Smart Garden System

DESIGN DOCUMENT

Team: 25
Advisers: *Diane Rover*

Team Members/Roles:
Bryanna Adamson - *Team Organization Lead*
Julia Condon - *Application Design/Frontend Lead*
Jasen Helsel - *Testing/IoT Lead*
Sarah Schoenke - *Client/Communication Lead*
Devon Sindt - *Embedded Systems Lead*
Jake Thomae - *Assistant Embedded Systems*
Nick Vaughan - *Scrum Master (AGILE)/Backend Lead*

Team Email: *http://sdmay22-25.sd.ece.iastate.edu*
Team Website: *sdmay22-25@iastate.edu*

Revised: *12/01/2021*

Executive Summary

Development Standards & Practices Used

List all standard circuit, hardware, software practices used in this project. List all the Engineering standards that apply to this project that were considered.

- IoT Standards
- Bluetooth Standards
- Wifi/ethernet Standards

Summary of Requirements

The Smart Garden Monitoring System (SGMS) requires many hardware components as well as a software to display the analyzed data to the user.

Functional:

- Monitor temperature of the environment in order assure plant health
- Time the growth of the plant
- Water the plant using a timed water pump
- Give the plant light using an led plant light

Resource:

- Funds to purchase Seeds, & Soil
- A location to host our project
    - The allocated amount of space offered for this project may constrain the size of our prototype
- This project must be completed by the end of April 2022(constraint)
- Sensors and circuitry to observe the plant and communicate findings to the user

Economic/Market:

- This project must cost under $1000 to create and operate (constraint)

Environmental:

- A small greenhouse will be constructed to avoid temperature variations while testing the prototype

User Interface:

- The user should be able to see all signals charted clearly with an automatically analysis of the plant's health and remaining time until maturity if starting from a seed

Other:

- Soil should be treated properly to reach a state of chemical balance and obtain maximum plant growth

Applicable Courses from Iowa State University Curriculum

List all Iowa State University courses whose contents were applicable to your project.

S E 319 - Construction of user interfaces

COM S 227 - Object-Oriented Programming

CPR E 281 - Digital logic

CPR E 288 - Embedded Systems 1

E E 201 - Electrical Circuits

S E / COM S 309 - Software development practices

New Skills/Knowledge acquired that was not taught in courses

List all new skills/knowledge that your team acquired which was not part of your Iowa State curriculum in order to complete this project.

Use of open source libraries, api's

Getting further experience with html, css and other front end applications while we got some experience in se 319 there is only that one course about it so we are learning more about it as we go

We have to learn how to connect software to hardware, while some classes have touched on this, we will be getting more practice with it as we complete this project.

Table of Contents

List of figures

1 Team

1.1 Team Members

Sarah Schoenke, Devon Sindt, Julia Condon, Jake Thomae, Nick Vaughan, Jasen Helsel, Bryanna Adamson

1.2 Required Skill Sets for Your Project

1. Experience in Agile development
2. Basic Programming languages we might use: Java, C, Python.
3. Knowledge of GitLab
4. Possible knowledge of app development
5. Knowledge of Embedded System
6. Basic knowledge of plants
7. Knowledge of testing

1.3 Skill Sets covered by the Team

1. Sarah, Nick, Julia
2. Sarah, Nick, Bryanna, Jake, Julia, Jasen
3. Sarah, Nick, Bryanna, Jake, Julia, Jasen, Devon
4. Sarah, Bryanna, Jake, Julia
5. Jake, Julia, Jasen, Devon
6. Sarah, Bryanna, Julia
7. Jasen, Sarah, Devon

1.4 Project Management Style Adopted by the team

As a team, we decided that we want to do an Agile style of project management.

1.5 Initial Project Management Roles

Sarah Schoenke: Client Interaction / communication

Jasen Helsel: Testing/IoT

Nick Vaughan: Scrum Master (AGILE)/Backend

Jake Thomae: Assistant Embedded System/Team Mascot

Bryanna Adamson: Team Organization

Devon Sindt: Embedded Systems

Julia Condon: Application Design

2 Introduction

2.1 Problem Statement

Growing plants can be not only a time consuming task, but a very sensitive one as well. With our smart garden monitoring system, we hope to automate as much of the plant growing process as feasibly possible in order to make it more consistent and manageable for the user. Doing so will save time and money, and will in turn make growing plants a more accessible task for a wide range of potential users.

2.2 Requirements & Constraints

The Smart Garden Monitoring System (SGMS) requires many hardware components as well as a software to display the analyzed data to the user.

Functional:

- *Monitor temperature of the environment in order assure plant health*
- *Time the growth of the plant*
- *Water the plant using a timed water pump*
- *Give the plant light using an led plant light*

Resource:

- *Funds to purchase Seeds, & Soil*
- *A location to host our project*
- *The allocated amount of space offered for this project may constrain the size of our prototype*
- *This project must be completed by the end of April 2021(constraint)*
- *Sensors and circuitry to observe the plant and communicate findings to the user*

Economic/Market:

- *This project must cost under $1000 to create and operate (constraint)*

Environmental:

- *A small greenhouse will be constructed to avoid temperature variations while testing the prototype*

User Interface:

- *The user should be able to see all signals charted clearly with an automatically analysis of the plant's health and remaining time until maturity if starting from a seed*

Other:

- *Soil should be treated properly to reach a state of chemical balance and obtain maximum plant growth*

2.3 Engineering Standards

Some common standards we may need to follow are IoT standards regarding security and communication protocols. These standards may also coincide with bluetooth standards which we could use for short distance wireless communication and wifi/ethernet standards which would be used for long distance communication. The short distance bluetooth may be useful for wireless probes or other tools/features, whereas the long distance communication may be useful for controlling the system with a cellphone or computer that is located in a different location.

2.4 Intended Users and Uses

Due to not having a client, despite our best effort of trying to get one, the only real users of this project at the current time will be us. That being said, the design of something like this would be very beneficial to be able to produce food in a small environment with a busy schedule. Our design has the potential to be used in the future to a food shelf or a family that needs it. All this includes that it will help the user grow healthy food which will: save on money, time, and lead to healthier available food options. They would use it to grow food for themselves or those who need it but do not have time or a green thumb to grow it themselves.

Beneficiaries from our project:

- On a smaller scale:
    - Hobby Gardeners
    - Educational tool
- On a larger scale:
    - Food insecure populations
    - Humanitarian efforts

Use case:

- Grow plants for hobby gardeners
- Work as an educational tool for students
- Grow healthy food for a family in need, shelter, or food shelf
- Present healthier food options to the user
- Help grow a space, plants, in which the user takes comfort in
- It should be able to be used to track and monitor the levels of water, light, temperature and growth time of the plant.
- Display crucial information on the application, and ability to adjust based on the information that is being given.
- If one goes into the application they can check the level of water, and light that the plant has been given for the past week
- They are able to track, write down on the application if they can see new growth on the plant to track if the light and water levels are working

3 Project Plan

3.1 Project Management/Tracking Procedures

Which of agile, waterfall or waterfall+agile project management style are you adopting. Justify it with respect to the project goals.

We'll be using AGILE as our project management style for this project. Tasks will be posted to 2 week incremented sprints & assigned to group members as the group sees fit. This will make breaking down tasks easy for the group. Having a way to modularly assign tasks will be crucial for any development the group does, as well as research/physical tasks needed for the project. AGILE will also help keep the group on track with its 2 week sprints.

What will your group use to track progress throughout the course of this and the next semester. This could include Git, Github, Trello, Slack or any other tools helpful in project management.

The group will use a few different softwares including: Gitlab, Monday.com, & Google Drive. Gitlab will be used for our codebase, allowing any dev work to be shared amongst the group. Monday.com is our AGILE tool, equipped with a kanban board, sprint planner, & other useful tools. Finally, google drive will be used to share any design documents, canvas assignments, & other documentation the group compiles.

3.2 Task Decomposition

Task 1

- **Determine hardware requirements**
  - **Research different smart greenhouse designs**
  - **Decide on design for our test bed**
  - **How many and types of plants we want to test**
  - **Decide on sensor placement based on what components we are monitoring**

Task 2

- **Determine software requirements**
  - **Database**
    - **What do we want to monitor**
    - **How often do we want the database to pull data**
  - **App**
    - **What do we want the app to show**
    - **Decide how the app shows it**
    - **Decide on different ways the user can access (i.e. app, website)**

Task 3

- **Cost analysis**
  - **Fixed costs: Lights, sensors, greenhouse structure, soil**
  - **Variable costs: Plants, power costs(i.e. electricity, water)**

Task 4

- **Create proposal for Iowa State University**
  - **Determine optimal design**
  - **Create final design documents**

3.3 Project Proposed Milestones, Metrics, and Evaluation Criteria

In an agile development process, these milestones can be refined with successive iterations/sprints (perhaps a subset of your requirements applicable to those sprint).

- completion of a rev 1 for hardware design, allowing us to start developing software for this hardware.
- software implementation to get us to the point we have a user interface and can start adding functionality to other hardware and software components such as sensors and long distance wireless communication.
- individual sensor hardware and software will each be its own milestone.
- reaching a MVP (minimum viable product).
- creation of a working database.
- creation of a working phone app to have a user friendly interface.

3.4 Project Timeline/Schedule

Week		1	2	3	4	5	6	7	8	9	10
Date		10/3 - 10/9	10/10 - 10/16	10/17 - 10/23	10/24 - 10/30	10/31 - 11/6	11/7 - 11/13	11/14 - 11/20	11/21 - 11/27	11/28 - 12/4	12/5 - 12/11
Task	Task 1: Hardware Design	■	■	■							
	Task 2: Software Design				■	■					
	Task 3: Cost Analysis						■	■			
	Task 4: Proposal								■	■	■

Figure 1:

3.5 Risks And Risk Management/Mitigation

Task 1 - potential risks while determining hardware requirements:

- **Researching different smart greenhouse designs:**
  - **Risk factor: 0** - There is no risk in looking at other designs for inspiration.
- **Settling on a testbed design:**
  - **Risk factor: 0.5** - The initial testbed design may end up changing quite a bit as testing progresses.

  - **Risk mitigation:** Although there is nothing we can do to mitigate risks prior to testing, we will at least be able to correct any design flaws when we do begin testing.
- **Choosing plant types:**
  - **Risk factor: 0.2** - If any problem arises with a certain type of plant, we are still able to choose another.
- **Sensor layout:**
  - **Risk factor: 0.2** - Like most of the project, sensor layout is subject to change as the project progresses. There is little risk other than any problems that may go unforeseen later into the design process.

Task 2 - potential risks while determining software requirements:

- **Database design:**
  - **Risk factor: 0.2** - The database design can be changed at any point in the process, so there is no major risk.
- **App design:**
  - **Risk factor: 0.2** - As with the database design, the app design can also be changed at any time.

Task 3 - potential risks during cost analysis:

- **Fixed costs:**
  - **Risk factor: 0.4** - Because we will compile our list of fixed costs before ordering everything, we should be able to make sure we stay under our budget. A potential risk is finding out later that we need to order a new part, however this should not be too much of an issue.
- **Variable costs:**
  - **Risk factor: 0.5** - The variable costs will be the hardest to control, as the more we are able to optimize our design, the less power we will consume.
  - **Risk mitigation:** We need to spend a lot of time working down the variable costs in the late stages of the design process.

Task 4 - potential risks while creating a proposal:

- **Determining optimal design:**
  - **Risk factor: 0.4** - As designing our project will be an iterative process, we may make mistakes in determining the optimal design before enough testing has been done.
- **Finalizing design documents:**
  - **Risk factor: 0.1** - As long as we follow good design practices, there should be no risk.

3.6 Personnel Effort Requirements

Task	Summary	Person-hours
Determine Hardware Requirements	Researching/designing greenhouse testbed for project including sensors and produce	~30 hours
Determine Software Requirements	Researching software requirements to utilize hardware in greenhouse	~30 hours
Cost Analysis	Estimating costs for hardware/software requirements	~3 hours
Create Proposal for ISU	Brainstorming and designing proposal for design	~10 hours

Figure 2:

Task	Summary	Person-hours
Creating Test bed	Crafting physical greenhouse testbed to grow and observe progress of plants	~6 hours
Installing Testbed Sensors	Place sensors around testbed to observe variables	~3 hours
Integrating sensors into microcontroller	Programming microcontroller to read sensor values	~8 hours
Programming Microcontroller	Programming microcontroller to react to sensor values i.e. triggering heaters/fans and water pumps	~15 hours
Installing environment gadgets	Place pumps and fans/heaters around testbed to enable resource manipulation	~8 hours

Figure 3:

3.7 Other Resource Requirements

- **IKEA furniture -> that can be turned into a greenhouse/grow box**
    - **Depending on the greenhouse/furniture chosen to be our green house we will need glass**
- **optional growing information for the plants**
- **Camera**
- **Sun/grow lamp**
- **Timer for the lamp**
- **Camera**
- **Plants**
- **Moisture monitoring equipment**
- **Water hose**
- **Water tank holder**
- **Thermometer**
- **Soil**
- **Plant food**
- **Identify a platform and language for the application**

4 Design

4.1 Design Context

4.1.1 Broader Context

Area	Description	Examples
Public health, safety, and welfare	The users of this product will be able to grow healthier foods right from their own home. This also includes being able to share that food with their neighbors and family.	An example includes: being able to grow food and make healthier measles rather than having to go out and get fast food to be able to afford food for their families.
Global, cultural, and social	Growing food and being able to supplement it relates to many cultures, religions, and values to be able to name them all. It also strengthens families and communities when that resource can be shared with others Which in turn helps make those ties that are needed to survive in our world.	If a family has leftover food they can share with that neighbor who just moved to town and knows nobody this creates a community and friendship that was not previously available.
Environmental	The impact our smart garden system will have is being able to grow your own produce which helps lessen plastic waste that otherwise could be used. It also will have an electricity timer so that the light is not powered on in order to preserve those resources.	By using it families will no longer need to buy as much veggies at the grocery store and that will save on plastic that goes back in the environment/ocean
Economic	Our product will help lessen the need for users to buy food from the market. It will also be economically efficient for our client and will be a good saving time so that it doesn't cost them anymore than needed.	It will need to be able to efficiently and safely grow food to be able to provide what is needed. It will also need to be able to turn the light off when needed to save on electricity.

Figure 4:

4.1.2 User Needs

Note: We really only have one user group, those who are purchasing the product.

Purchaser's of this product need a way to grow food in their own home because of the rising expense of food.

4.1.3 Prior Work/Solutions

Smart Garden Monitoring Ideas

Things plants need to grow:
Sunlight, water, air, proper temperature, nutrients

A guide on what plants need:
What Makes Plants Grow?

University of Florida. (1997, May). *3 what makes plants grow? plant ... - university of Florida*. What Makes Plants Grow? Retrieved October 17, 2021, from https://edis.ifas.ufl.edu/pdf/4H/4H36000.pdf.

Things to measure in garden:
- Sunlight
  - UV lights for plant, simulate day + night if needed
- Water
  - Soil moisture, some sort of watering system to keep regulated (misting system?)
- Air
  - Humidity
  - Temperature, need like a tiny space heater or temp regulator
- Nutrients
  - In soil (Could be hydroponic garden, more complex but might be easier to maintain in the long run)
  - Ph of soil
  - Oxygen in soil

- ➔ All of these are subject to change, depending on what you are growing.
  - ◆ If we can find optimal numbers for a variety of plants, it will be easy to control what we can control
- ➔ Might be best to create a system that can take values for measurables and adjust behavior based on that.

Best produce to grow:
1. Lettuce - Its hardy, grows quickly, and we do not need to do anything to prep the soil
2. Radishes - fastest growing vegetable so if we mess up, there's a quick turnaround time
3. Kale - hardy, grown in a variety of temperatures. Can be harvested at many different stages depending on wh taste you are going for

4.1.4 Technical Complexity

Our project involves multiple parts to make a product that is sufficient for consumer use, causing it to be complex. The system will require knowledge from multiple areas including embedded systems, user interface development, networking, and web application development. All of these components need to be combined with each other, using communication between team members, to create an adequate product. This project is plenty complex for the project's timeframe.

4.2 Design Exploration

4.2.1 Design Decisions

We have Decided to use Ikea Furniture as a greenhouse. We preferably want one that has a shelf on it so that we can split up the different layers.

We want the design to have a camera that can visibly see the plants/thermometer. As well for a heat/light source we will want a heat/sun lamp to be able to give the proper lighting it will need. A way to turn off the light threw the website without having to be at the greenhouse. There will also be a setting for a timer so that you can just set it and it will turn off or on via the website.

The website application will monitor -> water, temperature and light. There could be an option to water with fertilizer should the customer choose to wawn to do that but we will not be monitoring that, due to how many specific plants need different times and types of fertilizer.

For the backend of the website we will use a combination of node, and mysql, for the front end we decided to go with html and css to be able to design the website. We will be writing the code on html.

We will need to make a decision on the kind of microprocessor that we will be using for this project, that decision will be decided if not this week then next after a bit more research on our options.

4.2.2 Ideation

We identified potential project options through the combination of **rapid ideation**, and **round-robin brainstorming**. During our group meetings, we enlisted these techniques to help generate ideas. Starting on a random member, we'd go through each member of the group asking for thoughts on a particular topic. These ideas are written down in the meeting notes, and the process is repeated until we have a wide range of ideas. The team then votes on which ideas are best, pusing majority-decision ideas through.

One thing we considered when trying to come up with our final design was what components of the smart garden we wanted to monitor. There were many different options, some more complex, some less. Below are 5 of the options we considered.

1. We considered just monitoring when the plant needed to be watered based on a timed schedule and what type of plant it was
2. We considered monitoring the soil moisture to decide when the plant would be watered
3. We were thinking of going more complex and monitoring multiple different things like humidity, temperature, soil moisture, water times, fertilizer, and sunlight.

4. Something that was also considered was not making it automatic and leaving it to the user based on the readings from the sensor that were displayed on our app.
5. Lastly we considered making everything automatic for the user, they would enter the type of plant and then the green house would take care of it instead of a semi automatic greenhouse.

4.2.3 Decision-Making and Trade-Off

Demonstrate the process you used to identify the pros and cons or trade-offs between each of your ideated options. You may wish you include a weighted decision matrix or other relevant tool. Describe the option you chose and why you chose it.

When trying to identify pros and cons of each option we really just thoroughly discussed it as a group. Each of us spent time researching different options and designs for smart gardens and half of us know quite a bit about plants, so with that combined knowledge we were able to have a discussion about the different pros and cons for each option we came up with. We decided on the option of just monitoring temperature, water and light because the pros for this more suited our wants and needs for this project. We wanted a design that would be able to be used by anyone who wanted to try to own a smart garden in their home. It was also simple enough that it could be used for almost any plant because all plants need some form of these three things.

4.3 Proposed Design

We researched and tried to find the different ways that we could go about implementing a greenhouse. Along with that some of us have researched and tested out some different software that is available to create the front end of the website, we ended up deciding that a combination of html and css was the way to go for us.. Two of us also use Agile in our work so that could be considered a form of testing as well.

4.3.1 Design Visual and Description

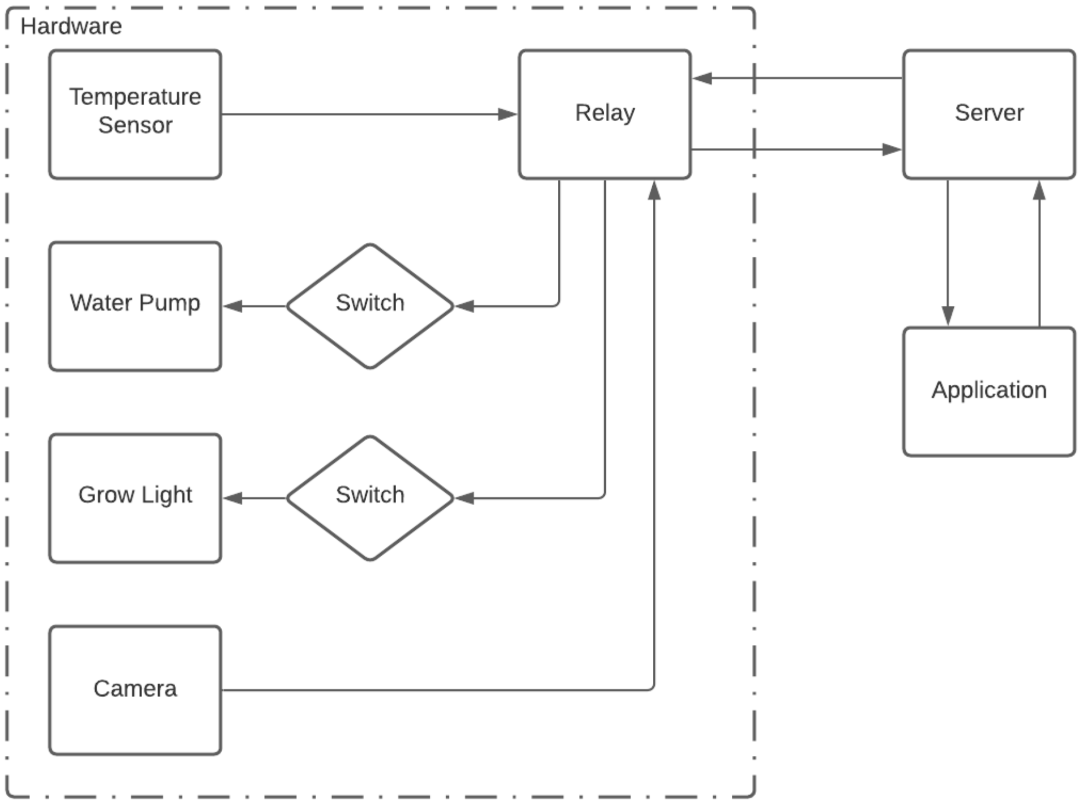

Figure 5:

This block diagram above describes a general overview of our smart garden design. All the hardware will be placed in our mini greenhouse testbed. A temperature sensor and camera will output data to the relay. The temperature will be placed into a table to be analyzed, while the camera output will be sent directly to the user via the software application. The water pump and grow light will be individually switched on and off. The water pump will also have a sensor to tell how much water is left in it's container. Based on the user input into the application on what plant is being grown, the water pump and grow light will automatically switch on and off at predetermined intervals of time. If the user wishes, they may manually enter in their own pump and light intervals. The application will display a grade to the given plant's health based on the collected data as well as give the option for the user to see the raw data graphed.

4.3.2 Functionality

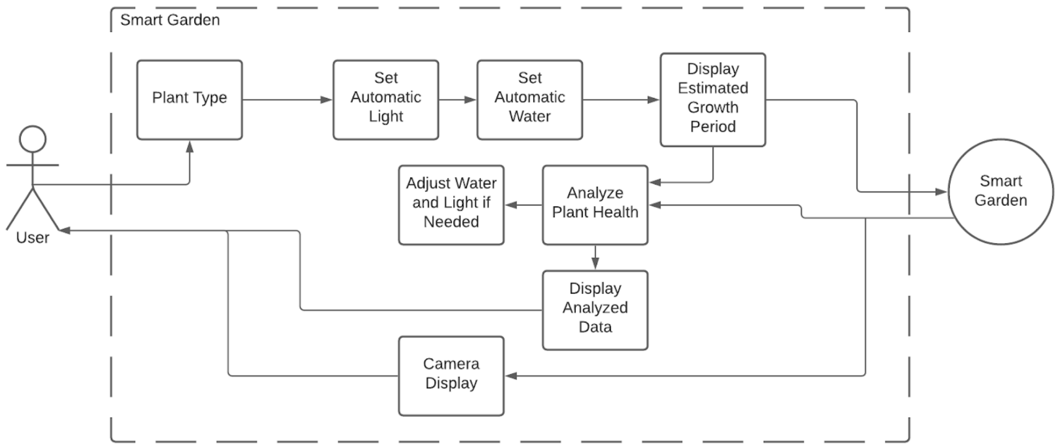

Figure 6:

Above is the use case diagram for the Smart Garden. This design satisfies all of our determined base requirements for this project.

4.3.3 Areas of Concern and Development

Our primary concern with the project is creating a system that is easy to run/maintain. As most clients will likely be buying this because they would rather have an automated system than do everything manually, it is important that we do everything we can to take work away from the user. That being said however, we will likely not be able to fully automate every aspect of plant growth due to the scope of our project and the time and resources available to us. As long as we are able to cover the basic requirements of plant care, we will consider the client's needs to be met.

4.4 Technology Considerations

For the budget and experience available, much of this project will be, for the most part, cheaper versions of technologies needed and barebones code to get the job done. Strengths of this include simplicity of design and implementation. Weaknesses include cheaper supplies that may need to be replaced at some point and possible oversights in efficiency.

4.5 Design Analysis

Upon further analysis we reworked our design through small corrections. These included adjustments for the necessary energy requirements of the hardware. The initial design served more as an estimate to our final prototype. Now our design is more thought out as we know what our hardware requirements are for our various monitors. We tried our best to limit the cost of the parts utilized in our design. We also looked into various other monitors to add to our design as we gained a better understanding of a plant's necessities. Now our design is more flushed out.

4.6 Design Plan

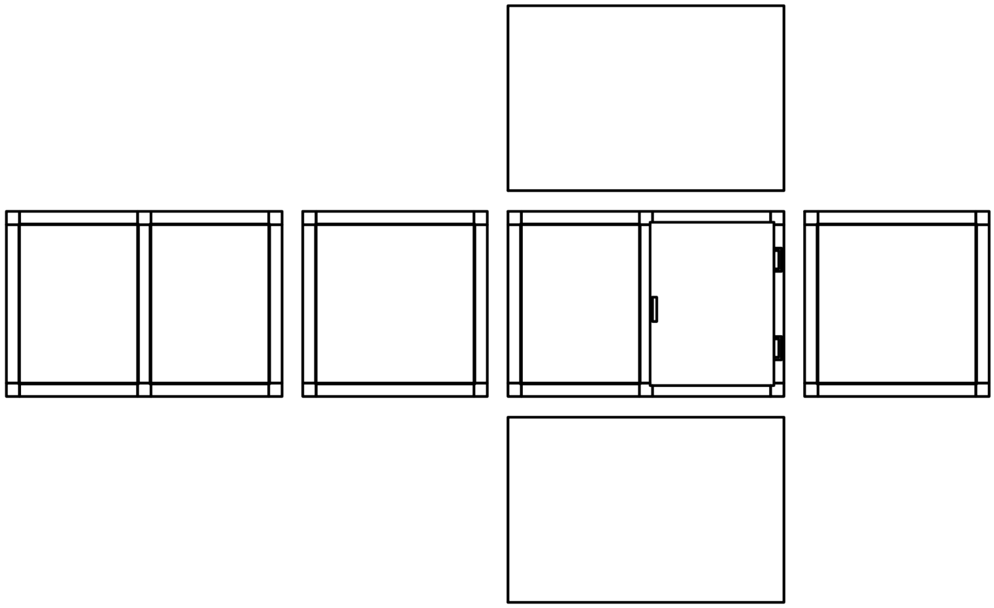

Figure 7:

The plant enclosure will be a wooden frame with clear vinyl windows. The top and bottom of the enclosure will be plywood. The inside of the enclosure will be accessible through a hinging door. Wiring will be passed through holes that have been drilled in the clear vinyl windows.

Inside the enclosure the UV grow lights, plant mister and camera will be mounted on the ceiling. The sensors will be placed in appropriate locations throughout the space. The rest of the space will be used to grow multiple plants at a time.

5 Testing

5.1 Unit Testing

For both our backend & frontend system we will be using Jest for unit testing. Jest is a javascript testing framework, and since we're using both javascript & node.js it will be perfect for our team. Jest will allow us to write modular tests on all functional parts of both frontend and backend.

5.2 Interface Testing

We will be creating a website application to display and analyze our sensor data. We will be testing this user interface with the Postman tool. This allows us to properly test our website even if a sensor is not connected.

5.3 Integration Testing

As our project develops and we integrate more parts into the project, we will test each component. As the project grows, we will need to perform system integration testing, testing the system as a whole to make sure all parts are working together.

5.4 System Testing

Complete end to end testing will have to wait until all the integration and interface testing has finished, since our sensors and camera will have to update the web server we will be using. I don't exactly know what tools can be used to test this, it must be more of a "trial and error" testing process, since any error could be any number of causes.

5.5 Regression Testing

We are going to be using an agile approach to testing our project, and as such as we go along and add new features we will make sure that they do not break what was already there. On the chance that they do break something, that gives us a chance to learn, grow and change what we have. We can make sure new functionality does not break, and meets our requirements, by doing Regression testing when the new feature is added. This will be going through our website and making sure the functionality that we expect to happen works. That would entail coordination with the hardware implemented like the cameras, and the watering system.

We need to make sure that anything that coordinates with the hardware does not break. As stated earlier this will be the camera, watering system, lighting timer. We also need to make sure that the website does not crash and can be accessible at all times by the user.

5.6 Acceptance Testing

We will make sure to have tests for all aspects of our design to make sure that all requirements are being met. This means we will do various kinds of tests to ensure that all of the functional requirements we have are working properly. So we will want to test that all our sensors and interfaces are working accurately. We will want to go back and see if everything we agreed to be in our project is actually there. Typically this is done to make sure the contract requirements are met, but in our case we don't have a client so we will go back to what we finalized as our main project and goal and make sure that what we decided on is up and running.

5.7 Results

We will measure our results as parts of a modular system. So long as the basic communication between each subsystem meets the requirements that we decide upon, we will consider that subsystem to function properly.

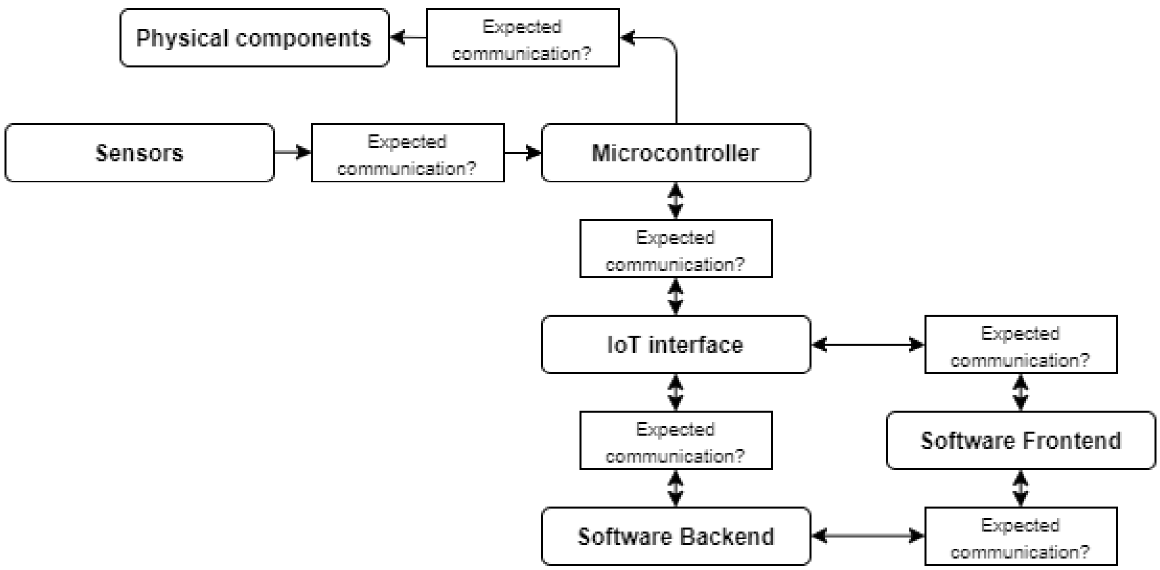

Figure 8:

Of course we want to make sure that the system as a whole functions as expected as well, so broader scale testing will also be measured against the benchmarks that we determine.

6 Implementation

1. Start with ordering and assembling cabinet for holding plants and equipment
2. Install lighting and watering system and set up for microcontroller implementation
3. Install camera and connect to microcontroller
4. Design a website for communication with microcontroller
5. Test equipment
6. Place plants in greenhouse and record results
7. Adjust settings based on plant growth

7 Professionalism

7.1 Areas of Responsibility

Area of Responsibility	Definition	NSPE Canon	SE code of ethics
Work Competence	Perform work of high quality, integrity, timeliness, and professional competence.	Perform services only in areas of their competence; Avoid deceptive acts.	They need to take responsibility for their work, what goes well and what does not go well. If software is approved it needs to be good quality, meet standards, exc. The difference we see between NSPE and SE is that there is an emphasis on the Approved work needing to meet certain standards of: Privacy. Specifications, code standards./ Privacy is huge in writing good code and as such there is an emphasis put that if the code is out there for the consumer there needs to be a certain standard of privacy met.
Financial Responsibility	Deliver products and services of realizable value and at reasonable costs.	Act for each employer or client as faithful agents or trustees.	One principle mentioned that is something to note is that any bad software or potential danger needs to be reported - those dangers can affect the cost the company has to deal with. There is no definitive 'financial responsibility' column in the SE code of ethics however, there are elements for it sprinkled throughout the code of ethics - one of these examples in 3.01.
Communication Honestly	Report work truthfully, without deception, and understandable to stakeholders.	Issue public statements only in an objective and truthful manner; Avoid deceptive acts.	In the SE code of ethics there are mentions of communication. Communication touches almost everything we do. As such there is an entire section on professionalism, as well as an emphasis on being truthful in your work; this can be seen in 6.07 and 6.08 among others.
Health, Safety, Well-being	Minimize risks to safety, health, and well-being of stakeholders.	Hold paramount the safety, health, and welfare of the public.	The SE code of ethics and NSPE are very similar in terms of health. The main difference is that NSPE focuses on the public's effects while the SE code focuses on all stakeholders. Stakeholders refers to anyone who might be affected by a project. The public is not necessarily always included in that scope.
Property Ownership	Respect property, ideas, and information of clients and others.	Act for each employer or client as faithful agents or trustees.	In the SE code of ethics, we see a strong concern for the property of others and the security of that property. When compared to the NSPE, they focus more on being trustworthy in general terms. This is more of an umbrella view of property security.
Sustainability	Protect the environment and natural resources locally and globally.	Encouraged to adhere to the principles of sustainable development in order to protect the environment for future generations	When looking through the SE code of ethics, there was no mention of sustainability. In NSPE it isn't even a definite rule, it's just an encouragement. So from my understanding, sustainable acts are to be done if possible, but not a priority.
Social Responsibility	Produce products and services that benefit society and communities.	Conduct themselves honorably, responsibly, ethically, and lawfully so as to enhance the honor, reputation, and usefulness of the profession.	In the SE code of ethics, it is the responsibility of software engineers to identify, document, and report significant issues of social concern, of which they are aware, in software or related documents, to the employer or the client. They have the same idea but the SE code of ethics is a little more specific to the client they are working for. It's so much for the good of the profession but for the good of the people they work for.

Figure 9

7.2 Project Specific Professional Responsibility Areas

Work Competence: This responsibility absolutely applies to our project. So far our performance in this area has been high.

Financial responsibility: We have briefly touched on this, in the coming weeks we will cover this more as we order parts and refine our product.

Communication honesty: This responsibility we have applied to our project throughout its existence. Everything we do is communicated clearly to everyone involved. Therefore we have performed high in this area.

Health, safety, and well-being: Our project luckily has not involved any hazardous tasks so far. If it were to we would take proper precautions to ensure healthy and safety to all parties involved.

Property ownership: We respect everyone's property and ideas. We have performed high in this category

Sustainability: Our product can actually be used to help with sustainability. Users are able to grow their own plants and lessen their impact on the environment.

Social responsibility: Going off of what was said for sustainability, our product can be used to lessen their impact on the environment, which benefits everyone.

7.3 Most Applicable Professional Responsibility Area

Financial Responsibility is one of our team's highest priorities when it comes to our project. Our goal is to make an affordable greenhouse, usable for the common person. While it's important to not sacrifice quality, the cheaper our greenhouse is, the more widespread it's usage can be. It's important to us that our consumers feel our product is within their budget. We've demonstrated this by creating a design that uses low cost parts and materials, and that sufficiently accomplishes the goals we set for the project. In doing so, we've had to sacrifice some of the more complex functionality while maintaining the most important systems. This is a tradeoff that allows us to deliver an affordable product that also covers the necessities of plant care.

8 Closing Material

8.1 Discussion

Since we do not have a client, the requirements were created by us, so far that is to design a garden system that will have a camera, a way to water the system, lights, and a thermometer. This so far is all created and designed into our syte. We wanted to focus on the design this semester and as such we have not started the current implementation but we have everything in place to do so in the coming semester. As for designing the system, it has worked. We had to tweak some thoughts as we went along but overall the system is a go!

8.2 Conclusion

So far we have gathered a list with prices of the material we need, we are currently in the process of ordering those materials so that when next semester comes we can get a head start on going in and hit the ground running with the actual execution of what we need to do. Our goal is to create an efficient, inexpensive garden system. To achieve this we have carefully calculated our costs and plan on creating the first design of this system starting next semester. So far nothing we can see could be done differently, this semester we found it valuable to only focus on the planning phase, that way we can set up everything for next semester to be ready.

8.3 References

Related work we have looked at are garden systems. Some of those include:

Triple Family Garden

Single Family Garden

Aero Garden

8.4 Appendices

8.4.1 Team Contract

Team Members:

1) Sarah Schoenke 2) Devon Sindt

3) Julia Condon 4) Jake Thomae

5) Nick Vaughan 6) Jasen Helsel

7) Bryanna Adamson

Team Procedures

1. Day, time, and location (face-to-face or virtual) for regular team meetings: a. Meetings will place over discords voice chats once a week, Wednesdays at 8 pm 2. Preferred method of communication updates, reminders, issues, and scheduling (e.g., e-mail, phone, app, face-to-face):
    a. Discord and weekly meetings will be the main route of communication for our group
2. Decision-making policy (e.g., consensus, majority vote):
    a. All big decisions for the project will be settled by a majority vote.
3. Procedures for record-keeping (i.e., who will keep meeting minutes, how will minutes be shared/archived):
    a. Meeting minutes will be kept in our discord, in the chat pertaining to the type of meeting

Participation Expectations

1. Expected individual attendance, punctuality, and participation at all team meetings:
    a. All members will be held accountable for being on time for all meetings unless they have made it clear that they are not available or they have notified the group that they cannot attend the meeting for a good reason.
2. Expected level of responsibility for fulfilling team assignments, timelines, and deadlines:
    a. All members are expected to fulfill their assignments to the best of their ability by whatever deadline has been set. If the deadline cannot be met, then the member should notify the rest of the project team and advisor and attempt to find a new, reasonable deadline.
3. Expected level of communication with other team members:
    a. All members are expected to communicate clearly to each other on a regular basis. This does not mean responding to every message immediately when sent, but responding in a timely manner so as to not hinder group decisions or anything important.
4. Expected level of commitment to team decisions and tasks:
    a. All members are expected to be 100% committed to the project's success. If the group makes a decision based on a majority vote that an individual does not agree with, it is the individual's responsibility to communicate their grievances with the project team so that they may come to a resolution.

Leadership

1. Leadership roles for each team member (e.g., team organization, client interaction, individual component design, testing, etc.):

Sarah Schoenke: Client Interaction
Jasen Helsel: Testing/IOT
Nick Vaughan: Scrum Master (AGILE)/Backend
Jake Thomae: Assistant Embedded System/Team Mascot
Bryanna Adamson: Team Organization

Devon Sindt: Embedded Systems
Julia Condon: Application Design

2. Strategies for supporting and guiding the work of all team members:
   a. Be open to answering questions, this includes having to teach people new things if needed.
3. Strategies for recognizing the contributions of all team members:
   a. We can weekly go over what we all worked on each week or so at team meetings. b. If someone has a really good idea, or does something really good we can mention it as a group

Collaboration and Inclusion

1. Describe the skills, expertise, and unique perspectives each team member brings to the team.
   a. Sarah Schoenke: some agile, a lot of testing development and experience, Front end Experience
   b. Jasen Helsel: embedded systems programming, experience with basic electronics sensors, comfortable with Java and C
   c. Nick Vaughan: Agile practices, backend software development, & some embedded programming skills
   d. Jake Thomae: some embedded systems programming (mostly arduino), fluent in C + Java, very minimal app development(android studios)
   e. Bryanna Adamson: Frontend software development (mostly in Java), app development (android studios), Java, and a little C
   f. Devon Sindt: Strong communication skills, Extensive experience developing embedded systems (at least for a college student)
   g. Julia Condon: Agile environment, full-stack development, and a pinch of embedded systems
2. Strategies for encouraging and support contributions and ideas from all team members: a. When in team meetings, iteratively go through team members (order above) asking for ideas/thoughts about the week's tasks.
   b. Group brainstorming sessions during team meetings for upcoming design/implementation tasks

3. Procedures for identifying and resolving collaboration or inclusion issues (e.g., how will a team member inform the team that the team environment is obstructing their opportunity or ability to contribute?)
   a. During weekly team meetings, members will be asked the status of their current tasks
      i. It's up to to the team member to discuss collaboration/inclusion issues during this time
   b. Team members can post at any time on discord/slack about any team environment problems they're facing.

Goal-Setting, Planning, and Execution

1. Team goals for this semester:
   By the end of this semester, we would like to:
   a. Gain experience working and communicating with a team in a professional manner.
   b. Apply skills we have learned in previous classes to our project.

2. Strategies for planning and assigning individual and teamwork:
    a. Team members will be able to elect which parts of the project they would prefer to work on, and work will be divided up accordingly. If any parts of the project are left unassigned, they will be assigned based on whichever member has the lightest workload for that week.

3. Strategies for keeping on task:
    a. Team members will briefly discuss what they have worked on over the week during weekly meetings.

Consequences for Not Adhering to Team Contract

1. How will you handle infractions of any of the obligations of this team contract? a. Have a discussion with the offending party about the expectations we have for each other and what the contributing factors to their infractions were. Try to mediate any conflict by either giving the individual less responsibility or swapping their responsibilities with a different team member.

2. What will your team do if the infractions continue?
    a. Have a discussion with our project advisor and TA about the infractions and decide our best course of action with the individual responsible.
    b. Murder by Mochi (the cat) :)

****

a) *I participated in formulating the standards, roles, and procedures as stated in this contract.*
b) *I understand that I am obligated to abide by these terms and conditions.*
c) *I understand that if I do not abide by these terms and conditions, I will suffer the consequences as stated in this contract.*

1) Sarah Schoenke DATE 9/15/21
2) Jasen Helsel DATE 9/15/21
3) Bryanna Adamson DATE 9/15/21
4) Jake Thomae DATE 9/15/21
5) Nick Vaughan DATE 9/15/21
6) Devon Sindt DATE 9/15/21
7) Julia Condon DATE 9/15/21