HydroBotics Technical Design Report

Kira Benton, Jade Palewicz, Jonathan Lavoie, Codi Petersen

***Abstract*—The UWF HydroBotics team began designing their Autonomous Underwater Vehicle (AUV), ArgoPelago, in 2024. As first-time RoboSub competitors, the HydroBotics team has cultivated its skills to design a proficient AUV that will perform now and serve as a foundation for future UWF teams. The competition strategy prioritizes efficiency and versatility, seeking to maximize their scoring from the course. After completing the base of ArgoPelago, this year has been spent on integrating the necessary systems for all tasks, while focusing on mastering the data collection, navigating, and mapping tasks. This competition strategy has been the fundamental motivation behind HydroBotics' design process and decision-making. This paper discusses these strategies and their implementation on ArgoPelago.**

I. Introduction

The HydroBotics (HB) Underwater Robotics project at the University of West Florida promotes student learning surrounding underwater autonomous systems with our three sub-teams: Mechanical, Electrical, and Software. HB also promoted student engagement in the local community by attending local AI meetups.

Presently, HB is developing an AUV for the RoboSub 2025 competition. The team was started as a Capstone project for 6 senior Mechanical and Electrical Undergraduate students. 3 summer research students have also been added to the team because many of the previous team members graduated. The goal of the design was to allow for easy modifications for future teams while still containing each of the necessary subsystems to complete the gate, torpedo, and maze challenges.

In the future, HB plans to expand significantly and build on the research completed this year. Because the team started from scratch at the beginning of this past school year, the sub was designed in two stages. First, a tethered stage to allow for easily obtaining underwater pictures for the image training, and secondly, the autonomous stage to have the sub ready for the competition. The team bought the main hardware components for reliability and easy integration into the system, but is building a custom claw, marker, and torpedo system to conserve money.

This paper will detail HB's strategic vision for RoboSub 2025, describing ArgoPelago's development, highlighting the design methodology, and explaining the testing methods used to verify systems.

II. Competition Strategy

A. Main Goal: Proof of concept

The main purpose of the HB team is to develop an autonomous submarine from scratch and have it deployable within a year. The team is attempting to create each sub-system for the RoboSub competition to the best of their abilities, but strives to identify any issues or areas for improvement while building a foundation for future teams to improve upon. A summary of the school-year goals included:

- Designing a remote controllable submarine from the ground up.
- Use the tethered sub design as a basis to implement the autonomous system and take videos of the HB-built course.
- Build a competitive autonomous system and integrate the competition-specific components

The Capstone team that began the project planned to complete the tethered sub design by the end of their class. They successfully completed that goal, leaving the summer team with a tethered vehicle and a custom-trained image detection model that was able to distinguish between red and white pvc pipe using YOLO.

B. Project Timeline

From Sept 2024 - Dec 2024, HB laid out a plan for ArgoPelago's design and placed material orders. From Jan - March 2025, HB focused on building ArgoPelago and getting it test-ready. In April the first successful remote-controlled pool test was completed. The next month, five of the seven original members of the HB team graduated, so three new teammates were recruited to integrate the final components of the system. Over the summer the team had five members working on ArgoPelago.

Because the team was limited to a single year to design and build the submarine the team designed the submarine in two phases in order to test the equipment. First, the sub began as a computer model and then an operable tethered submarine with no extra features. Next, while the team was waiting for the doppler velocity log to arrive, the team successfully designed and built the claw and created the custom-trained computer vision models. Now that the team has the doppler velocity log, half the team will focus on integrating this system through the end of June, while the rest of the team continues to work on the pinger and torpedo system. Then, July will involve integrating each of the task-specific components in with the autonomous system and the team will test the system frequently.

C. *Team Challenges*

There were two main challenges that the team encountered. The first was funding. Because the team only consisted of a small number of engineering students and was a new team, it was difficult to find sponsors. This resulted in the team not being able to purchase and receive the doppler velocity log (DVL) from Teledyne [4], until June. Also, the team prioritized less expensive options over more reliable options in the beginning. In many cases such as the thruster mounts and claw enclosure the components were 3-D printed and coated in resin to prevent instability due to the water seepage. Also, instead of buying Blue Robotics thrusters, the team bought a less expensive option from amazon and 3-D printed shroud covers to meet the competition requirements as shown in figure 1.

Figure 1. Shroud Covers on Thrusters

The second challenge was the learning curve presented with programming and understanding the software surrounding autonomous systems. The team lacked a background in computer science or computer engineering knowledge and had very limited experience with Python or Linux-based systems. The team overcame this challenge by taking free online Python courses[1] and completing the Robot Operating System tutorials[2] to better understand how the system package would need to be designed.

The managerial team first consisted of two co-team leads and each member was responsible for a specific subsection of the project. Once summer began, the team consisted of one team lead and each member was focused on a specific section of the course.

D. *Course approach*

To prioritize the tasks for RoboSub 2025 the following considerations were discussed:

Challenge Plan:

1. Collecting Data: HB will take the coin flip to gain extra points and have the sub programmed to turn clockwise until seeing the gate. ArgoPelago will also enter the gate with style to achieve maximum points.
2. Navigate the Channel: Since this is one of the simpler challenges, the team will plan to complete this challenge first in case the sub gets stuck on one of the more difficult challenges later in the run.
3. Tagging: ArgoPelago will attempt to earn extra points in this section by shooting the torpedoes from a “far” distance, as denoted by the Robosub handbook [3].
4. Ocean Cleanup: Based off of last year’s competition, many of the other teams’ points were collected by completing the style spins after each challenge. The team will plan to rotate clockwise for each item in the bins detected by the camera.
5. Drop a BRUVS: Because this challenge would require a downwards-facing camera to operate, the team has decided to place it last until the dual-camera operation can be implemented.
6. Pass back through the gate: This will end the run for the team.

III. Design Strategy

ArgoPelago Overview and High Level Development

ArgoPelago is shown in Figure 1 at its tethered stage. The frame was constructed from PVC allowing for modularity in case the team decided to alter the design.

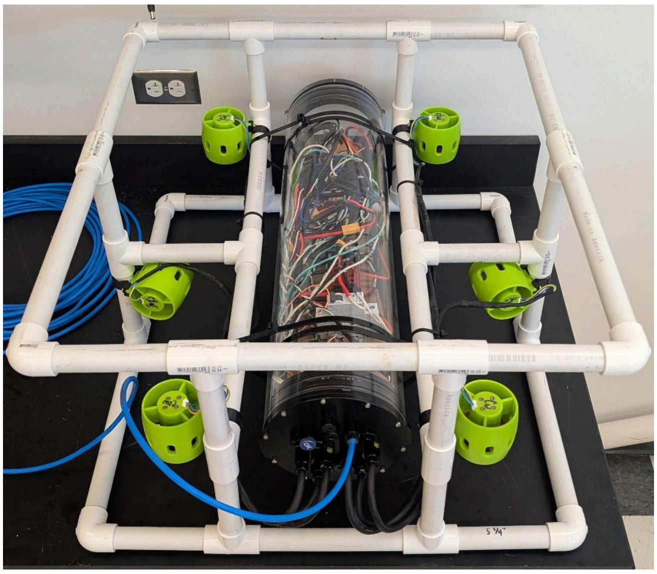


Figure 2: ArgoPelago

The following design sections highlight each of ArgoPelago's attributes along with the creative design and integration with the rest of the system.

Electrical system

A. *Power System*

ArgoPelago utilizes one lithium-polymer battery as shown in Figure 2 and during a typical 2 hour pool session, only around 30% of the battery will be drained. The battery is in the same enclosure as the rest of the electronics and the team has not had any issues with needing to swap out the batteries during sessions. If there was a need, the batteries simply unplug and the next one can be plugged in within a couple of minutes. The battery is connected to a power distribution board which interconnects all of the electrical components. This power module ensures all of the thrusters and the rest of the components receive the correct amount of power from the battery.

Figure 3: Battery connection

The lithium polymer batteries used in the sub are 22000(mAH) with a C-rating of 12(1/hr), so the maximum current draw is 264A The idle current of the electronic components is 400mA. This means the battery should not have to be recharged or be swapped out during a competition run.

B. *Motion Control*

During the tethered stage, the Raspberry Pi 4b is the main computer for the sub. However, once the team receives the usb adapter for the DVL, a Jetson Orin Nano will be integrated as the new central computer. The Jetson will perform object identification, and once a navigation solution has been determined, it is passed to the flight controller: a PixHawk 4. The Pixhawk was chosen over a Navio Flight controller because of its built in gyroscope and ability to perform autonomous navigation.

The sub will use two cameras for underwater imaging. The front facing camera is a low light camera from BlueRobotics implemented by the Capstone team. The bottom facing camera will be implemented when the Jetson is integrated.

Software System

The software system runs on a Jetson Orin Nano and utilizes a ROS2 package to run different python modes which execute each task.

HB is using YOLOv11 for the image recognition node. The team has decided to create two separate Robot Operating System 2 (ROS2) packages. One for testing and pre-qualification and a second one for the competition specific tasks. The team plans to run separate image models within a package. There will be a model which recognizes each of the competition courses and then once a challenge is seen and selected by the robot, a smaller challenge-specific custom-trained model will be run. Once the function is run, it will return to the larger image model. This approach will allow for easier testing and development. Sample videos were taken with the sub's internal camera and uploaded to Roboflow for annotation. These datasets were trained locally using Ultralytics's Python library and YOLOv11.

The package breakdown consists of nodes for the custom image model, movement controls, leak sensor, and Arduino commands. The arduino controls the extremities (claw, acoustic receiver, torpedoes, & BRUVs system). The image training team member utilized RoboFlow to keep the dataset organized. Then, it was trained locally to obtain the best.pt file, and this was plugged into the ROS2 package.

Mechanical System

A. Claw

The claw was modeled after general submarine claw designs. It is equipped with two grippers that are actuated by the motion of a screw shaft. It will mount to the AUV face down. The housing is watertight and holds the mechanical components and the motor. The claw is controlled by an arduino, but to reduce the volume of the claw, the arduino was put on board the AUV and not in the housing.

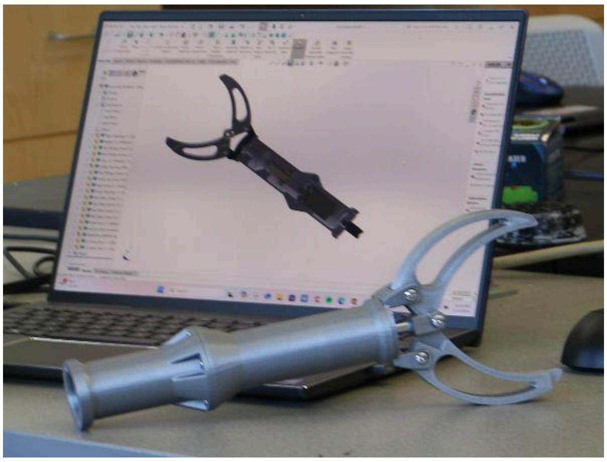


Figure 4. Custom Claw

B. Torpedo/Marker Dropping System

The systems used for the tagging and BRUVS challenges are being designed to be housed in the same mechanism. This strategy is being used to reduce the cost of both systems by sharing the same underwater servo. The torpedo will be launched by a spring-loaded mechanism. The torpedo and launching system was designed to be 3d printed for simplicity. This will allow the ability to test the system and adjust the design of parts easily for best possible results.

IV. Testing Strategy

A. Frame testing

With the short amount of time the HBs team had available to design and build the sub, there was not much time left for post-build testing. Because of this, the team decided to run simulations in IsaacSim before building in order to minimize the number of changes required later on. An image of the basic sub in the software is shown in Figure 5.

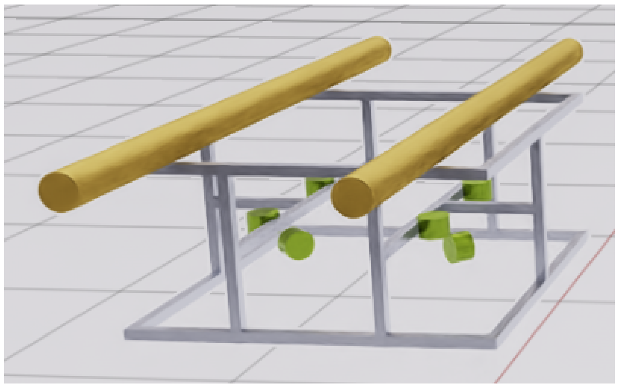


Figure 5: IsaacSim Simulation design

Originally a four-thruster design was considered to reduce the amount of power needed from the batteries. However, the team quickly realized that the sub needed at least 6 thrusters to allow for stability. When the team only had two thrusters to propel the sub up and down the simulation sub started doing flips. To solve this issue, the team altered the design to include a rectangular pattern of four vertical thrusters to move the sub up and down while two horizontal thrusters were placed in the center on the bottom of the sub.

Once the frame was built, the electrical enclosure was mounted without electrical components inside to determine how buoyant the sub would be in the water. The frame successfully floated and no water entered the tube. Next, a Raspberry pi was used running BlueOS to test the motor connections. The motors were also connected to the Jetson Nano just to see how they would work while running from a python script. The thrusters worked with BlueOS and would work in manual mode with a PixHawk flight controller. However, if the team tried changing the flight controller to guided mode, there was an error stating that the PixHawk could not access position. The team realised this was due to the lack of a doppler velocity log and after obtaining funding was able to order one in June 2025. In the meantime, HB developed ArgoPelago as a remote operated vehicle in order to take videos of the course and develop the image recognition system.

During the initial test of ArgoPelago in the water, HB found that the air inside the PVC frame prevented the sub from being able to submerge even when the thrusters were at full throttle. This was easily fixed by drilling holes in the PVC to prevent the holding of air. After this the sub was driven around underwater to test for viability. Each of the test plans and procedures are outlined in Appendix B. Each electrical component was also tested separately upon arrival to understand the procedure of operation.

B. *Claw Testing Highlight*

The team decided to build a custom claw instead of purchasing one in order to reduce cost and allow the team to learn how a claw is constructed. The original design was made from PLA and broke easily. The original idea was to have a single long tube which would allow for the motor and shaft to fit inside, but the difficulty with managing and mounting the inner components was quickly discovered. Next a two part system made from ABS was designed to allow the motor to be mounted on one half and have the shaft fit on the other side to open and close the claw. The shafts were custom fabricated, but we discovered that the aluminum material chosen stripped easily and a new part was required. The grippers and the shaft-to-gripper connector had a tendency to crack when in use. These parts were redesigned and their printing orientation was changed to prevent this. A custom PCB motor controller was also designed to control the claw and is seen in Figure 4. The electrical and mechanical systems of the claw were then combined and tested with an arduino. In the final design, the claw will be controlled by the Jetson Orin. The team is still in the process of finalizing the claw.

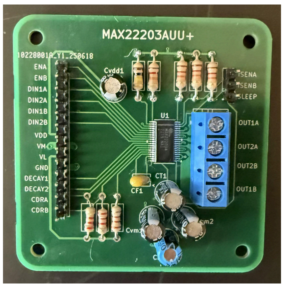


Figure 6: Custom Motor Controller Design

C. *Pool Testing:*

As the summer months approached, HB increased pool testing time in June to discover any software or hardware issues that may arise with excess use. The team discovered that the electrical tube had leakage issues because the inner diameter of the electrical tube was outside of the company-advertised tolerances and the team did not discover this until it was too late to return in. The team decided to fix the issue by ordering O-Rings that are 1mm thicker in hopes that this will better seal the tube. A leak sensor was also purchased to be mounted inside the enclosure so the sub can power off and rise to the surface in the case of a leak.

During early summer pool testing, the task obstacles were placed in the pool and the sub was controlled manually to obtain footage of the obstacles. These videos were used to create the datasets for YOLOv11 models set to run on the Jetson Orin Nano. Later, pool sessions will test the effectiveness of the image models once they are implemented. Then, the models will be adjusted and refined as necessary.

V. Conclusion

HB has utilized the year to learn and develop a submarine which integrates mechanical, electrical, and software components. These developments have laid a foundation for the current team to compete at the RoboSub 2025 competition and for future teams to improve upon them. Current team members will be able to mentor future team members on the topics of their individual research, advancing the team each year.

VI. Acknowledgements

The RoboSub team would first like to thank the team's dedicated advisor Dr. Tarek Youssef who introduced us to the idea of creating a RoboSub and has been there to help us at every step of the way. He also was kind enough to donate the Jetson Orin Nano and PixHawk 4 to the team. The team would next like to thank the team's contacts Renan Barbosa and Robert Murrer from the Gathering of Aspiring Technologists Club for helping guide the team in the right direction on the software side of the project.

The team learned python skills through the use of the MIT OpenCourseware programs and Codecademy courses.

The team also received a generous $2500 donation from Florida Power and Light which greatly contributed to the team's ability to purchase the acoustic receivers and electrical enclosure. Next, a thank you to the UWF Electrical Engineering department for purchasing a Doppler Velocity Log for the team. We would lastly like to thank the University of West Florida's Office of Undergraduate Research for providing $3000 for us to buy supplies and paying for us to have a few extra students on the team over the summer.

VII. References

[1] "Syllabus | Introduction to Computer Science and Programming in Python | Electrical Engineering and Computer Science," *MIT OpenCourseWare*. https://ocw.mit.edu/courses/6-0001-introduction-to-computer-science-and-programming-in-python-fall-2016/pages/syllabus/

[2] "ROS 2 Documentation — ROS 2 Documentation: Humble documentation," *docs.ros.org*. https://docs.ros.org/en/humble/

[3] "Team Handbook - RoboSub 2025," RoboNation, 2025. [Online]. https://robonation.org/app/uploads/sites/4/2025/04/2025-RoboSub_Team-Handbook-04_25_25.pdf

[4]"Wayfinder DVL," *Teledynemarine.com*, 2025. https://www.teledynemarine.com/wayfinder (accessed Jun. 23, 2025).

VIII. Appendices

Appendix A: Bill of Materials

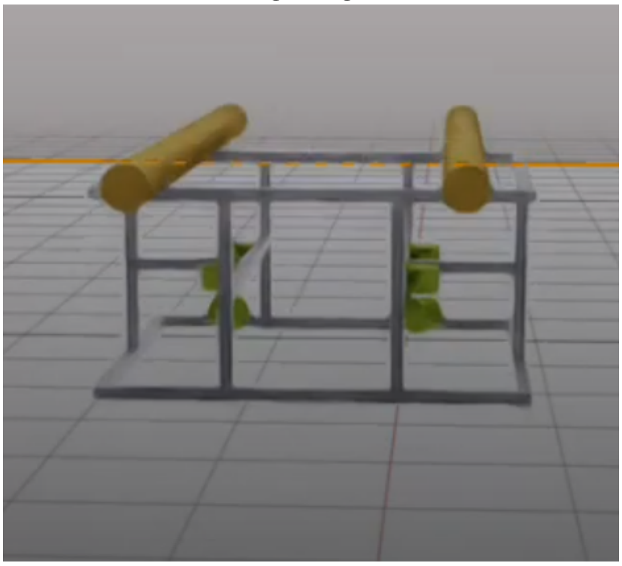
Item	Vendor	Cost	Quantity	Total Cost	Purchase Year
ArgoPelago's PVC Frame	Home Depot	$145.05	1	$145.05	2024
6" Diameter Acrylic Enclosure	Eplasics	$242.00	1	$242.00	2025
6" Flange water seal kit	Blue Robotic	$654.00	1	$654.00	2025
Waterproof Connectors	Blue Robotics	$10.80	10	$108	2025
Directional Thrusters	U1	$49.98	6	$299.88	2025
ESCs	Diamond Hobby	$21.89	6	$131.34	2025
Claw	HB	Custom	1	$180	2025
Torpedo System	HB	Custom	2		2025
Markers	HB	Custom	2		2025
Wayfinder Doppler Velocity Log	Teledyne	$6612.50	1	$6612.50	2025
Pressure Sensor	Blue Robotics	$75.00	1	$75.00	2025
Leak Sensor	Blue Robotics	$32.00	1	$32.00	2024

Low-Light USB Camera	Blue Robotics	$99.00	1	$99.00	2024
Bottom of Sub Camera	GoPro	Donated	1	Donated	2024
Liperior 22000mAh 4S 12C 14.8V Lipo Battery	Rcbatteries	$205.99	2	$411.98	2024
Hydrophones	Aquarian Hydrophones	$413	3	$1239	2025
din rail kit	International Connector Store	$35.99	1	$35.99	2025
buck convertor	Klnuoxj	$9.99	1	$9.99	2025
Power Module Board	Holybro	$60.00	2	$120.00	2025
Jetson Orin Nano	Nvidia	Donated	1	Donated	2024
Pixhawk 4 Flight Controller	Holybro	Donated	1	Donated	2024
Vision	HB	Training: YOLOv11		Custom	2025
Localization	HB			Custom	2025
Autonomy	HB	ROS2		Custom	2025
Team Size			Tracked: 11 Active: 5		
Mechanical Team Size:			Tracked:33 Active: 2		
Electrical Team Size:			Tracked: 3 Active: 1		
Software Team Size:			Tracked:54 Active: 2		
Simulation Test Time			~30 hrs		
In Water Test Time			~10 hrs		
Programming Languages			Python		

Appendix B: Testing Plans and Results

B1. Simulation Testing

- Scope
    - ArgoPelago can successfully move steadily vertically and horizontally in a virtual environment
    - Determine Successful thruster configuration
- Resources and tools
    - Google Cloud Services or GPU
    - IsaacSim installation
    - Computer or Laptop
    - Virtual ArgoPelago
- Environment
    - Tested in a plain non-water simulation stage to get an initial understanding of movement



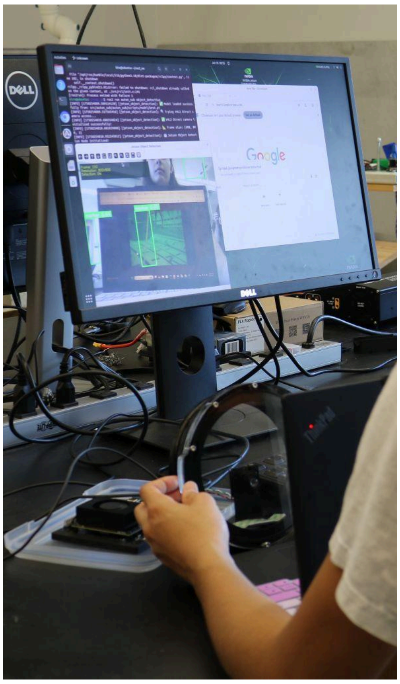
- Test Instructions
    - Start a Cloud server instance
    - Open IsaacSim through the terminal
    - Open the custom digital twin of ArgoPelago's frame
    - Systematically move the thrusters two at a time in 2 inch intervals to different placements
    - Apply 100 N of force to each of the vertical thrusters
    - Test the system
    - Repeat steps 4-6 until the system can stability elevate and descend smoothly
    - Repeat steps 4-7 with the horizontal thrusters
    - Note the stable system and observations
- Risk Management
    - Thruster damage
        - Performed this simulation to reduce damage to the thrusters and frame by reducing the amount of times the thrusters must be unmounted and remounted
- Results
    - The simulation could stably move vertically and horizontally - pass

B2. Image Detection - Dry Testing

- Scope
    - The image detection can successfully run on the Jetson
- Resources and tools
    - YOLOv11
    - OpenCV

- Python
- ROS2
- Blue Robotics Low Light Camera
- Jetson Orin Nano
- Monitor
- Keyboard
- Mouse
- HDMI-DisplayPort Cable
- Power Cable
- Underwater video
- Laptop
- custom-trained best.pt file

- Environment
  - The custom image detection model was tested on land

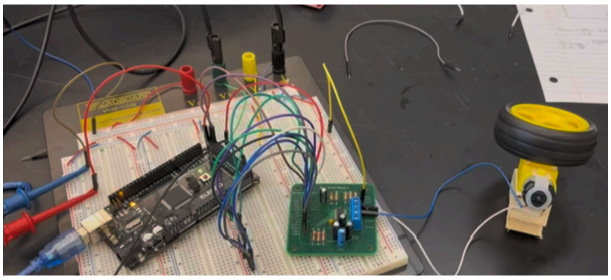


- Test Instructions
  - Connect the Jetson to a monitor via the HDMI-displayport cable on dry land
  - Plug in the mouse and keyboard
  - Power on the Jetson by plugging it into the power cord
  - Plug in the Camera to the Jetson
  - Navigate to the ROS2 directory
  - Source the ROS2 package
  - Run the Object Detection File
  - Place a laptop with a previously filmed underwater video of the course in front of the camera
  - Monitor the terminal for errors and make a note
  - Monitor the camera feed running the best.pt file and note whether the boxes are drawn around the desirable items
- Risk Management
  - Jetson Damage
    - Position the Jetson on a stable surface and a nonconductive material to prevent static effects or drops.
- Results

- The image detection model could successfully identify desired objects in a live video stream - pass

B3. Claw testing

- Scope
  - Claw is able to detect when it has gripped an object.
- Resources and tools
  - Claw
  - MAX22203 Motor Controller
  - Arduino
  - C Programming Language
  - 5V 3A Power Supply
  - Bread Board
  - Dupont Wires
  - Aluminum Beam
- Environment
  - Team's Lab

- Test Instructions
  - Connect Claw to motor controller
  - Connect motor controller to arduino
  - Supply 5V to claw motor controller
  - Open claw
  - Close claw on beam
- Risk Management
  - Limit power supply to 3A to prevent burning electronics and breaking components
- Results
  - Claw does not break - pass
  - Aluminum beam is securely held an can be picked up - pass

B4. Component Testing with Jetson

- Scope
  - Each component successfully responds when connected to the Jetson
- Resources and tools
  - Jetson Orin Nano
  - Leak Sensor
  - Python

  - ROS2
  - Blue Robotics Low Light Camera
  - Pixhawk
  - Monitor
  - Keyboard
  - Mouse
  - HDMI-DisplayPort Cable
  - Power Cable
  - Thrusters
  - Batteries
  - Discord
- Environment
  - The system was tested in a lab
- Test Instructions
  - Connect the Jetson to a monitor via the HDMI-displayport cable on dry land
  - Plug in the mouse and keyboard
  - Power on the Jetson by plugging it into the power cord
  - Plug in each individual component to the Jetson
  - Install needed software
  - Test by running a simple python script associated with that component
  - Navigate to the ROS2 directory
  - Source the ROS2 package
  - Run the ROS2 python node associated with that component
  - Monitor the terminal for errors and make a note
  - Repeat this process for each of the smaller components
  - HolyBro power distribution board
- Risk Management
  - Jetson Damage
    - Position the Jetson on a stable surface and a nonconductive material to prevent static effects or drops.
  - Water Damage
    - When testing the leak sensor ensure that the water does not spill and come in contact with the Jetson or any other electrical components
- Results
  - The leak sensor accurately detected a leak when placed in contact with water - pass
  - The camera successfully showed a live stream when joining a video call - pass
  - The Pixhawk successfully connected with the Jetson - pass
  - The thrusters successfully received commands from the Jetson when connected in conjunction with the Pixhawk, power distribution board, and battery

<u>B5. In-Water Testing of System</u>

- Scope
  - ArgoPelago can operate successfully underwater when controlled by a Raspberry Pi without leaking
- Resources and tools
  - ArgoPelago
  - Laptop Stand
  - Blue Robotics Low Light Camera
  - Raspberry Pi
  - Laptop with Ethernet Port
  - BlueOS
  - Power Cable

  - PS5 Controller
  - 2 Telemetry boards
  - Ethernet Cable
  - Pool
- Environment
  - Pool
  - 
- Test Instructions
  - Connect the telemetry boards to Ethernet cable and plug into laptop
  - Plug in the Battery and turn on the sub
  - Open BlueOS
  - Verify the camera is working and clean the camera lens if blurry
  - Inspect the O-Rings for dust and clean/relubricate if needed
  - Press both ends of the electrical tube on tightly and tighten down the pressure plug
  - Submerge the sub and check for leaks
  - If no leaks present, drive the sub to the bottom of the 14 foot pool and back up
  - Check for leaks and take note if there are any
- Risk Management
  - Water Damage
    - Ensure the O-Rings are compressing correctly and no leaks are present
- Results
  - The sub should not take on water
  - The camera should have a clear video stream - pass