Urban Green

March 3, 2016

Dr. Andrew Rawicz
School of Engineering Science
Simon Fraser University
Burnaby, British Columbia
V5A 1S6

Re: ENSC 440 Design Specification for EcoGarden, a Completely Self-contained Aeroponic Growth System

Dear Dr. Rawicz,

Please find the design specification document for the completely self-contained aeroponic growth system, EcoGarden our project for ENSC 440 Capstone Engineering Science Project.
System is an Aeroponic growth system designed to provide an environment that allows people to grow plants indoors and in non-conducive farming climates such as high population density areas.

In this design specification report, we outlined how we will achieve each prototype deliverable in our Functional Specifications document for our self-contained aeroponic growth system. This document will determine the requirements when developing and presenting our final prototype.

Urban Green is a spirited Company consisting of five ambitious students with variety of knowledges and skills: Eric Ganzert, Timothy Horita, Michael Foo, Mahbod Tork-Tatari, and Anita Khoshnavaz. If the reader has any questions or comments, please feel free to contact us by email at eganzert@sfu.ca for any questions or concerns.

Sincerely,

Eric Ganzert

Eric Ganzert
CEO
Urban Green

Enclosure: Functional Specification for EcoGarden

Urban Green

Design Specifications for EcoGarden

Team:	*Michael Foo* *Eric Ganzert* *Timothy Horita* *Anita Khoshnavaz* *Mahbod Tatari*
Contact:	*Eric Ganzert* *eganzert@sfu.ca*
Submitted to:	*Andrew Rawicz* *Steve Whitmore* *School of Engineering Science* *Simon Fraser University*

Issued:	*March 3, 2016*
Revision:	*1.1*

Abstract

Most of people who are living in an urban area have difficulty getting access to fresh and healthy vegetables and herbs. Due to a poor growing climate people with farming expertise cannot grow all the specialty crops that they want. Aeroponic systems are the process of growing plants in an air or mist environment without the use of soil or other growing medium; which is useful for people who want to grow their fresh vegetables and herbs indoors. EcoGarden by Urban Green features a self-contained environment with dynamic temperature, humidity, and sunlight control. It monitors different conditions including temperature and humidity to maintain optimal growing conditions.

EcoGarden differentiates itself from competitors by controlling all aspects of the growing process while using aeroponics to maximize growing potential. It uses a heater, fans, lights, and pumps to attain and maintain the desired conditions. It monitors both the temperature and humidity using a single sensor and uses that information as a feedback signal to maintain inputs conditions. All of the peripherals are controlled using a single Arduino Uno, which also displays current and inputted conditions. The Arduino also controls a watering cycle that runs regularly to ensure optimal growth.

Table of Contents

List of Figures and Tables iv

Glossary v

1 - Introduction and Background 1

2 - Watering System 1

2.1 - Main Watering System Design Requirements 2

2.2 Humidity System 3

3 - Enclosure 4

3.1 - Lower Enclosure 4

3.2 - Upper Enclosure 6

4 - Control System 8

4.1 - Sensors and Actuators 8

4.2 - Microcontroller 10

4.3 - Control Panel User Interface 11

4.4 - Wiring and Hardware Installation 13

5 - Software 14

5.1 - Select Variables 14

5.2 - Humidity Control 15

5.3 - Temperature Control 15

5.4 - Pump Control 16

5.5 - LCD Display Screen 16

6 - Sustainability and Life Cycle 16

7 - Conclusion 17

References 18

Appendix A - Test Plan 19

List of Figures and Tables

Figure 2.1: Illustration of Aeroponics 2

Figure 2.2: Growing cup 2

Figure 2.3: Watering apparatus 2

Figure 2.4: Water system container 3

Figure 3.1: Solidworks of entire enclosure 4

Figure 3.2: Solidworks of lower enclosure 4

Figure 3.3: Lower enclosure dimensions 5

Figure 3.4: Solidworks of upper enclosure 6

Figure 3.5: Assembled enclosure 7

Figure 3.6: Upper enclosure dimensions 7

Figure 4.1: Control system diagram 8

Table 4.2: Sensor and Actuator Table 8

Table 4.3: DHT11 Sensor specifications 9

Table 4.4: Arduino vs Raspberry Pi 10

Figure 4.5: Control panel diagram 11

Figure 4.6: User Interface circuit diagram 12

Figure 4.7: Relay wiring 13

Glossary

Aeroponics: the process of growing plants in the open air without use of soil or a growing medium.

Arduino (Uno): A type of microcontroller programmed using an Arduino integrated development environment.

GPH: Gallons per hour, a unit of flow rate.

GUI: Graphical User Interface, what the user sees and interacts with on a screen.

LCD: Liquid crystal display, a type of display screen.

LED: Light-emitting diode, a p-n diode that releases light when activated.

PSI: Pounds per square inch, a unit of pressure. 14.7 psi is approximately 1 atmosphere.

Raspberry Pi: A small, single-board computer that uses a Linux-based operating system.

1 - Introduction and Background

The EcoGarden is an automated aeroponics apparatus which allows the user to grow a variety of vegetables and herbs with relative ease. The system controls all aspects of the growing process including:

- Chamber humidity
- Ambient chamber temperature
- Timed simulated sunlight exposure

It does so by using a temperature and humidity sensor, in conjunction with the Arduino's built in clock functions to control the timing of the lights and watering system. The system diagnostics are displayed to the user via an LCD display connected to the arduino. The EcoGarden features a control panel that will allow the end user to adjust the desired humidity, chamber temp and timed simulated sunlight.

The purpose of this document is to outline the main design aspects for each of the features of the EcoGarden. For each feature the following aspects will be considered:

- Functionality requirements
- Physical specifications (dimensions, construction and materials)
- Safety considerations
- Usability requirements (customer use and maintenance/service/repair)

The document also outlines the design specification for the proof-of-concept model for the EcoGarden and explains how some of the design features will satisfy the functional requirements for the EcoGarden.These requirements are to ensure proper performance, quality, and safety of EcoGarden. User documentation are also provided to prevent confusion and improper use of the system.

The design specification document is intended to serve as a references to the member of Urban Green team during design, development and testing stages of the EcoGarden.

2 - Watering System

The watering system serves the purpose of applying a fine mist of water to the roots of the plants. In aeroponics systems the roots are hanging freely in air and a pump system applies the mist of water and nutrients to the roots at regular intervals, keeping them wet at all times (Figure 2.1). This method allows much more oxygen to be absorbed through the roots improving the crop yield.

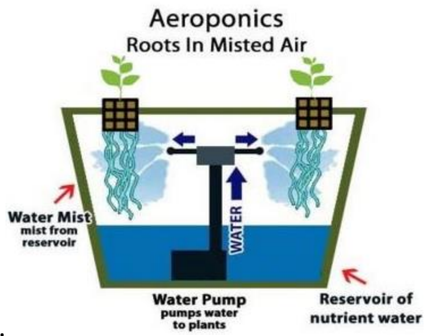

Figure 2.1: Aeroponics illustrated

In aeroponics the crops are planted in a small plastic cup that has holes in its bottom and sides for the roots to grow out of (Figure 2.2). A couple of weeks in, the plant has sprouted and the roots begin to hang down below into the watering chamber.

Figure 2.2: Growing cup

2.1 - Main Watering System Design Requirements

The pipe structure that delivers the mist to the roots has been designed as shown in Figure 2.3 below. The following list outlines the design specifications and how they have been met with this design.

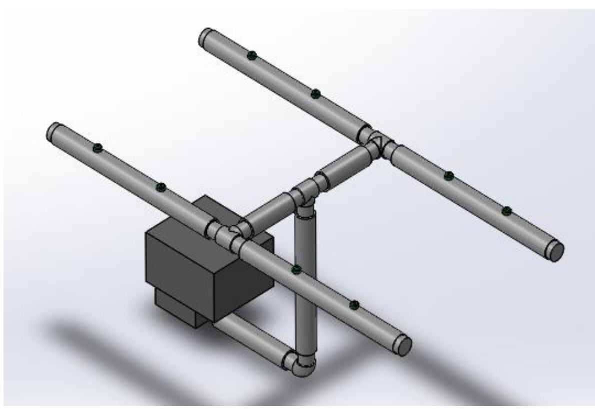

Figure 2.3: Aeroponic watering system

- The pump's output is 500 gallons per hour (GPH). Since it only has to pump up vertically about 10 inches this is more than enough pressure for the misting valves to make a fine mist.
- There is a ball valve at the pumps output (not shown) that can be adjusted to reduce the water flow exiting the pump. With this feature, the pressure can be tuned to make the flow rate coming out of the misting valves to a suitable value.
- The misting valves are screwed directly into the pipes. It is believed that 4 on each side will provide adequate coverage of all the roots.
- The pipes and pipe fittings are made of rigid plastic and are 1/4" in diameter. This is perfect because it is far cheaper than metal plumbing while still strong enough to contain the pressure.
- The two top long pipes will be secured to the sides of the plastic box that contains the watering system.
- The overall size was designed to fit in the watertight plastic enclosure that is discussed next.

The Plastic box that will contain the pipes is shown below in Figure 2.4. Following is the design specifications that led us to this box selection.

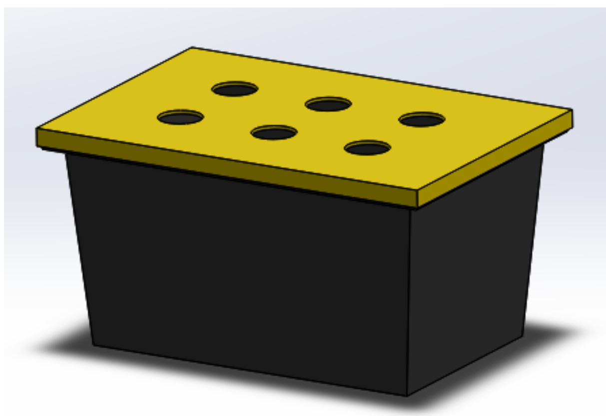

Figure 2.4: Water system container

- The box is watertight when the lid is attached, so that there will be no leakage when the watering system is spraying mist.
- Six growing sites allows for growing a diverse variety of crops. This spacing allows a large crop yield, while preventing plant crowding that would negatively impact the plants access to light to all its leaves.
- The dimensions of this box are 15 inches tall, 28.5 inches long and 19 inches wide

2.2 Humidity System

Also part of the overall watering system is the humidity pump misting system. This is a separate, smaller capacity pump that has a single pipe leading up through the lid of the containing box of Figure 2.4. A single misting valve will put a small output of mist into the top chamber when the sensor reads that the humidity is too low.

- The humidity pump system will need to fit in the water system container box around the main root watering system.
- The GPH output of the humidity pump is 264, a readily available output that is cheaper than a powerful pump and can still do the job proven by recent tests.
- The single pipe will go through the box lid near the edge and the mist will be angled towards the middle of the top enclosure.

3 - Enclosure

The enclosure is the main assembly for the EcoGarden. It is comprised of two parts: the upper enclosure or lid and the lower enclosure. Both of these parts have different requirements and will be considered as two separate entities for the purpose of analyzing the required design specifications in depth, the design is shown in Figure 3.1.

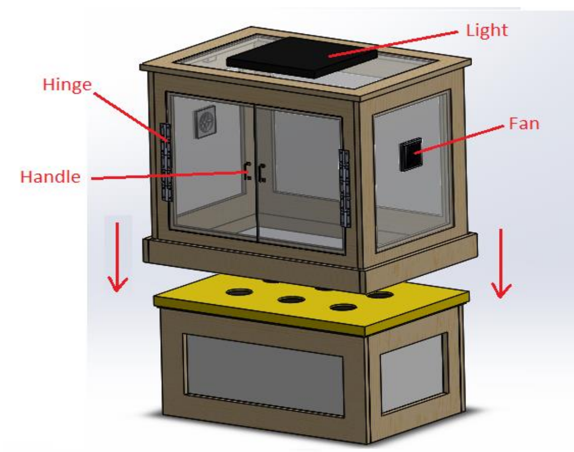

Figure 3.1: SolidWorks model of full enclosure assembly

3.1 - Lower Enclosure

The lower enclosure contains the growing apparatus as well as the watering system for the EcoGarden. Based on the functional specifications, the design for the lower enclosure was selected and shown in Figure 3.2.

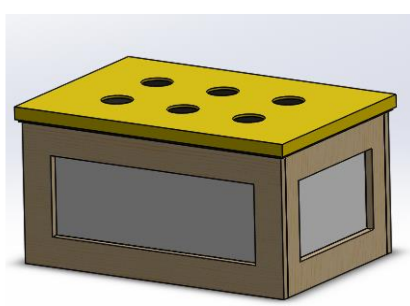

Figure 3.2: SolidWorks model of the lower enclosure

The first design consideration was what material the lower enclosure should be constructed from. When selecting the materials, the following options were considered.

- Plastic
- Wood
- Aluminium

After research and consideration, wood was selected as the main material for the enclosure. Wood was the ideal material for a number of factors:

- Cost
- Ease of construction
- Durability
- Sustainability
- Aesthetics

It was felt that the biggest advantage of using wood was the sustainability factor. Unlike plastic and aluminium, the wood in the enclosure is biodegradable and easily recyclable which reduces the waste at the EcoGarden's end of life. For increased product sustainably, reclaimed wood can also be used to construct the base. Another large consideration was the price, as wood in general is much cheaper than plastic or metal.

The second design consideration was to ensure the enclosure had sufficient dimensions to securely fit the tank. Upon purchasing a tank for the watering system, the box was built using the dimensions in Figure 3.3 (all units in centimeters).

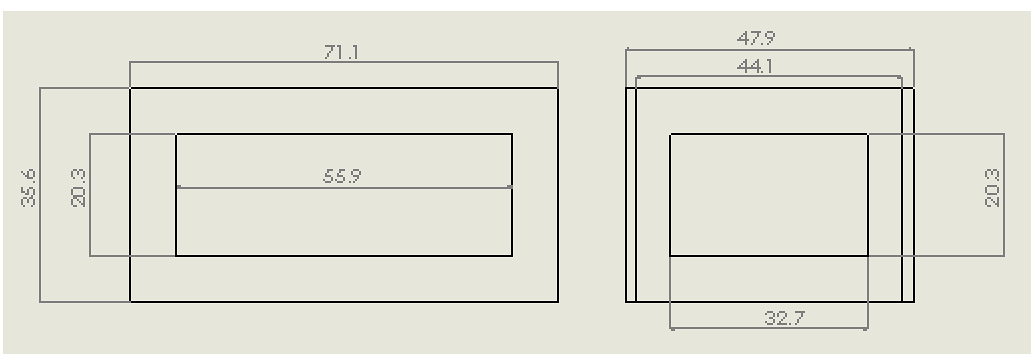

Figure 3.3: Dimensions of the lower enclosure

To tackle the issue of latching the top and bottom halves of the EcoGarden enclosure, a mechanical latch was selected to simplify the design. One consideration was using a sliding bolt locking mechanism, similar to the one used on doors and gates. It was determined that this design was simple and robust, but the drawbacks for were determined to be too great to be of benefit. Employing such a mechanism would make the overall design more difficult while potentially creating pinch points which could result in injury to the end user.

Another design considered was not having a latch system at all, rather have an outside lower rim on the enclosure such that the upper enclosure sits snugly inside. The rim system was selected instead of a latch because it simplifies the overall design and reduces the production cost of the EcoGarden. Furthermore, the upper enclosure will fit into the rim snugly so the assembly can withstand some minor force without coming detached.

3.2 - Upper Enclosure

The upper enclosure contains the access hatch to the growing chamber as well as hold the fans, temperature/humidity sensor and the growing light, as demonstrated in Figure 3.4.

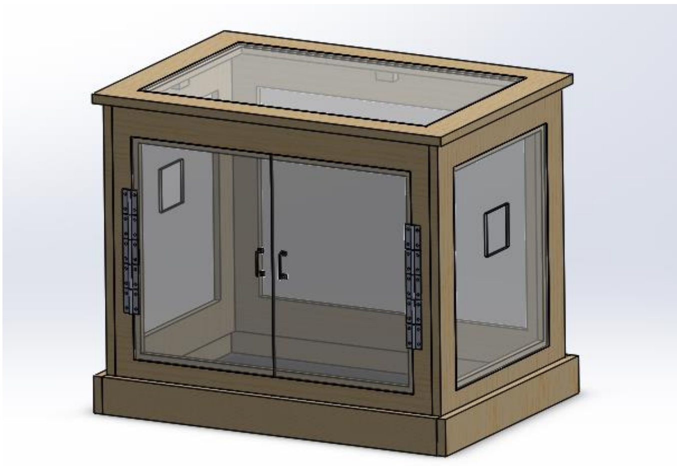

Figure 3.4: SolidWorks model of the upper enclosure

The first design consideration is where and how to mount the sensors and actuators will be mounted to the assembly. The first actuator considered was the LED growing light array. The light was mounted on the outside of the upper window of the enclosure for ease of access for any potential maintenance or service on the light array. Additionally, having the lights on the outside limits its exposure to the moisture inside the chamber, which could prevent any damage to the light array. The next consideration was where the pair of fans should be mounted in order to introduce fresh air into the growing chamber. As seen in the diagram above, the fans were mounted to the side panels to introduce an air flow from two directions and not obstruct the user's view into the chamber from the front and rear panels.

The second design consideration for the assembly was selecting the materials to construct it out of. Like the lower enclosure, the upper enclosure was constructed out of wood. However, unlike the lower enclosure, a clear material was needed so the growing chamber can be viewed by the user and allow the light from the growing light array through. There were a couple of materials up for consideration:

- Glass
- Clear acrylic panels
- Clear plastic sheets

After researching into the three options, the conclusion was that the acrylic plastic panels were the ideal material to construct the windows out of. Unlike glass and the clear plastic sheets, the acrylic panels offer a lower cost solution, simple to use and rigid structure ideal for the EcoGarden. It also determined that a major flaw in using glass is that it carries the risk of shattering and can be quite dangerous for users with young children or pets.

The final major feature on the upper enclosure was the access hatch or door to the growing chamber. When the door was designed, one goal was for the door to be opened without taking up too much space and still be able to preserve the humidity levels in the chamber. After research, only one design for the door seemed to fulfill both of these requirements, a double door configuration (see Figure 3.5).

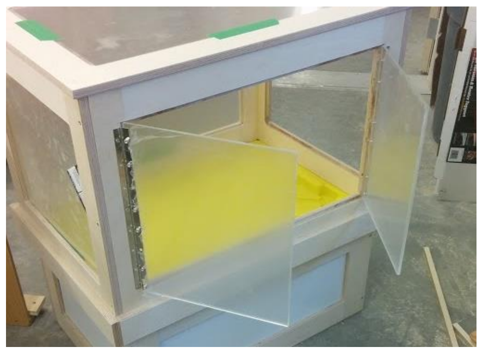

Figure 3.5: The fully assembled EcoGarden enclosure

An advantage of using two doors over one door is that opening the door takes half the space than a whole door. Clear acrylic panels were chosen to construct the doors for aesthetics and so the end user can easily view what is currently being grown in the chamber.

The upper enclosure was constructed using the dimensions in Figure 3.6(all units in centimetres):

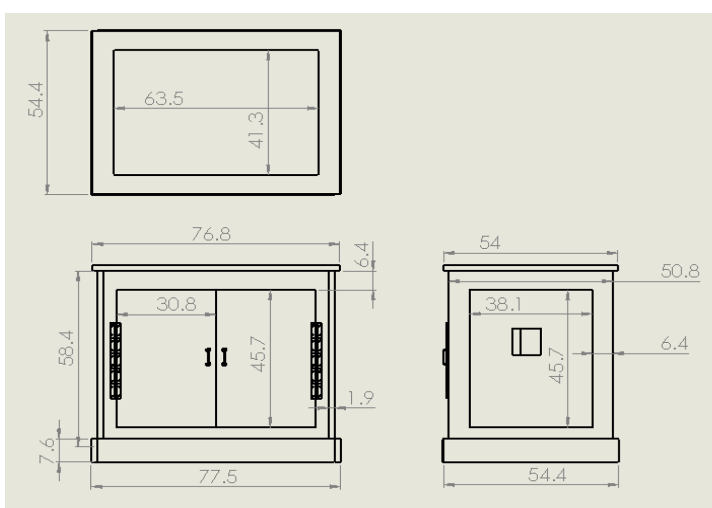

Figure 3.6: Dimensions of the upper enclosure

4 - Control System

The control system is the operation of all of EcoGarden's sensors and actuators according to the software logic uploaded to the Arduino. This section introduces the sensors and actuators used, as well as a description of the micro controller. Also included in this section is a design description of the control panel user interface and the hardware installation. Figure 4.1 is an illustration of the control system and its inputs and outputs.

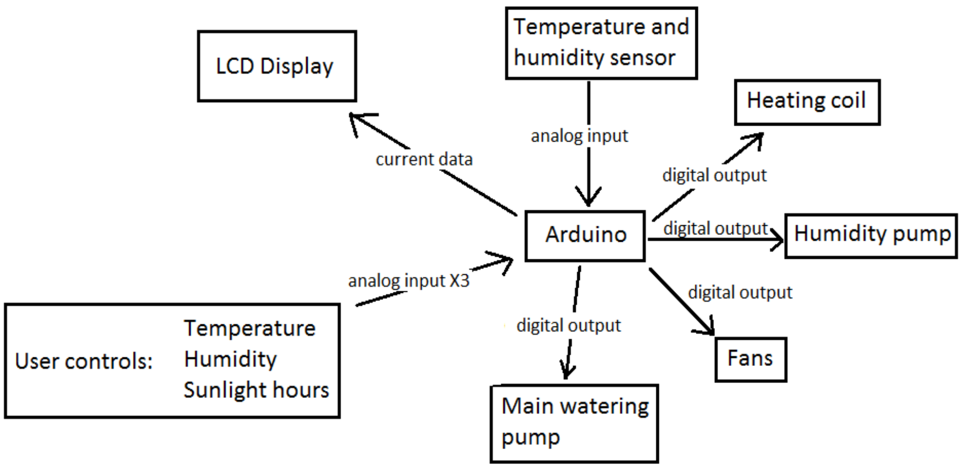

Figure 4.1: Control system diagram

4.1 - Sensors and Actuators

The Ecogarden system contains 6 main peripherals: a joint humidity-temperature sensor, a light source, a heat source, a watering pump, a humidity pump, and fans. Those peripherals can be seen in Table 4.2.

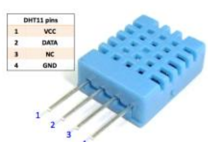 a)	 b)
 c)	 d)
 e)	 f)

Table 4.2: a) The temperature-humidity sensor [1], b) The LED array [2], c) The watering pump [3] d) The heating source [4], e) The humidity pump [5], f) One of two fans [6]

The sensor, shown in Table 4.2a above, is a DHT11 Humidity & Temperature Sensor. Its relevant specifications can be seen in Table 4.3.

Maximum current intake	2.5 mA
Required voltage	3 or 5 volts
Humidity Accuracy	1% Relative Humidity
Reliable Temperature Range	0-50°C
Temperature Accuracy	±5% Max
Maximum Sampling Rate	1 Hz

Table 4.3: Technical properties of the DHT11 Humidity & Temperature Sensor [7]

The sensor outputs a digital signal that can be translated by the Arduino. It was chosen for it's easy interfacing with the Arduino and because of its dual sensing capabilities.

The light source that's being used is the Erligpowht Indoor Grow Light, shown in Table 4.2b. It's a matrix of 225 LEDs, 165 650nm red lights and 60 465nm blue light, built into a plastic casing. It comes attached to an American standard power adapter. This array was chosen mainly due to it's light wavelength being conducive to plant growth as well as its relatively heatless operation.

The main water pump is a Blackwater 500 GPH Pump, shown in Table 4.2c. As the name suggests, it's capable of moving 500 gallons per hour. It's completely submersible and also runs off of a standard power adapter.

The humidity pump is a Blackwater 264 GPH pump, shown in Table 4.2e. It functions exactly the same as the main watering pump, except it's smaller. It can only produce 264 GPH, which is fine since it only has to move water through a single pipe. There were multiple alternatives that were considered for humidity control. The main consideration was a moving portion of the middle divider during the standard watering cycle to allow mist to enter the upper section. It was decided that controlling a separate pump was more reliable and was more cohesive with the rest of the system.

The heating unit is a modified foot heater. The original heater, that is shown in Table 4.2d, was quite large and included a fan. It runs off of a standard power adapter as well meaning it interfaces with the rest of the design well. Other heating mechanisms were considered, but this method was chosen due to the way it synergizes with the previously chosen control system.

The fans are hard plastic and are similar to those that are found on computer towers; they can be seen in Table 4.2f. They were chosen specifically for being light, durable, and water-resistant. They draw very little current and require very little power to run.

Most of the controlled peripherals in the system plug into a power bar that plugs into a wall; this design has two main benefits. The first is that it will allow the user to power the entire system by simply plugging it into a standard wall socket, helping usability considerably. The second is that it allows the arduino to control them simply by attaching a relay to each power cord to function as a switch. The alternative would have been to open up each peripheral and try to control each mechanism directly, which quickly becomes very complicated and generally unreliable.

The two that aren't directly on the main power bar are the sensor and the fans. Both of these get power through the Arduino, which is fine to do since they require so little in the way of current and voltage.

4.2 - Microcontroller

The final design uses only a single Arduino Uno, even though original plans included a coupled Raspberry Pi. A comparison of the two can be seen in Table 4.4.

Feature	**Arduino Uno**	**Raspberry Pi**
Cost	$30 plus accessories	$35 plus accessories
Graphics Driver	No	Yes
Internal storage	Flash memory SRAM	SD card
Full OS?	No	Yes
I/O max voltage level	5V and 40mA	3.3V and 50 mA
Analog I/O	Yes	No
Power supply required?	No	Yes
Networking	No network connectivity	Built-in ethernet port

Table 4.4: Arduino vs. Raspberry Pi comparison

As shown in Table 4.4, the Raspberry Pi was integral to the original design that included a higher level GUI and an internet connection option. Both of those features were removed in the current design for various reasons. The internet connection requirement was changed because it's unlikely that a user would need to change the condition settings remotely. The main reason for changing conditions is due to a new type of plant in the enclosure and that can only be done when physically present. The GUI change is discussed further in section 3.3, but essentially a high quality screen would be unnecessary for this device. The final strike against using a Raspberry Pi was that it has no analog inputs to deal with the prefered user controls, discussed in section 3.3 as well. The design could therefore be streamlined and simplified by removing the second microcontroller completely. This brought the added benefit of saving money that would have been spent on a Raspberry Pi kit (>$35) and of a graphical display (>$70).

The Arduino is capable of handling all of the required peripherals as well as the entire user interface without needing to pass off to another microcontroller. If more features and peripherals would be added in the future, it's likely that a second Arduino would need to be used as the prototype design maximizes the capabilities of a single Arduino.

4.3 - Control Panel User Interface

There were a few key attributes that the user interface design was required to include. It had to be intuitive, it had to be relatively simple, and it had to provide adequate feedback. The final design of the control panel can be seen in Figure 4.5 below.

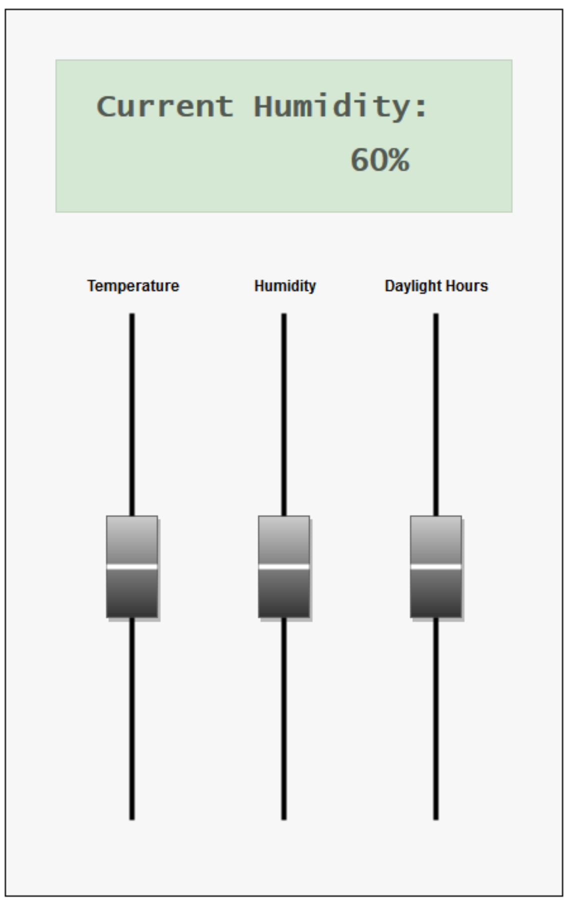

Figure 4.5: General design of the control panel

The control box on the prototype is 20cm x 12cm x 5.5cm. The upper screen is a two-line LCD screen that will cycle through all of the current settings and current conditions. The user input is limited to three parameters that are controlled by sliders. Each slider is a 7.5 cm, 1kΩ sliding potentiometer whose output feeds directly to one of the Arduino's analog inputs. Each analog input converts voltage values from 0 to 5 volts to digital values from 0 to 1023, so the circuit diagram looks like Figure 4.6.

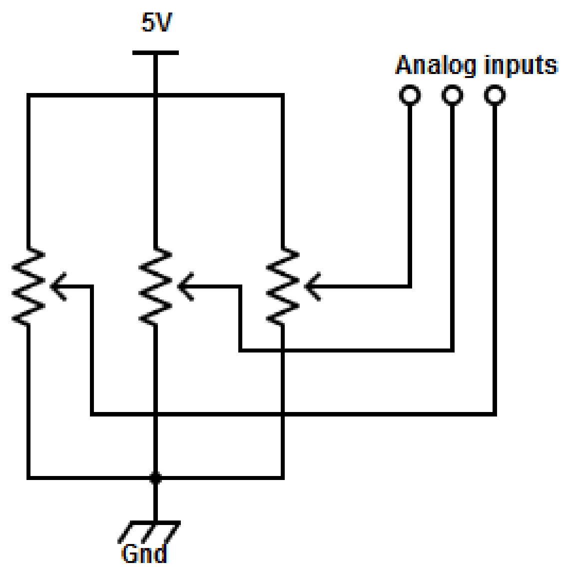

Figure 4.6: Circuit diagram of the user interface

For the prototype design the three user-controlled parameters are temperature, humidity, and amount of light. For future iterations of the product additional parameters, like amount of water and amount of air, could be added relatively easily if so desired. Each slider varies from minimum to maximum allowable settings that are constrained by the current peripherals described in section 3.1. If required, it would be fairly straightforward to implement more accurate inputs by simply increasing the length of the potentiometers.

Other designs that were considered for the control panel included various buttons configurations. For example, one button for each parameter and pressing the button cycled through some low, medium, and high pre-sets. Or a variation of the same concept where each setting for each parameter had its own button to select it, so an array of 9 or so individual buttons. Another idea was buttons like to be used like enter and arrow keys to select a parameter then alter it. The main problems with all of these designs was that they either weren't intuitive or didn't provide adequate user feedback. The brilliance of the slider design is in its simplicity, especially once it's labelled. You slide up to increase a parameter, down to decrease it, and the user has instantaneous feedback on their input.

The other strong design contender was a graphical LCD screen with an advanced GUI. This was one of the earliest ideas and required a Raspberry Pi to run the higher level graphics. The advantage of this higher quality screen would be that all of the current conditions and settings could be displayed at once along with more elaborate and user friendly graphics. There are two main issues with this design, the first is inputs. Raspberry Pis do not have analog inputs so to use the slider design the input signal would have to be read through the Arduino which would introduce up to a second-long latency period. In general people are not patient with electronic systems, so that long of latency would likely just frustrate users. The solution to that problem would be to input directly to the Raspberry Pi digitally, so a keyboard or something similar. That raises its own usability problems it would be difficult to ensure the interface was intuitive. So the obvious design is to have graphical sliders that can be controlled via touch screen, which introduces the second large issue with this design. A graphical screen, or touch screen, is much more expensive than a small LCD screen. Not only the direct cost, but since the Arduino can run a small LCD screen itself the second microcontroller is unnecessary which reduces the overall cost of the system drastically. More discussion on this can be found in section 3.3. It's important to note that the system mainly loses aesthetic features by using a two-line screen, so user feedback and usability is only slightly affected.

4.4 - Wiring and Hardware Installation

In order to properly install all of the sensors and actuators careful planning must be taken to organize the wiring and to allow access to components such as the relays. All of the main power cords will lead to a main power bar installed on the back panel.

Also installed on the back panel will be the relays that allow the Arduino to turn the actuators on and off. The cord casing will be opened up and expose the white ground wire while leaving the black current source "hot" wire inside the casing. The relays will be placed in series with the ground wire to open and close the circuit like a switch when the Arduino provides a high digital output signal. An illustration of this concept is shown in Figure 4.7.

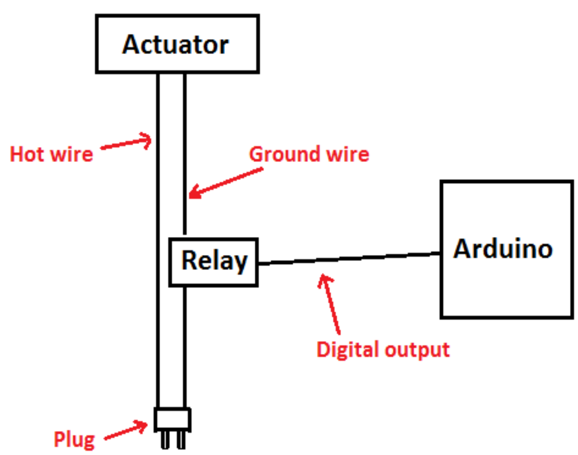

Figure 4.7: Relay wiring

The relay that was selected is capable of running switching 110V AC current using the 40mA maximum current draw from the Arduino output pin.

The fans do not use a 110V AC supply. They run off a 5V DC source with 400mA current draw. Since a regular outlet plug is not available and the current draw is too high for the Arduino pin, a 5V battery with another relay was implemented in a very similar configuration to Figure 4.7.

5 - Software

The Arduino code runs all the sensors and actuators. As described in section 3 it takes in user commands from the control panel. Based on the user input, the fans, humidity pump and heat source are turned on and off based on sensor feedback to keep the environmental conditions within the specified range. The environmental conditions that EcoGarden controls are temperature, humidity and sunlight hours. The pump is controlled by the Arduino's inner clock.

The main code is a loop that runs through about once every second. Each of the following sections represents a function called once a loop. The functions are all designed to not cause significant delays, so the sensor readings will be refreshed frequently.

5.1 - Select Variables

The purpose of the Select Variables function is to determine the target range for temperature, humidity and sunlight hours. To do this the user will adjust the three sliding potentiometers of the control panel. The analog input reads a number between 0-1023 that is proportional to the resistance of the potentiometer. This reading will be mapped to software variables that mark the targets that the actuators work to achieve.

This function will not cause the algorithm to halt and wait for user input. It will simply check the current settings of the slider potentiometers and update the software variables accordingly if they have changed since the last loop

Temperature (analog input 0)

- The user input will correspond to a minimum temperature. The maximum temperature will remain at 30 degrees Celsius. This is because heat can be easily added but not reduced below the ambient temperature outside of EcoGarden.
- The range that the minimum temperature can be set to will be 15-29. To do this, the analog input is mapped to 0-1023 linearly to 15-29.

$$minTemp = (0.01955) * analogRead(0) + 10$$

Humidity (analog input 1)

- The user input corresponds to a target range of humidity of 10% for example it may be set to 66%-76%.
- The lowest allowed selection is 40%-50% and the maximum is 85%-95%. To do this, the analog input is mapped to 0-1023 linearly to 40-85 for the minimum and 50-95 for the maximum.

$$minHumid = (0.043988) * analogRead(1) + 40$$
$$maxHumid = (0.043988) * analogRead(1) + 50$$

Sunlight Hours (analog input 2)

- The final user input for sunlight hours. The user input marks the amount of hours per day that the led lights will be activated.
- The lowest value is 8 hours and the maximum is 24 hours. To do this the analog input 0-1023 is mapped linearly to 9-24.

$$sunHours = (0.01564) * analogRead(2) + 9$$

5.2 - Humidity Control

The humidity control function is designed to raise the humidity, lower it or keep it the same depending on if the current humidity reading falls in or out of the user specified range.

- If the current humidity is too high, the fans are activated to clear out humidity at a gradual rate. In later loops once the humidity has fallen into the acceptable range the fans will be deactivated.
- If the current humidity is too low the humidity pump is activated, spraying a small amount of mist into the top chamber.
- If the humidity is correct then both the fans and the humidity pumps are deactivated

5.3 - Temperature Control

Temperature control primarily operates around maintaining an adjustable minimum temperature, while making sure that it does not rise above a constant maximum. It was designed this way because heat can be added to the system but cannot be lowered below the surrounding air temperature. Cooling systems are far too expensive to implement for this iteration of EcoGarden.

- If the current temperature is too low, the heater is activated. It remains on until in subsequent loops the temperature has risen into above the user specified minimum.
- If the current temperature is above the maximum, then the fans are activated to blow out some of the over-heated air.
- If the temperature lies within the correct range, then both the heater and the fans are deactivated

5.4 - Pump Control

The pump is set to operate based on the internal clock of the Arduino to keep the roots always wet, but not constantly be spraying water unnecessarily wasting power. The current plan is to run the watering pump for the first 10 seconds of every minute, but can adjust it to find the perfect balance.

5.5 - LCD Display Screen

The main purpose of the LCD screen is to display the following fields to the user:

- Chamber humidity
- Chamber temperature
- System time

One design challenge is how the above screen can be displayed to the user. It was determined that displaying one field at a time on the screen then cycling through them on timed intervals was the best solution to this problem. This way, the screen would not be cluttered up with too much information, overwhelming the user. The time interval between cycling the display fields needed to be chosen such that:

- The system does not have to query the system constantly
- The user has sufficient time to read the information on the screen

The ideal time period for this was determined to be

6 - Sustainability and Life Cycle

Regarding the significance of sustainability in the world, the EcoGarden's potential environmental impact was heavily considered during the design phase. As a guideline for determining sustainability, McDonough and Braungart's Cradle to Cradle (C2C) certification criteria was consulted [8]. By applying this Cradle to Cradle design methodology, the EcoGarden will leave a reduced footprint on the environment from initial production to final scrapping. The C2C certification criterion considered were as follows:

When selecting the materials and parts for the EcoGarden assembly, the guideline concerning material health criteria played a significant role. Parts that contained harmful materials (i.e. lead, mercury, etc..) were never considered for usage in product assembly, even if they had better performance or properties than their C2C compliant counterparts. As a result, each of the materials used in the EcoGarden's construction was verified to not contain any harmful elements, from the electronics to the enclosure.

Many components in the EcoGarden are easily recyclable and disposable at the end of their life. Wood was used for the upper and lower enclosures of the assembly because it was easily recyclable or could be reclaimed or recycled wood. The acrylic panels can also be melted down and reused for other products. By using these materials, the EcoGarden's components can be reused in new products and only a minimal amount will be sent to the landfill.

At this stage of development, sustainability in the assembling procedure has been achieved, based on the primary principles of C2C design. Since the main stages of assembly can be performed manually using power tools, large industrial processes will not be necessary. Omitting large industrial manufacturing procedures leads to a reduction in waste, since these processes consume massive amount of energy and produce a large amount waste.

In summary, the EcoGarden's design is sustainable from its production to its eventual scrapping at the end of the product's life. The C2C criteria outlined by McDonough and Braungart heavily influenced the selection of materials and fabrication techniques.

7 - Conclusion

The primary goal of the development and design phase of the EcoGarden was to meet all of the desired functions outlined in the functional specification document. Additionally, the process also had the following goals:

- Be cost effective for the end user
- Allow the end user to operate easily
- Be compliant with C2C sustainability criteria

Each of these elements were heavily considered during the whole design phase. These design considerations were employed to best achieve the overall the goal of providing a valuable product for those who cannot access fresh healthy produce at a reasonable price.

References

[1] Stak. *DHT11 Humidity and Temperature Sensor*. 2016 [Online]. Available: http://stak.com/DHT11_Humidity_and_Temperature_Sensor [Accessed: 3-March-2016]

[2] Erligpowht. *Erlighpowht 45W LED Red Blue Indoor Garden Plant Grow Light Hanging Light*. 2016 [Online]. Available: http://www.amazon.com/Erligpowht-Indoor-Garden-Plant-Hanging/dp/B00S2DPYQM/ref=pd_sim_86_6?ie=UTF8&dpID=51TEstig-FL&dpSrc=sims&preST=_AC_UL160_SR160%2C160_&refRID=13B50GBZ6RN64T4PN8NQ [Accessed: 3-March-2016]

[3] Blackwater Pumps. *Blackwater 500 GPH Pump*. 2016 [Online]. Available: http://www.brewtek.ca/en/pumps/submersible-pumps/blackwater-500-gph-pump-7004 [Accessed: 3-March-2016]

[4] Khoshnavaz, Anita. *Heater*. 2016.

[5] Blackwater Pumps. *Blackwater 264 GPH Pump*. 2016 [Online]. Available: http://www.brewtek.ca/en/pumps/submersible-pumps/blackwater-264-gph-pump-7002 [Accessed: 3-March-2016]

[6] Ganzert, Eric. *Circulation Fans*. 2016.

[7] D-Robotics UK. (2010). *DHT11 Humidity & Temperature Sensor* [Online]. Available: http://www.micropik.com/PDF/dht11.pdf

[8] McDonough Braungart Design Chemistry. *Certification Criteria* [Online]. Available: http://c2cproducts.com/detail.aspx?linkid=2&sublink=9

Appendix A - Test Plan

Test	Result		Comments
Physical Systems			
Watering System			
Water pumps through the pipe system	☐ Pass	☐ Fail	
Mist covers all potential roots	☐ Pass	☐ Fail	
Humidity System			
Water pumps through the pipe	☐ Pass	☐ Fail	
Mist sprays into upper enclosure	☐ Pass	☐ Fail	
Lighting System			
Lights turn on/off	☐ Pass	☐ Fail	
Heating System			
Heat turns on/off	☐ Pass	☐ Fail	
Circulation System			
Fans turn on/off	☐ Pass	☐ Fail	
Control System			
Inputs			
Able to read temperature and humidity conditions	☐ Pass	☐ Fail	
Able to read slider inputs	☐ Pass	☐ Fail	
Outputs			
Waters roots on a cycle	☐ Pass	☐ Fail	
Lights run on a daily cycle based on the slider input	☐ Pass	☐ Fail	
Temperature and humidity change based on the inputs	☐ Pass	☐ Fail	
Displays to the screen	☐ Pass	☐ Fail	