UNIVERSITY OF ILLINOIS URBANA-CHAMPAIGN

Automatic Ice Fishing Rod

Team 60

James Niewiarowski
Luke Boelke
Andrew Osepek

Team Introduction

Andrew Osepek
Computer Engineer
Application & Mechanical

James Niewiarowski
Computer Engineer
Application & Control

Luke Boelke
Computer Engineer
Sensor, Power, & Control

Agenda

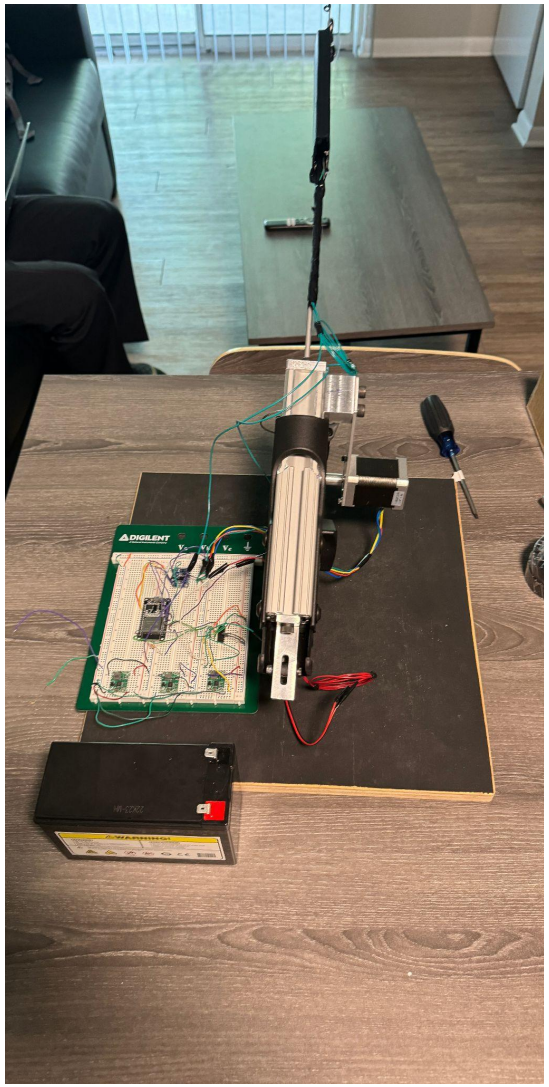
1. Problem & Solution
2. High-Level Requirements
3. Block Diagram
4. Physical Design
5. Subsystem Overviews and R&Vs
6. Challenges
7. Successes
8. Next Steps

Project Introduction

Ice Fishing Problems

- Requires constant attention to jig the rod
- Potential for long periods with no fish activity
- Risk of exposure to extreme weather conditions like cold, wind, and snow
- Can be uncomfortable

Project Introduction



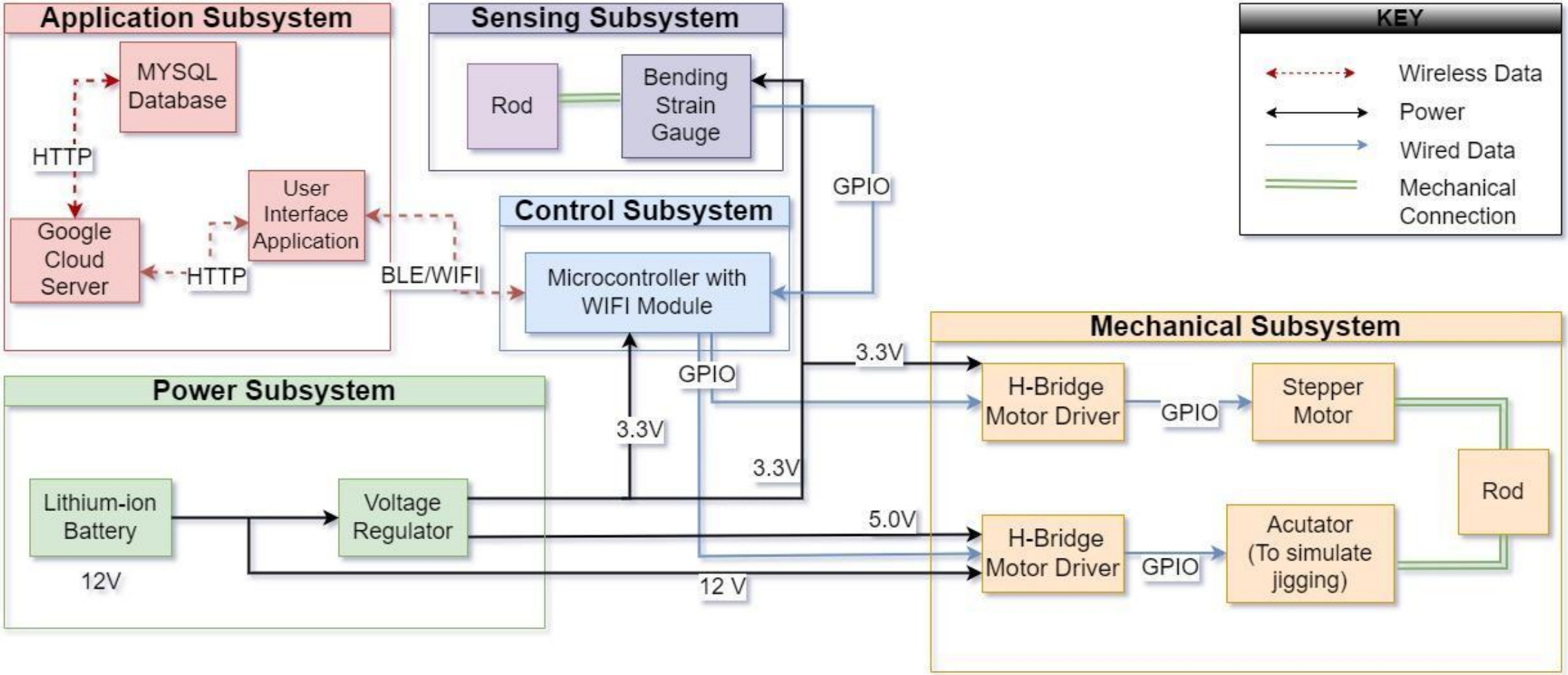
Ice Fishing Rod Solution

- Reel-in fish automatically
- Adjustable line depth feature
- Automated jigging mechanism
- Fish tug detection system
- Mobile app
- Battery powered
- Database to store catch information

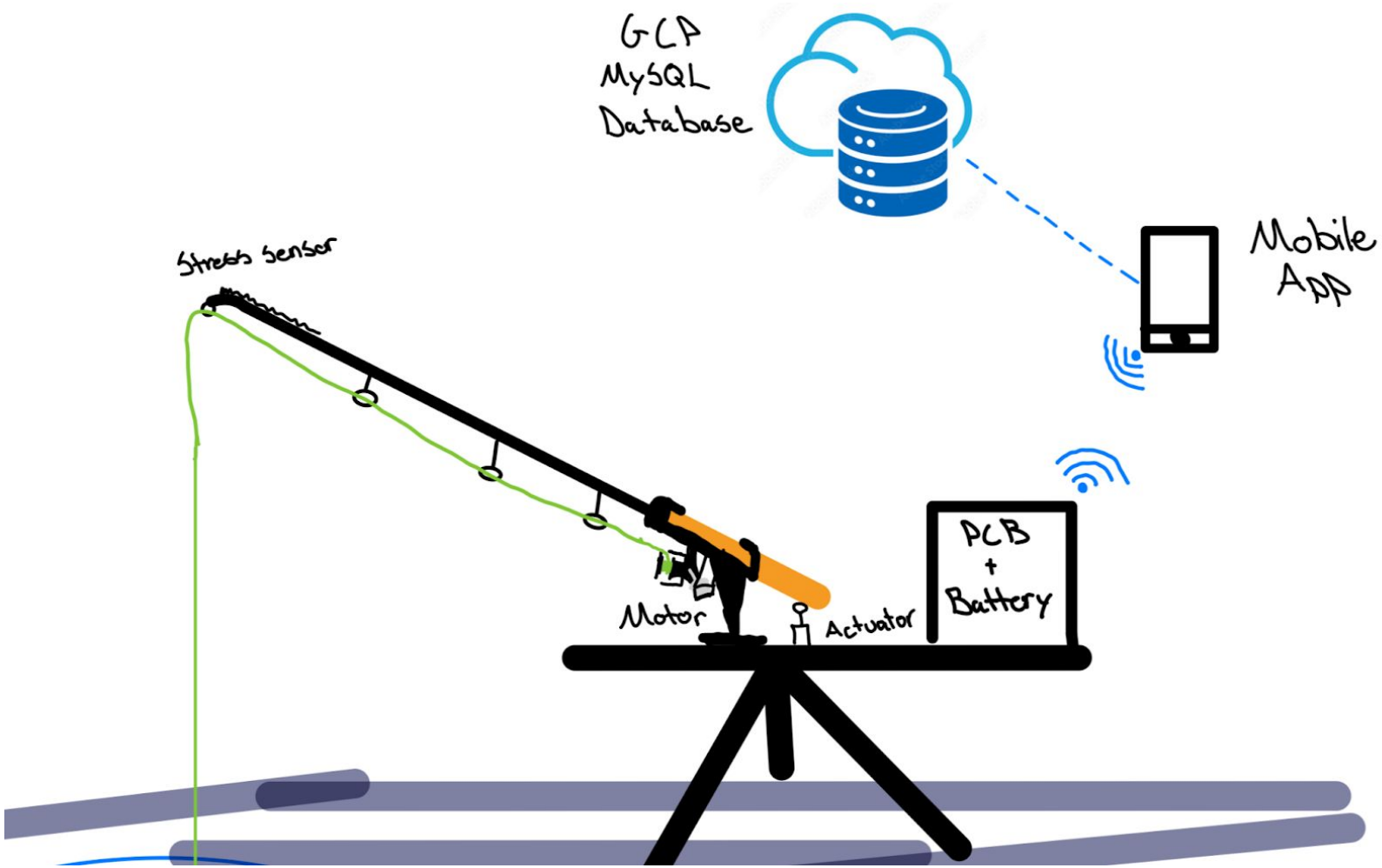
High-Level Requirements

- The user will be able to set up to 3 different jigging frequencies, a lure depth up to 50 feet (+/- 5 feet) in increments of 1 foot (+/- 0.25 feet).
- When a bend angle of 30 (+/- 10) degrees is detected, the jigging will halt within 5 seconds of detection, a notification will be sent to the user application, and the line will be reeled in automatically.
- The user will be able to record their catches in the user application with 7 different data fields. Previous catch information can be viewed in the application.

Block Diagram

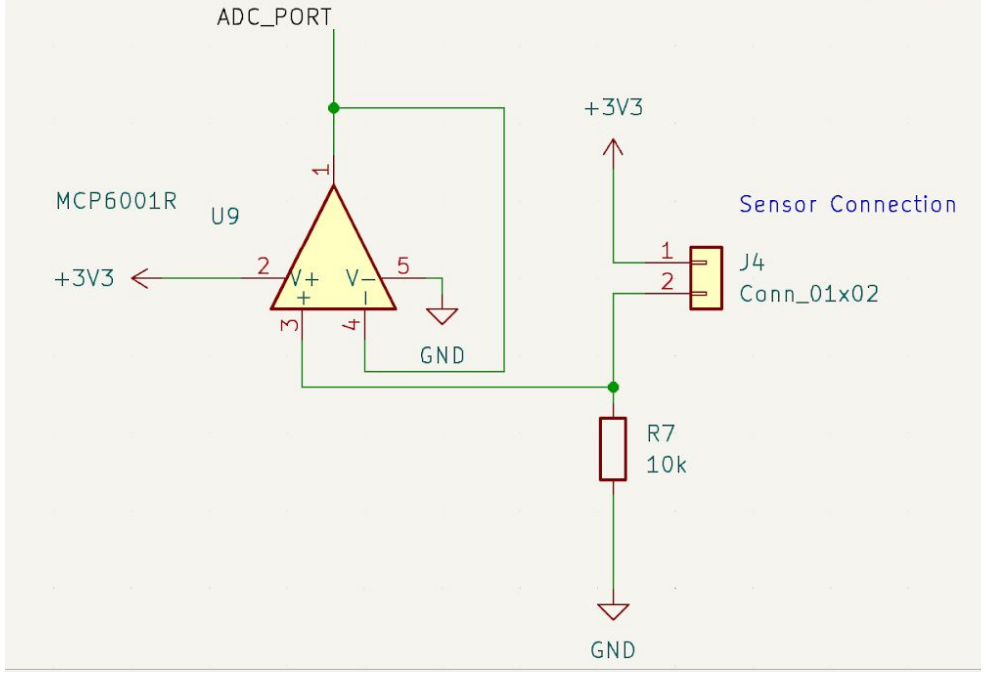


Napkin Idea


GCP
MySQL
Database
Mobile
App
Stress sensor
Motor
Actuator
PCB
+
Battery

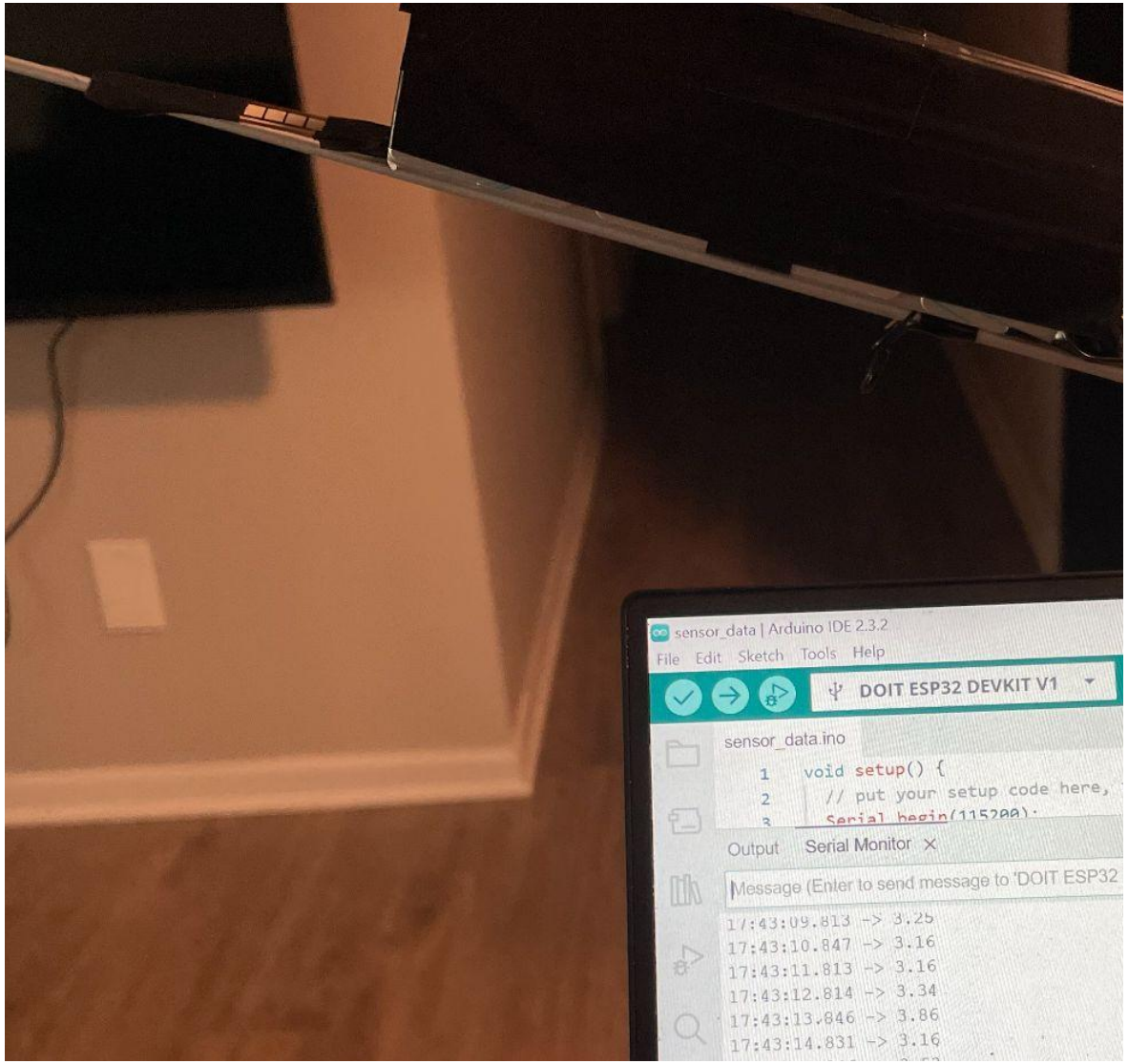
Final Product

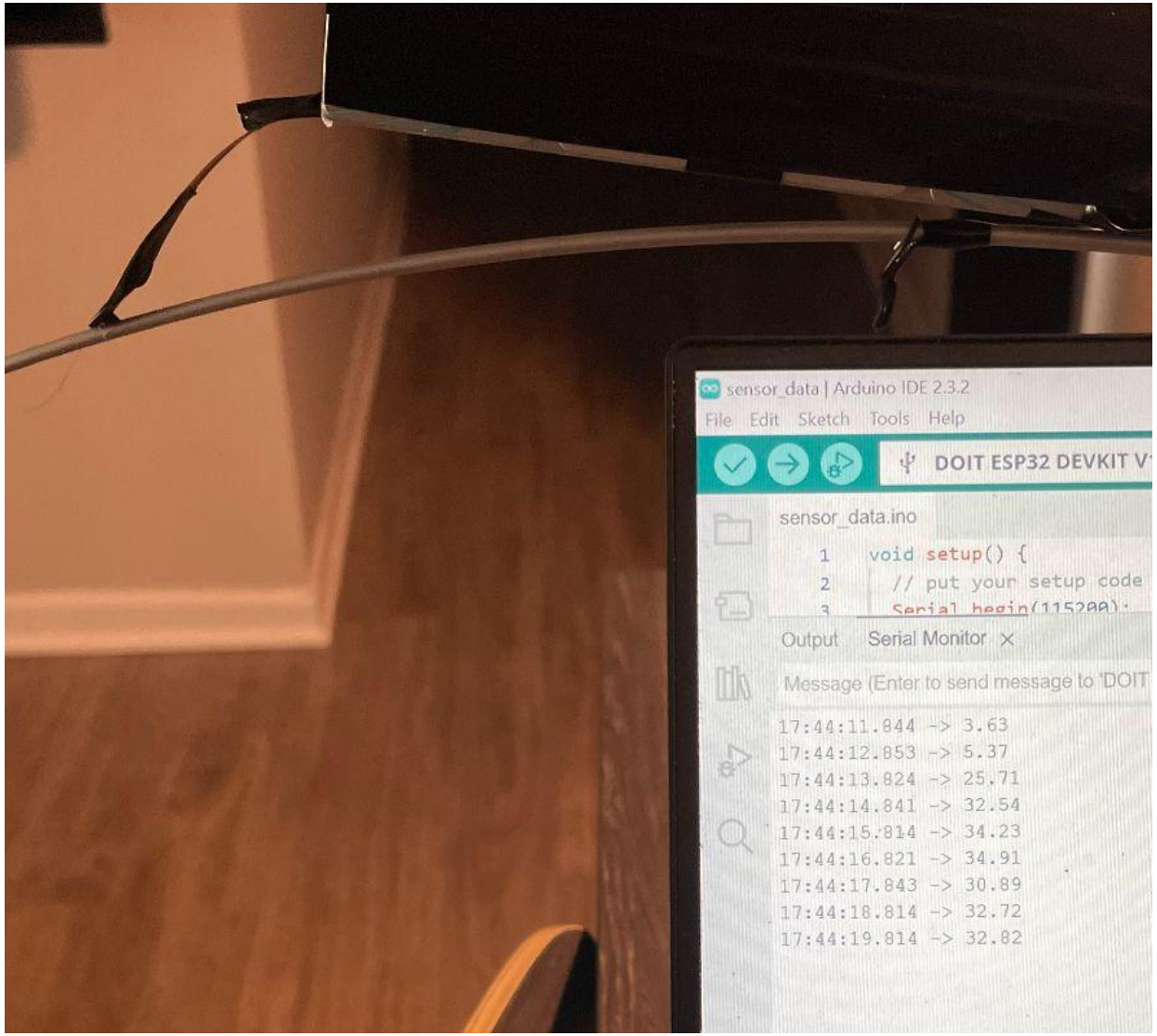
Sensor Subsystem



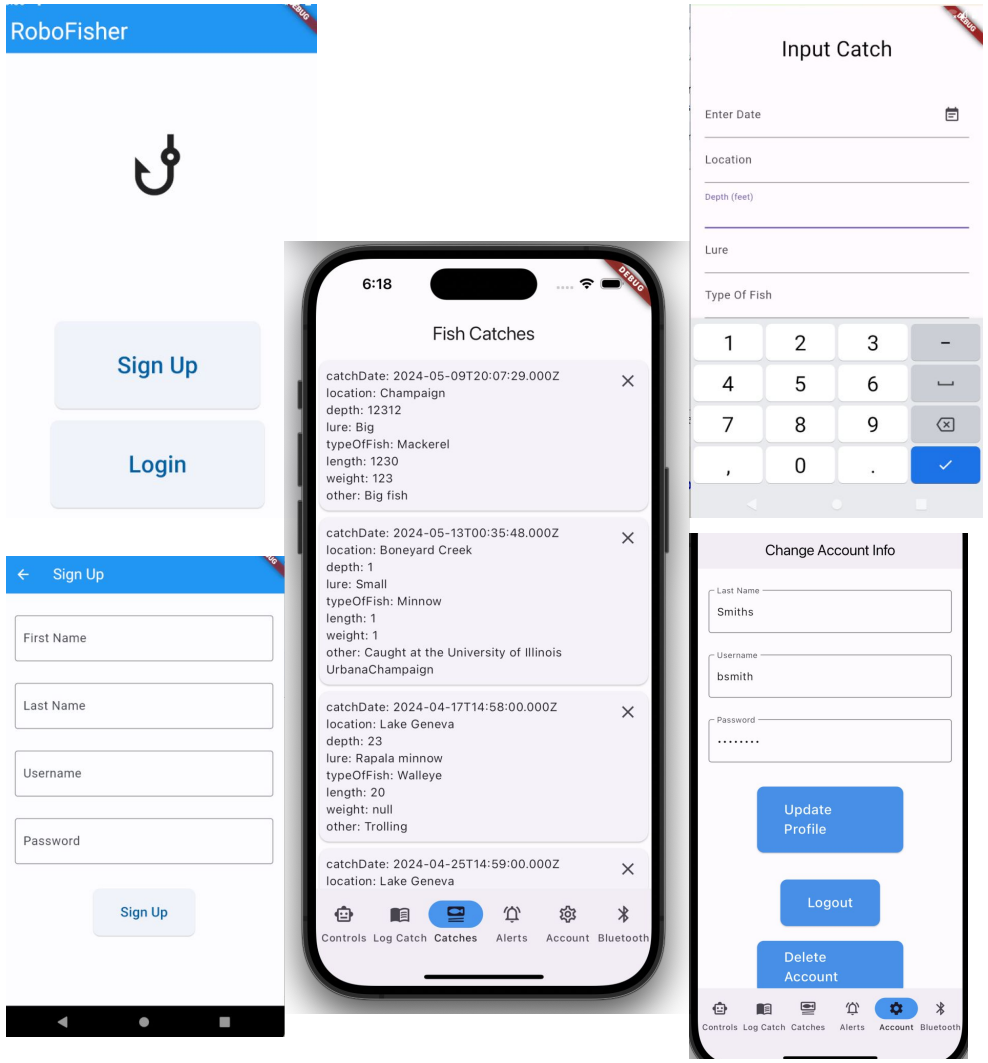
Requirements	Verifications
• Calculated bend angle is within 10 degrees of the actual bend angle of the sensor	• Bend the sensor 45 degrees • Use a protractor to measure the actual bend angle of the rod and take note of this value • Print the value of the bend angle calculated by the microcontroller • Ensure the calculated angle is within 10 degrees of the actual angle
• Line is reeled in when the sensor is bent at an angle of 30 (+/- 10) degrees	• Set the line depth to 3 ft • Bend the rod to an angle of 30 (+/- 10) degrees to signal a fish on the line • Ensure that the stepper motor fully reels in the line

Results


sensor_data | Arduino IDE 2.3.2
File Edit Sketch Tools Help
DOIT ESP32 DEVKIT V1
sensor_data.ino
1 void setup() {
2 // put your setup code here,
Output Serial Monitor
Message (Enter to send message to 'DOIT ESP32
17:43:09.813 -> 3.25
17:43:10.847 -> 3.16
17:43:11.813 -> 3.16
17:43:12.814 -> 3.34
17:43:13.846 -> 3.86
17:43:14.831 -> 3.16


sensor_data | Arduino IDE 2.3.2
File Edit Sketch Tools Help
DOIT ESP32 DEVKIT V
sensor_data.ino
1 void setup() {
2 // put your setup code
Output Serial Monitor
Message (Enter to send message to 'DOIT
17:44:11.844 -> 3.63
17:44:12.853 -> 5.37
17:44:13.824 -> 25.71
17:44:14.841 -> 32.54
17:44:15.814 -> 34.23
17:44:16.821 -> 34.91
17:44:17.843 -> 30.89
17:44:18.814 -> 32.72
17:44:19.814 -> 32.82

Application Subsystem - Frontend



Requirements	Verifications
● The user interface is easy and simple to use.	● Have our peers test our mobile applications UI and listen to their feedback.
● The application can allow the user to set the lure depth up to 50 feet and set 3 different jigging settings	● Navigate to the rod controls in application ● Input depth into field and option to enable jigging ● Print packet sent from application to microcontroller and ensure data passed matches input ● Ensure that microcontroller has received the packet/request from the mobile application

Application Subsystem - Backend

Database

- MySQL 8.0 database
- Stored remotely in Google Cloud Platform services
- Composed of a User and Fish Table

Server

- Built using Node.js and Express framework
- Server runs remotely on Compute Engine in Google Cloud Platform
- Supports CRUD operations from mobile application to database

Requirements	Verifications
• The user is able to create a unique account and login to the system. They have the opportunity to modify their account info.	• Turn on the DB instance and application server • Have the user create an account in the system • Check the database to see that the users' first name, last name, username, and password have been stored
• The user is able to store catch information between 7 different attributes while logged in	• Ensure the instance is running the same as above • Have the user logged in already • Input a catch information entry into the app • Query DB to see successful store for user • Log out of the application • Log back in and go to view catch information to see that the user can still view ONLY their catch information

Results

Database

```
mysql> select * from User;
+--------+-----------+----------+-----------+-------------+
| userID | firstName | lastName | username  | password    |
+--------+-----------+----------+-----------+-------------+
|      2 | Bill      | Smith    | bsmith    | password    |
|      3 | John      | Smither  | jsmith    | password    |
|      6 | john      | smalls   | jsmalls   | password123 |
|      7 | Andrew    | Ose      | ose123    | password    |
|      9 | bill      | smith2   | bsmith2   | 123         |
|     11 | james     | n        | jamesnie1 | password    |
+--------+-----------+----------+-----------+-------------+
6 rows in set (0.00 sec)
```

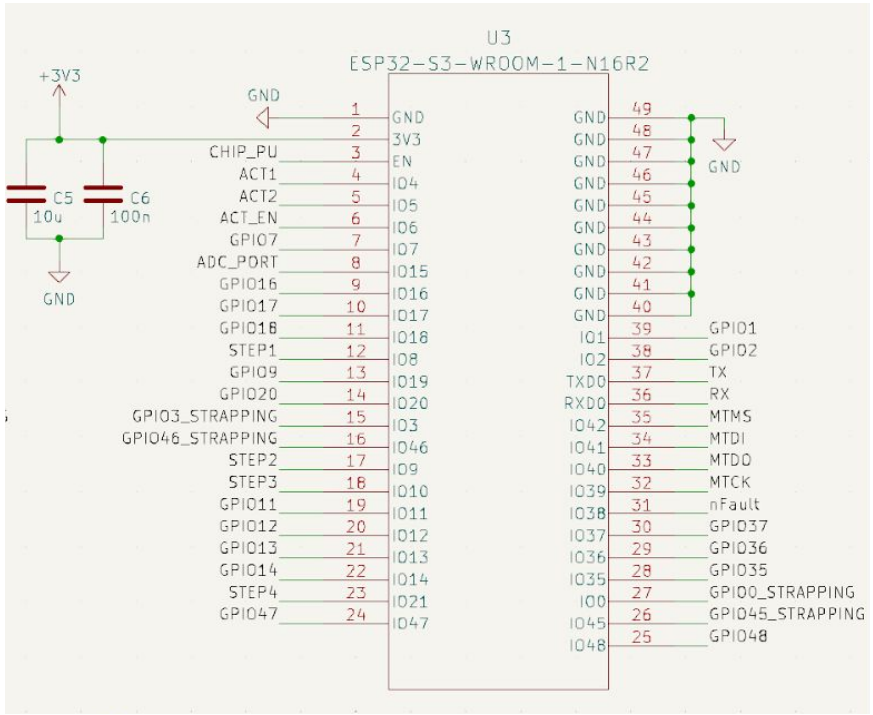
```
mysql> select * from Fish;
+--------+--------+---------------------+----------------+-------+--------------------
| fishID | userID | catchDate           | location       | depth | lure
+--------+--------+---------------------+----------------+-------+--------------------
|     26 |      2 | 2024-05-09 15:07:29 | Champaign      | 12312 | Big
|     31 |      2 | 2024-05-12 19:35:48 | Boneyard Creek |     1 | Small
|     32 |      3 | 2024-02-08 09:49:00 | Lake Michigamme|    15 | Worm on jig
|     33 |      3 | 2024-04-10 08:51:00 | Dupage River   |     9 | Wacky worm
|     34 |      3 | 2024-03-08 09:52:00 | Lake Geneva    |    22 | Slip bobber with le
|     35 |      3 | 2024-04-05 09:55:00 | Lake Geneva    |    18 | Rapala minnow
|     36 |      2 | 2024-04-17 09:58:00 | Lake Geneva    |    23 | Rapala minnow
|     37 |      2 | 2024-04-25 09:59:00 | Lake Geneva    |    17 | Rapala Minnow
|     38 |      2 | 2023-12-14 10:00:00 | Lake Michigan  |    26 | 1 oz Jig with leech
|     39 |      2 | 2024-04-11 07:02:00 | Lake Tomahawk  |     5 | Wacky worm
|     40 |      2 | 2024-05-15 16:11:32 | Champaign IL   |     1 | Large Lure
+--------+--------+---------------------+----------------+-------+--------------------
11 rows in set (0.01 sec)
```

Server

```
> nodeserver@0.0.0 start
> node app.js

AutoIceFishing app is running on port 3000
Connected to Auto Ice Fishing DB
POST /login 200 27.398 ms - 16
POST /add_catch 200 19.739 ms - 22
GET /catch 200 21.665 ms - 1505
GET /account 200 2.369 ms - 81
POST /add_catch 200 16.326 ms - 22
POST /add_catch 200 18.471 ms - 22
POST /add_catch 200 16.046 ms - 22
POST /add_catch 200 17.245 ms - 22
POST /add_catch 200 15.345 ms - 22
GET /account 200 2.260 ms - 81
POST /update_account 200 16.367 ms - 22
POST /logout 200 2.446 ms - 20
```

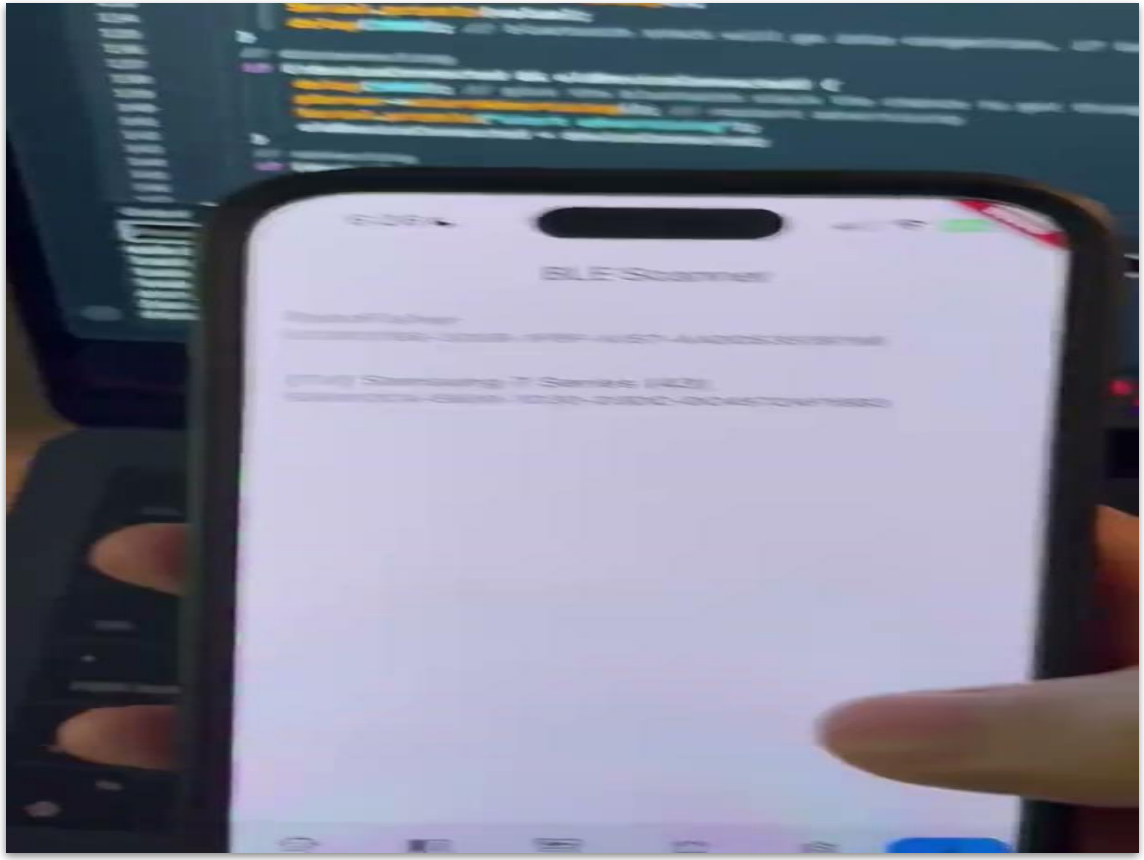
Control Subsystem



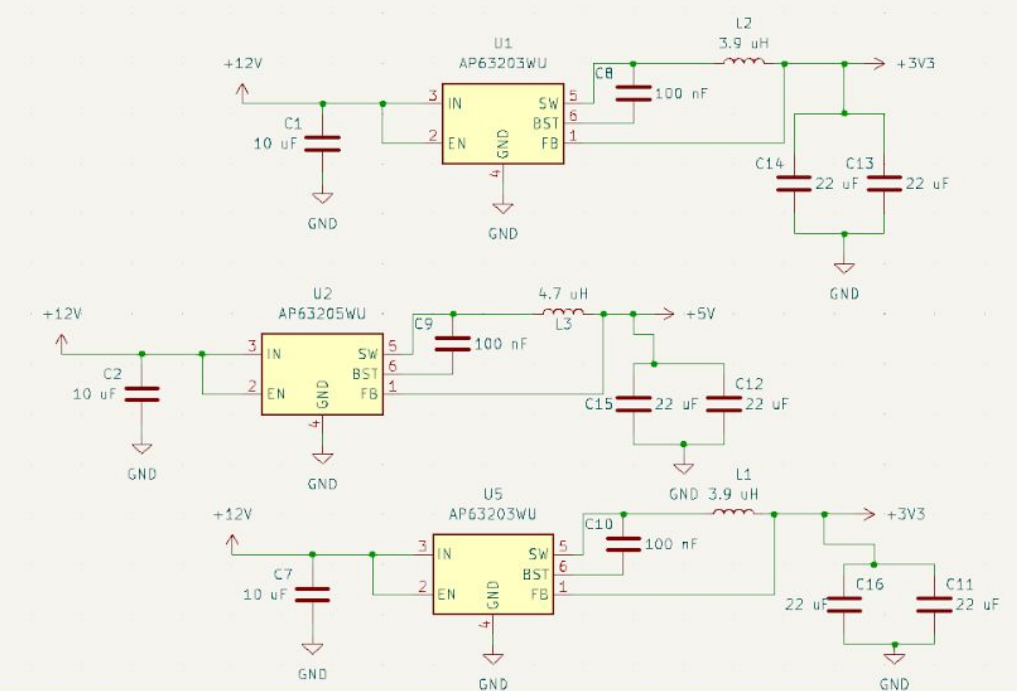
Requirements	Verifications
• The control subsystem can receive instructions from the application subsystem to reel the line out (in) at 1 ft (+/- 0.25 ft) increments	• Change the settings in the user application to a line depth of 2 (3) ft • Change the settings in the user application to a line depth of 3 (2) ft • Ensure that this difference is between 0.75 ft and 1.25 ft
• The control subsystem can receive instructions from the application subsystem to set the jigging frequency	• Set the jigging frequency to the lowest setting in the user application • Ensure that the actuator begins jigging at the corresponding frequency • Repeat for the other two jigging frequencies
• When the bend angle of the rod is 30 (+/- 10) degrees the control subsystem will signal the mechanical subsystem to halt the jigging motion and reel in the line, and notify the application subsystem	• Bend the rod to an angle of 30 (+/- 10) degrees • Ensure jigging halts and stepper motor reels in line • Ensure application subsystem displays that there is a fish present

Results

Sending and Receiving Data



Power Subsystem



Requirements	Verifications
• The power subsystem is able to output 3.3V (+/- 0.1V) for the sensor, microcontroller, and stepper motor driver, 5V (+/- 0.1V) for the actuator driver's VCC1, and 12V (+/- 0.1V) for the actuator driver's VCC2	• Connect one end of a voltmeter to the 3.3V/5V/12V output voltage line of the power subsystem and the other end to ground • Note this voltage value and ensure it is within 0.1V of the desired value

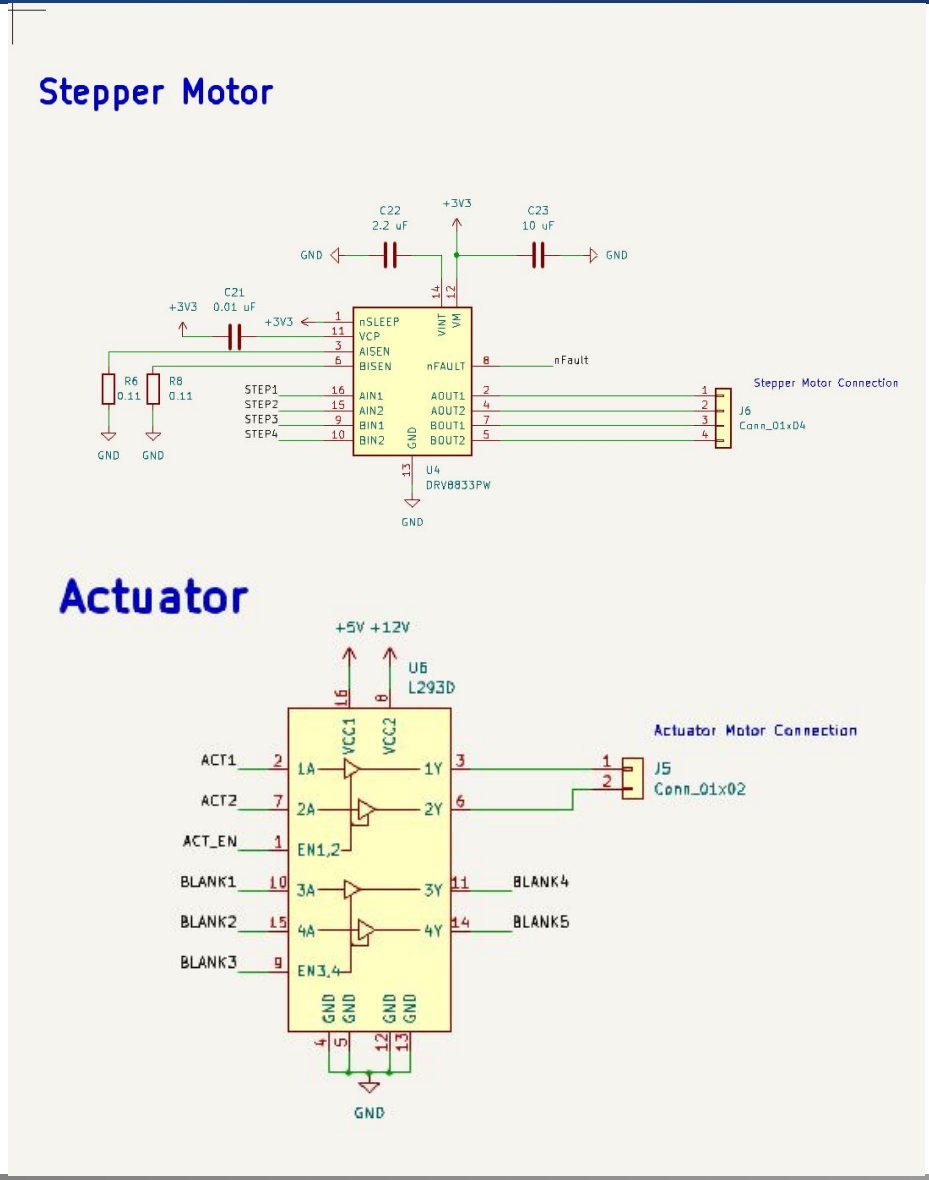
Results

Delivered to ESP32 and sensor:

Delivered to stepper motor driver:

Delivered to actuator driver:

Mechanical Subsystem



Automatic Reeling

- 3.3 V 4-wire stepper motor
- TI DRV8883 motor driver

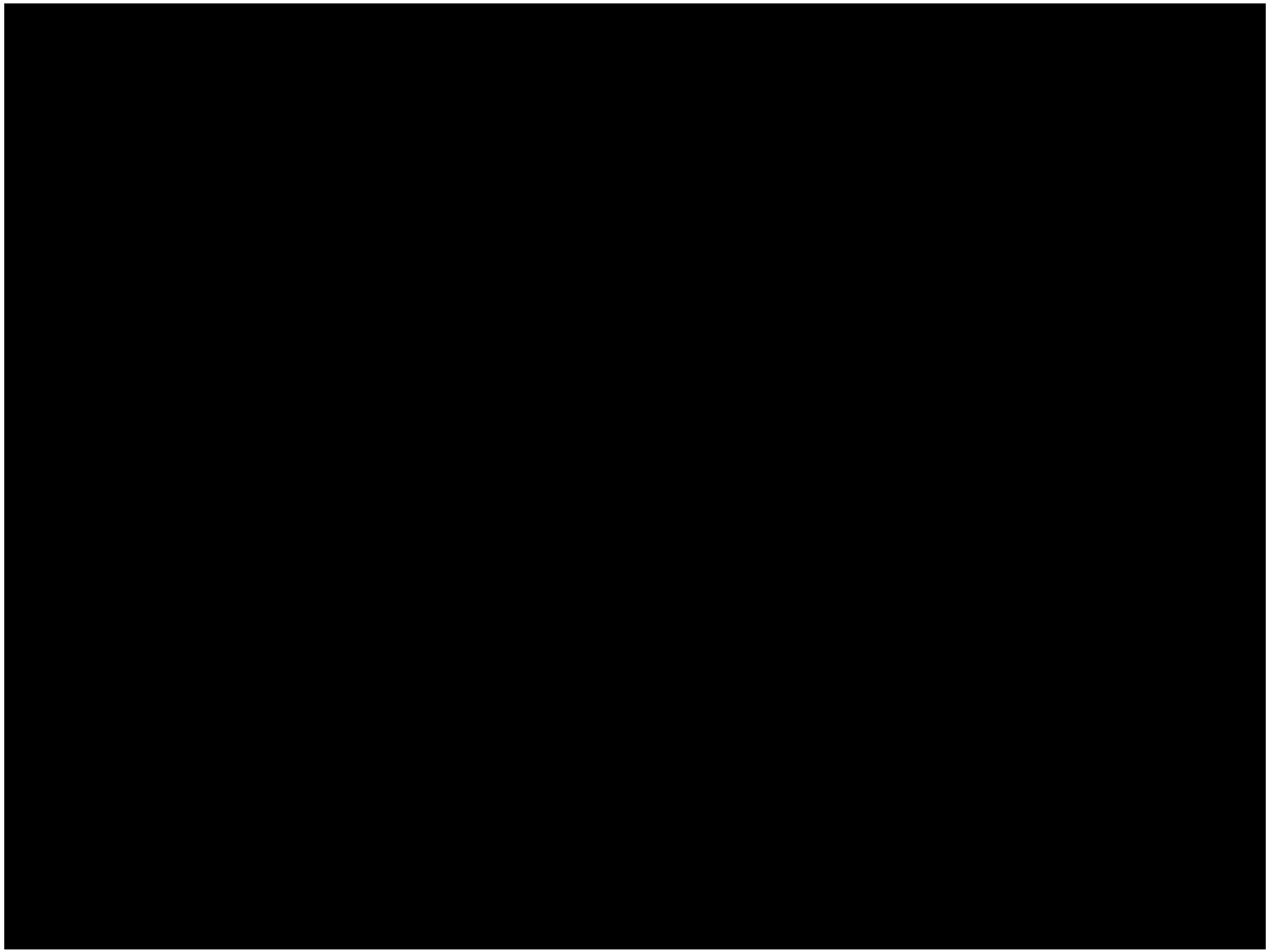
Automatic Jigging

- 12 V linear actuator
- TI L293D motor driver

Mechanical Subsystem - Requirements & Verifications

Requirements	Verifications
• The stepper motor is able to lower the fishing line to 50 ft (+/- 5 ft) when fully reeled out	• Measure the length of line out with tape measure.Note this length and ensure that it is between 45 ft and 55 ft
• The stepper motor is able to lower/raise the fishing line in increments of 1 ft (+/- 0.25 ft)	• Use a tape measure to measure the actual length between the start and end of the line and note this length • Ensure that this difference is between 0.75 ft and 1.25 ft
• The actuator is able to jig the rod up and down within 0.1 Hz of the desired frequency	• Use a stopwatch to measure the jigging frequency to ensure its success
• When a fish is detected by the sensor subsystem, the stepper motor fully reels in all of the line currently out, returning to the original position of a depth of 0 ft (+/- 1 ft)	• Bend the rod to an angle of 30 (+/- 10) degrees to signal a fish on the line • Wait for the stepper motor to reel in the line • Measure the length of the remaining line out, and ensure it returns to its original position.
• The stepper motor is able to fully reel in a weight of 1 lbs (+/- 0.5 lb)	• Attach a 1 lb (+/- 0.5 lb) weight to the end of the line • Bend the rod to an angle of 30 (+/- 10) degrees to signal a fish on the line • If the stepper motor is able to fully reel in the weight, the test succeeded
• The jigging motion halts within 5 seconds when the bend angle of the rod is 30 (+/- 10) degrees	• Bend the rod to an angle of 30 (+/- 10) degrees to signal a fish on the line and start a stopwatch • Wait until the actuator stops the jigging motion and stop the stopwatch • Ensure that the time on the stopwatch is less than 5 seconds

Project Demonstration



Challenges

PCB Design

Many components on our original design were extremely small and difficult to solder, and some did not have the proper footprints. In our initial design, not all of our components were correctly grounded.

Stepper motor

We burnt our motor driver forcing us to reevaluate our motor driver's needs, resulting in us raising the current control limit of our driver.

Flutter Bluetooth

We had difficulty developing bluetooth within Flutter, because the development environment required a physical device (Iphone or Android) to use. Apple requires Macbook to test on iPhone, meaning we only had one computer to test with.

Successes

User Application

Built a mobile application that allows the user to successfully set the depth and jigging settings on the rod. User App is connected to a MySQL database which is stored in Google Cloud platform services along with a Node.js server. The database allows the user to store catch information.

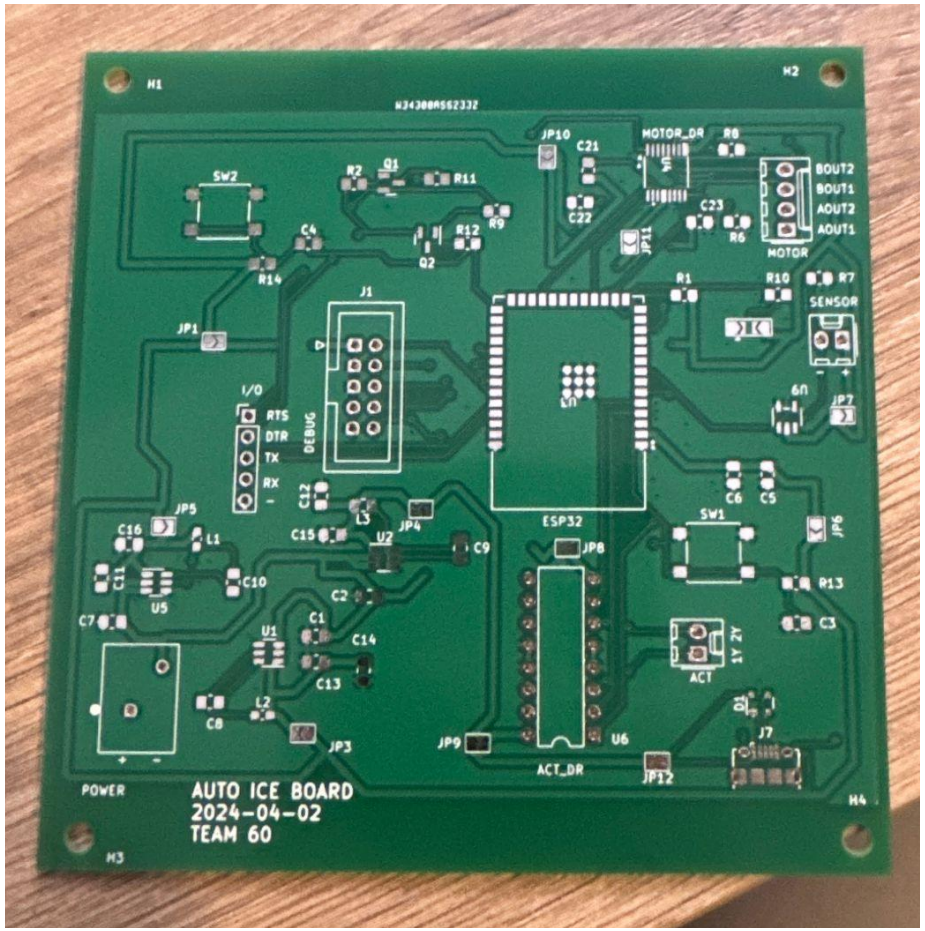
Sensing

User is notified when a tug on the line occurs, and the fishing line is reeled in.

Jigging & Reeling

User can successfully jig their lure at 3 different jigging frequencies. The user can reel-in/out their line at increments of one foot up to 50 ft.

Next Steps



- Fully integrate all subsystem hardware on PCB
- Make appearance of device more professional
- Modify motor to be more powerful and handle more strain/weight
- Demonstrate ice fishing rod in a real setting (frozen lake)
- Place application subsystem within a Docker container image

Conclusion

What we Learned

- Project management
- Integration of technical knowledge
- Team collaboration
- New technical skills

Thank you

Questions?

The Grainger College of Engineering

UNIVERSITY OF ILLINOIS URBANA-CHAMPAIGN