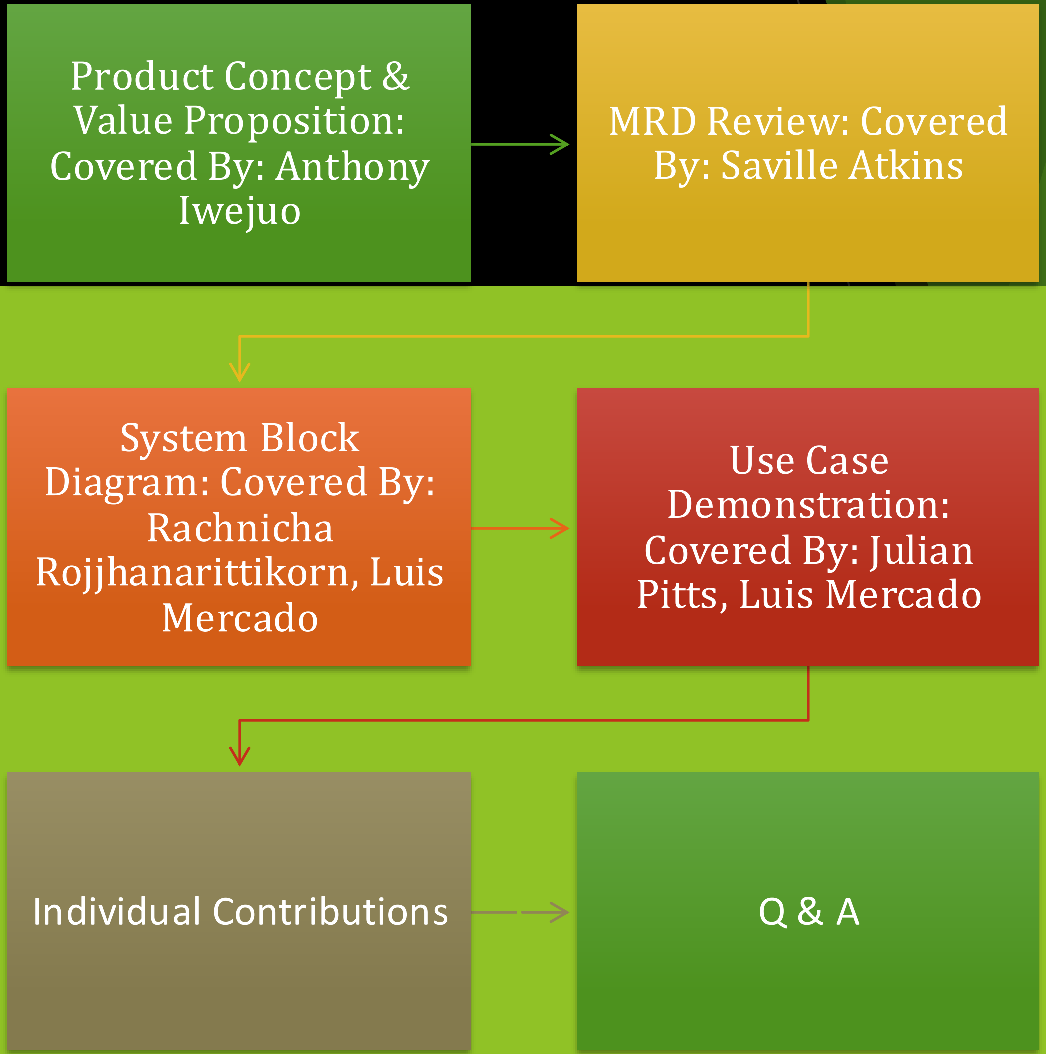
LeafIT

SMART GARDENING TECHNOLOGY

Agenda



Product Concept & Value Proposition: Covered By: Anthony Iwejuo

MRD Review: Covered By: Saville Atkins

System Block Diagram: Covered By: Rachnicha Rojjhanarittikorn, Luis Mercado

Use Case Demonstration: Covered By: Julian Pitts, Luis Mercado

Individual Contributions

Q & A

What is LeaFIT?

What is LeaFIT?

- 🌱 LeaFit is an AI-powered, IoT-based gardening assistant designed to automate plant care for home growers, hobbyists, and small-scale farmers.
- 💡 It tackles the challenges of inconsistent plant care due to busy schedules, limited experience, or remote locations.
- 🎯 With autonomous monitoring, solar power, irrigation, LoRa connectivity, and AI-driven plant health detection, LeaFit empowers users to grow healthier plants with less effort—anytime, anywhere.

Let's talk about our achievement!

Met Requirements

✅

Environmental Temperature Monitoring: Continuously measures ambient air temperature for optimal plant growth. (Must Have)

Soil Moisture Detection: Detects soil moisture levels to prevent overwatering or underwatering. (Must Have)

Automated Irrigation Control: Automatically waters plants based on real-time soil moisture levels. (Must Have)

AI Camera: Captures images of plants for monitoring AI functions. Slow to send video so we opted for images only for classification. (Must Have)

Plant Type Detection: Uses AI to identify and classify plant types. (Must Have)

Plant Disease Detection: Uses AI to identify diseases in the plants(Must Have)

Met Requirements

✓

Live Data Connectivity:	Sends real-time data to a mobile app for monitoring conditions and device status. (Must Have)
Manual Override Functionality:	Allows manual control of irrigation and functions via app. (Must Have)
Device Power & Selective Feature Monitoring:	Allows users monitor/select active features to conserve power. Operates on solar power(Must Have)
Solar Charging Capability:	Runs entirely on battery power and has solar charging capabilities (Must Have)
Customizable Watering Cycles:	Allows users to set specific watering schedules. (Must Have)
Humidity & Light Level Sensors:	Monitor air humidity and light levels for plant health. (Must Have)

Requirements Not Met ❌

Fertilization Alerts:

- Notifies users when it is time to fertilize based on predefined schedules or sensor data. (Nice to Have)

Weather Forecast Integration:

- Adjusts watering schedules based on upcoming weather conditions.(Nice to Have)

Multiple Plant Profiles:

- Allows users to set up profiles for different plants with specific care requirements.(Nice to Have)

Voice Assistant Compatibility:

- Integrates with voice assistants like Alexa or Google Assistant.(Nice to Have)

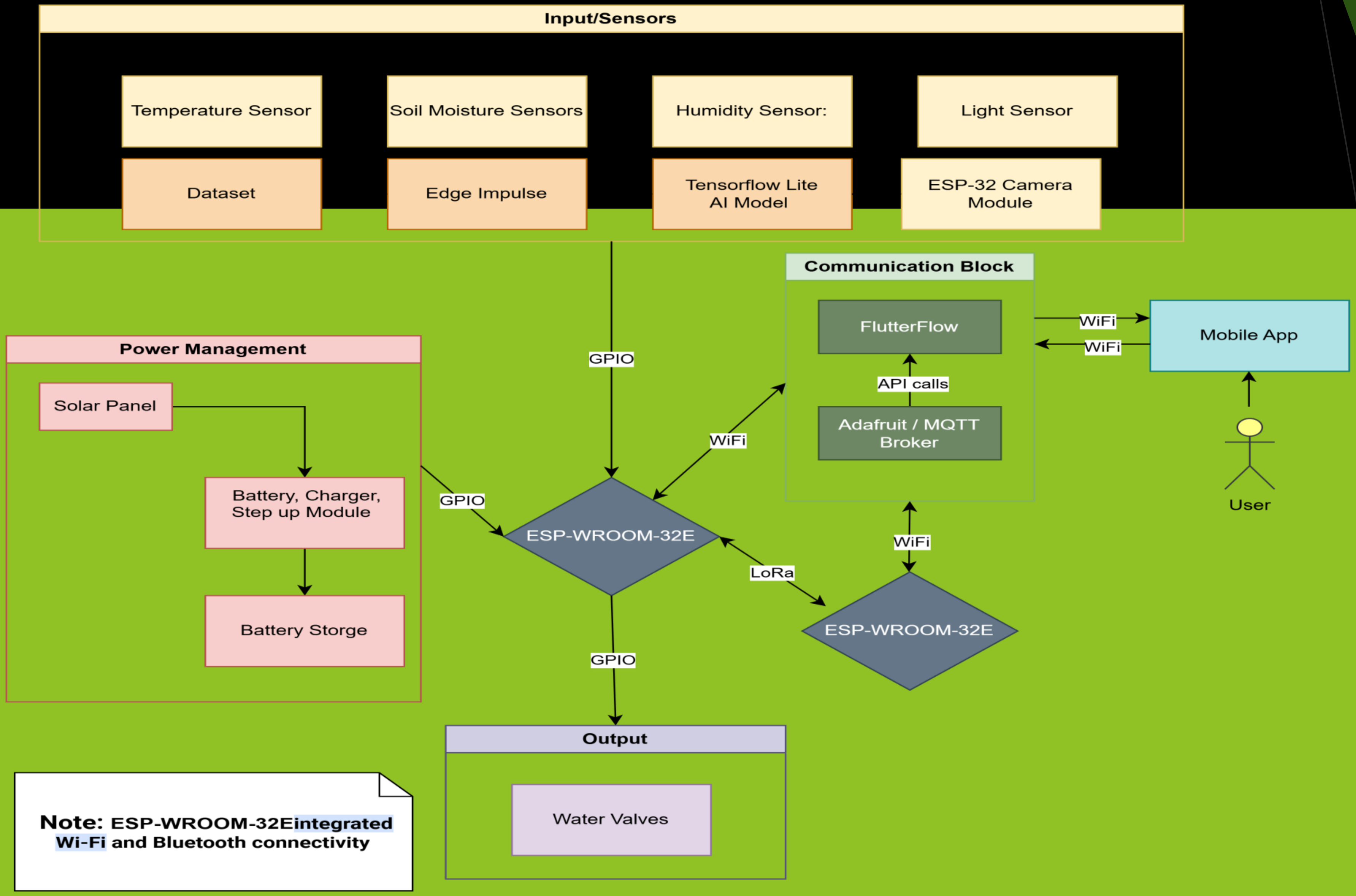
Data Logging and Analytics:

- Logs historical data to show trends and support plant care decisions.(Nice to Have)

Community Sharing Platform:

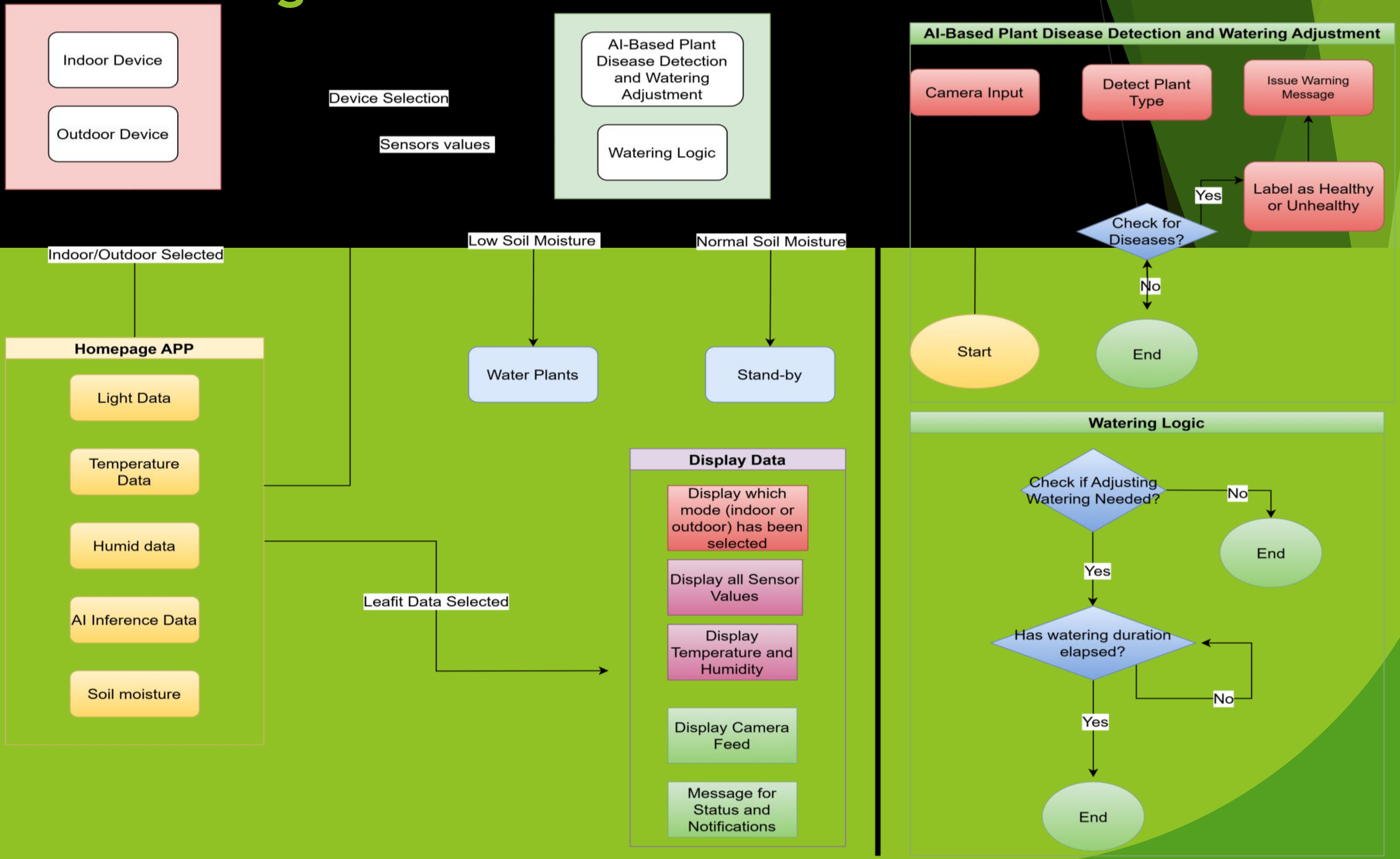
- Connects users to a community for support and sharing tips.(Nice to Have)

System Block Diagram

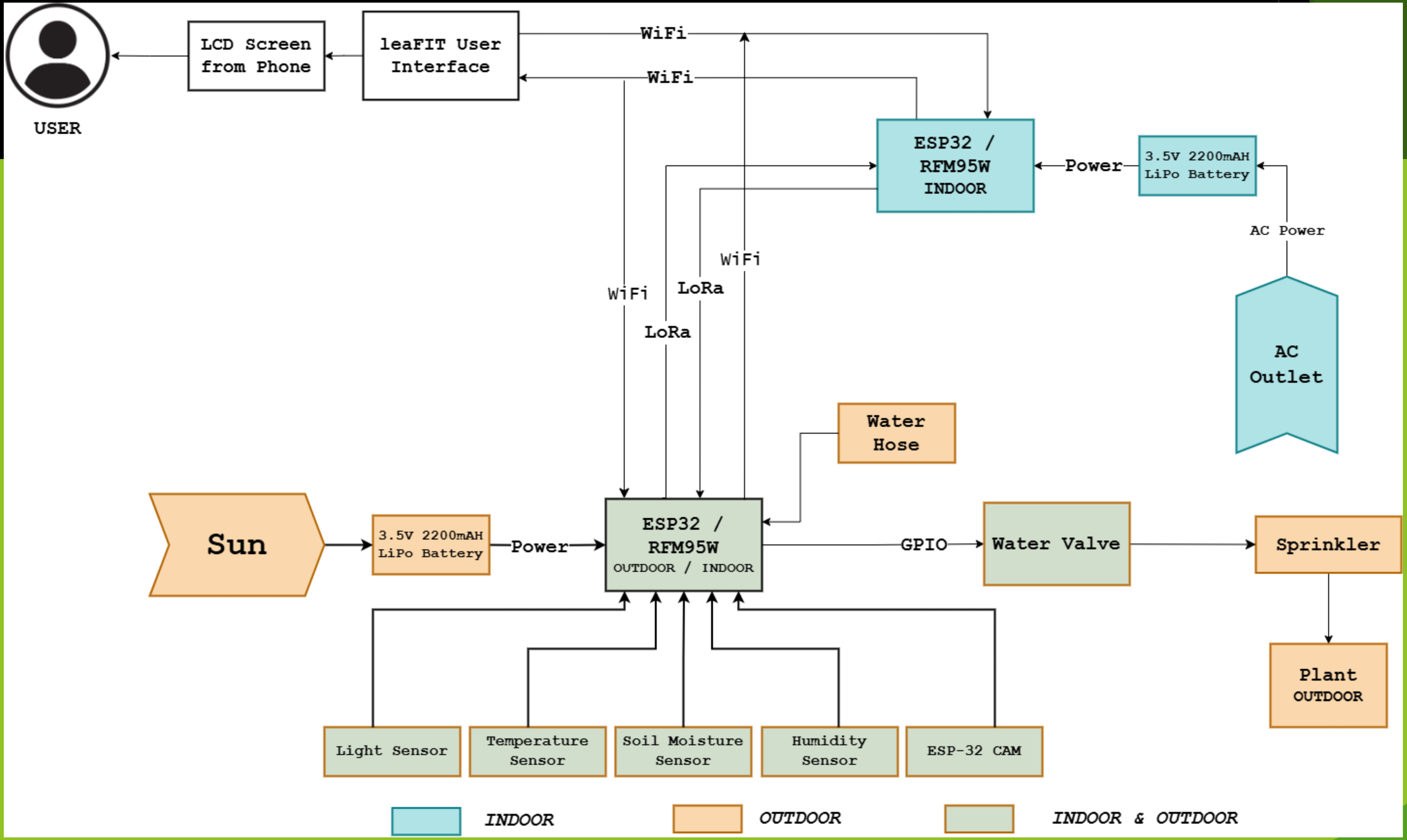


Note: **ESP-WROOM-32Eintegrated Wi-Fi and Bluetooth connectivity**

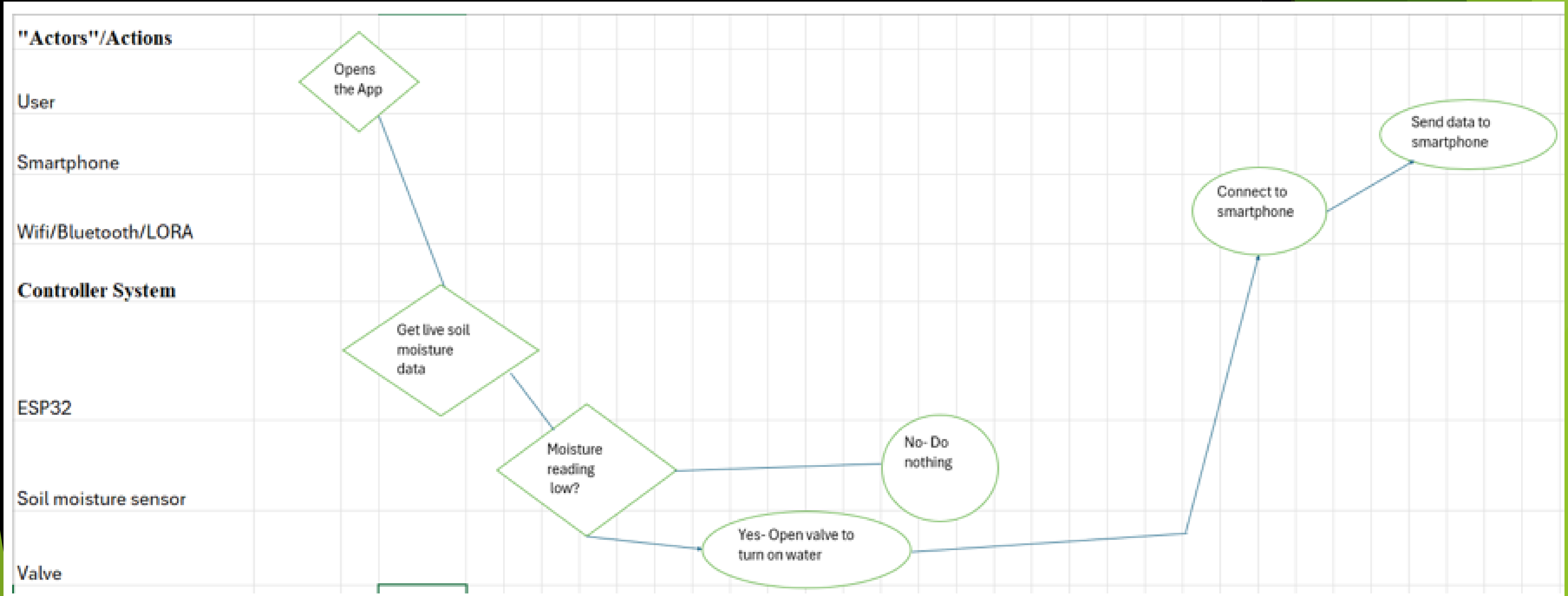
Software Diagram



Hardware Diagram

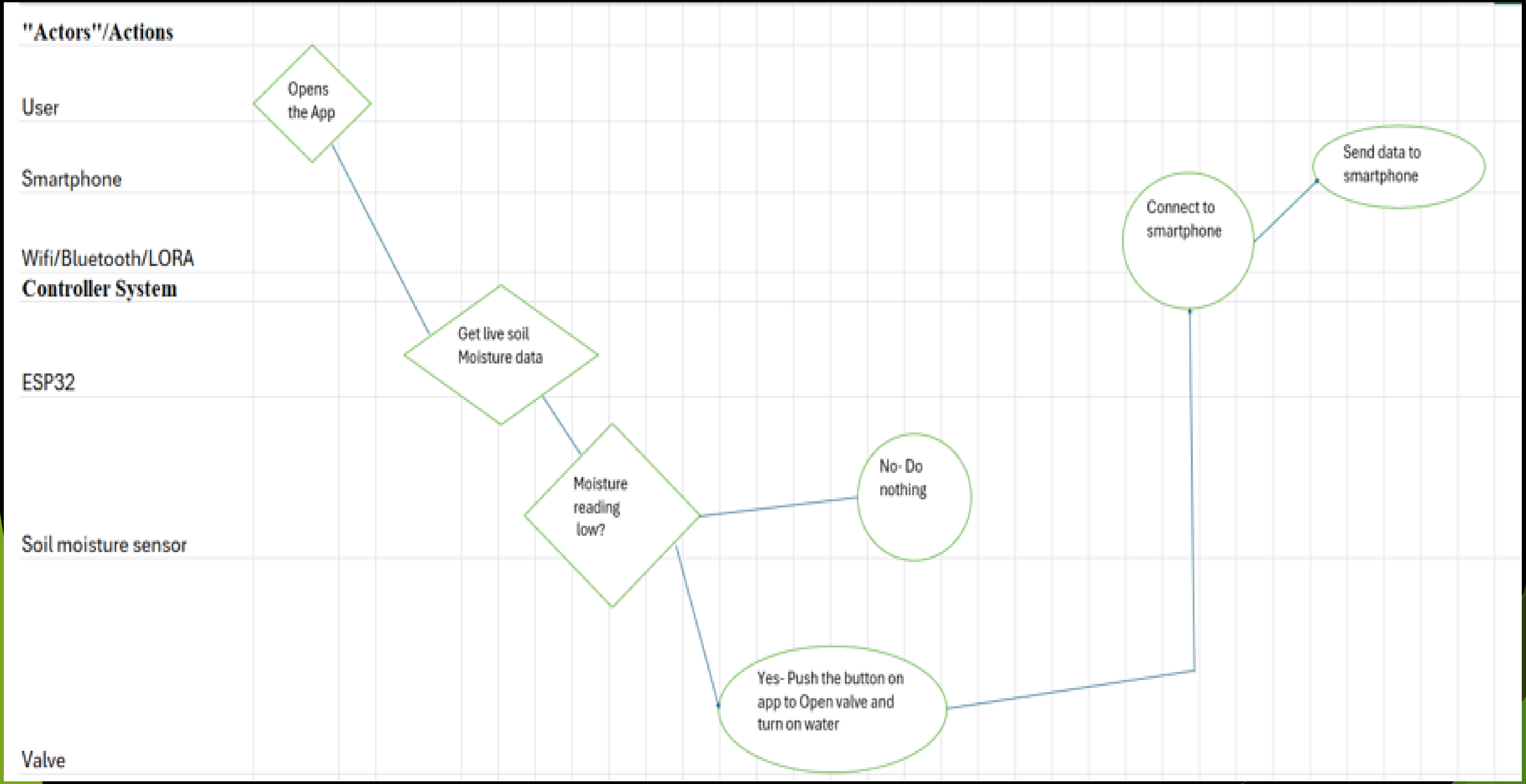


Use Cases 1: Auto Irrigation mode W / DEMO



The first Use-case Scenario is used to identify the live data of the soil moisture sensor. Once the soil moisture sensor reads that it is low, it will automatically open the valve and turn. on the water

Use Cases 2: Manual Irrigation mode W / DEMO



"Actors"/Actions

User

Smartphone

Wifi/Bluetooth/LORA

Controller System

ESP32

Soil moisture sensor

Valve

Opens the App

Get live soil Moisture data

Moisture reading low?

No- Do nothing

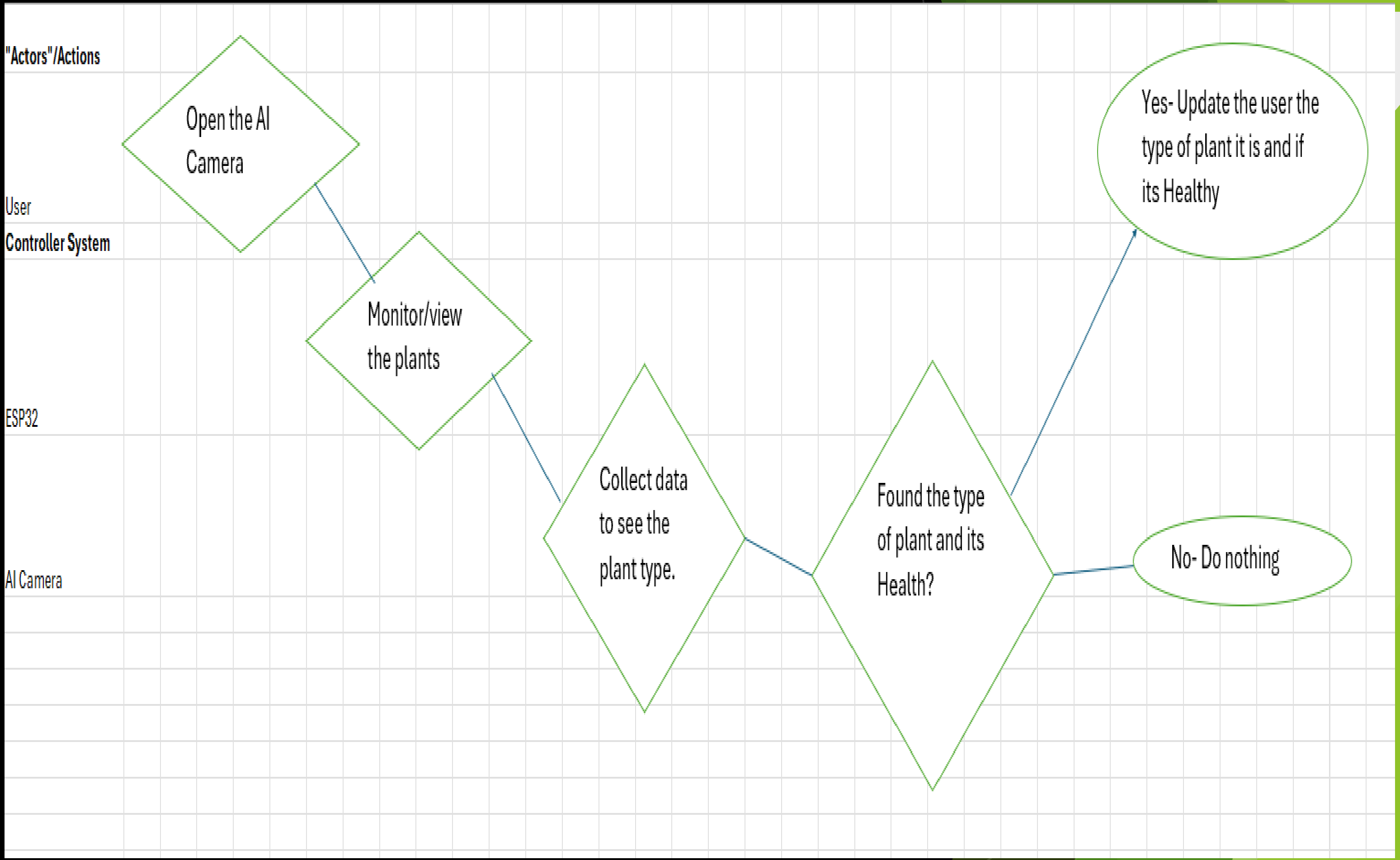
Yes- Push the button on app to Open valve and turn on water

Connect to smartphone

Send data to smartphone

Use Cases

- The live update automatic Use-case Scenario is used to identify the live data of the soil moisture sensor. Once the soil moisture sensor reads that it is low, it will automatically open the valve and turn on the water
- The live update manual use case reads the soil moisture as well but once it is low the user can turn on the water.
- The AI Camera use case reads the plants and displays the type of plant and its health.

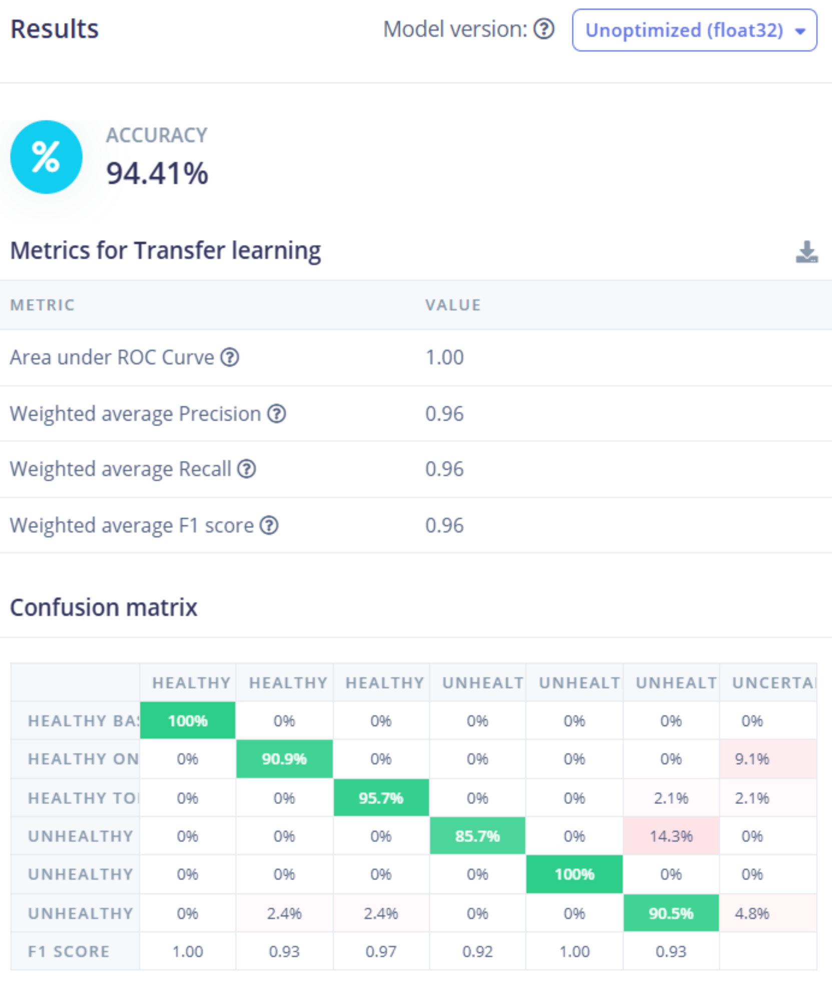

"Actors"/Actions
User
Controller System
ESP32
AI Camera
Open the AI Camera
Monitor/view the plants
Collect data to see the plant type.
Found the type of plant and its Health?
Yes- Update the user the type of plant it is and if its Healthy
No- Do nothing

Application

- The software we used for the application is FlutterFlow.io
- The user can Monitor the Soil Moisture, Humidity, Temperature, and light.
- The user can use commands to turn on/off the watering set up water presets and check on plant details.

Julian and Rachnicha

AI Modeling: Plant Type and Health



Results

Model version: ⓘ Unoptimized (float32) ▾

ACCURACY

94.41%

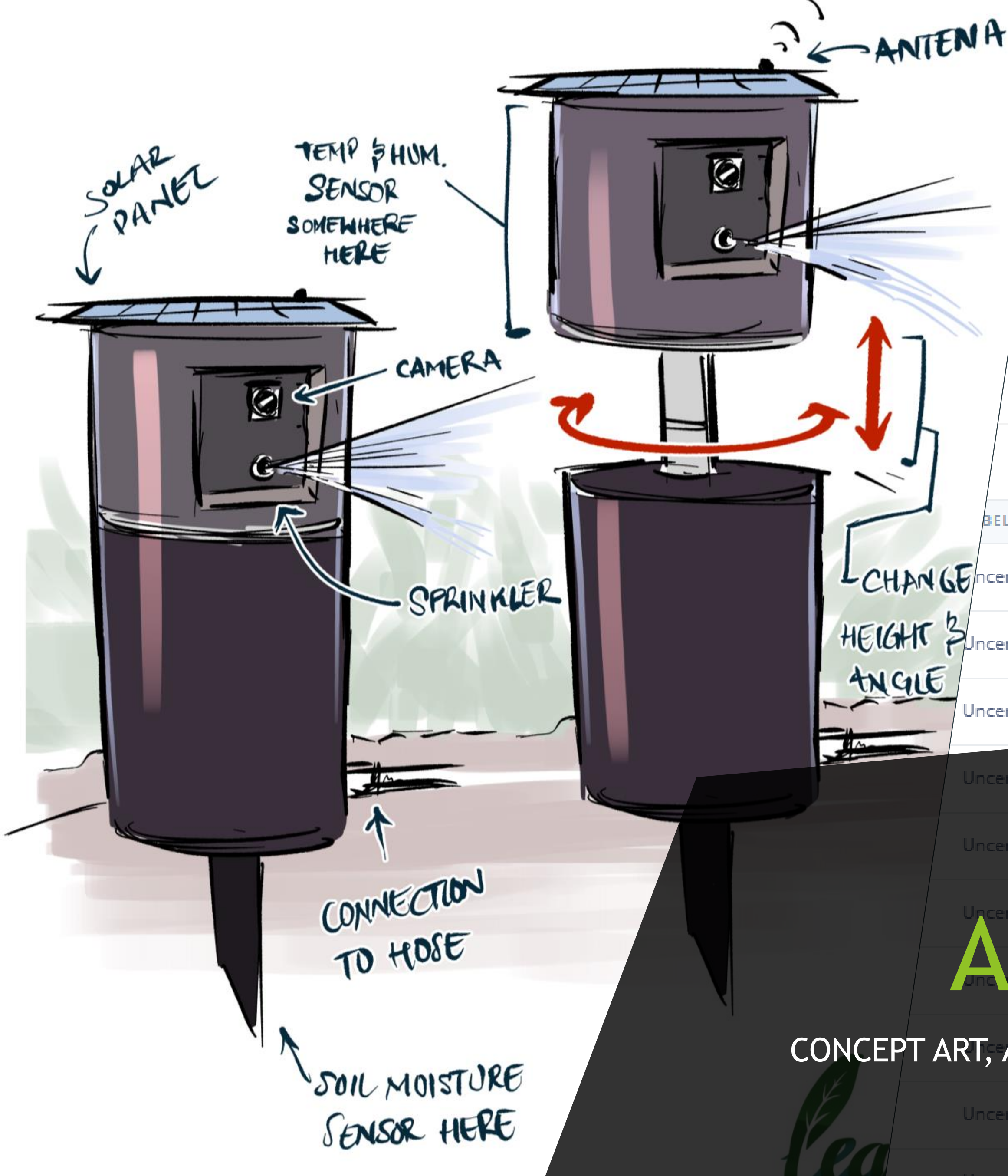
Metrics for Transfer learning

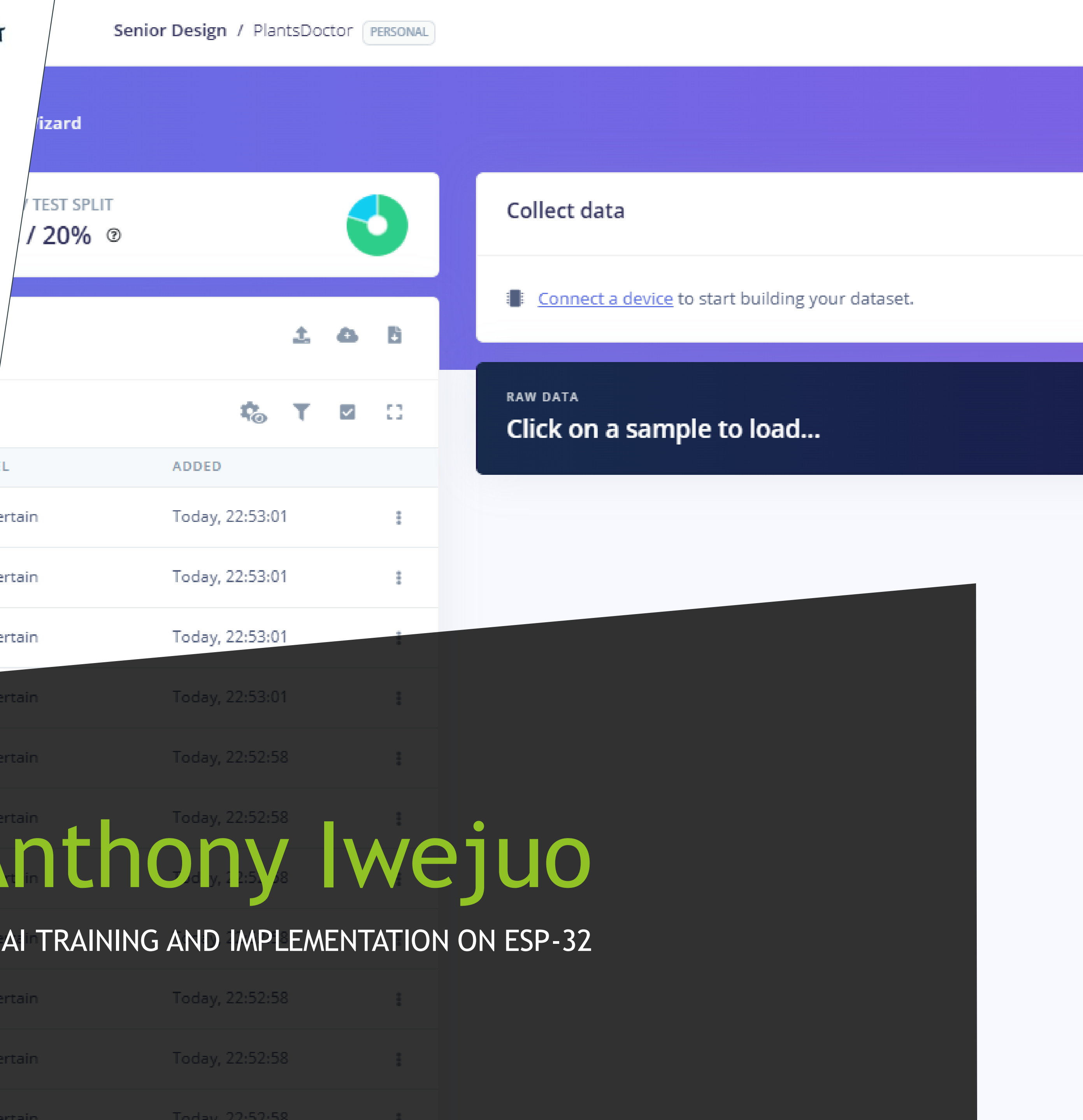
METRIC	VALUE
Area under ROC Curve ⓘ	1.00
Weighted average Precision ⓘ	0.96
Weighted average Recall ⓘ	0.96
Weighted average F1 score ⓘ	0.96

Confusion matrix

	HEALTHY	HEALTHY	HEALTHY	UNHEALT	UNHEALT	UNHEALT	UNCERTAI
HEALTHY BA	100%	0%	0%	0%	0%	0%	0%
HEALTHY ON	0%	90.9%	0%	0%	0%	0%	9.1%
HEALTHY TO	0%	0%	95.7%	0%	0%	2.1%	2.1%
UNHEALTHY	0%	0%	0%	85.7%	0%	14.3%	0%
UNHEALTHY	0%	0%	0%	0%	100%	0%	0%
UNHEALTHY	0%	2.4%	2.4%	0%	0%	90.5%	4.8%
F1 SCORE	1.00	0.93	0.97	0.92	1.00	0.93	

Anthony and Rachnicha




Senior Design / PlantsDoctor PERSONAL

izard

TEST SPLIT
/ 20%

Collect data

Connect a device to start building your dataset.

RAW DATA
Click on a sample to load...

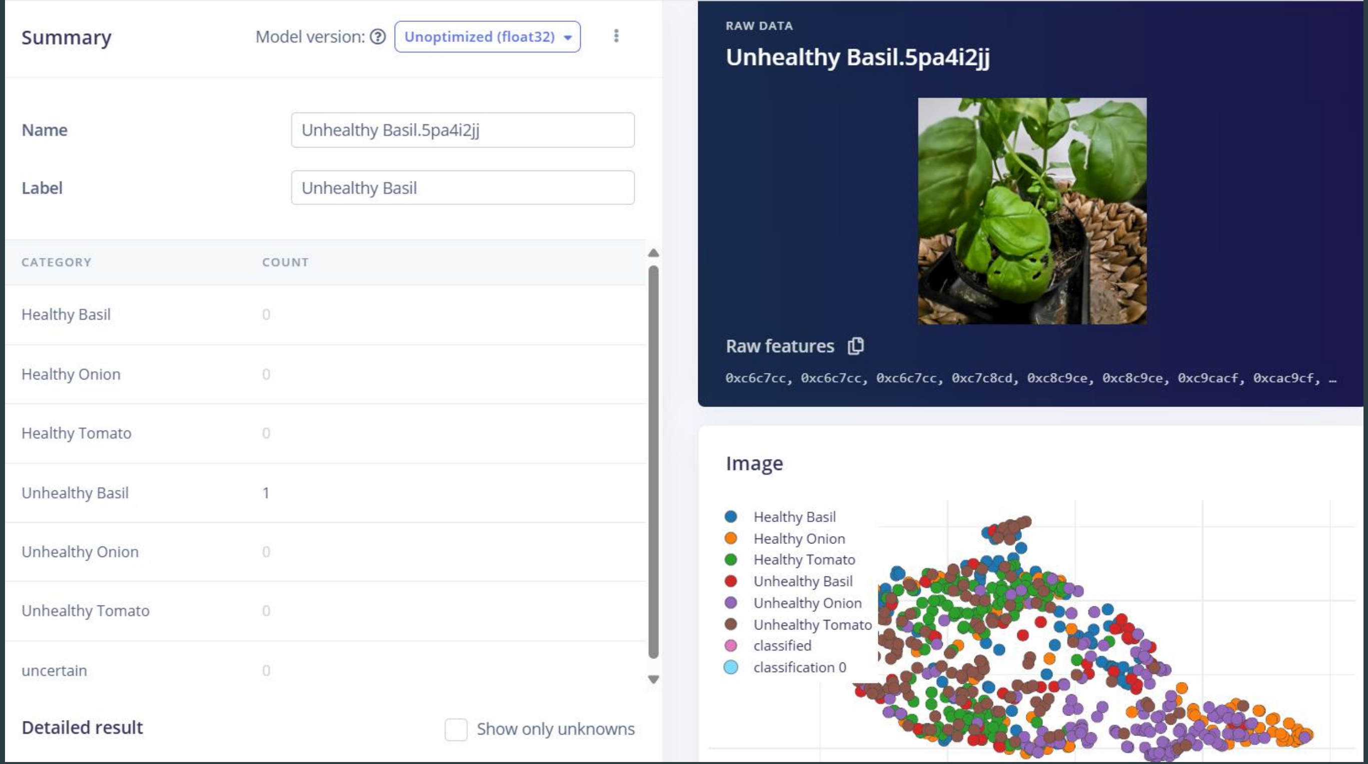
BEL	ADDED
ncertain	Today, 22:53:01
Uncertain	Today, 22:53:01
Uncertain	Today, 22:53:01
Uncertain	Today, 22:53:01
Uncertain	Today, 22:52:58
Uncertain	Today, 22:52:58
Unc[illegible]in	Today, 22:52:58
Uncertain	Today, 22:52:58
Uncertain	Today, 22:52:58
Uncertain	Today, 22:52:58

Anthony Iwejuo

CONCEPT ART, AI TRAINING AND IMPLEMENTATION ON ESP-32

Rachnicha Rojjhanarittikorn

- App Development : Create the UI for user interface
- Firmware/Software: API calls to link with Adafruit
- AI modeling: Collecting Data for training AI and build AI modeling



Summary

Model version: ⓘ Unoptimized (float32) ▾

Name	Unhealthy Basil.5pa4i2jj
Label	Unhealthy Basil

CATEGORY	COUNT
Healthy Basil	0
Healthy Onion	0
Healthy Tomato	0
Unhealthy Basil	1
Unhealthy Onion	0
Unhealthy Tomato	0
uncertain	0

Detailed result

□ Show only unknowns

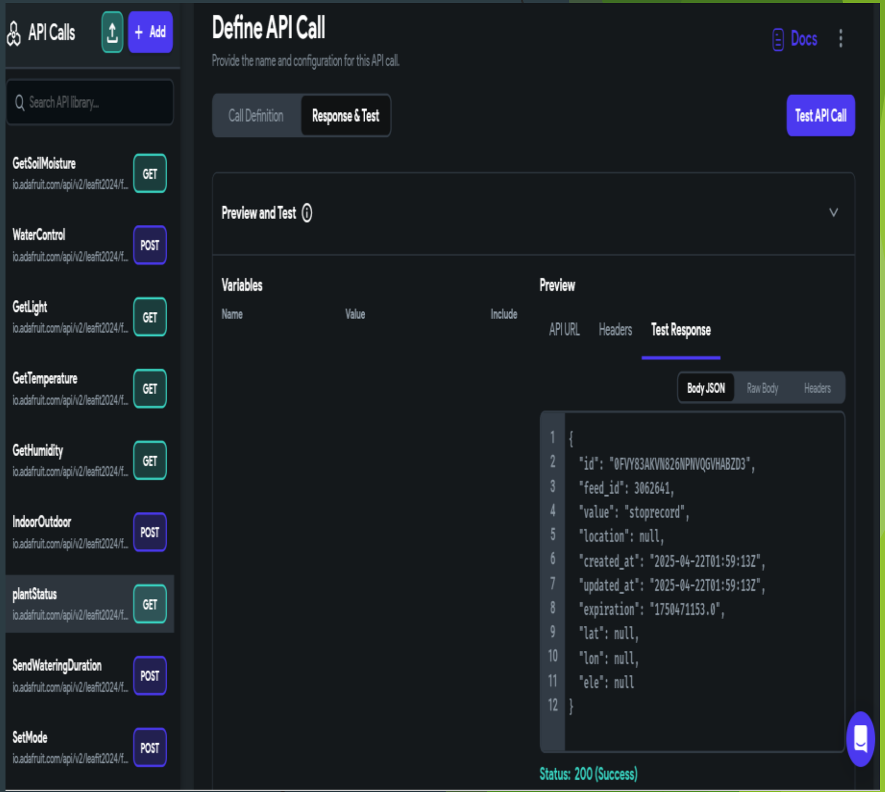
RAW DATA

Unhealthy Basil.5pa4i2jj

Raw features

```
0xc6c7cc, 0xc6c7cc, 0xc6c7cc, 0xc7c8cd, 0xc8c9ce, 0xc8c9ce, 0xc9cacf, 0xcac9cf, …
```

Image



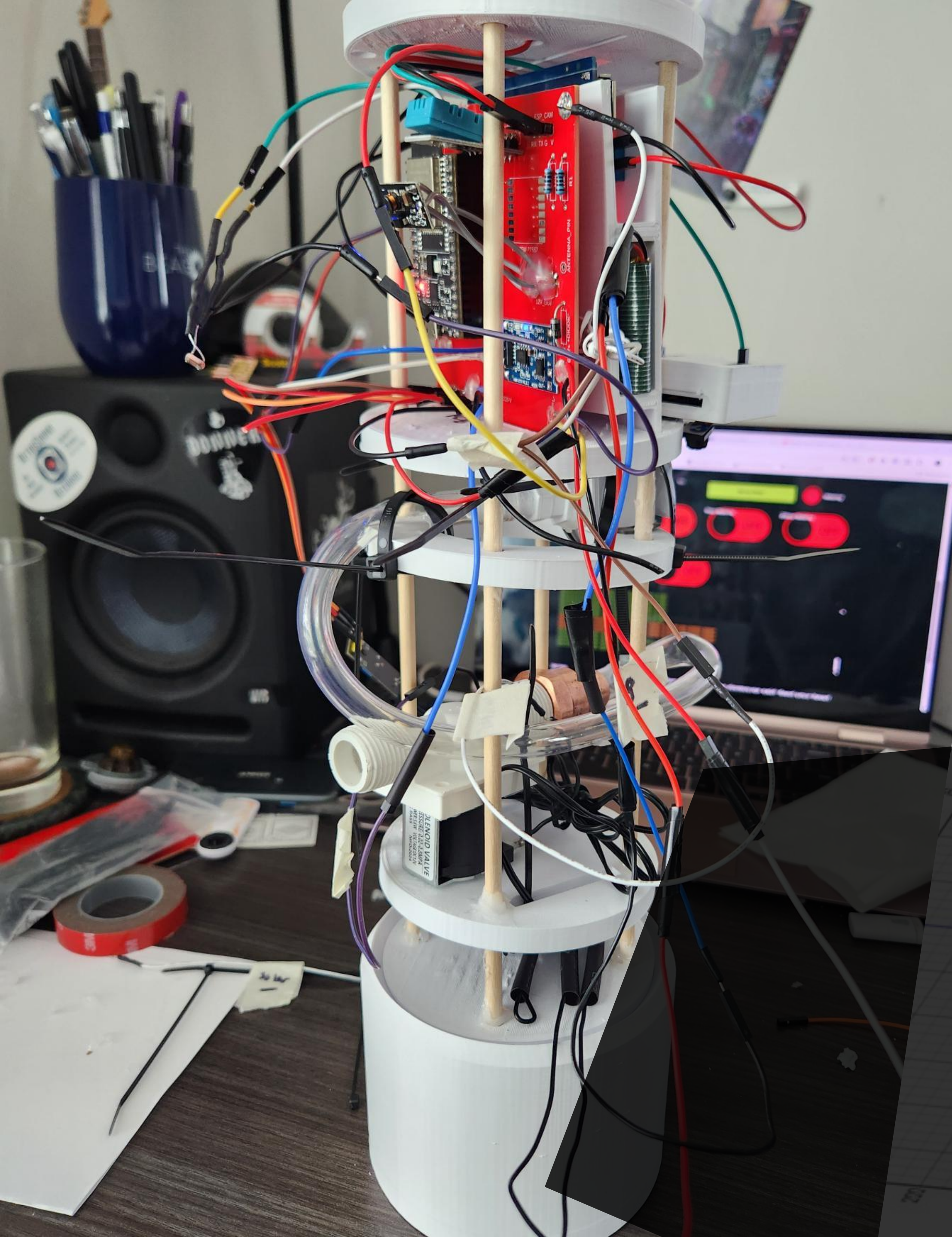
GetSoilMoisture	io.adafruit.com/api/v2/leafit2024/f...	GET
WaterControl	io.adafruit.com/api/v2/leafit2024/f...	POST
GetLight	io.adafruit.com/api/v2/leafit2024/f...	GET
GetTemperature	io.adafruit.com/api/v2/leafit2024/f...	GET
GetHumidity	io.adafruit.com/api/v2/leafit2024/f...	GET
IndoorOutdoor	io.adafruit.com/api/v2/leafit2024/f...	POST
plantStatus	io.adafruit.com/api/v2/leafit2024/f...	GET
SendWateringDuration	io.adafruit.com/api/v2/leafit2024/f...	POST
SetMode	io.adafruit.com/api/v2/leafit2024/f...	POST

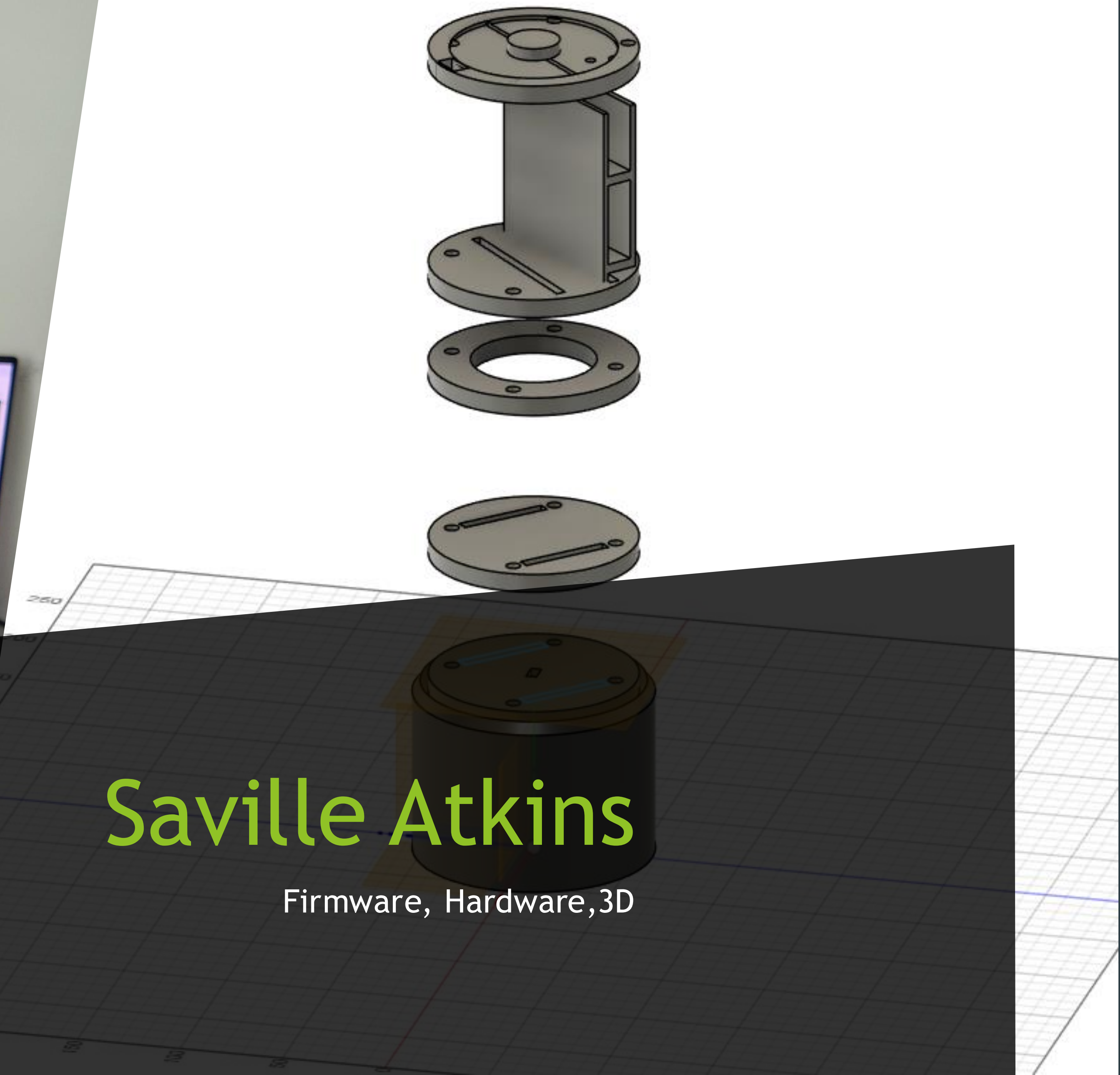
Define API Call

Provide the name and configuration for this API call.

```
{
  "id": "0FVY83AKVN826NPNVQGVHABZD3",
  "feed_id": 3062641,
  "value": "stoprecord",
  "location": null,
  "created_at": "2025-04-22T01:59:13Z",
  "updated_at": "2025-04-22T01:59:13Z",
  "expiration": "1750471153.0",
  "lat": null,
  "lon": null,
  "ele": null
}
```

Status: 200 (Success)



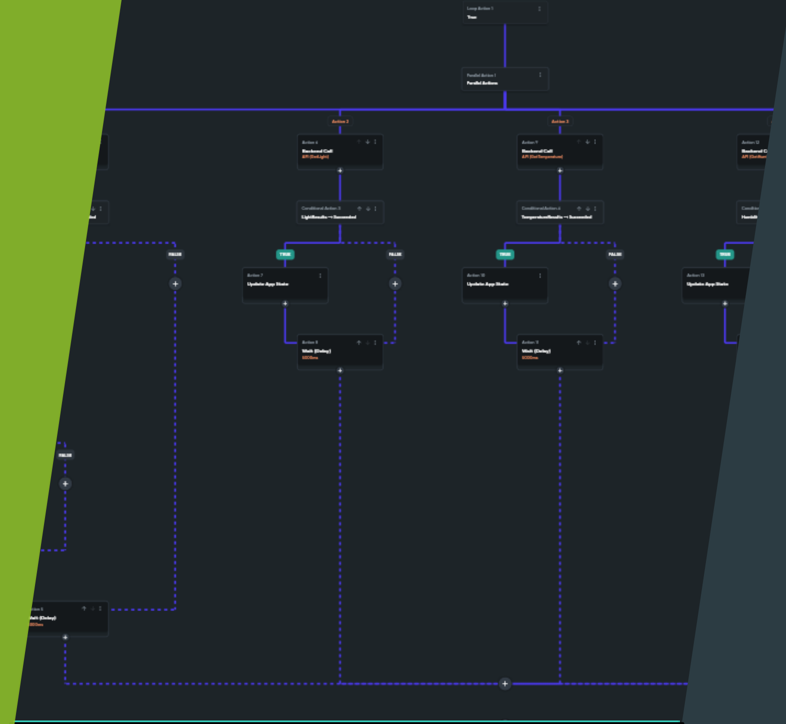


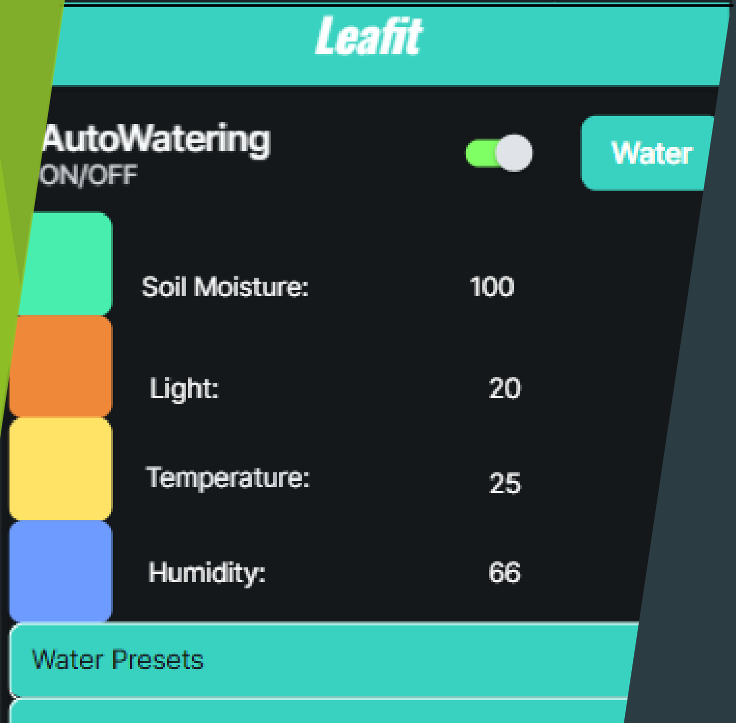
Saville Atkins

Firmware, Hardware,3D

Saville Atkins

- Hardware
  - Assisted in researching hardware requirements and designing the electronics for the system
- Firmware
  - Programmed the Custom and Auto watering modes
- 3D Modeling
  - Designed the 3d models for the inner frame, sprinkler and the covering sleeve





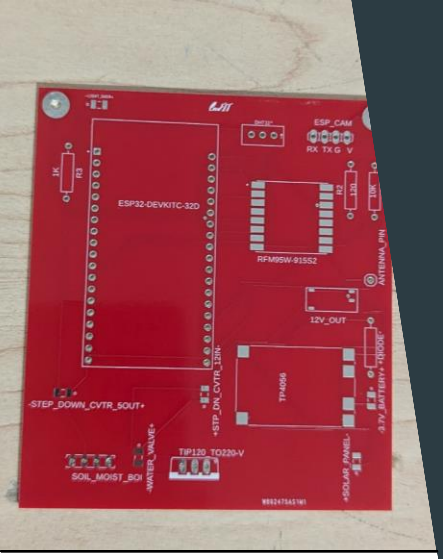
Leafit

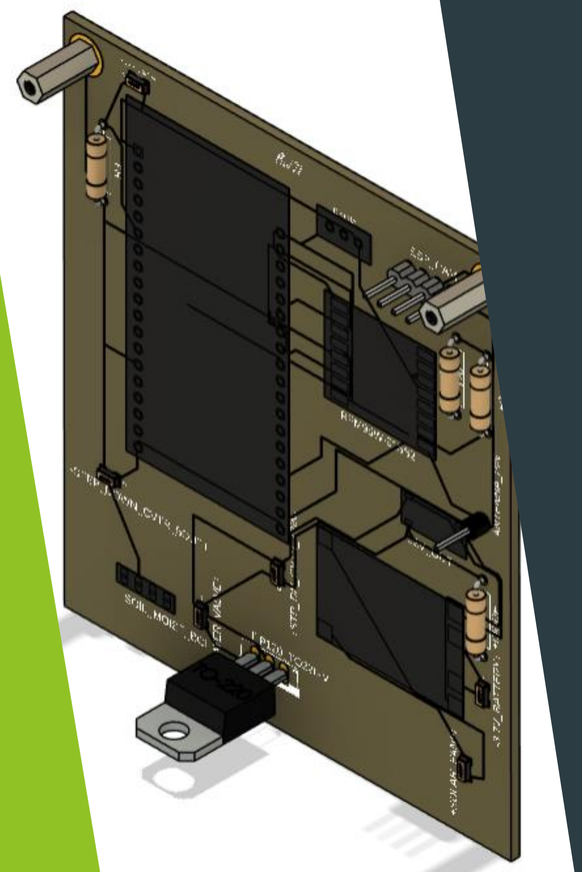
AutoWatering
ON/OFF

Soil Moisture:	100
Light:	20
Temperature:	25
Humidity:	66

Julian Pitts

- Software/Networking:
  - Created API feeds to communicate with flutter.
- App Development:
  - Created API calls in flutter to link with Adafruit
  - Created Actions for each button in flutter to follow commands from Adafruit and get live data as well (Ex: soil moisture)
  - Created App states in flutter to display the data from Adafruit.





Luis Mercado

Hardware

- Assisted in researching hardware components needed
- PCB Design and assembly of board with components
- Wire management
- Demo ecosystem assembly

Software/ Networking

- Networking code portion of ESP programming code
- Assisted with overall programming logic and functionality

Project Management

- Organized times for meetings
- Followed up with team member individual progress

Thank you

Questions?