Faculté de génie
Faculty of Engineering

Automated Greenhouse Maintenance System

Final Report

CEG 4912: Computer Engineering Design Project

Fall 2022 - Winter 2023

Group #5

Name	Student Number
Ashair Imran	300071365
Alex [redacted]	[redacted]
Edward [redacted]	
Ibrahim [redacted]	
Ulysses [redacted]	
Jai [redacted]	

Table of Contents

Section 1. Project Charter

Section 1.1 Charter Introduction

1.1.1 Document Change Control

Revision Number	Date of Issue	Author(s)	Brief Description of Change
1.0	2022-09-30	Alex Mushyirahamwe Edward Zhu Ibrahim Akiode Ashair Imran Ulysses Vaughan Jai Chauhan	Creation of the document.
2.0	2022-12-04	Alex Mushyirahamwe Edward Zhu Ibrahim Akiode Ashair Imran Ulysses Vaughan Jai Chauhan	Updated document for final report submission

1.1.2 Executive Summary

This project aims to design a fully automated and computerized greenhouse system to grow plants in an efficient and sustainable way. This project was created to simplify and automate the tasks involved with gardening, while also maintaining an ideal environment for plant growth. This system will allow anyone to grow plants without any involvement by automatically collecting data, watering, and providing light for the plants. The sensors will collect data about the soil, such as the humidity, temperature, and light exposure. This data will be used to optimize and automate the gardening process and notify the user if any problems are encountered. The final project will include hardware sensors, mechanisms, microcontrollers, and a web app that the user will be able to use to monitor their greenhouse.

This project was initiated to create a fully monitored and automated environment for plant growth that is far more efficient than a manual greenhouse. The downsides of manual greenhouses are that they require constant maintenance and that they are expensive to upkeep. The automation of menial tasks can decrease costs, decrease the impact on the climate, and make growing plants a more enjoyable experience. One important factor is that we will use scientific studies and statistics to expedite plant growth. The fully monitored greenhouse will provide various data that will be used to optimize watering and the provision of light. Overall, we plan to provide an automated system to help farmers and amateur gardeners alike.

Section 1.2 Project Overview

1.2.1Project Summary

This project charter is an in-depth overview of our Automated Greenhouse Maintenance System. This document contains the scope of the project, its goals, milestones, estimated costs, risks, assumptions, organization, and so on. The goal of the project, as stated earlier, is to create a fully automated greenhouse system to grow plants efficiently, and sustainably with minimal manual labour. The system will collect data from the garden and automate systems such as watering and lighting. Scientific studies and research will be used to perfectly optimize all the variables involved with plant growth.

This project was initiated to improve the gardening experience for gardeners and professional farmers alike. This system will be of interest to any person who wishes to improve their plant growing experience. This includes amateurs and professionals alike. The automated greenhouse maintenance system will collect important data, reduce labour, and increase efficiency and sustainability. All these factors will help users save money and time when gardening. The project will be developed over 8 months and consist of 4 main components, including the project design, software application, physical prototype, and final report.

1.2.2 Project Goals, Business Outcomes and Objectives

No.	Goals	Objectives	Business Outcomes
1	Automated monitoring of greenhouse environment Automated monitoring of - Air humidity - Air temperature - Soil humidity - Ambient Light Level	Accuracy > 95%	Duplicable prototype of hardware sensors as well as fully complete firmware that is able to interface with the hardware prototype and web application to provide charts of monitored values
2	Automated regulation of greenhouse environment Automated regulation of - Soil humidity - Light Level	Accuracy > 95%	Duplicable prototype of hardware necessary to regulate environment as well as fully complete firmware that is able to interface with hardware prototype and web application to allow for user

			control of regulated parameters
3	Storage of data gathered from monitoring	Storage of data gathered during monitoring of greenhouse conditions	Cloud storage of collected data as well as cached local values for use by web application
4	User Interfacing to access information and change settings	Creation of a user accessible web app.	Application suitable to be published to Google and Apple app stores for commercial use

Table 1: Project Goals, Business Outcomes and Objectives

1.2.3 Project Scope

1.2.3.1 Scope Definition

The project will deliver a product that can automate the process of growing plants in a greenhouse. The features and functions that characterize the product are:

- Continuous Monitoring

Parameters that influence plant growth such as temperature, humidity, and soil moisture, must be continuously monitored to provide feedback to the user as well as to the controller.

- Automated Control of Physical Tasks

The automated system will use the data acquired from continuous monitoring to adjust watering and humidity of the plants.

- Web Application

The web application will provide user with charts showing data collected from continuous monitoring and allow for manual control of watering and humidity.

1.2.3.2 Boundaries

Activities In Scope	Activities Out of Scope
1. Develop requirements	Interaction with physical security systems.
2. Develop environment monitoring module	Monitor plant conditions and keep inventory of plants.
3. Develop automation module	
4. Develop data storage module	
5. Develop web interface	

6. Develop web application	
7. Prepare project charter & documents	
8. Prepare interim report	
9. Prepare midterm report	
10. Prepare midterm presentation	
11. Prepare final report	
12. Prepare final presentation	

Table 2: Boundaries

Activities In Scope

Since the project is focused on automated maintenance of a greenhouse's environment, this includes both monitoring the environmental conditions inside the greenhouse, both ambient and soil conditions, as well as correcting any deviations from the settings the users defined. This includes developing systems for monitoring, for changing the conditions, data storage, a web interface for the user to interact with as well as an associated web application.

Activities Out of Scope

Since the project is not concerned with the structure of the greenhouse, or any other factors associated with running a greenhouse, it will not have features related to the physical structure of the greenhouse. Therefore, it will not have integration with camera surveillance, physical security, or other features. Additionally, it will not be able to monitor plant conditions, and keep track of the number of plants in the greenhouse.

1.2.4 Milestones

Project Milestone	Description	Expected Date
1. Design System	Determine hardware and software specifications. Design overall system, user interface, physical requirements	2022-12-15

2. Develop App	Write software for user application. Develop and implement user interface	2023-02-22
3. Develop Prototype	Acquire necessary hardware for project. Write software for Raspberry Pi. Assemble physical prototype.	2023-02-22
4. Conduct Testing	Perform hardware and software tests. Fix defects and bugs that appear.	2023-03-31
5. Finalize report and presentation	Summarize project and present final system.	TBA

1.2.5 Deliverables

Project Deliverable 1: Environment Monitoring	
Description:	Measurement of environmental conditions including: - Ambient Temperature - Ambient Light - Ambient Humidity - Soil Humidity
Acceptance Criteria:	Can get measurements from the various monitoring sensors.
Due Date:	2023-01-07
Project Deliverable 2: Application Framework (Server)	
Description:	System can get information from the sensors and present it on a web page.
Acceptance Criteria:	Information from all sensors is logged and presented on a web page.
Due Date:	2023-01-11
Project Deliverable 2: Manual Environment Control	
Description:	Control of environmental conditions including: - Soil Moisture - Light Level
Acceptance Criteria:	Can manually control pumps to water plants and manually adjust light level
Due Date:	2023-01-21
Project Deliverable 3: Automated Environment Control	
Description:	Maintaining environmental conditions based on preset values.
Acceptance Criteria:	The system can maintain the environmental conditions without manual intervention.

Due Date:	2023-02-10
Project Deliverable 4: Web Application	
Description:	Web application is fully implemented in terms of features. Allowing a user to monitor conditions of their greenhouse and set conditions to be maintained.
Acceptance Criteria:	Web application is free of obvious bugs and the interface displays environmental conditions to the user. Web application allows the user to manually control environment conditions and set parameters for automated control.
Due Date:	2023-02-22

Section 1.3. Project Organization
1.3.1 Project Governance

A democratic system will be used for the governance of this project, and every member of the team must agree for us to reach a final decision. If a consensus is unable to be reached, we must change the decision in order to reach unanimity.

1.3.2 Project Team Structure

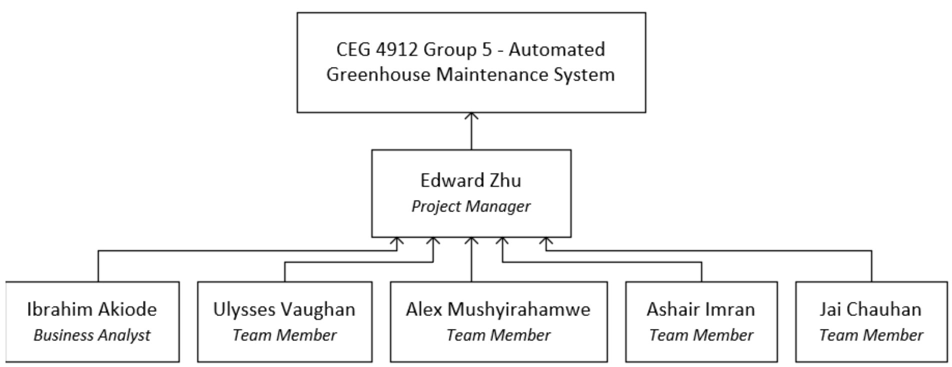

Figure 1: Team Structure

1.3.3 Roles and Responsibilities

Project Role - Project Manager
Responsibilities

- Coordinating responsibilities for group members
- Organizing meetings and lab time (if needed)
- Setting deadlines for tasks
- Assigning priorities to tasks

- Overseeing software and hardware design

Assigned to Edward Zhu

Project Role - Business Analyst
Responsibilities
- Analyzing business case for project
- Suggesting features based on market research
- Evaluating components based on cost benefit analysis

Assigned to Ibrahim Akiode

Project Role - Project Review Committee
Responsibilities
- Reviewing progress based on pre-determined milestones
- Approving additional features based on feasibility and need
- Producing and updating documents relating to project management and features
- Testing features to ensure system requirements are met

Assigned to all group members

1.3.4 Project Facilities and Resources

The project will not require any specific facilities, but hardware will have to be acquired including a Raspberry Pi, various sensors, relays, lights, and pumps. Soldering may have to be performed which will require the use of a soldering iron and ventilated workspace. Development of the web application will require a laptop/computer for testing. Development of the firmware on the Raspberry Pi will require a monitor, mouse, and keyboard. The lab space provided for the course may be used for collaboration.

Section 1.4. Project References

More information concerning this project can be found in the following documents:
Document Title
CEG 4912 Group 5 – Automated Greenhouse Maintenance Project Charter
Version #
1
Date
21-Oct-2022
Author and Organization
CEG 4912 Group 5
Location (link or path)
Brightspace Submission

Section 1.5. Glossary and acronyms

N/A

Section 2. SRS

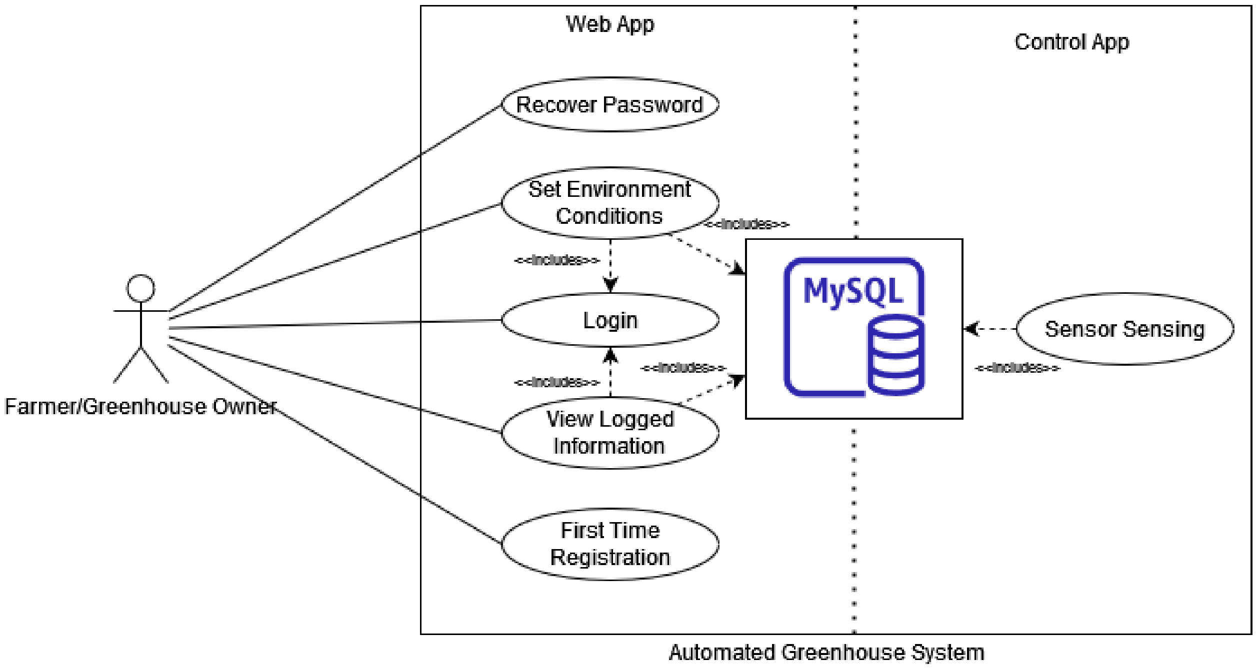

Figure 2: System Requirements Specifications

Actor:

Farmer/Greenhouse Owner

- Set Environment Conditions
- View Sensor Information
- First Time Registration
- User Login
- Password Recovery

Use Cases:

Set Environment Conditions

Summary: The farmer/greenhouse owner can set the environment conditions the system must maintain. The system will get information from various sensors and will modify conditions based on the settings by powering the relays controlling the lights and watering systems.

Actor: Farmer/Greenhouse Owner

Main Sequence:

Step 1: Farmer/Greenhouse Owner sets the light level and soil humidity level to be maintained in the greenhouse.

Step 2a: If the measured environmental conditions differ from the settings, adjust lights and watering system to reach the desired environmental conditions.

Step 2b: If the measured environmental conditions match the settings, the system will continue monitoring until deviation occurs.

View Sensor Information

Summary: The farmer/greenhouse owner can view the logged information collected by the various sensors.

Main Sequence:

Step 1: Farmer/Greenhouse Owner can view the logged information on the web application UI.

First Time Registration

Summary: The farmer/greenhouse owner registers for an account with a username, email and password.

Step 1: Farmer/Greenhouse Owner starts the system for the first time.

Step 2: Farmer/Greenhouse Owner inputs the desired username, password and email address.

Step 3a: If the email address input is invalid (not an @domain address) The Farmer/Greenhouse Owner receives a prompt and will need to enter a valid email address.

Step 3b: If the email address is valid. The Farmer/Greenhouse Owner then proceeds to the login page with their account registered.

User Login

Summary: The farmer/greenhouse owner logs in to the system with their information, with errors if the password is incorrect.

Step 1: Farmer/Greenhouse Owner inputs their username and password.

Step 2a: If the password is incorrect, then the Farmer/Greenhouse Owner is shown a prompt that the password is incorrect, and Farmer/Greenhouse Owner will need to reenter the password.

Step 2b: If the password is correct, then the Farmer/Greenhouse Owner proceeds to the main page.

Password Recovery

Summary: Farmer/Greenhouse Owner has forgotten their password and will need to provide a valid email address to reset their password.

Step 1: Farmer/Greenhouse Owner presses the forgot password button, and proceeds to the password recovery page.

Step 2: Farmer/Greenhouse Owner enters their email into the field and presses the submit button.

Step 3a: If the email is correct and account exists, then the Farmer/Greenhouse receives an email with a link to recover the password.

Step 3b: If the email is incorrect, then the Farmer/Greenhouse receives a prompt that the user does not exist and will need to reenter the password.

Step 4: The Farmer/Greenhouse clicks on the link and proceeds to the change password page.

Step 5: The Farmer/Greenhouse inputs the new password and presses the confirm button.

Step 6: The Farmer/Greenhouse account password is changed and the user proceeds to the login page.

Functional Requirements:

- The user shall be able to monitor the ambient temperature of the greenhouse.
- The user shall be able to monitor the humidity of the greenhouse.
- The user shall be able to monitor the humidity of the soil in the greenhouse.
- The user shall be able to monitor the ambient light level in the greenhouse.
- The user shall be able to set the desired ambient humidity in the greenhouse.
- The user shall be able to set the desired ambient light level in the greenhouse.
- The user shall be able to set the desired soil humidity in the greenhouse.
- The user shall be able to automatically maintain the desired humidity and light level in the greenhouse.
- The user shall be able to interact with the system through a web application.
- The user shall be able to interact with the system through a mobile phone using the web application.
- The user shall interact with the system after a login process.
- The user shall be able to recover his password through the system.

Non-functional Requirements

- The system should have a monitoring accuracy greater than 95% compared to real values.
- The monitoring frequency of the sensors will be no less than 5s between intervals.
- The system should present the monitored data in an intuitive graphical manner.
- The system should allow the user to view data from various time periods.
- The system should have an intuitive graphical interface.
- The system should start automatically when recovering from a power loss.

- The system should lose minimal data upon a power loss.

Constraints

- The system must use voltages lower than 50V DC.
- The system must be waterproof and will not be an electrocution hazard.
- The system must maintain an uptime of 99% over a 24hr period.
- The system must be operable between –10C to 40C.

Section 3. Detailed Design

3.1 Hardware Design

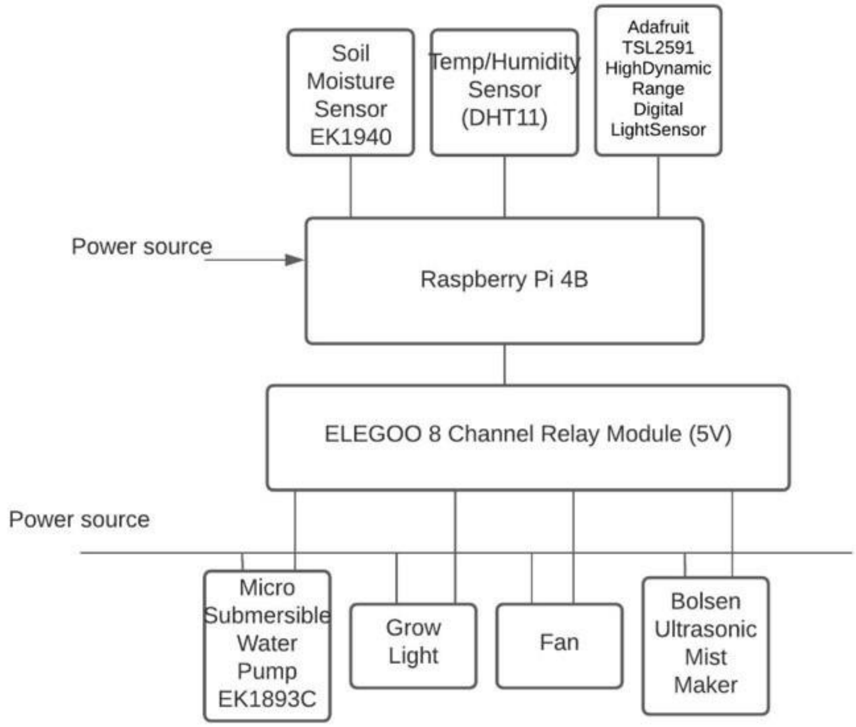

Figure 3: Hardware Block Diagram

The above figure shows the basic hardware block diagram for the hardware design. There are three sensors connected to the Raspberry Pi. The Raspberry Pi is connected to the 8-channel relay, which then connects to the various hardware devices. This hardware block diagram will be expanded upon below.

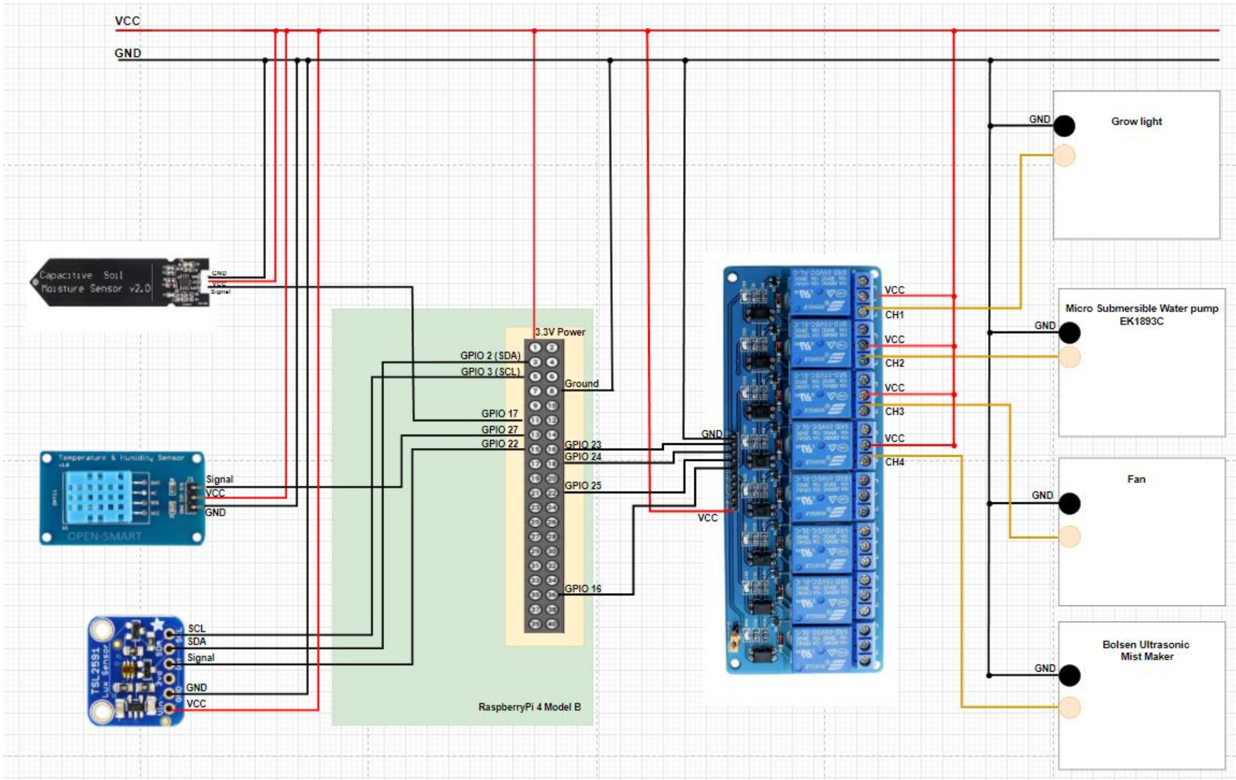

Figure 4: Hardware Wiring Diagram

The above figure shows the hardware wiring diagram, a physical translation of the hardware block diagram. This diagram shows the component models and the exact wiring inputs as well as the pin layout of our hardware design. The exact models are subject to change, but the models shown are the ones we currently plan to acquire for the project.

There are four main sections to the diagram:

- Sensors
  - The sensors compose the lowest layer of the entire design. These are basic sensors that will collect raw data from the greenhouse environment.
  - Moisture, Humidity, Temperature, Light
- Raspberry Pi
  - The Pi will take this raw data and perform all the necessary computation, such as storing the data in the database, and using the data to calculate when to activate the necessary hardware devices.
  - pin 6 is connected to the ground line on the breadboard and pin 1 is connected to the voltage bus on the breadboard.
  - GPIO 23, 24, 25 and 16 are connected to the first four channel inputs of the 8-channel relay.
- Relay
  - The relay channel will connect the water pump grow light, fan, and mist maker components to the Raspberry Pi, and act as an electrically, automated switch.
  - The first pin is connected to the ground and the last pin is connected to the VCC power supply on the breadboard.

  - The first four channel inputs are connected to the Raspberry Pi as aforementioned.
- Hardware Components
  - The four hardware components that are to be connected to the 8-channel ELEGOO relay as follows:
  - channel 1 relay terminal is connected to the grow light input
  - channel 2 relay terminal is connected to the micro submersible water pump input
  - channel 3 relay terminal is connected to the fan input
  - channel 4 relay terminal is connected to the Bolsen ultrasonic mist maker input

3.2 Control Software:

Control Language:

- Python

Database:

- MySQL

Library:

- MySQL Connector, which is like a MySQL driver which Python needs to access the database

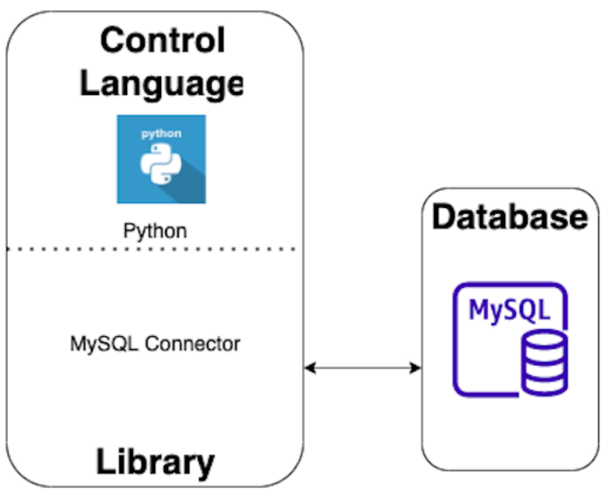

Figure 5: Control Software

All the code for the control software will be written in Python. We chose Python as it is one of the most accessible programming languages available because of its simplified syntax. Moreover, Python codes can be easily written and executed much faster than most of the programming languages.

Our code will include methods that do the following:

1. Getting the readings from the sensors.
2. Parsing and converting that data into simple and understandable form.
3. Storing the data into the database using MySQL Connector.

The code will also contain comparison functions which would compare the sensor readings with the data that the user has put in settings.

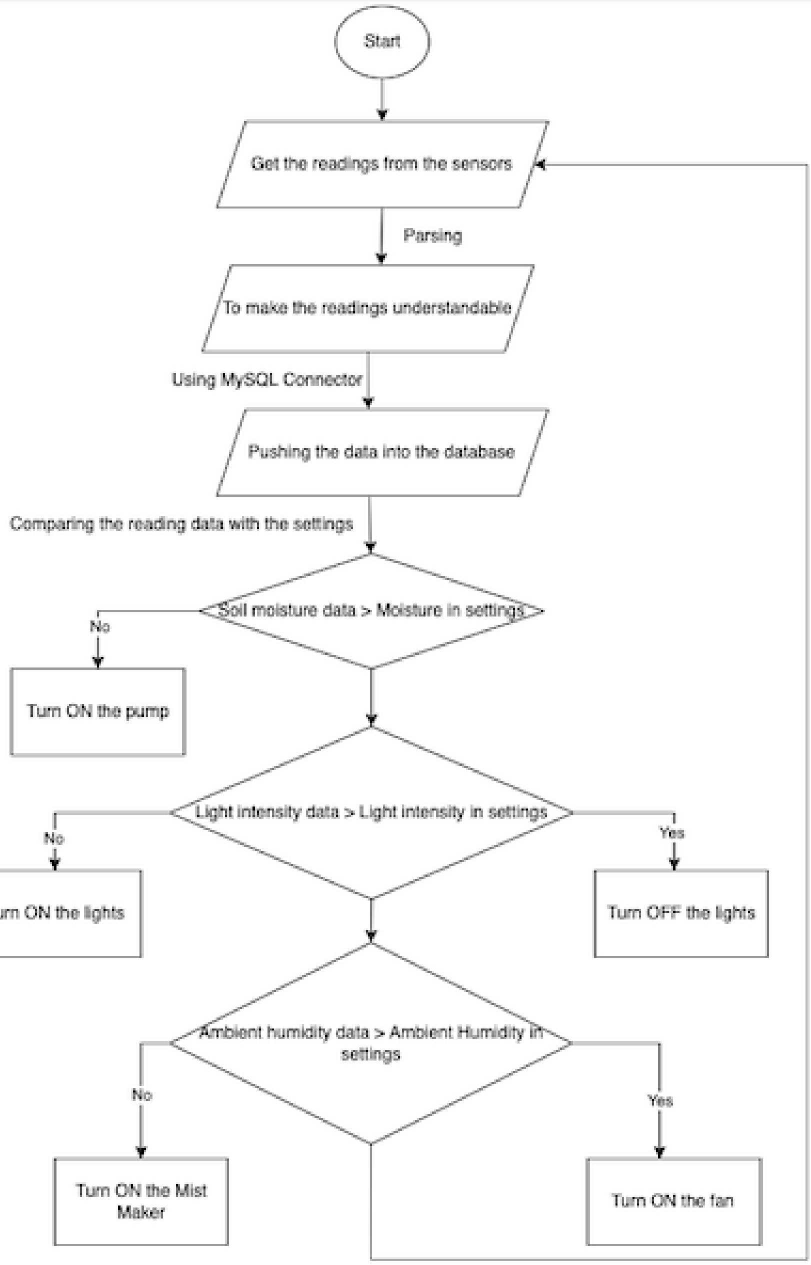

Figure 6: Control Software Flowchart

Our code is going to work with the hardware, which will turn on or turn off the different functional components like the water pump, grow light, fan and the mist maker.

3.3 Web App:

Backend:

- Django

  - Framework for building the website. Contains library that implements basic website functions.
- Apache
  - Web server used for hosting a web page.
- MySQL Database
  - Database used for reading data to display graphs, storing settings for the system and storing user account.

Frontend:

- React
  - JavaScript UI library that is used for developing the UI of the website.

Interconnect:

- Rest API
  - API that is used for better functionality between Django and React.

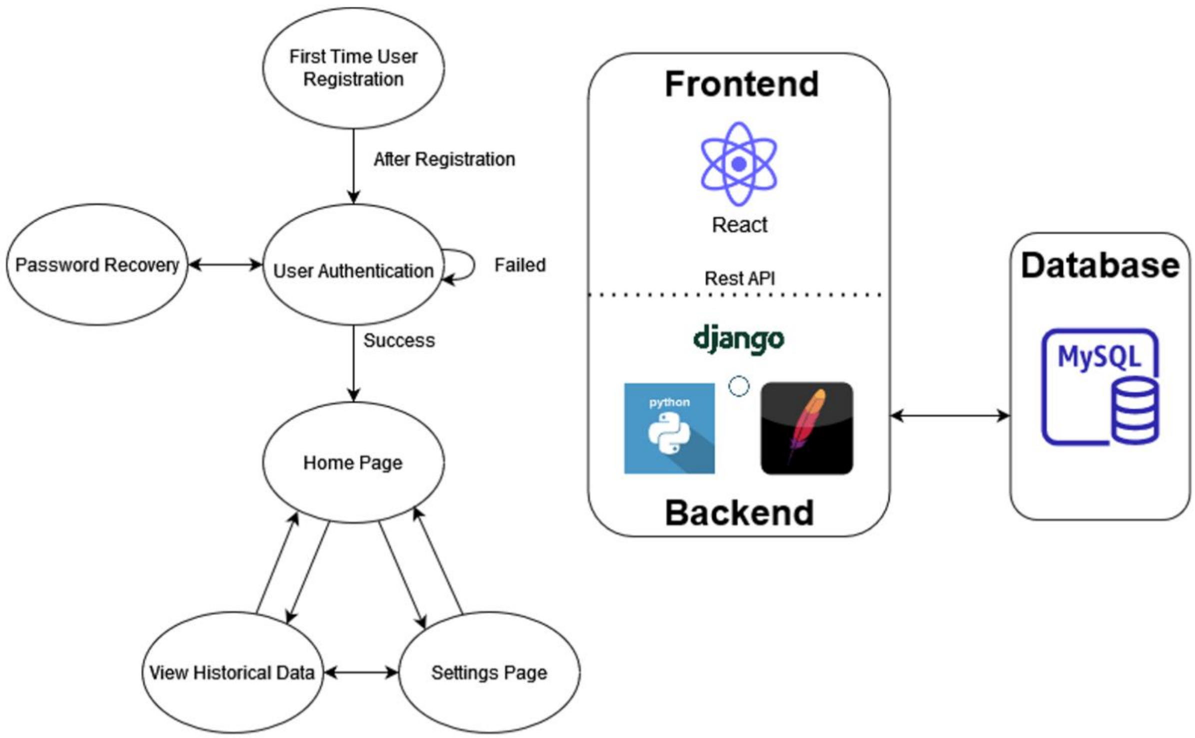

Figure 7: Software Flow Diagram

First Time User Registration

- Allows the user to register an account, only accessed during the first time start up.

User Authentication

- Login page for the user. The user will need to login with username and password and can proceed to reset their password from this page.

Password Recovery

- Password recovery page for the user. The user will need to provide a valid email address and then they will receive an email that allows them to change their password.

Home Page

- First page the user reaches after authentication. The page displays dynamic graphs for the various conditions. The user will be able to navigate between this page, View Historical Data and the Settings Page.

View Historical Data

- Here, the user can view the graphs of data over varying time periods. The user will be able to navigate between this page, View Historical Data and the Settings Page.

Settings Page

- Here, the user can view the current environment settings, as well as changing the settings to alter the desired environmental conditions. The user will be able to navigate between this page, View Historical Data and the Settings Page.

Section 4. Schedule and Budget Outlook

4.1 Milestones

Project Milestone	Description	Expected Date
1. Design System	Determine hardware and software specifications. Design overall system, user interface, physical requirements	2022-12-15
2. Develop App	Write software for user application. Develop and implement user interface	2023-02-22
3. Develop Prototype	Acquire necessary hardware for project. Write software for Raspberry Pi. Assemble physical prototype.	2023-03-02
4. Conduct Testing	Perform hardware and software tests. Fix defects and bugs that appear.	2023-03-31
5. Finalize report and presentation	Summarize project and present final system.	TBA

Table: Milestones

4.2 WBS

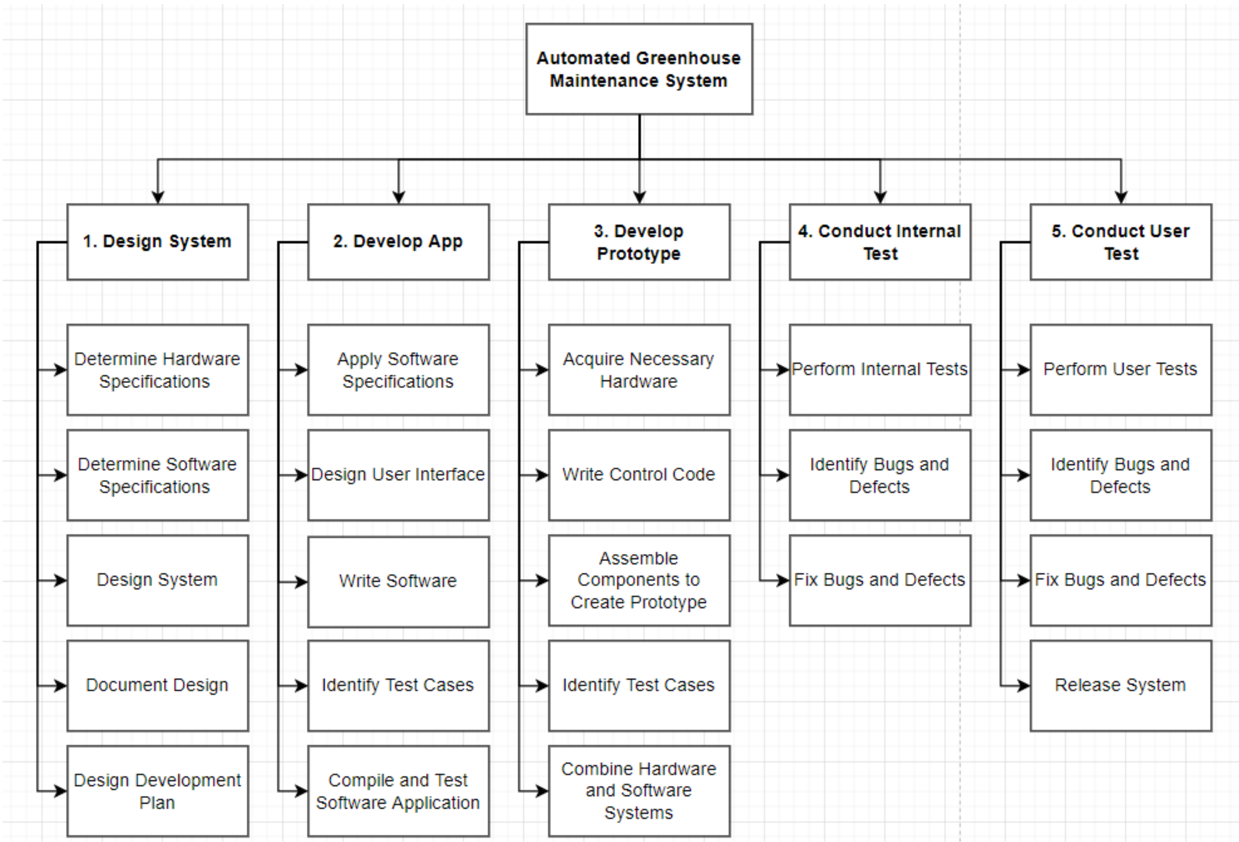

Figure 8: WBS

4.3 Gantt Chart

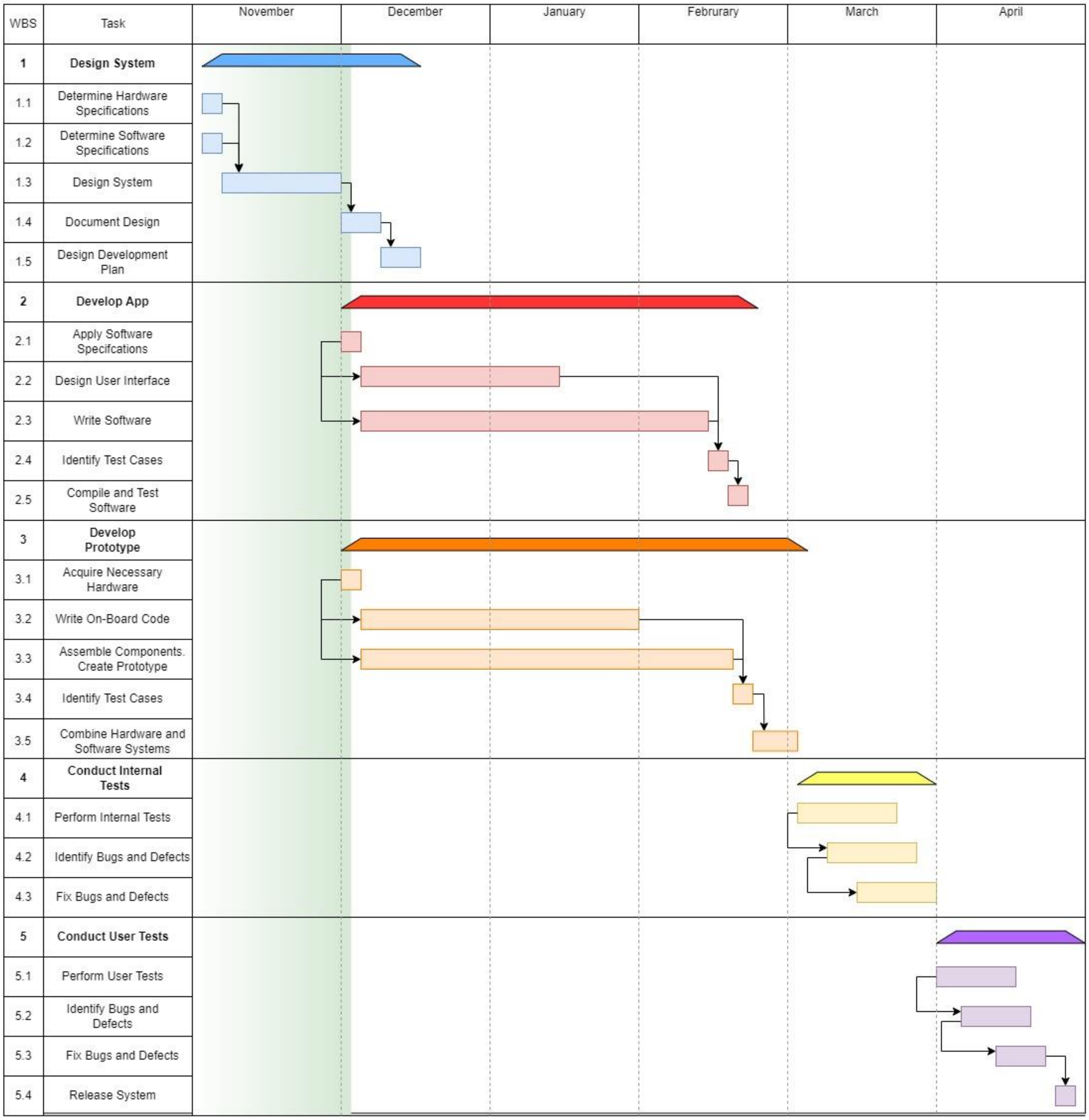

Figure 9: Gantt Chart

4.4 Trello Board

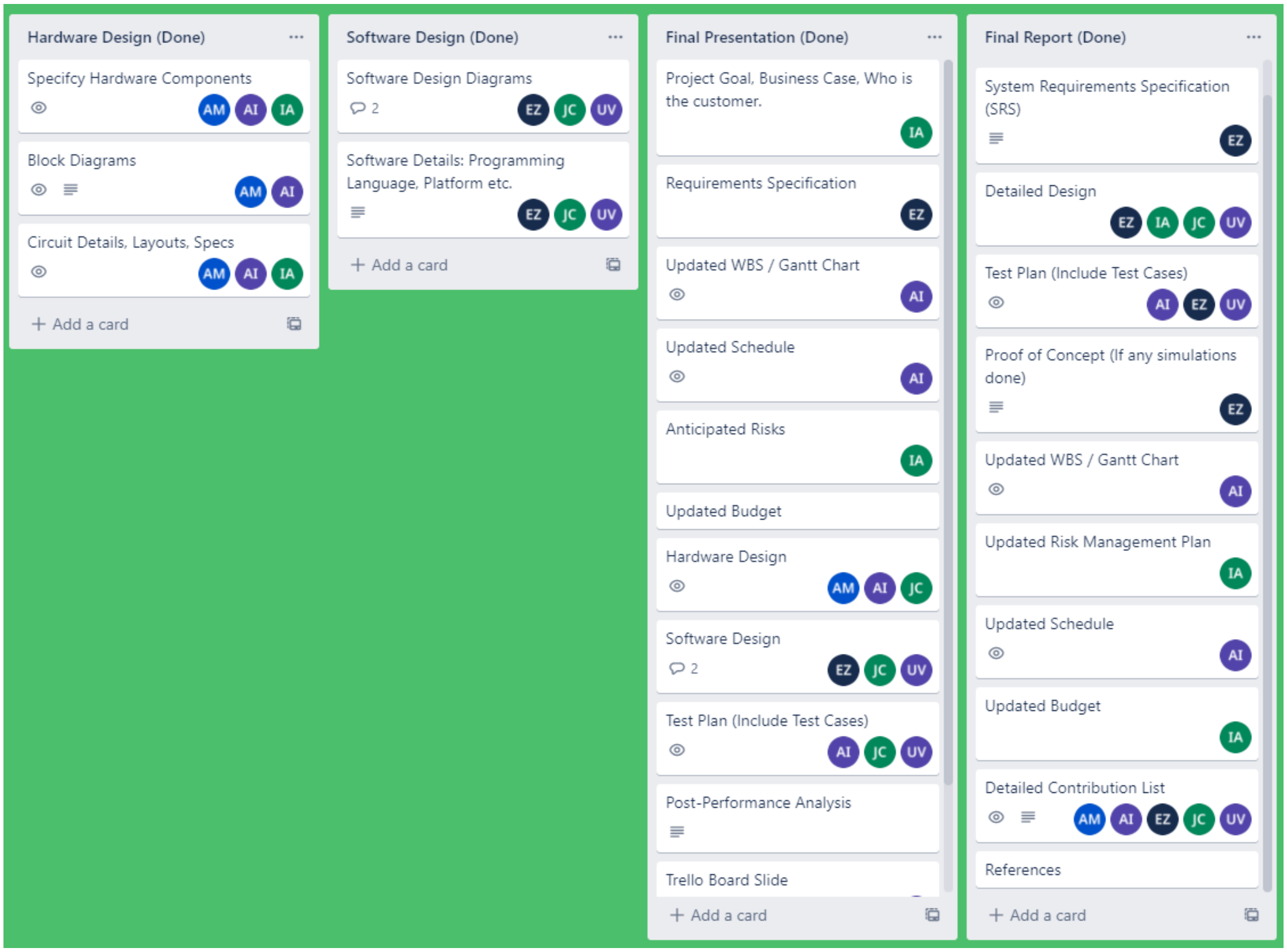

Figure 10: Trello Board

4.5 Project Cost Estimate

Product Phase or Deliverable Cost Category	**Quantity (Units)**	**Price Per Unit (CAD)**	**Total (CAD)**	**Already Obtained (Y/N)**
Miniature Greenhouse	1	-	-	N
Raspberry Pi 4B kit	1	69.95	69.95	Y
SanDisk Pre-Installed NOOBS MicroSD Cards	1	12.95	12.95	N
Adafruit TSL2591 High Dynamic Range Digital Light Sensor	1	11.94	11.94	N
SparkFun Soil Moisture Sensor (with Screw Terminals)	1	9.95	9.95	Y
Arduino Compatible Digital Temperature Humidity Sensor Module (DHT11)	1	5.95	5.95	Y
GDSTIME 14025 5.51in 12V Muffin Fan 140 x 25mm 140mm 14cm 12V DC Brushless Cooling Fan with XH2.54 Plug 1500RPM	1	12.99	12.99	N
8-Channel Relay Module for Arduino & Raspberry Pi - 5V	1	8.95	8.95	Y
LED Strip	1	18.99	18.99	N
Sub-Totals	151.67			
Total	171.39			

Table: Cost Estimate

4.6 Source of Funding

This project will be completely funded privately by the team members. Participants of the project will use pre-owned hardware and share the cost of any new items required.

Section 5. Risk Management Plans

5.1 Risks

No.	Risk Description	Probability (H/M/L)	Impact (H/M/L)	Planned Mitigation
1.	Hardware design does not finish on time	M	H	Meticulous planning and scheduling of tasks and hardware development
2.	Sensors have lower than expected accuracy.	M	H	New sensors with a higher degree of accuracy will be purchased, existing sensors will be used for prototyping.
3.	The system may not have access to a stable internet connection in its installed location.	M	H	The system will be able to run without an active connection to the internet, with potentially a wifi-connection to access the web app without internet.
4.	The greenhouse will not have backup power. If the power goes out or there is an electrical problem, it cannot function	L	H	A backup power system can be introduced. A human can check in occasionally.
5.	Greenhouses have a chance of insect or rodents trespassing which can damage wiring etc.	L	H	The system will have to be protected. The wires covered and the greenhouse should limit insect or rodent entry.
6.	There could be a shock/electrocution hazard, due to the wet environment and electrical components	L-M	H	Regular maintenance should be performed to ensure wire are properly connected.
7.	The electrical components create a fire hazard	L	H	Regular maintenance should be performed. Fire extinguishers and smoke detectors should be placed near-by in case of fires.
8.	The web app not working properly (glitches)	L	M	Thorough testing of the app

Table: Risks

5.2 Assumptions

The following table lists the items that cannot be proven or demonstrated when this Project Charter was prepared, but they are taken into account to stabilize the project approach or planning.

No.	It is assumed that:
1	The final products will be in a weather resistant enclosure.
2	The customer will either use a web interface or application to interact with the system.

3	The user is using 115V AC system, with support for 230V in the future.

5.3 Constraints

The following table lists the conditional factors within which the project must operate or fit.

No.	Category	Constraints
1	Logistics	Some members will be working remotely, without access to the hardware.
2	Budget	Limited funding from team members' personal funds.
3	Technical	Not allowed to work with a voltage higher than 50v DC.
4	Reliability	Since the system is expected to be installed in a greenhouse. It must withstand the humid conditions inside of a greenhouse. Therefore, all electrical components must be grounded and insulated to protect the electronics and prevent accidental electric shocks to the user.

Table: Constraints

Section 6. Test Plan

6.1 Hardware and Control Software Test Plan

The hardware and control software test plan involves the methods used to test all the hardware components as well as the python control coding. This test plan will thoroughly test the sensors, Raspberry Pi functionality, hardware device functionality as well as on board code.

Sensors
The sensors in the greenhouse environment must be tested thoroughly for accuracy, functionality, as well as durability.

- Test for accurate readings from all sensors (Soil moisture, Temperature-Humidity, Light)
    - This will be done by comparing the results to other accurate tests. For example, the temperature sensor can be compared with a couple of regular thermometers to see if the results match.
    - Sensors will be tested multiple times to ensure the same readings are retrieved from the same conditions
- Test that the sensors will work effectively in a greenhouse environment with varying conditions
    - A greenhouse environment can have many different conditions that can be harsh to electronics. These conditions include wet, dry, or hot environments.
    - The sensors will be tested in the extreme ends of certain conditions to ensure they will be effective throughout the greenhouse's lifecycle.

Functional Components
The functional components, such as the fan, pump, mist maker, and grow light, will require rigorous testing to ensure their functionality and effectiveness.

- Proper functionality of the fan, pump, mist maker, and grow light will be tested
    - The components must be activated through Raspberry Pi when the correct conditions are met
    - The components must be activated through the web application
    - The components must be effective in varying greenhouse conditions
        - The components must be maintained to be water-resistant, or secure from any environmental damage that may occur in the greenhouse

Python and MySQL code
- Building and running the code with different conditions to find any runtime or logic errors.
- Ensuring that MySQL Connector works and enables out Python code to access the database.
- Testing thoroughly to check if the parsing functions can translate sensor readings into understandable data.
- Using MySQL's 'SELECT' query to fetch data from the database to make sure that the correct sensor readings have been store in the database.

- Hard-coding the sensor readings and comparing them with the settings data to test if our code works in conjunction with hardware to perform certain tasks. For example, turning on the water pump, turning on/off the grow lights, turning on the Mist Maker, etc.

6.2 Web Application Test Plan

Frontend

The frontend of the web application uses a React UI to display content and navigate the webpage, so this is what the frontend test plan will address. The test cases are as follows:

- Proof-of-Concept Webpage

A template website will be created with all the various pages, and the navigation will be tested to ensure that all pages can be reached.

- Data Graphing

A dummy data set will be used to ensure that data is displayed to the user graphically on the home page and historical data page and that the timescale can be modified as desired.

- Dynamic Graphs

A dynamic dummy data set will be used to test the live graphing capabilities (i.e., ensure that the graphs update in real time as data is added).

Backend

The backend of the web application uses the Django framework alongside a SQL DB to provide data to the frontend of the web application. Thus, the test cases for the backend are as follows:

- Get data from DB

The reading of data from the MySQL DB through MySQL connector will have to be tested to ensure that the backend can retrieve sensor readings to supply to the frontend. We will have to test this functionality for varying timeframes of data (1 day, 3 days, 7 days, 2 weeks, and 1 month), and for readings from all the sensors (ambient light, ambient temperature, soil humidity, and ambient humidity).

- Update soil and ambient environment settings in DB

The backend must be able to update setpoints for soil and ambient environment parameters in the database.

- Add user to DB

The backend must be able to add a user with login credentials to the database.

- Change user password in DB

The backend must be able to change a user’s password that is stored in the MySQL DB

- Get user password from DB

The backend must be able to email the user for password recovery. This may also necessitate setting up a mail server.

Hosting

The hosting aspects of the web application must be tested to ensure that the web application is reachable from the web.

- Dynamic DNS

The DDNS must be pointing to the correct IP address, and automatically update whenever the Raspberry Pi connects to the internet.

- Reverse Proxy

The domain name from DDNS must be routed to the correct port and IP address.

- Apache Server

The website must be correctly hosted on the web server.

- Mail server

The mail server must be able to send an email to the user containing password reset instructions

6.3 Integration Test Plan

Integration Testing (Frontend and Backend)

In order to ensure the correct functioning of the web application, integration testing of the whole web application must be performed to ensure the frontend and backend properly communicate via the REST API. The test cases are as follows:

- User login from login page

The user must be able to login to the web application from the UI using valid credentials. Invalid logins must be tested as well, and must be rejected.

- Update settings from settings page

The user must be able to update settings from the settings page, and these changes must be reflected in the database.

- Update database with new readings

When the database is manually updated with new sensor readings, these updated values must be displayed graphically on Historical Data page and on the Home Page.

- Add user from registration page

The user must be able to add a new user from the registration page, and the credentials must be added to the database.

- Password recovery page

An email must be sent to the user containing password reset instructions when the password recovery page is used.

Integration Testing (Whole Greenhouse System)

In order to ensure correct functioning of the entire greenhouse system, integration testing must be performed to test communication between the frontend, backend, control software, and hardware. The following test case will be used:

- Update settings to induce change in physical greenhouse and observe result

Firstly, a setpoint will be changed in the settings page of the web application that will induce a change in greenhouse operation (Ex. Raise light setting to threshold where light must turn on). Next, it must be checked that the desired change is made (Ex. The light inside the physical greenhouse turns on). Finally, it must be ensured that the change in state is reflected in the historical data in the web application (Ex. The light levels increase gradually to the new setpoint). This test must be repeated for all controllable setpoints in web application.

Section 7. Proof of Concept

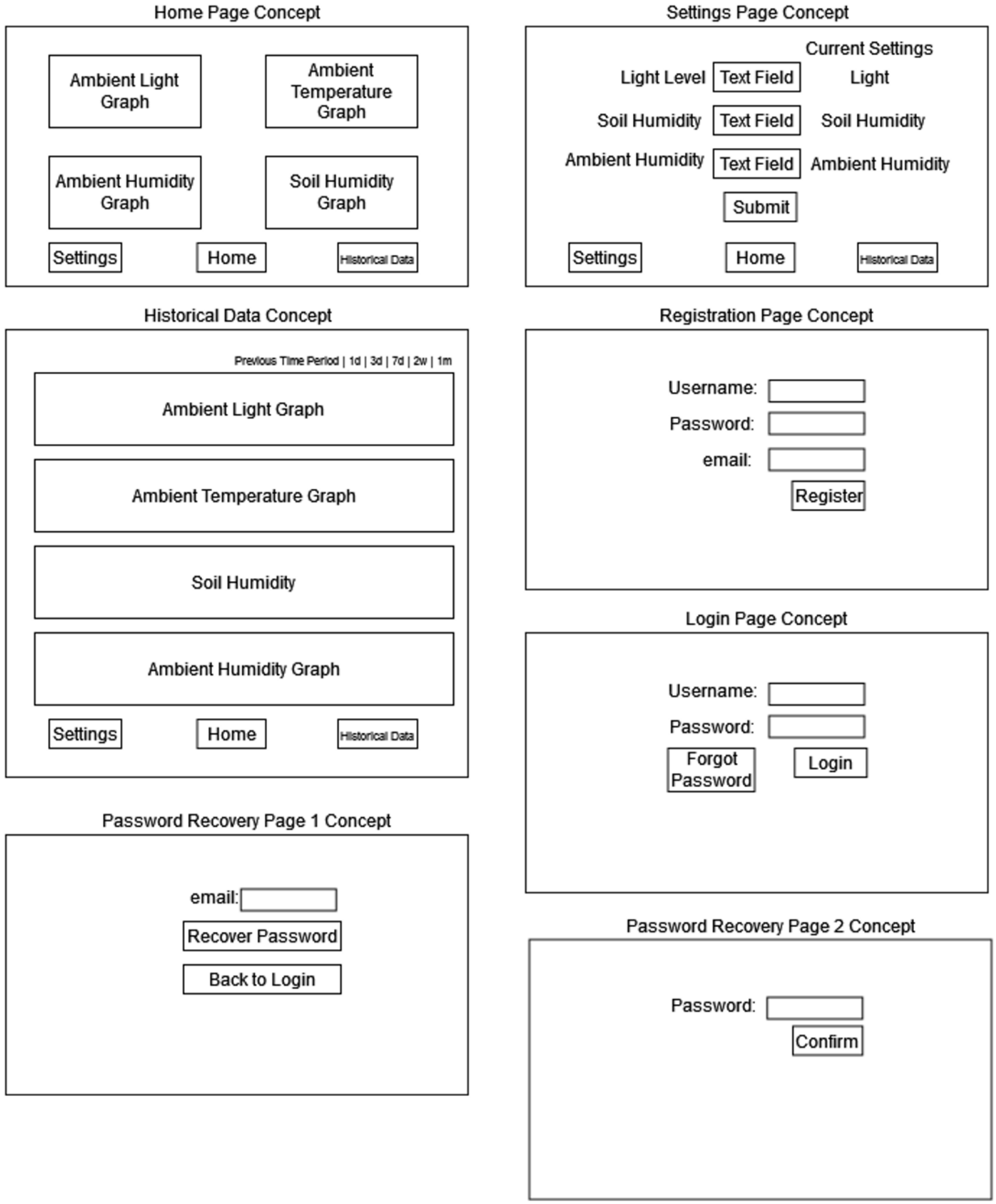

Figure 11: Proof of Concept

Section 8. PPA

In hindsight, we realize we could have scheduled our work more efficiently. All tasks were completed, but at times with little time to spare. In the future we have agreed to schedule tasks ahead of time and have them completed well before the deadline.

On the positive side, we were able to collaborate on tasks very well. Allowing each of us to show our individual strengths with a clearly stated division of the work. The Trello board was especially useful in organizing work that was to be done and completed.