Reservoir High School | Maryland | Cheren Song, Jonathan Ye, Dominic Ye, Ife Adeleye | Sponsor: Katherine Yienger

Game Changer: Engineering Adaptive Controllers for Barrier-Free Play

Designing an accessible gaming controller to provide a level playing field for individuals with limited dexterity

Primary Objective

- Design an **equitable** video game controller that individuals with **various kinds/cases of limited dexterity** can use conveniently

Secondary Objectives

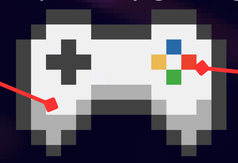
- Reduce **financial barriers** to accessible technology
- Make the controller **as customizable as possible**

Problem Statement

Individuals with limited dexterity struggle using traditional video game controllers and lack access to alternatives to experience gaming's benefits, undermining UN SDGs #10 (Reduced Inequalities) and #3 (Good Health & Well-being)

User Survey

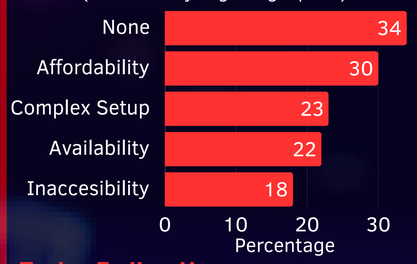
Traditional (inequitable) gaming controller:



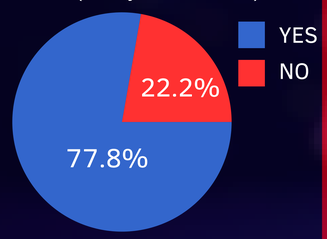
difficult-to-grip shape for those with limited dexterity

small, close-clustered buttons (hard to navigate)

Barriers faced by gamers with disabilities (select all that apply)
("Accessibility in gaming report")



Have you or someone you know avoided or stopped gaming due to difficulty using the controller?
(Survey we conducted)



Data Collection:

- Interviewed 3 **adaptive technology professionals** (Mr. Darrell Hilliker, Mr. Kyle Ott, Mr. Marshall Tumperi)
- **Surveyed and interviewed** local schools' special education staff, students, and other community members in our user base
- Referenced **surveys conducted by other organizations**

Explicit Requirements:

1. **Affordability:** must cost less to make than the Xbox Adaptive and Playstation Access controllers (<$70)
2. **Customizability:** interchangeable buttons and controls to play various games, code a user-friendly app to personalize button-key mappings and joystick sensitivities
3. **Accessibility:** larger controller and buttons with optimized arrangement for minimal hand movement

Implicit Requirements:

1. **Durability:** strong buttons, joystick, and shell
2. **Consistent Performance:** no lag or delay during use

Citations

RHS special education department. (2024, November 15). [Personal interview by RHS NEDC team].
Hazel, J., Kim, H. M., & Every-Palmer, S. (2022). Exploring the possible mental health and wellbeing benefits of video games for adult players: A cross-sectional study. *Australasian Psychiatry*, 30(4), 541-546.
Accessibility in gaming report. (n.d.). Scope. https://www.scope.org.uk/campaigns/research-policy/accessibility-in-gaming

Prototype

PLAY

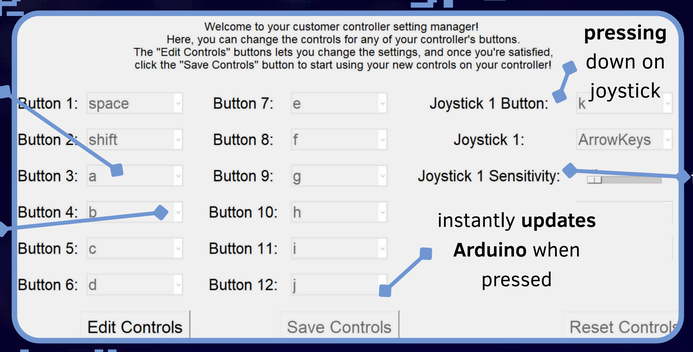
App

button # on controller (below)

current **button mapping**

pressing down on joystick

adjustable for varied tremor and types of games



instantly **updates Arduino** when pressed

Controller

*various button **designs and textures** (arrow keys, LEGO block, shapes)

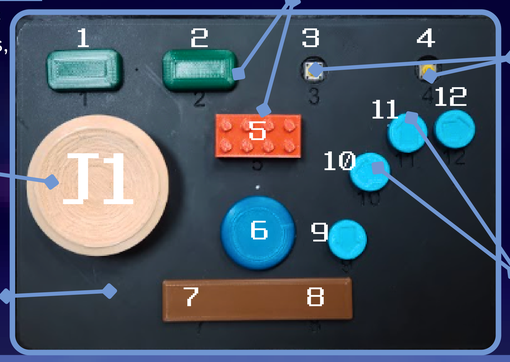
screw-on joystick

Arduino **Leonardo** with jumper wires **linked to buttons/joystick** (inside)

spread-out buttons for different hotkeys

switchable button caps connect to Arduino SMD buttons, **soldered** to PCBs (inside)

buttons **emulate hand** shape (for minimal hand movement)



Code

```
void processJoystick(int joystickNum) {
  // joystickNum represents which joystick (Joystick 1: 0 or Joystick 2: 1
  // Loops through list of joysticks and processes the
  // X and Y variables to correspondingly use
  // up, down, left, and right arrows

  if (joystickNum >= 0 && joystickNum < 2) {
    bool controlType = gameJoysticks[joystickNum].isWASD;
    int sensitivity = gameJoysticks[joystickNum].sensitivity;

    // Convert raw analog values (0-1023) to centered values (-512 to 512)
    int centeredX = analogRead(gameJoysticks[joystickNum].xPin) - 512;
    int centeredY = analogRead(gameJoysticks[joystickNum].yPin) - 512;

    // Apply deadzone (ignore very small movements)
    const int DEADZONE = 50;

    if (abs(centeredX) < DEADZONE) {
      centeredX = 0;
    }
    if (abs(centeredY) < DEADZONE) {
      centeredY = 0;
    }

    // Apply sensitivity (10 - 100) to readings
    // At sensitivity = 10: very slow movement (scaled by 0.2)
    // At sensitivity = 100: very fast movement (scaled by 2.0)
    float sensitivityFactor = map(sensitivity, 10, 100, 20, 200) / 100.0;

    int adjustedX = centeredX * sensitivityFactor;
    int adjustedY = centeredY * sensitivityFactor;

    // Ensure values don't exceed limits
    adjustedX = constrain(adjustedX, -512, 512);
    adjustedY = constrain(adjustedY, -512, 512);

    // Process based on control type, whether it is WASD or cursor
    if (controlType == true) {
      // WASD movement
      handleWASDmovement(adjustedX, adjustedY);
    } else if (controlType == false) {
      // Cursor movement
      handleCursorMovement(adjustedX, adjustedY);
    }
  }
}
```

processJoystick function: processes joystick movements (Arduino code)

Reads x, y values from the analog pins for the joystick

Ignores **small, accidental movements** inside deadzone (helps with tremor)

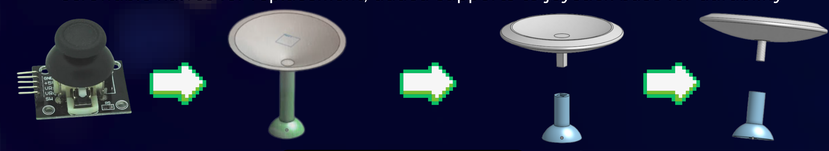
Multiplies x, y reading by user-chosen sensitivity (personalizable based on dexterity level

Calls either handleWASD or handleCursor helper functions for **arrow keys/cursor movement**

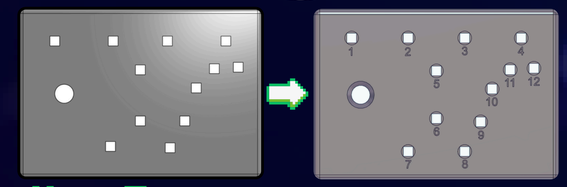
Design Process (4 iterations)

Buttons: Based on Arduino tactile buttons; 3D-printed shell design to fit button cap, reduced brittleness. Added variety of shapes, designs, and textures for customizability

Joystick: Used Arduino joystick. First designs lacked customizability; 3D-printeed in screwable halves for replacement; added supports to joystick base for durability



Shell: Rectangular shape based on keyboard. Optimized **joystick hole** and **button holes**, engraved **button labels**



Testing Process

- Self-testing for functionality using limited dexterity gloves (app, controller)
- **Feedback** from volunteer testers (tremor, arthritis, Parkinson's, MECP2 Duplication Syndrome)
- "If I was to use a PC, **gaming would be a lot harder** because the buttons are too small and can't be gripped like your joystick"-77 y/o with Parkinson's (translated from Mandarin)

Testing Data

Iteration #	Cost	Responsiveness	Accessibility Rtg. (survey/testing)
1	$70	77%	6.5/10
2	$63	85%	7/10
3	$57	93%	7/10
4	$52	99%	8/10

Results

Using interchangeable and larger buttons, an app for customization, and cost-effective materials, Game Changer allows all to equally enjoy the vast benefits of gaming regardless of physical ability or financial status.

Conclusion

Fulfillment of user requirements:

- Our controller is **durable** (strong 3D-printed shell, buttons last ~500,000 clicks) and costs **$52** to make (<$70.00)
- Game Changer has **large, easy-to-press** buttons and a joystick that are **adjustable** in controller/app
- **No lag** when testing, Leonardo board has adequate clock speed of 16 MHz

Improvements for future:

- Host inclusive **gaming events** at schools
- Lower **actuation force** with weights, magnets, or different SMD parts
- **Second joystick** and **reversible** controller design
- **Text-to-speech** for app