Nurture: Autonomous Plant Care System

sddec24-16

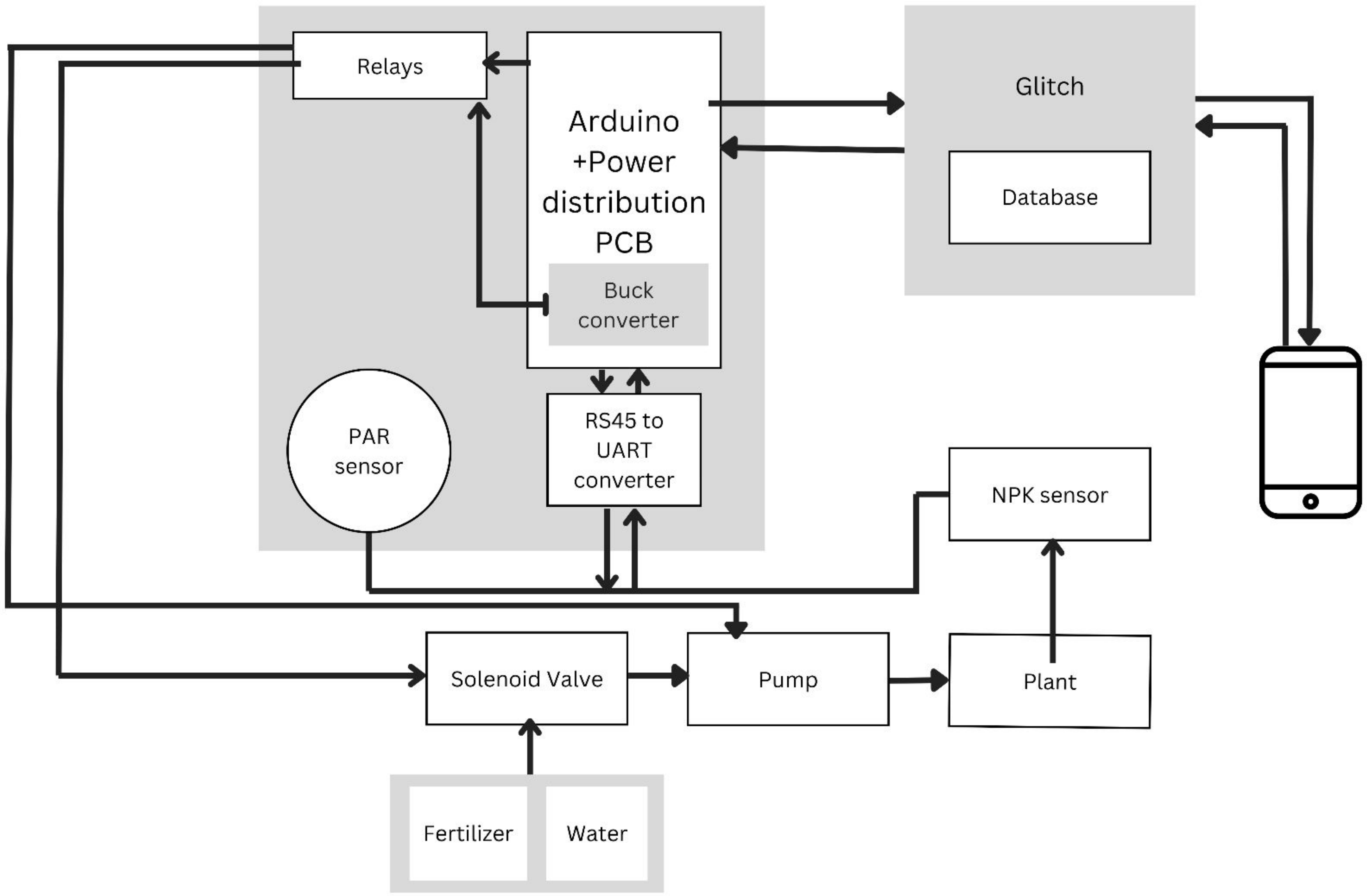
Cameron Jones, Chase O’Connell, Cayden Kelley, Blake Hardy, Tejal Devshetwar, Holden Brown

Advisor: Md Maruf Ahamed

Product Need + Goals

- **Convenience:**
  - Eliminates Guesswork (watering, fertilizing).
  - Automates Critical Tasks.
- **Accessibility:**
  - Empowers beginners
  - Affordable Design
  - App Remote Monitoring
- **Reliability:**
  - Consistent Real-Time Care
  - Reduced Human Error
- **Scalability:**
  - Adaptability to different plant types and growing conditions.

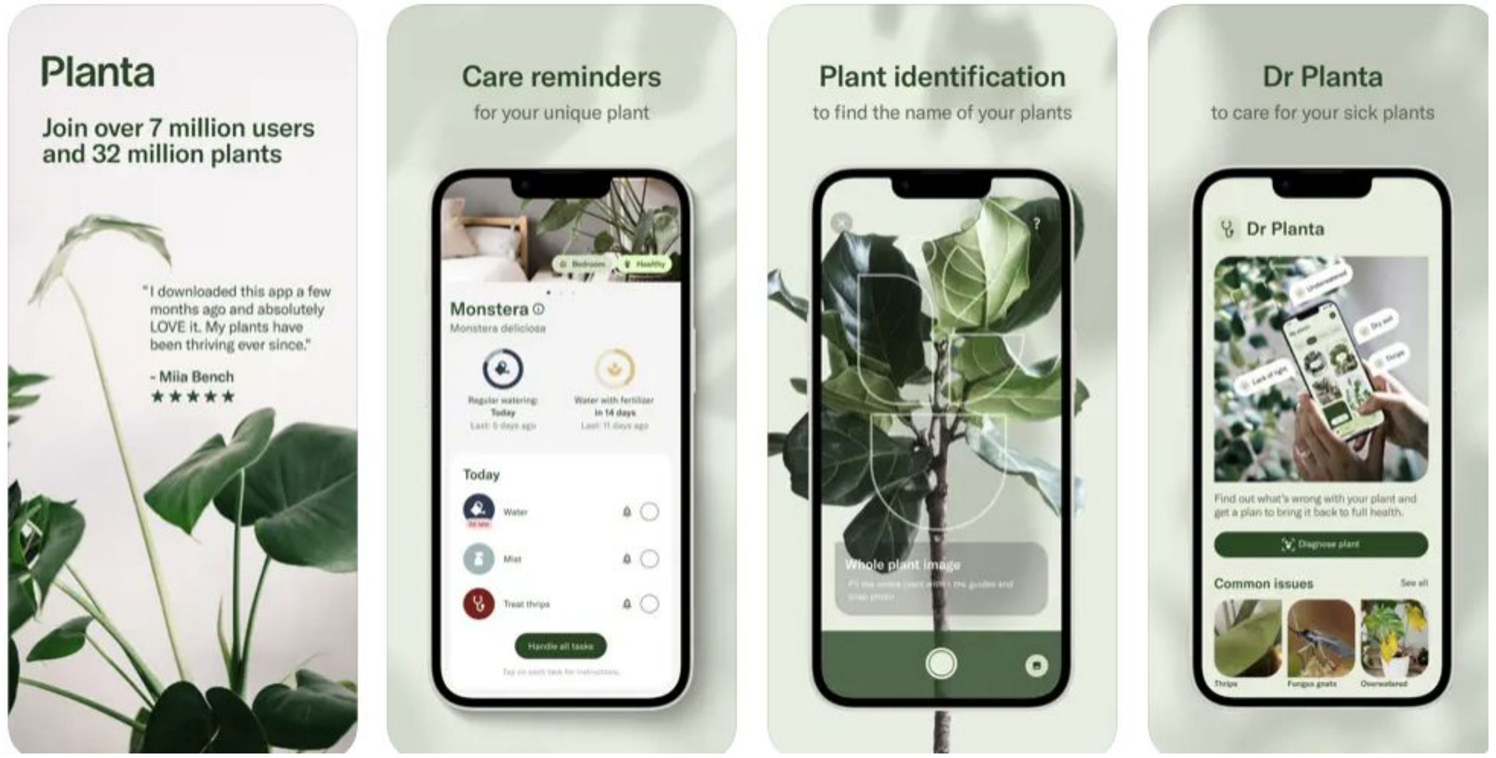
Project Overview



Market Research

LetPot

Planta



Planta

Join over 7 million users and 32 million plants

"I downloaded this app a few months ago and absolutely LOVE it. My plants have been thriving ever since."

- Miia Bench

★★★★★

Care reminders

for your unique plant

Monstera

Monstera deliciosa

Regular watering: Today

Water with fertilizer in 14 days

Today

Water

Mist

Treat thrips

Handle all tasks

Plant identification

to find the name of your plants

Whole plant image

Dr Planta

to care for your sick plants

Dr Planta

Find out what's wrong with your plant and get a plan to bring it back to full health.

Diagnose plant

Common issues

See all

Thrips

Fungus gnats

Overwatered

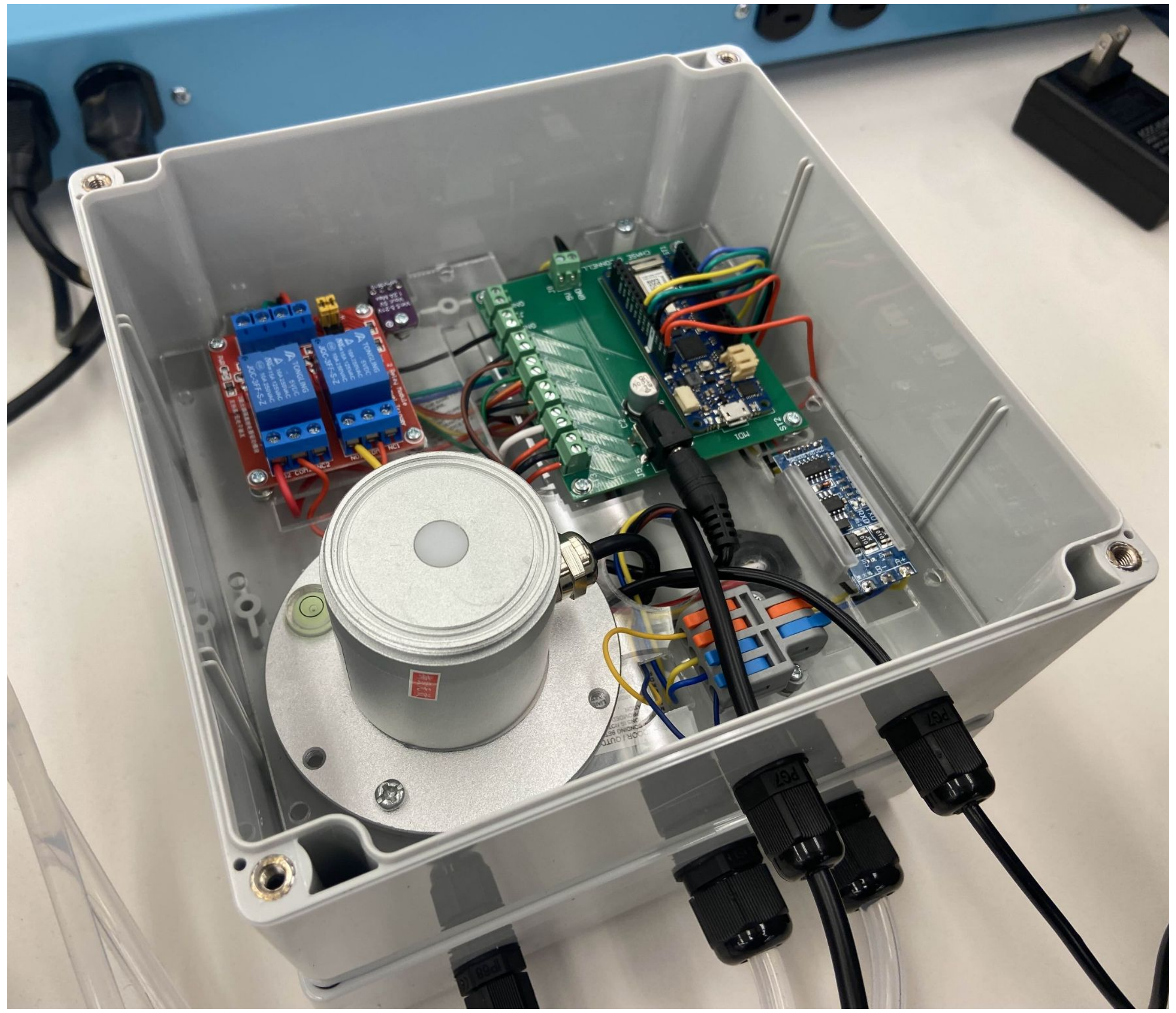
Sinbeda

Product Comparison

	Sensor Connectivity	Soil Moisture Monitor	Soil Nutrient Monitor	Automated Watering and Fertilizing	Plant Care Recommendations
LetPot	Bluetooth	No	No	Watering	No
Planta	No Sensor (Camera)	No	No	No	Yes
Sinbeda	Bluetooth	Yes	Yes	No	Yes
Our Design	WiFi	Yes	Yes	Yes	Sensor-Based Recommendations

Hardware Design

Hardware Requirements



Resources:

- **Power:** AC Adapter
- **Liquid System:** Pump, Solenoid Valve, Pipes
- **Sensors:** Soil Metrics, Light
- **General:** Microcontroller, IP68 Enclosure

Functional:

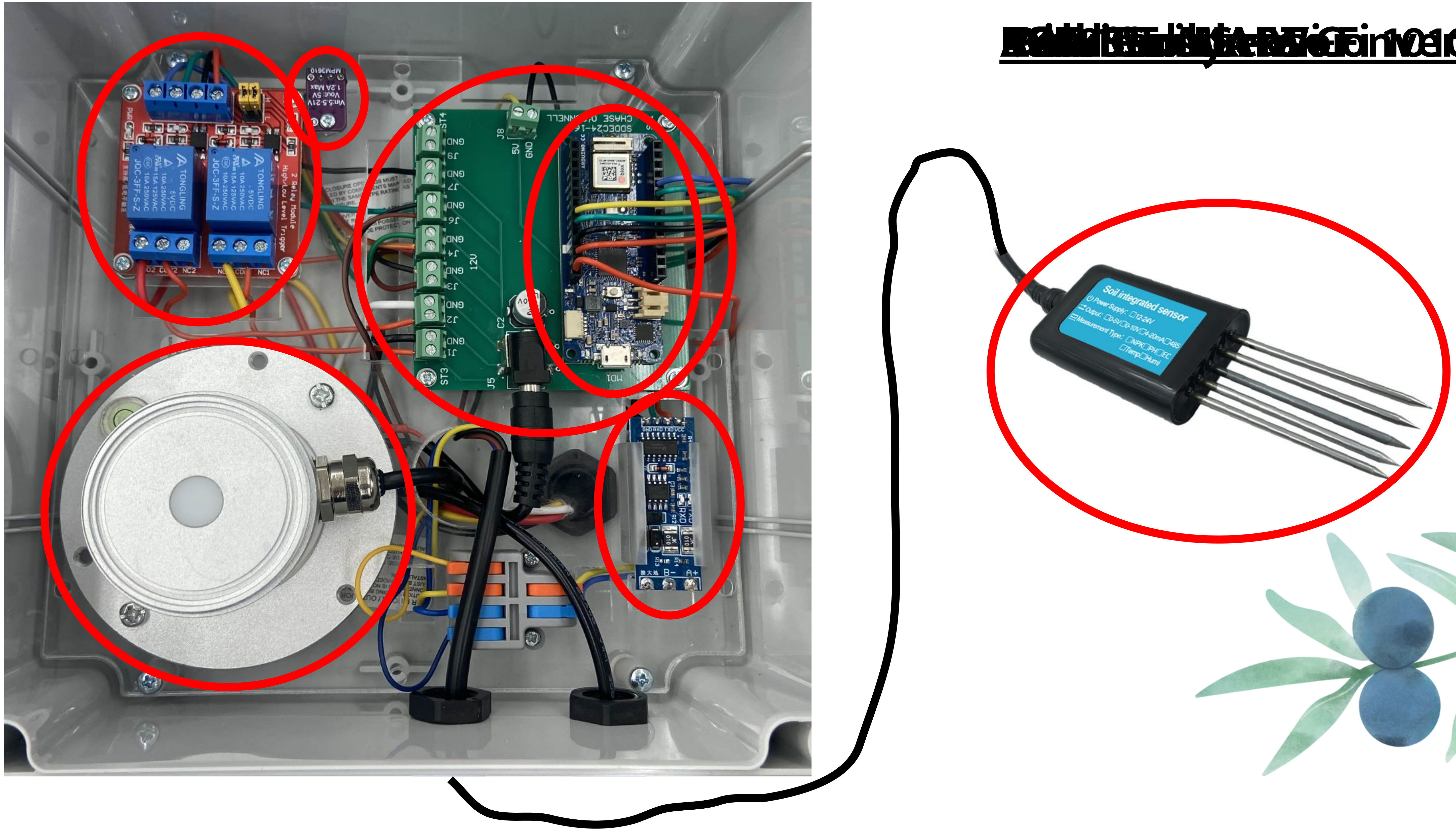
- Working automated liquid distribution.
- Sensor data transmission.

Non-Functional:

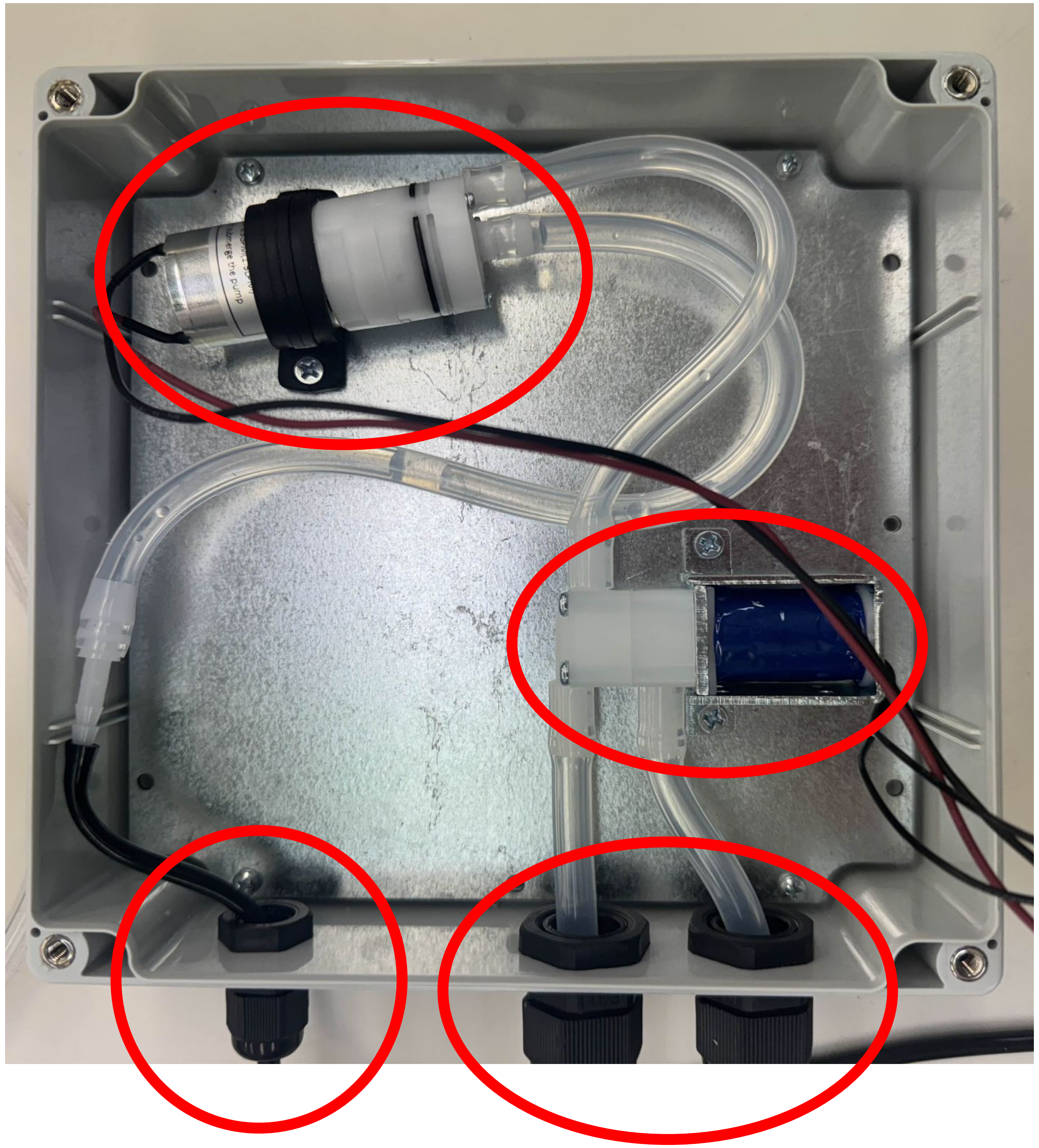
- **Size:** 3+ inch diameter pot compatibility.

Hardware

[illegible]


2 Relay Module High/Low Level Trigger
TONGLING
JQC-3FF-S-Z
MPM3610
Vin:5-21V
Vout: 5V
1.2A Max
GND
J8
5V
12V
Soil integrated sensor
Power Supply: □12-24V
Output: □0-5V□0-10V□4-20mA□485
Measurement Type: □NPK□PH□EC □Temp□Humi

Hardware



Inputs: Water (Right) & Fertilizer (Left)

Output

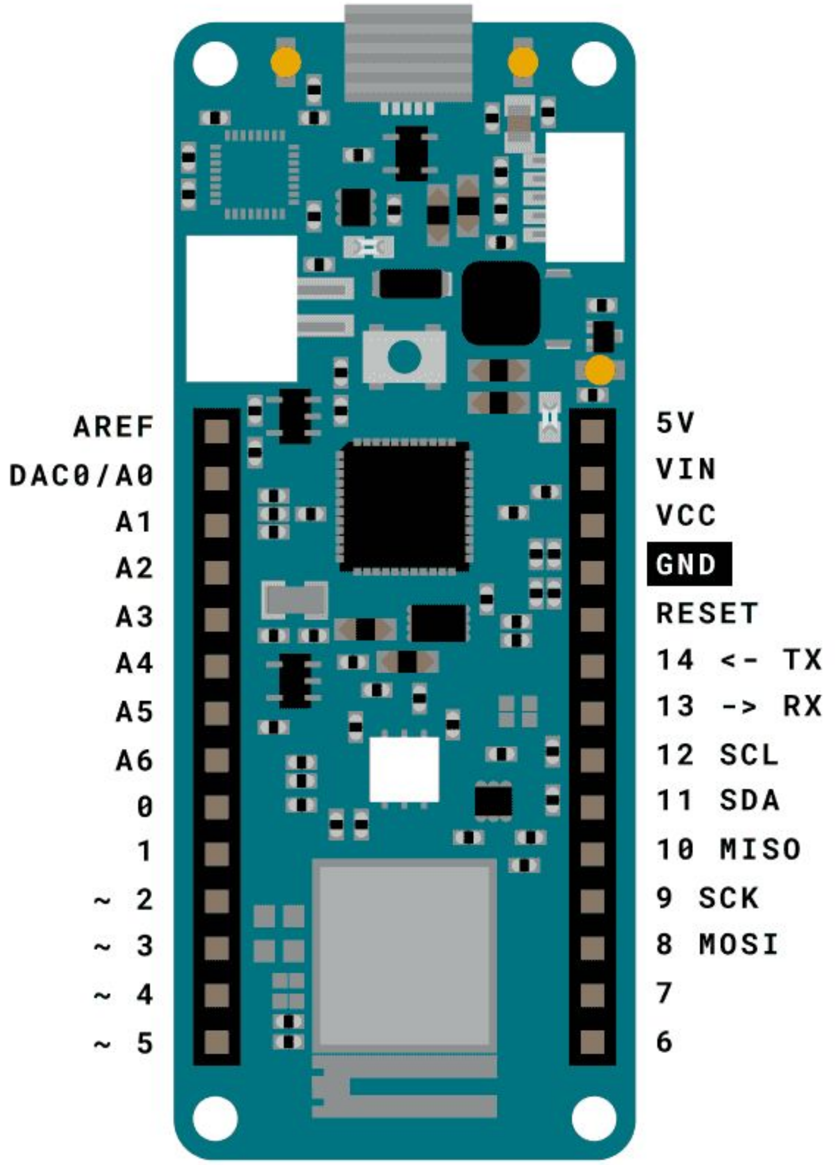
Solenoid Valve

Pump (Water or Fertilizer)

Microcontroller

Arduino MKR Wi-Fi 1010

- Low Power
- Small Form Factor
- Great Library Support:
  - Modbus
  - WIfinina + HTTP
  - Low-Power
  - FreeRTOS
- Wi-Fi Capable - Server Communication
- Sensor Data Handling
  - Format and send NPK + PAR data



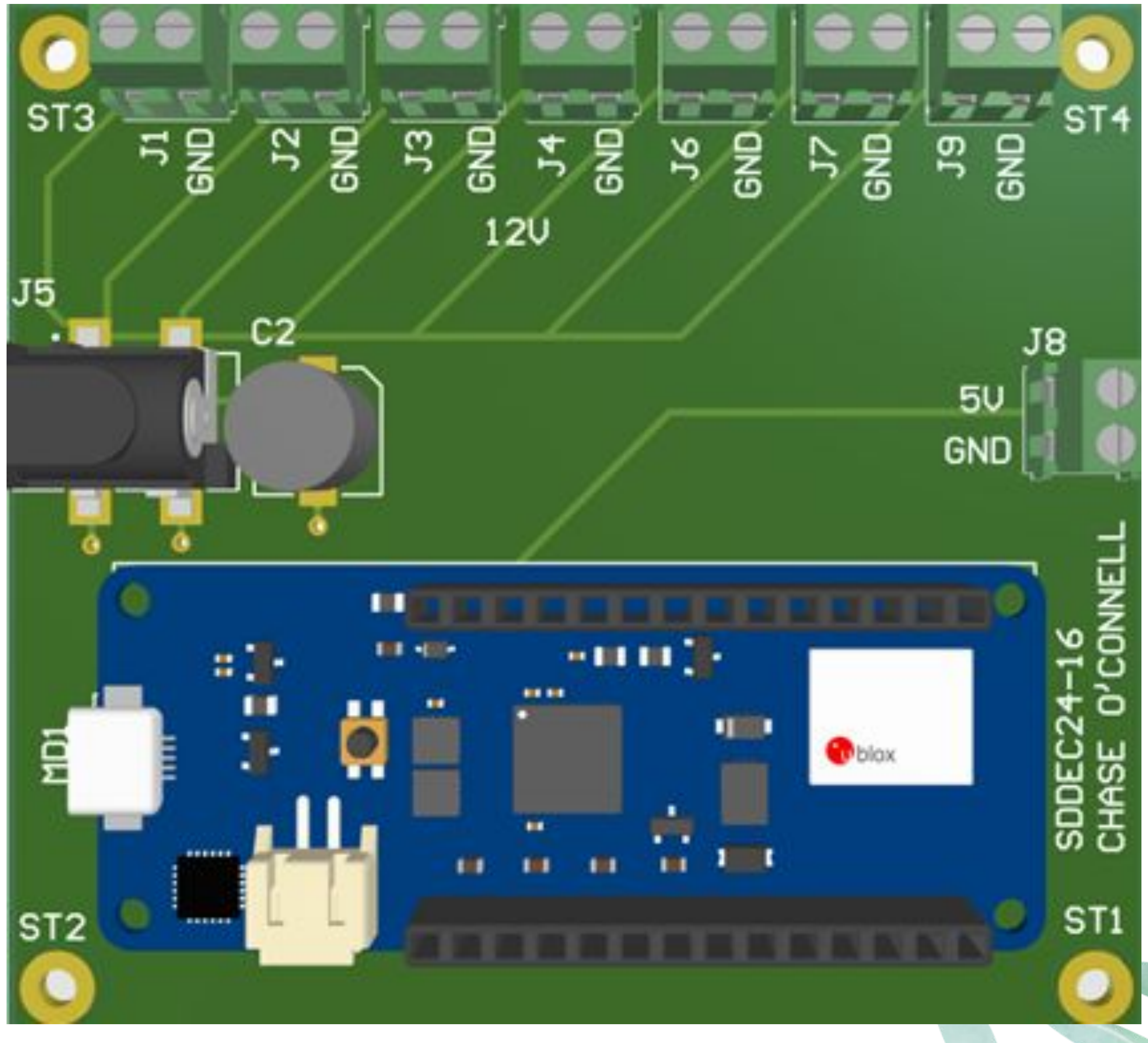
PCB Design

General:

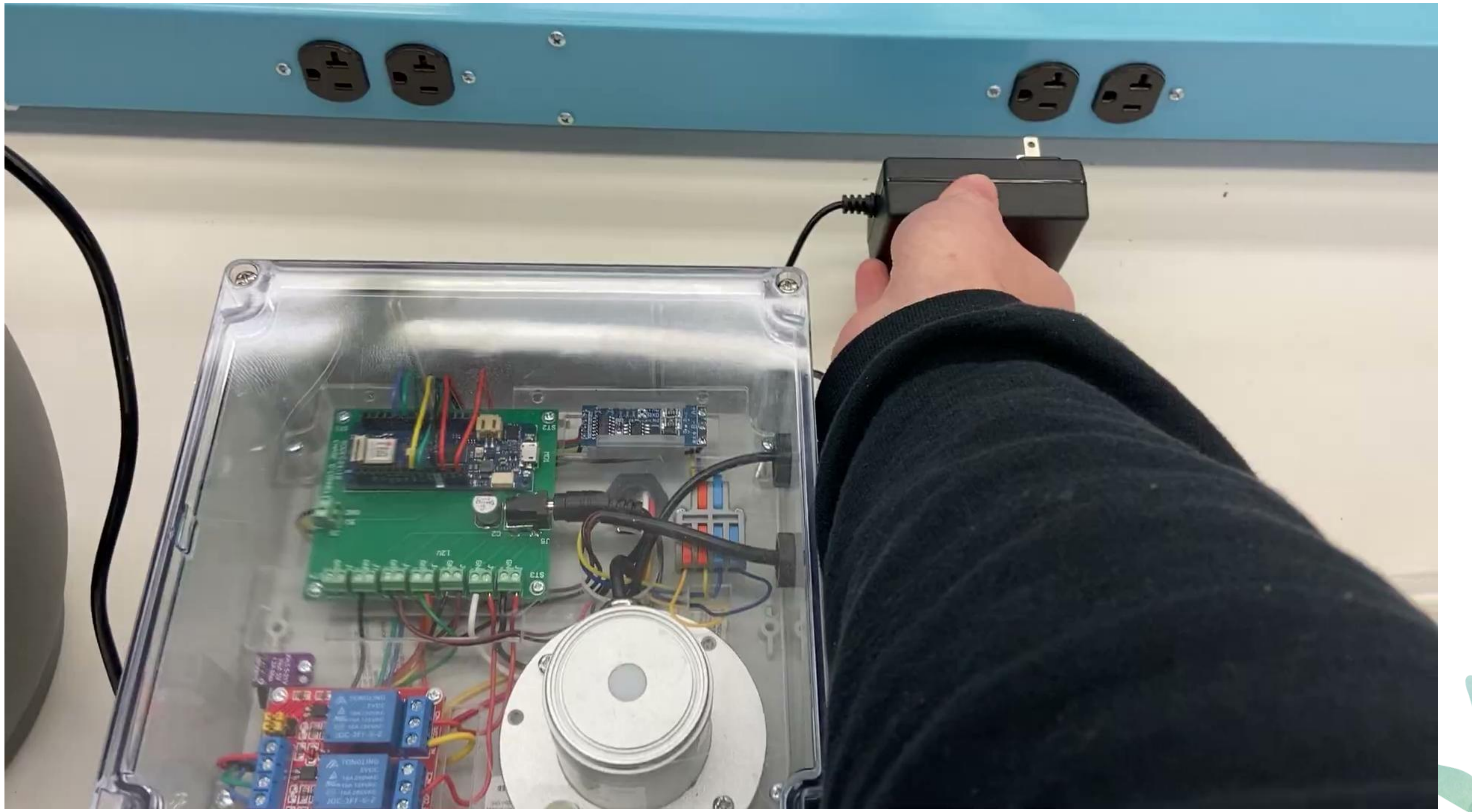
- Altium Designer
- Power Distribution to All Peripherals
- Time, Cost, Integration

Design:

- 2-Layer:
  - Top: 12V and 5V Power Traces
  - Bottom: GND Polygon Pour
- Components:
  - Terminal Blocks - Peripherals
  - Barrel Jack + Bulk Capacitor
  - Arduino Footprint - Mounting



Hardware Demo



Software Design

Software Requirements

Resources:

- **Development Tools:** React Native, MongoDB, Node.js
- **Server:** Glitch

Functional:

- **UI:** Graphical & Numerical Data, User Plant Creation + Viewing Pages
- **UX:** Reliable (no crashes, freezing), App Synchronization, Stored User Plant Data

Non-Functional:

- **Backend:** >99% Uptime, Secure
- **Codebase:** Modular

React Native

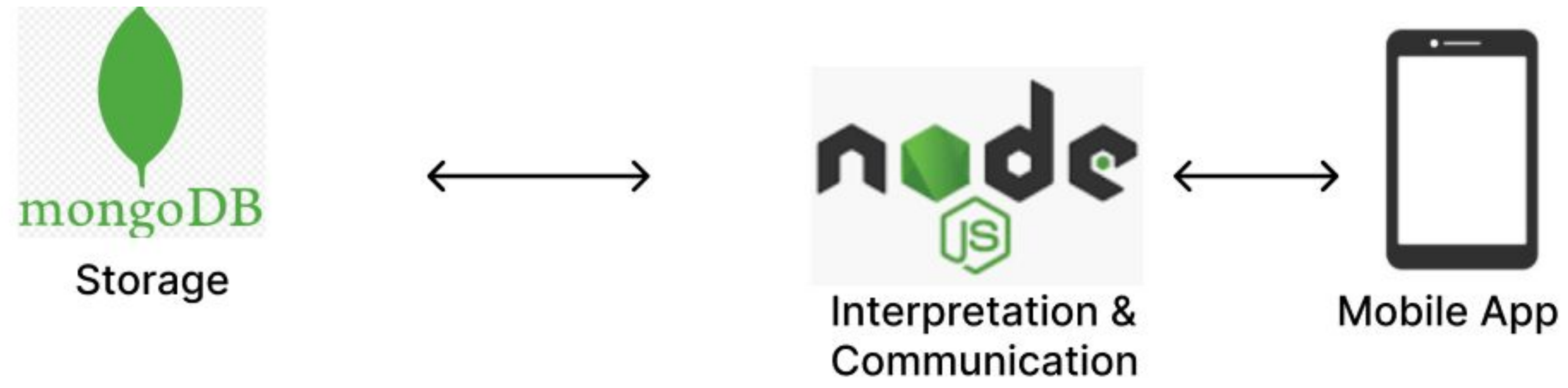
mongoDB

node JS

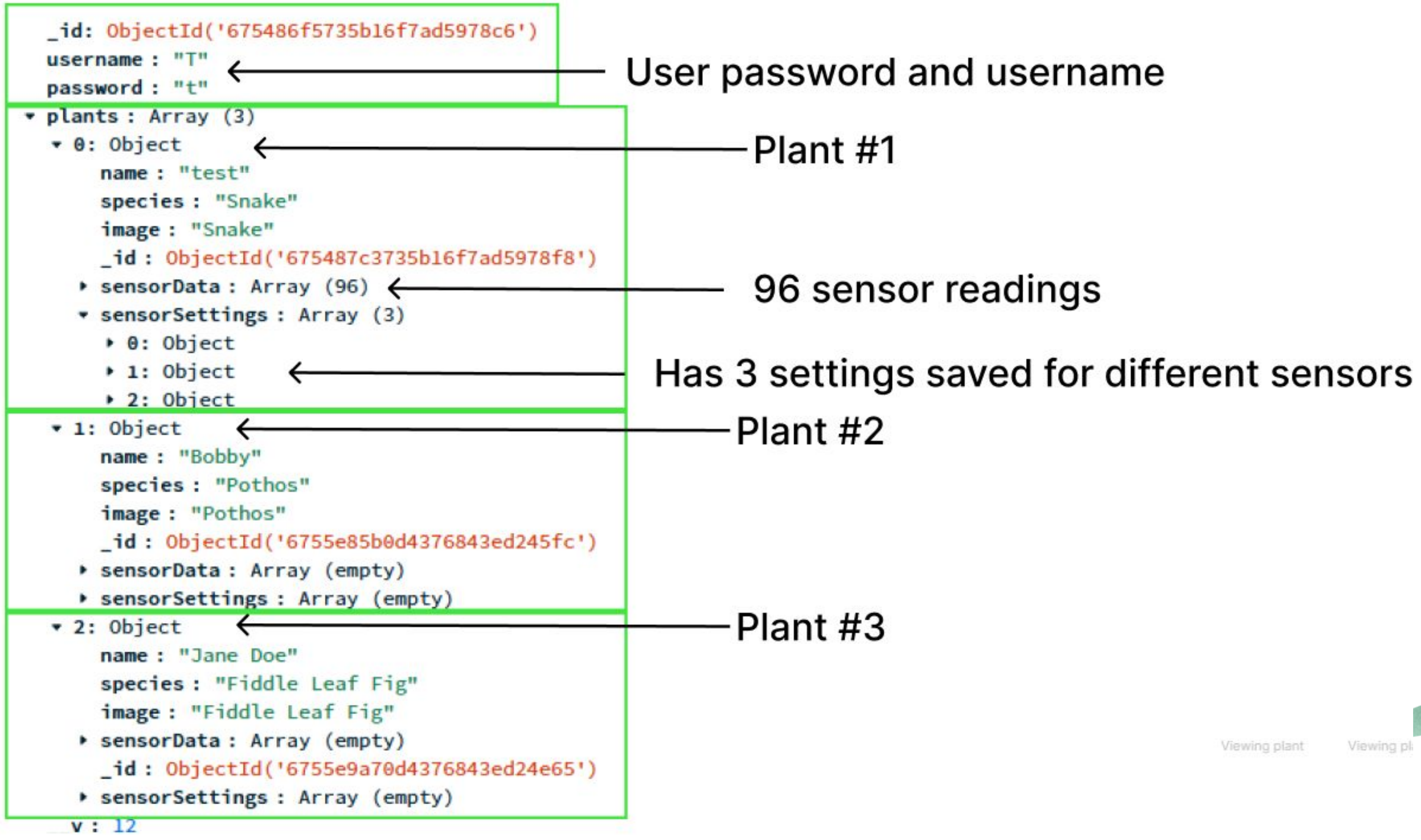
Glitch

Software Overview

- **Frontend:** App
  - Developed with React Native
  - Displays sensor data graphically.
  - Plant and user management
- **Backend:** Express, Mongoose, and MongoDB
  - Node.js: communication between the database and app.
  - MongoDB: stores user data
  - Nested schema for easy management
- **Key Technologies:**
  - Unified JavaScript stack for backend and frontend
  - Asynchronous programming model



User Object in Storage:



```
  _id: ObjectId('675486f5735b16f7ad5978c6')
  username : "T"
  password : "t"
▾ plants : Array (3)
  ▾ 0: Object
      name : "test"
      species : "Snake"
      image : "Snake"
      _id : ObjectId('675487c3735b16f7ad5978f8')
    ▸ sensorData : Array (96)
    ▾ sensorSettings : Array (3)
      ▸ 0: Object
      ▸ 1: Object
      ▸ 2: Object
  ▾ 1: Object
      name : "Bobby"
      species : "Pothos"
      image : "Pothos"
      _id : ObjectId('6755e85b0d4376843ed245fc')
    ▸ sensorData : Array (empty)
    ▸ sensorSettings : Array (empty)
  ▾ 2: Object
      name : "Jane Doe"
      species : "Fiddle Leaf Fig"
      image : "Fiddle Leaf Fig"
    ▸ sensorData : Array (empty)
      _id : ObjectId('6755e9a70d4376843ed24e65')
    ▸ sensorSettings : Array (empty)
  __v : 12
```

User password and username

Plant #1

96 sensor readings

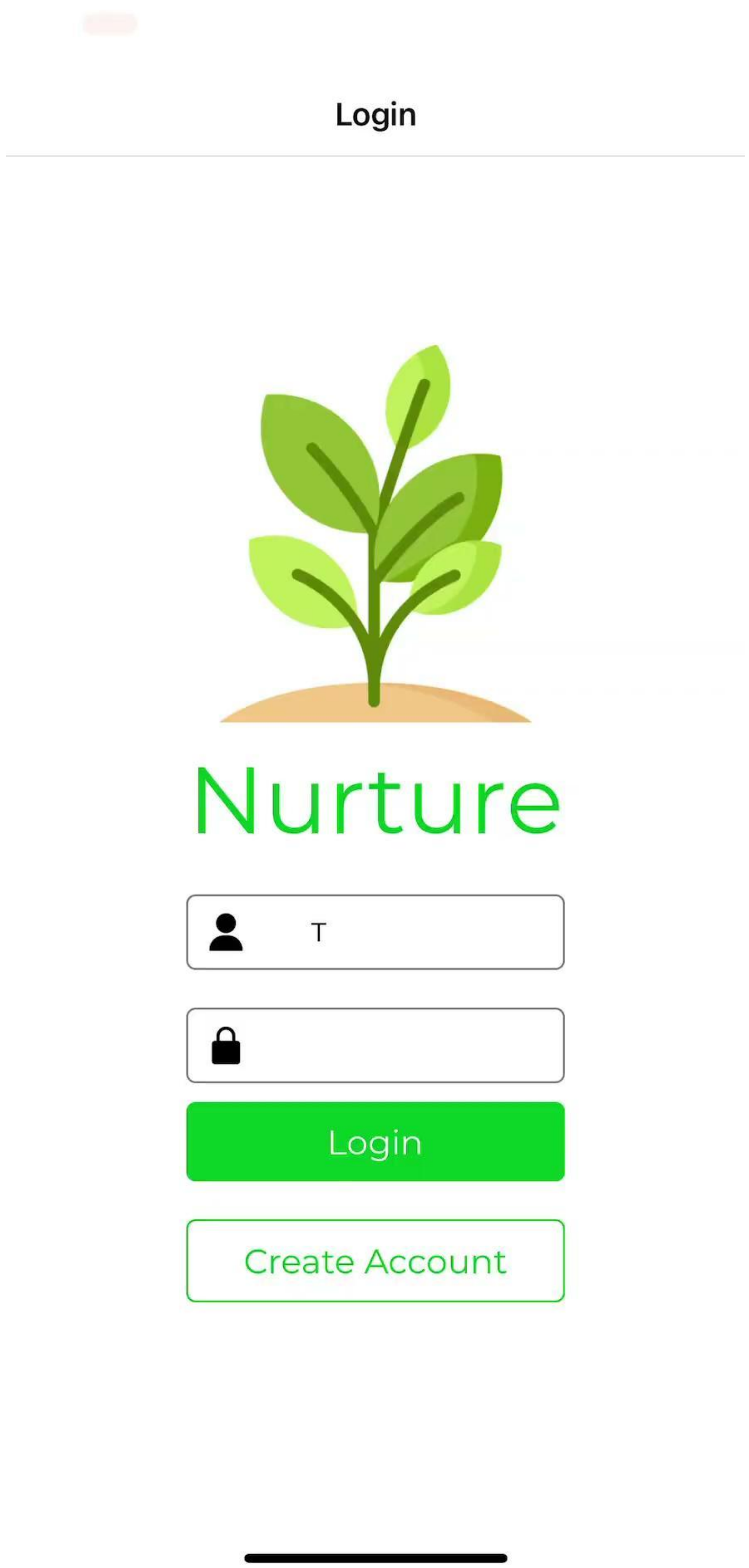
Has 3 settings saved for different sensors

Plant #2

Plant #3

Viewing plant Viewing pl

Mobile App Demo



System Analysis

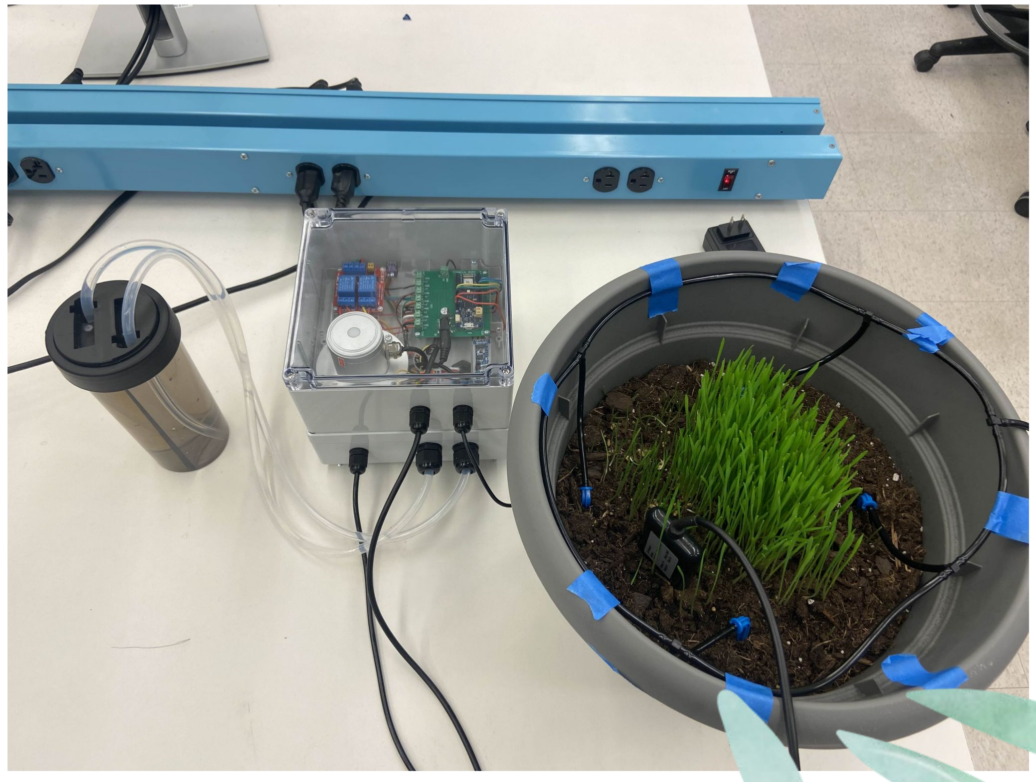
Risks and Mitigation

Risks:

- Liquids interfere with electronics
- Selected sensors do not integrate
- Database data is lost

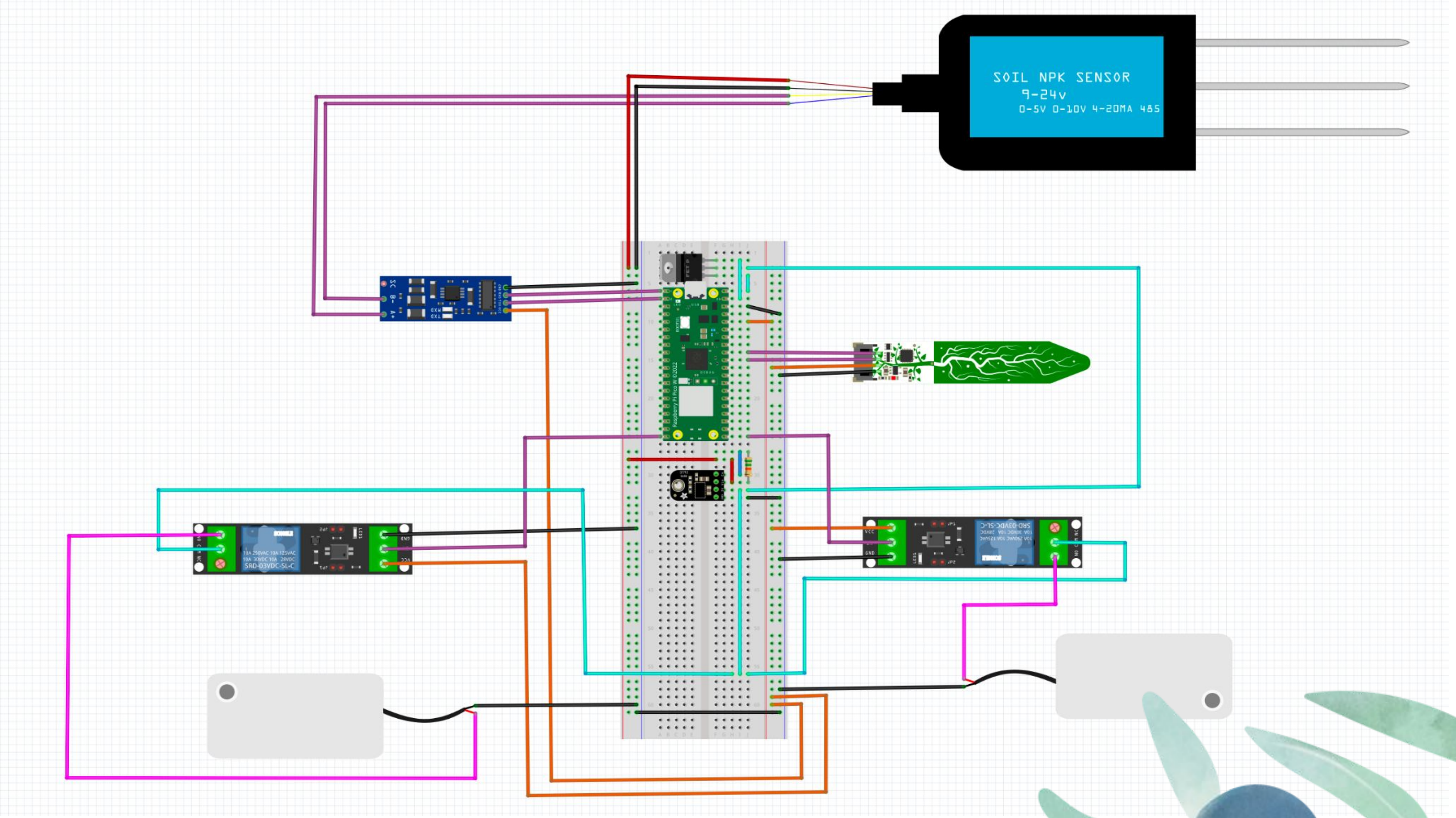
Mitigation:

- Waterproof enclosure selection
  - Isolated pump and reservoir system
- Thorough sensor research
  - Select well-documented components
  - Compare voltages, protocols, etc.
- Perform regular data backups



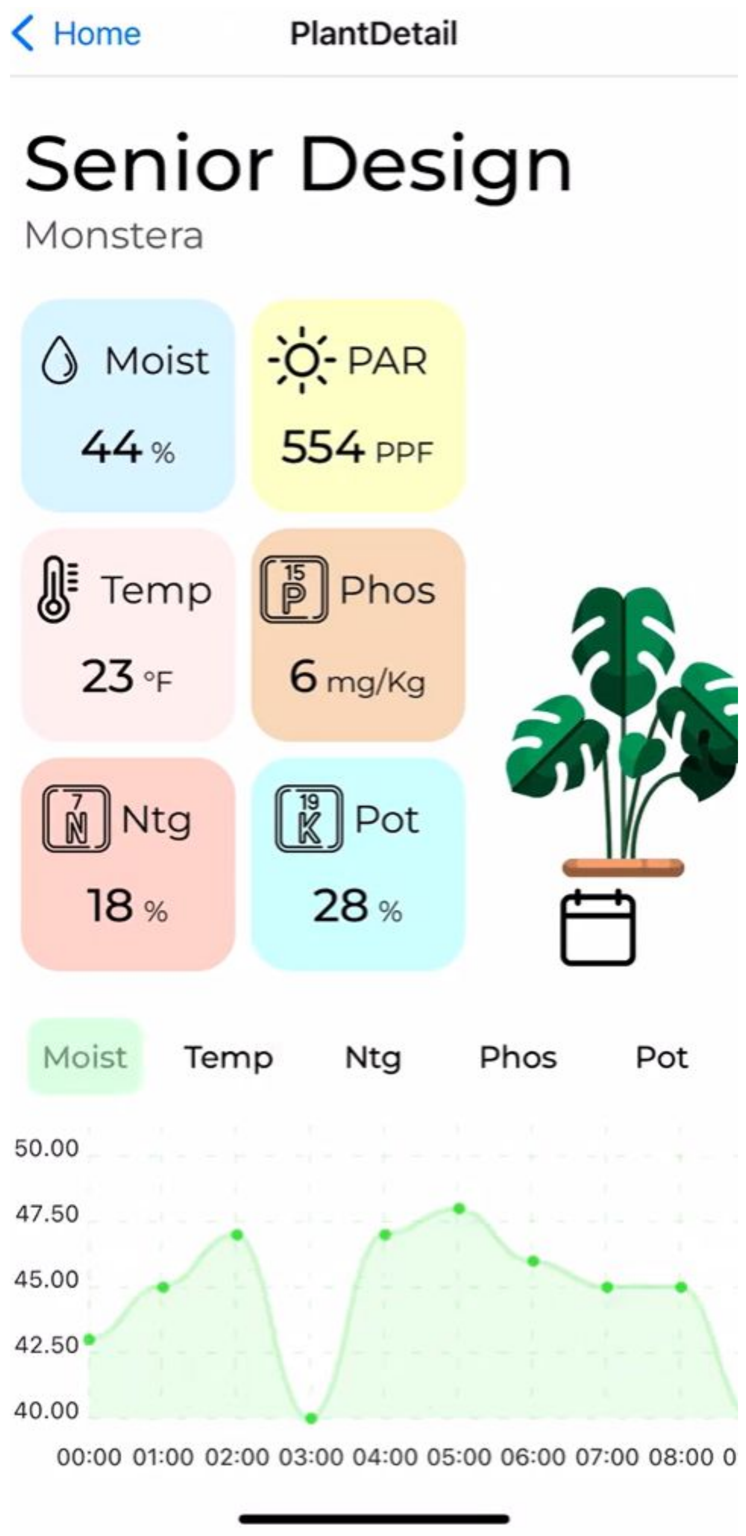
Hardware Challenges

- **Initial Iteration:**
  - Pi Pico: $6
  - Limited Soil Sensor
- **Final Iteration:**
  - Arduino MKR Wi-Fi 1010
  - Enclosure
  - Pump + Valve System
  - Removed Redundant NPK Sensor
  - Custom Power Board



Software Challenges

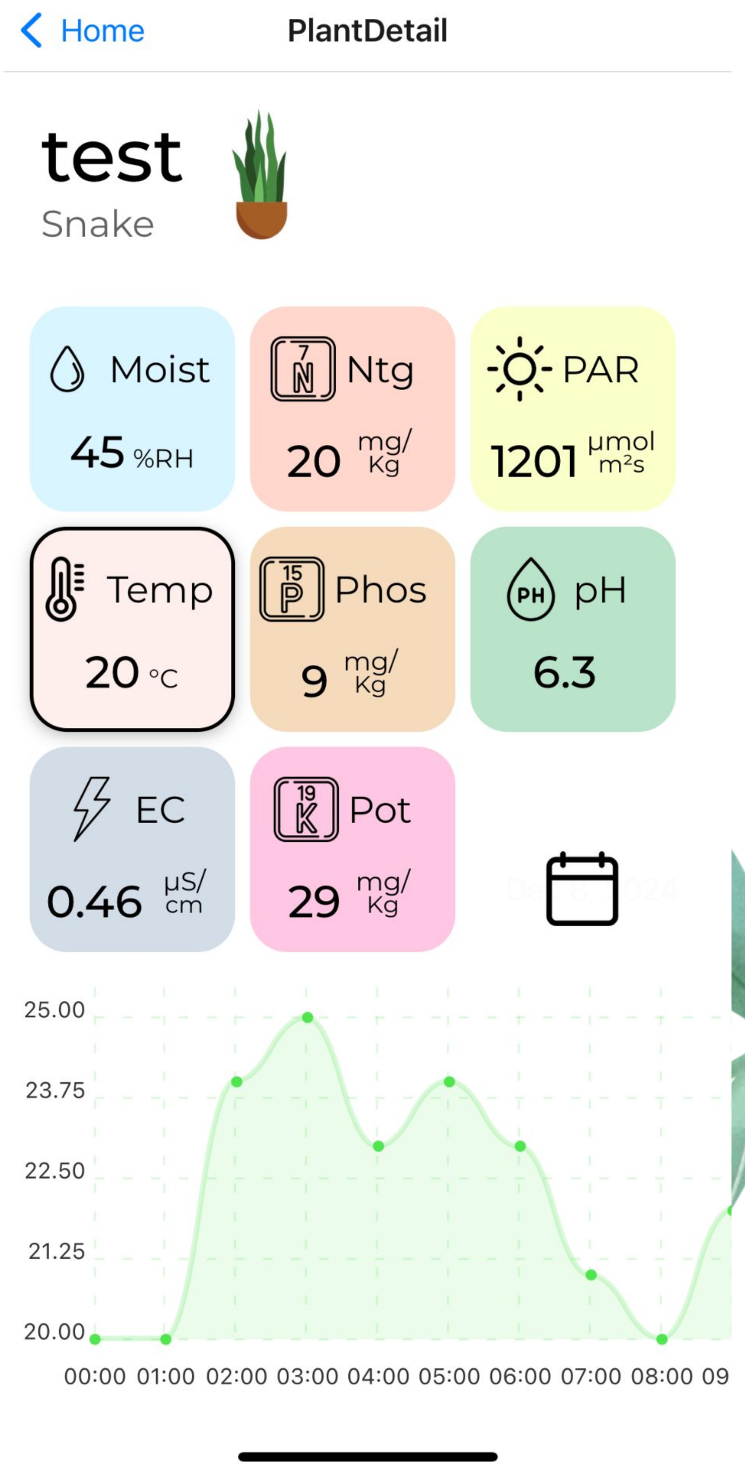
- Keeping the database and mobile app synced
- Backend data validation
- Boxes displaying values for sensor data act as buttons for displaying graphical data for that sensor.
- Updating the plant image on correctly based on the user plant.



< Home PlantDetail

Senior Design
Monstera

Moist 44 %
PAR 554 PPF
Temp 23 °F
Phos 6 mg/Kg
Ntg 18 %
Pot 28 %



< Home PlantDetail

test
Snake

Moist 45 %RH
Ntg 20 mg/Kg
PAR 1201 μmol/m²s
Temp 20 °C
Phos 9 mg/Kg
pH 6.3
EC 0.46 μS/cm
Pot 29 mg/Kg

Testing

Testing Procedure

Approach: Isolated unit testing with gradual system integration.

RS485 (Modbus) Sensors:

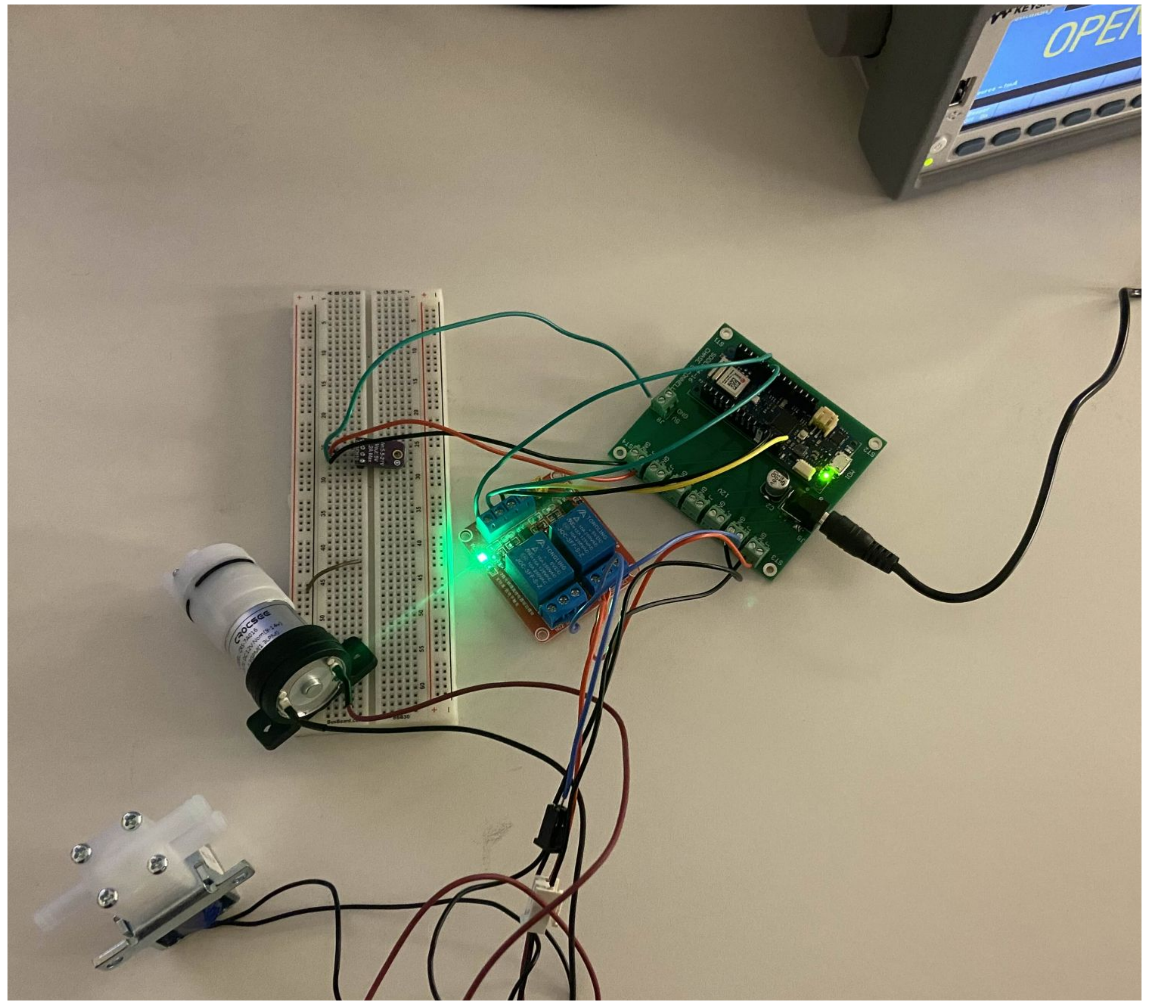
- Responsive TTL Converter
- Arduino Serial Monitor to check raw data.

PCB:

- Multimeter - checking voltages, peripheral functionality.

Mobile App Backend:

- Postman “post” and “put” commands to update database.



Test Results + Conclusions

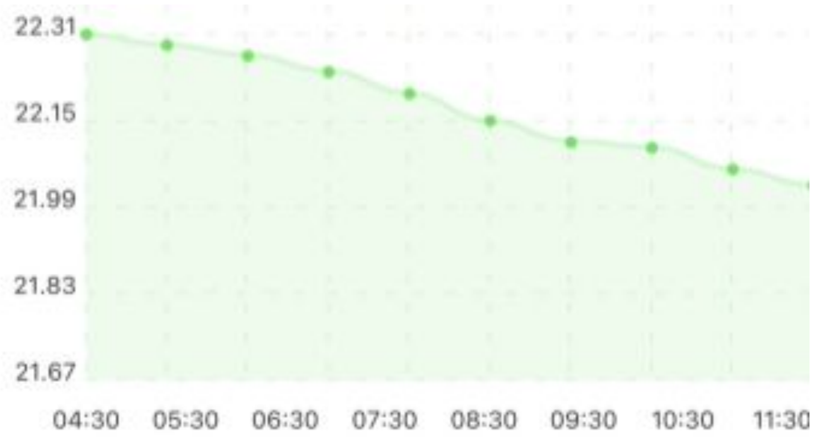
Results:

- 3+ Days of NPK and PAR sensor data
- Working liquid threshold system
- Complete data transmission pipeline

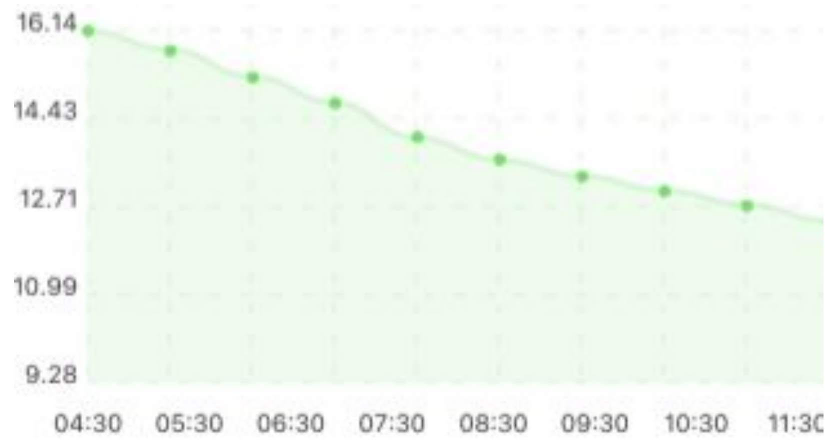
Conclusions:

- PAR Sensor: Required minimum, lower than expected.
- Linear decrease of NPK nutrient availability as soil moisture decreases.

Soil Moisture (RH):



Nitrogen (mg/kg):



Future Considerations

Software:

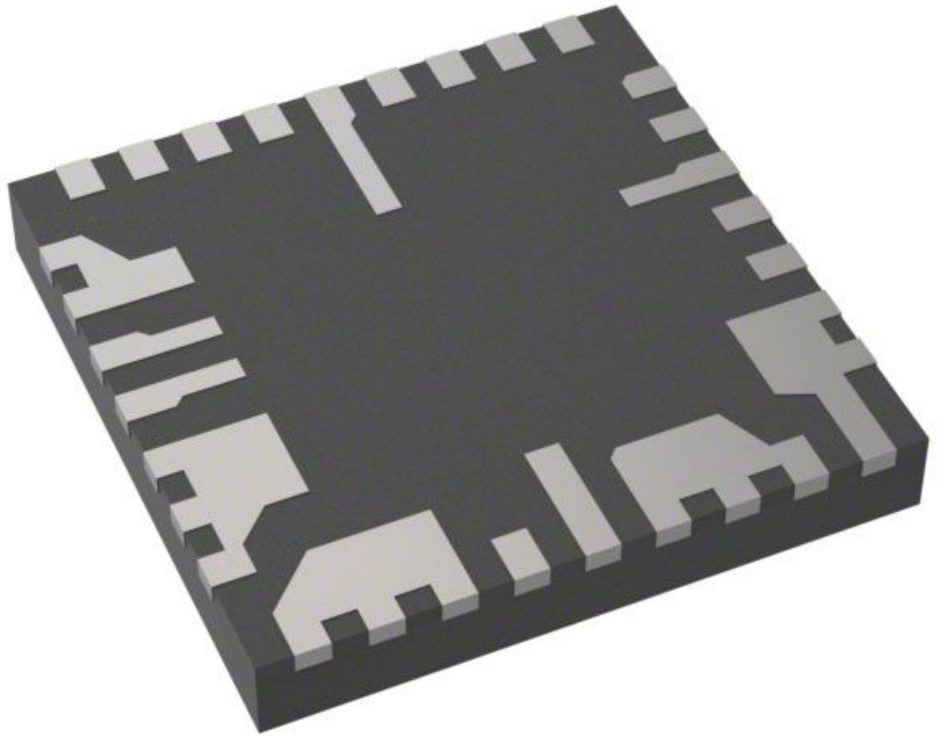
- Compare NPK threshold values only when in a set moisture range (Ex: 40-50% RH).

Hardware:

- Battery-Powered: Outdoor Usage
- Longer, Outdoor Capable, AC Adapter Cable

Integration:

- Direct server to Arduino communication
  - Threshold limits: NPK, Moisture
  - Manual Watering + Fertilization



12V Battery Regulator IC in Consideration

Questions?