

Aquamods

C3: Alan Abraham, John Pedraza, Angela Xu

18-500 Capstone Design, Spring 2025

Electrical and Computer Engineering Department
Carnegie Mellon University

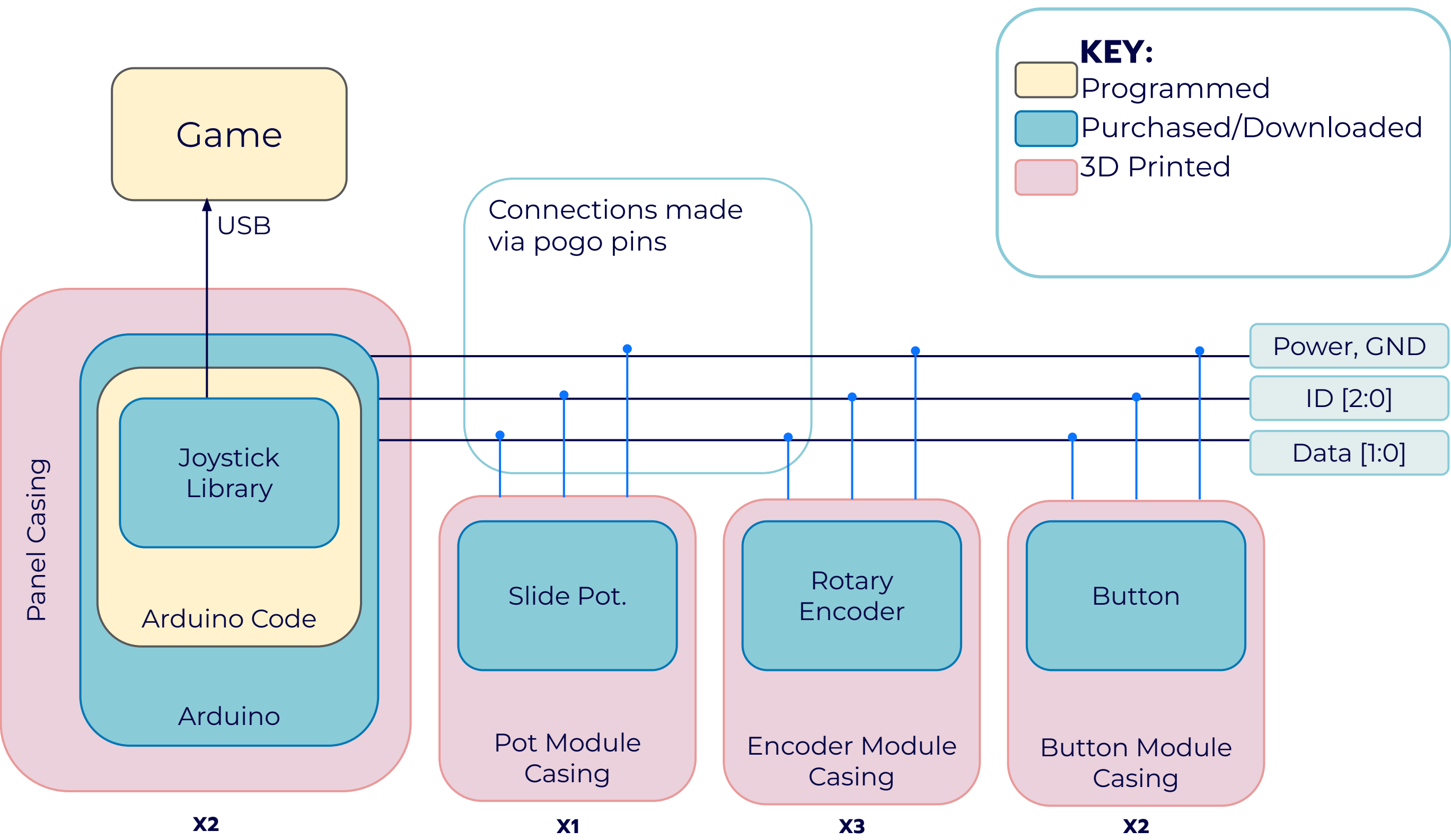
Product Pitch

Aquamods is a chaotic and cooperative video game where two players have to control a submarine to defend against fish while trying to maximize their score. The twist is that each player can only control two of the six submarine functions at a time and will need to share these and effectively communicate in order to survive.

This sharing feature is implemented through a custom hardware controller consisting of six modules that can be interchangeably inserted into slots to operate the ship.

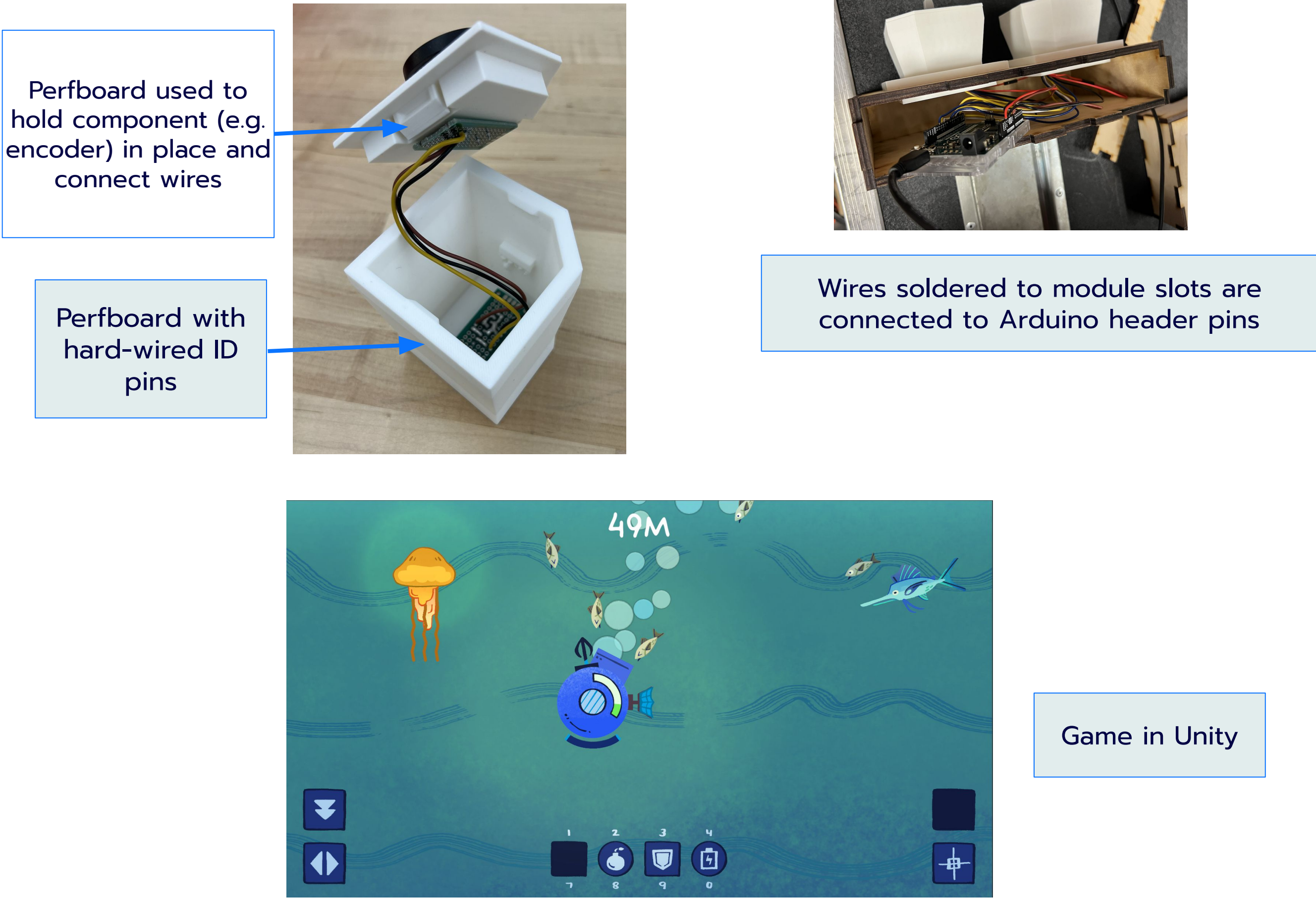
System Architecture

Include figure(s) that convey your system architecture along with an explanatory description. Please make sure this is an updated architecture, e.g. you can have 2 block diagrams, one for HW and one for SW. (or a state or flow diagram for SW if that is more descriptive).



System Description

Each of the six modules houses an electronic peripheral wired through perfboard(s) to a pogo pin connector. The controller has two pogo pin equipped slots to fit a module, and houses an Arduino connected to these pogo pins. The Arduino can read voltages from the modules to identify them and their data. The Arduino communicates with the laptop via a usb connection using a joystick library. The laptop hosts the software for the game which was created in Unity.



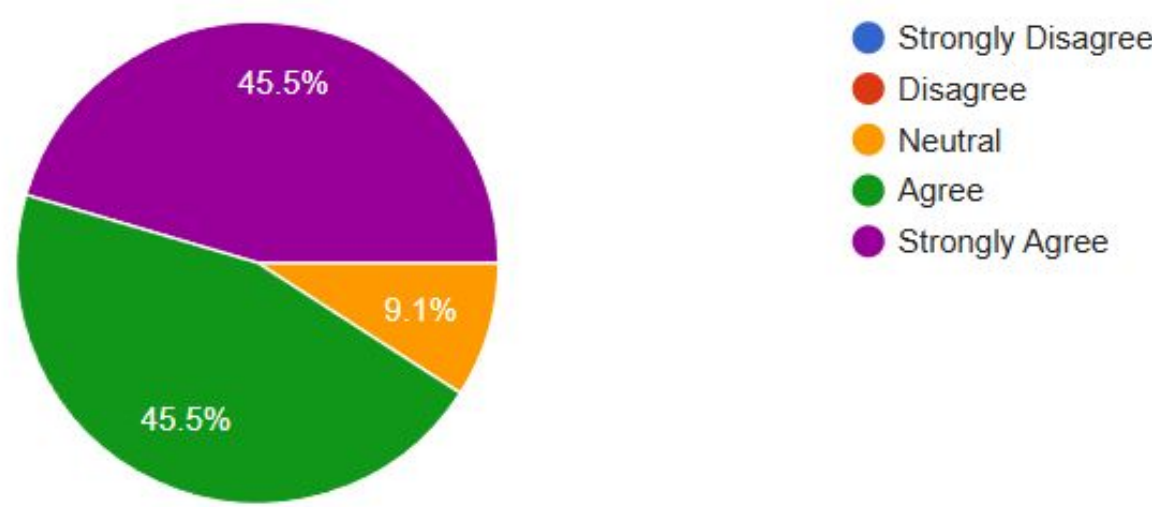
System Evaluation

For the tests regarding the latency and timing of using the controller, we incorporated data collection code in the Arduino to record the appropriate data. For playtesting we visited the Hunt playtesting night to get feedback from other players.

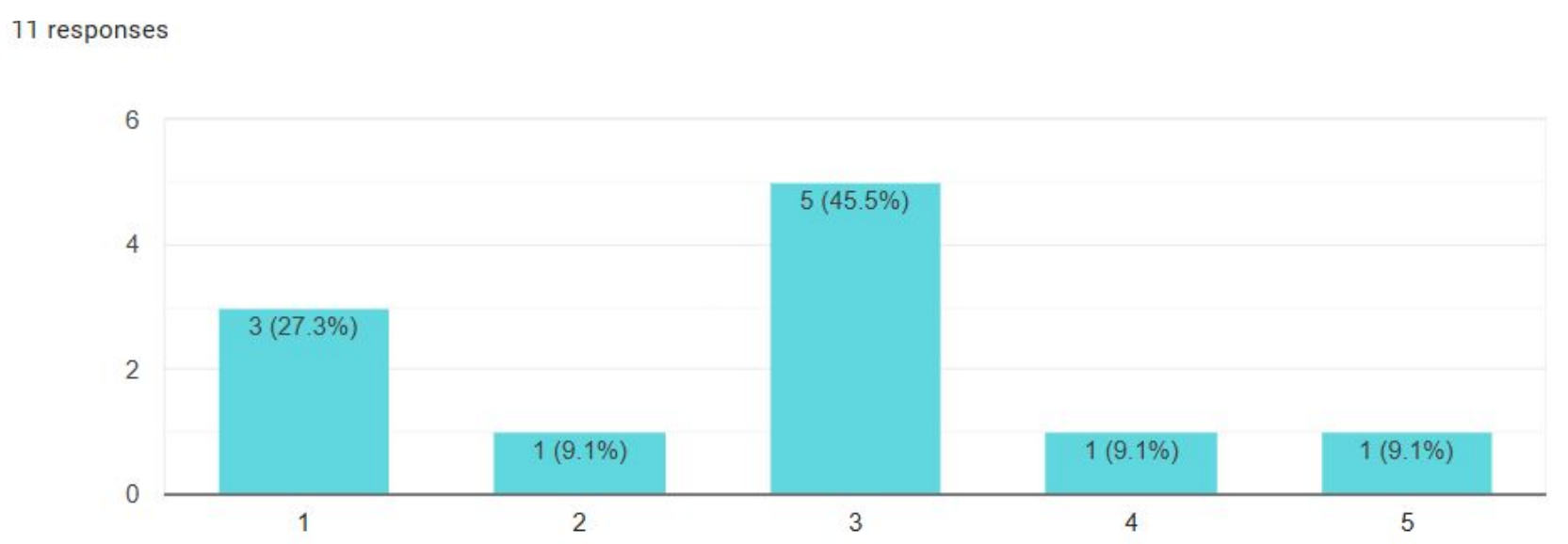
Use-Case Requirements:

Metric	Target	Actual
Weight	< 3.5 lbs	3.11 lbs
Panel Size	< 5.5 x 9.5 x 2.5 in	5.47 x 9.47 x 2 in
Cost	< \$150	\$116.87
Controller Latency	< 30 ms	1.9-5.6 ms
Ease of Swapping	< 3 s average	1.5 s average
Fun	> 80% positive results	Results in graph

Switching between different modules felt natural



How was the difficulty?



Conclusions & Additional Information

Our overall system met the requirements we set out to meet. Our Arduino code is a bit messy and could be cleaned up a bit, but it does perform its function. Many playtesters spoke about adding more modules or more enemies, we think this would be a good addition for the future. Some lessons we learned are to sometimes just try something to see if it will work out, get end to end working as fast as possible, and leave enough time for debugging for unforeseen problems.



<https://course.ece.cmu.edu/~ece500/projects/s25-teamc3/>