

Nole Navigators Technical Design Report: RoboBoat 2026

Cesar Narvaez-Suarez, Katrina Le, Isaac Presley, Kamryn Gilbreath, Vinny Ivers, Jemma Smith, John Moore, Marla Rinehart, Aaron Presley
Florida State University, Panama City, Florida, United States of America

Abstract—This report summarizes the design strategy, competition plans, and future goals for the Nole Navigators RoboBoat Team of Florida State University Panama City (FSU-PC) and our vehicle, Bluey. This is our team’s first year participating in RoboBoat since being reorganized in 2025, therefore our vehicle for this year is a completely new design. We originally built off of a Blue Robotics BlueBoat platform, but have added several critical systems to the original kit to create a capable, competition-ready vehicle. Some of the most notable improvements we have made include an emergency stop system and additional, more powerful computing for vision workloads. By the beginning of 2026, we have successfully tested and operated our vehicle in real-world conditions using GPS waypoint navigation while continued development focuses on advancing its autonomous decision-making capabilities.



Figure 1: Our Team’s Vehicle

ACRONYMS AND ABBREVIATIONS

ASV Autonomous Surface Vehicle
 ESC Electronic Speed Controller
 RF Radio Frequency
 FSU-PC Florida State University Panama City
 ESTOP Emergency Stop
 IMU Inertial Measurement Unit
 SBC Single-Board Computer
 LDPE Low-Density Polyethylene

I. COMPETITION STRATEGY AND GOALS

This section of our report will discuss our team’s goals for the RoboBoat 2026 competition holistically and then break them down for each individual challenge on the course. We will also discuss our plans to adapt our boat for future competitions and detail how we intend to build upon what we have accomplished this year.

A. General Strategy

Since this is FSU-PC’s first year competing in RoboBoat, our team is primarily focused on familiarizing ourselves with the competition itself and networking with other teams to build connections and become involved in the RoboBoat community. However, we have still managed to build an Autonomous Surface Vehicle (ASV) with essential navigation capabilities using GPS waypoints within the operating system and QGroundControl mission control software framework.

B. Course Strategy

Since our vehicle is still in development, this year we will be focused on completing the simpler

navigation-based challenges, which from our assessment, consist of tasks one, two, and three. Our course strategy involves gaining an understanding of the basic autonomy required for us to complete the beginner challenges, ensuring that we meet the basic safety level requirements to get our boat approved for the courses. Additionally, mastering these tasks will also give us a firm foundation that we can build off of to tackle more complicated tasks in future competitions.

Task 1: Evacuation Route & Return

We plan to complete Task One by setting course waypoints for our vehicle using our mission controller, QGroundControl. We aim to build the appropriate autonomy system so that the vessel can set its own waypoints and guide itself through the set of gates and buoys.

Task 2, 3: Debris Clearance, Emergency Response

Our approach to Tasks Two and Three this year will be similar to Task One, utilizing waypoints set manually in QGroundControl to maneuver the ASV through the debris field and complete the speed challenge. However, once we enhance our boat's autonomy capabilities, we will be able to utilize our Luxonis OAK-D Wide camera's object detection algorithms to identify and avoid debris for Task Two and interpret the color indication buoys for Task Three.

C. Future Competition Plans

Our future competition plans involve improving team strategy, fostering a stronger team community, and building on to current and new capabilities to strengthen our vessel into a more robust, sophisticated platform. We aim to improve on each system of the boat, with an emphasis on the computer and software systems. Ultimately, we aim to improve our vessel so that we exceed the qualifying tasks in the next competition.

II. DESIGN STRATEGY

This section of our report will focus on the strategy behind our ASV's design and manufacture, including hardware and software, focusing especially on modifications we made to the original BlueBoat platform.

A. Mechanical Structure

The mechanical structure of our boat has remained relatively close to its original BlueBoat configuration, using a catamaran design consisting of two low density polyethylene (LDPE) hulls connected by anodized aluminum crossbars, with each hull having a removable, water-resistant cover for access to the electronics bays. However, some modifications have been made by adding 3D-printed components we designed for a custom camera mounting bracket and enclosures for the manual emergency stop (ESTOP) switches. The emergency stop enclosures were designed to be water resistant, utilizing Blue Robotics Wetlink cable penetrators to connect the switches to the ESTOP system to retain waterproofing without using any traditional sealants like epoxy or silicone. This allows for a modular system that can easily be disassembled, allowing access to internal components by simply removing bolts and screws while maintaining a high level of water ingress protection.

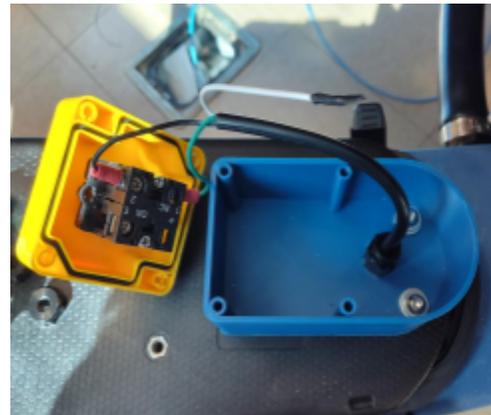


Figure 2: Early ESTOP Enclosure Prototype

B. Electrical Systems

Our BlueBoat kit came pre-equipped with a Navigator flight controller, Raspberry Pi 4 single board computer (SBC) to handle all navigation computing, a combined Inertial Measurement Unit (IMU) and GPS, wireless router, two Blue Robotics M200 thrusters with Blue Robotics Basic ESC 500s, an LED signal light, and power distribution bus units for 5V and 12V power. In addition to these components, we have added our own ESTOP system, an Nvidia Jetson Orin Nano computer for computer vision workloads, a RadioMaster ER8 RF receiver for emergency manual control of the ASV,

and a Luxonis OAK-D Wide camera for vision and navigation. We also replaced the original, shroudless M200 thrusters with T200 thrusters, which have an identical DC motor, but have a different propeller design which includes a propeller shroud to comply with RoboBoat's safety policies.

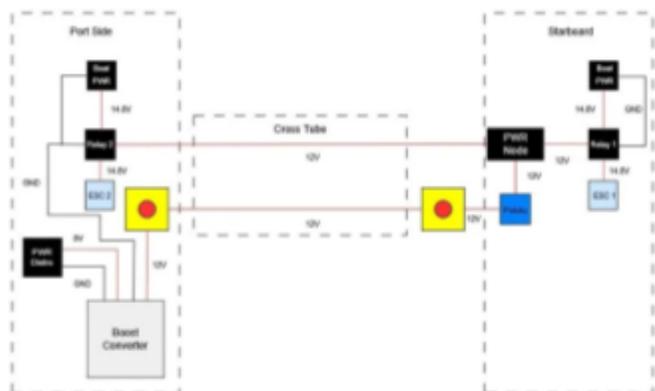


Figure 3: ESTOP System Diagram

Our boat's ESTOP system consists of two distinct circuits; the first of these is a series circuit with three different cutoff points: a manual latching pushbutton switch located on each side of the boat and a Pololu-branded relay (represented as "Pololu" in the diagram above) wired to the RF receiver for a remote killswitch. The second part of the ESTOP system is a parallel circuit with relays connecting to the ESC for each one of our T200 thrusters, which mechanically disconnects power from the thrusters when they are triggered by the first circuit being broken by any of the three triggers. The first circuit is powered using a 12V boost converter connected to the 5V output on the boat's original power distribution bus, while the second circuit uses the boat's full 14.8V to be able to actuate the larger relays to cut off thruster power.

C. Computer Systems/Software

The BlueRobotics BlueBoat kit is the foundation of our entire vessel, including the hardware and software components. In addition to the Raspberry Pi 4, the vessel came with an SD card preflashed with BlueOS, which we would still have to configure. BlueOS is BlueRobotics' open-source, modular operating system for controlling underwater vehicles and other robotics. [1] This specific operating system supports custom extensions and provides a modular cockpit for control. BlueOS is equipped with various tools,

aimed at running on an onboard computer to assist an underlying flight controller board that communicates sensor data with a control station computer.

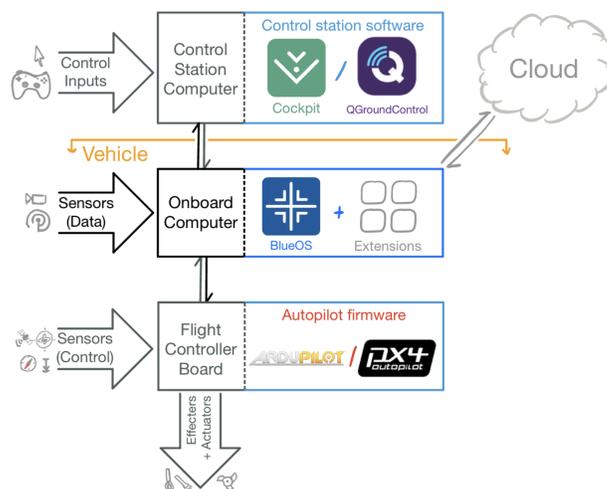


Figure 4: Overview of Components in BlueOS-related Software Stack [1]

We spent the majority of our time familiarizing ourselves with the BlueOS environment, exploring the increasing amount of new capabilities created by the open-source developers. Since the Raspberry Pi from the BlueBoat kit was already equipped with BlueOS for vehicle navigation and networking, we needed to find a way to integrate more sophisticated sensors and autonomy capabilities with the preexisting system.

Over the Fall 2025 semester, we decided to use a separate onboard computer designated for autonomous decision making and sensor integration in addition to the one already installed on the vessel. Our goal is to use the Jetson Nano Orin to process live video streams from the OAK-D Wide camera and send Python scripts to the Raspberry Pi Navigator in order to control its movement autonomously. We wanted to take on a simple approach before composing a more sophisticated autonomy system. By the time we arrive at the competition, we hope to have our boat create its own waypoints as it traverses through the course.

Utilizing two SBCs on our vessel has challenged our strategy approach with regards to the computer systems and software aspect. Using one computer to process data and send navigation instructions to the other is a task that makes our system more complex,

as compatibility issues arise between seamlessly connecting the two SBCs in one network. We are currently testing sending certain python instructions to coordinate with BlueOS and QGroundControl. We have researched MIT Marine Autonomy Lab's MOOS-IvP open-source C++ modules, which are specialized for autonomy in marine robotic platforms. [2] These modules combine the "Mission Oriented Operating Suite" (MOOS) middleware with "Interval Programming" (IvP) or behavior-based, collaborative decision-making. Further testing is required to determine whether implementing our own version of these modules would solve our difficulties.

III. TESTING STRATEGY

As this is our team's first year competing, we deemed it crucial that we test components and subsystems of our newly christened vessel, giving us a solid foundation for future competitions. Although we did not follow a strict testing plan, we strove to be thorough in our methods; we focused on quickly and effectively resolving any discrepancies or errors that appeared during testing/development.

Additionally, our team began implementing simulation testing, striving to set up a good system by which to use for future iterations and evolutions of our vessel and the tasks.

A. Real-World Testing

When we had opportunities to test our boat in the real world, we utilized our campus' access to Panama City's North Bay as a testing area, mainly focusing on assessing functionality with RF controls and in wireless mission control using GPS waypoints set in QGroundControl.



Figure 5: Testing GPS Waypoints, April 2025

We used Blue Robotics' BaseStation to establish a wireless connection with our ASV during early tests. The control system worked as follows: first, a laptop would be used to run QGroundControl and give instructions to the boat with GPS waypoints or a videogame controller; next, a USB-C connection would allow the instructions from QGroundControl to reach the BaseStation, which has a wireless router that sends instructions to the boat. However, we knew that this control configuration would have too many liabilities if it were to be relying on a WiFi connection for control of the boat, so after our early testing in February-April 2025, we began conceptualizing and designing our RF emergency control system.

B. Individual Part and System Testing

Unfortunately, the work we were performing on other aspects of the boat like the ESTOP system left our ASV in a disassembled and inoperable state for several months, preventing us from doing any further in-water testing for an extended period of time. Despite these challenges, during this period, we still managed to successfully conduct tests of our vehicle's individual systems as we developed them.

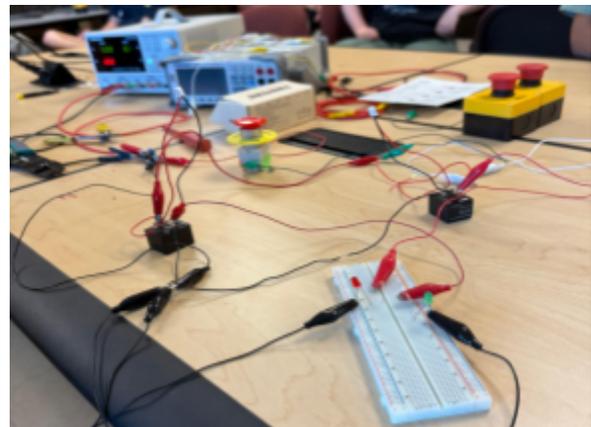


Figure 6: First Prototype of ESTOP System

To exemplify, we constructed several prototypes of our ESTOP system and tested them rigorously before completing its installation on the boat. The initial ESTOP prototype was built using breadboards and alligator clip wires, powered by a programmable power supply, and connected the relays to LED lights which stood in the place of the thrusters' ESCs as proof of concept. Once the system was refined, it was further tested for both

function and durability; we verified its water ingress resistance by spraying the ESTOP enclosures with water in varying intensities from several angles before installing the electronics. After every test, the inside of the enclosures remained completely dry. We also verified that the ESTOP was able to completely cut off power from each thruster by activating each thruster individually and terminating them with each of the three killswitches to ensure full functionality.

C. Simulation

In order to aid autonomy development and reduce reliance on physical testing, our team decided to use Gazebo as our simulation software of choice, as the same software is used by RoboNation's Virtual RobotX competition. [3] This simulation platform was selected due to the wide variety of resources available, including meshes for basic roboboat hulls and the actual testing site of the competition. [4]

Our team was able to spawn in a simulated environment of the Nathan Benderson Park and a preset RoboBoat model from the publicly available VRX repository. [5] Additionally, we were able to publish commands to the vessel and see it directly respond to those commands. This confirmed that vehicle spawning and command input function correctly, meaning that our simulation environment setup could support closed loop testing. The next stage of testing will include deploying our autonomy software in the environment in order to test algorithm performance.

ACKNOWLEDGMENTS

This project would have been impossible to complete without the advice, technical assistance, and sponsorship of our team's most valuable partners. The Nole Navigators RoboBoat team would like to thank:

Dr. Damion Dunlap, our primary technical advisor and mentor, who also donated the BlueBoat kit our vehicle was mostly comprised of, for inspiring us to participate in RoboBoat and helping us make it to Sarasota,

Dr. Twan Capehart, our faculty advisor and mentor, the Student Government Council of FSU-PC for their financial grant to cover our registration and lodging for the competition,

and our team members who made essential contributions to key aspects of our project, but were ultimately unable to join us in Sarasota: Kamryn, Angelica, Morgan, Nicole, Randy, Frank, Noah, and Carlos.

Finally, we would like to thank RoboNation for making this competition possible and for inspiring the next generation of STEM.

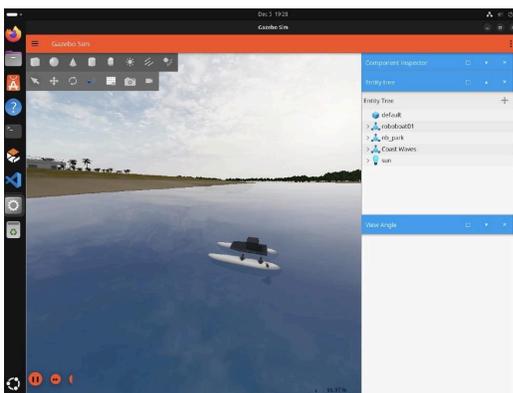


Figure 7: RoboBoat Model in Gazebo

References

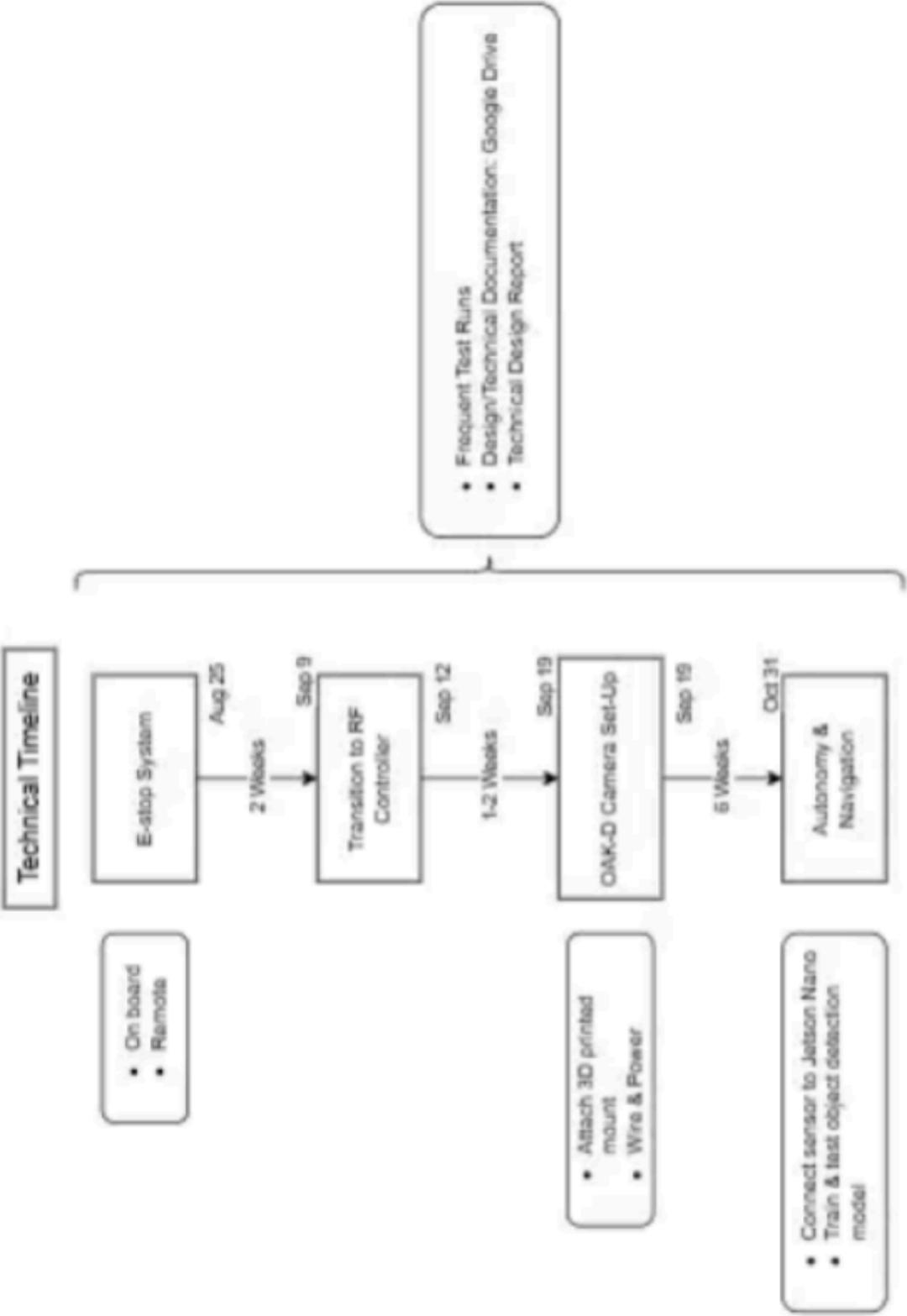
- [1] Blue Robotics, “BlueOS Documentation,” *Blueos.cloud*, 2026. <https://blueos.cloud/docs/stable/usage/overview/> (accessed Jan. 14, 2026).
- [2] MIT Marine Autonomy Lab, “MIT Marine Autonomy Lab : Main - Home Page browse,” *Mit.edu*, 2016. <https://oceanai.mit.edu/autonomylab/pmwiki/pmwiki.php> (accessed Jan. 14, 2026).
- [3] Robonation, “VRX Competition Technical Guide,” Apr. 2023. Accessed: Jan. 14, 2026. [Online]. Available: https://robonation.org/app/uploads/sites/2/2023/04/VRX2023_Technical-Guide_v1.0.pdf
- [4] Open Robotics, “running_robotboat_tutorial,” *GitHub*, Dec. 22, 2023. https://github.com/osrf/vrx/wiki/running_robotboat_tutorial (accessed Jan. 14, 2026).
- [5] Open Robotics, “Virtual Robotics Repository,” *GitHub*, Aug. 28, 2025. <https://github.com/osrf/vrx> (accessed Jan. 14, 2026).

APPENDIX 1: PARTS LIST

Vendor	Model/Type	Specs	Custom/ Purchased	Cost	Year of Purchase
ASV Hull Form/Platform	Blue Robotics BlueBoat Hulls	2x LDPE Hull	Purchased	Donated	2025
Waterproof Connectors	Blue Robotics Wetlink Cable Penetrators	Using several of each size: M10, M14	Purchased	Donated	2025
Propulsion	Blue Robotics T200 Thruster	Thrust: 11.6 / 9.0 lb f at full throttle 16V	Purchased	Donated	2025
Power System	Blue Robotics 14.8 V LiPo Batteries, BlueBoat Power Distribution Bus & Fuse Boards	Battery Voltage @ 60 A (Direct Connection) Battery Voltage @ 10 A (Fuse Board) 5V Auxiliary @ 5 A (Fuse Board)	Purchased	Donated ; 2 Extra Batteries Purchased at \$425.00 each	2025
Motor Controls	Blue Robotics Basic ESC 500	50 A Max Current	Purchased	Donated	2025
CPU	Raspberry Pi 4; Nvidia Jetson Orin Nano	2 GB Model w/ BlueOS; 8 GB Model w/ Ubuntu OS	Purchased	Donated	2025
Teleoperation	MikroTik RBGroove- 52HPn Router; RadioMaster ER8 Receiver; RadioMaster Boxer Transmitter; Blue Robotics Base Station	2.4 Ghz Wireless Router; 8-Channel ELRS RF PWM Receiver; ELRS RF Transmitter; 2.4 Ghz Wireless Router with USB-C Connection to Laptop	Purchased	Donated ; \$34.99; \$159.99; Donated	2025
Compass	Blue Robotics Navigator	Dual 3-DOF compasses integrated into Blue Robotics Navigator PCB	Purchased	Donated	2025
Inertial Measurement Unit (IMU)	Blue Robotics Navigator	6-DOF IMU integrated into Blue Robotics Navigator PCB	Purchased	Donated	2025

Doppler Velocity Logger (DVL)	N/A	N/A	N/A	N/A	N/A
Camera(s)	Luxonis OAK-D Wide	AI Processing Chip: Intel Movidius Myriad X VPU (RVC2), 150° DFOV for stereo pair, 12 MP RGB camera module with 120° DFOV	Purchased	Donated	2025
Hydrophones	N/A	N/A	N/A	N/A	N/A
Algorithms	N/A	N/A	N/A	N/A	N/A
Vision	Luxonis DepthAI	Open-source software API with support for Python and C++ development languages	Purchased	Free to Use	2025
Localization and Mapping	N/A	N/A	N/A	N/A	N/A
Autonomy	QGroundControl	Mission Control Planner	Purchased	Free to Use	2025
Open Source Software	BlueOS	Open-Source operating system for BlueBoat's Raspberry Pi 4	Purchased	Free to Use	2025

APPENDIX 2: 2025-2026 Team Goals Timelines- Technical



APPENDIX 2: 2025-2026 Team Goals Timelines- Logistical

