Soilmate

ILLINOIS | *Team 32*

Electrical & Computer Engineering

Team Members

Sigrior Vauhkonen

Ysabella Lucero

Emma Hoeger

Overview

1

Objective

Problem, Solution, and High-Level Requirements

2

Design

Design Overview and Subsystem Verification

3

Modifications

Revisions and Final Build

4

Conclusion

Successes and Future Improvements

The Problem

Many houseplant owners struggle to take proper care of their plants:

Of millennial plant owners…

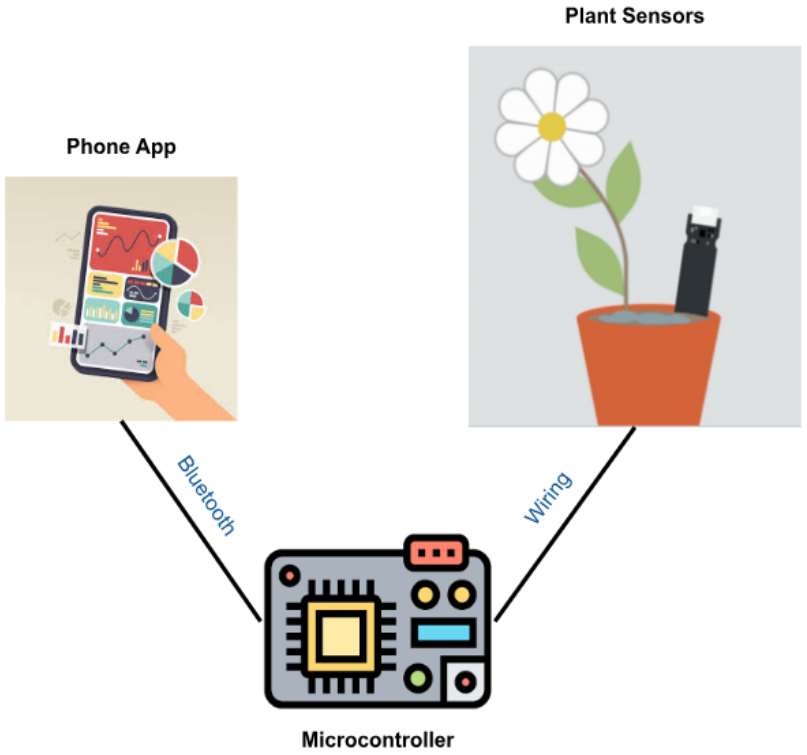
7
…is the average number of plants one has killed

48%
…stress about keeping their plants alive

60%
…worry about ensuring proper sunlight exposure

56%
…are concerned about how much to water their plant

Solution



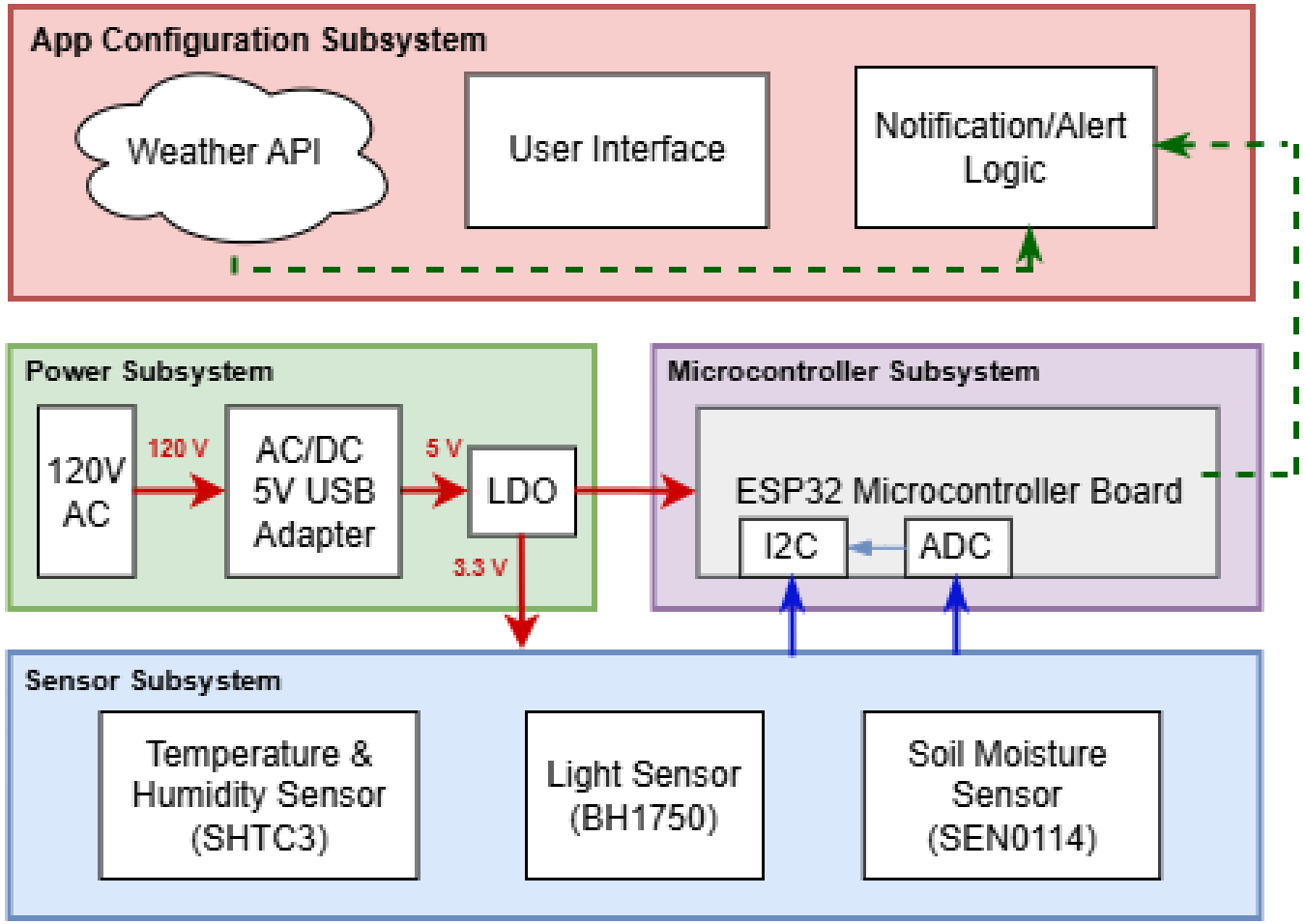
High Level Requirements:

1. The temperature and humidity sensor, light sensor, and soil moisture sensor must gather accurate and usable data.

2. The microcontroller must be able to send sensor data via Bluetooth to the mobile app.

3. The mobile app must send sensible notifications to the user about the care of the plant, utilizing the sensor data and weather API data.

Block Diagram



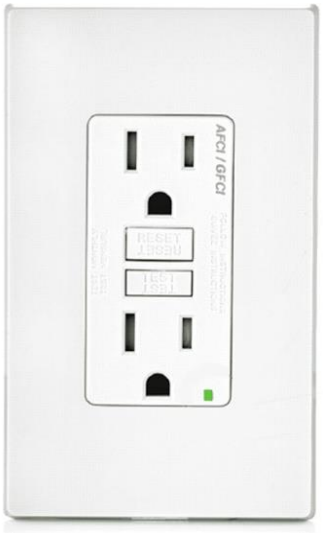
Power Subsystem

Requirements:

- Wall adaptor must output a voltage of 2.5V to 5.5V to connect to our regulator
- LDO voltage regulator must output a voltage of 3.0V to 3.6V to power all sensors and MCU

Verifications:

- Both requirements can be verified by probing the output pins on the barrel jack connecter and LDO using a DMM to measure voltage



120V

5V

3.3V

Sensors & Microcontroller

Microcontroller Subsystem

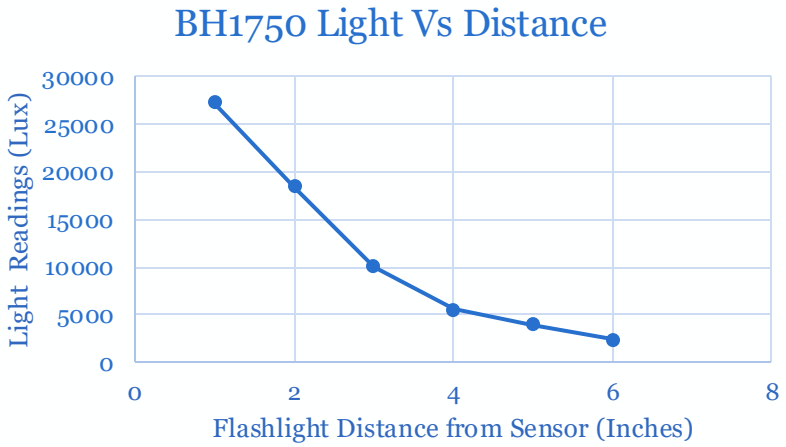
Requirements for Microcontroller:

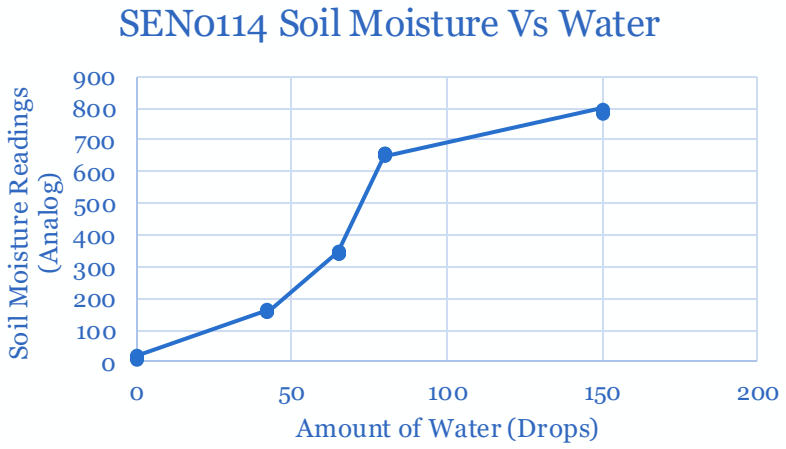
1. Must successfully transfer sensor data to app via Bluetooth
2. Must remain operational during transmission and gather sensor data without resetting/shutting down

Verification For Microcontroller:

- Establish a two-way serial Bluetooth communication and request data from I2C bus to send to the app
- With the previous requirement in place, create a loop to continuously pull sensor data at least 100 times and ensure data is expected

Sensors Subsystem





Requirements for Light Sensor:

1. Must give accurate data of ambient light conditions

Verification For Light Sensor:

- Shined flashlight at varying distances and documented sensor readings
- Compared values with Lux Light Meter Pro App

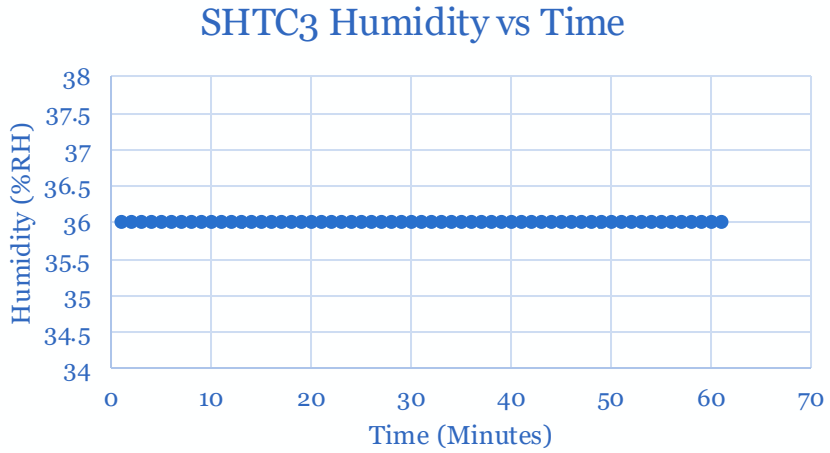
Requirements for Soil Moisture Sensor:

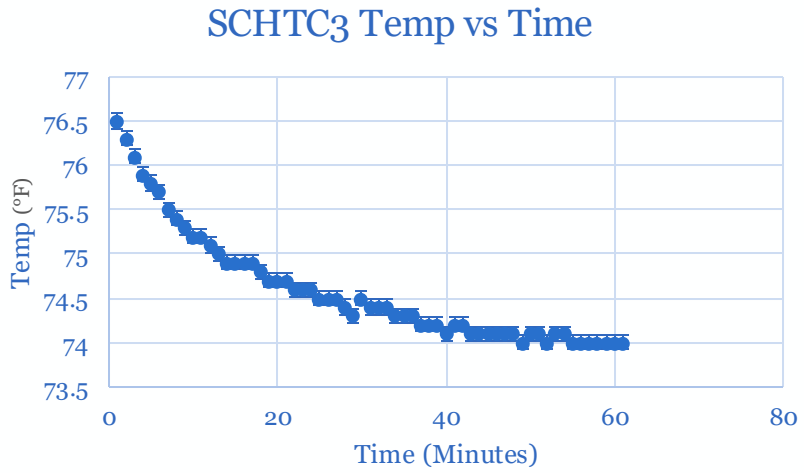
1. Must output 0-300 for dry soil
2. Must output 300-600 for moist soil
3. Must output 600-900 for saturated soil

Verification For Soil Moisture Sensor:

- Added increasing amounts of water to dry plant and documented sensor readings

Sensors Subsystem





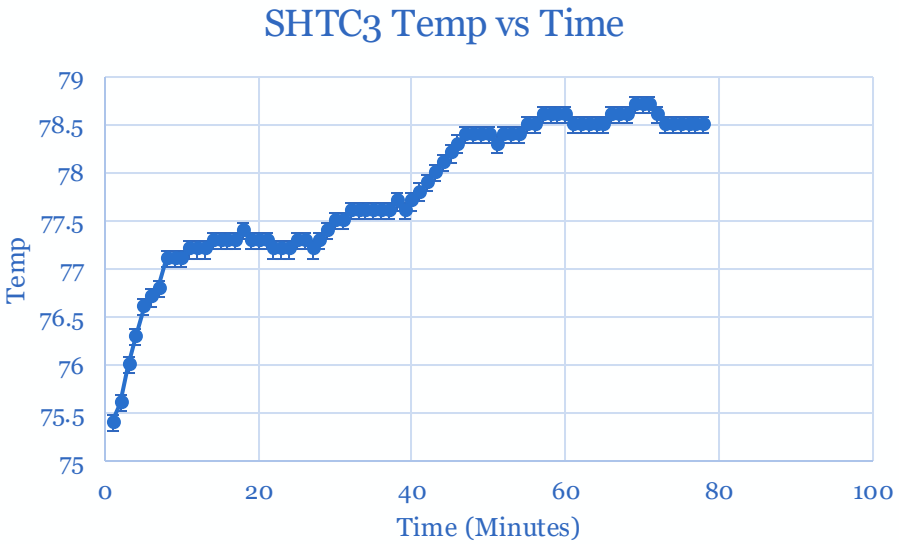
Requirements for Temp/Humidity Sensor:

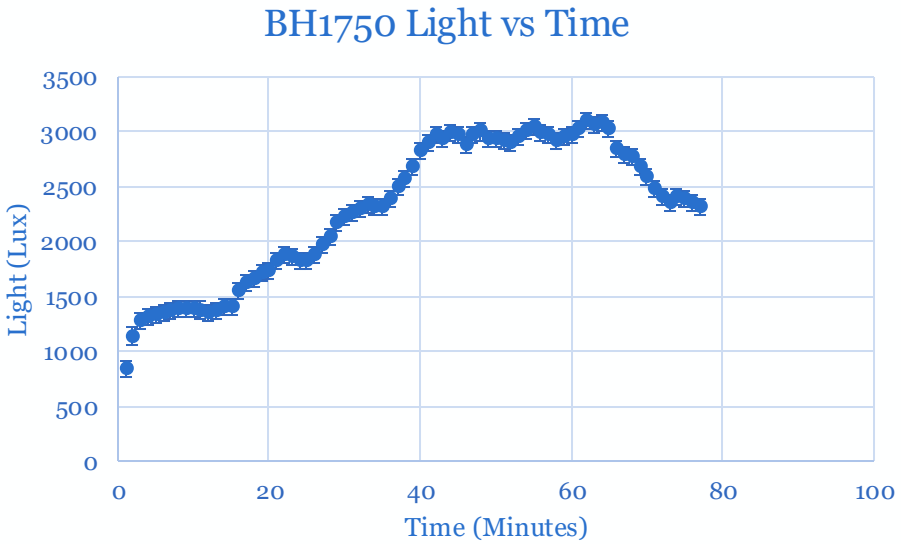
1. Must give accurate data of ambient temperature conditions
2. Must give accurate data of ambient humidity conditions

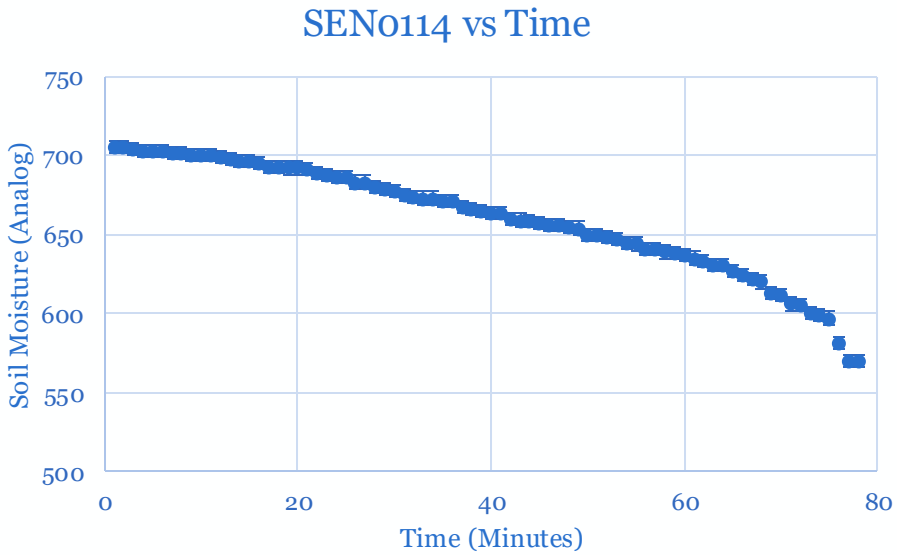
Verification For Temp/Humidity Sensor:

- Continuously measure temp/humidity conditions for hours to see fluctuations
- Compare readings with digital thermometer and digital hygrometer to determine accuracy

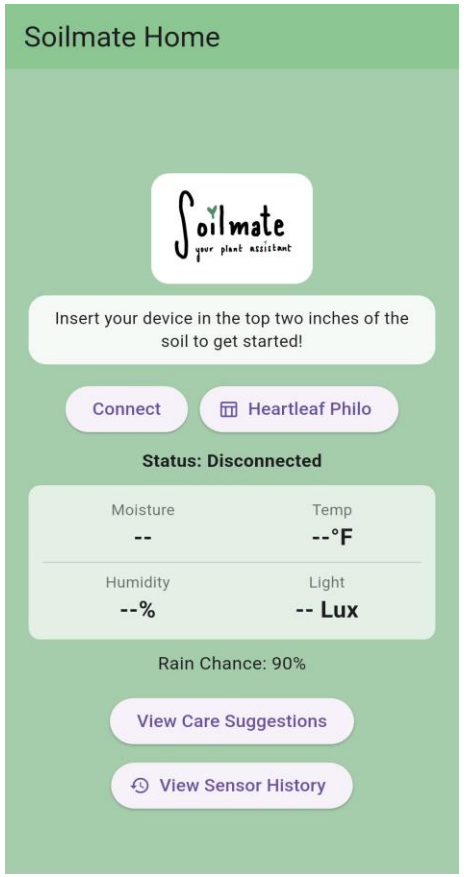
Sensors Subsystem







App Configuration Subsystem



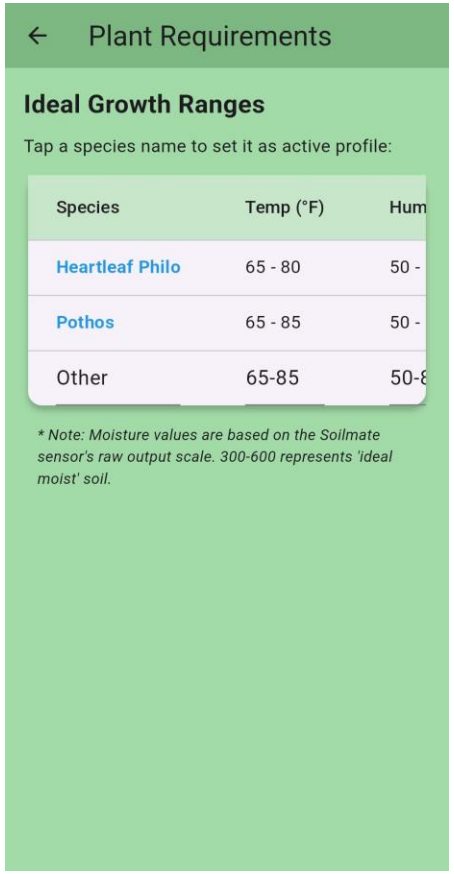
Requirements for Mobile Application:

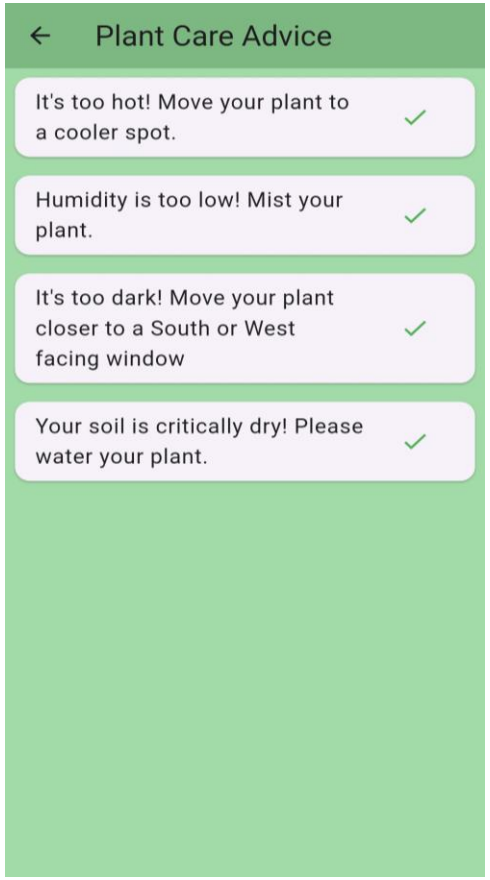
1. The app must display accurate BLE data from the ESP32 microcontroller
2. The app must only send a watering suggestion when the correct conditions are met, and not otherwise.
3. The app must generate and send care notifications after receiving sensor data that indicates plant conditions are not met.

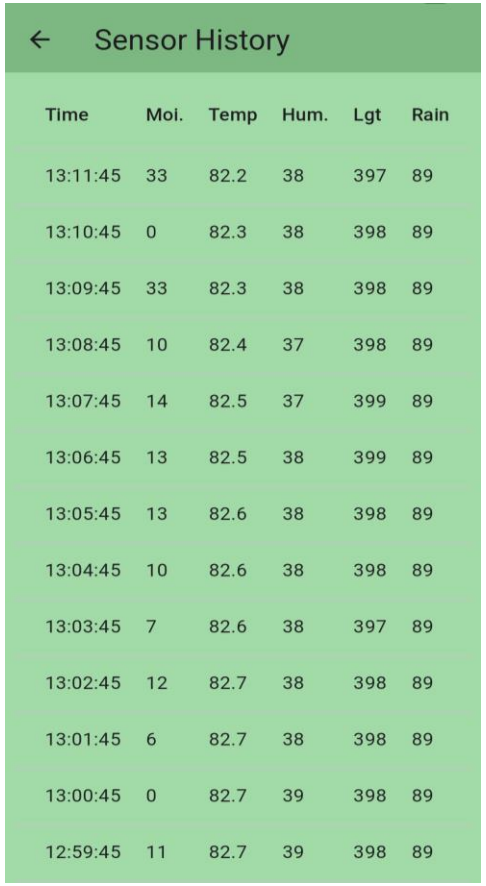
Verification For Mobile Application:

- Send known test values from ESP32, confirm that values match in the app UI
- Check that the following conditions generate an alert:
  - Bone-dry soil
  - Oversaturated soil
  - Soil is not oversaturated and chance of rain is above 50%
- Check that alerts are generated for unsuitable temperature, humidity, and light conditions

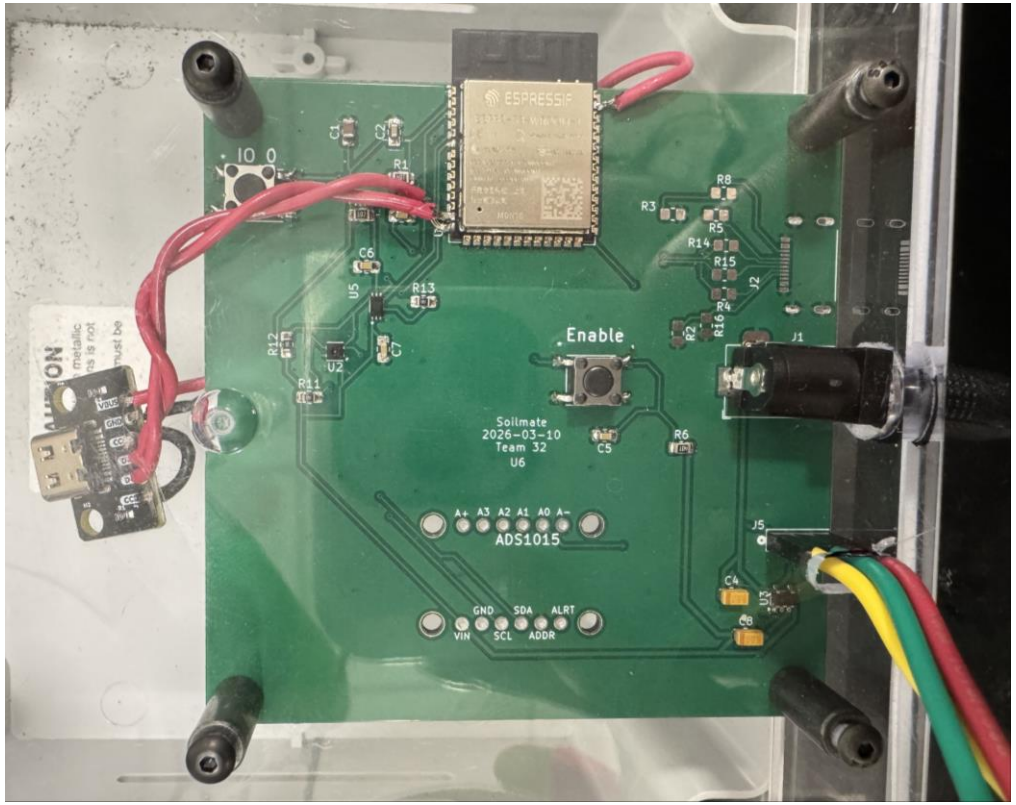
App Configuration Subsystem

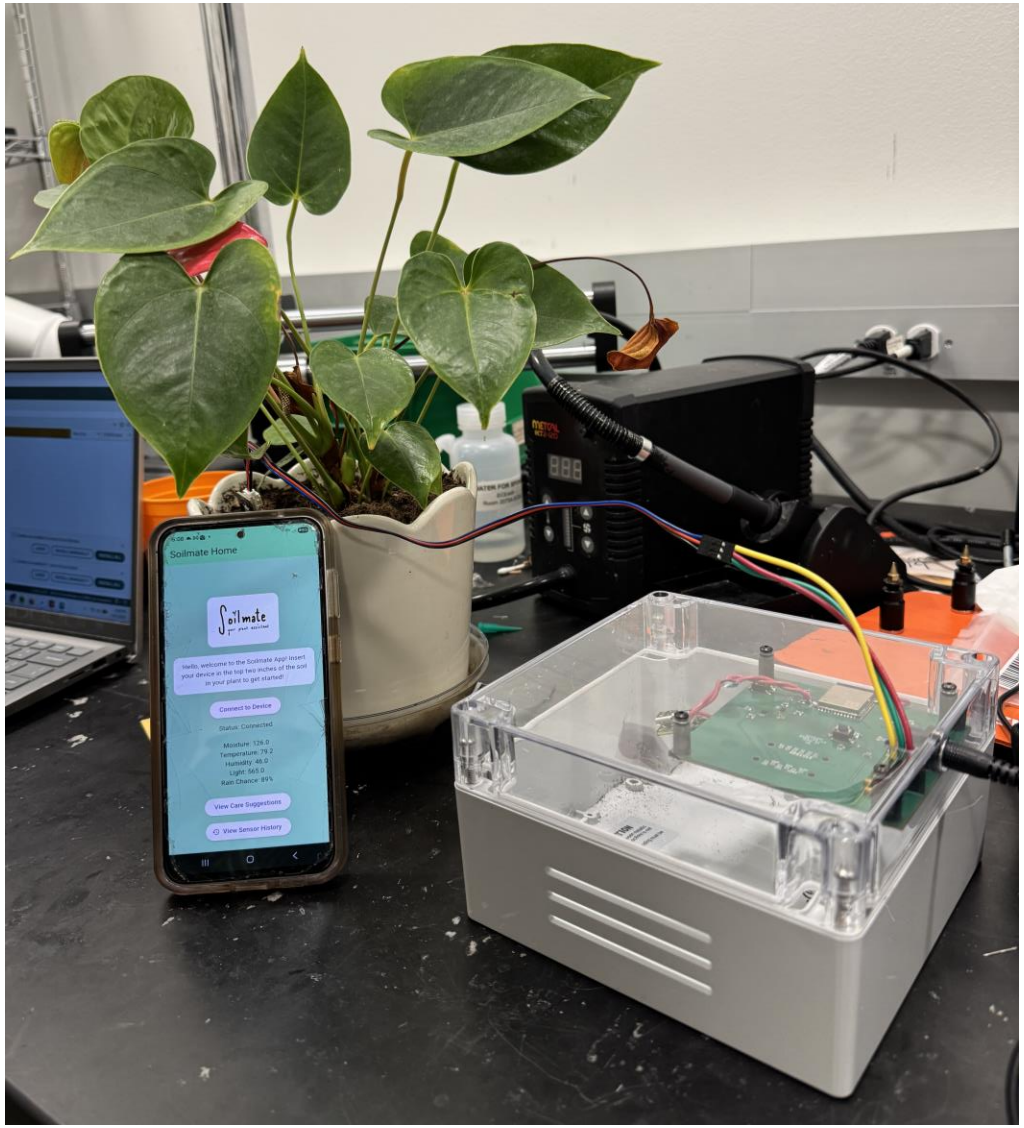






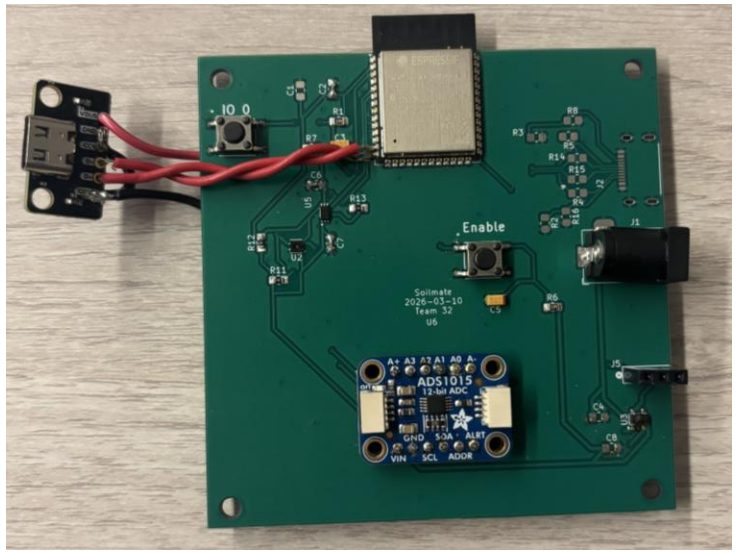
Complete Physical Build

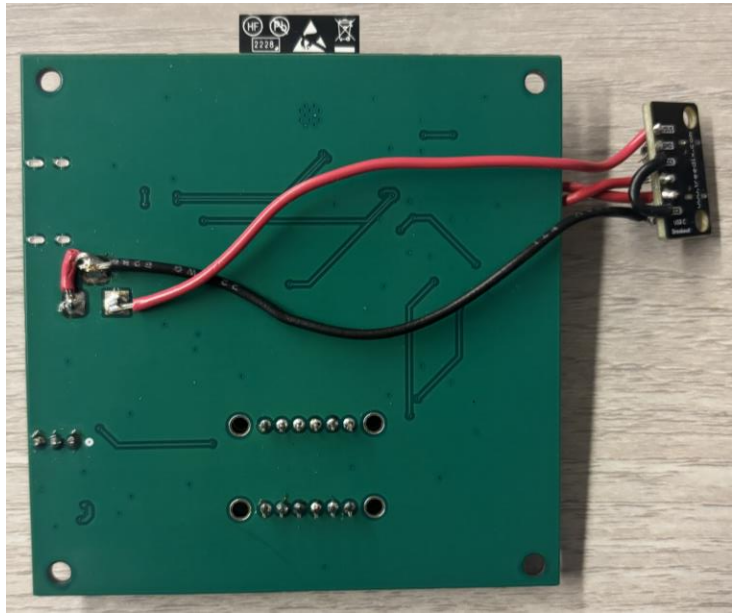




Revisions

- Changed 6V wall adapter to 5V
- Shorted ground pins on barrel jack together
- Incorrect footprint for USB-C, so we soldered a USB-C breakout board to the microcontroller
- Included pin headers to connect GPIO0 and Enable pin to ground but switched to buttons
- Used ADC on chip rather than ADC break out module





Successes

- ✓ Power subsystem delivered correct voltages
- ✓ Programmed microcontroller using through USB-C connection
- ✓ Accurate data collection from all sensors over long periods of time
- ✓ Displayed sensor readings on the app
- ✓ Integrated weather API into app
- ✓ Generated plant care suggestions and sent notifications based on sensor and weather data

Future Improvements

- Implement battery power for portable applications
- Reduce size of enclosure and use epoxy between wires and cutouts to make product waterproof
- Allow user to connect to multiple plants through one app
- Include a pH sensor to collect data about the pH levels
- Send notifications confirming that issues are resolved once suggestions are addressed by the user and environment is optimal

THANK YOU!

ILLINOIS

Electrical & Computer Engineering