Boston University
Electrical & Computer Engineering
EC464 Capstone Senior Design Project

User's Manual
ICHI

Submitted to

Michael Hirsch
Address 1
Address 2
Phone
mhirsch@bu.edu

by

Team #10
Intuitive Computer-Human Interface

Team Members

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

Submitted: Apr 17, 2023

ICHI User Manual

Table of Contents

Executive Summary.......... ii
1 Introduction.......... 1
2 System Overview and Installation.......... 2
2.1 Overview Block Diagram.......... 2
2.2 User Interface.......... 2
2.3 Physical Description.......... 2
2.4 Installation, Setup, and Support.......... 2
3 Operation of the Project.......... 3
3.1 Operating Mode 1: Normal Operation.......... 3
3.2 Operating Mode 2: Abnormal Operations.......... 3
3.3 Safety Issues.......... 3
4 Technical Background.......... 4
5 Relevant Engineering Standards.......... 5
6 Cost Breakdown.......... 6
7 Appendices.......... 7
7.1 Appendix A - Specifications.......... 7
7.2 Appendix B – Team Information.......... 7

(Right click on Table of Contents to update fields and page numbers automatically)

Executive Summary

Computer literacy is gradually becoming a prerequisite for participation in this technology-driven world. However, traditional mouse and keyboard input can prove to be difficult or impossible to use for individuals who suffer from illnesses that hinder their fine motor skills. Our product, ICHI, utilizes the Tobii Eye Tracker 5 and speech-to-text technology to offer a more intuitive and accessible method of working with a computer. ICHI incorporates all of the same functionalities as a computer mouse in an ambidextrous handheld controller in addition to a built-in microphone for speech-to-text functions. Further, this product will be demonstrated in a Unity Game environment, where the eye tracker will be used to focus the cursor on specific objects and a text box to show speech to text.

1 Introduction

The ICHI system makes computers more user-friendly and accessible to those who suffer from impairments of fine motor skills–the movements that require precise coordination of small muscles, such as those in the hands, fingers, and wrists. Our system is designed to aid people with arthritis, carpal tunnel syndrome, or other fine motor impairments that make using a keyboard and mouse painful or difficult, but any individual who seeks to have an improved experience with their personal computer can rely on ICHI to help increase productivity and functionality on a computer. Our device is more intuitive than the traditional computer system setup because of its use of Tobii's eye-tracking and speech-to-text functionality and is currently compatible with the Windows 10 operating system.

Our product eliminates the need for a computer mouse and keyboard through the use of our wireless hand-held device in conjunction with the Tobii Eye Tracker 5. The Tobii Eye Tracker 5 is a device that uses advanced eye-tracking technology to track and analyze the movements of the user's eyes. Once the user's eyes are calibrated with the Tobii Eye Tracker, their gaze will represent the mouse cursor. The hand-held device was inspired by the Wii nunchuck controller primarily because of its "one size fits all" size, yet our controller acts as a two-in-one through which all basic computer functionality is performed through its use. That is, instead of a computer keyboard, the user will push down on the push-to-talk button that will then wait for audio from the microphone they speak into. Then, speech-to-text technology will imitate typing on the computer.

The Raspberry Pi Zero 2W acts as the control center for all things related to ensuring the controller operates properly. It is mainly responsible for the bluetooth communication between the controller and the Windows PC, in which it handles the execution of the mouse inputs. The buttons on our controller are wired to specific pins of the Raspberry Pi that through the use of Python emulate left and right mouse clicks. While, the joystick imitates the functionality of scrolling like the computer mouse.

The end result of our year's work in completing ICHI has proven that it is possible to make computers more user friendly and more inclusive of those with fine motor impairments. In order to become accustomed to this version of interacting with a computer, our users can play around with a simple game created in Unity. This will be the onboarding process in which our users become more comfortable with using the controller as both mouse and keyboard. Once our users are comfortable with using the controller and navigating where the buttons and joystick are located, they should have a better experience with their personal computer.

2 System Overview and Installation

[This section should tell the user how to install and set up the system. Some descriptive material is included here to make the system's structure more apparent.]

2.1 Overview Block Diagram

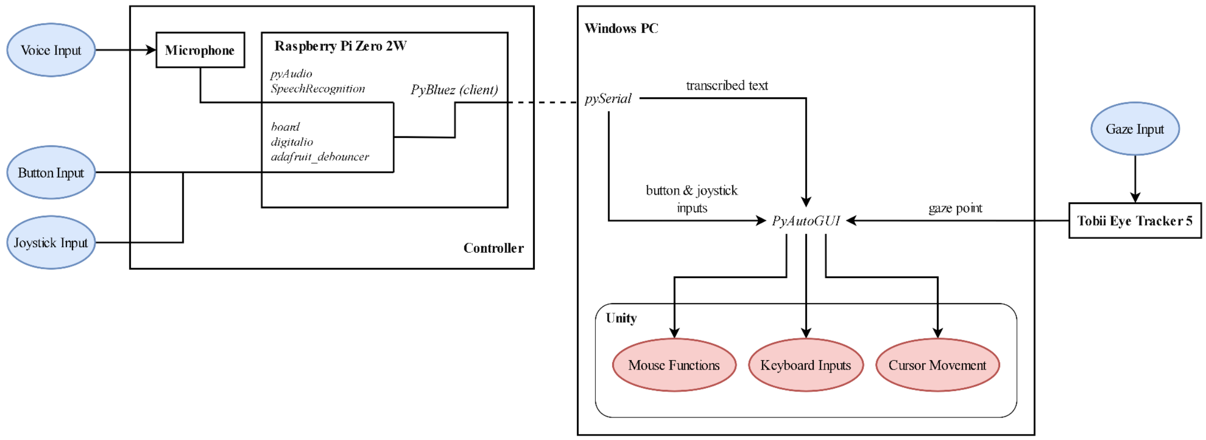
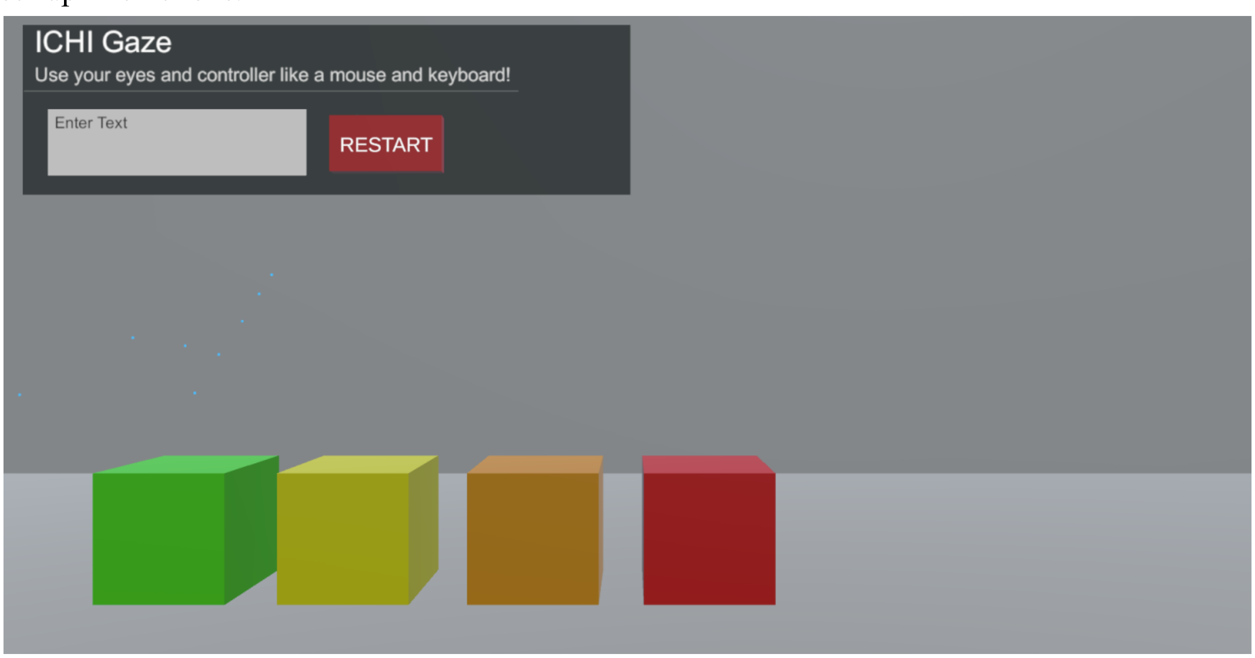
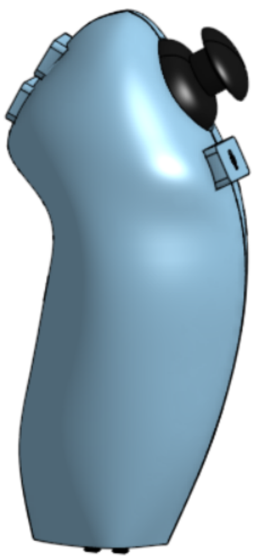


Figure 1. ICHI System Diagram
Components in blue are the inputs, whereas those in red are the outputs.

2.2 User Interface.

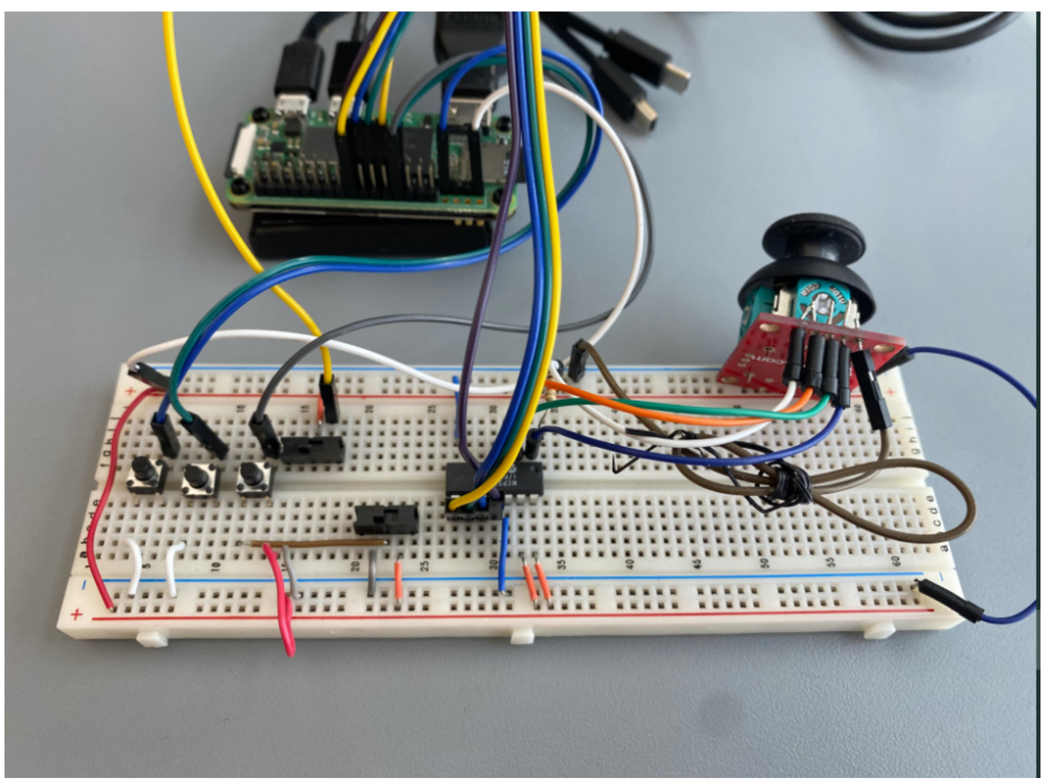
If appropriate, include screenshots of specialized set-up GUI(s), and subsequent important screen set-up interactions.

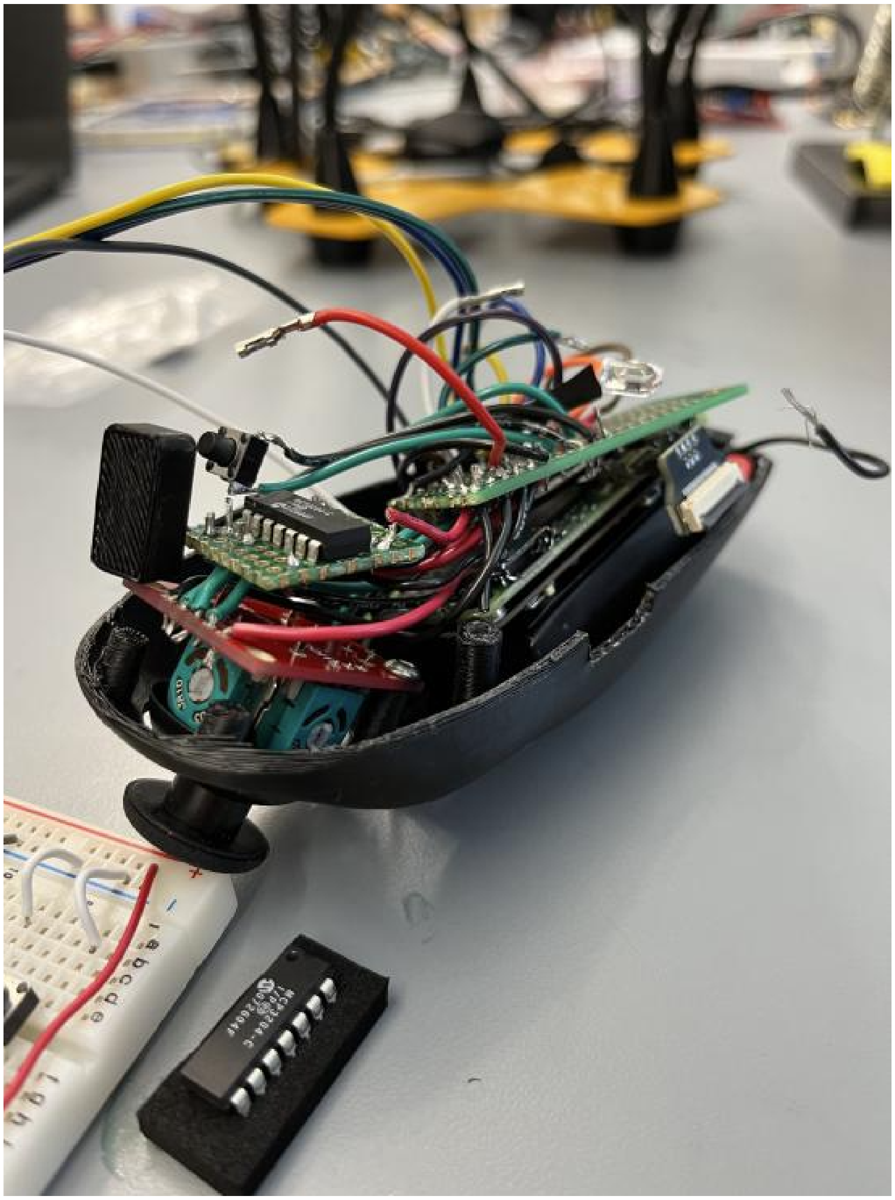




2.3 Physical description.

Provide a sketch of your project hardware (accurate and <u>to scale</u>, in 3-D or as a series of planar views), or photograph.





2.4 Installation, setup, and support

1. Launch the Tobii Experience App. This app will run in the background.
2. Turn the power switch on the wireless controller to the ON position.
3. Open Unity Hub and click on ICHI.
4. Run the Unity program and interact with objects on screen

Describe the setup process and services needed. For software, describe installation, unzipping, wizards or install shields, and any preferences that must be set. If defaults are present, document them. If hardware, describe initial assembly and power-up.

3 Operation of the Project

[This section describes how to use the project. Anticipate what the User needs to know and do Set-up and configuration were discussed in Section 2 already.]

3.1 Operating Mode 1: Normal Operation

- Turn on the controller using the power switch.
- Connect the controller to the user's computer via Bluetooth.
- Open the Tobii Experience App to calibrate the eye tracker.
- Open Unity and run the program.
- Turn on the gaze for the eye tracker and look at the circular objects on the screen.
- Observe how the color changes when your gaze and mouse are both fixed on the object.
    - Color should turn blue.
- When the object is right-clicked, the object should turn pink.
- When the object is left-clicked, the object should turn black.
- For speech-to-text, press and hold the push-to-talk button located on the front of the controller.

3.2 Operating Mode 2: Abnormal Operations

At times, the gaze tracker in Unity may not be accurate and it could be hard to focus on objects. When this happens, exit the program and recalibrate the eye tracker in the Experience app.

3.3 Safety Issues

- Lithium-ion battery may become hazardous if it is punctured or damaged.
- When handling the battery inside, avoid overheating, overcharging or contact with dangerous chemicals.
- Prolonged use of the computer while using our device may cause eye strain or fatigue.

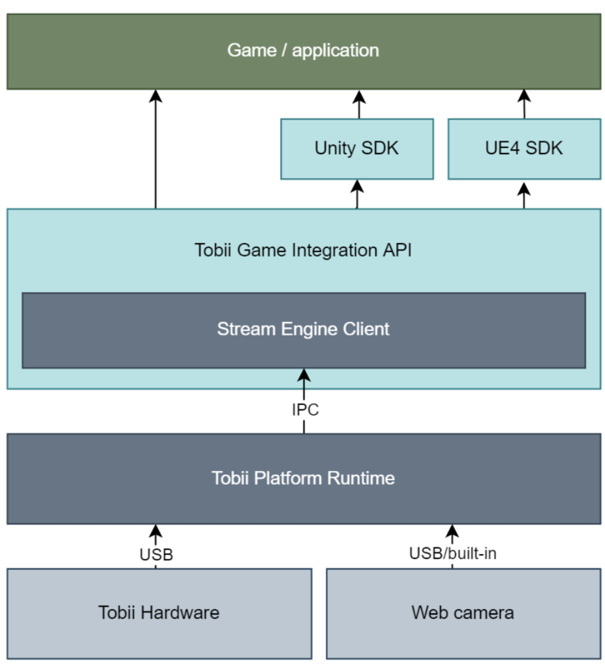
4 Technical Background

[The User's Manual will be the <u>primary reference</u> for your project when others want to understand your creative work and how to operate it. While the Resource CD and individual logbooks are also helpful, this User's Manual will be consulted first by users and by future teams considering your project.]

In order to build a project that will replace a traditional mouse and keyboard, we utilize the Tobii Eye Tracker 5 to work in conjunction with our handheld controller. The Tobii Eye Tracker 5 is a powerful eye tracker made for head and eye tracking, engineered for PC gaming. The eye tracker works by using a technique called pupil center corneal reflection (PCCR). First, the eye tracker uses its illuminators to create and send out a pattern of near-infrared light which is reflected in the user's eyes. The eye tracker's camera then takes high-resolution images of the user's eyes and the patterns. The image processing algorithms are used to find specific details in the user's eyes and reflection patterns. Based on this information, the eye's position and gaze point are calculated using a 3D eye algorithm. It is important to calibrate your gaze point to increase accuracy and precision. This is made possible by a quick calibration walk-through on the Tobii Experience App. The eye tracker takes the calibration data and combines it with a 3D model of an eye to create an accurate eye-tracking experience.

Since the Tobii Eye Tracker 5 was developed mainly for a gaming environment, we decided to demonstrate the functionalities of our product by building a similar environment on Unity. Unity is a powerful cross-platform game engine used by game developers for creating 2D and 3D games, simulations, and animations. Users can also interact with certain objects within their environment, which is one of the main focuses of the ICHI program. It is compatible with various platforms such as desktop, mobile, and virtual reality. One of the most important features that Unity offers is the ability to edit scripts to create our own logic and interaction for our game. C# is the primary language that is used, and there are dozens of other tools used for creating and controlling animations. They also provide an Asset Store where developers can access and download a wide range of pre-built games, environments, and animations.

We initialized and integrated the Tobii Eye Tracker into our project by obtaining a license from Tobii that provides development rights with Tobii SDKs and eye-tracking related software and APIs. We were able to get the Tobii Unity SDK for Windows desktop, which provides a native C++ API called Tobii Game Integration API. This API is built on top of Stream Engine, which is Tobii's low-level SDK. This allows faster implementation, robustness, and access to specific eye-tracking features on Unity. The API provides stable connection to our Tobii Eye Tracker, access to eye tracking, head tracking and presence data streams, automatic DPI-scaling and remapping of gaze points to game window coordinate, and full access to Tobii's 6DOF Extended View.



The speech-to-text feature in our Raspberry Pi is a Python program that uses Google Speech Recognition and Pyaudio libraries to detect and display audio input. The Pyaudio library allows developers to detect where the USB microphone is located on the Pi and enable audio input from the microphone. The Google Speech Recognition library uses machine learning algorithms to recognize audio data picked up by the microphone and transcribes it into text.

In addition to the eye tracking and speech-to-text features, our device will utilize a Raspberry Pi Zero 2W. The Zero 2W is similarly powerful to the Pi 3 A and B but has a much smaller form factor, making it a perfect device for our handheld controller. The Raspberry Pi Zero 2W is designed to act as a brain for projects, with multiple input ports including Mini-HDMI, Micro-USB 2.0 OTG, Camera Serial Interface and 40-pin GPIO header. These inputs allow for the Raspberry pi to take inputs of all kinds, including audio inputs,

5 Relevant Engineering Standards

Include a brief discussion of the most relevant engineering standards to your project. These can consist of standards related to electrical design and construction (eg. National Electrical Safety Code), software design, coding standards, communication and internet protocols, operational environment, governmental requirements, etc.

Eye tracking is a technology that involves tracking eye movements, specifically, the activity of pupils, in order to understand the behavior and intentions of the person. The most important engineering standard that ensures the safety and reliability of our project is from the Food and Drug Administration (FDA), which provides information on the use of eye-tracking devices used in medical settings. Since our project will be targeting individuals with certain motor disabilities, it is important that these guidelines are followed. However, our final product is not considered a medical device. The FDA also points out the potential risks eye trackers could pose to patients, such as laser hazards. Therefore, our product should implement appropriate safety features and warnings from the Tobii website.

There are several video game development standards that are relevant to the Unity program. These topics include game design, user interface, and accessibility. It is important that Unity is user-friendly to ensure ease in developing certain games or animations. Unity must also adhere to certain software standards such as frequent testing, documentation, and version control. Since C# and C++ are the main programming languages used in Unity, it is necessary to follow certain updates they might have.

6 Cost Breakdown

Project Costs for Production of Beta Version (Next Unit after Prototype)				
Item	Quantity	Description	Unit Cost	Extended Cost
1	1	Tobii Eye Tracker 5	220.15	220.15
1	1	Raspberry Pi Zero 2W	89.90	89.90
2	1	PiSugar 1200 mAh Battery Module	38.99	38.99
3	1	MCP3008 8-Channel 10-Bit ADC	4.50	4.50
4	1	Mini USB Microphone	5.95	5.95
5	1	Micro-USB Male to USB A Female Dongle	6.26	6.26
6	1	Protoboard	4.50	4.50
7	1	2-Axis Analog Joystick w/ Select Button + Breakout Board	5.95	5.95
8	3	Tactile Switch Buttons	0.25	0.75
9	1	Linear Switch	0.76	0.76
10		Wires		
			Beta Version-Total Cost	377.71

Our beta prototype continues using the Raspberry Pi Zero 2W as the primary microcontroller and the Tobii Eye Tracker 5 as the foundation of the ICHI system. All of the electronics are soldered more permanently onto a protoboard that fits within the controller's form factor. To enable wireless operation, we use a PiSugar battery module that is purpose built for the Pi Zero 2W. The MCP3008 allows us to convert the joystick's analog inputs to digital outputs that can be read by the Pi Zero 2W's digital GPIO pins. To enable speech-to-text, we use a small form factor USB microphone that connects to the Pi through a dongle. Lastly, we have tactile buttons and sliding switches for mouse functions and power, respectively.

7 Appendices

7.1 *Appendix A - Specifications*

Eye tracking (Tobii)	Must be calibrated in Tobii Experience app (at least one time before starting project and a second time if needed)
Unity	Use eyes and mouse clicks to interact with objects in the game (3-4 blocks including one restart button)
Controller	Mimics left and right clicks of a mouse (one button) Use joystick for scrolling feature One button for activating speech to text

7.2 *Appendix B – Team Information*

Jeanette Villanueva (jivillan@bu.edu)
I am from West Covina, California and currently pursuing my Bachelor's degree in Computer Engineering. Growing up my aunt (mechanical engineer) and uncle (electrical engineer) exposed me to small but fun activities that mimicked what engineering was about, like helping me light up a Christmas tree by soldering the LED lights on the circuit board. After attending Latina in STEM conferences, I had an interest in how computers worked, how when you push a button on the keyboard you could see the letter pop up on the screen, and what pieces actually made up a computer. I wanted to know everything about a computer. Thus, Computer Engineering seemed like the right major for me. Now, as a Senior looking toward my future plans post-graduation, I am actively looking for full-time positions in Software Engineering.

Henry Galindo (hgalindo@bu.edu)
I am from New York City, NY and I am studying Electrical Engineering. I attended the High School for Math, Science and Engineering at CCNY where I was introduced to the fundamentals of engineering design and technology. This is where my interest in Digital Systems and electronics grew, and I decided to pursue a future in Electrical Engineering. I was able to learn how to solder rudimentary circuits including random number generators and 1-bit adders.

Grace Kim (vzgrace@bu.edu)
I am from Queens, NY and I am majoring in Computer Engineering and minoring in Visual Arts. I went to Queens High School for the Sciences and found my interests to lie in many of my STEM courses. I decided to study engineering and further decided on Computer engineering after taking my first programming class in college. I enjoyed the problem-solving aspects of engineering and grew an interest in the workings of both software and hardware components of a computer.

Ronald Huang (rhuang@bu.edu)

I am from Danville, California, studying Biomedical Engineering. I attended Monte Vista High School and developed an interest in engineering through the classes I took as well as influences from my family. I am interested in applying computational methods such as machine learning and computer vision into the biomedical sciences. I want to pursue a Masters degree in either ECE or computer science after my graduation in May 2023.

Ande Chen (achen965@bu.edu)

I am from Rowland Heights, California and am studying Computer Engineering with a concentration in machine learning. I attended Fairmont Preparatory Academy where I became interested in engineering by participating in the First Robotics Competition with the school's robotics club. The challenging technical problems and collaborative, interdisciplinary environment motivated me to major in Computer Engineering. After completing my undergraduate degree, I would like to pursue a career in technical consulting.