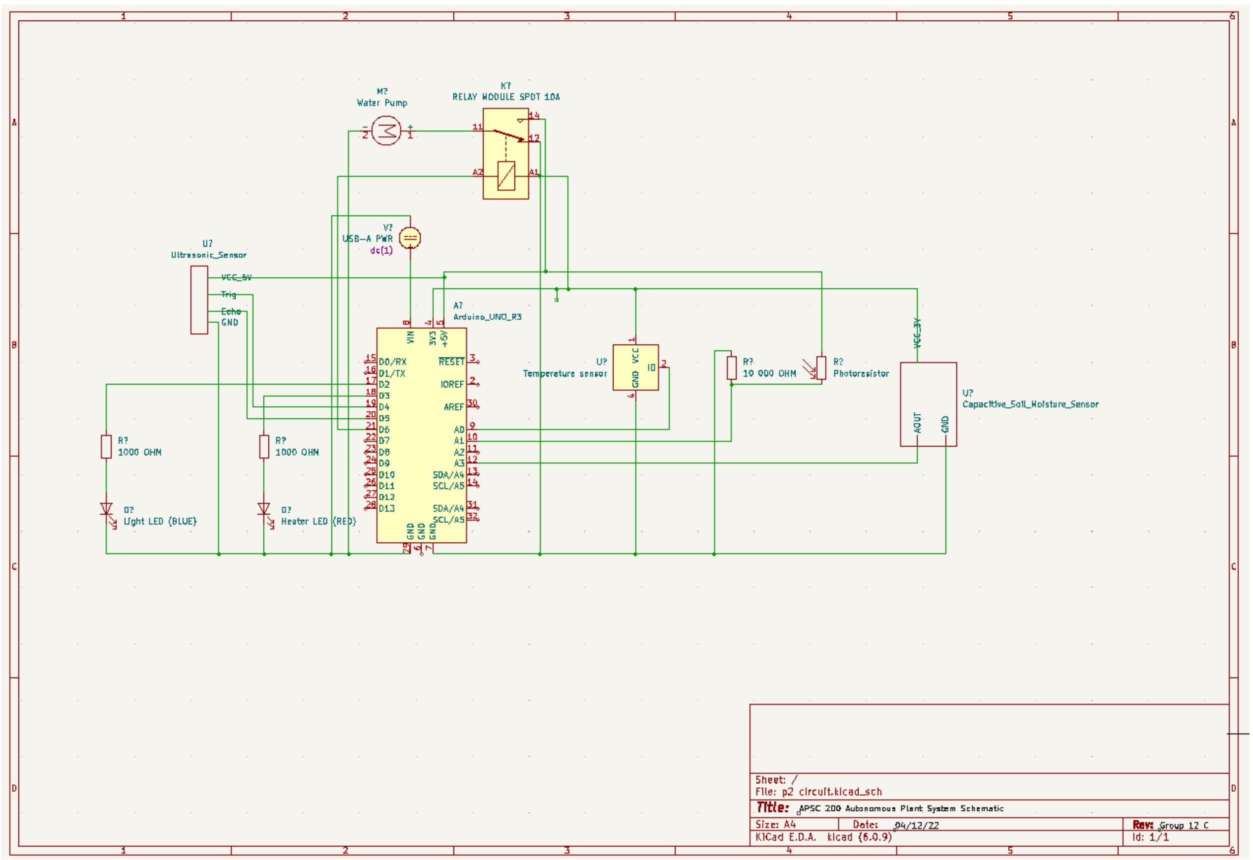

Figure 4: Schematic of the circuit used in the P2 project created using KiCAD software

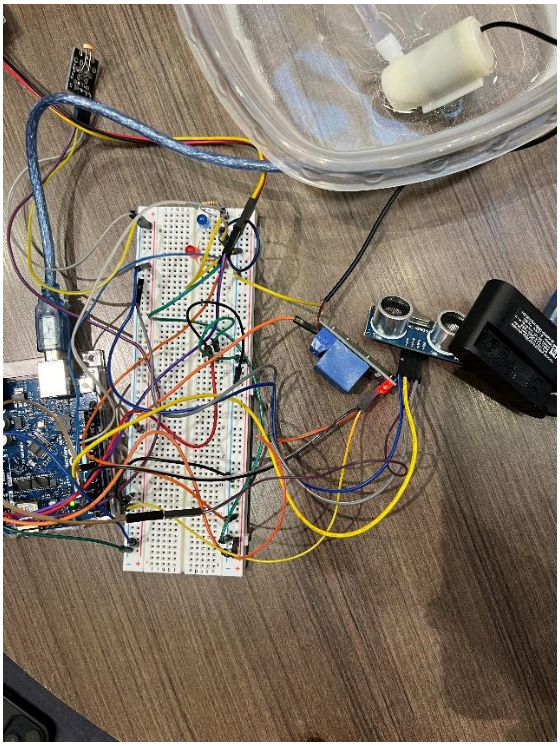

Figure 5: Implementation of the schematic in Figure 4

7.3 Programming the System

For the circuit to operate as intended, the microcontroller is programmed to read values from the various sensors and store these values digitally. After these values are read, they can be compared to the soil moisture percent, temperature, and amount of light required for certain plants to grow. The outcomes of these comparisons are used to control the water pump and LEDs that represent the heater and grow light. Also, by measuring the distance of an empty water tank using the ultrasonic sensor, it can be compared to measurements when there is water in the tank to determine if the tank must be refilled.

The getTemp(), getMoisture(), getLight(), and getWaterDistance() from the program depicted in the Appendix (line 440 to line 502) are functions that read the input values of each sensor. These values are stored digitally using variables. Then, by using the decisions() function, (line 411 to line 437) decisions can be made using the researched values for the plant's requirements found in the array called plantsThreshold (line 31). These functions are used in the loop function. The loop function keeps the Arduino running, reiterating the code when the end of the loop is reached until the end function is called. A clock is also required to keep track of the time passed since the plant has started growing. It is necessary to determine if the plant has finished growing so that it can be harvested. Also, the clock is used to calculate how long the growing plant has been in sunlight, and if it requires more time in light than it received. This is calculated by determining if the plant has received light every hour for a day, then once the day has passed, the light will turn on if the respective value in the plantsThreshold array has not been reached. The clock works by using the delay function. Since the Arduino works by using a loop function, it is possible to calculate the delay of every function such that the total delay is a minute. Then, a series of counters can be used to keep track of how many minutes, or iterations of the loop, have passed.

The Arduino Wi-Fi Rev 2 microcontroller was chosen because of its ability to connect and communicate to other devices on a wireless network. This means that values can be read from and printed to a local server for the user. By using this, the user can wirelessly decide which plant they would like to grow and review information from the sensors as well as advice on when to refill the tank or harvest the plant. Figure 8 and Figure 9 in the Appendix show what this server looks like when accessed by the user. The printWEB() function (line 264 to line 404) is used to print to the server and receive data from hyperlinks. Several other functions, denoted by the “Wi-Fi” in their name, contribute to this connection.

Although the final design does not have the ability to maintain an environment for a plant to grow, the system represents a theoretical solution to an autonomous plant growth. The substitution of LEDs for the high-powered electronics can not achieve the same function, however, if they were replaced by these devices the system would be able to operate as intended. The solution is still within the scope and satisfies the needs of the stakeholders as the design is only a prototype. The user can receive information and make decisions based on the growth of a plant, and the part manufacturers would be satisfied as the components have all been tested to work sufficiently. However, the industrial stakeholders’ success would be assessed over a long period of time after the autonomous plant system is released to the public. Based on the requirements for success defined in Section 1: Problem Definition, this design is satisfactory as the system has the ability able to grow more than one plant, even though it cannot physically do so because of the constraints associated with safety and time.

8 Discussion on Ethics for AI (Artificial Intelligence) Software

8.1 Privacy

Ensuring that the information of a user is protected it is important to having an ethically sound solution. For the P2 project, the information that the user sends and receives is passed through a wireless network. This network is password protected using WPA2 (Wi-Fi Protected Access 2) security. This level of security implements all aspects of the 802.11i security standard that is agreed on by the Wi-Fi Alliance, an organization that regulates Wi-Fi standards [28]. Although this information is protected by the security of the local network that it is connected to, this means that anyone on the network could view the user's information. Thus, if the design were to release to the public, it would be ethical to provide a second password that only the user knows. By doing this, the user could be reassured that their information is protected regardless of its importance. Furthermore, if the project were to begin collecting data from users for profit or for use elsewhere, the PIPEDA (Personal Information Protection and Electronic Documents Act) for Canadian regulation and GDPR (General Data Protection Regulation) for European Union regulation would need to be considered. This would mean developing a term of service document for users to comply with and notifying the users if an information breach were to occur. Although these are laws that must be followed, they are also ethically important to consider as they can impact the user's experience and personal life. Privacy of information is an important consideration when working with a wireless network to transmit and receive information.

8.2 AI Bias

Sources of bias in the program would stem prevalently from the output determined by the code based on the sensor inputs. For the scope of the project, when programming the sensors, the values determined for certain sensors based on what was read by the Arduino were designated based on testing through trial and error to determine as accurate a conversion method as possible. The use of trial and error did not account for states of extremes in the sensor's perception. As such, it cannot be guaranteed that the system would act in certain conditions or certain areas. Examples of this could be considered in areas of extreme cold, where the temperature sensor may render incorrect values. This could result in the heater being unable to heat up the system to an adequate temperature, or not turning on whatsoever. Most of the biases that could present themselves in the system stem from measurement biases in which enough varying samples and tests could not be completed within the scope.

8.3 Decision Harm and Radicalization

When considering decision harm and radicalization for our solution, the following four ethical principles of respect for human autonomy, prevention of harm, fairness, and explicability were emphasized to ensure that the AI systems are developed, deployed, and utilized in a lawful, ethical, and trustworthy manner. To make sure the principle of respect for human autonomy was adhered to, the solution was purposely designed to leave opportunities for decision-making like which plant type users would want to grow and periodic status updates which they can supervise whenever they felt like rather than giving the user a lack of freedom to make choices of their desires. To make sure the principle of prevention of harm was respected, the solution was made operable under low voltage to mitigate any harm to the user, plants, and natural environment in the vicinity. For the principle of fairness, ensuring the prevention of unfair prejudice, discrimination, and stigmas were key to reduce AI risk mitigation. Personal inquiries like gender, age, and race were not involved in the solution's functions as data to collect from the user to commit to equal and fair distribution of benefits from the plant system. Lastly, when considering the principle of explicability, a decision was made to establish and maintain as much trust from the user in AI systems. To make sure the plant system is openly communicating with the user, information is always updating every minute to make sure the user constantly knows what is going on. Additionally, decisions making by the plant system like turning on a light source, turning on the heater, and draining water from the tank can all be explained to the user through prompts that display on the user's screen.

9 Conclusion

For the project to be deemed successful, the proposed solution must satisfy all or most of the needs of the stakeholders. Despite having not fully tested the prototypes capabilities by growing a plant, the autonomous plant system that has been designed theoretically would satisfy the user based on the test that have been concluded. By using the conditions that a certain plant would require for optimal growth, the autonomous plant system was coded and tested. Its efficacy was demonstrated when all the sensors performed their functions simultaneously and correctly based on the simulated plants conditions. In addition to satisfying the user, the autonomous plant system would also satisfy the Arduino component manufacturers seeing as all the components within the system are operating and functioning as they should be. Finally, since the autonomous plant system is theoretically fully functional, it can be assumed that the other stakeholders, including the gardening and construction industry would also be satisfied. Since the prototype was a success, the gardening industry can expect an increase in demand of products such as seeds due to the shift towards home farming. Additionally, the construction industry can expect an increase in sales due to the implementation of the project in newly designed buildings. Seeing as all the stakeholders involved in the project would be satisfied, it can be deemed that, overall, this project was a success.

10 Recommendations

The system designed is very functional but lacks scaling, a visually pleasing website, and testing. In the future a scaling option should be included as people will want different sizes of growing systems. If a way to easily scale the system is designed, then larger systems should be more affordable as all the parts can be used to their maximum eficiency. Additionally, only one Arduino Uno will have to be used. This makes the system more desirable as people looking for large indoor gardens can achieve what they desire compared to someone who just wants to grow individual plants. Next updating the website to look more pleasing as the current florescent green can be hard to look at for longer periods of time. Also changing fonts, adding pictures, and buttons to give the website a more professional and eye-catching look. This will make the website easier and more rewarding to use. Testing will be very important in the future as the growing system is yet to be implemented with an actual plant. Using a control plant monitored by the group the system can be compared to this to figure out any problems with the system or to just make it more efficient. Also, if any groups of people find the growing systems does not work for them, a solution will be created for said problem and implemented.

References

[1] J. Haskins, "Gardening: A fun hobby that’s good for your health," The Nation's Health, July 2017. [Online]. Available: https://www.thenationshealth.org/content/47/5/17#:~:text=Gardening%20may%20be%20a%20fun,and%20herbs%20that%20you%20grow.. [Accessed 6 November 2022].

[2] H. Ritchie and M. Roser, "Urbanization," Our World in Data, November 2019. [Online]. Available: https://ourworldindata.org/urbanization#:~:text=UN%20estimates%3A%2055%25%20of%20people,extending%20from%201960%20to%202020.. [Accessed 30 November 2022].

[3] D. Jost, "What is an Ultrasonic Sensor?," Fierce Electronics, 7 October 2019. [Online]. Available: https://www.fierceelectronics.com/sensors/what-ultrasonic-sensor. [Accessed 5 November 2022].

[4] "Complete Guide for Ultrasonic Sensor HC-SR04 with Arduino," Random Nerd Tutorials, [Online]. Available: https://randomnerdtutorials.com/complete-guide-for-ultrasonic-sensor-hc-sr04/#:~:text=The%20HC%2DSR04%20ultrasonic%20sensor,good%20for%20most%20hobbyist%20projects.. [Accessed 30 December 2022].

[5] T. E. o. E. Britannica, "Hygrometer," Britannica, [Online]. Available: https://www.britannica.com/science/hygrometer. [Accessed 5 November 2022].

[6] A. Newton, "Interface Capacitive Soil Moisture Sensor v1.2 with Arduino," How2Electronics, 19 November 2022. [Online]. Available: https://how2electronics.com/interface-capacitive-soil-moisture-sensor-arduino/. [Accessed 4 December 2022].

[7] "Capacitive Sensors," BYJU's, [Online]. Available: https://byjus.com/physics/capacitive-sensors/. [Accessed 4 December 2022].

[8] F. J. Veihmeyer and A. H. Hendrickson, "Soil Moisture in Relation to Plant Growth," *Soil Moisture in Relation to Plant Growth,* vol. 1, no. 1, pp. 285-304, 1950.

[9] M. A. Ford and G. N. Thorne, "Effects of Atmospheric Humidity on Plant Growth," vol. 38, no. 2, pp. 441-452, 1974.

[10] L. Ada, "TMP36 Temperature Sensor," 29 July 2012. [Online]. Available: https://can01.safelinks.protection.outlook.com/?url=https%3A%2F%2Flearn.adafruit.com%2Ftmp36-temperature-sensor&data=05%7C01%7C21eo4%40queensu.ca%7Cd741e79f74b54211fbb008dac04100cd%7Cd61ecb3b38b142d582c4efb2838b925c%7C1%7C0%7C638033682658132077%7CUnknown%. [Accessed 6 November 2022].

[11] R. Keeting, "How does the temperature affect the plant growth?," Science Projects, 17 April 2016. [Online]. Available: https://can01.safelinks.protection.outlook.com/?url=https%3A%2F%2Fwww.scienceprojects.org%2Fhow-does-the-temperature-affect-the-plant-growth%2F&data=05%7C01%7C21eo4%40queensu.ca%7Cd741e79f74b54211fbb008dac04100cd%7Cd61ecb3b38b142d582c4efb2838b925c%7C1%7C0. [Accessed 6 November 2022].

[12] "LMT86LPGM," Texas Instruments, 5 February 2010. [Online]. Available: https://can01.safelinks.protection.outlook.com/?url=https%3A%2F%2Fwww.digikey.ca%2Fen%2Fproducts%2Fdetail%2Ftexas-instruments%2FLMT86LPGM%2F9462684%3Futm_adgroup%3DTexas%2520Instruments%26utm_source%3Dgoogle%26utm_medium%3Dcpc%26utm_campaign%3DPMax%3A%252. [Accessed 6 November 2022].

[13] "Photoresistor," EEPower, [Online]. Available: https://eepower.com/resistor-guide/resistor-types/photo-resistor/#. [Accessed 6 November 2022].

[14] ArduinoModules, "KY-018 PHOTORESISTOR MODULE," Arduino Modules, 3 December 2021. [Online]. Available: https://arduinomodules.info/ky-018-photoresistor-module/. [Accessed 4 December 2022].

[15] "Photosynthesis," BYJU'S, [Online]. Available: https://byjus.com/biology/photosynthesis/. [Accessed 6 November 2022].

[16] P. Robinson, "How much light do my indoor plants need?," PlantMaid, 2020 June 1. [Online]. Available: https://www.plantmaid.com/how-much-light-do-my-indoor-plants-need/. [Accessed 6 November 2022].

[17] ECOgardener, "4 Factors That Affect Plant Growth," ECOgardener, 4 March 2018. [Online]. Available: https://ecogardener.com/blogs/news/4-factors-that-affect-plant-growth#:~:text=Water%2C%20light%2C%20temperature%2C%20and,not%20given%20the%20proper%20nutrients.].. [Accessed 3 December 2022].

[18] B. Hall, "Is PPE required when working on 120-volt circuits?," Rozel, 25 November 2020. [Online]. Available: https://www.70econsultants.com/is-ppe-required-when-working-on-120-volt-circuits/. [Accessed 4 December 2022].

[19] J. Michael, "How Long Does It Take For Seeds To Sprout?," Backyard Digs, 12 May 2021. [Online]. Available: https://www.backyarddigs.com/gardening/how-long-does-it-take-for-a-seed-to-germinate/. [Accessed 3 December 2022].

[20] "TRAINING COURSE (TWO DAYS) (QUALIFIED ELECTRICAL WORKER [QEW] IN SASKATCHEWAN)," Team Power Solutions, [Online]. Available: https://www.teampowersolutions.ca/high-voltage-electrical-safety-qualified-electrical-worker-training. [Accessed 4 December 2022].

[21] P. Hearn, "This is how much it costs to run a smart garden, and how much it can save," Digital Trends, 5 November 2022. [Online]. Available: https://www.digitaltrends.com/home/true-cost-smart-garden/. [Accessed 4 December 2022].

[22] M. Kalajdzieva, "What is the Average Cost of Food Per Person in Canada?," ReviewLution, 29 November 2022. [Online]. Available: https://reviewlution.ca/resources/average-cost-of-food-per-person-in-canada/. [Accessed 4 December 2022].

[23] L. Knerl, "The Cost Effectiveness of Growing a Garden," Investopedia, 1 May 2022. [Online]. Available: https://www.investopedia.com/financial-edge/0312/the-true-cost-of-growing-a-garden.aspx. [Accessed 4 December 2022].

[24] L. Garfield, "Food prices are insanely high in rural Canada, where Ketchup costs $14 and Sunny D costs $29," Business Insider, 21 September 2022. [Online]. Available: https://www.businessinsider.com/food-prices-high-northern-canada-2017-9. [Accessed 4 December 2022].

[25] S. Moonilal, "Vertical Farming: The Future of Agriculture?," CENGN, 1 March 2022. [Online]. Available: https://www.cengn.ca/information-centre/innovation/vertical-farming-the-future-of-agriculture/#:~:text=Vertical%20farms%20use%20significantly%20fewer,to%20control%20the%20growing%20environment.. [Accessed 4 December 2022].

[26] "6 Reasons Why Indoor Gardening Is Great for the Environment," Novagrow, [Online]. Available: https://novagrow.io/en/blog/6-reasons-why-indoor-gardening-is-great-for-the-environment/#:~:text=You%20reduce%20your%20ecological%20footprint,just%2010%20to%2015%20days.].. [Accessed 4 December 2022].

[27] "Environmental factors affecting plant growth," Oregon State Univsersity, [Online]. Available: https://extension.oregonstate.edu/gardening/techniques/environmental-factors-affecting-plant-growth. [Accessed 4 December 2022].

[28] "Wi-Fi Protected Access 2 (WPA2)," Gartner, [Online]. Available: https://www.gartner.com/en/information-technology/glossary/wpa2-wi-fi-protected-access-2. [Accessed 4 December 2022].

Appendix

Table 4: Work breakdown structure for the duration of the P2 Project

Activity Identifier (Based on which week task takes place)	Task Description	Expected Duration	Actual Duration	Group/Individual
8-12	Constructing a prototype	20 days	TBD	Team
8	P2 proposal	5 days	TBD	Team
9	P2 Progress report 1	4 days	TBD	Team
10	Career Readiness	3 days	TBD	Individual
11	P2 Progress report 2	4 days	TBD	Team
12	P2 Final report	7 days	TBD	Team
12	P2 Presentation slides	4 days	TBD	Team
12.1	P2 Oral Presentation	1 day	TBD	Team & Individual
13	P2 Peer review	1 day	TBD	Individual

Table 5: Important milestones for P2 Project

Week #	Description of Activity	Status
8	Choose P2 Project and submit proposal	Complete
9	Consolidate feedback and submit P2 Progress report 1	Complete
10	Consolidate feedback and submit P2 Progress report 2	Complete
11	Review, edit and submit P2 final report	Complete
12	Oral Presentation (P2)	In progress

TASK	ASSIGNED TO	PROGRESS	START	END
Weeks 7-9				
P2 Proposal	All Group Member	100%	11-2-22	11-6-22
meeting	Team	100%	11-3-22	11-5-22
research	Individual -All Group Members	100%	11-3-22	11-5-22
write	Team	100%	11-3-22	11-5-22
edit	Team	100%	11-3-22	11-5-22
meeting	Team	100%	11-3-22	11-5-22
P2 P.R. 1	All Group Members	100%	11-8-22	11-12-22
meeting	Team	100%	11-9-22	11-11-22
research	Individual -All Group Members	100%	11-9-22	11-11-22
write	Team	100%	11-9-22	11-11-22
edit	Team	100%	11-9-22	11-11-22
meeting	Team	100%	11-9-22	11-11-22
Weeks 10-12				
Career Readine	Indiviudal -All Group Members	100%	11-16-22	11-20-22
P2 P.R. 2	All Group Members	100%	11-23-22	11-27-22
meeting	Team	100%	11-24-22	11-26-22
research	Individual -All Group Members	100%	11-24-22	11-26-22
write	Team	100%	11-24-22	11-26-22
edit	Team	100%	11-24-22	11-26-22
meeting	Team	100%	11-24-22	11-26-22
Prototype	Team	100%	11-12-22	12-2-22
Photo resistor	Team	100%	11-12-22	11-15-22
Hydrometer	Team	100%	11-16-22	11-19-22
Thermometer	Team	100%	11-20-22	11-23-22
Ultrasonic sens	TEam	100%	11-24-22	11-27-22
P2 Final Report	All Group Members	100%	11-30-22	12-4-22
metting	Team	100%	12-1-22	12-3-22
research	Individual -All Group Members	100%	12-1-22	12-3-22
write	Team	100%	12-1-22	12-3-22
edit	Team	100%	12-1-22	12-3-22
meeting	Team	100%	12-1-22	12-3-22
P2 Slides	All Group Members	100%	11-30-22	12-4-22
P2 Presentatio	Indiviudal & All Group Members	50%	12-5-22	12-8-22
Rehersal	All Group Members	80%	12-6-22	12-7-22
P2 Peer Review	Individual -All Group Members	0%	12-9-22	12-9-22

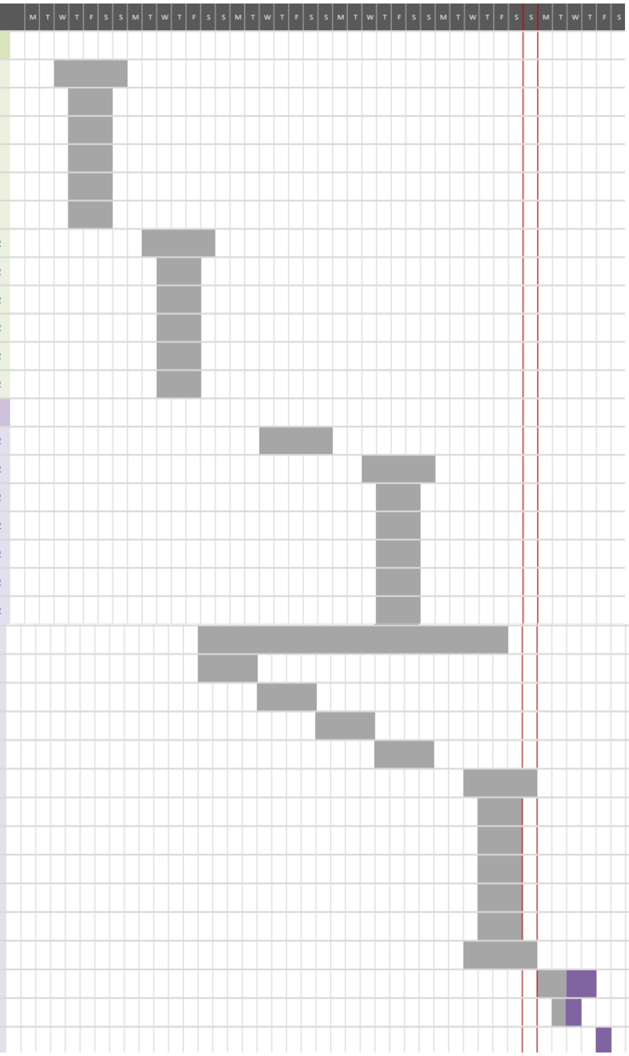

Figure 6: Gantt Chart for the duration of the P2 project

```
#include <WiFiNINA.h>
#include <math.h>
#define echoPin 2 // attach pin D2 Arduino to pin Echo of HC-SR04
#define trigPin 3 //attach pin D3 Arduino to pin Trig of HC-SR0
char ssid[] = "";             //  your network SSID (name) between the " "
char pass[] = "";      // your network password between the " "
int keyIndex = 0;                 // your network key Index number (needed only for WEP)
int status = WL_IDLE_STATUS;      //connection status


long duration; // variable for the duration of sound wave travel (Ultrasonic)
int distance; // variable for the distance measurement (Ultrasonic)
const int AirValue = 469;   //value of 0% moisture
const int WaterValue = 310;  //value of 100% moisture
int soilMoistureValue = 0; //voltage recieved from sensor
int soilmoisturepercent = 0; //moisture % read from sensor

int photoPin = A4; //analog pin for photoresistor

int heatPin = 8; //digital pin for heater (Red LED)
int lightPin = 12; //digital pin for grow light (Blue LED)
int relayPin = 5; //digital pin for relay (Water Pump)
int light; //stores value read from the photoPin

int LMT86 = A0; //analog pin for temperature sensor
int A0_Read = 0; //stores value read from temperature sensor
float Temperature = 0; // manipulated temperature reading value to give celcius reading.
//Array to store info about plants
//0 is chives,  1 is tomatoes. rows 0 & 1 are for temp, rows 2 & 3 are for light, rows 4 & 5 are for moisture, row 4 is for the time it takes to grow
int plantsThreshold [2][7] = {{15, 23, 6, 8, 60, 70, 60}, {15, 34, 6, 8, 60, 80, 90}};

int emptyTank = 6; //distance away from sensor when tank is empty [cm]
int selection = -1; //variable that is changed depending on what plant the user picks to grow. pulls differnet values from plantsThreshold array
int tempStatus = 0; //will turn to 1 if heat is needed/ temperature drops below threshold temp
int lightStatus = 0;//will turn to 1 if light is needed/ light below threshold time
int lightCounter = 0; //counts how many hrs of light are recieved, resets daily
int moistureStatus = 0; //will turn to 1 if plant needs water / moisture % below threshold time
int tankStatus = 0 //will turn to 1 if tank needs water / threshold distance is reached for ultrasonic sensor
int minutes = 0; //counts minutes passed, resets every hour
int hours = 0; //counts hours passed, resets every day
```

Figure 7: Program written in the Arduino IDE that is responsible for operating the autonomous system

```
int days = 0; //counts days passed
int printTemp; //temperature value in c (truncated)
int printMoisture; //moisture percent
int endProcess = 0; //if 1, loop ends
int lightConstant = 100; //threshold light value
int lightTime; //time for how much extra light is needed
int finished = 0; //will print to harvest if 1
int testnumber = 0; //what test is selected to preform by the user
WiFiServer server(80);           //server socket

WiFiClient client = server.available();
int sensorValue = 8;
//setup pins and connect to wifi
void setup() {
  Serial.begin(9600);
  pinMode(LED_BUILTIN, OUTPUT);
  pinMode(lightPin, OUTPUT);
  pinMode(heatPin, OUTPUT);
  pinMode(relayPin, OUTPUT);
  digitalWrite(relayPin, HIGH);
  pinMode(trigPin, OUTPUT); // Sets the trigPin as an OUTPUT
  pinMode(echoPin, INPUT); // Sets the echoPin as an INPUT

  while (!Serial)

  enable_WiFi();
  connect_WiFi();
  server.begin();
  printWifiStatus();
}
int minute = 0;
//loop that runs while the arduino is active
void loop() {
  if(endProcess == 1){
    exit(0);
  }
  //test section or nothing selected
  if (selection == 2 || selection == -1){
    Serial.println(testnumber);
    printWifiStatus();
```

```
    getTemp();
    getMoisture();
    getLight();
    getWaterDistance();
    decisions();
    delay(3000);
    if(testnumber == 1){
      if(light < 150){
        digitalWrite(lightPin, LOW);
      }
      else{
        digitalWrite(lightPin, HIGH);
      }
    }
    if (testnumber != 1){
      digitalWrite(lightPin, LOW);
    }
    if(testnumber == 2){
      if(soilmoisturepercent < 60){

        digitalWrite(relayPin, LOW);
        delay(2000);
        digitalWrite(relayPin, HIGH);
      }
      else{

        digitalWrite(relayPin, HIGH);
        delay(6000);

      }
    }
    if(testnumber == 3){
      if(distance < 6){
        digitalWrite(relayPin, LOW);
        delay(6000);
        digitalWrite(relayPin, HIGH);
        delay(6000);
      }
```

```
      else{
        digitalWrite(relayPin, HIGH);
        delay(6000);
      }

    }
    if(testnumber == 4){
      if(printTemp > 20){
        digitalWrite(heatPin, LOW);
        delay(250);
      }
      else{
        digitalWrite(heatPin, HIGH);
        delay(6000);
        digitalWrite(heatPin, LOW);
        }
    }
    if(testnumber != 4){
      digitalWrite(heatPin, LOW);
    }
    client = server.available();
    if(client){
      printWEB();
    }
    delay(500);
  }
  //actual plant growing, starts clock for minutes as well
  else{
  if(days == plantsThreshold[selection][6]){
    finished = 1;
  }
    Serial.print("The value is printed after: ");
    Serial.print(minute);
    Serial.print(" minute\n");
    minute = minute+1;
    getTemp();
    getMoisture();
    getLight();
    getWaterDistance();
    Serial.println(selection);
```

```
  if(selection != -1){
    decisions();
  }
  if(minute == 60){
    //To add: if the light counter isn't 0, subtract 1 from it.
    if(lightTime != 0){
      lightTime = lightTime - 1;
    }
    hours = hours+1;
    //replace threshold for light with readable value
    if(lightStatus == 1){
      lightCounter = lightCounter+1;
    }
    minute = 0;
    if(hours == 24){
      days = days+1;
      if(lightCounter < plantsThreshold[selection][2]){
        lightTime = plantsThreshold[selection][3] - lightCounter;
        if(lightTime < 0){
          lightTime = 0;
        }
      }
      lightCounter = 0;
      hours = 0;
    }
  }
  if(lightTime != 0){
    digitalWrite(lightPin, HIGH);
  }
  else{
    digitalWrite(lightPin, LOW);
  }
  if(tempStatus == 1){
    digitalWrite(heatPin, HIGH);
  }
  else{
    digitalWrite(heatPin, LOW);
  }
  int i = 0;
  if(moistureStatus == 1){
```

```
      Serial.println("pump on");
      digitalWrite(relayPin, LOW);
      delay(3953);
      digitalWrite(relayPin, HIGH);
      i = 67;
    }
    // turn on pins for certain functions
    while(i < 1000){
      client = server.available();
      if(client){
        printWEB();
      }
      i = i+1;
      delay(59);
    }
    }
  // delay(59000);
}
//prints the status of the wifi conenction to the terminal
void printWifiStatus() {
  // print the SSID of the network you're attached to:
  Serial.print("SSID: ");
  Serial.println(WiFi.SSID());

  // print your board's IP address:
  IPAddress ip = WiFi.localIP();
  Serial.print("IP Address: ");
  Serial.println(ip);

  // print the received signal strength:
  long rssi = WiFi.RSSI();
  Serial.print("signal strength (RSSI):");
  Serial.print(rssi);
  Serial.println(" dBm");

  Serial.print("To see this page in action, open a browser to http://");
  Serial.println(ip);
}
```

```
void enable_WiFi() {
  // check for the WiFi module:
  if (WiFi.status() == WL_NO_MODULE) {
    Serial.println("Communication with WiFi module failed!");
    // don't continue
    while (true);
  }

  String fv = WiFi.firmwareVersion();
  if (fv < "1.0.0") {
    Serial.println("Please upgrade the firmware");
  }
}

void connect_WiFi() {
  // attempt to connect to Wifi network:
  while (status != WL_CONNECTED) {
    Serial.print("Attempting to connect to SSID: ");
    Serial.println(ssid);
    // Connect to WPA/WPA2 network. Change this line if using open or WEP network:
    status = WiFi.begin(ssid, pass);

    // wait 10 seconds for connection:
    delay(10000);
  }
}
//functuion to communitate with user with words, sensor values, and hyperlinks for the user to make decision.
```

```
void printWEB() {

  if (client) {                             // if you get a client,
    Serial.println("new client");           // print a message out the serial port
    String currentLine = "";                // make a String to hold incoming data from the client
    while (client.connected()) {            // loop while the client's connected
      if (client.available()) {             // if there's bytes to read from the client,
        char c = client.read();             // read a byte, then
        Serial.write(c);                    // print it out the serial monitor
        if (c == '\n') {                    // if the byte is a newline character

          // if the current line is blank, you got two newline characters in a row.
          // that's the end of the client HTTP request, so send a response:
          if (currentLine.length() == 0) {

            // HTTP headers always start with a response code (e.g. HTTP/1.1 200 OK)
            // and a content-type so the client knows what's coming, then a blank line:
            client.println("HTTP/1.1 200 OK");
            client.println("Content-type:text/html");
            client.println();

            //create the buttons
            // client.print("Click <a href=\"/H\">here</a> turn the LED on<br>");
            // client.print("Click <a href=\"/L\">here</a> turn the LED off<br><br>");

            //only show status if a plant is already growing? Also dont offer to grow something if plant is already growing
            client.print("<body bgcolor=”#f0e926>");

            //put all readings on screen for user to see, temp, need to refil water, etc.
            if(finished == 1){
              client.print("<center>Please harvest plant and press the 'Force End' button.</center><br>");
            }
            else if(selection == -1){
              client.print("<center>What would you like to grow?:</center><br>");
              client.print("<center><a href=\"/Chives\">Chives</a></center><br>");
              client.print("<center><a href=\"/Tomatoes\">Tomatoes</a></center><br>");
              client.print("<center><a href=\"/LiveDemo\">Live Demonstration</a></center><br>");
              //if done, print that the process has finished and the plant must be harvested, else say how much time left on clock.
              //force end button, maybe have like an "all done button when the plant is harvested?"
            }
            else{
              client.print("<center>CURRENT STATUS:</center><br>");
              client.print("<center>");
              client.print("Temperature: ");
              client.print(printTemp);
              client.print("<span>&#176;</span>C");
              client.print("<center/><br>");
              if(tempStatus == 1){
                client.print("<center>");
                client.print("The heater is currently ON");
                client.print("<center/><br>");
              }
              else{
                client.print("<center>");
                client.print("The heater is currently OFF");
                client.print("<center/><br>");
              }
              client.print("<center>");
              client.print("Soil Moisture Percentage: ");
              client.print(printMoisture);
              client.print("%");
              client.print("<center/><br>");
              if(lightTime != 0){
                client.print("<center>");
                client.print("Light is currently ON");
                client.print("<center/><br>");
              }
              else{
                client.print("<center>");
                client.print("Light is currently OFF");
                client.print("<center/><br>");
              }
              if(tankStatus == 1){
                client.print("<center>");
                client.print("The water tank is running low. Please refill as soon as possible.");
                client.print("<center/><br>");
              }
```

```
              else{
                  int tankPercent = (distance)/(emptyTank) * 100;
                  client.print("<center>");
                  client.print("The water tank is ");
                  client.print(tankPercent);
                  client.print("% full");
                  client.print("<center/><br>");

              }
              if(selection == 2){
                client.print("<center><a href=\"/testlight\">Test Light</a></center><br>");
                client.print("<center><a href=\"/moisturesensor\">Test Moisture Sensor</a></center><br>");
                client.print("<center><a href=\"/emptytank\">Test Empty Tank</a></center><br>");
                client.print("<center><a href=\"/raiseheat\">Raise Heat Threshold</a></center><br>");
              }
              client.print("<center><a href=\"/END\">Force End</a></center><br><br>");
              client.println();
            }
            break;
          }
          else {      // if you got a newline, then clear currentLine:
            currentLine = "";
          }
        }
        else if (c != '\r') {    // if you got anything else but a carriage return character,
          currentLine += c;      // add it to the end of the currentLine
        }

        if (currentLine.endsWith("GET /Chives")) {
          digitalWrite(LED_BUILTIN, HIGH);
          selection = 0;

        }
        if (currentLine.endsWith("GET /Tomatoes")) {
          digitalWrite(LED_BUILTIN, LOW);
          selection = 1;
```

```
        }
        if (currentLine.endsWith("GET /LiveDemo")) {
          digitalWrite(LED_BUILTIN, LOW);
          selection = 2;
        }
        if (currentLine.endsWith("GET /Force End")) {
          digitalWrite(LED_BUILTIN, LOW);
          endProcess = 1;
        }
        if (currentLine.endsWith("GET /testlight")) {
          digitalWrite(LED_BUILTIN, LOW);
          testnumber = 1;
        }
        if (currentLine.endsWith("GET /moisturesensor")) {
          digitalWrite(LED_BUILTIN, LOW);
          testnumber = 2;
        }
        if (currentLine.endsWith("GET /emptytank")) {
          digitalWrite(LED_BUILTIN, LOW);
          testnumber = 3;
        }
        if (currentLine.endsWith("GET /raiseheat")) {
          digitalWrite(LED_BUILTIN, LOW);
          testnumber = 4;
        }

      }
    }
    // close the connection:
    client.stop();
    Serial.println("client disconnected");
  }
}
```

```
//add function that decides what needs to be turned on
void decisions(){
  if(soilmoisturepercent < plantsThreshold[selection][4]){
    if (distance >= 6){
      moistureStatus = 0;
      tankStatus = 1;
      }
    else if(distance < 6){
      tankStatus = 0;
      moistureStatus = 1;
      }
  }
  else {
    moistureStatus = 0;
  }
  if (printTemp < plantsThreshold[selection][0]){
    tempStatus = 1;
  }
  else{
    tempStatus = 0;
  }
  if(light < lightConstant){
    lightStatus = 1;
  }
  else{
    lightStatus = 0;
  }
}
```

```
int getTemp(){
  A0_Read = analogRead(LMT86);
  // Serial.println(A0_Read);
  Temperature = (426-A0_Read) / 2.14;
  Serial.print("Temperature: ");
  Serial.print(Temperature, 1);
  Serial.println(" C");
  printTemp = (int)Temperature;
  delay(250);
  return (int)round(Temperature);
}
//function that reads and calculates the soil moisture percent, stores it and prints to the terminal
int getMoisture(){
  soilMoistureValue = analogRead(A2);  //put Sensor insert into soil
  Serial.print("Soil Moisture: ");
  // Serial.println(soilMoistureValue);
  soilmoisturepercent = map(soilMoistureValue, AirValue, WaterValue, 0, 100);
  if(soilmoisturepercent >= 100)
  {
    soilmoisturepercent = 100;
    Serial.println("100 %");
  }
  else if(soilmoisturepercent <=0)
  {
    soilmoisturepercent = 0;
    Serial.println("0 %");
  }
  else if(soilmoisturepercent >0 && soilmoisturepercent < 100)
  {
    Serial.print(soilmoisturepercent);
    Serial.println("%");
  }
  printMoisture = soilmoisturepercent;
  delay(250);
  return soilmoisturepercent;
}
```

```
//function that reads the light value, stores it and prints to the terminal

int getLight(){
  light = analogRead(photoPin);
  Serial.print("PhotoResistor: " );
  Serial.println(light);
  delay(250);
  return light;
}
//function that reads and calculates the distance of the water, stores it and prints to the terminal

int getWaterDistance(){
  digitalWrite(trigPin, LOW);
  delayMicroseconds(2);
  // Sets the trigPin HIGH (ACTIVE) for 10 microseconds
  digitalWrite(trigPin, HIGH);
  delayMicroseconds(10);
  digitalWrite(trigPin, LOW);
  // Reads the echoPin, returns the sound wave travel time in microseconds
  duration = pulseIn(echoPin, HIGH);
  // Calculating the distance
  distance = duration * 0.034 / 2; // Speed of sound wave divided by 2 (go and back)
  // Displays the distance on the Serial Monitor
  Serial.print("Distance: ");
  Serial.print(distance);
  Serial.println(" cm");
  delay(250);
  return (int)distance;
}
```

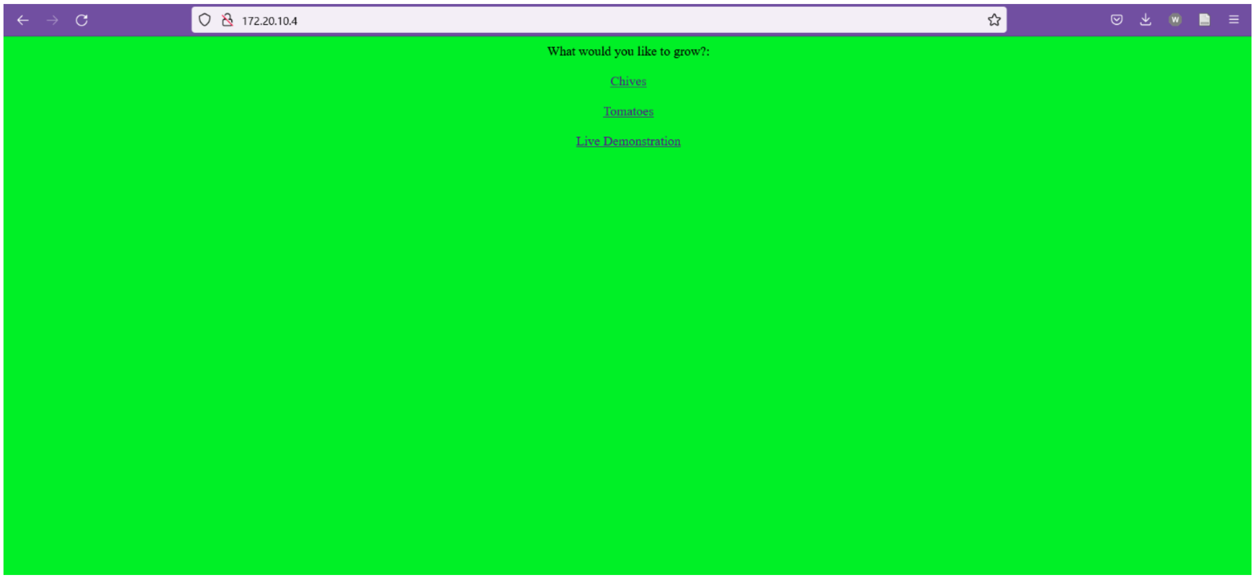

Figure 8: Depiction of what the user sees when they open the server

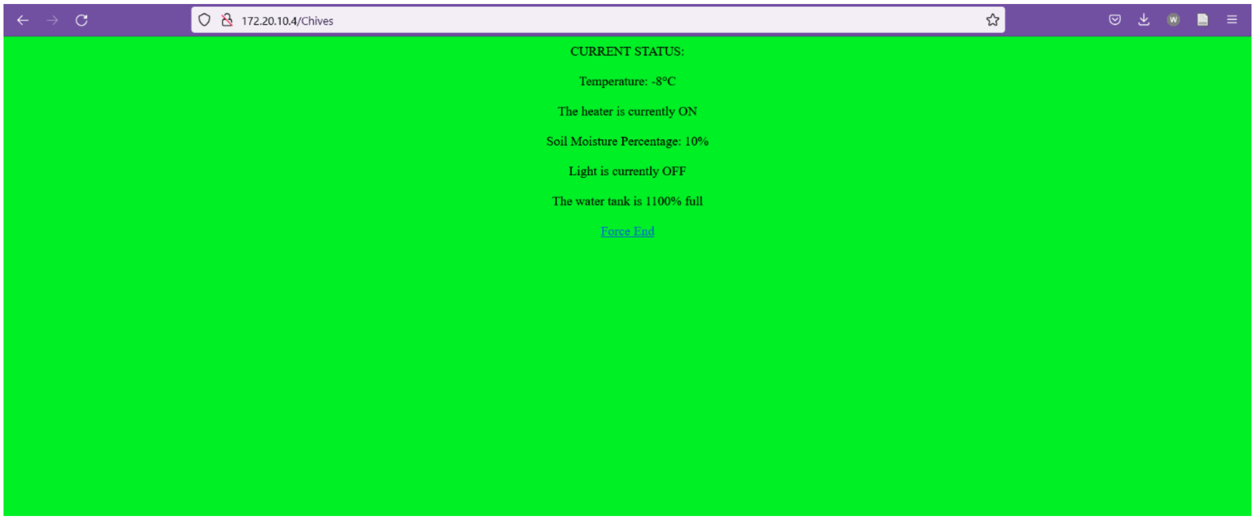

Figure 9: Depiction of what the user sees after they begin growing a plant