IOWA STATE UNIVERSITY

Department of Electrical and Computer Engineering

SDMAY20-53: Smart Backpack Sprayer

David Hayes (Communication Engineer)
Madison Kriege (Communication Engineer)
Kevin Davis (Hardware Research Engineer)
Sean Doran (Logistics Engineer)
Donald Laracuente (API Research Engineer)
Shuangquan Li (iOS Engineer)

Client: IntelliSpray

Advisor: Dr. Daji Qiao

http://sdmay20-53.sd.ece.iastate.edu/

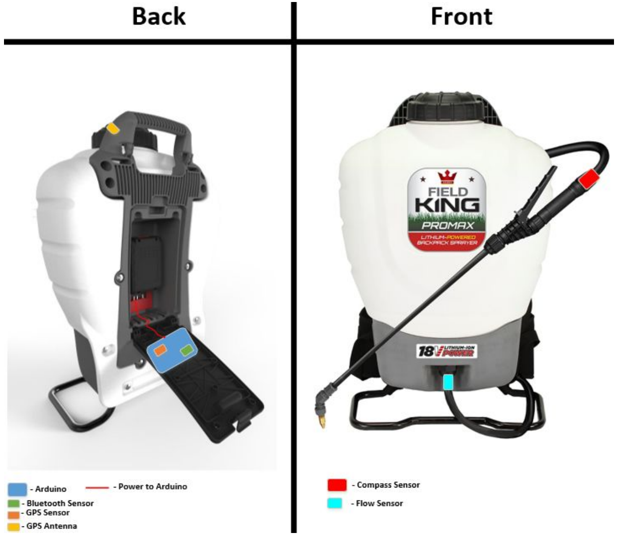
Problem Statement

- Problem:
  - Tracking when and where a user has sprayed is hard
  - User could forget where they have sprayed and spray areas twice or never
- Solution:
  - Collect data from a backpack sprayer on a phone
  - Store the data for the user to reference to ensure the best coverage

David

Functional Requirements

- The hardware shall
  - Use a GPS sensor with accuracy to 3 meters
  - Have a battery life at least 3 hours
  - Be mountable inside backpack sprayer
  - Package data in JSON format
  - Be able to send data using Bluetooth LE
- Data shall be collected in 1 second intervals
- The app shall
  - Display data in the map with a pin
  - Sync data between cloud and local
  - Display the data to the user
  - Support multiple users (login feature)



Non-Functional Requirements

- The system shall
  - Be water resistant
  - Be operable in temperatures between 0-40C
  - Be under 50 pounds
  - Be wearable on one’s back
- The app shall
  - Only allow authorized access to the data
  - Support large amounts of data transmission

Madison

Technical and Other Constraints

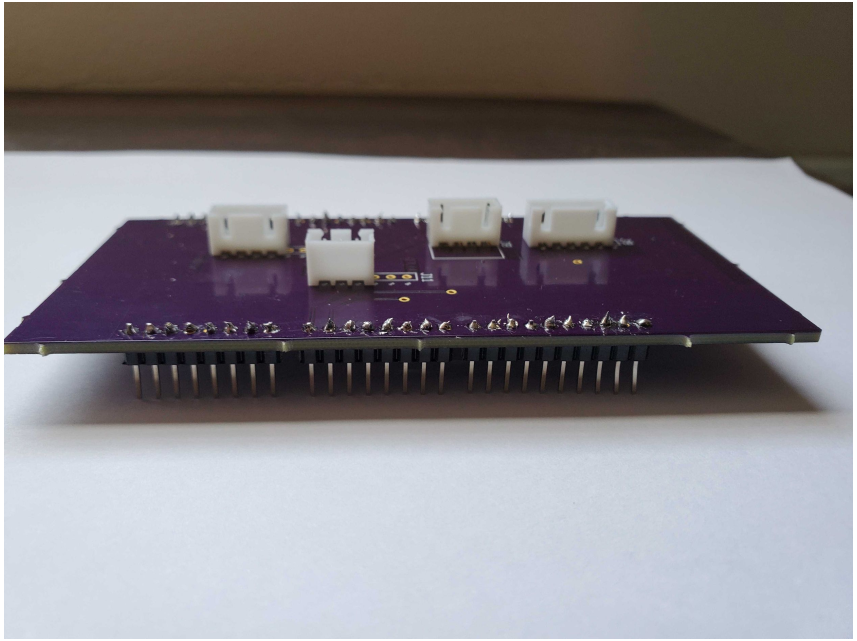
- Mobile platform - iOS
- Data format - JSON
- Close range communication - Bluetooth LE
- Low/off network connectability

Madison

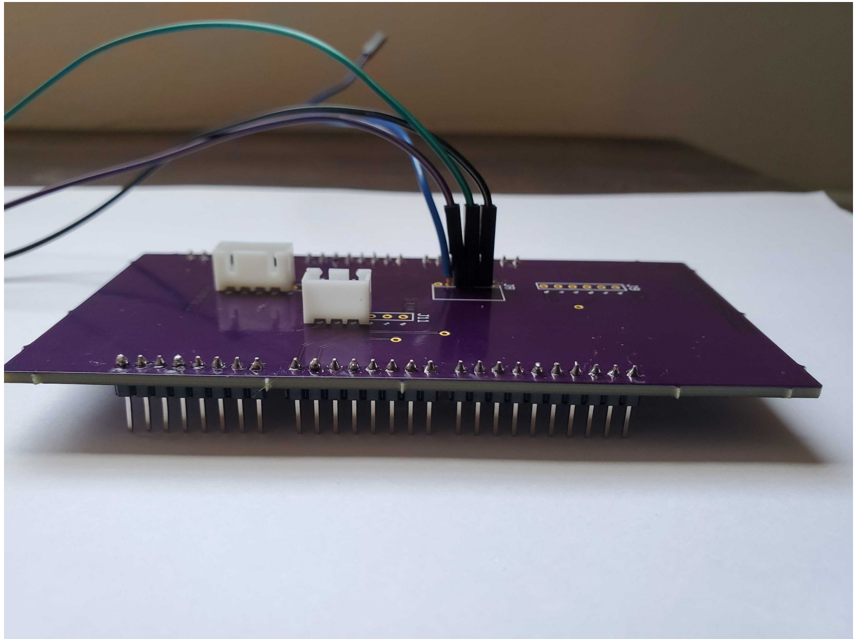
Challenges, Risks, and Mitigation

- Issues
  - Size constraints
  - iOS backwards compatibility with iOS 13
  - Soldering
- Solutions
  - Using a smaller Arduino
  - It is not backwards compatible / Roll Back to 12
  - More practice

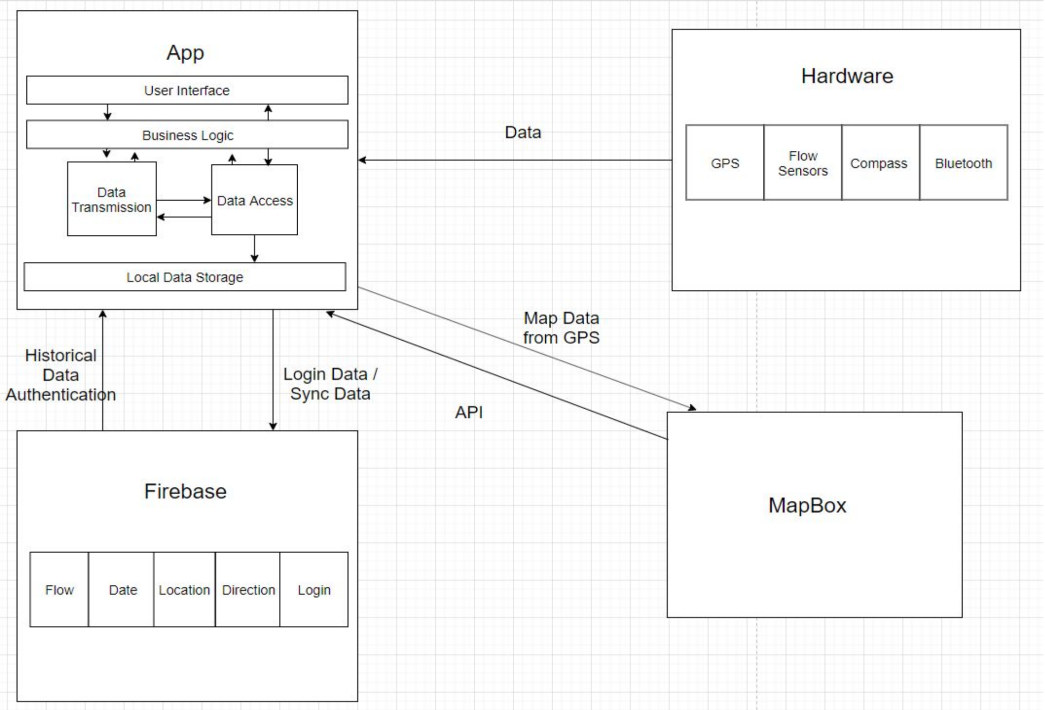
First Attempt



Second Attempt



System Architecture Diagram

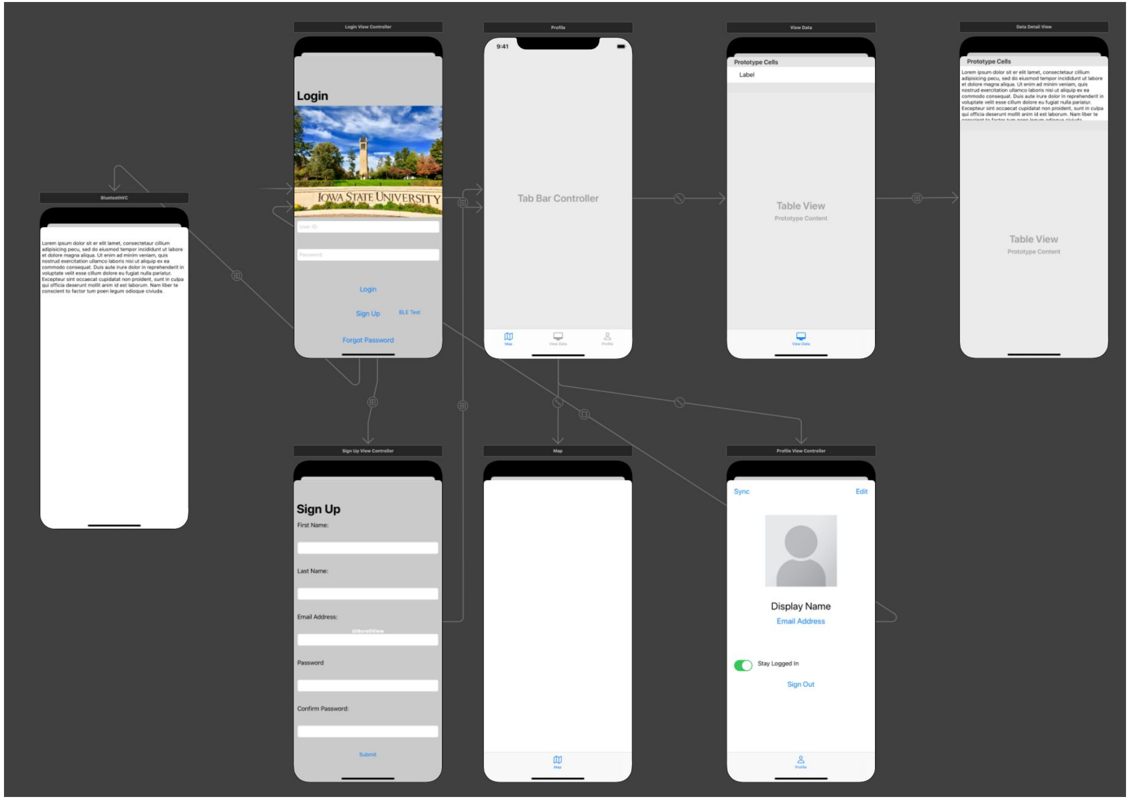


Detailed Design - iOS Application

- Emphasis on user interface, system, and container/modular design
- Data Persistence
  - Receive data from hardware device
  - Store/cache data in local database
    - Sync to cloud at the same time
  - After passing the threshold, send data to cloud, and release local storage
- Technologies Used
  - SwiftUI, UIKit
  - CoreData, CoreBluetooth, CoreLocation
  - Firebase, Mapbox

Shuangquan

Detailed Design Interface Diagram



Lorem ipsum dolor sit er elit lamet, consectetaur cillium adipisicing pecu, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum. Nam liber te conscient to factor tum poen legum odioque civiuda.

Lorem ipsum dolor sit er elit lamet, consectetaur cillium adipisicing pecu, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum. Nam liber te conscient to factor tum poen legum odioque civiuda.

Shuangquan

Runtime User Interface Diagram

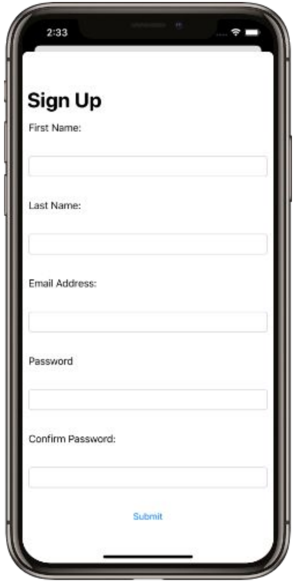


Image by **picapp.net**

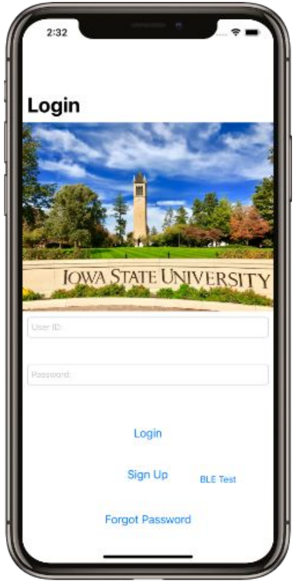
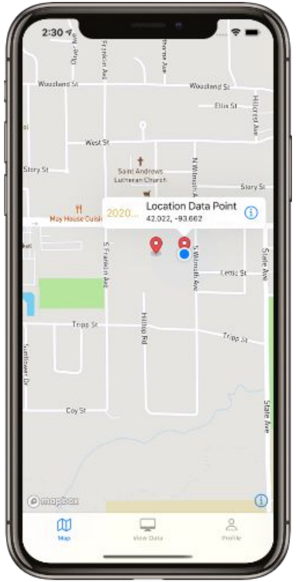
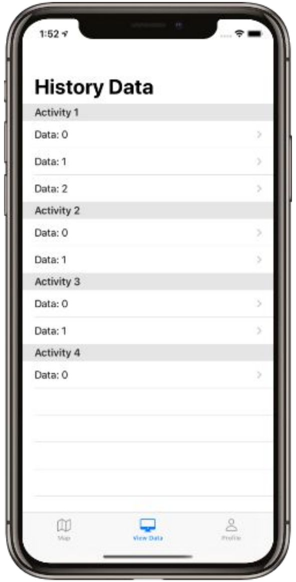


Image by **picapp.net**



2020... Location Data Point
42.022, -93.662

Image by **picapp.net**



History Data

Activity 1

Data: 0

Data: 1

Data: 2

Activity 2

Data: 0

Data: 1

Activity 3

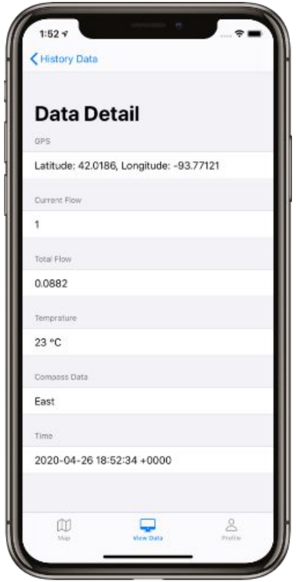
Data: 0

Data: 1

Activity 4

Data: 0

Image by **picapp.net**



Data Detail

GPS

Latitude: 42.0186, Longitude: -93.77121

Current Flow

1

Total Flow

0.0882

Temprature

23 °C

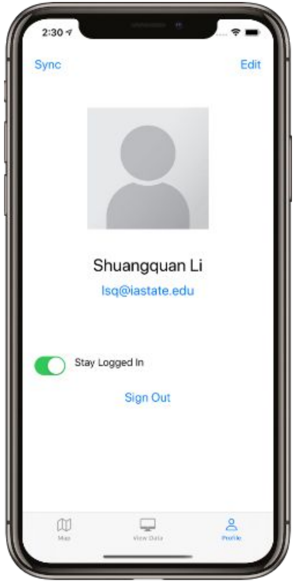
Compass Data

East

Time

2020-04-26 18:52:34 +0000

Image by **picapp.net**



Shuangquan Li

lsq@iastate.edu

Image by **picapp.net**

Detailed Design - API

- Emphasis on intuitive APIs
- Clean, Simple Database design
- Simple two-way API calls
- Technologies Used:
  - Firebase API
  - Mapbox SDK
  - Swift

Detailed Design - Hardware

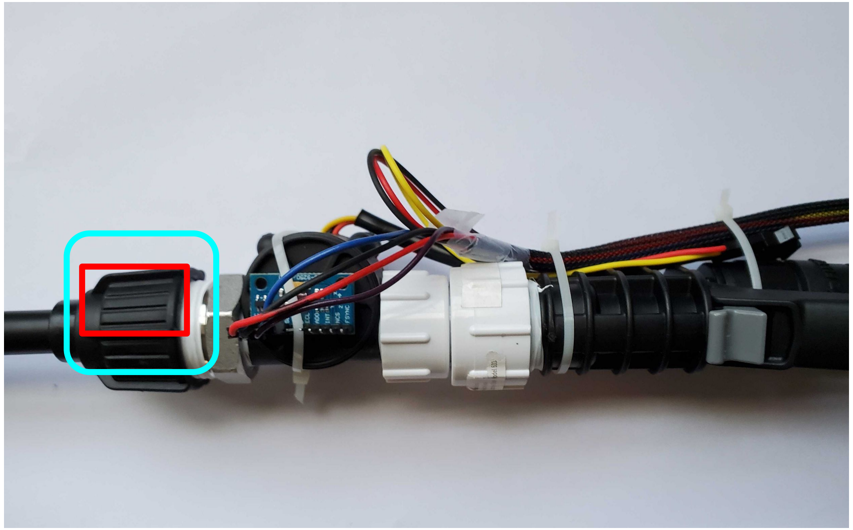
- Emphasis on PCB Design
- Technologies Used
  - Arduino
    - ArduinoJSON
    - TinyGPS++
    - SoftwareSerial
  - CircuitMaker

Detailed Design
Hardware

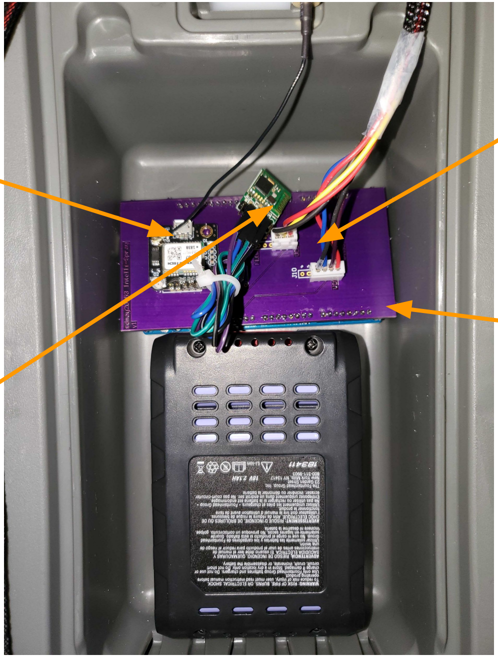
Image Key

Blue - Flow Sensor

Red - Compass/Accelerometer



Detailed Design Hardware



GPS Module

BLE Module

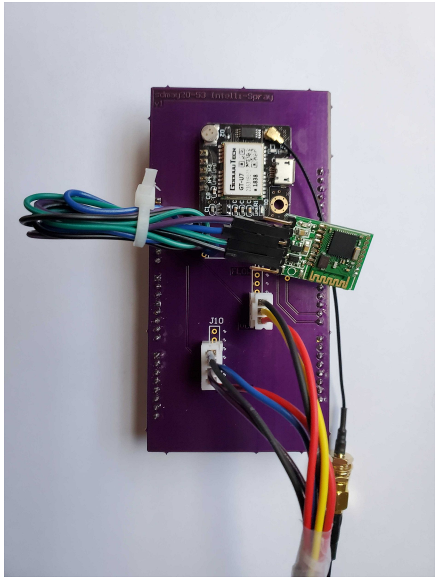
Flow and Compass Module

Arduino Mega 2560

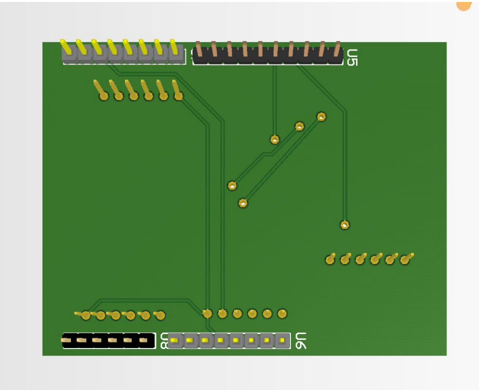
Arduino UNO R3

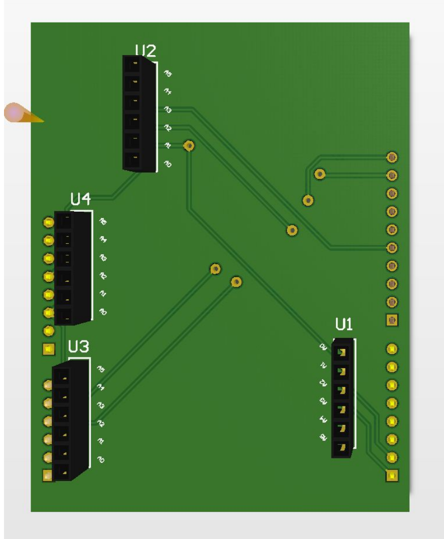
Detailed Design Hardware

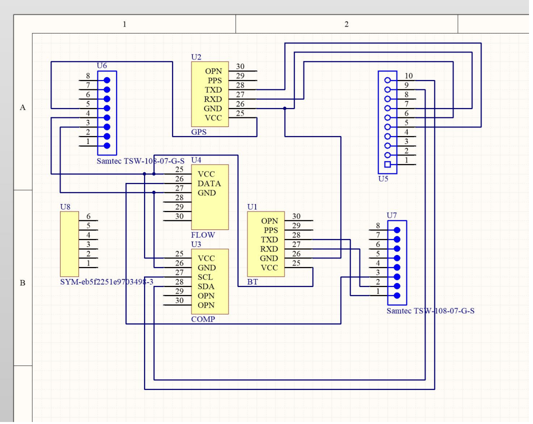
Arduino Mega 2560 Version



Arduino UNO Version







Kevin

Demo

Test Plan

- How is testing performed?
  - Simulator - iOS 13
  - Real device - iPhones
- Component/Unit testing
  - Testing each methods
  - Testing the modules of the hardware - flow sensors, GPS sensors
- Interface/integration testing
  - Testing where top-level units are tested first and lower level units are tested step by step after that
- Acceptance testing
  - Testing all the listed requirement in the design document.

Donald

Test Results

- Hardware
  - GPS module tested individually - Successful
  - Bluetooth module tested individually - Successful
  - Flow sensor tested individually - Successful
  - Compass sensor tested individually - Successful
  - Integration of parts with backpack - Successful
- Software
  - Login on app - Successful
  - Received data through bluetooth - Successful
  - Map plotting and storing mock data - Partially Successful
  - Map plotting and storing real data - Incomplete

Engineering Standards and Design Practices

- Software Development
  - ISO/IEC 12207
    - Stakeholder needs and requirements
    - Systems/Software requirements
    - Integration
    - Etc
- Digital Design
  - IEEE 1016
    - Data, Architecture, Interface, and Procedural Design

Kevin

Future Work

- Analyze spray patterns to better plot them on a map
- Try a different communication method
  - WiFi
- Try a different controller
  - ESP32 - Dual core processor
- Additional/Integration Testing
- Future development utilizing gyroscope data
- Other use cases
  - Proof of application
  - Drone application

Sean

What we learned/would change?

- Lessons learned
  - Develop to client requirements/project
  - Communicate openly with client and advisor throughout project
- Changes
  - More smaller deadlines
    - Collect more test data
  - Implement all of user interface in SwiftUI
  - Use WiFi instead of Bluetooth
  - Integrate earlier
  - Have a proper managed git repository

Conclusion

- Summary
    - Built an application and hardware that connect to a backpack sprayer allowing for data collection
- Issues encountered
    - Unable to get to field testing & client feedback
    - Implementation and integration logistical roadblocks
- Project Status
    - Rough Prototype created
        - Components are created
        - Testing in progress

Market Research

- Other Smart Backpack Sprayers:
  - AgTerra
    - Tracks location of user
    - Tracks what chemical has been sprayed
    - Android only

- Ours:
  - Tracks location of user
  - Tracks what chemical has been sprayed
  - Tracks where has been sprayed - Coverage
  - iOS only