# PongPal

**Team A8** Project Proposal Presentation

Michael Bahner, Alex Kireeff, Seung Yun Lee

# Use Case

**Problem**: People might not be able to play water pong for different reasons like sickness, lack of proximity, or disabilities.

**Solution**: PongPal™ makes water pong more accessible by allowing one to remotely control the robot and get feedback.

**ECE Areas**: Software Systems, Signals and Systems

# Use Case Requirements: Accessibility

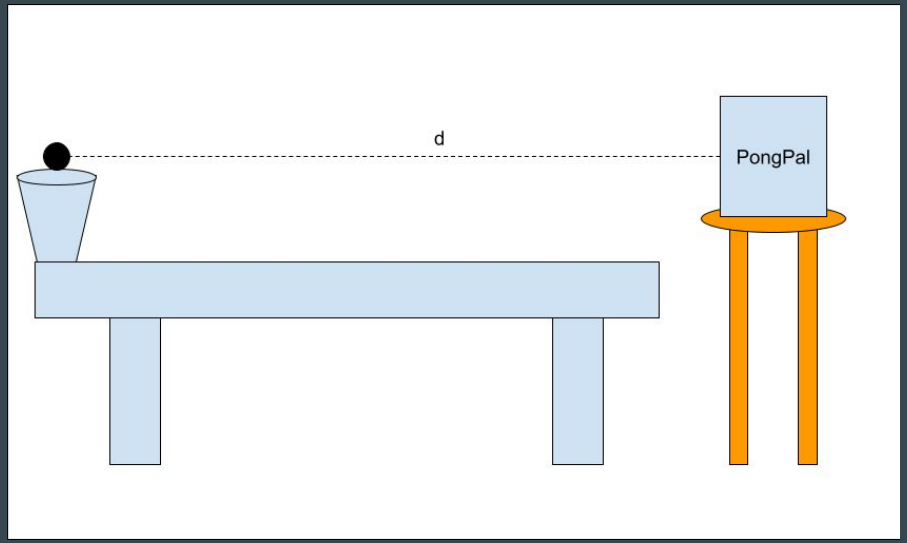
The user should be able to easily access and set up the robot so the experience is seamless and enjoyable as possible.

- Portable
  - Make the robot light (< 5 kg) with a small form factor (40cm * 40cm)
- Intuitive User Interface
  - Display where cups are on the table and where the ball has landed previously
  - Intuitive radial and azimuth controls for the barrel as well as firing controls
- Responsive
  - 100ms latency between the input and action
  - 5s latency from shot landing to being displayed to the user
  - Need to balance responsiveness with portability

# Use Case Requirements: Reliability

The user should experience minimal noise under the same robot settings to maximize the skill aspect of the game.

- Consistent Trajectory
  - **5 cm depth variance** when the ball is shot with the same power settings
  - **2 cm horizontal variance** when the ball is shot without reaiming

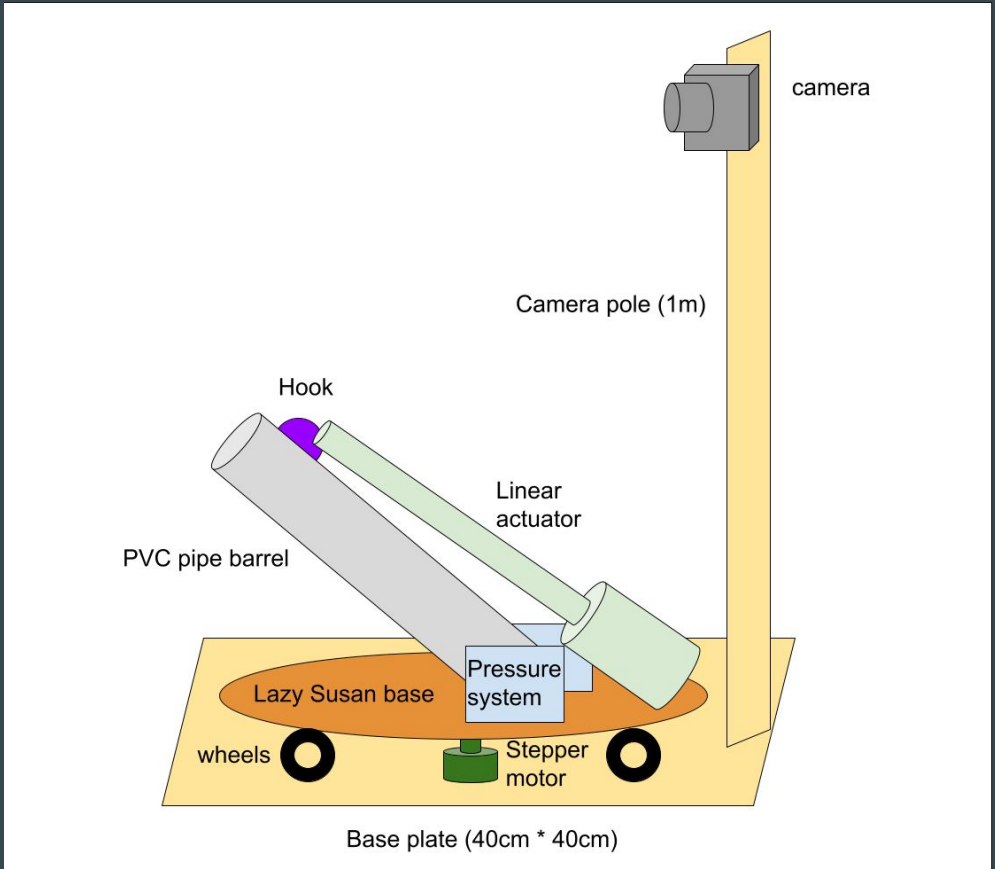



d is the distance from the front edge of the robot to the cup **(1.5m to 3m)**

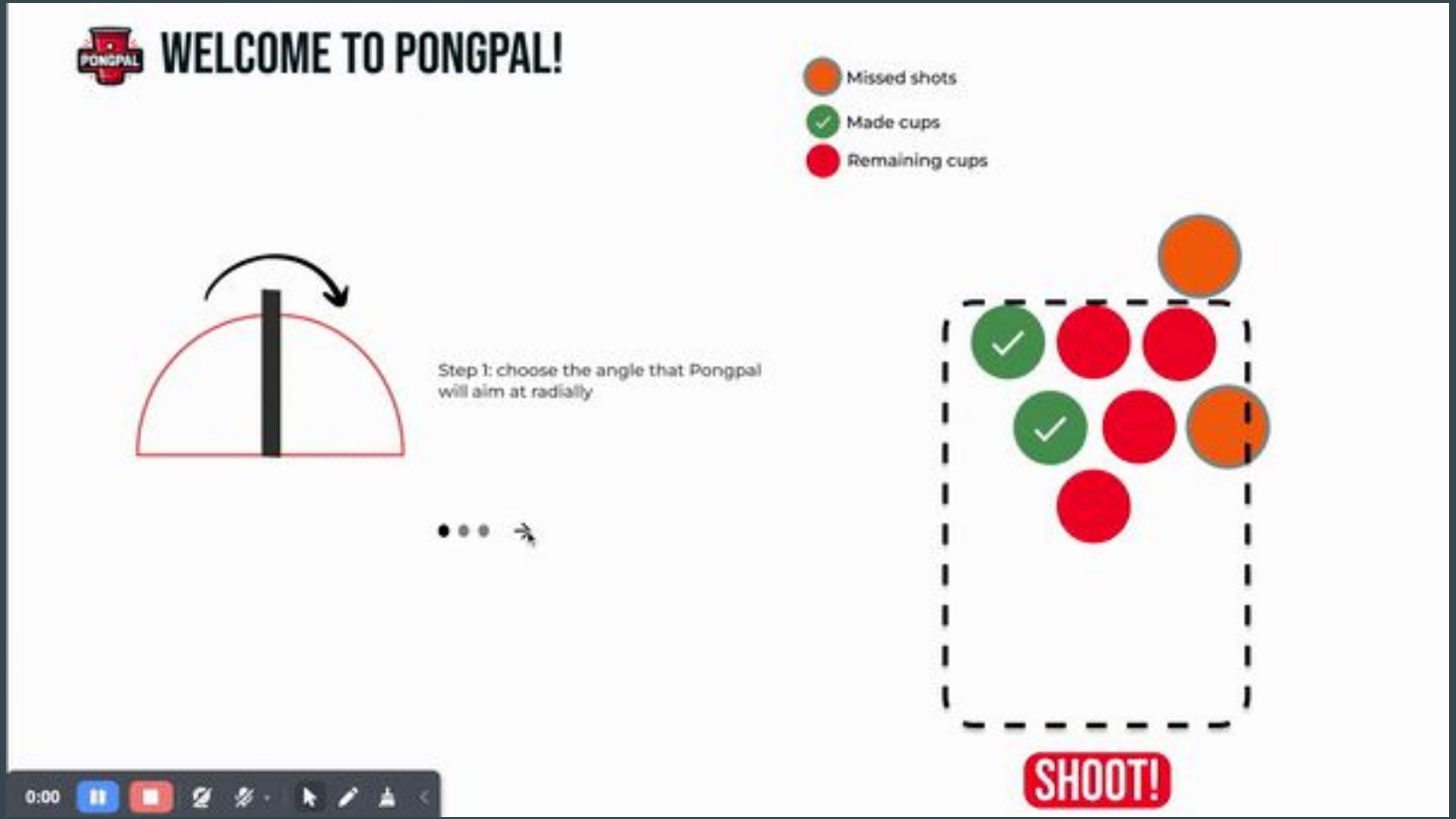
# Technical Challenges

- Firing Consistency
  - Minimize the variance of all physical components
- Detection Consistency
  - Maximize correct cup detections while minimizing incorrect detections in low/variable light conditions
- Cup Localization Accuracy
  - Minimize jitter of the cup locations in low/variable light conditions
  - Accurately determine where ball landed
- Interaction Latency
  - User needs to control the robot in real time
- Portable
  - Robot needs to be easy to move

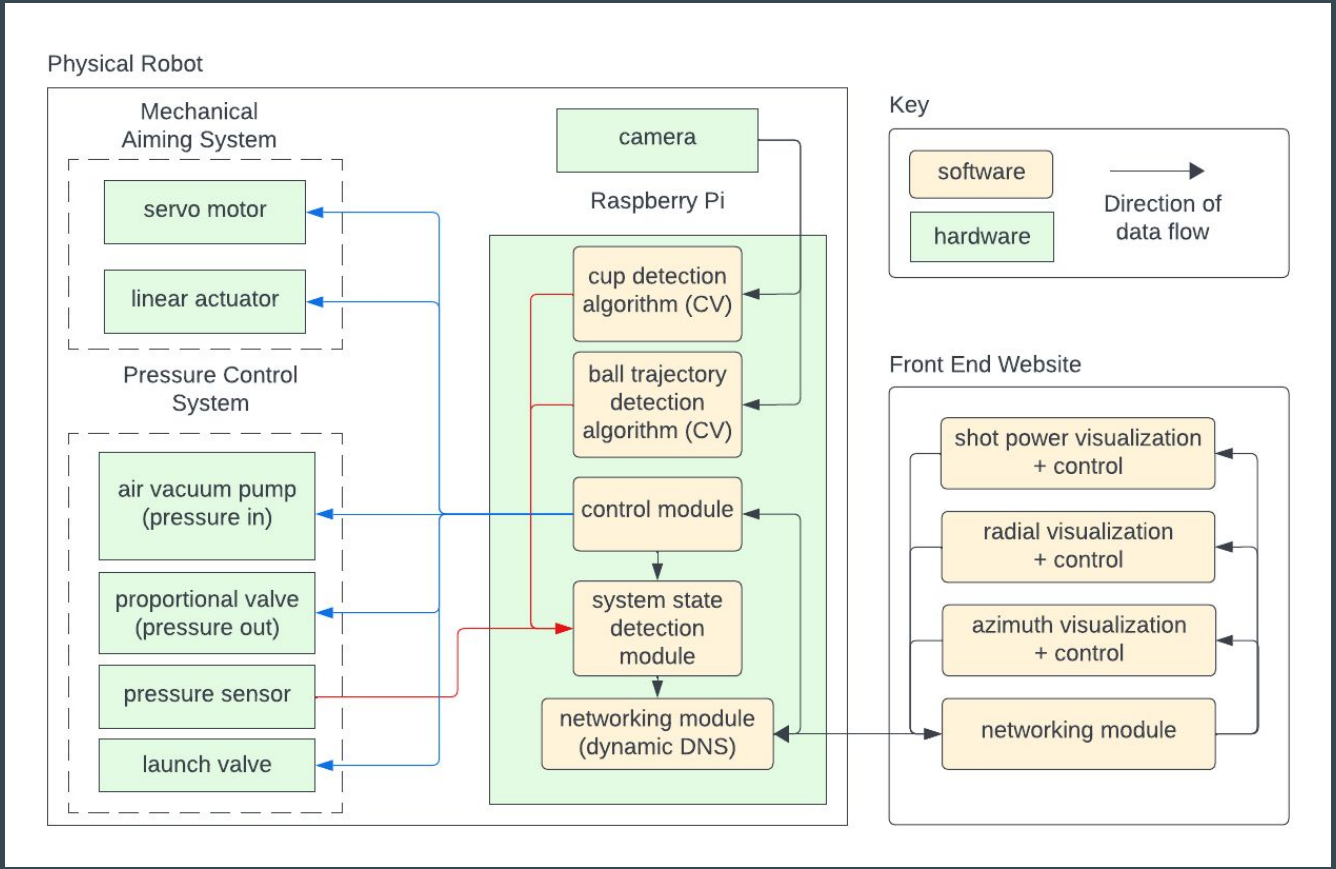
# Solution Design



# Solution Design



# Solution Block Diagram

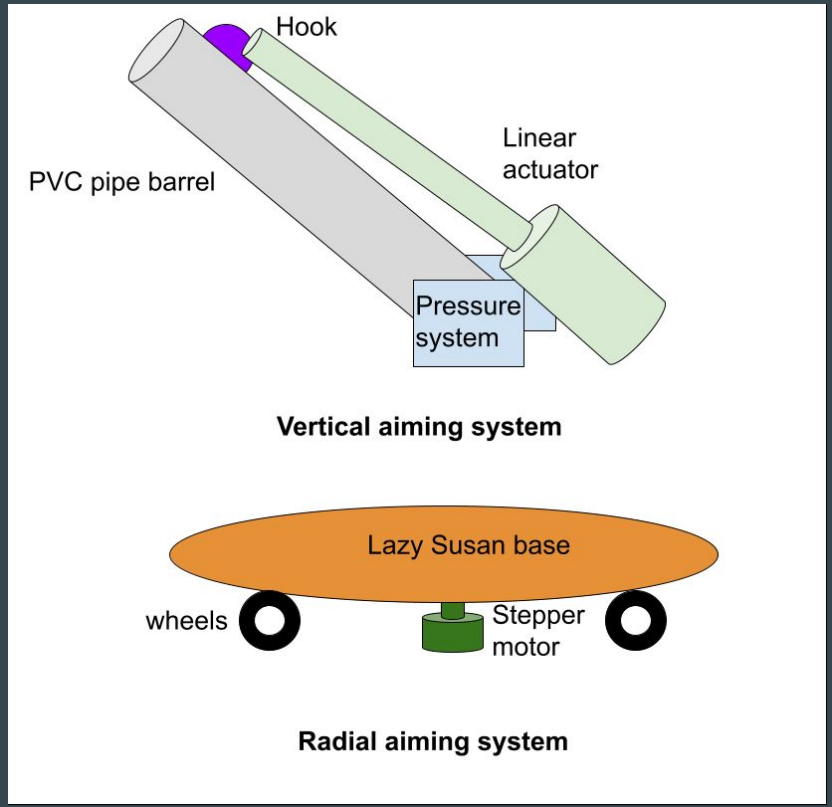


# Testing, Verification, and Metrics

| Requirements | Testing | Metrics |
|---|---|---|
| **Accuracy**<br>minimal noise under the same robot settings | Shoot 10 shots under the same setting, and record where the ball landed on the table (once from 1.5m away, once from 3m away) | Maximal bounding box that contains all 10 shots < 5cm * 2cm |
| **Responsiveness**<br>real-time feedback loop for user | Perform 10 user inputs, measure the time it takes from input to robot's movement (latency A)<br><br>Perform 10 shots, measure the time it takes from pressing the button to result being displayed (latency B) | Latency A average < 100ms<br><br>Latency B average < 5s |
| **Accessibility**<br>intuitive UI for users to control the robot with minimal guidance | Perform user testing by having a randomly selected individual to control the robot using the UI | 90% of the users can play the whole game without guidance |

# Tasks and Division of Labor

- Aiming Subsystem (Simon)
  - Vertical Aiming
    - Linear actuator with stepper motor to move barrel up and down
  - Radial Aiming
    - Lazy Susan mechanism that is controlled by a stepper motor



# Tasks and Division of Labor

- Launching Subsystem (Alex)
  - Pressure Chamber
    - Increase/decrease pressure in chamber using vacuum pump/proportional valve
- Detection Subsystem (Alex)
  - Use computer vision to find cup position (relatively static) and where the ball lands (analyze video)
- Device Interface Subsystem (Mike)
  - Interactive Website to control the PongPal
  - Visualization for current game state
- Networking Modules (Mike)
  - Server hosted on RPi with dynamic DNS configuration
  - Reliable communication between the web client and the RPi


February 2024
March 2024
April 2024
4
11
18
25
3
10
17
24
31
7
14
21
Website initial setup • Feb 1 - Feb 3
Project proposal • Jan 31 - Feb 4
Design presentation slides • Feb 12 - Feb 18
Design document • Feb 19 - Feb 29
Ethics assignment • Mar 13 - Mar 13
Interim demonstration • Apr 1 - Apr 1
Final presentation slides • Apr 15 - Apr 21
Finalize Design • Feb 4 - Feb 10
Sanity Check Design • Feb 11 - Feb 17
Ordering parts • Feb 7 - Mar 20
Spring Break • Mar 2 - Mar 10
Integration • Apr 1 - Apr 8
Integration testing • Apr 8 - Apr 15
Construct base, barrel, and lazy susan plate (LSP) • Feb 18 - Feb 23
Integrate Vertical Aiming Subsystem and attach to microcontroller • Feb 23 - Mar 1
Write software to control barrel attitude • Mar 16 - Mar 23
Attach stepper motor to base, LSP and microcontroller • Mar 8 - Mar 15
Write software to control barrel azimuth • Mar 16 - Mar 23
Attach pressure chamber to barrel • Feb 19 - Feb 23
Attach air pump to chamber • Feb 23 - Feb 26
Attach variable valve to chamber for pressure release • Feb 26 - Feb 27
Attach solenoid valve between pressure chamber and barrel for firing • Feb 27 - Feb 28
Attach air pump to microntroller • Feb 28 - Feb 29
Attach variable valve to microcontroller • Feb 29 - Mar 1
Attach solenoid valve to microcontroller • Mar 1 - Mar 4
Write software to control launching subsystem • Mar 4 - Mar 8
Cup detection/localization in well lit environment • Feb 19 - Feb 23
Cup detection/localization in poorly lit environment • Feb 26 - Mar 1
Cup detection/localization in variably lit environment • Mar 4 - Mar 8
Record ball after shot • Mar 11 - Mar 15
Ball frame-by-frame detection/localization • Mar 18 - Mar 22
Obtain ball landing location • Mar 25 - Mar 29
Set up microcontroller operating system • Feb 19 - Feb 21
set up microcontroller camera • Feb 21 - Feb 23
Frontend design • Feb 19 - Feb 23
Frontend prototype • Feb 23 - Mar 1
Frontend user testing • Mar 9 - Mar 12
Backend API design and review • Feb 19 - Feb 23
Backend API implementation • Feb 23 - Mar 1
set up DDNS • Mar 11 - Mar 15