Boston University
Electrical & Computer
Engineering

Boston University
Electrical and Computer Engineering
EC463 Senior Design Project

Second Prototype Testing Plan and Report

ICHI

By

Team 10

ICHI

Team Members

Ande Chen achen965@bu.edu
Grace Kim vzgrace@bu.edu
Henry Galindo hgalindo@bu.edu
Jeanette Villanueva jivillanu@bu.edu
Ronald Huang rhuang@bu.edu

Required Materials

Hardware:

- Raspberry Pi Zero 2W
- Joystick
- 3 push buttons
- 1 switch
- MCP3204 Analog to Digital Converter
- USB Microphone
- Tobii Eye Tracker 5
- Pisugar 1200mAh Battery Module

Software:

- Unity
  - Gaze Point with Tobii Eye Tracker 5
  - Recognize mouse clicks and holds
- Tobii Experience App
  - Recognize user’s eyes

Hardware Pinout

Raspberry Pi Pin #	Usage/Description
GPIO4	Right push button input
GPIO22	Left push button input
GPIO10	Serial Data In for Analog-to-Digital Converter (ADC) for Joystick
GPIO9	Serial Data Out for ADC
GPIO11	Clock for ADC
GPIO6	Push to Talk button input
GPIO8	Chip Select/ Shut Down for ADC
5V	5V connection to battery pack
GND	Grounding connections for all components
3V3	3.3V power for all components

Setup:

The Raspberry Pi Zero 2W should be plugged into a power source. The hardware inputs (buttons, joystick) as well as the external microphone connected to the Raspberry Pi Zero 2W will be wirelessly transmitted over Bluetooth. The Tobii Eye Tracker 5 should be plugged into the PC and attached to the monitor. The user will then open the Tobii Experience App and follow the steps in connecting their eyes with the tracker. After that, the user will run the Unity program which will test how accurate the Tobii Eye Tracker 5 is along with detecting left and right mouse clicks as well as the speech to text functionality on the controller.

Pre-testing Set up Procedure:

Software Side:

Bluetooth Communication

1. Run Python script

Unity

1. Open Unity and make sure the program has no errors prior to running

Hardware Side:

1. Make sure Raspberry Pi Zero 2W is plugged into PC
2. Plug in Tobii Eye Tracker 5 and mount on monitor

Testing Procedure:

1. Connect Raspberry Pi Zero 2W to computer, compile and upload Python program
2. Test each component (buttons, joystick, microphone)
3. Run the Unity program

Measurable Criteria

The criteria for successful running and output is as follows:

I. The Raspberry Pi Zero 2W should successfully detect hardware inputs:
   - A. Push Button inputs will be emulated as left and right mouse clicks on the Unity program
   - B. Joystick input will be emulated as a vertical scroll on the program

II. Push to talk button will trigger the speech to text function of the controller
   - A. Microphone inputs will be printed to as the minimum, maximum, and delta values when push to talk button is pressed
   - B. Text will be displayed in Unity

III. Unity program should successfully change the color of the cube depending on which push button is pressed on the controller
   - A. Cubes are initially colored and will turn gray when gaze point is on the sphere
   - B. If the left push button is held down, the cube will follow the user’s gaze, emulating a click and drag.

C. The cubes can be stacked and arranged in any desirable fashion.
D. When the restart button is looked at and clicked, it will reset the objects on the screen and clear the text field.

Degree of end-to-end completion achieved and demonstrated by this testing

Demonstrated by this testing, we have completed the minimum viable product of our project. On the software side, we were able to build the appropriate game environment that is compatible with the Tobii Eye tracker and successfully integrated the eye tracker to interact with objects in the game. Our game demonstrates the use of the eye tracker as well as scroll, speech to text, mouse clicks and mouse drags that are registered in combination with the user's eye gaze. We were also able to build a script within the raspberry pi environment that emulates the button clicks as well as joystick inputs as mouse functions. The USB microphone is able to detect audio and displays it as text on the text input field in the Unity program. In addition, there is a red restart button that turns gray when looked at, and when the controller is pushed while the gaze is on the button, the scene will reset. On the hardware side, we were able to set up the Raspberry Pi Pico 2W with the external battery, joystick, analog to digital converter and push buttons. We have also set up the means to use the USB microphone for speech to text detection. We designed and implemented the 3D print of the shell and buttons of the controller. We have also set up bluetooth communication between the pi and Unity to integrate the hand-held controller and computer actions.

Discuss conclusions based on test data

Based on our results, we were able to successfully simulate mouse functions from the controller's button clicks and joystick movements onto the Unity game. Additionally, we were able to demonstrate the controller's USB microphone picking up audio only when the push-to-speak button is pressed. Further, the speech-to-text function accurately transcribed audio input from a defined microphone and displayed it on the text input field of Unity. Thus, we can conclude that we have all of our features working appropriately in addition to successful bluetooth communication between the controller and the Unity environment.