EEL 4915C
Senior Design

Modular Hydroponic System

Group 7 Members:
Adam Loree | Computer Engineering
Don Ellington | Computer Engineering
Brandon Mitchell | Computer Engineering
Jonathan Rodriguez | Electrical Engineering




Table of Contents

List of Figures

List of Tables

1 Executive Summary

For our Senior Design project, this document lays out the research, design, technical objectives, and implementation of a modular hydroponics system. The goal of many hydroponics systems is to make growing plants and food simple, easy, and automated to some degree. Our project strives to have a hardware and software component that works together to allow for the growth of a plant while being able to check it's status through the internet on a laptop or other smart device.

In order to create and monitor an environment sustaining growth, our hydroponics system will be modular with two main hardware components: a control unit and plant unit. There will be software components supplementing the hydroponics system that consist of a website, mobile application, database storage, a webserver, and microcontroller processes. Our system fosters a "stem" and "leaf" concept where the control unit stores and can send nutrients to the plant unit. The control unit will contain the components necessary to supply and store nutrients for the plant within its plant unit. The plant unit will contain the plants and sensors that monitor the plants' health. Enough nutrients will also be monitored to maintain the plant growth independent of the central unit.

The plant and control unit will be combined when resupply is necessary, but each unit will be able to operate separately. The intention is to allow for a more convenient system to allow for plant growth without the rest of the bulky components. The plant unit will be able to gather and communicate plant and nutrient data through a Wi-Fi and Bluetooth module that will be stored on an online database and viewed through a smart device or website. Our system will collect temperature, humidity, light, pH, total dissolved solids, and water levels. The user will be able to connect the plant unit and control unit to resupply the plant unit. The plant unit can be scaled in quantity so that multiple units can resupply to a single control unit. All the components and subsystems working together should allow for a hydroponics system that is flexible to the user's daily routine and living circumstances.

2 Project Description

Chapter 2 will describe the project introduction, goals, specifications, and house of quality. This section is intended to give the viewer a foundation of understanding to why we are focusing on this specific type of technology and what we could improve upon.

2.1 Introduction/Background

The world today is constantly competing for control over the dwindling resources that are becoming harder to obtain. Rural agriculture for thousands of years relied on land and soil to grow and maintain crops to make food and support civilizations. This same technique is being implemented today, however there is a scarcity of arable land due to climate change, expanding populations, and pollution. Within the past few hundreds of years scientists have been experimenting and improving upon the idea of hydroponics.

Hydroponics combines the english words *hydro*, meaning water, and the greek word *ponos,* meaning labor, to describe this process of farming. This farming technique relies mainly on water to conduct the sustaining and maintaining of plants without the need for soil. However, this definition applies to plants that can be supported with gravel and other materials to provide structure.

In the current times, hydroponics is a common investment for people to provide fresh food for themselves. The advantage of hydroponics is the reduction in water and space needed to grow plants. However, the disadvantage of hydroponics is the upfront cost, installation, and constant attention required.

Hydroponics allows for higher plant density. While traditional farming requires large stretches of soil for plants to be put into the ground, hydroponics takes a different approach to this. A hydroponic system can have an exponentially greater plant density than traditional farming techniques [3]. While traditional farming requires the use of the ground and can only be planted at certain times of the year hydroponics can be planted year round. Hydroponic systems can also be stacked on top of each other or placed in very close proximity to one another without too many problems. This close proximity also helps with the germination of plants whereas in traditional farming farmers have to rely on insects and animals to germinate the plants [3].

Hydroponic systems have the ability to control the specific set of nutrients that are involved in plant growth. Nutrients have always been an important part in plant growth and farmers have always used nutrients to change the composition of their soil with fertilizers and other types of soil helpers. Hydroponics farming allows for complete control over the nutrients that the plants encounter.

Having control over the nutrients allows for enhanced plant yields. The culmination of nutrient control, Ph control and light control allows for this. All these different aspects lead to a more affordable way to grow plants. The only downside to hydroponics is the cost of electricity. Hydroponics costs more because you need to run lights on the plants while they are growing and in our case we need electricity for the pumps and the microcontroller.

2.2 Project Motivation and Goals

Our group saw that shifting the production of food from soil to a system that could be implemented anywhere would be impactful and we wanted to improve upon it.

Besides the increased water efficiency of hydroponics and flexibility of growing location, hydro farming can also increase the supply and variety of food in diets that may lead to an overall increase of health and happiness for people.

We are determined to create a more modular and flexible system than the typical hydroponics setup that can be scaled and shaped to the users needs while still providing sufficient food output.

2.3 Objectives

Our goal is to design a modular automatic hydroponic system. The basic goal is to provide fresh garden goods to any home, but there are many factors to be considered to build a device like this. Listed below are some goals we strive to achieve with this project .

- Make Organic vegetables accessible for any household
- Make a seamless user-friendly gardening experience
- Optimize the plant growth process
- Fully automate the gardening experience
- Create a high performing, yet aesthetic home appliance
- Have a sleek and modular structure for a flexible, customizable design

This system should be small enough to be stored in any room of a house and modular so the device can be stacked and sorted in any arrangement. We want the customer to see this product as any other kitchen appliance, a device with reliable performance and a sleek and minimal design that can fit almost anywhere and accent the room. The system is the perfect solution for those who do not have an outdoor space for gardening or any experience growing plants. The system will also be light enough to be easily transported to the central unit. Even at full capacity, the garden pod will be well under 50 pounds making it easily transportable. Keeping the garden pod small and light is very important to ensuring all people can take advantage of its growing features. At the bare minimum, this product will be able to grow plants with minimal human interaction and resource costs as effectively as possible. Maximum yield, minimum effort.

2.4 Requirements Specifications

The hydroponics system is intended to be a fully automatic garden apparatus. This system will house multiple sensors to measure water, pH, nutrient levels, temperature, and humidity levels. To ensure each crop has an environment best suited for plant growth. The central unit will combine water, pH solutions, and other essential plant nutrients via peristaltic pumps to produce a well-balanced plant feed best suited for whatever crop is grown and then deliver the solution to the peripheral Garden Pod to irrigate the plants in an ebb-and-flow fashion. The Garden Pod will be powered by a battery such that it will be removable and able to fully function separate from the central unit. When it is time to replenish the water, The central unit will recycle the water and readjust the pH through the main water reservoir. The Garden Pod will also feature specialized UV LEDs, The LEDs will be especially designed for horticultural applications and will supply the plants with sufficient supplemental light, The device will also be able to adjust the lighting to suit a specific crop's needs. The central unit will be powered by an AC voltage source and could be plugged into any traditional american outlet which can be found in any household.

Software will be developed to allow monitoring and management of the hydroponics system through a website and mobile application. The temperature, humidity, TDS, pH, water level, and light status will be the readings available on the sensor data page. To ensure consistent user interface, the mobile application and website will be as similar as possible. The mobile application and website will also record data from the sensors that can be accessed and controlled by the user. The sensor data will be displayed in a very readable format so that the user can completely understand the status of their garden pods. The sensor data will be stored in a database, lending the opportunity for the user to view previous data that is not real time if that is something they would like to do.

	Market	Value
X	The system must be at a reasonable weight	<50lbs
	The software will display necessary data in a nice user interface	pH, Light, Water quality
	The system shall be able to communicate with a device for controls and data	WiFi, Bluetooth
	The unit must not leak into surrounding electronics	N/A
	The software will be able to control the system remotely	WiFi

Table 2-1: Market Specifications

Engineering		Value
	The system must be able to self regulate PH	Within 0.01 increment accuracy (0-14)
	The system must be able to self regulate nutrient levels	Within 1.0 accuracy (mg/L)
	The system will have sufficient water reservoir capacity	5 Gallons
	The system will efficiently convert AC to DC power	120V AC -> 12V, 5V DC
	The system will effectively communicate information wirelessly via Bluetooth	1Mbps (9m/30ft)
X	The system will refill the garden pod quickly	< 30 seconds

Table 2-2: Engineering Specification

Since this is our first iteration of a design document the constraints we face are not very palpable. The ones that we were able to brainstorm follow the constraints of any group project: time management, cost, research… etc. We expect that our constraints will lie in creating a suitable management system for our treated water to be given to each unit. This must be done properly and without leaks. Since the electronics are in such close proximity this could be detrimental.

In **Table 2-2** the engineering specifications are laid out. These specifications can be used to drive our project constraints. To be able to verify and implement each of these specifications could take a lot of effort and time. There is also a need to keep the market specifications laid out in **Table 2-1** in mind when creating the project to give the product a more user friendly feel. The items to be demonstrated are marked by an X.

2.5 House of Quality

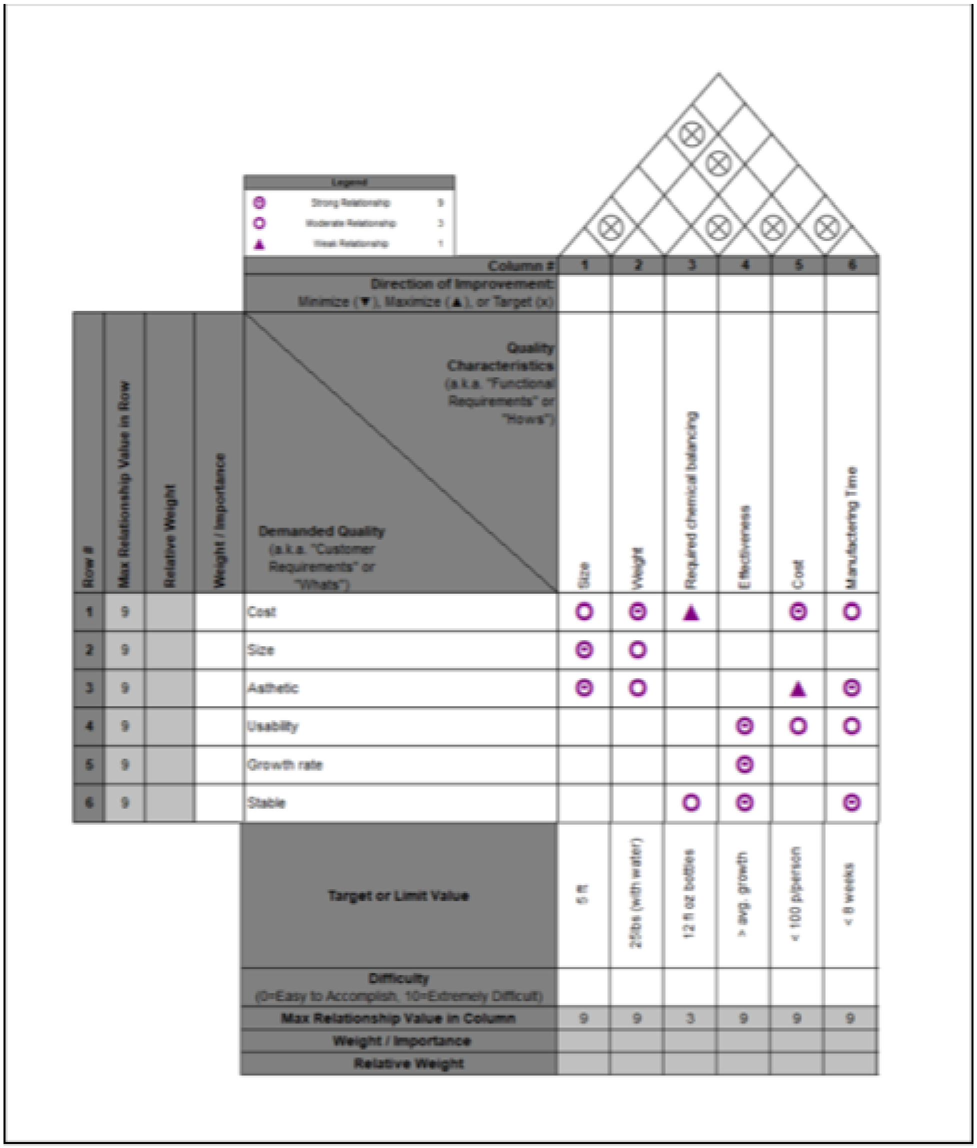


Figure 2-1: House Of Quality

The house of quality describes the relationship between multiple different facets of the entire project; the link between them is described in **Figure 2-1** and it goes on to identify whether it is a strong or weak relationship. This helps to identify what may be important to us as a whole or what could be overlooked. It also helps to identify target values of each functional requirement.

3 Research Related to Project Definition

Chapter 3 covers technology old and new that is related to hydroponics. This technology ranges from API research (software sided) to sensors and pumps (hardware sided). This chapter starts with a look at projects related to our project and how we can use the tools they used for their project in our project. It then goes on to show research on different apps that could be similar to what we aim to create for our project. The next section is on different APIs useful for finding grow times and specifics for different plants. The chapter goes on to look at different types of horticulture lighting, irrigation, pumps, water level sensors, humidity sensors, pH sensors, Bluetooth, wifi, and assorted specific components. The chapter finishes by taking a look at possible architectural designs and the science of nutrients and pH in hydroponics. The chapter specifies every component that is considered for a part and why a specific selection is made.

3.1 Existing Similar Projects and Products

There are many devices and systems geared toward gardening. These systems are centered towards optimizing plant growth, which at face value simply requires water, dirt and light, but each system has its own method of accelerating plant growth, while balancing efficiency, maintenance and cost. When it comes to agriculture, there are a multitude of factors that could be considered. There are many examples, but also much potential for projects of this nature.

3.1.1 Hydroponic Systems

There are also professional companies who have created their own industrial design for indoor agriculture, but vertical and hydroponic. These businesses not only franchise these automated farms but also provide resources to students interested in the subject. The materials offer plenty of insight on common standards and practices, and provide a good vantage point in exploring our design.

Figure 3-1: Hydroponic Systems Example

3.1.1.1 Home Hydroponics

A one example is a hydroponic system designed, in the Fall of 2017, by a senior design team in this very same institution. This hydroponic system would monitor, and deliver, PH-balanced water and light to a garden system. This system was capable of Wifi communications and was powered by an AC voltage source. This project will function similar to this hydroponic system, however our design is focused on portability and connectivity.

3.1.1.2 Greenery

Greenery is a state-of-the-art Indoor vertical hydroponic system, housed inside customized shipping containers. This system was developed by Freight Farms, Inc. all sold worldwide to be used in almost any climate. Their systems spare no expense and have a wide range of features that provide any sort of benefit to plant growth, including but not limited to a bluetooth audio system (Some researchers theorize sound waves can stimulate plant growth). While this is the perfect example of a quality hydroponic system, a system of this caliber would require major levels of funding and research, and not feasible for the scope of this project.

3.1.1.3 Hydropotomous

Hydropotomous, a 2014 senior design project from Colombia university that designed a hydroponics system using clear plastic (acrylic) for the majority of the device. The interior of the systems is transparent due to the material and because of that group's desire to peer into the processes of the system. Their hydroponic system is a nutrient film technique (NFT) which requires a minimum of a water reservoir, water pump, plant channel, and controls for any electronics like Arduino.

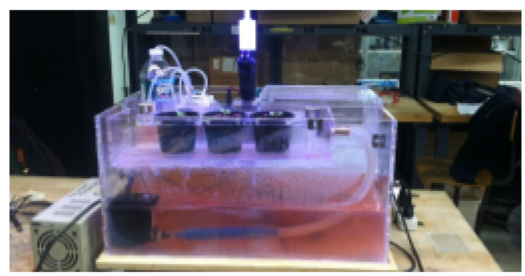


Figure 3-2: Hydropotomous

3.1.1.2 Grobo Hydroponic Grow Boxes

Grobo Hydroponic Grow Boxes is an expensive and large kit that contains a power unit for the product Grobo Solid. This item comes with a carbon filter, packs of five nutrient bottles and a Coco pod. To adjust the pH level of the water this system uses 5 nutrient bottles. On the higher end of retail hydroponics a product like this will be considerably more effective at maintaining the health of plants compared to ours. This system is a good idea of what parts to consider when making purchases.

Figure 3-3: Grobo Hydroponic Grow Boxes

There are also a plethora of recordings online of people constructing their very own Do-It-Yourself garden systems constructed by gardening enthusiasts, which we have drawn inspirations from and hope to build upon. Arduino and other electronics distributors also offer a multitude of measuring tools that could be potentially used for gardening applications.

3.1.2 Mobile Apps

In designing our app there were similar hydroponic apps in place that we researched that we intended to implement beyond the normal sensor reading we are implementing. We invested time researching these popular hydroponics mobile apps to get an idea of design and functionalities that go well with our project. Our focus is designing something that is simple, engaging, and gives relevant information.

3.1.2.1 GroLog

The app GroLog is an iOS application that has the ability to monitor the information that you want for several different garden systems. The app has functions like taking notes of things like room conditions, nutrient solutions, your lighting settings, and even taking photos and notes to track the system's progress. We intend to initially allow the user to only read the sensor data and allow the system to be automatic, but these are features that are doable. This app also has the ability to set up a schedule for the hydroponics system that could be implemented.

3.1.2.2 MQ Greenthumb

MQ Greenthumb is neat in that your iPhone or iPad can turn into a light meter to determine if your plants are getting the correct amount of light. We won't need this feature for our project because the light sensor in the hydroponics system will already relay the information. Something we could implement however, is that MQ Greenthumb also offers growing tips, resources, and information on common houseplants. This could assist in the engagement of users.

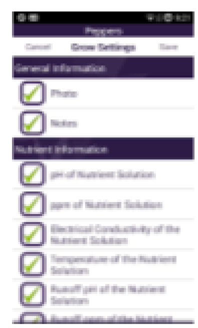


Figure 3-4: MQ Greenthumb

3.1.2.3 Plant Doctor

The Plant Doctor One deals with maintaining plant health against diseases, pets, or other plant illnesses. This application allows a user to diagnose the issues with the plant the sensors for the hydroponics system can't sense. This most likely won't be implemented into our project, but it does bring up other factors affecting plant health not considered before in other aspects of our project.

Figure 3-5: Plant Doctor

3.1.2.4 When To Plant

The When To Plant mobile application is geared towards outdoor farmers because it takes user zip code data and returns weather data that is specific towards farming and what plants can change like frost.

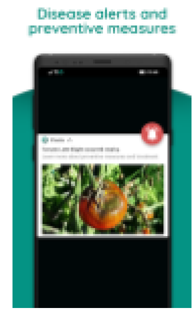


Figure 3-6: When To Plant

Overall, we found these apps to be good goals to move towards when designing our app and what would be beneficial for a hydroponics system application. The challenge will be having the wireless module on the microcontroller communicate data to our application and then allow the user to access all the functionalities.

3.1.3 API

Mentioned in a previous section about differentiating our project from other hydroponics with similar mobile apps, our group is planning to use a recipe search functionality to allow the user to go from planting to plate in the same system software. We are looking to create an application for chefs, nutritionists, farmers, and foodies to turn their vegetables into meals. There are also a ton of food APIs documented already for us to use.

3.1.3.1 FoodData

Figure 3-7: Food Data

Our group researched the [2]USDA FoodData Central API which allows software developers to utilize nutrition and food component data into their websites and applications. This API was interesting in that it gives access to five unique data types: Foundation Foods, Food and Nutrient Database for Dietary Studies 2013-2014, National Nutrient Database for Standard Reference Legacy Release, USDA Global Branded Food Products Database, and Experimental Foods. We saw its use but we were aiming for something recipe and cooking oriented.

3.1.3.2 Food2Fork

The [2]Food2Fork Recipe API was considered and is a recipe sharing website. The Food2Fork Recipe API is unique in that it provides JSON responses to access recipes based on

Figure 3-8: Food2Fork

a large recipe database, with powerful ingredient search Function, and an interesting social-media based ranking algorithm. However, we did not need a social media aspect so we moved on.

3.1.3.3 BigOven

The [2]BigOven API was a competitive choice because it had lots of recipes, but it also is a social network to some degree. Making shopping lists and planning menus is the social aspect that we didn't entirely need. However, the BigOven Recipe API does advertise to have 350,000+ recipes and with photos to match nearly all the recipes. There are a host of other features that are useful for other apps, but these features aren't needed and some are behind paywalls.

Figure 3-9: BigOven

3.1.3.4 Spoontacular

The API we found that fit best with what we wanted was simply called Food API. [2]The company Spoonacular's Food and Recipe API provides the most access to over 360,000 recipes, 80,000 food products, and more importantly includes information about over 2600 ingredients. This API also enables users to search for recipes using colloquial language like pasta with basil. The recipes generated also come with ingredient lists and pictures to give the user a glimpse at what our hydroponics farm can contribute towards.

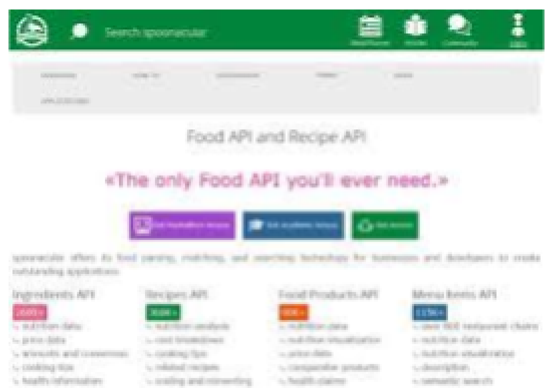


Figure 3-10: Spoontacular

3.1.4 Webstack

Inorder to create a hydroponics systems that will be able to communicate to a mobile app and website we need to have multiple software pieces work together to gather, store, and interact with data relating to the plant and user. The main ones our group deliberated based on experience from classes and personal use on were LAMP, MERN, MEVN, and MEAN. These web stacks are all being implemented in many web and mobile applications and can all do the task needed.

3.1.4.1 LAMP

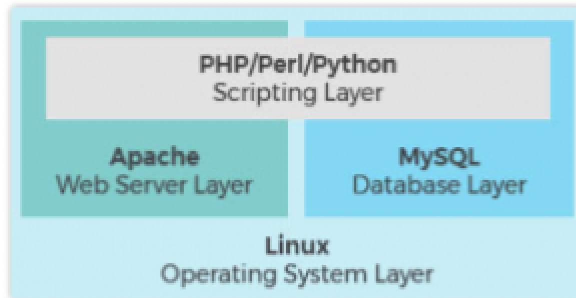


Figure 3-11: LAMP Stack

The LAMP stack is a webstack that comprises Linux, Apache, MySQL, and PHP. Linux is the operating system that sits as the lowest layer below the Apache web server, MySQL database, and the PHP programming language. The LAMP stack starts to process information when the user requests a website from the web browser a HTTP Apache web server responds. Data from the MySQL database fetches information through the PHP programming language. This stack is useful because this webstack allows the creation of a webpage and web server to present and store data for the user to use. Overall, LAMP is very simple, efficient, and flexible [21].

3.1.4.2 MERN, MEAN, MEVN

Figure 3-12: MERN, MEAN, MEVN

These three webstacks are very popular in today's software development world. The backend technologies are MongoDB, Express.js, and Node.js. The MongoDB is the database of the stack, Node.js is the runtime environment, and Express.js works with Node as a framework for designing web applications and APIs. The main difference between MERN, MEVN, and MEAN is the use of either React, Vue, or Angular for the front-end framework. The MEN web stacks are all JavaScript based programming language stacks that provide similar functionalities to each other and LAMP stack. These web stacks were developed by people working for or the companies Google and Facebook so they are extremely viable in developing web and mobile applications [22].

3.1.5 Web Server

From the webstack section we have an overview of what a webstack is and how they perform. We looked into two main types of web servers to help implement the website. We considered the setup, learning curve, and compatibility with other web stack components when conducting our research. The webstacks in the previous section had the Apache server and Node.js as web servers that were built with other components. The

Apache server has been around for a while and proven to be efficient, but the newer Node.js is also a competitive and comparable web server system.

3.1.5.1 Apache HTTP Server

Apache is a web server and not a physical server. This is a type of software that runs on a server with its job being to bridge a connection between a server and the website that visitors are trying to access [23]. Websites like Google, Firefox, Safari, etc. While establishing connections within a client-server structure, Apache and the servers try to deliver files back and forth between the users and website. Apache servers have been around since 1995 and are cross-platform software working on both Windows and Unix servers. Within the LAMP web stack Apache is used frequently with PHP. Apache is most known for and works with the Hypertext Transfer Protocol (HTTP). Configuring the Apache server is through configuration files where the behavior is controlled through modules to be able to run PHP. Also, configuration files require a hosting website and ports through which data and traffic is directed. This can be confusing and a lengthy process to understand and modify the configuration files correctly. The steps for setting up an Apache HTTP server with PHP requires [24], [25]:

1. Configure Internet Information Services (IIS)
2. Download the Apache HTTP server and PHP from their web pages
3. Configure Apache and store it on web page root level
4. Configure PHP as Apache Module, storing as an environment variable
5. Finally running a test with PHP on an Apache server

Setting up an Apache HTTP server with PHP may seem like only a few steps, but it requires an initial upfront setup compared to Node. PHP is used with Apache frequently because they were both developed around the same time during the 1990s so both software components are combined to power a large number of websites to this day. PHP as a language is diminishing in popularity compared to JavaScript and Python in the 21st century.

3.1.5.2 Node.js

Node.js is a big component of the MERN, MEAN, or MEVN web stacks. Node uses JavaScript as the programming language to create a

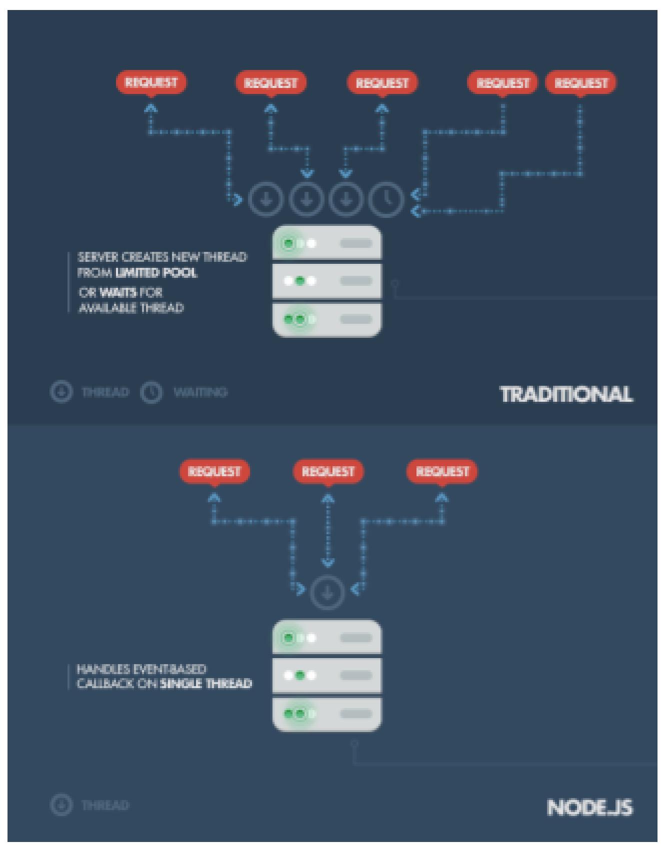


Figure 3-13: Node.js

web server with a web page in the same environment. Everything that Apache and PHP allows for web application functionality, Node can do in a unified language with real-time web applications. Node.js uses non-blocking, event-driven I/O to stay efficient with real-time applications with lots of data that run across different devices. It's main benefit for our project is that it can build fast and scalable network applications. Setting up Node.js only requires a user building a web server to download the Node.js software then they can start a project. Node being used within the MERN, MEAN, or MEVN web stacks allow the different components to communicate all within JavaScript. Node also comes with a wide variety of modules like mongo that allows for a database to be connected with to expand functionality. Computation and server-side processing is an issue with Node because it may be able to handle multiple concurrencies on a single thread, but cause problems for all users as incoming requests would be blocked until the computation was completed. This would be an advantage to our project because we need to read in and handle multiple readings from different devices in real-time quickly rather than computing numbers to produce an output [26].

3.1.6 Databases

As a significant part of the software stack, our group researched databases and what types would be compatible for our project. The main types of databases that were applicable were Relational and the converse Non-relational. There is no universal database, web server, webstack or anything for websites and mobile applications, but there is a best fit to deliver the best capabilities for our needs.

3.1.6.1 Relational Database

Relational databases are also known as SQL databases. SQL is a domain-specific language for querying and manipulating data in a relational database [27]. In the 1970s, the IBM researcher E.F. Codd the "relational model" of data management. This system was devised and popularized in a number of subsequent database systems starting with System R.

SQL abstracts data as a set of tuples organized into relations, which allows for abstraction over the physical representation of data and access paths. While SQL is not the only possible language, it is the most popular for implementing queries over the relational model. Example relational database systems that use SQL are: Oracle, Microsoft SQL Server, Ingres, and more [28]. Even though they all have unique modules and libraries, the standard commands such as "Select", "Insert", "Update", "Delete", "Create", and "Drop" can be used within a database to accomplish almost everything that is required.

3.1.6.2 Non-Relational Database

Non-relational is a type of database that does not use the typical database structures with rows and columns. This database employs a storage model for optimizing data with specific requirements of storage. Some examples of non-relational databases are JSON documents, columns, Key/value, and graph data stores. The application type influences the data differently depending on what data store is being used [29].

Big industry research papers like Google's BigTable and Amazon's Dynamo furthered the innovation with these types of databases to address problems within Non-relational Databases. Namely, the horizontal scalability being insufficient and inflexibility of the static table design in relational type systems. The tables 3-1 and 3-2 below outlines the different types and their storage techniques [30].

Document Storage Types	**Description**
Key-Value Stores	This NoSQL database is the simplest by utilizing a key-value store. The database stores all the data elements in a key value pair with characteristics of a “key” and a value. The key can be considered as the name of the data. Compared to a relational database, they are similar because only the key and value columns are used.
Graph Databases	Deriving from graph theory, the graph database maps the relationship between data elements within the database. The database consists of nodes and connections between the nodes called links. Usually, a graph database is optimized to capture and search the connections between data elements. This circumvents the issues with joining tables in SQL.

Table 3-1: Google Database Storage

Document Storage Types	**Description**
Wide Column Oriented Databases	A simple difference between a relational database store and column-oriented database is that a relational database reads data with rows, but a column database uses columns instead. Performing analytics on columns is efficient if your task is to only read data within columns and not waste data space reading rows.
Document Databases	Document databases put data into storage using BSON, XML, or JSON documents styles. Documents are useful because they can be nested within each other. When looking for specific data elements the documents can be indexed for faster querying. The objects for data used in application databases can be stored and requested in a form that is more relevant.

Table 3-2 Amazon Database Storage

Overall, these NoSQL or non-relational databases are great for data that doesn't have a strict structure. These database structures are dynamic and scalable when manipulating data from unique information environments.

3.2 Relevant Technologies

Indoor agriculture is by no means recent, but to this day it is gaining recognition for its sustainability and efficiency over conventional farming practices and the industry is projected to only rise as the demand for sustainable, quality food rises. With recent developments in UV LED technology farms can now simulate direct sunlight and even adjust the light spectrum of the LED to improve photosynthesis in crops.

Vertical farming has also revolutionized the land use of agriculture, allowing us to grow on multiple planes with a much higher yield while using space much more efficiently. While there have not been many advances in plant growth and farming, technology has pushed the limits of farming.

3.2.1 Horticultural Lighting

When considering a lighting system for a garden there a severals factors to consider

Figure 3-14: Horticulture Lighting

Cost - Each lighting solution will have its own upfront cost depending on the materials, durability, purpose, and efficiency.

Power consumption - A highly efficient lighting source may be more expensive upfront but are usually cheaper to operate in the long term. Since our device would be battery powered, we prefer the most power efficient in order to extend the system's battery life

Durability - If the longevity of your system is not a priority, it could be more practical to consider cheaper options. However for the sake of our project, we would like to minimize as much maintenance as possible.

Heat emission - While heat can be beneficial to plants, too much of it can have a negative impact on the crops and energy efficiency

PPF/PPFD - This is a reliable metric to measure photon density, rather than just brightness, and thus how effective a light source can be in the photosynthetic process. We need to be sure our light system can provide proper PPF levels to maintain optimal plant health

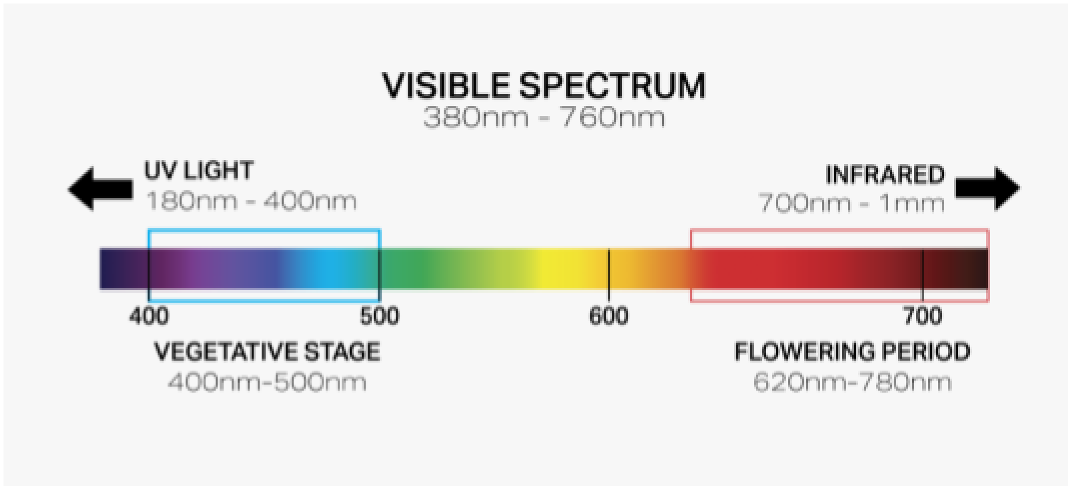


Figure 3-15: Photosynthesis Light Spectrum

Light spectrum - The spectrum of a light has a significant impact on the type of growth that occurs in plants. The optimal spectrum varied from crop to crop, depending on the desired yield.

Lighting types:

Incandescent - Most cost efficient but produces too much heat and must be supplemented with natural sunlight

Fluorescent - More effective than incandescent with a wider lighter spectrum, but not sufficient for all stages of plant growth, and produces less heat while also being budget friendly

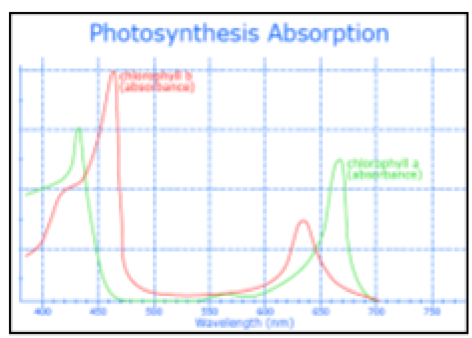


Figure 3-16: Photosynthesis Absorption

Discharge - Can achieve both sides of the full optimal for vegetatie growth and flowering but not simultaneously and must be used in tandem to achieve the full spectrum. While these bulbs are expensive, they are power efficient.

LED - Not only can it provide full spectrum control, but it also has a very small footprint, low heat rating, and little power requirement. LEDs are more expensive but more functional, and prices are steadily declining as technology develops. Which is why an LED grow light was our top choice.

3.2.1.1 Wavelength Band Spectrum Effects

Far-Red/Red light- Promotes flowering in plants, stem growth, and internodal spacing.

Green light - Due to its high transmittance, the green light can penetrate through the top layer and reach the lower branches. Improving overall health

Blue light - Promotes root development, thicker leaves, and shorter stems, but low levels of PPFD

Full spectrum (White) - Instead of focusing on a single section of the light spectrum, we can incorporate all three to create a well-balanced spectrum optimal for a broad variety of plant species.

3.2.1.2 PPF/ PPFD

When it comes to lighting there is more than meets the eye. There are many metrics which we measure light, however not all are created equal. In an agricultural setting, we have to consider more than lighting and color if we want to create the best lighting source for horticulture. We must distinguish between a few metrics before we can decide which light to use.

- Lumens
- PPF
- PPFD

Lumens is one of the most commonly used metrics. In short, it is used to describe

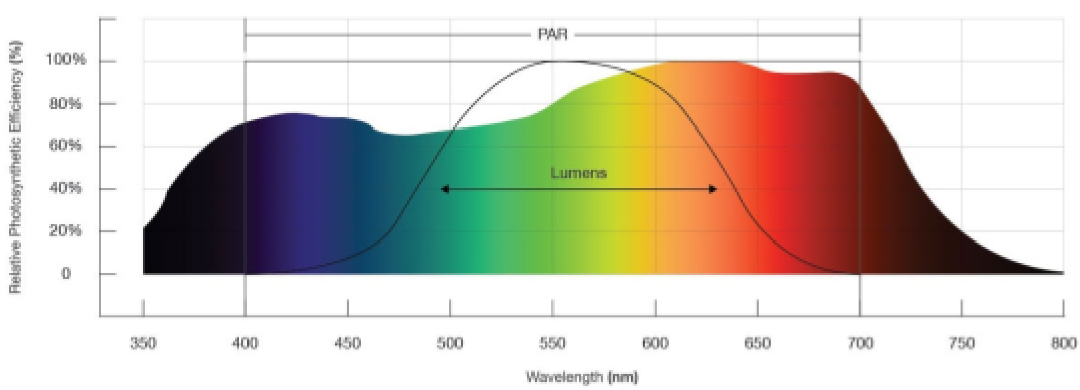


Figure 3-17: Photosynthesis Efficiency vs. Wavelength

brightness. However, Lumens is in the context of the human eye and is purely for human use. Since lumens only exist in the realm of visible light, it does not properly describe the effectiveness of a light in photosynthesis.

PPF, also known as PAR, stands for Photosynthetic Photon Flux. Plants perceive light differently than humans. PPF dwells within the spectrum of 400nm to 700nm, because it is the photons in this region which contribute to photosynthesis. Instead of the appearance of light, PPF values the abundance of photons emitted by a light source. Hence, PPF is expressed in micromoles per second (umols/s). It is also important to note that PPF does not consider direction.

PPFD, Photosynthetic Photon Flux Density, takes PPF a step further. While photon flux flux indicates the total photon flux, photosynthetic photon flux density specifies the photon flux per unit area. This makes PPFD more useful, as it describes the volume of photons which will reach the garden plot.

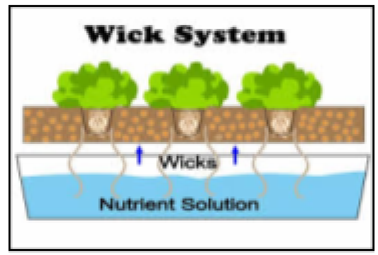


Figure 3-18: Wick System

3.2.2 Irrigation

What defines Hydroponic farming is the absence of soil and its use of aqueous solutions to deliver nutrients to crops instead. That being said, water is the most crucial part of any hydroponic system. While every hydroponic system has the same use for water, there are many different ways to manage the supply and flow of water.

This system involves using wicks to passively supply liquid to the plant roots, making this the simplest form of hydroponics, while this system is easy to assemble and does not require mechanical devices to transport the water, the roots can only absorb water as fast as the wicks can. This constrained water flow may be insufficient for plants which require ample water.

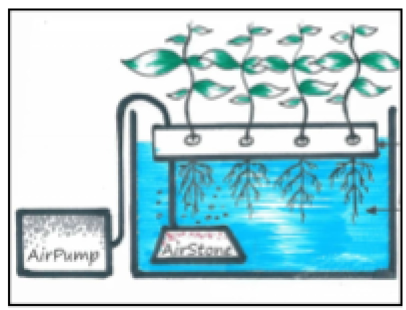


Figure 3-19: Air Pump/Stone Design

Using the water culture method, the plant roots will be entirely submerged in water, however, in most crops, roots need air exposure to absorb oxygen, so this method also requires an active air pump to supply the roots with a continuous stream of air bubbles. While this system is suitable for leafy greens and water-intensive crops, it is not suitable for long-term crops. Most plants are not compatible with this form of hydroponics. Also having to constantly run an air pump can draw a lot of power.

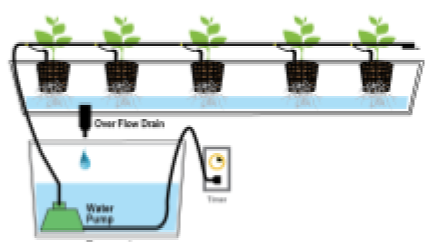


Figure 3-20: Overflow Drain

This system involves an irrigation system which runs on a timer. The plants will not be submerged but be placed in a grow medium. When this system is activated it will draw water from the nutrient solution reservoir and drip this solution on the base of the plants, directly watering the roots of the crop. Some drip systems can also incorporate drain back into the reservoir to collect excess water, this will save water and nutrients but may cause shifts in pH and nutrient levels.

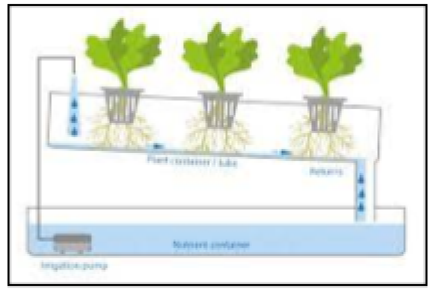


Figure 3-21: No Growth Medium

In this form of hydroponics, the plants will not have a growth medium but the roots will sit in a constant flow of liquid solution. The solution then flows back into the reservoir. The Nutrient film technique is great for leafy greens but has a few drawbacks. Having to constantly pump water makes this system vulnerable to power outages and since there is no growth medium plants can quickly dry out, also this configuration is better suited to plants with shorter roots.

Aeroponics is similar to the drip system but there is no growth medium, instead the plants will hang above the reservoir and the liquid will be directly sprayed onto the root systems. To prevent the roots from drying out, the hanging roots must be frequently and thoroughly sprayed. While aeroponics requires less space and less susceptibility to disease, Hydroponics is more cost effective, efficient with water usage and overall more sensible for small scale applications.

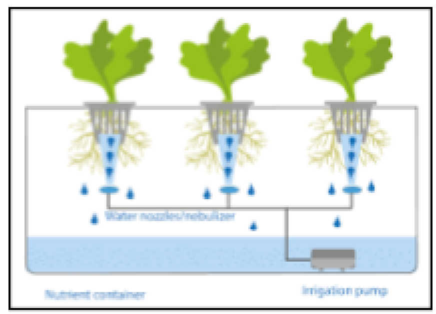


Figure 3-22: Aeroponics

This system utilizes a combination of passive and active watering. Simply put, the plants will sit in a tank separate from the reservoir. A pump in the reservoir will deliver water to the plants and fill up the tank. The pump will run on a timer and a drain to also ensure the plant tank does not get overfilled, this drain will lead back to the reservoir. The grow solution will be refilled and cycled periodically to maintain proper moisture and nutrition levels.

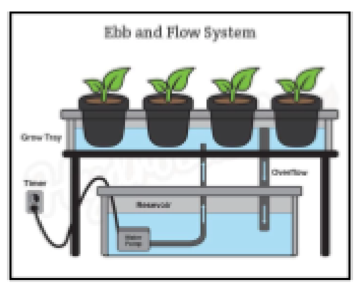


Figure 3-23: Ebb and Flow System

3.2.3 Solar Power

Solar power could be an interesting part of our project. Adding solar power would allow us to power our hydroponics system with ambient light as well as store excess power in the system. Solar power has been used for a long period of time. Solar power dates back as far as 7th Century B.C. when humans used sunlight to light fires with magnifying glass materials. When the bi-junction transistor was created, solar energy we know today was formed. Solar energy uses semiconductors to generate electricity. When light strikes the semiconductors electrons flow through the semiconductor causing what's called the photoelectric effect. If the hydroponic module is mounted outside then there is a possibility for solar panels to
be mounted. This is most certainly a stretch goal and not a necessity.

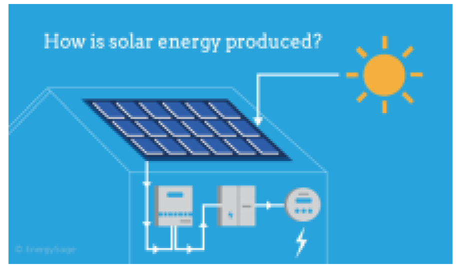


Figure 3-24: Solar Energy

3.2.4 Extraterrestrial Hydroponics

One of the unique applications for hydroponics is in environments where soil is not relatively abundant and space is limited. The low gravity environments of outer space are useful for moving equipment with little effort, but transporting nutrients to plants that are suspended is difficult because the nutrients and plant roots don't make direct contact. This challenge is being researched by NASA to enable astronauts to live for long periods of time independent of Earth resupply. The soil found on semi-inhabitable planets is lacking the necessary nutrients for plants to take root and grow. From the Verge website, *"Mastering plant growth in space and on other worlds will be important to future crews traveling on long-duration missions off our planet."* Our project isn't intended for space, but the application for space this project is the basis for demonstrating that small technologies can be built upon to achieve greater projects and missions [31].

Figure 3-25: Extraterrestrial Hydroponics

3.3 Strategic Components and Part Selections

This section will summarize our rationale behind our design and the methodology in selecting which parts to use. When selecting parts and functions to include in our hydroponic system our team had to consider cost, feasibility, size, weight, power consumption, and last but not least, efficacy to improve plant growth in the system. This system will deliver all the essential components of plant growth but as the project develops we wish to include as many features as practically possible to ensure the system is as seamless, user-friendly, and autonomous as possible. However, before we consider any other features, we must first consider the two fundamentals of gardening, lighting and watering.

In this section, we will provide some insight in the part selection process. The parts may be subject to change. To decide which sensors would work best for our design we must consider the following factors:

- Size
- Cost
- Reliability
- Consistency
- Accuracy
- Durability

3.3.0 Grow Light

Since they are the cheapest, most efficient, and flexible in terms of design we decided on LED grow lights. We had several options to consider which were mostly based on size, shape, and power usage and finalized our selection to three choices. Although buying the individual LED surface mounted device would be the cheapest option up-front, the cost and time of self-soldering the components ourselves would not be a viable option. So the options we choose come pre-mounted on pcb or strip.

3.3.1 Main Water Pump

One of the benefits of a hydroponic system is the simple method of irrigation. Since the vegetation will sit above a tank of water, we just need to replace the water less frequently without having to worry about a sprinkler system. This simpler form of irrigation requires less piping and less water pressure, and could easily be driven by a smaller pump. A basic 12V DC water pump with a flow rate of 1-2 gallons per minute should have sufficient water pressure to refill the hydroponic tank.

3.3.2 Peristaltic Water Pump

One of the main features of this automated hydroponic system will be the automatic balancing of the liquid ph levels. If the pH level of the hydroponic system is not at the optimum level, one of the two peristaltic pumps will be activated. Attached to the main reservoir will be bottles of certain ph concentration solutions, The peristaltic pump will dispense one of the two solutions into the main reservoir, balancing the Ph of the water to the desired level, then the mixture will be transferred to the Garden Pod. Peristaltic pumps are fairly simple devices which are inexpensive and widely available online. The Gikfun 12 volts peristaltic dosing pump is a suitable choice for this part. This device has a flow rate from 0 to 100 ml/min and could be used in environments with up to 80% relative humidity.

3.3.3 Liquid pH-sensor

To properly balance the ph levels of the hydroponic system, this project will require a ph probe suitable for long term monitoring and capable of being permanently submerged in water. The sensors must also be cost effective, durable, and reliable. Which is why we selected the Gravity: Analog pH Sensor / Meter Pro Kit For Arduino. The pH sensor kit includes an industry pH probe, a BNC connector, and a pH sensor interface. This probe is the optimal balance between performance and cost, and should provide reliable results.

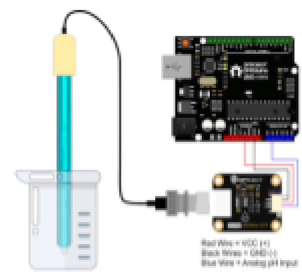


Figure 3-26: Arduino pH Sensor

3.3.5 Water Level Sensor

One of the purposes of this project is to make indoor gardening as easy and autonomous as ever. The goal is to keep maintenance to a minimum, so to ease the effort of remembering to add water to the system, we decided to include water level sensors to detect the amount of water in the reservoir tank and garden tank. That way, the garden can be watered automatically and the user will be reminded when the main tank needs to be filled. Water level sensors come in a wide range of sizes and can be fairly expensive but for the sake of this project, we do not need a high performance device. A fairly inexpensive model should suffice. We have a choice between non-contact and contact, if the non-contact has similar or greater range we will choose this option.

3.3.6 Temperature/ Humidity Sensor

Humidity/ temperature sensors can measure the moisture and temperature in the air by detecting changes in electrical currents. There are several types of Humidity/ temp sensors

- Resistive
- Capacitive
- Thermal

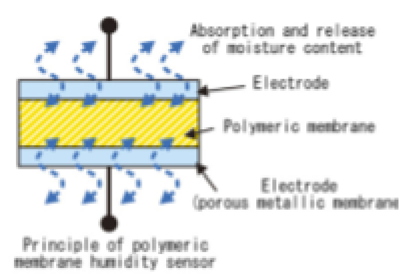


Figure 3-27: Temperature/Humidity Sensor (Capa

A capacitive sensor is composed of a thin strip of metal oxide placed between two electrodes, with this configuration the sensor can measure the humidity in the atmosphere by detecting changes in capacitance, alternatively some sensors use changes in the resonance frequency. The output of these sensors is linear and fairly stable, even over long uses. These sensors could also be used in a wide range of environments. The only drawback is the limited range.

Resistive sensors, also known as conductivity sensors, measure humidity through the conductivity between two electrodes. The two electrodes are coated with a layer of some non-metallic conductive medium. As the medium absorbs moisture in the air, the conductivity will change.

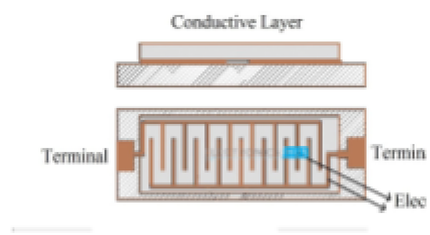


Figure 3-28: Temperature/Humidity Sensor (Res

Resistive sensors are typically low-cost and have a very small footprint, but this type of device is sensitive to chemical vapors.

Thermal humidity sensors, or Absolute humidity sensors, will measure the thermal conductivity of both dry air and the surrounding air, the difference between these two will result in absolute humidity. This process is accomplished by two thermistors, one thermistor is sealed in an airtight compartment with dry nitrogen and the other is in a compartment exposed to the environment through small vent holes.

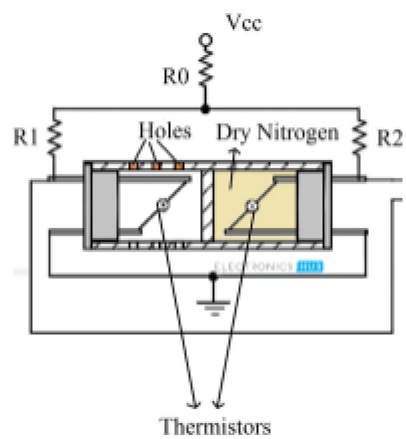


Figure 3-29: Thermal Humidity Sens

This model of humidity/temperature sensor is very durable and could withstand very hot and corrosive environments, while also dealing with a high degree of accuracy.

3.3.8 Wireless communication

One of the main concepts of our design is that the units must be modular and able to function cohesively while separated. The consumer should be able to move each unit freely as well as control and manage these devices wireless. That being said, we will need to implement a wireless communication system. In most electronics, Wifi and bluetooth compliment each other, however, wifi may not be necessary for this project. Which method of communication we decide to include depends on which type of experience we would like to offer the consumer. Bluetooth and wifi are both very useful forms of communication but each come with their own benefits and drawbacks, this section will explore factors and help us to arrive at a solution.

3.3.9.1 Bluetooth module

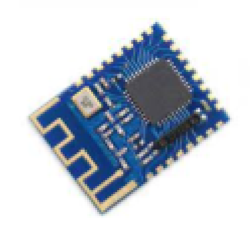


Figure 3-30: Bluetooth Module

Bluetooth is the short-ranged form of wireless communication, which transfers data at a frequency of 2.4GHz. Wireless data protocol can achieve ranges of 5 - 100 meters, depending on the class of bluetooth module. This would be ideal for a home setting and would allow the user to monitor and manage the system in close proximity, through a smart phone or bluetooth compatible device. Bluetooth also requires little amounts of power which makes it suitable for battery powered devices and is fairly inexpensive. **Table 3.3.9** outlines some of the different bluetooth modules compatible with Arduino that we could implement.

3.3.9.2 Wifi Module

Figure 3-31: WiFi Module

Incorporating a wifi module would add a lot of functionality to the system and improve overall flexibility. Some wireless modules even include bluetooth functionality. Wifi would also allow the user to access and monitor the system virtually anywhere, as long as there is internet access, but we also have to consider its power usage. The following devices are WiFi modules that we investigated for our hydroponic system.

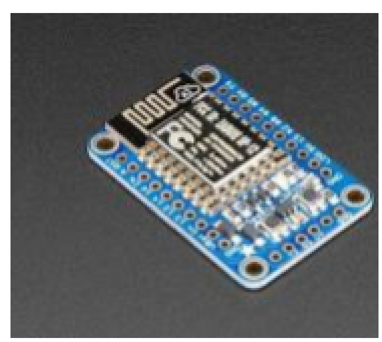


Figure 3-32: ESP8266

ESP8266 - this WiFi module is arguably the most popular on the market today. Because of it’s on board processing and storage, communication with other sensors is simple to configure. It has low power consumption and has a very competitive price of just $10. Using this module with the Adafruit HUZZAH board will allow us to have enough pins to integrate it into our Arduino. This board also has a built in reset button which could prove helpful for product testing/troubleshooting.

Additional Specifications:

- 2x 3-6V input
- 2x UART pins
- 64KB RAM

ESP32 Development Board - Coming in at a similar price point ($10), the ESP32 also adds a Bluetooth 4.2 version capability. It supports 802.11b/g/n/e/i and can do up to 150Mbps. With the added features of this module, we will incur more challenging implementation, but it does have an attractive set of capabilities.

Figure 3-33: ESP32 Development Board

In order for our system to function as originally intended, our best course of action is to implement both bluetooth and wireless modules.

3.3.10 MOSFET Voltage regulators

This project will be powered by large AC and DC sources and controlled by a microcontroller which cannot effectively channel the rated power to the electronics. In order to supply the electronics that require higher voltages, with their proper voltage and manipulate them with the microcontroller, we will use MOSFETs. MOSFETs are Metal-oxide semiconductor field effect transistors. MOSFETs have an insulated gate which uses voltage to control the conductivity of the device. By manipulating the applied gate voltage, we can use this shift in conductivity to amplify and switch electrical signals. MOSFETs are the most widely used transistor in digital circuits and are especially useful in amplification, since they do not require input current to be controlled.

A MOSFET device has four terminals which are source, gate, drain and body. The body terminal is typically tied to the gate terminal, so only 3 terminals are used. The flow of charge carriers enters through the source terminal and exits through the drain terminal, the width of the channel depends on the voltage applied at the gate terminal.

Roughly speaking, through are four kinds of MOSFETs:

- Depletion-type N-channel
- Depletion-type P-channel
- Enhancement-type N-channel
- Enhancement-type P-channel

For Depletion-type MOSFETS, The conductance of the transistor is at its max when the voltage between gate and source terminals is zero. If a negative voltage is applied to the gate the channel conductivity will decrease as the magnitude of the gate-source voltage increases, meaning the channel is normally wide open.

Conversely, in the Enhancement-type, The channel is closed until voltage is applied to the gate. The channel conductance increases with the gate-source voltage, meaning the channel is normally closed.

MOSFETs can also be classified as either P-channel or N-channel, depending on the composition of the transistor's substrate.

P-channel MOSFETs have heavily doped p+ regions at the source and drain terminal with an n-type substrate in between. If a negative voltage is applied to the gate, the electrons from the metal-oxide layer are propelled into the substrate and electron-holes are attracted to the channel, widening the channel. In contrast, positive voltage would narrow the channel for a depletion-type p-channel mosfet.

N-channel MOSFETs are heavily n-doped at the source and drain and the body is composed of a p-type substrate. When a positive voltage is applied to the gate, the metal oxide layer will attract electrons from the substrate and repel holes. Widening the channel of free electrons between the n doped regions of the source and drain. Although p-channel and n-channel MOSFETS sound very similar the electron mobility is 2.5 times faster than hole mobility. This gives N-channel MOSFETs faster responsiveness and P-channel higher drain resistance.

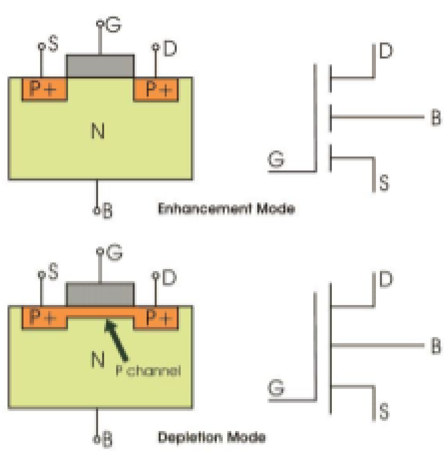


Figure 3-34: N-Channel MOSFETs

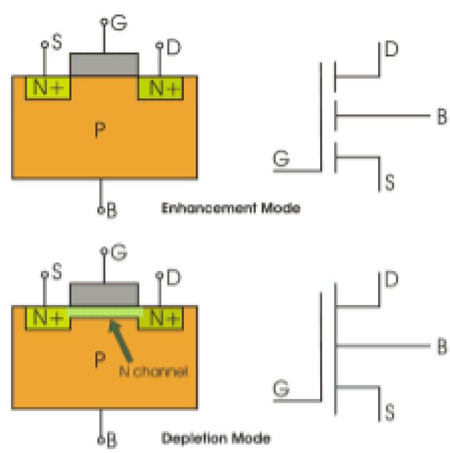


Figure 3-35: P-Channel MOSFETs

Which MOSFET we choose will depend of a few conditions, such as

- Which polarity would best suit of application
- Whether we would like our switch to be Normally open or Normally close
- The maximum rating of operating voltage and current values

Depending on the voltage levels of the microcontroller pins, a MOSFET driver may be needed to properly activate the MOSFETs.

3.3.11 MOSFET Drivers

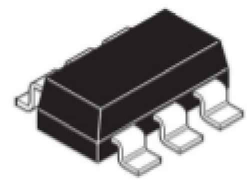


Figure 3-36: MOSFET Driver

MOSFET transistors will be used to operate some of the electronic devices of this project. We plan on operating the main irrigation pump, dosing peristaltic pumps, and ultraviolet horticulture LEDs. These devices should be able to be controlled by the logic level voltages of the microcontroller unit, i.e. the speed and pressure of the water pumps and the dimming of the UV LEDs, using pulse width modulations (PWM). Simultaneously, each of these devices will be powered by an external power supply. Whether or not we implement a MOSFET driver depends on a couple of factors.

- If the logic level voltage is enough to drive gate voltage
- If the drive current is high enough to trigger a fast response in the mosfet

We will explore this subject more in testing, but to put it simply the LEDs require a high voltage supply, which may need a MOSFET with higher capacity. Logic level voltage might not be high enough to properly operate a bigger MOSFET. In this case, we would need a MOSFET driver to boost gate voltage.

Another advantage of a MOSFET driver is the decrease in switching time, meaning the MOSFET will spend much less time in the saturation region rather than the linear or ohmic region, minimizing power dissipation across the transistor. The LEDs will draw a large portion of power from our system so the components must be as efficient as possible in terms of power usage. We will be considering this in the later testing section.

3.3.12 AC to DC Converter

AC stands for “Alternating Current” and DC stands for “Direct Current”. AC is the main method of transporting power over long distances and is very convenient for converting voltage levels through transformers,which is why all power outlets are an AC source. The central unit of design will be fully powered by AC power via electrical socket. However, most electronics, including the ones in the project, are operated by DC power, because of this the AC to DC converter is the most vital part of this design. The function of an AC to DC power converter is to transfer AC power to DC by storing power with reactive components, rectifying the sinusoidal signal into a constant flow of voltage and current, and supplying stable power to a load, i.e. our hydroponic system. There are a couple ways to implement AC to DC converters.

- Flyback converters
- Buck-Boost Converters

Flyback converters and buck-boost converters can both be used to step-up and step-down voltages but the flyback converter utilizes a transformer, which isolates the output power. Converters could also come in different modes.

- Constant voltage
- Constant current

With constant voltage AC to DC converters, the circuit detects the output voltage and regulates the switching ratio to the desired voltage. Constant voltage is the simpler method. Constant current uses a current sensing loop by measuring both the output voltage and the inductor current. This allows the converter to quickly respond to changes in load.

3.3.13 TDS Sensor

Since the system will also be providing nutrients directly to the water supply, we will also need to include a TDS sensor to detect the total dissolved solids in the plant feed solution. A TDS sensor does so by measuring the electrical conductivity in a solution caused by the dissolved ionized solids in the water. The measurements from this sensor will indicate the potency of the nutrients in the water, the system can then steadily dispense nutrients in the water to adequate levels without over-saturating the solution. A lower end should suffice for this small-scale hydroponic system.

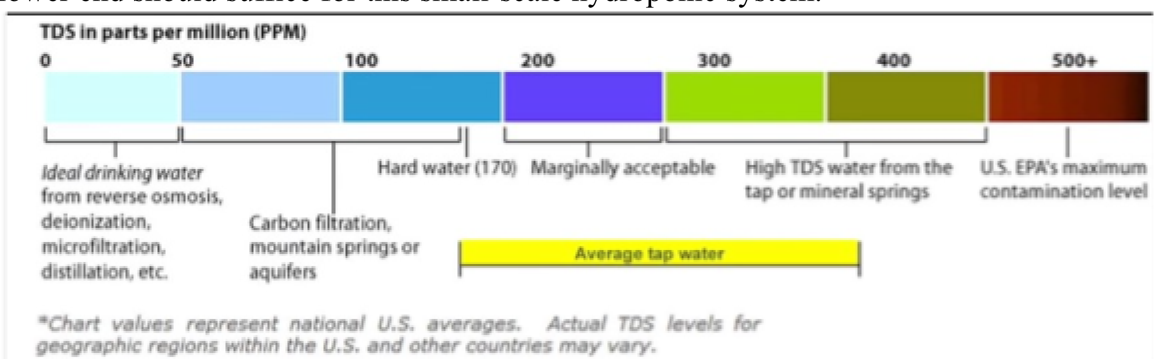


Figure 3-37: TDS In Parts Per Million(PPM)

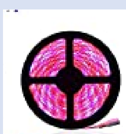
	Brand	Length	Forward voltage/ Current	Light Spectrum	Price
	Samsung Horticulture-L2 SL-B8R5C9H1A WW	281mm	21.5V/1.2A	Full Spectrum-White: 470nm-1000nm	$24.12
	Thomas Research Products - HORTICULTURE LED	300.61mm	28V/1.2A	RED: 660nm BLUE: 460nm	$9.31
	Plant Grow Light LED Phyto Lamps	5m (Clippable)	12V/1A	Red 625-660nm; Blue: 450-465nm	$19.99

Table 3-3: Grow Light Investigation

	Brand	Self priming (Y/N)	GPM	PSI	Power consumption	Price
	bayite 12V DC Fresh Water Pressure Diaphragm Pump	Yes	1.0 GPM	80	36 Watts	$24.99
	OMMO 12V DC Fresh Water Pump, Diaphragm Pump	Yes	1.2 GPM	85	60 Watts	$19.95
	DC 12V Mini Submersible Water Pump 63 Gal Pump	No, but submersible	1.0 GPM	63	4.8 Watts	$10.99

Table 3-4: Irrigation Pump Investigation

	Brand	Size	Flow rate	Voltage	Current	Price
	Gikfun 12V DC Dosing Pump Peristaltic Dosing Head with Connector	Driver size: 27.6×37.9 mm Pump head size: 31.7 ×20.1 mm	0-100ml/min	12V	80mA	$24.99
	Gikfun 12V Adjustable Peristaltic Dosing Pump Liquid Metering Pump	100 ×68×50 mm	8-70 ml/min	12V	2A	$28.98
	12V High Flow Peristaltic Liquid Pump Vacuum Pump	100×100×50 mm	0-400ml/min	12V	0.5-1.4A	$44.99

Table 3-5: Peristaltic Dosing Pump Investigation

	Brand	Size	Response time	Accuracy	Power	Price
	ICQUANZX PH0-14 Value Detect Sensor Module + PH Electrode Probe BNC Compatible for Arduino	Module Size: 42mm×32mm×20mm	Response time:≤5S	± 0.25pH	≤0.5W	$44.99
	Gravity: Analog pH Sensor / Meter Pro Kit For Arduino	Module Size : 43mmx32mm	Response Time: ≤ 1min	± 0.1pH	≤0.15 W	$56.90

Table 3-6: pH Sensor Investigation

	Item	Size	Features	Power	Price
	UV-C Algae Bloom Clean Light for Aquarium Water	6.9 x 0.98 x 0.98 inches	▪ Submersible ▪ Shatter proof glass	5W	$15.98
	Submersible UV Light Sterilization Lamp	10 x 0.75 x 0.75 inches	▪ Submersible ▪ Suction cups for adjusting position freely ▪ Quartz glass tube for high transmission of ultraviolet ray	7W	$16.99

Table 3-7: UV Sterilization Lamp Investigation

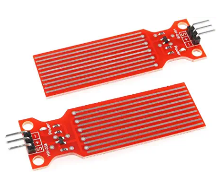
	Brand/Item	Type	Detection area	Input Voltage	Input Current	Price
	CQRobot Ocean: Water/Liquid Level Sensor	Non-contact	31.6×30mm	5V	5mA	$21.99
	DGZZI Water Level Sensor Module	Contact	40x16mm	3-5V	< 20mA	$4.99

Table 3-8 Water Level Sensor Investigation

	Item #	Type	Detectable wavelengths	Voltage - Collector Emitter Breakdown (Max)	Power - Max	Price
	BPW85B	Photo-Transistor	450~ 1080nm	35 V	200 mW	$0.61
	VEMT2500X01	Photo-Transistor	470~ 1090nm	20 V	100 mW	$0.74
	1528-2141-ND	Photo-Resistor	-	-	-	$0.95

Table 3-9: Photo Sensor Investigation

Model	BT Protocol	Range	Power Supply	Transmission	Price
BLE Link Bee	4.0	60m (197ft)	3.3V (DC) 50mA option for 5V	1 Mbps (Asynchronous)	$10
HC-06	2.0	9m (30ft)	3.3V (DC) 50mA	2.1Mbps (Asynchronous)	$9
JY-MCU	2.0	9m(30ft)	3.6-6V (DC)	3Mbps	$6
BlueSMiRF	2.0	100m(328ft)	3.3-6V	3Mbps	$27
Adafruit Bluefruit LE	2.0	10m(32ft)	3.3V	2Mbps	$17
Adafruit Feather	2.0	10m(32ft)	3.3V	2Mbps	$30

Table 3-10 Bluetooth Module Investigation

	Brand	**Size**	**On board/ Off board**	**Current**	**Price**
	ALITOVE AC to DC Converter 12V 2A 24W Power Supply	Module Size: 5.2 x 2.56 x 2.44 inches	Off board	2A	$8.99
	ECE40US12-SD AC/DC CONVERTER 12V 40W	Module Size : 3.78 x 1.57 x 1.12 inches	On board	3.33A	$75.45

Table 3-11: AC to DC Convertor Investigation

	Brand	**Size**	**PPM Range**	**Accuracy**	**Price**
	Gravity: Analog TDS Sensor for Arduino	Module Size: 42mm×32mm×20 mm	0 ~ 1000 PPM	± 10% (25°C)	$16.80
	CQRobot Ocean: TDS (Total Dissolved Solids) Meter	Module Size : 43mm×32.2mm	0 ~ 1000 PPM	± 10% (25°C)	$13.99

Table 3-12: TDS Sensor Investigation

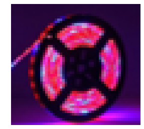
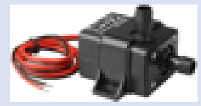
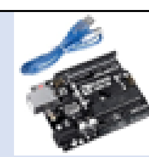
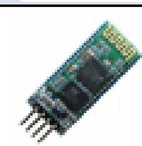
Device	Name	Description/Function	Image
Liquid PH sensor + PH sensing module	Gravity: Analog pH Sensor / Meter Pro Kit For Arduino	Measure the PH levels of the water dispensed into the hydroponic tank during water exchanges	
UV LED Array	Phyto Lamps Full Spectrum UV Lamp	Supply the Plants with UV Radiation	
Main water pump	DC 12V Mini Submersible Water Pump 63 Gal Pump for Aquarium Fish Tank Hydroponic Fountains	Pump fresh, PH-balanced water in the hydroponic tank	
Peristaltic pump	12V DC Peristaltic Dosing Pump	Pump PH concentrations into the water	
TDS Sensor	CQRobot Ocean: TDS (Total Dissolved Solids) Meter Sensor	Measure density of nutrient content	
Water level Sensor	Gravity: Non-contact Digital Water / Liquid Level Sensor For Arduino	Detect water levels in hydroponic tank	
Temperature/ Humidity sensor	DHT11	Monitor Temperature and humidity levels	
Microcontroller	Raspberry Pi	Process data and control Central Unit	
Microcontroller	ATmega328P	Process data and control Garden Pod	
Bluetooth module	HC-06	Allow for wireless communications between units	

Table 3-13: Initial Parts Selection

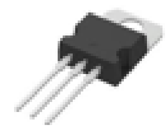
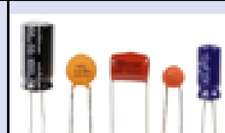
Device	Name	Description/Function	Image
Wireless module	TBD- MakerFocus 4pcs ESP8266 ESP-01 Serial Wireless WiFi Transceiver Receiver Module	Transfer data and control the device via application	
Logic-Level Mosfet	IRF520	Allow for the pumps to be supplied by a higher voltage source, and be controlled by the MCU simultaneously	
Air flow fan	Sunon fan ME40101VX-000U-A99	To provide air flow and dissipate heat	
Various Resistors	Surface-mounted resistors	Circuit design/ Power dissipation	
Various Capacitors	-	Circuit design/ decoupling/ stabilizing	

Table 3-14: Initial Parts Selection (2)

3.4 Possible Architectures and Related Diagrams

Our ultimate goal for the hydroponic system is modularity. This means that each pod system must be designed with two things in mind. First, the system should be designed in a way that all consumers can configure the system to best suit the layout of their home. This feature would allow the system to be placed in a room or location that is most suitable for the hydroponic system. The second design focus is ease of use. Because each pod will need to be recharged, emptied, and refilled, the pod and central unit need to be designed such that these actions can be done efficiently and easily.

The central unit design will be a box large enough to house the main water reservoir, nutrients reservoir, the central electronic components, the main pump, and power distribution components.
Unlike the pods, this unit will not be 3D printed because of the cost of printing something that large. The central unit will be housed in a waterproof container to ensure any internal leaks would be contained in the system as a safety measure.

A storage container will be the best to house our central unit for many reasons. It's low cost means that we can modify it to make sure that it fits all its components effectively. Keeping the cost of this low will also allow us to use the money elsewhere. Plastic storage containers are readily available in numerous stores, which means we will not have to wait for it to be delivered and we will be able to shop for the best possible container.

The central unit will need to be partitioned into at least 4 sections. Those sections will be the main water reservoir, the nutrient bottles, the pumps, and the electronics/sensors. All sections will need to be waterproof and isolated for safety reasons. There are multiple ways to divide these sections, the simplest method would be to use smaller containers inside the larger one and place them in their own respective location. Another method would be to create waterproof dividers in the form of walls. The wall design can be seen in **Figure 3-38.** The blue section denotes where the main water reservoir will be held. The grey section will hold the nutrient bottles, the green section is where the pumps will be located, and the red section is partitioned for all electronics. Orienting the central unit like this will ensure that each section is serviceable without having to alter the position/function of the other sections.

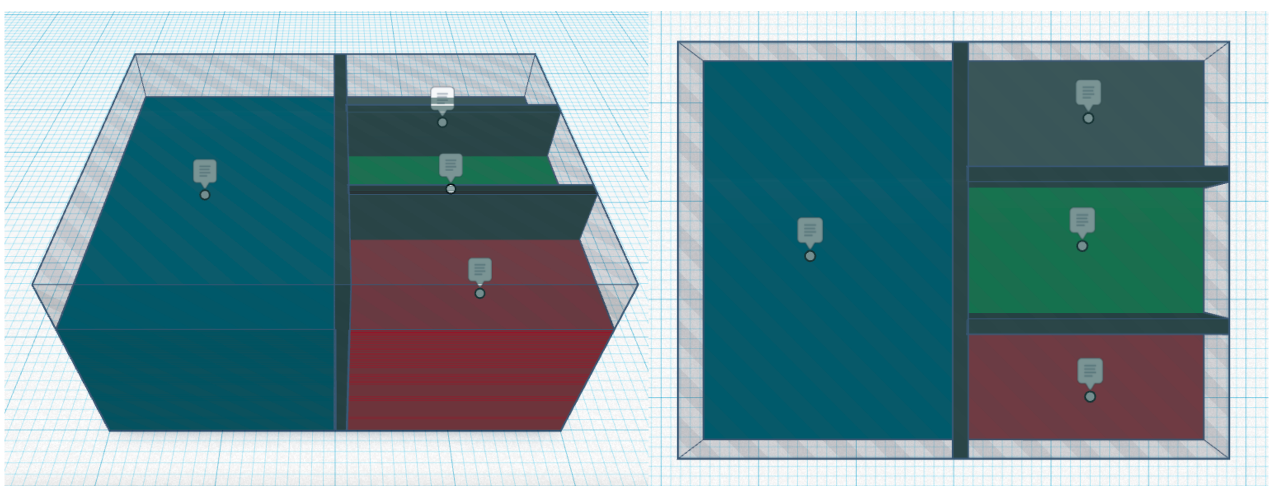


Figure 3-38: Central Unit Layout

There were numerous pod designs proposed by the team during initial meetings. With the design focused in mind, we landed on two to choose from. The first design was a central unit utilizing a spine to connect to each pod. In this design, the power supply would be centralized in the central unit and the pods would not have their own power source. All other communications between the central unit and pod would be handled via wire harness running through the spine.

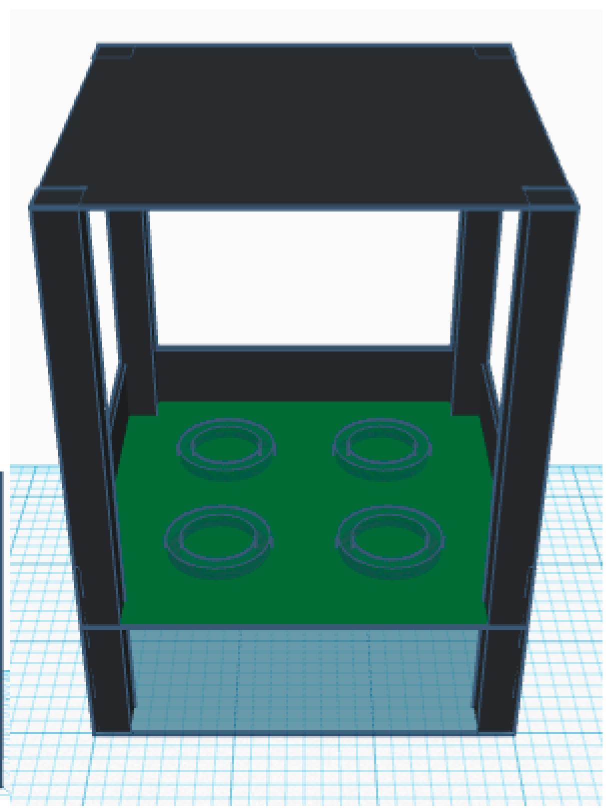
The second design came in the form of cubed pods that would stack on each other. This design was proposed in interest of scalability and modularity. The first page of the project document shows a hydroponic system made entirely of cubes which is where some design inspiration came from. In this design, the cubes could easily be stacked based on how many different pods the consumer would like; however, this design had nothing in place for connection to the central unit.

Further investigation of the pod design resulted in a design that implements both of the methods above. We will be combining both the spine design and the cube design to ensure that all of our initial objectives for the hydroponic system are accomplished. The spine creates a simple way to handle replenishing pod water and will also be used to charge the now battery powered pod. The pod itself will be modeled in the form of a cube for its stackable characteristics and ease of design. All other sensor equipment will be specific to each pod and will communicate wirelessly to the central unit. The consumer will only have to move the pod back to the central unit whenever it is time to be serviced.

A major point of investigation was the method for draining the existing water in the pod, and whether or not the water would be constantly moving into and out of each pod. After deciding that the water would be changed on a schedule, we wanted to implement a Pythagorean drain in the pod so that all of the water would exit the pod when filled to a certain level. This concept proved too complicated to implement and was abandoned. The focus was shifted towards a standard pump in/pump out system. The following images reflect those to ports.

Front of Pod Design

- A maximum of 4 plants will be placed into the pod. That location is depicted in the circular receptacles shown below. The green platform indicates separation from the upper free air portion and lower water filled portion.



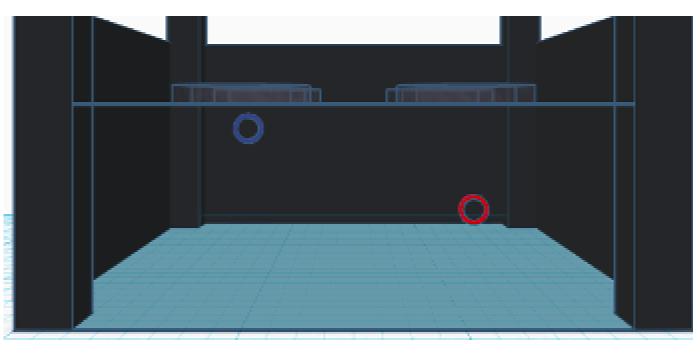
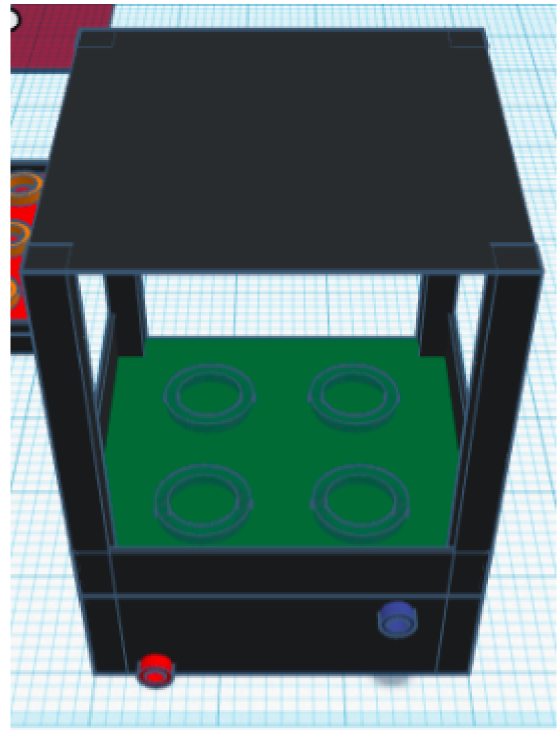


Figure 3-39: Garden Pod Front

Rear of Pod Design:

- Two valves will be used to manage water changes for the pod. The upper valve (blue) will be used primarily for filling and the lower valve (red) will be used to drain the water out.
  .



The pod electronics will be placed on the top of the pod under a cover. This is in interest of keeping the electrical components as far away as possible from the water reservoir. From a marketing perspective, we would offer multiple options for the height of the pod so that various plants could fit. The two options would be 12 inches tall or 18 inches tall. Our prototype will feature 12 inches of growing space.

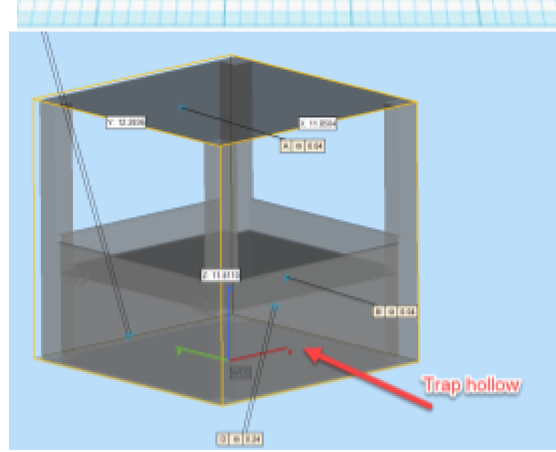


Figure 3-40: Trap Hollow

3D printing the pod will be a challenge depending on how much of it we want to assemble ourselves. Firstly, printing anything with an enclosed space creates a trap hollow (shown in **Figure 3-40**) which complicates the printing process. Most designs include an outlet that the printer can exit from to prevent this from happening as well as using support material. Another challenge with printing this pod will be the printing envelope of the 3D printer. The printer at ucf has a maximum printing envelope of 12x12x10 inches, which is not enough to print the pod in one piece. To prevent exceeding the printing envelope, we separated the pod into an upper and a lower section since height was the main dimension that was exceeding that envelope. Upon reviewing our STL files, the printing lab found that our design is very close to a box and is made up mostly of panels. For this reason they recommended that we consider laser cutting the panels for the pod. This is where we must consider the assembly mentioned earlier. Ideally the less assembly the better, both for structural integrity and the pursuit of a waterproof design. Assembling the pod in panels increases the work that we will need to do to ensure that the strength of the pod is not compromised.

When 3D printing, choosing the correct material is very important. There are many factors and applications to consider when choosing. **Table 3-(14-15)** details the areas to focus on when choosing 3D print material [20].

Material Property	**Definition**	**Function**
Tensile Strength	Resistance of a material to breaking under tension	Fundamental property that shows the ultimate strength of a part. High tensile strength is important for structural, load bearing, mechanical, or statical parts.
Young's Modulus	Resistance of a material to stretch under tension (stiffness).	Good indicator for either the stiffness (high modulus) or the flexibility (low modulus) of a material.
Flexural Strength	Resistance of a material to breaking when bent.	Similar to tensile strength, but shows strength in bending mode. Also a good indicator if a material is isotropic (homogeneous).
Flexural Modulus	Resistance of a material to bending under load.	Indicates toughness, helps you figure out if a part will survive when dropped on the ground or crashed into another object.
Indentation Hardness (Shore)	Resistance of a material to deformation.	Helps you identify the right "softness" for rubber and elastomers for certain applications.
Water Absorption	Amount of water absorbed under specified conditions.	Mostly important during the processing of the raw material, high water absorption or humidity can lead to poor material properties in thermoplastics.
Heat Deflection Temperature	Temperature at which a sample deforms under a specified load.	Indicates if a material is suitable for high temperature applications.
Vicat Softening Point	Temperature at which the material becomes noticeably soft.	Used for materials that have no definite melting point. For high temperature applications it helps determine the upper temperature limit for continuous use.
Thermal Expansion	Tendency of a material to expand (or shrink) in response to a change in temperature.	Important for applications where a shape change in response to temperature is unacceptable or desirable.

Table 3-15: 3D Printing Material Properties

Based on our application and the factors detailed in **Table 3-14**, a plastic printing material such as ABS or PLA. Firstly these materials have very common use, are very accurate, and will allow us to keep the cost of production down to a minimum.

3.5 Nutrients

A very important part of hydroponics is the nutrients in the water. The very definition of hydroponics is just plants grown in water filled with nutrients. For plants to grow properly in a hydroponic system they require around 17 different nutrients. Processes that are critical to plant development and growth. These nutrients are categorized into macronutrients and micronutrients. The macronutrients are in higher quantities than the micronutrients. Below shows the distribution of nutrients that are necessary for hydroponic growth [2].

Nutrient (chemical symbol)	**Approximate content of plant (% dry weight)**	**Roles in plant**	**Source of nutrient available to plant**
Carbon (C), hydrogen (H), oxygen (O)	90+%	Components of organic compounds	Carbon dioxide (CO_2) and water (H_2O)
Nitrogen (N)	2–4%	Component of amino acids, proteins, coenzymes, nucleic acids	Nitrate (NO_3^-) and ammonium (NH_4^+)
Sulfur (S)	0.50%	Component of sulfur amino acids, proteins, coenzyme A	Sulfate (SO_4^-)
Phosphorus (P)	0.40%	ATP, NADP intermediates of metabolism, membrane phospholipids, nucleic acids	Dihydrogen phosphate ($H_2PO_4^-$), Hydrogen phosphate (HPO_4^{2-})
Potassium (K)	2.00%	Enzyme activation, turgor, osmotic regulation	Potassium (K^+)
Calcium (Ca)	1.50%	Enzyme activation, signal transduction, cell structure	Calcium (Ca^{2+})
Magnesium (Mg)	0.40%	Enzyme activation, component of chlorophyll	Magnesium (Mg^{2+})
Manganese (Mn)	0.02%	Enzyme activation, essential for water splitting	Manganese (Mn^{2+})
Iron (Fe)	0.02%	Redox changes, photosynthesis, respiration	Iron (Fe^{2+})

Table 3-16: Nutrients and Effect on Growth

Nutrient (chemical symbol)	Approximate content of plant (% dry weight)	Roles in plant	Source of nutrient available to plant
Molybdenum (Mo)	0.00%	Redox changes, nitrate reduction	Molybdate (MoO_4^{2-})
Copper (Cu)	0.00%	Redox changes, photosynthesis, respiration	Copper (Cu^{2+})
Zinc (Zn)	0.00%	Enzyme cofactor-activator	Zinc (Zn^{2+})
Boron (Bo)	0.01%	Membrane activity, cell division	Borate (BO^{3-})
Chlorine (Cl)	0.1–2.0%	Charge balance, water splitting	Chlorine (Cl^{-})
Nickel (Ni)	0.000005–0.0005%	Component of some enzymes, biological nitrogen fixation, nitrogen metabolism	Nickel (Ni^{2+})

Table 3-17: Nutrients and Effect on Growth(2)

Luckily for us these nutrients are already combined for us and can be purchased all together. These premixed solutions come in macronutrients and micronutrients

As seen in the table there are a mixture of cations and anions. The more prominent nutrients required for plant growth are CO_2 and H_2O which are carbon dioxide and water. These are basic components of organic compounds required for all plant growth. These nutrients are also readily available and well known as plant requirements. Nitrogen is also one of the key components in plant growth and is a prominent element in amino acids and proteins. The specific compounds required are nitrate and ammonium. All other elements and compounds of plant growth are equally as important and more complex in their effect on the plants and each effect of those compounds and elements have been detailed in the chart.

Luckily for us these nutrients are already combined for us and can be purchased all together. These premixed solutions come in macronutrients and micronutrients since the two solutions can not be mixed together stably. If the Macro and Micro nutrients are mixed together then they will react and precipitate because of the mixture of anions and cations. Some options are:

- Hydroponic Nutrient Solution - Complete Plant Food System - Part A and B
- General Hydroponics Maxigro, Maxibloom
- And others

Nutrients are not the most important as long as they contain the proper chemicals. The requirements for our own specific chemicals are that they are inexpensive and effective.

Before we look harder at the types of premixed nutrients we should use in our system we need to understand how they are advertised. Almost all hydroponic nutrients use the NPK system. The NPK numbers are a way to denote the different amount of nutrients in them. Different cycles of plant life require different amounts of chemicals. The NPK number essentially says how much nitrogen, phosphorus and potassium are in the nutrients mix.

The problem with using nutrients mixes that are useful at different plant stages is that the nutrient mix would have to be adjusted by the consumer so there are two routes that would be ideal when designing the system. Route one is to go with the best nutrients and have the consumer adjust them according to the stage in plant life and the other is to use nutrients that would work during all stages of plant life. Although nutrients are important there are some mixes that can be used to fertilize the plant year round. Using this method would mean less managing from the user which seems like the better option.

Here are some of the different mixes that we researched and analyzing to find the most optimal choice:

General Hydroponics Flora Grow, Bloom, Micro Combo Fertilizer

This option is one of the highest rated options out there. The flora grow is extremely popular and very consistent. Flora Grow splits up the nutrients into three cycles and provides recommendations on how to use them throughout the cycles. Flora Grow contains many trace minerals and elements to promote growth in the growth stage, flowering in the flowering stage and larger crops in the vegetation stage.

Figure 3-42: FloraGrow

AeroGarden Liquid Nutrients

Almost the opposite of Flora Grow, AeroGarden uses one formula year round to promote plant growth. With AeroGarden you only have to mix the liquid solution with the recommended amount of water in the instructions and it is good to go. The downside is with a single formula you don't have the ability to customize your nutrient formula with your plant cycle. Even

Figure 3-43: AeroGarden

so, the simplicity of the AeroGarden formula makes it a good option for beginners and a good option for end users who don't have much experience with hydroponics.

General Hydroponics 718281+718301 Maxigro and Maxibloom Fertilizer

The last option that we looked at is general hydroponics Maxigro and Maxibloom. Since we are focused on herbs mainly for our hydroponics system it makes sense to use a fertilizer that works well with herbs. After some research we found that Maxigro and Maxibloom encourage strong growth and work well with herbs to create healthy plants that produce a lot.

Figure 3-44: MaxiGro & MaxiBloom

3.6 Ph values

The ph of the water is the primary concern for the water in the hydroponics system. The pH of water is the measure of the relative acidity of the water, it is very important for plant nutrient availability in hydroponics. PH is measured on a scale of 0 to 14, 0 being acidic 14 being basic. The ideal range for growing plants hydroponically is around 5 to 7. We will use our liquid ph sensor to determine the ph level of the hydroponic system. According to what the ph level is there is an amount of ph up or ph down added. The pH up and pH down are premixed fluids that are purchased and mixed in automatically by the controllers embedded in the hydroponics systems. The pH up is a basic solution used to neutralize some of the acid. The pH down is an acidic solution.

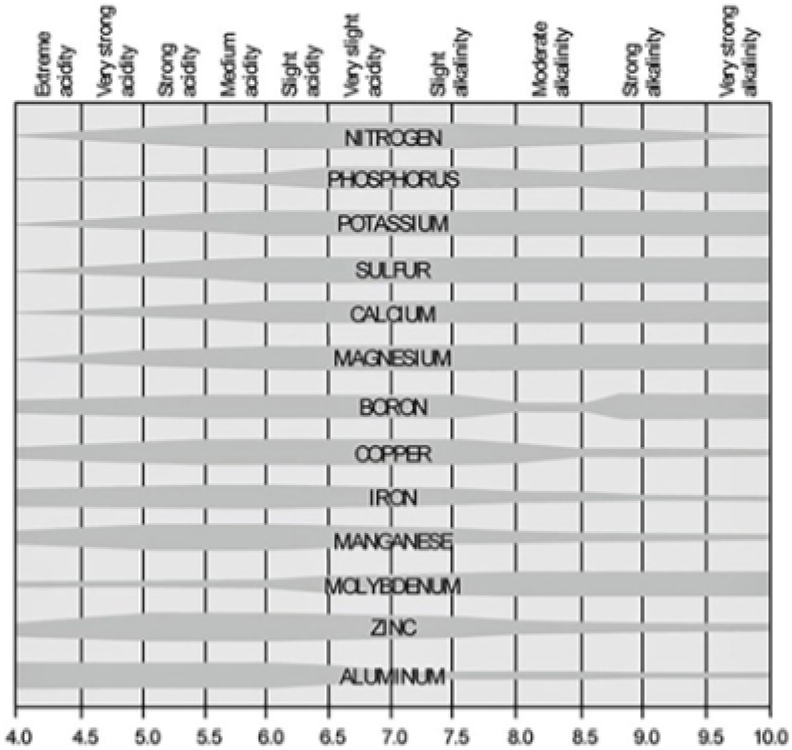


Figure 3-45: pH Values

PH is important as you can see in the table a different pH leads to different absorption of the nutrients required for plant growth. Different plants require different nutrients so for different types of crops there are different pH ranges required. Generally the range that most plants follow is slightly acidic.

To find out why pH is important we need to look at the germination of plants with different pH. In a study on “Effect of Soil pH on the Growth, Reproductive Investment and Pollen Allergenicity of Ambrosia artemisiifolia L.” we can see how germination is changed at different pH levels.

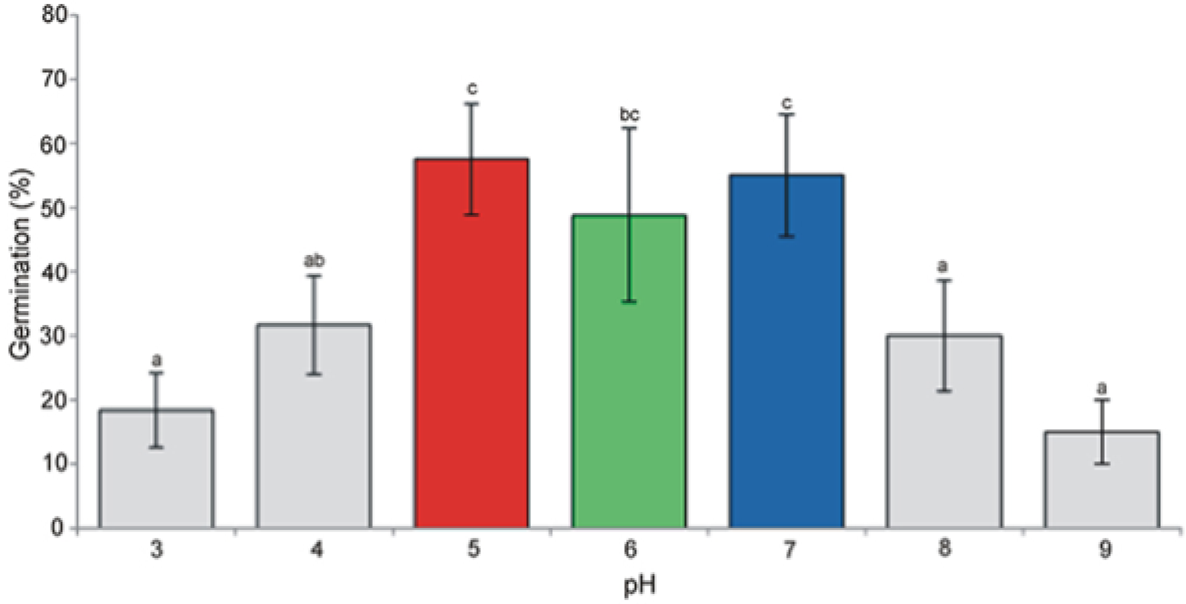


Figure 3-46: Germination Based On pH Level

We can see that plants prefer a range of slightly acidic values like we stated in the previous section for ideal germination [2].

4 Related Standards and Realistic Design Constraints

In this chapter, we will discuss some standards that were kept in mind during the design stages of our hydroponic system. These standards are what helps our product remain compatible with technologies in the market today. The last thing you want a consumer to do is to have to reconfigure their lifestyle to use your product. There are numerous standards and constraints in the world of technology; however, the ones discussed in this chapter are what we felt most influenced our design process.

4.1 Standards

American Development of Standards first began in the 1900s and have played a large role in professional and technical societies. It is important to have these standards because of the already present variation in the consumer market. These standards are guidelines and rules that regulate how products and materials should be used. Standards ensure uniformity, reliability, comparability, and safety.

4.1.1 Electromagnetic Fields/Interference

Often called radio-frequency interference(RFI), electromagnetic interference (EMI) takes place on an electrical circuit due to another electrical device in its vicinity. [6] These disturbances can range from performance loss to complete failure of the electrical system. EMI can result from man-made or natural sources and a lot of these sources are found in the average household. There are two industry standards to consider when dealing with EMI. The first standard is the IEEE standard on human exposure to EMI. IEEE SCC28 states: “Safety limits for the protection of persons against the established adverse health effects of exposures to electric, magnetic, and electromagnetic fields in the frequency range 0 Hz to 300 GHz are presented in this standard. These exposure limits are intended to apply generally to persons permitted in restricted environments and to the general public in unrestricted environments.” The second standard, concerned with making sure that the electronic circuits perform as intended, introduces the concept of electromagnetic compatibility (EMC). Electromagnetic compatibility standards guarantee that the electronic component can both operate without lack of performance while also not interfering with any other systems in the same area. Not only will we need to ensure that our system does not emit any harmful electromagnetic frequencies, but we will also need to make sure that it is not susceptible to the frequencies generated by other electronic devices. There are a lot of ways to mitigate both of these risks and ensure the system is compliant with active EMI standards.

The most common method used to protect against EMI is to use some sort of shielding around either the source electrical circuit or the susceptible circuit. [8]This can be done by using a metal screen over the circuit components. This screen follows the same principles as a Faraday cage, and must completely cover the source of EMI or the susceptible electronics to be effective. The shielding layer needs to be made up of a conductive material so it can absorb the field being emitted or absorbed. With these screens in place, heat management can also become a concern. If the circuit experiences high amounts of heat and needs more airflow, a screen with "grille" like holes can be implemented. The screening becomes drastically more effective if the holes surpass 80% of the ratio of hole to metal. Figure 4-1 shows an example of a shielding.

Figure 4-1: Electromagnetic Screening

Another way we can prevent any harmful EMI in our system is to ensure that we are only using the necessary amount of wire between all components. Using an excessive amount of cables means that there will be more surface area to be affected by EMI.

4.1.2 WiFi (802.11)

In order to be eligible to be a WiFi enabled device, our system must meet the IEEE 802.11 standards that describe both medium access control (MAC) and physical layer specifications for wireless local area networks (WLAN). The standards set under 802.11 were created to improve the LAN experience for consumers. These standards regulate throughput, range, and ranges of frequency. 802.11 standards are developed very frequently (every 2-3 years) in order to account for the rapid new development of the technology utilizing WiFi. The newest 802.11 standard is 802.11ax which will replace both 802.11ac and 802.11n LAN standards. Unlike the earliest WiFi standards, these will be backwards compatible and will be compatible with routers using earlier standards. **Figure 4-1** shows the evolution of these wifi standards.

Year	WiFi Versions/Protocols	Old name	New name
1999	First Generation	WiFi 802.11b	WiFI 1
1999	Second Generation	WiFi 802.11a	WiFi 2
2003	Third Generation	WiFI 802.11g	WiFi 3
2009	Four Generation	WiFi 802.11n	WiFi 4
2014	Fifth Generation	WiFi 802.11ac	WiFi 5
2019	Sixth Generation	WiFi 802.11ax	WiFi 6

Table 4-1: Wifi Versions/Protocols

Based on the standards mentioned, our main point of concern will be configuration of the wireless communication through WiFi and Bluetooth. Because of the numerous generations of 802.11, ensuring that every device we implement has a compatible protocol will be very important for functionality. Additionally, these wireless modules will add size to the electrical components being placed in the system, and the housing locations will need to accommodate the increase in space needed. Whenever making design choices for wireless communication, you have to consider the device receiving the wireless information. For example, if we want the user to be able to place the Garden Pod a larger distance from the central unit, we need to ensure that the transmitting and receiving components can handle that range.

4.1.3 Power Supply (Mains)

In most parts of the world, the 230 volt standard has become the most widespread voltage level. Our system will adhere to the North American (US and Canada) standards for primary power supply. This means that our system will be designed to operate on the current 120 volt standard. The North American power supply standards also come with safety measures that our system should comply with. Although the North American standards for electrical product safety are voluntary, they do provide guidelines that ensure safe operation. In the US, mains electric power is referred to by several names, but it typically concerns the primary power source to the standard American home. **Table 4-2** discusses the common terms used when working with mains power supply, and describes the naming conventions used for power applications. Prior to researching this field of industry, we would not have realized how many different conventions and traditions are used in this field. Using this information, we could better understand the other products we investigated to implement in our system as not every company uses the same voltage conventions when discussing powering devices from a mains power supply.

Voltage	Designation
110/220	The designations "110 volt" and "220 volt" represent an older, out of date standard that's no longer found in new equipment. However, this terminology is still familiar to many people, so it remains in use.
115/230	The designations "110 volt" and "220 volt" represent an older, out of date standard that's no longer found in new equipment. However, this terminology is still familiar to many people, so it remains in use.
120/240	The power delivered to your home is 120 or 240 volts. This is called "nominal voltage." That means it's the standard voltage as measured at the transformer outside your home. Nominal voltage can vary up to plus or minus 5 percent from its stated definition.
125/250	The outlets in your home are rated at the maximum voltage expected on the electrical circuit. They are designed to take up to 125 or 250 volts, depending on the nominal voltage of the circuit. Thus, outlets are marked at 125 volts or 250 volts.

Table 4-2: Mains Power[9]

4.1.4 Water Supply

Our hydroponic system will require the system owner to refill the main reservoir when it needs to be refilled. For this reason, we have to look at the quality of the water that will be cycled through the components of our hydroponic system. When choosing the tubing, pumps, tank material, etc., we referenced the [8] United States Environmental Protection Agency (EPA) to ensure that we know what kind of quality of water to expect. The following Figure is an excerpt from the Water Quality Standards Handbook, Chapter 3:

Water Quality Criteria

In accordance with 40 CFR 131.11(b)(2), in adopting water quality criteria, states and authorized tribes should "establish narrative criteria or criteria based on biomonitoring methods where numeric criteria cannot be established or to supplement numeric criteria."

The following provides an example of a narrative criterion (adapted from the EPA's *Model Water Quality Standards Template for Waters on Indian Reservations* (2016)):

All waters shall be free from toxic, radioactive, conventional, non-conventional, deleterious or other polluting substances in amounts that will prevent attainment of the designated uses specified.

All waters shall be free from substances, attributable to wastewater discharges or other pollutant sources that do one or more of the following:

1. *Settle to form objectionable deposits.*
2. *Float as debris, scum, oil, or other matter forming nuisances.*
3. *Produce objectionable color, odor, taste, or turbidity.*
4. *Cause injury to, are toxic to, or produce adverse physiological responses in humans, animals, or plants.*
5. *Produce undesirable or nuisance aquatic life.*

Figure 4-2: EPA Water Standards

4.2 Realistic Design Constraints

When designing any system that is intended to be used by consumers, one must consider many different design constraints. These constraints will allow us to determine how practical this hydroponic system will be. Throughout this section, we look at economic, time, environmental, social, political, ethical, health, safety, manufacturability, and sustainability constraints.

4.2.1 Economic and Time Constraints

By getting a head start on finding a price range for each component, our group will be able to make parts selections more confidently. During our initial meetings, we discussed what our maximum budget should be, as we are executing this Senior Design project with no sponsor. One of the major components that could alter the budget is the 3D printing of the structural components. The printing company and type of material will dictate exactly how much we will need to spend. This variation is why we chose multiple parts and determined a range of prices for each part so that we can make any necessary adjustments to ensure we are still on track to meet our budget.

Because we will have to wait for all the parts to be delivered, we will need to be done or almost done with all design components of the system so that we are not waiting for parts during our prototype phases.

The biggest time constraint is in research time. There are many possibilities for this project and many ways to approach our design. Designing an automated hydroponic system is no easy task, especially when competing with several other similar devices. For every part to select there are hundreds, if not thousands of selections to make and unfortunately not all of them are up to the task. There are counterfeit ultraviolet LEDs, Mosfets that are very close but don't meet specifications, the perfect parts but only with the longest lead times. Every project has its setbacks but as engineers we must find a way. There are a lot of options, which can honestly be paralyzing at first, but once you dissect the problem, realize the main details, and set small goals, you can get into a workflow and let those little achievements build up.

4.2.2 Environmental, Social, and Political Constraints

The production of hydroponic systems incur very little ethical backlash. Growing your own vegetables in your home is a very clean and green way of living that is perceived in a positive way by the general public. Localized farming encourages self-sustainability and reduces the need for large industrial farming companies that could be of detriment to the environment. However, there is a possibility of pushback from purest and traditional. If a device like this were to be fully realized and automate the farming industry, many people may be upset over the loss of human jobs. Others may not like that there is no human intervention in growth food either, as some may worry the process does not seem natural. Whatever the reason, there are not many things in this world that will please every single person, but we certainly will try to make this device as human friendly as possible and let the plants do the talking.

Another point of controversy related to hydroponic systems is the United States Department of Agriculture (USDA) classification of hydroponic farming as "organic."[18] Their study "Report and Recommendations on Organic Agriculture" states that "Soil is the source of life. Soil quality and balance are essential to the long-term future of agriculture. Healthy plants, animals and humans result from balanced, biologically-active soil." A short amount of research will reveal that most hydroponic systems do not use any soil at all, and if any it does not need to be organic. The issue that most organic farmers have with this definition is that hydroponic farming allows large farming companies to skip past hundreds of years of soil development while seemingly taking all the same credit with the organic label on their product. The reality is most organic farmers are not large enough to continue their operation without suffering major losses. In 2002 they would revise their classification of organic crops to remove the word "soil." [19]The controversy stems from two sides: farmers who go to great lengths to ensure that their product maintains the organic quality standard, and large industrial farming agencies who are taking advantage of vertical farming and similar forms of hydroponics. Big farming companies like this are taking advantage of this loophole by using this method of farming (on a much larger scale) and then labeling their product on the store shelves as being certified organic. There have been many objections to this established definition by organizations like the National Organic Standards Board

(NOSB) which is composed of farmers, scientists, and public advocates. Unfortunately their early efforts of objection were not successful. Not only does this contradict the foundational definition of organic produce, but it makes running and operating small traditional organic farms even more challenging. As with any industry, the presence of a free market allows those with more resources to capitalize in certain areas. For that reason, this is not an attack at the USDA by any means, but we must consider the way that the consumer market views a system that we plan to produce.

This hydroponic system allows the consumer to plant whatever they want in preparation to use its automated capabilities. It is no secret that hydroponic systems have been commonly used to grow illegal substances. Our goal with this system is far from allowing consumers to grow illegal plants; however, we cannot realistically regulate what each consumer chooses to grow. What we can do, is provide encouragement to follow the laws and regulations of the current legislation based on their state and country. This system is intended to grow plants, herbs and vegetables, the instruction and guidance will provide support only for legal items.

4.2.3 Ethical, Health, and Safety Constraints

We could not write this section, without acknowledging what a rough year it has been during this pandemic. The greatest challenge for our group has been the current pandemic and the effect it has had on the university. This class has been conducted entirely online which has presented multiple challenges in terms of group member collaboration in person. In fact, all of our teams meeting up to this point have been conducted on our Discord server to ensure that all of us could be present. However, online interaction is no full substitute for human, in-person, interaction. It has been somewhat difficult to collaborate and share ideas through a digital screen. We also cannot forget to mention the other “elephant in the room”, mental health. It is proven that lack of human interaction and outdoor time can have a significant impact on one’s emotional state. Motivation is a huge part of teamwork and productivity and it has not been easy to maintain, given the challenges of virtual school. This virtual format of learning is new to all of us, students and professors included, and maybe one day there will be better infrastructure in place in case this situation were to happen again. For now, These challenges will continue into the next semester during the construction of our prototype. That being said, we have full confidence that we will be able to pull through and develop our prototype.

The hydroponic system will have safety at the forefront of the design considerations. With there being both water and electrical components in the same system, we will have to make sure that they are properly sealed off from each other. Storing the concentrated pH balancing liquid must be secure, especially in households with small children. A bottle of these chemicals in the wrong hands could be very harmful if ingested.

When dealing with water that could be standing in a container for a prolonged period of time, we must consider the possibility of mold in the system. We can assume that if the main reservoir is not emptied for long, some mold will develop which will find its way in all water transport components and eventually the Garden Pod itself. Mold in the hydroponic system is not catastrophic, but it is a situation which will require extensive cleaning by the user and should be avoided. For this, we will need to ensure that the main reservoir has a maximum length of time for the water to be there. Providing clear and descriptive instructions to the user will ensure that most if not all of the safety concerns present themselves during normal use.

Another possible hazard of this system can be prolonged exposure to ultraviolet radiation. The horticulture light emits a spectrum of light akin to sunlight. Though it is only a fraction of the strength of the sun. We should exercise caution and assess any potential risk to human health. Ultraviolet radiation occupies the 100nm - 400nm range of the electromagnetic spectrum. Within the UV spectrum there are three regions, UV-A (315nm - 400nm), UV-B (280nm - 315nm), and UV-C (100nm - 280nm). The Sun emits ultraviolet light but the earth's atmosphere blocks UV-B and UV-C rays so we are only exposed to UV-A in the daylight. With no natural occurrence on the surface of the earth, Any exposure of UV-B and UV-C rays only comes from artificial sources, mainly lab equipment such as ozone generators, germicidal lamps, solar simulators, plasma etchers, UV curing systems, etc. Fortunately our design does not emit UV-B or UV-C, the grow light fixture only emits a narrow sliver at the end of the spectrum with the widest wavelength. The radiation should be safe for human exposure, but we will be using safety glasses to protect our eyes against any exposure while examining and testing the LEDs.

Finally, we must take note of the electromagnetic compatibility (EMC) standard discussed above. The products that we implement into this hydroponic system will restrict the Hazardous Substances Directive (RoHS) and the EMC directive to ensure that no risk is incurred.

4.2.4 Manufacturability and Sustainability Constraints

For troubleshooting, we intend to use products that are readily available in case the consumer would like to do his/her own repairs, but we also want to make the product as bulletproof as possible. This includes taking great considerations in structural components, build quality, and electronic operating requirements to prevent hardware failure. We will need to put things like heat management, material, and wire management at a high priority to ensure sustainability of the system. Since the pod is designed to only be moved when being placed back on the central unit for scheduled maintenance, we suspect a very low risk level as the majority of the pod's life will be spent untouched.

With systems that are mostly autonomous, the materials used need to be stronger than normal components as they will be sitting for a long period of time. This would mean designing a system that is intended for indoor use, but strong enough to be left outside. Only then can we be absolutely sure that the pod will withstand all of the elements it is subjected to.

As a stretch goal, we would like to design the system so that part replacements are not impossible. This would mean components with easy to access joints, screws, attach points, etc. Things like replacing the light source at the top of the pod should be very possible considering it is a cheap fix compared to the price of the entire system. From a producer perspective, making products that are 100% proprietary is highly unpopular in the consumer world. Consumers want to be able to access repairs easily, and having to wait for their product to be fixed by the only authorized repair source can be a stressful and expensive experience. For that reason, most people stray away from proprietary systems which is why we don't want our hydroponic system to adopt unpopular characteristics.

5 Project Hardware and Software Design Details

In this chapter, we will discuss the hardware and software design aspects used in this project. We will start by looking at the early diagrams used to specify our direction when choosing what type of hydroponic system we would like to pursue. After that we will consider the subsystems that will need to be implemented in order to make this system functional. Additionally, this chapter will break down the software architecture and how it will relate to the overall operation of the hydroponic system.

5.1 Initial Design Architectures and Related Diagrams

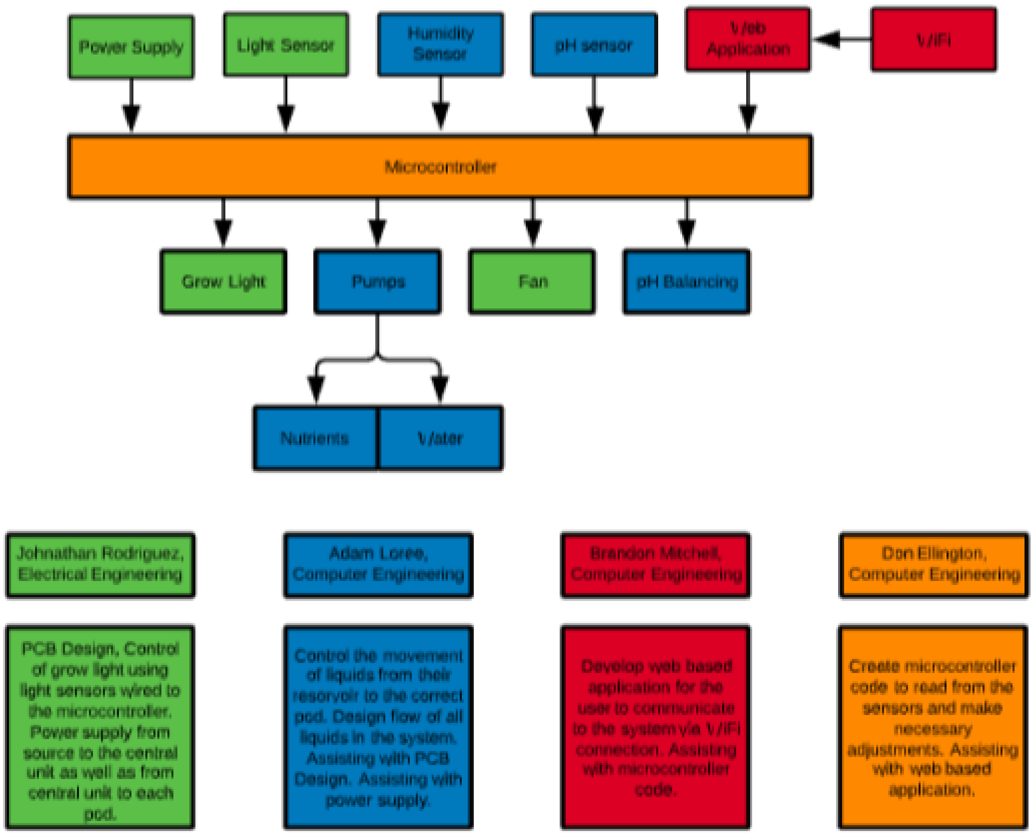


Figure 5-1: Initial Team Design Roles

The design components of the hardware can be shown in **Figure 5-1** above. Each team member's block color correlates to their responsibility with each component. For example, every green block represents components within Jonathan's scope of responsibility. Each component will have a primary team member and a secondary team member that will be the first to assist when needed. This way we will not have a component that becomes overbearing or mistakenly incompleted.

5.2 First Subsystem, Breadboard Test, and Schematics

This section will cover the implementation, testing, results, and conclusions about the products we selected to use for this hydroponic system. Not only will the equipment be tested to see if it is functional, these subsystems will be tested to make sure that they will perform the necessary requirements for our project specifically.

5.2.1 Central Unit Power Supply

The power supply for the central unit will adhere to North American mains electricity standards. Our central unit will use a standard AC adapter for connection to the consumer's wall outlet. Testing will reveal the exact voltage of each electrical component. It is highly likely that our electrical components will need to use voltage regulation to ensure they get the proper power supply. The utilization of MOSFET devices for voltage regulation is mentioned in Chapter 3.
(include initial testing, findings, conclusions, adjustments needed)

5.2.2 Central Unit Pod Charging

A major feature of our hydroponic system is the flexibility to place the Garden Pod anywhere in the house once in between water changes/charging. In order to ensure that the pod can operate independent of the central unit, there must be a power source for the pod to use during normal use times. We will implement a battery into the pod that can be charged during the regular maintenance (water, nutrients, etc.). The central unit must have the ability to charge the pod using the power supplied from the main power supply used to power the other electronics.

There are many design considerations to make when implementing a charging function to the central unit. Firstly, because the nutrient and water hose will need to be attached to the pod during scheduled maintenance, we will add a power charging cable to be connected to the pod. Next, we have options when it comes to where to place the technology responsible for safe charging. For example, if the pod utilizes a lithium polymer or lithium ion battery with multiple cells, the charger will need to be able to perform balanced charging to ensure safe charging and that the battery lifespan is maximized. If we implement a lithium ion battery, we can expect to trade a longer charge time for a longer power delivery cycle. These comparisons are explained in **Figure 5-2.** Further testing of the charging process will reveal where we want to locate the charger so that we know it is in the most optimal location.

In the later stages of our project, we decided to use a 12V DC 6800 milliamp-hour(mAh) battery that comes with both an input and output connector. This battery is not only rechargeable, but it also has various safety features that mitigate the risks of using a lithium ion battery. The battery we chose to implement has battery overcharge, over discharge, over current and external short circuit effective protection. Using a battery like this, along with the safe design of the power PCB will ensure that we can deliver power to the garden pod in a worry-free environment.

In the final construction of our garden pod and central unit, we decided to make an adjustment to the charging location of the garden pod. Because our battery has an easy to use plug for a charger to connect to it, the garden pod battery can be charged from any location using a simple 12V adapter with the corresponding bullet connector. Additionally, the garden pod can be powered using only a 12V adapter if the user does not want to use a rechargeable battery. The decision to include a cable for charging in the central unit was changed due to the features of the 6800mAh garden pod battery.

Figure 5-2: Li-Ion vs. LiPo[17]

5.2.3 Central Unit WiFi Module

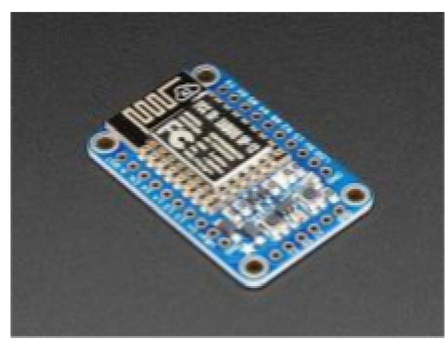


Figure 5-3: Adafruit HUZZAH WiFi Module

There will be one WiFi module located in the central unit. This module will allow the consumer to control and monitor the hydroponic system with the application that we are developing. There are many wifi modules present in today's market that can be implemented to the microcontroller with little to no modifications needed. The wifi module will be a device that is mostly plug-and-play into the microcontroller. After completing wiring, our next step would be to ensure that we can configure the module to an internet connection and ensure that it successfully communicates all the information needed between the application and the central unit. Pairing this module with the Adafruit HUZZAH breakout board will make testing and configuration a simple process. The breakout board includes buttons, logic shifters, and LEDs that will allow us to troubleshoot any of the wireless connections on setup. The logic shifters come on diodes on the communication pins.

These logic shifters are necessary for implementation with Arduino and any devices or sensors utilizing the Arduino interface. This is a very low power module, as it only needs a stable 3.3V connection to operate, and will likely be mounted on the PCB because it is going in the central unit.

5.2.4 Central Unit Bluetooth Module

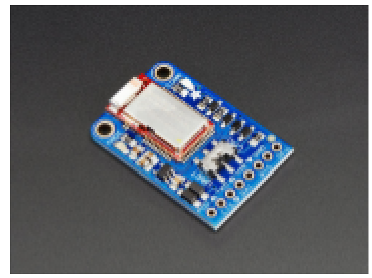


Figure 5-4: Bluefruit LE Bluetooth Module

The Bluetooth module located in the central unit will take care of receiving information from the sensors on the pods in use. The Bluefruitt LE UART Friend allows us to add low energy connection to our Arduino via serial port. The bluetooth module will function as a wireless UART adapter that will allow the central to communicate quickly and efficiently with the pod. It is an arduino friendly board that only requires 5V input and has an on-board 3.3V voltage regulator. Although we are using MOSFETs for voltage regulation, it is a plus to have this feature for additional devices. Because this module is easy to implement with the devices we have selected, we will not have to do much testing or development. All that is needed for setup is a serial connection and programming.

5.2.5 Pod LED Lighting

Figure 5-5: Samsung LED Strip

The lighting that will be used in each pod is meant to supplement the ambient lighting in the room, not replace. For this reason, picking a light source that can provide variable light intensity and wavelength is very important. Picking an LED that gives us this option will ensure that the plants are given the best environment to grow in.

Our first choice for lighting was the "Horticulture L2 Module" which is a 24 inch LED array. According to technical specifications, this module can provide 141.8 microjoules per second, two modules should provide sufficient radiation to fuel plant growth. Under typical conditions, this device will run 1.2 amperes of current with an ambient temperature of 25° celsius. The operating voltage is around 43.1 volts and will consume an average of 51.7 watts. However, due to pod size and battery size constraints. We decided on the Phyto Lamps: UV LED Strip, This strip claims to deliver UV radiation in the red and blue light spectrum. We will discuss this change further in the testing section.

We will first be testing the actual values of voltage, current, and power draw. Once we have determined that the LEDs are properly powered, we can measure the heat emission of the system and make sure the heat can be properly dispensed. Then we observe plant growth and evaluate if we will need more LEDs to improve the PPFD. If another array is required, we will repeat the testing process.

5.2.5.1 Initial Testing

In the initial testing process we began by testing a very large led lighting array. This LED lighting array was a Samsung led bar and although it was very nice and had everything that was needed it had many drawbacks. We learned a lot of lessons testing this led bar which I will talk about in this section.

Figure 5-6: LED Light Initial Testing

At first looking at the LED bar we realised it may have been a little bit too big to fit inside of the unit. The space inside the unit is limited and has to be carefully monitored to ensure proper growth of the plant. Having such a large LED bar could prove problematic since we would not be able to fit other components in with the LED bar. The second drawback that we realised when we started to test the LED bar was the power draw. The rating for this LED bar was at an average of 40V. When we began testing we realised that we did even have a big enough power supply to power the LED bar. Not only were we not able to turn on the LED bar in the lab but this showed us how hard it may be to power it in the unit.

The unit's power supply is from a battery. The battery that is in this unit needs to be small and compact so that we can keep the weight, cost, and energy drawn down. Using a large LED bar like the one we initially tested would prove to draw way too many volts causing us to require not only a 40V battery but also a charger for the battery. A 40V battery inside of the unit is not cohesive with the rest of our parts in general. For a battery of this side we would most likely have to go with a lithium ion battery. These batteries are very good but also more expensive the higher the voltage.

We learned a lot from the initial test of the first LED bar. We realised the constraints of the unit and how much we would have to abide by them. Not only the constraint of space but also the constraint of power inside of the unit. We plan to either buy smaller voltage LED bars or to create our own LED bar with 3d printed materials and individual LEDs purchased separately.

5.2.5.2 Secondary Testing

Our initial testing of LEDs did not go according to plan. With our secondary testing we used LEDs better suited for our design. The LED strip was cheaper than our original choice and had an operating voltage which was already compatible with our design and would not require a boost converter . These strips are much more useful in our situation. You can see with these LEDs they

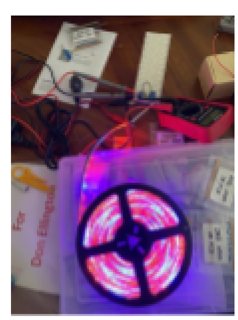


Figure 5-7: Secondary LED Testing

are much more manageable. These LEDs can be cut and stuck to our enclosure where they are needed. It is much easier to manage these LEDs with an Arduino as well. Using an Arduino we are able to modulate the brightness of these LEDs and also we are able to power them on and off using the Arduino. We are able to either switch them on and off using the Arduino as well as using the Arduino to control a mosfet to switch the LEDs on and off to prevent excess voltage to the Arduino.

5.2.6 Pod Power System

Figure 5-8: AC to DC Converter

Any potential overvoltages or excess of current can be catastrophic to electronics, but most especially to LED arrays. Using a voltmeter, We must first test the outputs of the AC to DC converter and measure for fluctuations in typical voltages at open load and typical currents with load. If they dwell within a decent range near expected values, we can make adjustments by implementing the necessary passive components. When the AC to DC converter is in working order, we can assess the output of the DC to DC converter. The current must be enough to drive all devices simultaneously.

Figure 5-9: 12V Adapter

Originally, we selected the UCC28600 8-pin Quasi-Resonant Flyback controller to construct an AC-DC converter. The UCC28600 is a PWM controller with advanced energy saving and high-level protection features. However, due to a shortage of voltage converters and time constraints, we had to opt for the 12V AC-DC adapter that came with the UV LED strip, which is capable of a maximum current of 6A.

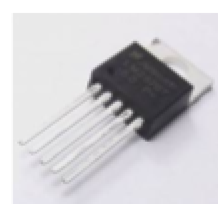


Figure 5-10: LM2576

For the DC-DC converter, we selected the LM2576. Using the LM2576 as a buck converter, The 12V DC input voltage is converted to a 5V source rated at a maximum of 3 ampere. The LM2576 is more efficient voltage regulation compared to the more popular three-terminal linear regulators and provides very consistent line and load regulation while requiring few external components. The voltage regulator is equipped with a rectifier diode, at its input, to protect it from overcurrent. To stabilize the regulator, capacitors and inductors will be at the output voltage

5.2.6.1 Initial Testing

To simplify our testing process we constructed our voltage regulator on a separate PCB. Using a multimeter, we tested the output voltages of the voltage regulators with the AC/DC adapter attached. In case the voltage regulator did not function properly, we installed multiple test points throughout the circuit. The PCB was outputting the proper voltage, so we attached the rest of the electronics, which ran as expected. Then we detached the adapter and ran the same test process for the 12V battery. Aside from faulty pin-headers the PCB had no issues.

5.2.8 Pod Water Level Sensor

To measure the water depth of the hydroponic system, we will include the Gravity: Non-Contact Digital Water/Liquid Level Sensor for Arduino. The sensor uses signal processing technology to measure water level. Since the water level is contactless the sensor will not be susceptible to corrosion and last much longer.

Figure 5-11: Water Level Sensor

The water level sensor will be continuously detecting the water level, monitoring the system for two instances, when the tank is empty, the system is notified that it is time for watering, and when the tank is sufficient filled, in this case the water level reaches a desired level and the water flow is cut off. This means this system will only require two flags. In this case, testing would only involve manually filling or emptying the tank to these water levels and observing the sensor's response.

5.3 Second Subsystem

This section will cover the central unit as a subsystem. We will look at devices to be implemented in the central unit such as the water tank, pumps, and sensors. All of these will be responsible for the transfer and management of liquids in the central unit. All other electronics concerning the central unit were discussed in the previous section.

5.3.1 Central Unit water tank

In the interest of keeping water isolated from other parts of the system. The reservoir used to store water will be its own container that will be accessed by the main water pump. Using the submersible water pump and water line, filling the pod during the maintenance cycle will be very easy to implement. If there is time and resources available, we will create a nozzle from the top of the central unit lid to the water reservoir so that you can refill the water tank without taking the lid off. This feature would function similarly to a vehicle gas tank.

5.3.2 Central Unit water pump

This DC 12V Mini Submersible Water Pump is specially designed for fountain and hydroponic systems and is resistant to moisture and corrosion. This diaphragm pump offers a flow rate of 1 gpm, and is

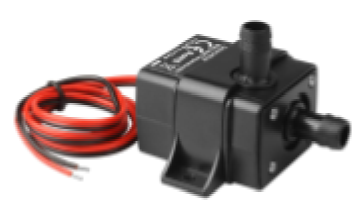


Figure 5-12: Central Unit Water Pump

capable of 3 meters of lift at a low operating amperage. This pump should be sufficient in driving the water flow of the garden.

We will test the range of fill rate this pump is capable of and the height at which the pump can deliver the water, then we can establish an appropriate fill rate for the tanks depending on the size of the tanks and establish how long the pump needs to run to adequately fill the tanks without overflowing.

5.3.3 pH Balancing Pump

If the pH sensor determines that the pH level of the pod needs to be corrected, the Gikfun 12V DC Dosing Pump will be used to pump the pH balancing solution to the pod from the central unit to the pod.

Figure 5-13: pH Balancing Pump

5.3.3.1 Initial Testing

We were able to test this pump in the lab using a small container with some water and a power supply to clip on the power leads. The pump worked as intended and it will be used in our system. The pump will only need to be powerful enough to transfer the liquid to the pod.

We did find that the connectors for the smaller pumps will need to use smaller lines compared to the water pump. Because the water pump will need to transfer more liquid and be used more often, we want it to be more powerful than the pump used for pH balancing. For this reason we will need to purchase different sized lines and different sized fittings for the pod. In the end, this difference in fittings will be a good thing because the user will be less likely to mix up the two connectors to the pod. Only the correct liquid line will fit on its respective female fitting. For this Peristaltic pump, we will use 2mm x 4mm water lines shown in that will simply slip on to the barbs on the input and output ports of the pump.

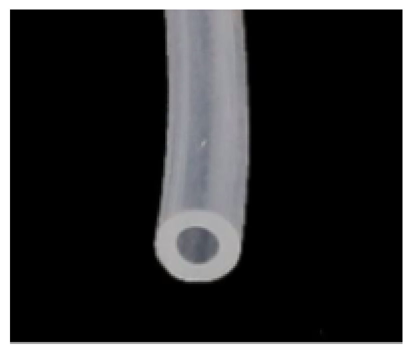


Figure 5-14: 2mm x 4mm Line

5.3.4 Pod pH Sensor

The Gravity: Analog pH Sensor will be used to measure the liquid pH levels of the water. This industrial pH probe uses a sensitive glass membrane with low impedance to quantify the pH. To test the accuracy of the probe, we have to use the probe on a solution with a known pH

Figure 5-15: pH Sensor

concentration, we can use the known pH as a reference and calibrate the probe to appropriate levels.

5.3.4.1 Initial Testing

Figure 5-16 shows the initial testing from the pH sensor. In the test the sensor was plunged into a slightly acidic bowl of liquid and the results were measured. The inputs of the Arduino are easy to use. The pH sensor has its own logic board and the results only need a voltage input, a ground and an analog input. You can see the pH results on the right of the image. The results show that the pH is being accurately recorded from the sensor.

This pH sensor in practice will be used to consistently manage the pH of the water in real time. The water that needs to be delivered to each of the pods needs to be a certain pH depending on the plant. We will need to analyze the pH in real time and adjust it with our pH control pumps to ensure the correct pH is delivered to the pods. This pH sensor will be measuring the values presented on the left side. It will analyze these incoming values in the Arduino and then use logic to deduce how much pH up or pH down is required.

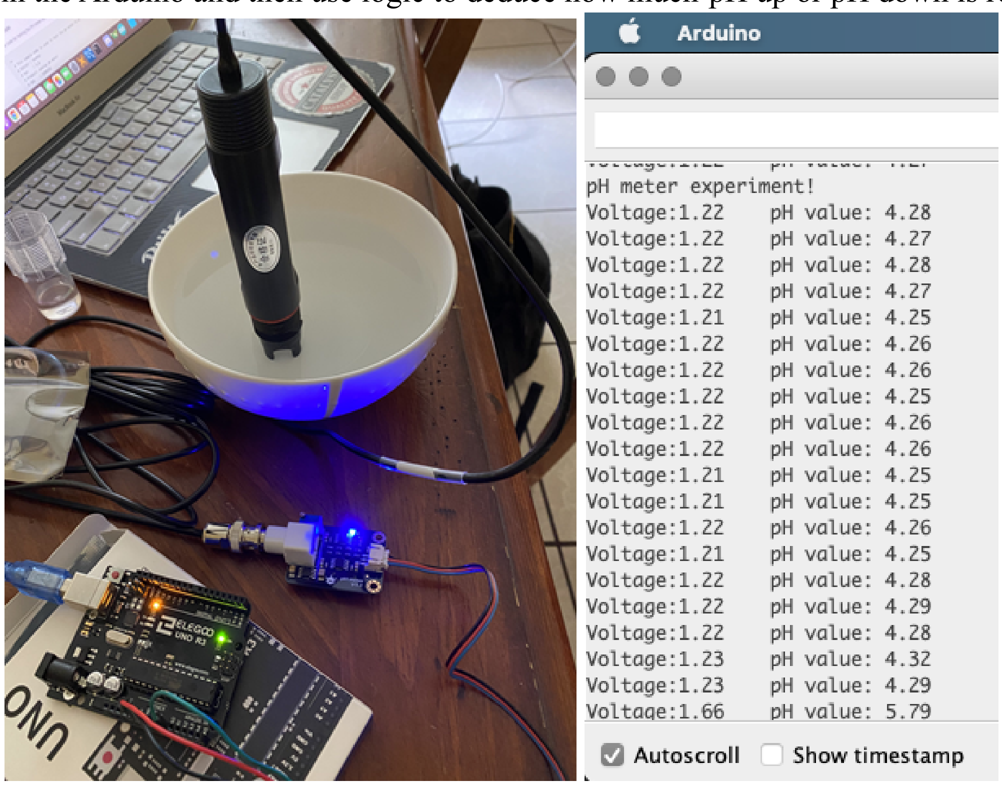


Figure 5-16: pH Sensor Testing

5.3.5 Central Unit Nutrient Bottle Storage/Pumps

The Gikfun 12V peristaltic dosing pump will be used to dispense nutrient solution into the water supply. These nutrient solutions will contain necessary nutrients for plant growth, however these solutions need to be properly balanced or can be detrimental to the growth of the plants. So we will have to test the strength and concentrations of these solutions and the flow rate of the peristaltic pumps. With this information, we can calibrate the strength of the flowrate and run-time to reduce fluctuations in pH and nutrients.

5.3.6 Pod Water Reservoir

The pod will be designed so that the bottom section will be used for water storage. This is both because of the function of hydroponic systems and also because it allows us to keep the electronics separated from the water. We will be using quick detachable water fittings, such as the one shown to make the refilling process more simple. After 3D printing the pod, we will know whether or not we will need to use an inner reservoir in the base. This would be a good feature to have in case the consumer would want to remove it for cleaning purposes.

Figure 5-17: Water Line Fittings

5.5 Software Design

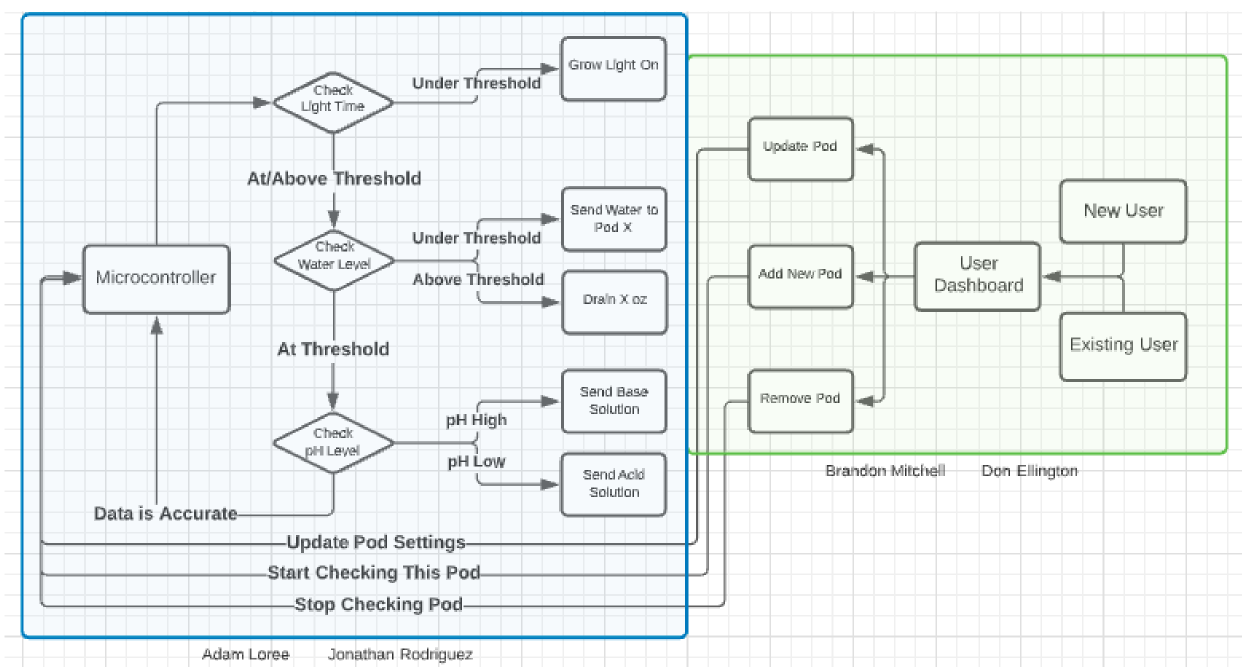


Figure 5-18: Initial Software Design

This initial software flow diagram shown in **Figure 5-18** shows the overall flow of the programming for the interaction between the user and the control unit, and then from the control unit to each pod. The different colored section details the team members responsible.

For the website and mobile application, Brandon Mitchell and Don Ellington will be the primary team members. For integration into the microcontroller, Adam Loree and Jonathan Rodriguez will be the primary team members. Possible overlap will occur as both sections need to be seamlessly integrated.

5.5.1 Software Development

This following section will cover the overall software design of the system by splitting the software into the following sections: webstack, web server, database, user applications, mobile app, and website. All of the sections will have a more detailed plan, considerations, and designs to give a better overview for the software aspect of our project. Depending on constraints, our intended design of the system is subject to change based on how the confirmed software components are coded and designed.

5.5.2 Web Stack

We wanted to have a dynamic application that could be accessed from mobile or a website. The website and mobile app will be able to create a unique user account allowing the user to search recipes with the ingredients grown and also access sensor data history from the system. To realise these functional goals we had to consider a web stack that would suit our project and give us a framework to develop our applications. We had many considerations on what configuration of web stacks would be appropriate to handle our project while being streamlined to learn and implement. For our group researching LAMP vs the MERN, MEVN, and MEAN stacks, we preferred the standardized language of Javascript that the LAMP stack didn't have uniform throughout its components. **Table 5-1** summarizes the web stack considerations [32],

Webstack Considerations		
Attributes	**Lamp**	**MERN, MEVN, MEAN**
Learning Curve	Requires multiple coding scripts and languages, but structure is easy to learn.	Only need the JavaScript programming language to learn with pleasant front-end frameworks
Scalability	MySQL - Relational data not needed	NoSQL - non relational data with documents storage applies to sensor data within project

Table 5-1: Webstack Investigation

Webstack Considerations (cont'd)		
Attributes	**Lamp**	**MERN, MEVN, MEAN**
Development Time	Apache web servers and PHP both require initial setup and management	Node.js webserver and front end frameworks intuitive and less initial setup
Documentation and Libraries	Wide variety of support and vast quantity of library modules	Open source with large development community

Table 5-2: Webstack Investigation(2)

Out of the three choices for the front-end framework of the MERN, MEVN, and MEAN stacks, we chose to use MERN with React because we've had the most experience developing with the framework compared to Vue and Agile. The mobile application will be developed with React Native that is a well developed component of the React software development process. Since our project will be using a MERN stack. [1]This stack comprises MongoDB, Express, React and Node.js. Below in **Figure 5-19** the webstack we will employ is pictured.

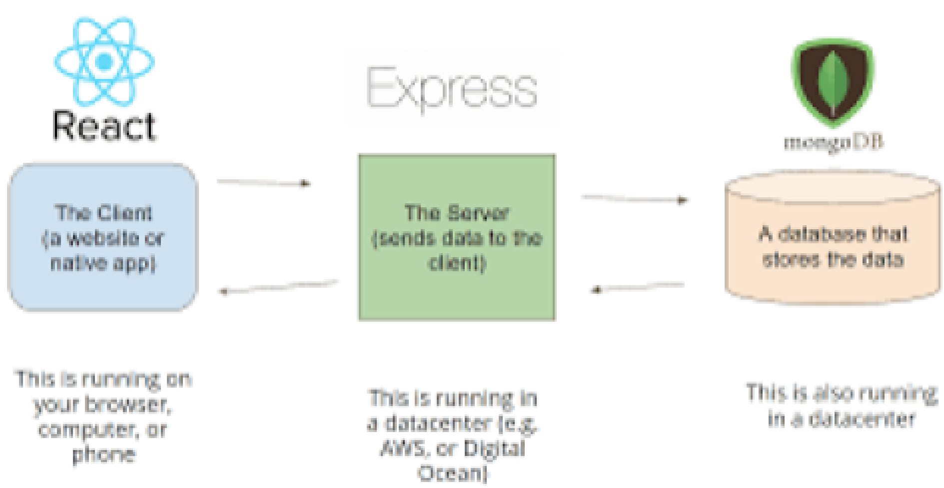


Figure 5-19: WebStack

Our choice with using the React front-end framework is because it has a [12]JavaScript library that offers great utility for building user interfaces. React has the ability to handle rapidly changing data which is great for when users are accessing hydroponic sensor information. A part of the React framework is React Native which is an open-source mobile application framework created by Facebook. This framework has also been used to develop applications for Android, Android TV iOS, macOS. Our group members have experience in these development environments as well. Express.js will be the server-side framework and the web server is Node.js which provides a JavaScript Environment which allows us to run code on a server outside the browser.

5.5.3 Web Server

Continuing from the previous section and more in depth on the topic of web server, Node.js was also an advantageous choice when deciding between the LAMP and MERN stacks. The Node.js web server is already familiar to our group from previous courses with less learning required compared to the Apache HTTP server. While also using the standardized language of JavaScript for development. [10] JavaScript is actually one of the most prevalent coding languages across the entire software development world. The javascript based web server, Node.js, creates active web pages and database accessing. Also, since Node.js is a JavaScript runtime environment which allows the infrastructure to build and run an application. The event-driven I/O model that Node.js employs increases the efficiency and makes a scalable network application. The aspect of the web server being asynchronous means a Node.js based server never waits for an API to return data allowing for live use of applications. Overall, it's a light, scalable, and a cross-platform way to execute code.

To connect to the online server, our hydroponics system will be able to be interacted with by the user through an android device or computer. Bluetooth and Wi-Fi modules from the control unit will be able to connect to the plant unit and other devices while on the same network. The Wi-Fi module can produce a web server of its own allowing devices to connect to it and access features. The Wi-Fi and Bluetooth modules will be an initial step in connecting the hardware and software together from the plant and control units to the website and mobile app. With the web server set up, the user's device can be connected. From there the hydroponics system can send and receive information from the online webserver and database. The user will also then be able to check on the status of the plant unit and control unit.

5.5.4 Database

The database for this system will be written in NoSQL.We will try to use a personal computer to store the information from the user and central unit. The use of MongoDB will allow our group to store and load data from a database. This database will be simple in its design and the group will try to remove any third-party processes that may be needed to maintain data verification. Below is **Figure 5-20** showing a basic database entity relation diagram;

User	
Data	Page Key
Name	
Email	
Password	
ID	

Sensors	
Data	Page key
pH	
tds	
temperature	
light_level	
water_level	
humidity	

Figure 5-20: Database ERD Diagram

The diagram is simple and can be applied to multiple pods given more time to develop the project. Once more work is done for the database this table will be expanded. The database will have the potential to be scaled to have multiple profiles and multiple pods to monitor the system. For now, we are sticking to only one profile and Garden Pod. The database should be fast to search through and query between the data tables for the Garden Pod data to be updated. The current Garden Pod data from the control unit can be found to provide the user with more information.

We found MongoDB to be attractive because of its NoSQL database. The document storage type is also chosen because each record is a document consisting of key-value pairs that are similar to JSON (JavaScript Object Notation) objects. Within MongoDB we can utilize it’s flexibility to create schema, databases, tables, to store our user and sensor data. Documents that are identifiable by a primary key make up the basic unit of MongoDB and the Mongo shell allows us to use the JavaScript interface to carry out document operations, like querying and updating records. Another aspect of the MongoDB we thought would be useful was the scalability of the database because there will be a lot of information gathered and stored from the Garden Pods that need to be accessed in bulk and quickly. The speed of this database is due to it being document-oriented allowing for easier indexing. Overall, this database provides great functionality to our system’s purposes.

5.5.4.1 MongoDB Realm

Incorporating MongoDB Realm with the mobile application has been a large consideration for the reason to use MongoDB. This service MongoDB provides for the synchronization of data between iOS and Android if we were to scale the app functionality for iOS devices. Currently, developing for Android first is the main priority in order to have at least one functioning mobile application. Below in **Figure 5-21** describes how MongoDB Realm manages data from the database between the local databases of the mobile applications for both iOS and Android.

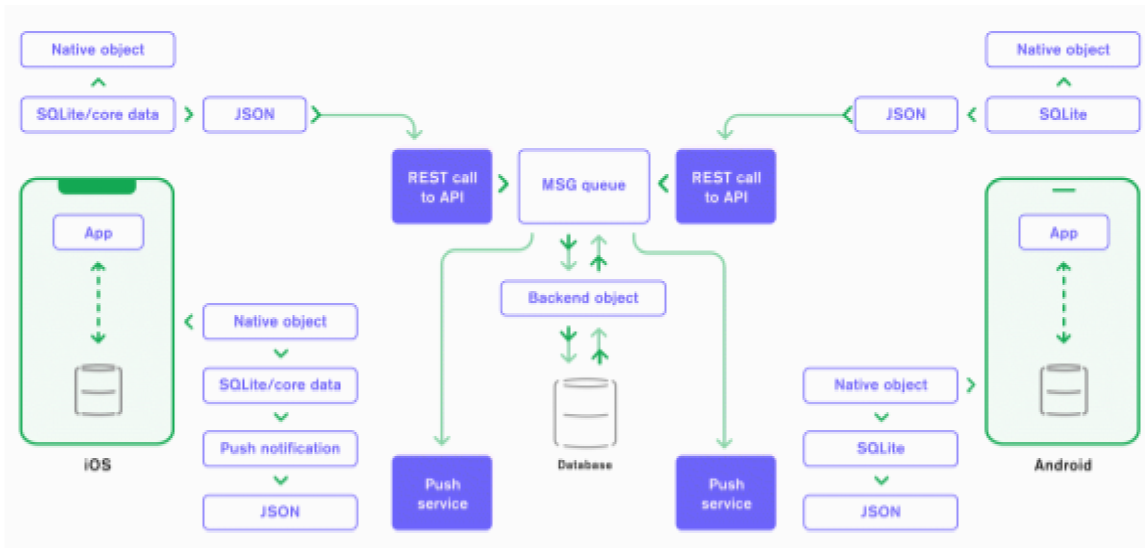


Figure 5-21: MongoDB Realm Data Management

The potential for our mobile applications to be seamlessly cross platform and efficient gave increased importance in the consideration of Realm. Also, the documentation, tutorials, and overall support for Realm motivated us to utilize this service that was already attached to our database. **Table 5-3** below describes the features of Realm that we found useful.

Function	**Description**
Serverless	Server-side applications do not need to set up and manage server infrastructure. Items including provisioning, deployment, operating systems, web servers, logging, backups, and redundancy. Functions help the cloud-based Realm systems that can calculate data or interact with third-party services Less API writing on top of the MongoDB Realm service. Any authenticated users or clients have direct access to modify parts of the database based on the permissions.

Table 5-3: Realm Features

Function	**Description**
Dynamic	Allows updates to data changes in the main MongoDB service, Atlas, where collections of data are managed. Data from incoming webhooks can be processed. Realm Functions in combination with Realm triggers allow for scheduled data processing.
Security	Users with previous authentication providers are able to log in due to them being built directly into MongoDB realm. Example, Google logins, Twitter, Facebook, and more. Realm provides Schema with validation logic that enforces data integrity to prevent mismatching storage of data documents.

Table 5-4: Realm Features (2)

5.5.5 User Applications

The user will have access to web applications that provide access to information regarding the user's hydroponics system. These will consist of a website and mobile application both supported by a database and server. The website and mobile application will be similar in function, but the website may have extra security or information features to improve the user's experience. The web server created with Node.js will allow the user to communicate through the applications on what data is to be requested or sent. The UI for the applications will be simple yet professional with colors that correspond to either the UCF colors or nature oriented.

5.5.5.1 Mobile App

Our mobile device application will be similar to other hydroponics systems apps with basic features, but will have extra quality of life features to provide a better user experience.

The hydroponics apps we've researched focus on either planning plant growth cycles or real-time sensor data of the plants. Our app design will focus on real-time sensor data on the plant's environment relating to its health while adding a feature to utilize and plan the use of grown hydroponic plants afterwards. This is an increase in scope from the initial design, but it allows the entire system to be more engaging for the user without adding too much software development work. The name we are going to use to describe the app will be Hydroponics Helper to be similar yet different to the website. **Figure 5-22** below describes the flowchart of the mobile application.

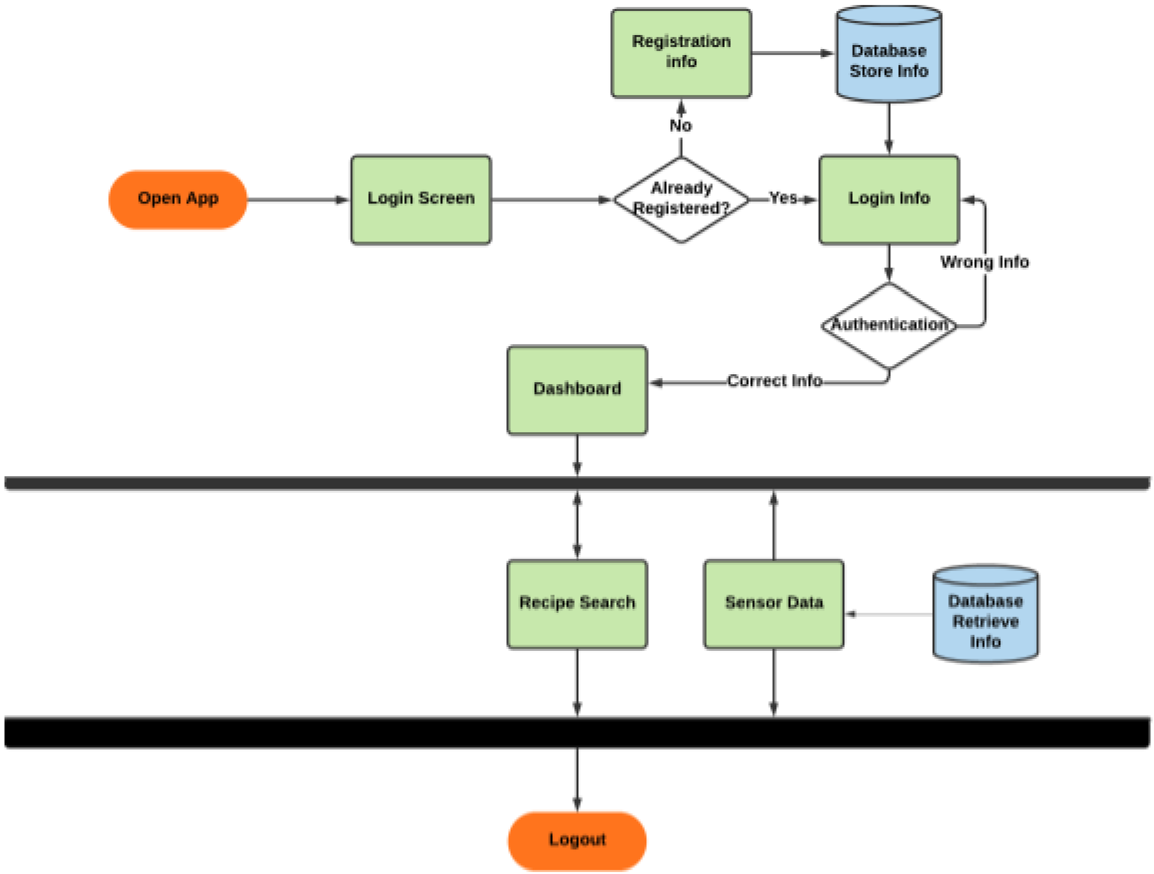


Figure 5-22: Mobile Application Flowchart

Our group is mainly computer engineers so we have experience working with web app development software. The main operating systems for phone apps today are iOS and Android. Also, we all have smartphones of those types that we would be able to interact with our system with. The operating system we settled on designing for was Android because it is most comfortable for us to design with and a prototype would be reasonable. Our group discussed other software platforms to assist in the structuring and development of this mobile application, but they didn't offer us the functionalities we needed. Our group has had experience with this software with previous computer science courses with many assignments being to develop mobile apps within the environment. As part of the web stack development, the front-end framework React has a component called React Native that will allow us to develop the mobile application.

We are designing the mobile application to have a front page that will have a login page, and registration page. A welcome screen is planned when the mobile application is first accessed. The login page will contain fields for the user to fill out to authenticate their identity. Navigation to the registration page if the user is not registered is a feature to be implemented. From the login page, there will be a user dashboard where the user can access the Garden Pod sensor data and search recipes with the plant as the plant is being grown. The Garden Pod sensor data page will pull information from the database and display it for the user in a format that is easy to read and understandable. The sensor data will also indicate if the Garden Pod needs resupply. Security is also a consideration when designing the user applications to ensure user safety and maintaining the integrity of the database. The control unit with the Wi-Fi module will transmit sensor data to the user applications through a local network and internet from the Garden Pod sensors. Connecting to the database will update all the information from the control unit which gets its information from the sensors on the Garden Pod.

The development of the mobile app will be with React Native and the Android Studio emulator. Our group has used Android Studio in the past with great success and is the [11]official fully integrated software development environment (IDE) for Google's Android software development. This development environment was built on JetBrains' IntelliJ IDEA software and designed specifically for Android development. The other great feature for the Android Studio development environment is that it allows the emulation of apps made within or in other coding editors that project onto a virtual phone. The phones emulated are also semi-functional and the phones span a wide variety of series to be tested with. Android Studio also makes use of a combination of technologies for the development of Android mobile apps. It combines a good amount of software and coding languages including Java, eXtensible Markup Language (XML), Gradle, and more. Overall, we had plenty of reasons to go with Android Studio to create a smooth software development process.

5.5.5.2 Website

The name of the website portion will be Hydroponics Guider, but this is subject to change. The website will have a few pages consisting of many pages namely, the home, about, login, register, pod sensor data, recipe searching, and settings. There will be a side menu bar and a home page button across the top for the user to access different web pages with important information. The goal is to have the website be familiar based on common website layouts on the internet and be simplistic in its design. None of the features should feel out of place and have a use to the user at some point. Below in **Figure 5-23** describes the flowchart of the website.

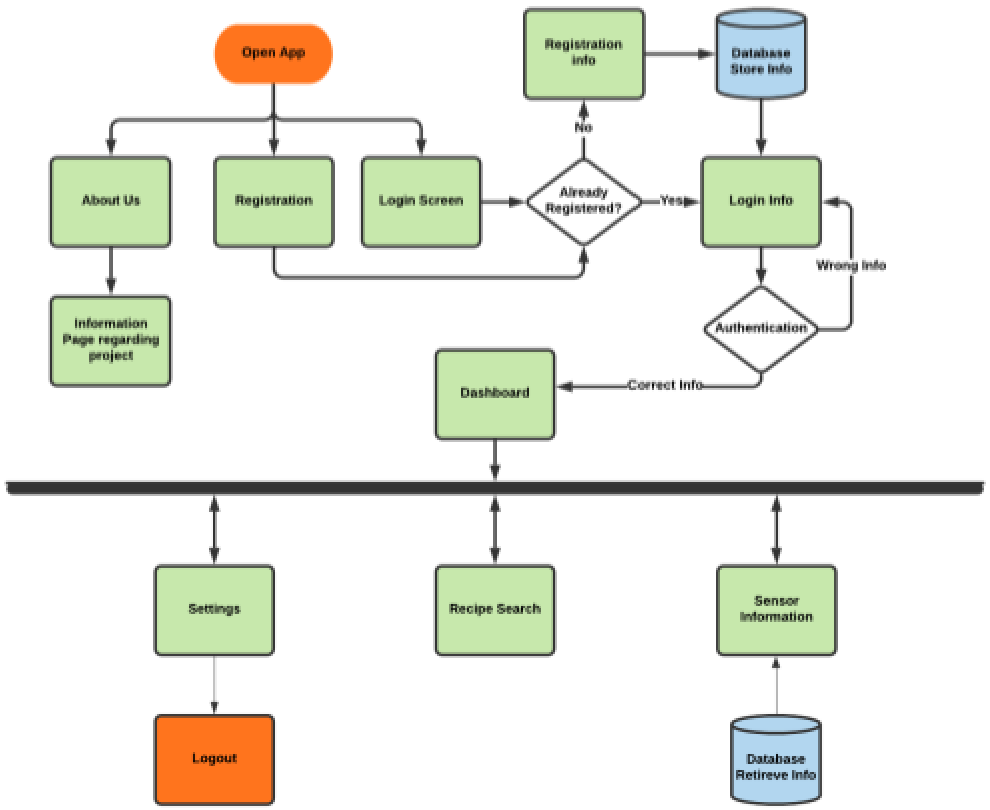


Figure 5-23: Website Flowchart

The mobile and website applications will be similar in a variety of ways. The website will provide more information about the project and systems involved while the mobile application will be designed to be easier to navigate and access Garden Pod sensor data. The website will be written in JavaScript with the front-end framework React.

We are designing the mobile application and website to have a front page that will have a login page, and registration page. However, only the website will have an About Us page. A welcome screen is planned when the website and mobile application are first accessed. The login page will contain fields for the user to fill out to authenticate their identity. Also, both applications will have navigation to the registration page if the user is not registered. The about us page will have information about the hydroponics system and our team members. From the login page, there will be a user dashboard where the user can access the account settings, sensor data, and search recipes with the plant as the ingredient being grown. The sensor data page will pull information from the database and display it for the user in a format that is useful. The sensor data will also indicate if the Garden Pod or Central Unit needs resupply. The control unit with the Wi-Fi module will transmit sensor data to the website through a local network and internet from the Garden Pod and Central Unit sensors. Connecting to the database will update all the information from the control unit which gets its information from the sensors on the Garden Pod and Central Unit.

5.5.5.3 Page Functions Summary

Table 5-2 summarizes the minimum design and software page requirements we will design for the project. The website's only page is the About Us page. This is designed so that the mobile application can focus on simplicity and the basic information.

Pages	Description
Login	Provides user with email/username and password fields to login to hydroponics account
Registration	User will be able to input email, password, first name, and last name to be stored within the database
Dashboard	A general welcome screen to give the user access to the essential pages. The settings, Garden Pod data, and recipe searching functions will be available to be navigated too
Settings	A page where the user will be able to change their password, logout, or delete their account
Recipes	This page will allow the user to plan out what to use the plant they've grown in dishes that they will be able to search
Garden Pod data	Enable the user to modify the Garden Pod settings and view sensor data to make decisions on resupplying the pod or harvest the plant
About Us Page	This page will only be used as an informational page regarding the project and offer instructions to the user when issues arise.

Table 5-5: Minimum Design/Software Requirements

The goal of the website and mobile application is to make the logic and programming simple from the hydroponic system microcontroller central unit and Garden Pod sensors to the user applications and database. The computer processing and logic of maintaining the Garden Pod will be the responsibility of the user and the hydroponic system microcontrollers.

6 Project Prototype Construction and Coding

This chapter will discuss the prototype construction and coding methodology our group has developed for realizing our hydroponics system. These sections will go in depth about the components of the hardware and software systems. In conjunction with testing, this will be an ongoing challenge that will be crucial to ensure our project meets the requirements set forth. The prototype is subject to change when official building begins and when constraints are encountered. Progressive prototypes will be documented and built upon to incrementally finish a final product.

6.1 Integrated Schematics

There will only be one PCB printed for the Garden Pod, so the schematic only displays the electronics on the garden pod but we will cover all the electronic systems in this section.

This project is designed to have two separate, yet harmonious devices, as described in previous sections. While these two units are not physically attached. They will communicate wirelessly and one of the units will be able to charge the other. The schematics reflect that and were designed separately. The Central Unit, or Power Pod, schematic will include a microcontroller unit, with several ports to connect the wifi module, bluetooth module, pH sensor, TDS sensor, Temperature/Humidity sensor, water level sensor, main irrigation pump, peristaltic pumps, and ultraviolet sterilization light, to the power supply and microcontroller. This system will ensure the water inside the tank is sterile and the nutrients, as well as pH levels are properly balanced.

The Garden Pod PCB will have a separate microcontroller which will feature ports for the UV LED grow light, axial fan, and another set of sensors, but will not be equipped with a wifi module as it will only need to communicate with the Central Unit. As shown in the figure below, The Garden Pod will not be directly connected to line power, but will be supplied by a 12 volt battery. The purpose of this design is to promote maximum vegetation, while minimizing power consumption.

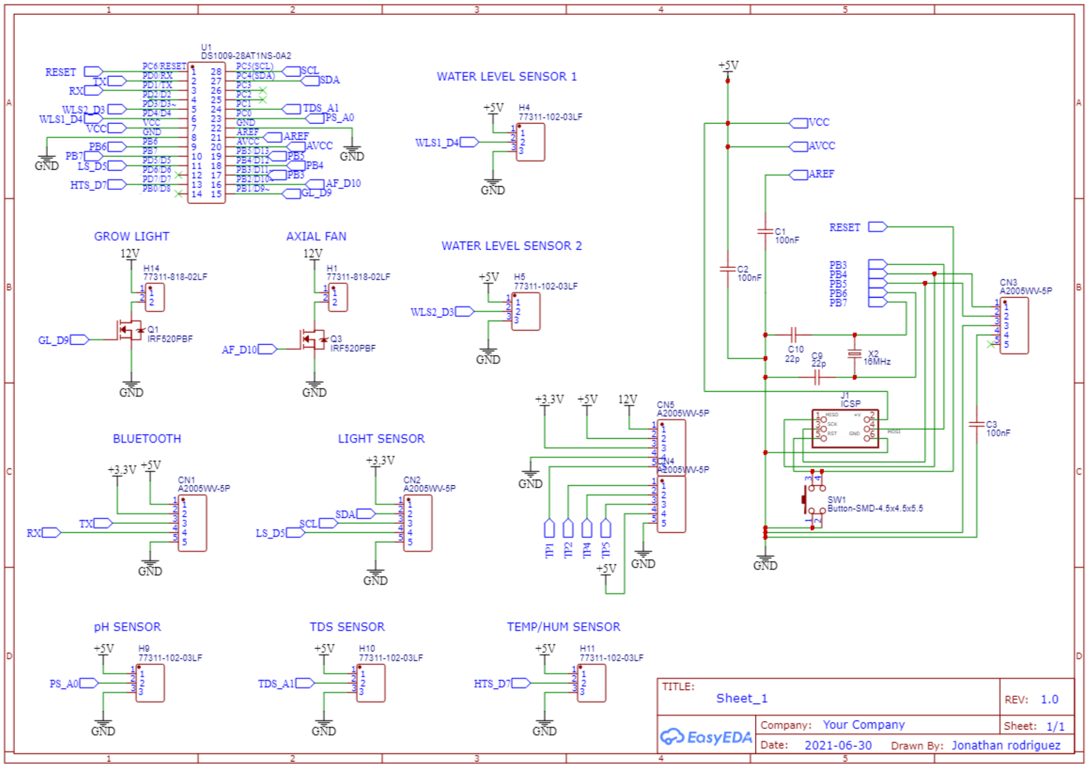


Figure 6-1: Garden Pod PCB Schematic

6.1.1 Central Pod Power System

The central unit will be fully supplied by line power, hence will comprise of an AC to DC adapter followed by DC to DC voltage regulators to power the DC pumps and Raspberry Pi. As mentioned above, The AC-to-DC converter we have selected will be rated at 12V/6A. The DC voltage is either 12 or 5 volts.

6.1.1 Garden Pod Power System

The peripheral garden pod can be charged by AC/DC adapter but will be battery operated when removed from the central unit. The battery will be rated at 12 volts, the Amp-hours capacity of the battery is yet to be determined but our goal is to achieve a battery of 24 hours or greater. The 12 volts will power the grow lights and axial fan, however we will need to step-down the voltage if the microcontroller can not supply sufficient current to the sensor.

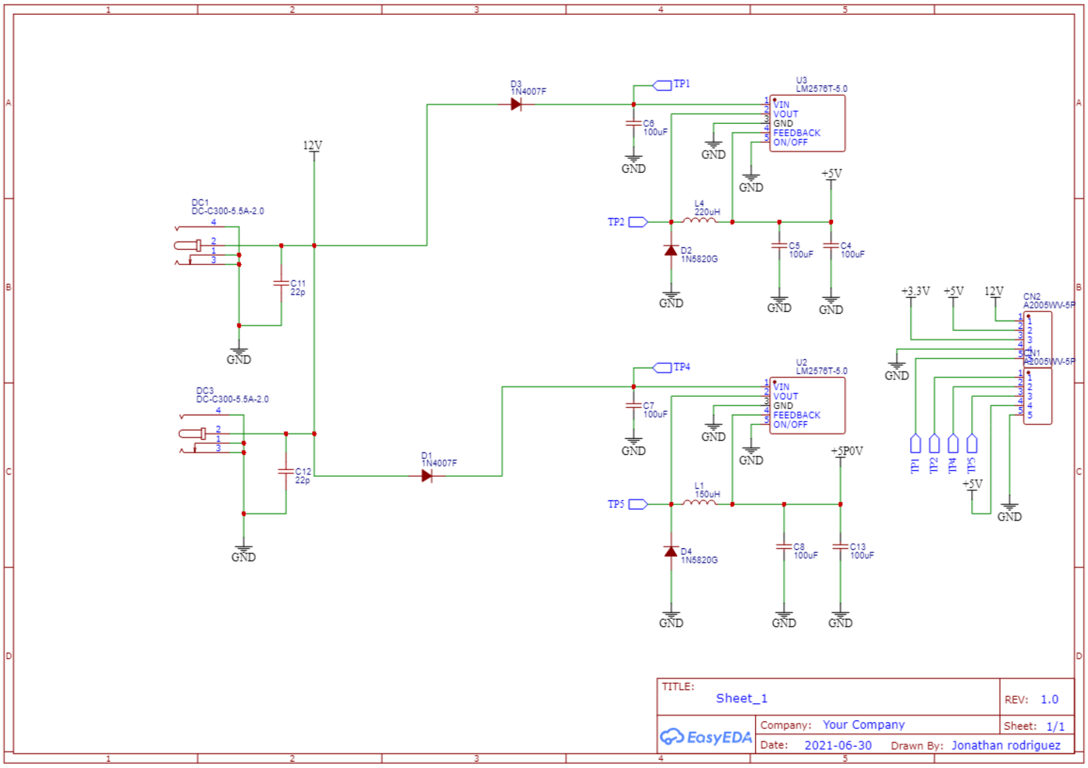


Figure 6-2: Garden Pod Power

The DC power jack is intended for a 12V input. The input power source will either be a 12V 6800mAH battery or 12V AC/DC adapter. A rectifier diode will limit the input current to 1 Amp. The LM2576 will be used as a 5V buck-converter, with a max of 3A. For testing purposes, we experimented with two voltage regulators with different inductor and capacitor values. The voltage regulator on the top is capable of higher amperage.

Since the voltage regulator is separate from the microcontroller, the voltage regulator will have a set of output pins and the microcontroller pcb will have a set of input pins.

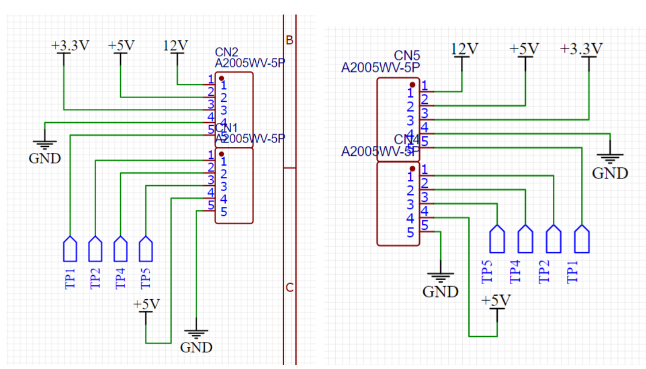


Figure 6-3: Microcontroller PCB Pins

6.1.2 Irrigation

The main water pump will be controlled through the microcontroller with the help of logic level mosfets. The IRF520 mosfet to be used in this design has a trigger voltage rated at 3 volts. With the mosfet, the microcontroller could activate the pump with an io pin at logic level voltage, While the AC to DC converter provides sufficient power to drive the device.

6.1.3 PH Sensor Pinout

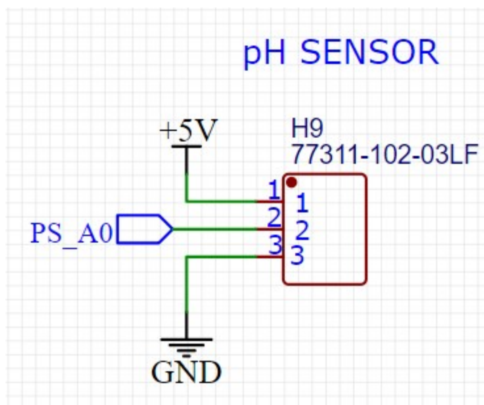


Figure 6-4: pH Sensor Pinout

Figure 6-4 shows how the pH sensor will be incorporated into the hydroponics system. The 5V power coming from the arduino will be directed into the pH sensor to power it and the analog input on the arduino will be used to catalog and record the pH sensor's readings. This will make our design compact but this does not come without cost. The issues with powering the pH sensor straight from the arduino could mean that if enough things were powered from the arduino, it could fry the internals of the arduino. This problem is discussed further in the hardware specific testing section.

6.1.4 Temperature/ Humidity Sensor

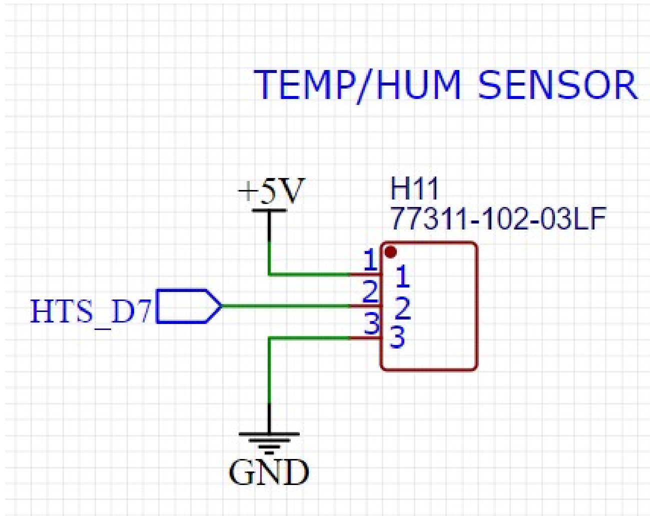


Figure 6-5: Temperature/Humidity Sensor Pinout

As shown in **Figure 6-5** the humidity/temperature sensor is connected to voltage supply, VDD. The supply voltage can range from 2.7 to 5.5V. In this case, the sensor will be connected to a 5V source and the common ground. The arduino could supply the 5V source, but we want to be cautious of overloading the arduino. The Humidity/Temperature sensor communicates over I^2C protocol and must be connected to the Serial data (SDA) and Serial clock lines (SCL). According to the data sheet these lines are open-drain lines which must be equipped with a pull-up resistor. The device will measure both proximate moisture and temperature, and has a voltage supply monitoring functionality which could be used for the battery-operated pod to alert the system when the battery needs to be charged.

6.1.5 Non-contact Water level sensor

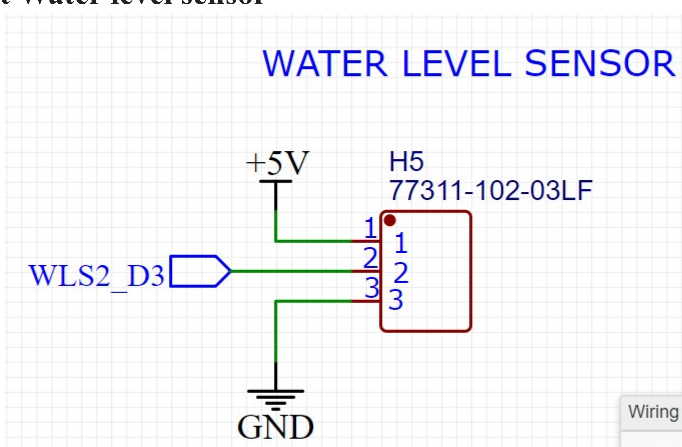


Figure 6-6: Water Level Sensor Pinout

Figure 6-6 shows the pinout for the water level sensor. VDD is the 5V voltage supplied by the arduino. The water level will monitor when the water has reached a certain fill line. The output of this device will be a digital signal, so it will be connected to an io pin on the ATMEGA.

6.1.6 Total Dissolved Solids (TDS) sensor

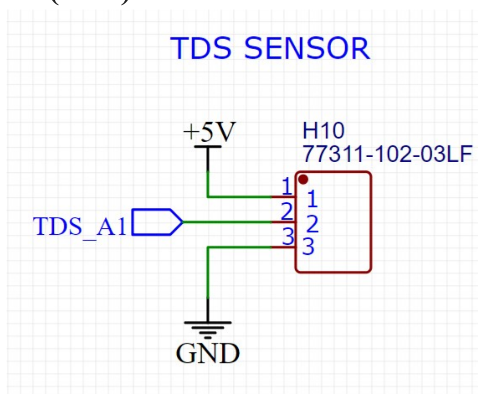


Figure 6-7: TDS Sensor Pinout

Figure 6-7 is the pinout of the Total Dissolved sensor which will be connected in series with an analog isolator. The sensor originally outputs an analog signal but as explained in **section 6.1.3.2**, since there will be multiple sensors submerged in the water, the manufacturer recommends we attach the analog isolator and use the digital signal instead.

6.1.7 Peristaltic dosing pumps

The Peristaltic Pumps function on 12 DC volts. There will be three pumps used to maintain water quality which will operate automatically depending on pH and TDS levels. We will achieve this by using logic-level mosfets, which can be toggled by the microcontroller. The microcontroller will activate the transistor through digital pins, allowing current to flow across the pump.

6.1.8 Central Unit Microcontroller

The Central Unit will use a Raspberry Pi as its microcontroller. This Microcontroller is compatible with input voltages from 7 to 12V. The input voltage pin will be connected to the AC/DC adapter and provide voltages to the electronics that will need 3V and 5V.

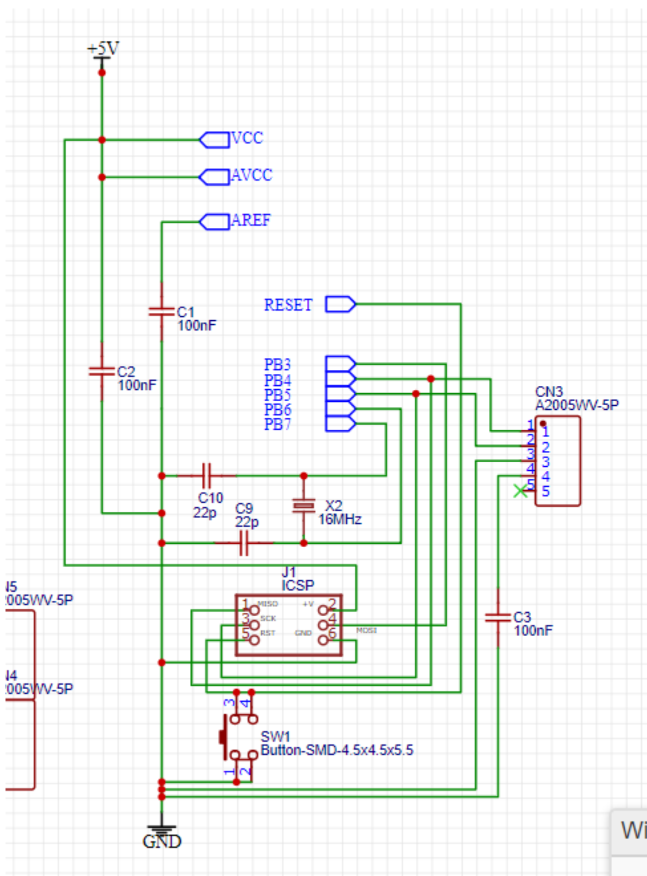
The analog pins will be connected to the humidity sensor. The humidity sensor will need the analog pins which can provide the serial clock and serial data line.

The rest of the sensor will be connected to digital pins, set to output, to control their respective devices. The Bluetooth and Wifi modules each require a pair of TXD/RXD pins. The microcontroller only has one hardware pair of the transmit/receive pins, however using the SoftwareSerial library, we can use any Digital pins as RX/TX.

6.1.9 Garden Pod Microcontroller

The Garden Pod will exclude some of the previously mentioned devices but will include an axial fan, grow light, and light sensor. Any 12V devices will be connected to the microcontrollers through mosfets. We initially intended on using a relay module to operate the grow light but, depending on the power dissipation of the grow light, we considered the more energy efficient options.

At the heart of the Garden Pod PCB there is an IC socket which the ATMEGA382P-PU will plug into. The circuit has capacitors and an oscillator on board to accommodate the microcontroller. We also installed an ICSP interface for programming and a reset button. Twenty-four of the pins will be connected to pin-headers to be attached to the peripheral electronics. The ATMEGA382P-PU will need an input voltage of 5V.



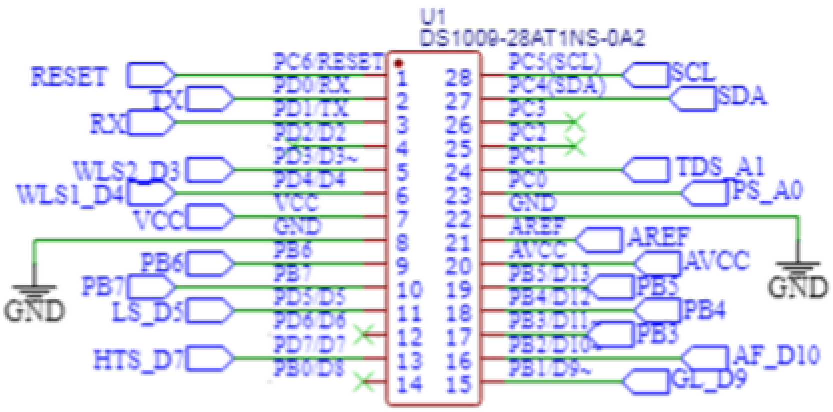


Figure 6-8: Garden Pod Pinout

6.1.10 Wireless communication modules

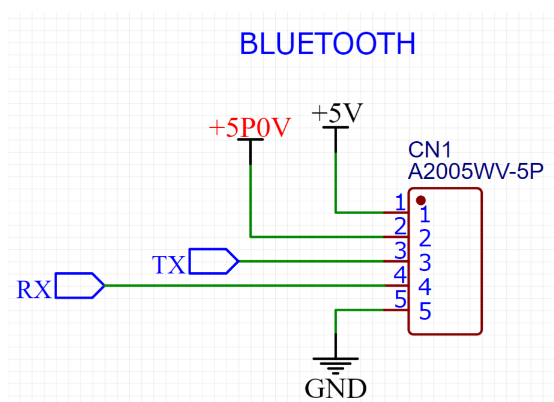


Figure 6-9: Bluetooth Module Pinout

The PCB will have four ports for the wifi module, both will be connected to the TX/RX pins and 3V pin of the microcontroller.

6.1.11 LED Grow Light

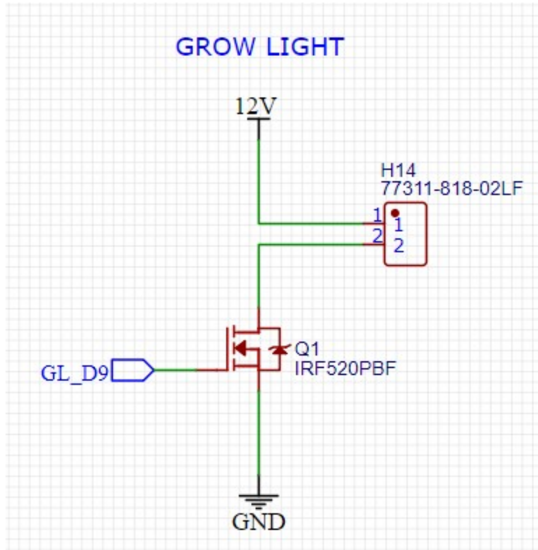


Figure 6-10: Grow Light Pinout

Figure 6-10 is the schematic design for the grow light system. The UV LED array will be connected to the pin headers and the mosfet will activate the current through the device.

6.1.12 Axial Fan

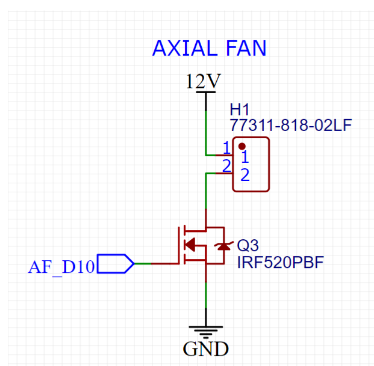


Figure 6-11: Axial Fan Pinout

Figure 6-11 is the schematic design for the Axial Fan. The Axial Fan will be connected to the pin headers and the mosfet will activate the current through the device when the gate voltage is active.

6.2 PCB Vendor and Assembly

To print a PCB a gerber file is first required. To create a gerber file we will be using PCB design software. There are many different types of PCB design software available for use. We will be using software called EAGLE for our PCB design. Using EAGLE we are able to make a 2-layer PCB with relative ease. EAGLE was taught to all of us in the junior design class so it is the most available to us and the most useful in my opinion. The software is relatively easy to use and very transferrable.

To use this software we start out by designing the actual circuit in EAGLE. We will model the circuitry involved and how it all connects together. Then we will use airtraces to connect all the components necessary in the circuit. Once all the components have been linked together we can then de-tangle the wires that are used in the circuitry. EAGLE uses a system for automatic routing of the wires to find close to the most optimal position of all the wires in the circuit. With all the wires routed in the proper place the user then goes through manually and checks all of the individual wires routed in the circuit before physically routing them on the PCB. The user goes through and routes all of the wires in the proper place and layer and ensures that they are all in the bounds of the PCB. The last step is to export the EAGLE project into a gerber file to be understood by the printer.

The design for this hydroponics system will have more than one PCB. At the controller unit therCreating a PCB is a very efficient way to put together all the components and wiring to create a unified circuit. The PCB eliminates the mess of wires that usually hold together circuitry and ties an electrical circuit together. With the involvement of water in our project we have to be extremely delicate with the circuitry involved. A PCB is useful to ensure the neatness of our project and to hopefully keep the water away from the circuits. Not only will it keep our project neat but it will also make the design look better in general. Some PCBs even have cool designs that we would be able to use. The PCB design process is not that complex. The first step is designing the circuitry and testing it physically. This process is a little more complex but must be done before PCB design. The next step is to design the PCB online and to check it to make sure it works correctly. We then need to test it once it has been printed to make sure it is functioning correctly. Most likely it will not be so then there is debugging, redesigning, and reprinting.

When designing PCBs there are many different things to take into account. Firstly one must look at the physical PCB design. The PCB must have traces a certain distance apart to prevent short circuiting. These traces also must be a certain width and the length of these traces must be taken into account. The wider the traces the more the resistance is affected and the more real estate on the actual PCB is taken up. PCBs are created in high detail and when a PCB is designed the traces must be designed with the actual size of the components in mind. Even if the traces are the right size and length there could be conflict with the actual electrical components mounted on top of the PCB. With all of these things in mind there are quite a few different variables to consider when designing the PCB. Because of this it can take weeks to fully design, test and get a PCB correct.
e will be a PCB firstly for converting AC to DC power. This PCB will be in charge of converting the 120V AC power to 12V DC power. This power will be used to control the wifi and Bluetooth modules. The next PCBs used in the projects will be located in the modules where the actual plants are grown. A PCB will be used to convert 12V to 5V power to power the microcontroller and various sensors. There may need to be more

conversions to power our sensors separately since there will need to be a Bluetooth module as well as other sensors powered by the microcontroller. If there is too much power running through the microcontroller it could cause it to break down and melt.

After designing the PCB it takes time to print and test the PCB. The faster the PCB is printed and mailed the more it costs the user so it makes sense that we would want to order our PCB early so we have ample time to test and redesign if necessary. To get an accurate source of what our PCB is going to look like and where we are going to get it from we need to compare prices of online retailers. To do this we look at different sources and compare the price and required number of orders for each board. We are comparing based on a board of 100mm x 100mm with 2 layers.

Vendor	Website	Min. Order	Cost p. PCB
jlcpcb	https://jlcpcb.com/	1	$2
Pcbway	https://www.pcbway.com/	1	$5
Pcbastore	https://www.pcbastore.com/	1	$5
allpcb	https://www.allpcb.com/	5	$5 ($0 w/ discount)
Pcbgogo	https://www.pcbgogo.com/	5	$5
7pcb	https://www.7pcb.com/	3	$55

Table 6-1: PCB Vendor Investigation

Table 6-1 shows different options of PCB vendors that we have considered. When ordering PCBs one must take into account not only the cost per PCB but also the minimum order number. By analyzing both of those numbers we are able to figure out which pcb vendor is the best to go to.

The first iteration of the PCB will be a standard through hole design until we have a proof of concept. After we successfully print these circuit boards and show that they are effective we will transfer to a surface mount design which will be a more professional approach. The surface mount makes a smaller and more compact design on the PCB to give our project a smaller form factor and sleeker design.

6.3 Final Coding Plan

The final coding plan will give an overview of the subsystems and how they connect to form the overall hydroponics system. This section will go in depth about the mobile and web applications, database, microcontroller, and sensors. Having the website the only access to the database is a consideration that would give users a reason to use the website.

The system flow chart visualizes how the user access to the system can be obtained from the website on a computer or on a mobile device with the appropriate application. The mobile and website will have similar functionality in viewing Garden Pod sensor data and connecting to the Wi-Fi or BlueTooth module on the microcontroller.

After the initial registration and Garden Pod setup, the control unit and Garden Pod will communicate with each other to create a growing environment for the plant. All the user will need to do is keep the Garden Pod resupplied by observing the plant status. An example of the overall system is shown in figure below.

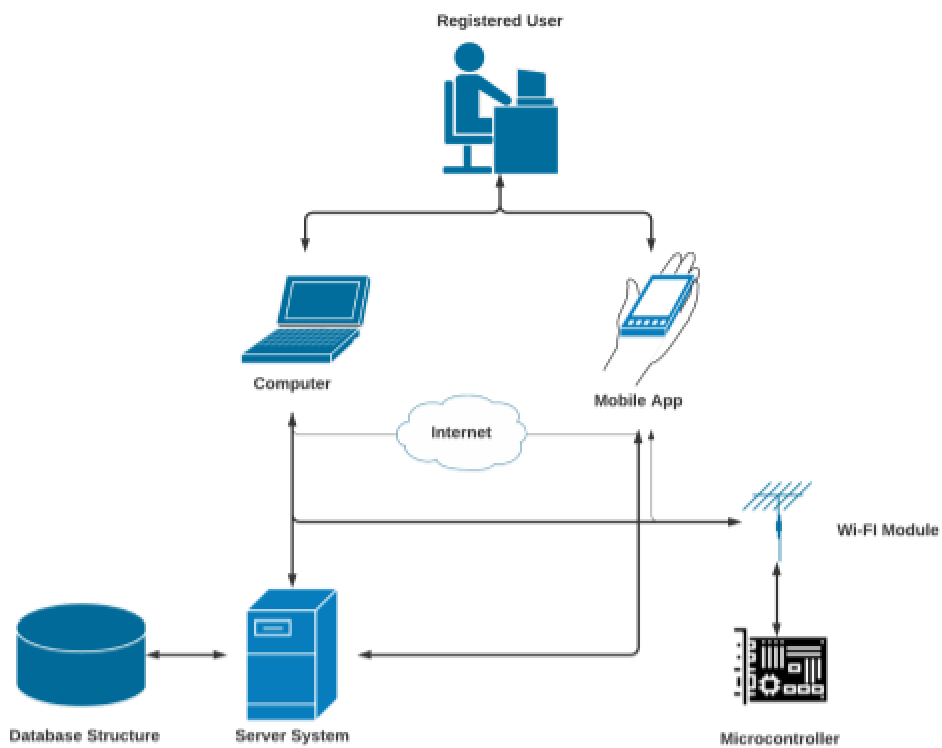


Figure 6-12: User Flowchart

The next flowchart describes the control unit and plant unit interactions. The Garden Pod unit will connect with the control unit and relay important sensor data information for the user to act with. The plant unit is intended to be modular and have the ability to be detached from the main control unit that stores nutrients, pumps, and the overall resupply system. The Garden Pod and control unit will periodically exchange information between each other and update the database when there is a wireless connection to the internet available.

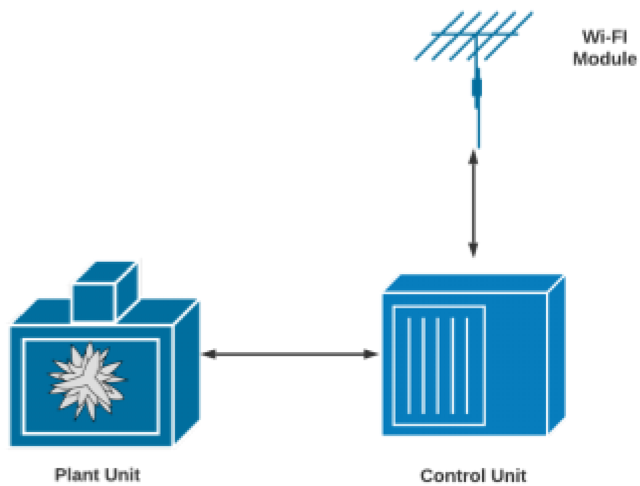


Figure 6-13: WiFi Communication

The Wi-Fi module will pass on all the Garden Pod sensor data from the control unit to the website and mobile applications. The plant unit will be able to communicate to the control unit while not connected physically then the control unit will store and pass on the sensor data when requested.

6.3.1 Mobile Application

The final coding plan of the mobile application will be discussed in this section with the flow diagram of the screens below. The mobile application will allow the user to access all of the functionalities the website has. This is still a work in progress, but this figure will be the basic structure for our app design. Below is a flowchart of the screens for the mobile app.

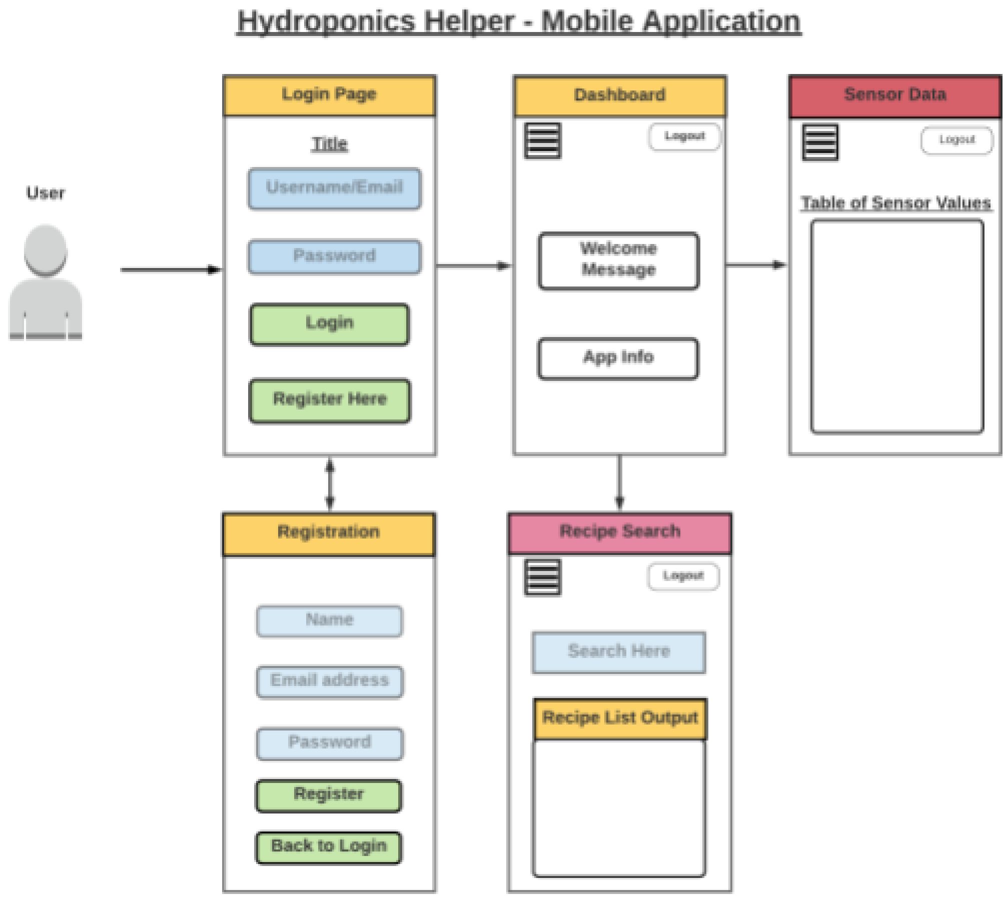


Figure 6-14: Mobile Application Final Coding Plan

The name of the app will be Hydroponics Helper The first page seen by the user is the login page. The user will be able to login or register as a new user. If the user is not registered, they will be redirected to the registration page which will give the user options to input their personal information for the database to store and authenticate later. After entering their email and password information then clicking log in, the user's information will be compared with the database for authentication. If the information isn't found the user will be prompted to try again. Correct credentials will allow the user to enter into their account and be greeted with the dashboard. The mobile application is meant to be more simple so the about us page from the website was left out intentionally.

The dashboard will provide the user with options to access all parts of their account. The navigation will be similar to the website in that there will be a sidebar menu allowing access to the different screens of the application based on whether the user is authenticated or not. The user will also be able to activate navigation by simply swiping from one direction instead of pressing the sidebar menu button. We didn't include the recipe search in the dashboard because we wanted the first screen after the login to not overwhelm the user.

The recipe search functions the exact same as the website, but will be accessed via the navigation bar on an icon. Logging out of the mobile app is always available when authenticated so there wasn't a need for a settings page.

The Sensor Data page will display all the information gathered from the Control Unit and Garden Pod when there is internet access. The data will be displayed in a tabular format with the last known measurements.

If the project was given more time, the user would be able to preview past sensor data and create graphs to chart the information. If given more time to develop multiple Garden Pods then we would include unique attributes to the pods including name, plant type, and more. Ideally, the Garden Pod's name will be listed above the sensor data for the user to know what plant's pod data it is.

6.3.2 Website

The final coding plan of the website will be discussed in this section with the flow diagram of the website pages below. The website will allow the user to access all of the functionalities of the mobile application including but not limited to sensor data viewing, account creation, and recipe searching. This is still a work in progress, but this figure will be the basic structure for our app design. We are choosing the name Hydroponics Guider for the website, but it is tentative to change based on group discussion.

The first page seen by the user is the welcome page that will showcase some hydroponic inspired images, a message with the website name, and a navigation bar on top. The navigation bar will be consistent in location throughout the entire website pages, but the redirecting information will change. The About Us page will redirect the user to get more information about the product and our group.

From the login page the user will be able to login or register as a new user. If the user is not registered, they will be redirected to the registration page which will give the user options to input their personal information for the database to store and authenticate. After entering their email and password information then clicking log in, the user's information will be compared with the database for authentication. If the information isn't found the user will be prompted to try again. Correct credentials will allow the user to enter into their account and be greeted with the dashboard.

The dashboard will provide the user with options to access all parts of the website. The recipe search function will also be shown for the user to quickly access information on dishes. Clicking on the settings button on the side menu icon will send the user to account settings so that they may logout of their account entirely. A stretch goal would be to implement a delete account function and give the website more purpose. Likewise, clicking on the home or dashboard button on any of the pages will redirect you back to the beginning.

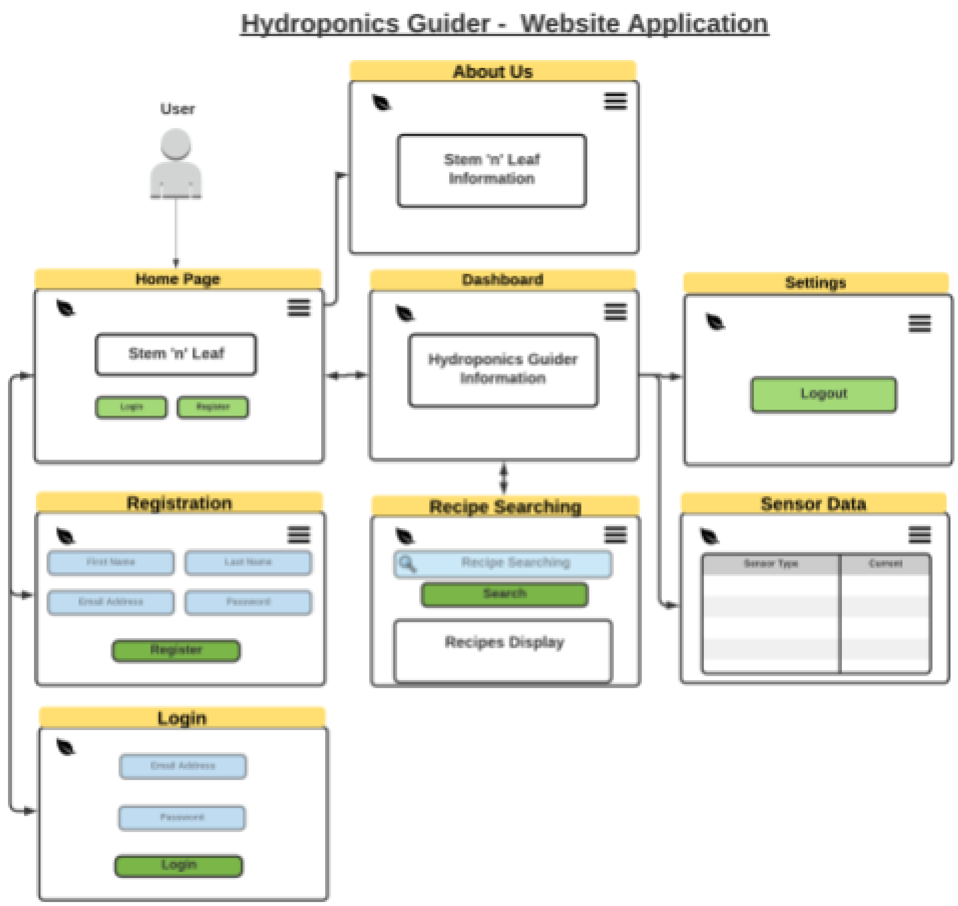


Figure 6-15: Website Application Final Coding Plan

The Sensor Data page will display all the information gathered from the control unit at the exact time they were last connected with the Garden Pod. The data will be displayed in a tabular format with the current measurement.

6.3.3 Database

Our final database structure describes the relationships between the user, plants, and the sensor data. The user will have their information from the registration stored and tied to their account.

The database from the figure is designed to be simple with it only currently having a few tables to track information. The account the user creates will hold their email, name, password, and an ID to track their document within the database. The microcontroller from the hydroponics system will relay the information to populate the database tables to be accessed later. The goal is to be able to access past sensor data from the Garden Pod and control unit to ensure the system has the proper items needed to continue supporting plant growth. This structure is also meant to be flexible and scalable if new information needs to be accounted for and stored. Below is the final entity relationship diagram.

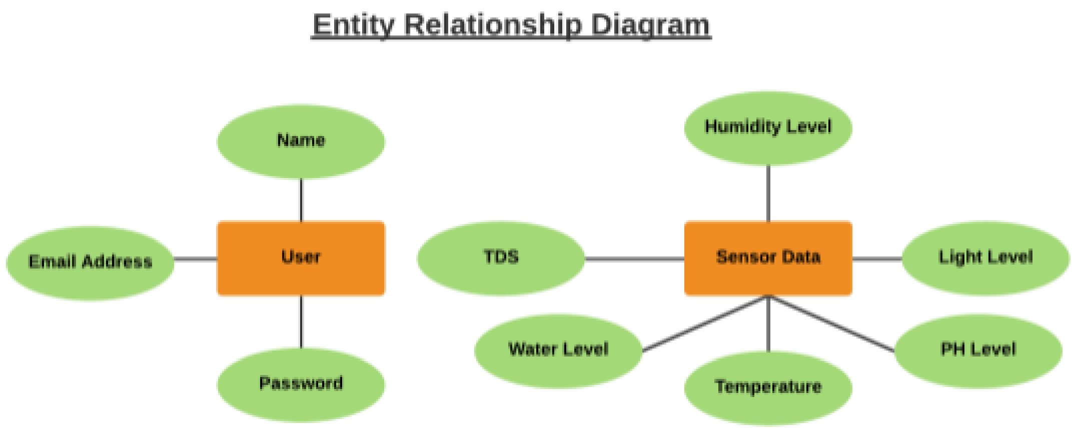


Figure 6-16: Entity Relationship Diagram

6.3.4 Microcontroller

A key component of the hardware system will be the microcontroller that will channel all data from the Garden Pod and central units between each other. The microcontroller will continuously gather information from the sensors and transmit that information when a wireless internet connection is available. The website and mobile applications will be able to interact and monitor the pods from the microcontroller and database. Settings will be able to be inputted for the plants to relay to the microcontroller on what to automate for the Garden Pod and resupply systems. Below is a flowchart detailing in what order the microcontroller will check the sensors and gather data.

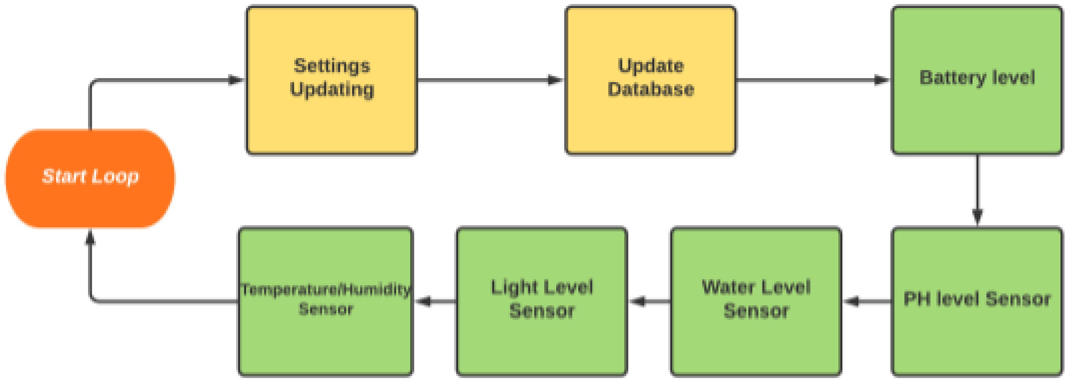


Figure 6-17: Microcontroller Monitoring

The microcontroller flowchart comprises many components with their own logical processes so each of the green sensor sections will be split to smaller flowcharts for readability purposes. Before we check any of the sensors we want to ensure that the battery is sufficient for the Garden Pod to not waste time or damage anything. The microcontroller will first update the settings based on the user's input and then update the database. If there is a battery issue the mobile and website applications will notify the user and not update sensor tables. Then the light sensor logic will be entered and checked.

6.3.5 pH Sensor

The microcontroller loop will first encounter the pH sensor and go through the logic of that loop. The logic will be to collect sensor data from the Garden Pod and then attempt to increase or decrease the pH solution if the water is not within limits. We intend for the microcontroller to collect data when the user prompts it or on a timed schedule. The user will set the values from the website or mobile application. When the system detects an imbalance the microcontroller will activate the pumps from the control unit and attempt to resupply the Garden Pod if connected. Testing of the microcontroller activating the pump will be done during Senior Design 2. Below is the flow chart describing the logic.

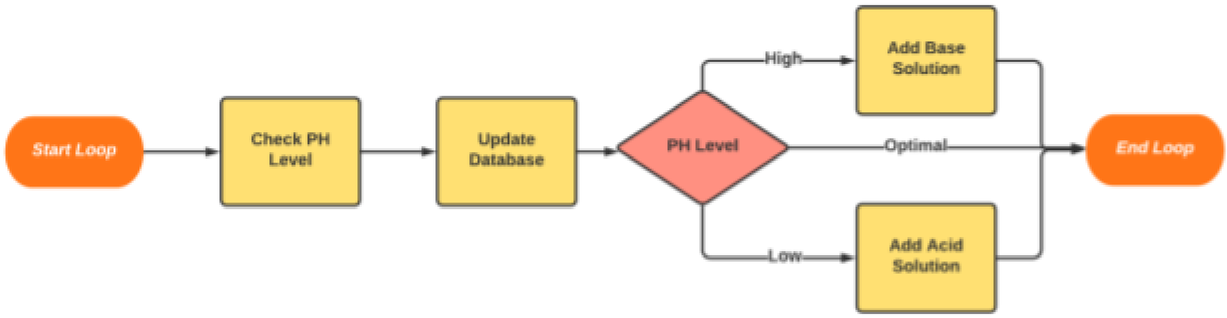


Figure 6-18: pH Sensor Monitoring

6.3.6 Water Level Sensor

Following the pH sensor, the water level sensor will measure the water level periodically to ensure the Garden Pod has enough liquids to grow. If all is well then the sensors will communicate the sensor data to the microcontroller when needed. The microcontroller will then relay the information to the database. The control unit will attempt to pump water to the Garden Pod once they are connected. Once the levels of the water are sufficient then the pumps will stop and the user will be able to safely remove the Garden Pod.

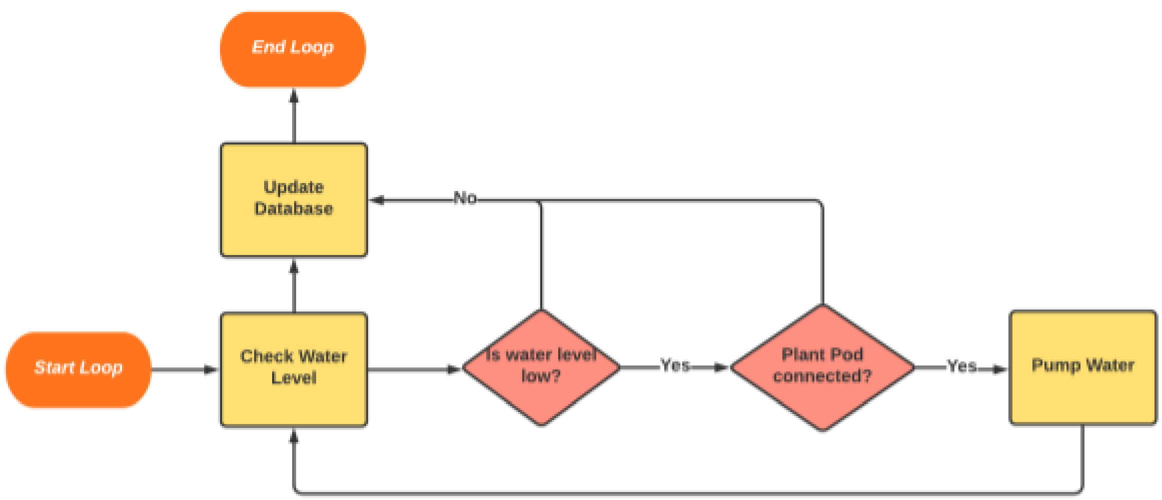


Figure 6-19: Water Sensor Monitoring

6.3.7 Light Sensor

The next subsystem after the water level sensor will be the light sensor. This sensor is important because it is required for the grow lights to stay on for a period of time for the plants in the Garden Pods to grow. Too long or too little could inhibit plant growth and waste power. Below is the flowchart for the sensor.

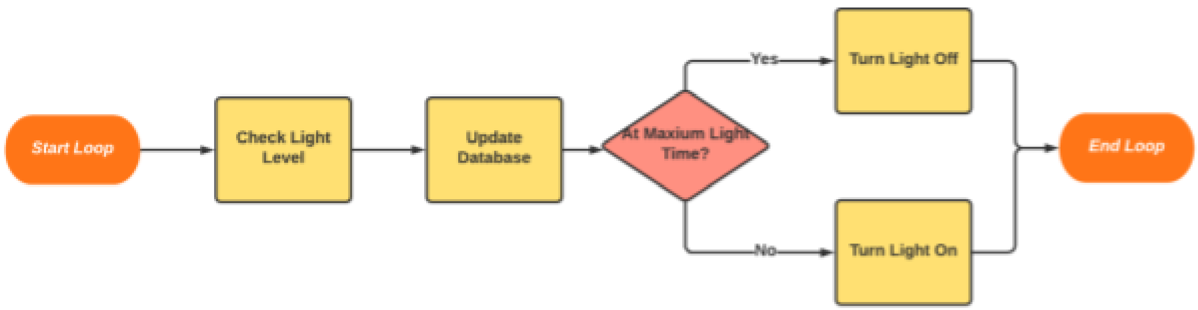


Figure 6-20: Light Sensor Monitoring

The light sensor loop is simple in only needing to check the light levels in the Garden Pod and the set light levels. If the light level in the pod is insufficient for plant growth then the microcontroller will relay that information to the database and attempt to turn on the Garden Pod grow lights, given sufficient Garden Pod battery.

6.3.8 Temperature/ Humidity Sensor

The last sensor loop will be the temperature and humidity Sensor. Below is a flow chart describing the logic.

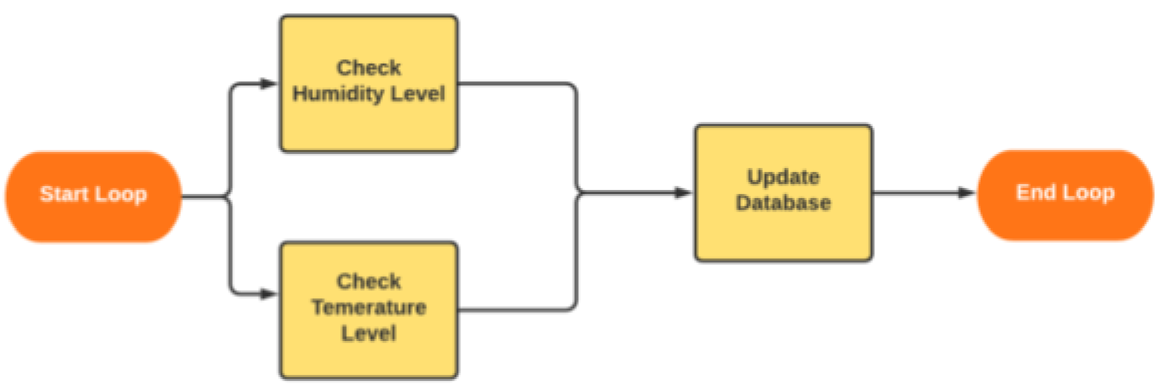


Figure 6-21: Temperature/Humidity Sensor Monitoring

This loop will check the sensor readings from the humidity and temperature sensor and relay that information to the database through the microcontroller. There will be no adjustment of resources, only prompting the user if the temperature and humidity levels are within safe levels for the plant and the hydroponics system.

6.4 Raspberry Pi

The central unit of the tank will need a relatively strong controller. The easiest answer to this problem is a small computer. Since we will need Bluetooth components, Wi-Fi components, and logic it makes sense to have a small computer inside the central unit. The most common and easily accessible controller we can use is a raspberry pi. A raspberry pi is a small computer that runs its own version of Linux. There are many benefits to using a raspberry pi in the central unit which will be covered in this section.

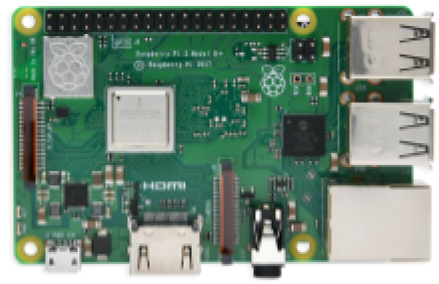


Figure 6-22: Raspberry Pi

The raspberry pi features a 4× ARM Cortex-A53 CPU. This is now the most powerful CPU in the world but it is good for use in our project. It has 1 GB of DDR2 RAM that runs at 900MHz. The newer versions of the raspberry pi have onboard Bluetooth and 802.11n wireless internet. They have a removable microSD storage that's capacity can be changed based on the need. The raspberry pi has a very important 40 pin GPIO header set that will be used to control sensors and pumps. The raspberry pi also has various parts for usage such as USB, HDMI, audio jack and others.

The first major benefit from using the raspberry pi would be the onboard features of it. The raspberry pi features on board Bluetooth and wireless internet. Bluetooth is very important for our use. We will be using Bluetooth (or some sort of wireless connection) to communicate with the microcontrollers in the modules. With the onboard Bluetooth of the pi we will not need to take into account separate Bluetooth controllers that are needed in the modules.

Another positive feature of the pi is the use of the onboard Wi-Fi card. Using this card not only gives us access to the internet but also the ability to host our own network or to host on another network. Using the onboard internet features of the pi we would be able to host an apache website to control the hydroponics system directly from the pi. The pi would host the website on the internet and the user would be able to view the controls directly from their network. This would also reduce the cost of having to host the website on another server costing us more.

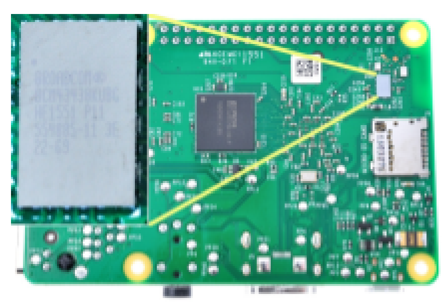


Figure 6-23: Raspberry Pi WiFi card

The raspberry pi has 40 GPIO headers. These headers can be controlled from inside the pi using a multitude of different languages. Python would be the easiest language to use since it is on the pi by default and very user friendly. We can use the GPIO headers to control mosfets to switch on and off pumps to manage the nutrients, pH, and to control the flow of water from the main tank to the modules.

The alternative to using the raspberry pi would be to design our own circuitry. We would take a microcontroller to manage the logic and design a circuit so that it would be able to work in tandem with a bluetooth controller and a wireless network card. This circuitry would be more difficult than just using the raspberry pi, however, it would prevent us from using a prebuilt device like the raspberry pi. Using our own circuitry may help to cut costs and it would give us more of a learning experience.

6.4.1 GPIO Pins

The GPIO pins on the raspberry pi are very useful. GPIO stands for general purpose in out pins. The pins on the raspberry pi can be used for input and output into electronic devices. This type of application is especially useful in our case since we need to control electronic devices with high or low electric impulses. Although the aspects of the GPIO pins are great for the raspberry pi they do come with their drawbacks.

The standard set of GPIO pins consist of a multitude of different purposes and in the next section I will discuss how they are used for each pin.

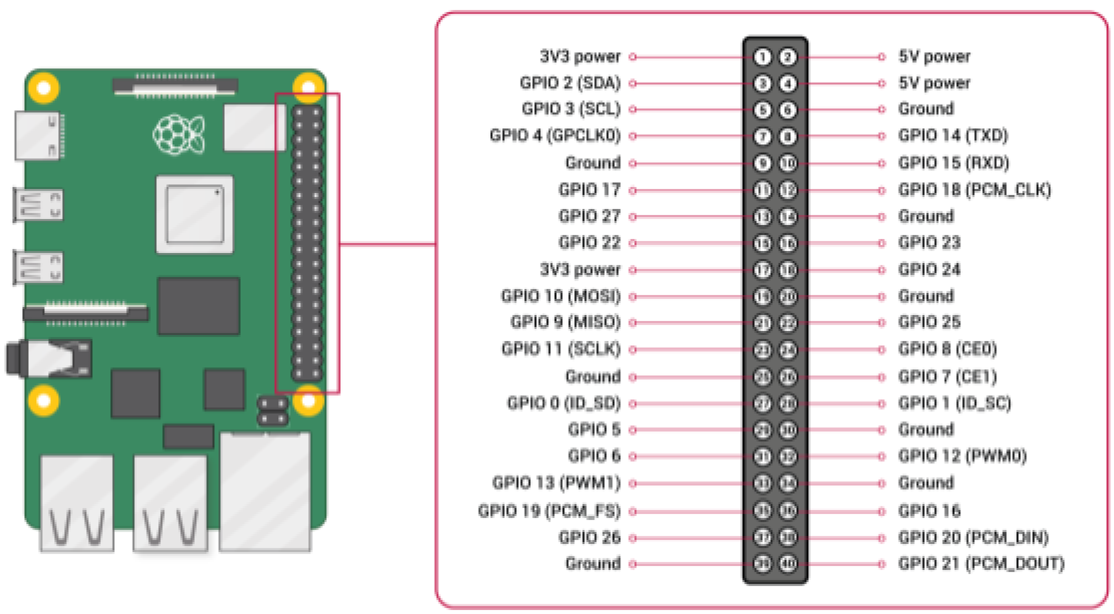


Figure 6-24: Raspberry Pi Pinout

There are 2 different types of power pins shown in **Figure 6-24**. There is 3V power and 5V power, these pins have their uses but for the most part there is no use in our situation.

We require higher voltages of power for our sensors so they will not be able to be powered by the raspberry pi. Moving down the GPIO pins are useful in each of their own aspects listed in **Figure 6-24**. The GPIO pins are able to use more than just inputs and outputs as well, in fact they are able to handle PWM, SPI, I2C, and serial functions.

The problem with GPIO pins is that they only have a high and a low output. Our sensors require an analog input to read them. Raspberry pi GPIO pins are not able to handle analog signals so there must be an in between to go from analog to digital. To do this there are multiple options. One of the options that we can use is by simply continuing to use arduinos with the raspberry pi. We would be able to handle analog inputs with the raspberry pi and float those inputs through the available digital communication protocols that the raspberry pi offers. Another option to consider is to use a store bought analog to digital converter. These converters can take in multiple analog signals and send them to the raspberry pi digitally.

7 Project Prototype Testing Plan

From prototype construction will be the testing phase, which this chapter will analyze the prototype testing methodology our group has developed for our project. These tests in these sections will go in depth about the components of the hardware and software systems. This will be an ongoing challenge that will be crucial to ensure our project's success. The tests like the prototypes are subject to change as official building begins and when components are linked together. Test scenarios will be documented and diverse to strengthen any weaknesses.

7.1 Hardware Test Environment

Testing environments are not difficult to find at UCF. UCF has labs specifically designed for students in the senior design course. The first lab that is the most main one is the senior design lab on the fourth floor of the engineering building. There is another lab named the Texas instrument innovation lab on the first floor of the engineering building. This lab is very well equipped with all sorts of technology. There are numerous 3d printers and other technological equipment. These labs are equipped with equipment such as multimeters oscilloscopes, function generators, resistors, capacitors, and inductors. There are plenty of test benches for all of the groups to test their projects.

7.2 Hardware Specific Testing

Each component chosen for this project needs to be individually tested to ensure that it is functioning properly. Plans are required to test each of the individual components. We must be able to uniquely identify that each component we purchase is functioning correctly before combining them and in this section the plans for each of the components are laid out. The proper techniques for testing will also be updated as the project progresses.

7.2.1 Pump Testing

Hydro pumps are extremely important for this project and testing them properly to ensure that they have similar flow and water lamination is important. The flow of the pump must be consistent for multiple reasons. When the hydroponic module is hooked up to the base the water will need to be changed with fresh, balanced water. The balancing with pH and nutrients will be discussed in a different section but the flow of the water from the base into the module needs to be tested. If the flow from the base to the module is not uniform then it is hard to measure just how much water is put into the unit without a water level sensor. To test the flow rate and the uniformity of the pump we will use a controlled setting. We will take a certain amount of water and check how long it takes to pump a certain amount. This will be repeated with different amounts of water to check and make sure the flow rate stays uniform no matter the volume of water. In practice we cannot expect to have the same amount of water put into the units every single time so failsafes must be taken into account. One way this can be implemented is by using a water level sensor to identify just how much is taken out of the base or how much is in the unit. This requires almost real time communication between the unit and the base.

7.2.2 LED Lighting

Lighting is a very important part of the hydroponics system. All plants need light to grow and if plants are inside they will not get the proper amount of lighting without help from an independent lighting source. For our project we are using full spectrum LED lighting to ensure a cost effective and proper lighting solution. The way the lighting will work is by sensing ambient light in the area which will be done by a light sensor discussed later. The value measured by the sensor will only be part of the solution. The microcontroller will also be measuring how long the plant was getting light from the ambient light. If there has not been enough light given to the plant within a day then the LED lights will have to be turned on. The LED lights will have to be variable to supplement ambient light instead of fully replacing it. The microcontroller will tell the LEDs how much light to output and adjust them as necessary. The LEDs will need to be tested to make sure they output the correct amount of light and the correct type of light. We will also need to establish a baseline measure to see how much light is outputted from a certain microcontroller input. We will do this with a light sensor and a variable microcontroller output. We will also need to ensure that the microcontroller can handle the power throughput of the LEDs and we will most likely decide that they need to be powered from an outside source.

7.2.3 Light Sensors

The light sensors are an integral part of the hydroponics system. There are many different ways the light sensors could be used and having multiple working together in conjunction could make the end product even better. Having multiple is more complex and is a stretch goal for us. We are going to focus on implementing and testing just one light sensor but I will describe producers for implementing and testing more than one. For just one light sensor the purpose is to measure the ambient light around the unit. The ambient light measured is used to decide how much light we need for the plants. If there is enough ambient light for the plant to grow then there is no need to waste power on our grow light. We measure the ambient light in two factors. We measure the light in strength and also duration. The duration of light is important because plants need a certain degree of light for enough time to grow. If there is not enough light then the LED lights are turned on. The light sensors will be tested by a known amount of light adjusted up and down and the values of the light sensors will be recorded to make sure that it follows how much light is given. If we do decide to implement more than one light sensor we will be able to have a variable light source. The light source will shine on the plants but also a light sensor that we place inside the unit to measure how much light is actually shown on the plants. Using this data we will be able to adjust the light inside of the unit to match how much we need for the plants. The testing process would be the same for a single light sensor.

7.2.4 PH Sensors

A pH sensor is one of the major components that we will be using in our project. The pH sensors will be used to measure the specific pH value in the unit and the base. The garden unit pH will be necessary to check if it needs to be changed and these values will be transferred to the base which will transfer them to the various outside sources. The pH number will be around a prespecified value and pH balancing solutions will be added to the water to adjust the pH. When the water is in the base it will need to be balanced and nutrients will need to be added before it is sent to the unit. The pH of the water will be recorded by the pH sensor and then balanced and then pumped out to the unit. The testing for the pH sensors will need to be twofold. Since we are using multiple pH sensors we will need to test that they are working individually and working together with precision and accuracy. The first test will need to show that each sensor can correctly identify a pH of a solution. We will measure the pH solution with each of the sensors and then compare to make sure they are the same. We will then need to check if the pH can be measured accurately by taking a solution with a known pH and measuring the pH of that solution with each sensor.

7.2.5 Water Level Sensors

The water level of the base is an important metric that we need to keep track of. There are multiple less stretch goal options for monitoring the level of the tank which I will go over first before discussing the water level sensor. If we were to use a clear base and to mark when more water needed to be added it would be simpler than using a water level sensor but a water level sensor would give us a good metric to see electronically where the water was at. If we are to use a water level sensor we will need to test it to make sure we get accurate results from the sensor. To test this sensor we will take known levels of water and measure the result in the sensor's output.

7.2.6 Power Systems

Power systems are an important metric to test when analyzing our system as a whole. With so many different moving parts, sensors, microcontrollers, and wires it is hard to ensure the system will be without noise with testing. We will analyze the system as parts first to check to make sure each part is functioning properly first before analyzing the system as a whole. When checking the system all together we will use an oscilloscope to check for noise and ripples in the system. Noise or ripples could indicate electromagnetic interference or even physical interference. To rectify this we will need to identify what is causing this problem and use trial and error to fix it. We will also need to ensure that the system itself isn't under too high of a load. If there is too much power running through some of our components this could cause problems. If, for instance, we attempt to power some of our higher intensity sensors through the microcontroller we may fry the circuitry in the controller by running too much current through it. We need to account for this by checking to make sure too much power is not running through our components with multimeters and oscilloscopes before running it at full load.

There will be several test points placed through the pcb which will allow us to measure voltage and current levels. An overage in power could be fatal to any electronic system and could bring harm to people as well. For these reasons, testing the performance of these power systems is very crucial. We also have to be sure each device receives a sufficient amount of power to function properly, even low voltage levels can damage the electronics in the long term. We will be paying especially close attention to the AC to DC converter since it will be connected directly to line power, as well as the power mosfet which could potentially break the ultraviolet LEDs.

7.3 Software Test Environment

The success of an automatic hydroponics system relies on the software components as much as the hardware. If our systems don't work together in unity the project will fail and the plant will not grow. We have many levels of systems working together. We must incrementally test all the components of the software system before we put pieces together and test the overall system. The time required will depend on the interactions and problems that arise from configuring the sensors and software.

The testing of the software will take place on our own personal computers and smart devices to fine tune our applications. Ideally, in the past semesters we would be able to access the Senior Design Lab without restriction and in safety, but unfortunately due to the COVID-19 pandemic there will be a risk if we do try to access the lab. The lab is great in that it offers testing equipment as well as a controlled environment to experiment, test, and implement.[13] It also provides lockers to keep our belongings locked up while we are not there so that there is no interference from other engineering students or unauthorized personnel, if we leave our project there.

The hardware that will be collecting or sending the data from the sensors will be the Wi-Fi module and microcontroller. We intend to program these components in Arduino because our group has shared experience within the software that the Arduino uses. [14]While testing the Arduino software debugging items should be straightforward in that we can use serial printing to get an overview of the state of the software program. Arduino is great in that it has a large open source community with extensive Arduino troubleshooting guides, which will help us to identify problems with our code. [15]The Arduino Software (IDE) is open source and has extensible software. The programming language can be expanded through C++ libraries or by finding examples of items we are trying to implement. There are also a ton of drivers for microcontrollers that allow us to have other design possibilities.

The system we are going to implement will be simple in only needing to detect, gather, and then store information from the plant units. The code we develop now will change as we apply new changes and encounter new challenges. To exemplify a challenge, the system detects if the plant unit has enough nutrients or not then sends a signal activating the appropriate pumps and solutions to resupply the plant unit. The software testing from there will consist of communicating and retrieving data from the hardware to software database and web based applications. The results from the tests will be apparent and direct to what issues or successes have occurred.

7.4 Software Specific Testing

Our hydroponics system will have software tests consisting of the web server, applications both web and mobile, and database. The tests will be conducted in stages with first testing each subsystem of the software unit as a separate component to each other and to the actual hardware.

Next, we will test each system joined together with increasing interconnection to validate proper communication between each section. The software and application compatibility with the different software systems will be simulated within coding environments and Android emulators. Implementation on real devices will be the final testing stage for building of the prototype. There will be incrementing steps in functionality as each stage is to be tested.

As we are testing each system, we will analyze the effects on changes across the entire system when the different software subsystems are connected together. Stability within the entire system is crucial when implementing new changes across the software components. The types of tests we intend to conduct will be basic functions expected for the system and subsystems to handle and then increasingly more challenging cases to identify weaknesses to strengthen. Any results that occur that stray from expected or average results will be recorded and traced. Finally, once the hardware and software become linked the accuracy of what's recorded versus the expected outcomes sent will be tested.

7.4.1 Web Server

The web server holds an important role in hosting the website to the user and any other visitors. Without the web server storing all of the website's data, users of the website will not and cannot access information regarding the sensitive sensor data information of their Garden Pod. The speed and accurate performance of our web server is needed to handle the user application interactions.

Data sent to the web server needs to be correctly identified and stored to be later used. Notification of Garden Pod sensor data lacking items to sustain plant health needs to be identified quickly and automated to fix. Server testing won't require the hardware of the hydroponics systems to be active and can be done on a command console from a computer. Over the course of the development cycle of this project the web server will be tested incrementally and continuously. As good practice and stated in the intro, the web server will have individual components tested before being combined and tested again.

The server testing will occur through a local Wi-Fi network and group members will try hosting it on their own laptops.

The main components of the web server to be tested will be sending and receiving data from the database. The most important requests from the server concerning the Garden Pod unit will be the plant's health based on sensor data. Also, requests from the microcontroller within the central supply unit that it is low on resources will be considered and recorded.

Receiving and storing the data from the hydroponics system will be a part of the functions for testing. This will be done by sending a custom request to the server and observing if the server sends back the correct data. The goal is to be able to do this through the website and mobile app without mismatching data or errors. Any results from this stage will be scrutinized and compared to the correct result to verify success. In contrast, sending data to the database can be tested by using the mongoose API within Node.js to access MongoDB. To test storing data, establishing a connection to the database will be required before anything else can be done. Messages from the console log will notify of any errors found. After connection is established and verified, a request with example data will be inputted through a function to store data in the database. The goal is to have no error messages be sent back to indicate that the data was stored in the database.

7.4.2 User Application Testing

The website along with the mobile app will be the interfaces in which the user will have access to the hydroponics system, their sensor data, and their plant. Testing and ensuring website functionality will focus on the site's ability to process users registrations and requests while communicating with the other software elements. The website will work with the web server to interact with the user's account and sensor data. The intricate components of the website that will need to be tested thoroughly will be the registration, authentication, pod sensor display, and recipe searching.

When the user first obtains the hydroponics system and wants to begin growing plants they will need to register and create a unique account in order to access recipe searching, view data from the sensors, or access any other functionalities of the hydroponics system. A name, password, and email address with correct formatting will be needed. The fields of these selections of information will need to be validated as well. Testing the registration will be done by using a set of example accounts that test correct and incorrect inputs from the user. The user will be notified of invalid or inconsistent inputs that must be fixed before continuing. The console will be able to print out proper and improper registration data. Below is a table listing the inputs and notifications expected.

Input	**User Access Allowed**	**Notification**
Correct Credentials	Yes	None
Incorrect Credentials	No	No user found or invalid password or username
Duplicate registration	No	User already created, try logging in

Table 7-1: User Application Testing

After the user has registered they will be required to login for future use. Now that the user must provide accurate credentials to authenticate their identity and be connected with the correct information. Registering duplicate users will not be allowed and notify the user appropriately There is a module for Node.js called passport that is middleware that allows for authentication. [16]Extremely flexible and modular, while also allowing the login and logout function of the website the user accesses are guaranteed. Testing the authentication requires the input the user has for the login matches some part of the database. Correct login credentials will pass the user to the dashboard and their data. While, false or incorrect information will be denied access to anything and be notified to try again. The testing will be trying to input login information and checking if the database can match the credentials correctly to give or deny access. Even if you pass the login page with correct credentials, the website must produce all the matching and correct information associated with that account during an active session.

Unauthorized users should be denied access to account pages and data if they simply enter a similar url of a closed session or logged out user into the web browser. Successful testing will consist of inputting a url of a previous closed session and redirecting the user to the login screen to continue. Otherwise, adjustments will need to be made to prevent loopholes. Once the user is logged in, redirecting pages from the website url would be allowed. Final testing after proper authentication is acquired, will be conducting secure logout trials where the user exits their account through an option or simply closing the website. All combinations of opening and closing the website pages will be tested so that there is a single entry to access an account, but reliable and clear session termination.

The recipe searching function of the website won't affect the settings or sensor data pages. This functionality is easy to implement and provides the user with access to ideas on how to utilize the plant they've grown if it's edible. This recipe search will be a simple input where the user can type ingredients they intend to grow to get a scope of what dishes can be produced. Testing will consist of submitting ingredients that are related to what is possible to be grown and verifying that the dishes being produced are related to the ingredient.

The sensor data pages of the website will display data from the information communicated from the database and from the control unit's microcontroller. The testing will be conducted by having the pod data displays display the correct information from the database.

The user registering with their email is required to prevent spam and to verify authenticity. Testing will be done by creating a generic email to be sent to the user upon successful registration with a valid email address. Input errors from the user will be notified of issues and won't be sent an email until corrected. Sending emails post registration should be fast and not create more issues for the user, website, or web server.

7.4.3 Database Testing

Database testing is required for the project to succeed because a key aspect of the software and sensors is to record and give the user access to plant growth environment information. The database will store all the user and Garden Pod information to be accessed later by different applications. The testing of the database will require similar materials to the web server in only needing a computer to interact with. To test the database, just fill the document tables and run a query returning all the information stored. If the database stores and outputs all the correct information in a desirable organization then we will have validated its performance. This type of testing will be scaled and continuous as we combine new parts to the overall system.

7.4.3.1 Database Testing Plan

The database is still under construction and the hardware of the hydroponics system is not completed yet with user application integration. Below is a list of test cases we plan to use to verify that the database is viable for extended usage.

1. <u>Storing User information</u>
    a. Objective: The database will store the user's email, password, first name, and last name
    b. Description: NoSQL commands will create to the user's document with their information
    c. Expect Outcome: The user's information will be added to the profiles document in the database.
2. <u>Updating User information</u>
    a. Objective: The database will update changes to the user's email, password, first name, and last name
    b. Description: NoSQL commands will update the user's information from the existing database
    c. Expect Outcome: The user's information will be added to the profiles document in the database.

3. Adding a duplicate user to the database
    a. Test Objective: To add a duplicate user that already exists to the database for security issues
    b. Test Description: A new user will be attempted to be created with the exact information found in the database
    c. Expect Outcome: The user will be denied access to the existing account and notified the user already exists.
4. User tries to access sensor data
    a. Test Objective: The user will interact with the database through the user applications to get accurate information about the Garden Pod
    b. Test Description: MongoDB will perform a query to retrieve the sensor data stored in the database
    c. Expect Outcome: Sensor information will be provided to the user from the correct profile
5. Requesting sensor data
    a. Test Objective: Upon receiving Garden Pod sensor data the database will be updated and stored to the correct user profile
    b. Test Description: MongoDB will query the database to update a user's profile that will modify the Garden Pod sensor data. There is no Garden Pod or sensor information from the hydroponics system to gather so we will reproduce the data for testing.
    c. Expect Outcome: The database will be updated with the sensor data to the correct profile.

8 Administrative Content

In chapter 8 we look at the division of work and milestones that we hope to achieve over senior design 1 and senior design 2. Cataloging our milestones and setting our goals early is important for us so that we can keep on track with our work and stay up to date on where we should be in the project. The hope of this section is that we will be able to stay on top of dates, organize, and hit each milestone successfully to complete our project and hit all the goals in a timely fashion.

8.1 Member Roles/Division of work

In this section we will be talking about the different division of work that has arisen from this project. We will talk about how we divided different roles in the group and how each of us did the work that we were divided into. A set division of work is important to us so that we know what each of our own goals will be and the goals of our group mates.

8.1.1 Project Management

We have chosen Don Ellintgon as the project manager for our senior design project. Being a project manager requires time management skills, coordination, and organization. This section will discuss how we came upon the idea of the project as a group and what challenges we faced organizing the project and how we overcame them.

As we began throwing around ideas for the project it became clear that we had a lot of ideas and not a lot of uniformity. We had plenty of ideas that were good and would even be easier than what we finally landed upon. A lot of our ideas were very technologically advanced. Although these would be super interesting to do it became apparent that the costs of some of these projects could stack up very quickly.

We started with a lot of ideas about drones. Drones have become quite the interesting topic. They are becoming more and more available to the basic user and the parts to build them have become cheaper. Some of the ideas we had with drone automation may be frowned upon by the law. Doing a project with this sort of legal acrobatics could prove to be difficult to implement in the long run. We wouldn't want to be jumping through hoops trying to stay within the range of the FAA while trying to do our project as well. Because of this it was difficult for us to attempt to use drones in our designs. We went through multiple phases of looking at different ideas and how we may implement them to create the best senior design project that we could. We eventually came upon our hydroponics idea, after the initial suggestions of the idea we added and changed until it was perfect for what we had in mind. Being a group of intelligent people who work well together we did not initially discuss the idea of project manager and chose to take a more diplomatic approach. The rest of this section will discuss how a leader rose in the group and how they became our leader in the project.

Managing people is never an easy task, everyone has a different set of ideas and visions and to be able to learn those and effectively manage everyone's individual conceptions is difficult. As a leader one must be able to evaluate the needs of a group and keep everyone's concepts and thoughts in mind while they make decisions for the group as a whole. To be able to make decisions for the whole of the group a leader must not let their emotions about the project get in the way of their decision making. They must be able to focus solely on the facts about the project and not be distracted by their emotions and reservations about decisions that they may have to make.

Knowing the goal of the project and keeping it in mind constantly is important when making decisions for the group. When making a decision it is important to have a goal that is defined thoroughly. We achieved this goal by first discussing our goal as a group together. When we met for the first time we talked just to be absolutely positive how our project would pan out and to make a unified goal together. We went on to form 2d models of how the units would be created. We shared a OneNote document so that we could all contribute ideas and formulate a unified picture of the system. After the system was created in writing more descriptive models were needed. We then went on to make a 3 dimensional model of the system. This model was more descriptive and it gave the project leader a better view of what the group as a whole was thinking. In the end we knew as a group that the overall project was going to be an automated modular hydroponic system. Knowing this and keeping it in mind made it simple to make decisions in the nature of the project.

Strong personalities can be dominating in a project setting. Someone who is focused on a single theory or idea can dominate over other prominent ideas that may be better in the group. Luckily for us our group practices good communication skills so we are able to freely discuss our ideas with each other and present things without fear of being shut down. We are able to sift through ideas presented by the group and detect which ones should be implemented and which ones should not. Keeping the unified goal in mind helps everyone to understand which ideas are good and should be implemented.

After the initial idea was presented we had to move on to selecting components for the specifics of the project. Choosing each major component proved to be more difficult than initially thought. One would think that they would just be able to select a part that would work for the system and then implement it but it is not that simple.

Every main component that we selected for use was carefully thought out and made sure that they all worked in unison. The components were selected one by one and since they all needed to be powered from one power source they would need to work together. The power source would have to have sufficient voltage and amperage to power the electronics. The electronics that were put together would have to have an amperage draw that would be less than what is given by the battery. There are many different components that can work in place of each of the different parts of the systems. Sorting through these components and finding the ones that work with our project can be difficult and being a strong leader requires making decisions about these parts.

The scope of this project is that all of the group members' graduations and college careers are riding on it. This project can be an excellent learning experience and a springboard for careers in the future. Through the challenging times we will need to make decisions in the aspect of making progress on the project. Making decisions on a project of this scope can be difficult since it is not just your grade and future that is in the balance; it is the whole group's grade and future.

8.1.2 Documentation

The documentation process is an important process in the senior design project. Having proper documentation is the main focus of the senior design 1 class. Having good documentation is important in making sure building the project for senior design 2 goes smoothly. At first glance the documentation for this project did not seem too daunting. The 120 page report did seem long but doable. As the semester stretched on and we wrote more and more pages we realized that it was going to be more difficult than originally expected to reach our goal.

To coordinate our documentation efforts we decided to use google docs so that we could all work on the document simultaneously. Google docs is an excellent tool that allows us to work on the same document all at the same time. Using google docs we can also see everything that our group mates are working on. We use comments in the google doc to show things that need changing or things that we may need to work on. This type of documentation tool is extremely strong since there is no time wasted on combining separate pieces of work from different group members. This type of document also helps to automate the table of contents process. With google docs we can make the headings of our sections headings in google docs so that they show up in an inserted table of contents.

In the beginning we started to create our documentation to fill out the 60 pages. This went very quickly and we filled up our pages with tables, pictures and plenty of words. Doing this gave us a certain false sense of security. We initially thought that the rest of the document would follow suit since it started out easy enough. We were very wrong for thinking this since it was very easy to start but it really dragged on towards the end of the report. Filling up the document at first was easy because we were able to put our part selection list in there and our research selections. We had plenty of tables and images to fill up extra space in our report but with the most recent updates to the document more and more wordy explanations are required which do not take up nearly as much space.

We decided after our initial draft to just go for the next draft together and not divide any of the sections to our individual members. This almost worked, however, we did not make our page deadline to the fullest. We got close but did not get all the way to the mark. This left us behind for the next draft which continued to snowball. On our next draft we decided to talk about which sections each one of the group members would work on. Doing this and dividing up the work allowed us to be more productive and work more efficiently on each section.

The formatting process for this paper went relatively quick. Being that the document was always one there was no problems with differing formatting from different group mates. The format was the same throughout the document. We started with IEEE formatting and retained it throughout creating our document.

8.1.3 Parts Management

The complexity of this project requires many different parts working together to form one system so parts management is very important to us. To manage all of our parts and to ensure no losses we have chosen Jonathan to be our parts manager. Managing parts is an important task, it does not only require holding parts but also ensuring that they are kept in a safe place and also ording parts and managing the tab associated with the cost of the parts.

In our system we utilize sensors, microcontrollers, and other miscellaneous electronic parts. These parts are all part of the whole system and need to be put together to create the desired result. If there are any parts that are missing it could mean the end of the project or higher costs for the rest of the group. If we need to replace any of the parts some parts are more expensive than others and costs already add up. If there are any parts that need to be replaced the cost will be put onto us as group members. Some parts can be so expensive that the replacement of one will break the budget entirely.

To limit the risk of broken parts and lost parts we designate a parts manager. The main duties of the part manager are to store our parts safely. Storing the parts safely includes making sure that the parts are not in any hazardous situations. Some of these situations that these parts should not be in are standard concerns for electronic components. These situations could be moisture or erroneous electricity. Keeping these components working is relatively simple; they could be kept in a plastic bin that is water tight so the parts are safe during storage and transportation.

Ordering parts is an essential for the entire group. Without the parts there is no project. The parts ordering has gone to Jonathan since he is the parts management, however, we all bring small parts to the project that we already had. Ordering parts early are important because the earlier we can order them the quicker we can test them. When we test parts early we can figure out what works and what needs to be changed. Once we find what is not working in the system we are able to order a replacement part early before it is down to the wire.

The parts that have been ordered need to be paid for. To do that we split the cost between the four of us. Splitting the cost means that we need to be able keep track of the price per each part. One of the duties of the parts manager is to record the cost of each of the parts. We are handling this by running a tab of all the parts. By the end of the parts ordering process we will have an itemized list of all the parts. We would be able to split up the whole price of the parts equally with the group.

8.2 Milestone Discussion

In this section, we map out the projected plan for the milestones we will look to complete each week. This chart will be referenced throughout the year in order to gauge our progress as we move through the design, testing, and implementation processes. It is important to note that this schedule would change if an unforeseen complication presents itself. If we have to push a deadline back and move something else up, we should update this table so that we continue to have an accurate list of goals. If we didn't then the very moment that we have to adjust could cause the future deadlines to be inapplicable given the new circumstances.

8.2.1 Senior Design 1 Milestone Discussion

#	Milestone	Planned Completion Week
1	Finish divide and conquer 1	3
2	Finalize design and get all team members on the same page	4
3	Create CAD designs	6
4	Finalize research on hydroponics and plants that we want	7
5	Design balancing system in base	8
6	Design feed system to units	9
7	Design units reservoir and electronics	10
8	Design electronics for base complete with access to networks	11
9	Design electronics for unit	12
10	Integrate unit and base to communicate with each other	13
11	Design PCB	14
12	Build system with finished electronics and printed parts	14
13	Testing and debugging	15
14	Finalize report	END

Table 8-1: Milestone Discussion

Table 8-1 describes when and what milestone should be achieved for Senior Design 1. These milestones are useful for organizing the flow of the project and for creating reasonable goals. Meeting these milestones each week will help to keep the project on track and to help us not be behind.

8.2.2 Senior Design 2 Milestone Discussion

#	Milestone	Planned Completion Week
1	Acquire Parts	1
2	Printing out parts	2
3	Test Components	3
4	Test Software	4
5	Build Prototype Software	5
6	Build Prototype Hardware	6
7	Test Prototypes systems	7
8	Begin Growing plants	8
9	Record results	9
10	Test Entire System	10
11	Finalize Project	11
12	Project Presentation	END

Table 8-2: Milestone Discussion(2)

Table 8-2 describes when and what milestone should be achieved for Senior Design 2. Since Senior Design 2 has a shorter semester the timelines are adjusted and our work will need to be sped up to meet the time constraint. Working through the gap between the spring semester and summer semester will allow us to comfortably achieve these goals. These milestones are reasonable for our group size and cover most of the aspects to complete and present the project.

Meeting with the group and discussing the milestones each week will help to keep the project on track and address any issues.

8.2.3 Software Milestone

#	Milestone	Planned Completion Week
1	Website Design	1
2	Mobile App Design	2
3	Database Structuring	3
4	Website Pages Coding	4
5	Mobile App Coding	5
6	Database Coding	6
7	Microcontroller Design	7
8	Microcontroller Coding	8
9	Test Entire Software System	9
10	UI Updating	10
11	Finalize functionalities	11
12	Project Demo	END

Table 8-3: Software Milestone Discussion

Table 8-3 describes when and what milestone should be achieved for the software during of Senior Design 2. During the spring semester most of the coding hasn't been implemented yet. After the spring semester ends work will begin on creating and testing the ideas put forth within this document. A lot of the basic coding and designs can be done quickly from previous projects already completed or with online tutorials and teamwork. Working to ensure proper testing is done on the individual components and then the combination of software and hardware will be the greatest time constraint. This milestone encompasses both Senior Design 1 and Senior Design 2. With the easier academic load during summer greater work can be done to fully achieve these milestones.

The tasks may be delayed or completed out of order due to constraints and the project requirements.

8.2.4 Hardware Milestone

#	Milestone	Planned Completion Week
1	Acquire Parts	1
2	Printing out parts	2
3	Test Components	3
4	Test Software	4
5	Build Prototype Software	5
6	Build Prototype Hardware	6
7	Test Prototypes systems	7
8	Begin Growing plants	8
9	Record results	9
10	Test Entire System	10
11	Finalize Project	11
12	Project Presentation	END

Table 8-4: Hardware Milestone Discussion

Table 8-4 describes when and what milestone should be achieved for the hardware during of Senior Design 2. During the spring semester most of the parts haven't been implemented yet and parts are still being acquired. After the spring semester ends work will begin building the subsystems found within this document and designs. Working to ensure proper testing is done on the individual components and then the combination of software and hardware will be the most challenging part. This milestone encompasses both Senior Design 1 and Senior Design 2. Spare parts may need to be rebought if components break or more is needed to perform the task successfully. The tasks may be delayed or completed out of order due to constraints and the project requirements.

8.3 Budget and Finance Discussion

Garden Pod*		
Part	**Quantity/Pod**	**Estimated cost/unit**
Photo sensor resistor	**4**	**$2 - $4**
Led Grow lights	**1**	**$5 -10**
Humidity/ Temperature sensor	**1**	**$2 - $11**
Liquid Ph Sensor	**1**	**$20 - $50**
Capsule w/ water tank (3D printed)	**1**	**$40 - $50**
Water level sensor	**1**	**$5 - $15**
Wireless Module	**1**	**$10 - $20**
Total		**~$100 - $180**

Table 8-5: Garden Pod Budget

Power Pod (Central unit)		
Part	**Quantity/unit**	**Estimated cost**
Liquid Ph Sensor	**1**	**$20 - $50**
Water Pump	**1**	**$30 - $50**
Small water pump	**3**	**$5 - $15**
Tubing	**1**	**$15 - $20**
UV light	**1**	**$20 - $25**

Table 8-6: Central Unit Budget

Part	**Quantity/unit**	**Estimated cost**
Microcontroller	**1**	**$10 - $15**
Pod Capsule w/ water tank (3D printed)	**1**	**$40 - $50**
Miscellaneous materials**	**N/A**	**$50 - $70**
Total		**~$190 - $295**

Table 8-7: Central Unit Budget (2)

*Given the modular nature of the project, the cost estimate will be listed per unit
**Horticultural materials, electrical components, minor parts, etc.

In **Table 8-(5-6)**, We have roughly estimated the cost of materials for this project. Due to the modular nature of this project, we estimate the cost for each potential pod built. We would need at least one each type of pod but could potentially build more than one Garden Pod if it is within our budget. As described in **Table 8-5,** The Garden Pod will be less costly as it mainly features sensors and led lights, the Garden Pod will also be composed mostly of growing space for plant growth. Since the Power pod will be housing most of the hardware, it will have a higher cost, fortunately only one is required, as shown in **Table 8-6**. As the project progresses. We will receive more details on pricing and total cost.

8.4 Final Cost Analysis

Without considering the expense of the building materials, our projections were quite accurate. Our Expenses rather exceeded our initial budget by about $100. This is mostly due to an oversight on building materials but may also have to do with shipping costs as some components were ordered directly from suppliers, which most of the time did not include free shipping.

Final Cost Analysis

Item	Supplier	Price/Unit	Units	Total Cost
Axial Fan	Amazon	$ 12.99	1	$ 12.99
Water Level Sensor	CQ-Robot	$ 21.99	1	$ 21.99
Temperature/Humidity Sensor	Amazon	$ 6.49	1	$ 6.49
UV LED Grow Light Strip	Amazon	$ 19.99	1	$ 19.99
HC-05 Bluetooth Transceiver	HiLetgo	$ 7.99	2	$ 15.98
pH Sensor	DF-Robot	$ 56.90	1	$ 56.90
ATMEGA382P-PU	Amazon	$ 3.30	1	$ 3.30
IRF520 Mosfet	Amazon	$ 1.36	4	$ 5.44
TDS Sensor	CQ-Robot	$ 13.99	1	$ 13.99
Perstaltic Pump	Gikfun	$ 10.43	3	$ 31.29
Subermersible Fountain Pump	Amazon	$ 10.29	1	$ 10.29
Raspberry Pi	Amazon	$ 59.99	1	$ 59.99
12V 6800 mAH Li-ion Battery	Amazon	$ 38.99	1	$ 38.99
Nutrients	Amazon	$ 41.28	1	$ 41.28
Building Materials	-	$165.60	1	$165.60
PCB Assembly	JLC	$100	1	$100.00
Sum				**$604.51**

Table 8-8: Final Cost Analysis

8.4 Conclusion

This report has consisted of the complete technical breakdown of the modular hydroponics system. In the beginning the start of the project was designing the system and setting accurate goals and expectations. The goals we set were to create a hydroponics system that was simultaneously modular and autonomous. The aim is to use sensors and pumps to create the most user friendly hydroponics experience possible. The first stage of our project was the research. The research was one of the longer stages of the project but for very good reason. Everything going into the project needed research. From the plant growing process to the technical selections. The next stage of our project was deciding how the project would look and function. This stage is one of the most important because without a clear definition of goals and expectations we would not be able to properly carry out the construction of the project without some problems.

Along with establishing the technical aspects of the project we also needed to be aware of our goals. More importantly we needed to be aware of the technical aim of our project. The aim of this project was to create a user friendly device that was also design-forward. To do this we needed an easy to use interface and a salubrious system.

After determining all related technical aspects of the system the next step was to finalize our technical selections and record them in the documentation. The technical selections are important so when the parts are ordered we do not need to worry about them not being correct or not working together. Then the parts were ordered. The main construction of the project consisted of microcontrollers, pumps, sensors, lights, fans, PCBs, and other electronic components. The components were assembled to create the automatic hydroponic system. The software was simultaneously developed alongside the assembly of the hardware. As well as the website and application. Once all the components were tested together with the PCB, it was ensured that the final project was assembled. Proper precautions were taken to ensure that the constructions went smoothly and we did not have too many incidents that caused us to remake or reassemble parts.

In the end we were able to develop a hydroponics system that met all of our goals and expectations. Our project was user friendly and worked well giving data to the user in real time. The effect of our project would have a positive influence for the user if it were to enter public production.

I Appendix

II References

[1] E. Sanchez, R. Berghage, Thomas Ford Extension Educator Expertise Greenhouse Production Nursery Production Landscape Management Turf Management Tree Fruit Production Vegetable and Small Fruit Production Hydroponic Production Specialty Cut Flower Production Grape Production Hops, F. D. Gioia, and Nick Flax Former Extension Educator, "Hydroponics Systems and Principles Of Plant Nutrition: Essential Nutrients, Function, Deficiency, and Excess," *Penn State Extension*, 25-Mar-2021. [Online]. Available: https://extension.psu.edu/hydroponics-systems-and-principles-of-plant-nutrition-essential-nutrients-function-deficiency-and-excess#:~:text=Macronutrients%20and%20micronutrients%20are%20both,chlorine%2C%20copper%2C%20and%20nickel. [Accessed: 31-Mar-2021]

[2] "Hydroponics: A Better Way to Grow Food (U.S. National Park Service)," *National Parks Service*. [Online]. Available: https://www.nps.gov/articles/hydroponics.htm#:~:text=What%20Are%20the%20Benefits%20of,the%20same%20number%20of%20plants. [Accessed: 01-Apr-2021].

[3] MongoDB. 2021. *What is the MERN Stack? Introduction & Examples*. [online] Available at: <https://www.mongodb.com/mern-stack> [Accessed 1 April 2021].

[4] Staff, R., 2021. *Top 9 Best Recipe APIs (for Developers in 2021) [40+ Reviewed]*. [online] The Last Call - RapidAPI Blog. Available at: <https://rapidapi.com/blog/recipe-apis/> [Accessed 1 April 2021].

[5] Maximumyield.com. 2021. *Top 5 Apps for Gardening and Hydroponics*. [online] Available at: <https://www.maximumyield.com/top-5-apps-for-gardening-and-hydroponics/2/3176> [Accessed 1 April 2021].

[6] Electronic Design, "EMI Standards For Electronic Systems," StackPath, 07-Jun-1998. [Online]. Available: https://www.electronicdesign.com/technologies/boards/article/21767353/meet-emi-standards-for-electronic-systems#:~:text=Physical%20shielding%20provides%20signal%20attenuation,by%20reflection%20improves%20with%20conductivity. [Accessed: 01-Apr-2021].

[7] EPA, "Water Quality Standards Handbook," Chapter 3: Water Quality Criteria. [Online]. Available:

https://www.epa.gov/sites/production/files/2014-10/documents/handbook-chapter3.pdf. [Accessed: 01-Apr-2021].

[8] Holland Shielding Systems, "100 Shielding Tips and Tricks," *International Appliance Manufacturing*, 13-Jun-2019. [Online]. Available: https://www.assemblymag.com/articles/94891-shielding-tips-and-tricks. [Accessed: 01-Apr-2021].

[9] Quick 220 Systems, "North American Voltage Ranges," *Quick 220® Systems*. [Online]. Available: https://www.quick220.com/page/cp-voltage.html. [Accessed: 14-Apr-2021].

[10] B. Eastwood, "The 10 Most Popular Programming Languages to Learn in 2021," *Northeastern University Graduate Programs*, 17-Mar-2021. [Online]. Available: https://www.northeastern.edu/graduate/blog/most-popular-programming-languages/. [Accessed: 15-Apr-2021].

[11] "Android Studio 4.1," *Android Developers Blog*, 12-Oct-2020. [Online]. Available: https://android-developers.googleblog.com/2020/10/android-studio-41.html. [Accessed: 15-Apr-2021].

[12] T. Holas, "Angular vs. React: Which Is Better for Web Development?," *Toptal Engineering Blog*, 27-Jun-2017. [Online]. Available: https://www.toptal.com/front-end/angular-vs-react-for-web-development. [Accessed: 15-Apr-2021].

[13] "Senior Design Laboratory," *Senior Design Laboratory – Department of ECE*. [Online]. Available: https://www.ece.ucf.edu/academic-laboratories/senior-design-laboratory/. [Accessed: 15-Apr-2021].

[14] "4 Simple Steps for Debugging Your Arduino Project," *circuito.io blog*, 06-Feb-2019. [Online]. Available: https://www.circuito.io/blog/arduino-debugging/. [Accessed: 15-Apr-2021].

[15] "Introduction," *Arduino*. [Online]. Available: https://www.arduino.cc/en/guide/introduction. [Accessed: 15-Apr-2021].

[16] *Passport.js*. [Online]. Available: http://www.passportjs.org/. [Accessed: 15-Apr-2021].

[17] T. RAVPower, "Lithium Ion vs. Lithium Polymer Batteries – Which Is Better?," *RAVPower*, 02-Aug-2017. [Online]. Available:

https://blog.ravpower.com/2017/06/lithium-ion-vs-lithium-polymer-batteries/. [Accessed: 15-Apr-2021].

[18] United States Department of Agriculture, “Report and Recommendations on Organic Agriculture,” *NAL Publications Archive*, 1980. [Online]. Available: https://pubs.nal.usda.gov/report-and-recommendations-organic-farming-usda-1980. [Accessed: 15-Apr-2021].

[19] AlterNet and E. Coleman, “Here's Why the USDA Is Wrong to Certify Hydroponics as 'Organic',” *Alternet.org*, 28-Feb-2017. [Online]. Available: https://www.alternet.org/2017/02/heres-why-usda-wrong-certify-hydroponics-organic/. [Accessed: 15-Apr-2021].

[20] FormLabs, “How to Choose the Right 3D Printing Material,” *Formlabs*. [Online]. Available: https://formlabs.com/blog/how-to-choose-the-right-3D-printing-material/. [Accessed: 15-Apr-2021].

[21] "LAMP stack explained: Master the basics and get started quickly", Ibm.com, 2021. [Online]. Available: https://www.ibm.com/cloud/learn/lamp-stack-explained. [Accessed: 25- Apr- 2021].

[22] "MEAN vs. MERN vs. MEVN Stacks ? What's the difference ?", DEV Community, 2021. [Online]. Available: https://dev.to/amrelmohamady/mean-vs-mern-vs-mevn-stacks-what-s-the-difference-29ge. [Accessed: 25- Apr- 2021].

[23] What is Apache? An In-Depth Overview of Apache Web Server", Hostinger Tutorials, 2021. [Online]. Available: https://www.hostinger.com/tutorials/what-is-apache. [Accessed: 25- Apr- 2021].

[24] "How to Install Apache Web Server on Windows - SitePoint", Sitepoint.com, 2021. [Online]. Available: https://www.sitepoint.com/how-to-install-apache-on-windows/. [Accessed: 25- Apr- 2021].

[25] "How to Install PHP on Windows - SitePoint", Sitepoint.com, 2021. [Online]. Available: https://www.sitepoint.com/how-to-install-php-on-windows/. [Accessed: 25- Apr- 2021].

[26] "Why The Hell Would I Use Node.js? A Case-by-Case Tutorial", Toptal Engineering Blog, 2021. [Online]. Available: https://www.toptal.com/nodejs/why-the-hell-would-i-use-node-js. [Accessed: 25- Apr- 2021].

[27] "SQL vs. NoSQL Databases: What's the Difference?", Ibm.com, 2021. [Online]. Available: https://www.ibm.com/cloud/blog/sql-vs-nosql. [Accessed: 25- Apr- 2021].

[28] "SQLCourse - Lesson 1: What is SQL?", Sqlcourse.com, 2021. [Online]. Available: http://www.sqlcourse.com/intro.html. [Accessed: 25- Apr- 2021].

[29] "Non-relational data and NoSQL - Azure Architecture Center", Docs.microsoft.com, 2021. [Online]. Available: https://docs.microsoft.com/en-us/azure/architecture/data-guide/big-data/non-relational-data#:~:text=A%20non%2Drelational%20database%20is,type%20of%20data%20being%20stored. [Accessed: 25- Apr- 2021].

[30] "Types of NoSQL Databases", MongoDB, 2021. [Online]. Available: https://www.mongodb.com/scale/types-of-nosql-databases. [Accessed: 25- Apr- 2021].

[31] The Verge. 2021. How NASA is learning to grow plants in space and on other worlds. [online] Available at: <https://www.theverge.com/2018/9/21/17883780/nasa-veggie-plants-space-station-mars-moon-soil-food#:~:text=Hydroponics%20involves%20delivering%20water%20and,in%20a%20misty%20air%20environment.&text=Through%20that%20research%2C%20astronauts%20have,eat%20%E2%80%94%20plants%20on%20the%20ISS> [Accessed 26 April 2021].

[32] R. Mathur and R. Mathur, "MEAN vs MERN vs MEVN vs LAMP Stack for Development", | Technology Insights By Arka Software, 2021. [Online]. Available: https://www.arkasoftwares.com/blog/mean-vs-mern-vs-mevn-vs-lamp-stack-for-development/#Advantages_of_LAMP_Stack. [Accessed: 25- Apr- 2021].