# Slingstone

Caroline Broady[1] (ctb55) Julian Narvaez[1] (jmn112) Diana Zavela[2] (dfz3)

Technical Advisor: Dr. Marc Buchner, Associate Dean, Academics, Case Western Reserve University

Instructor: Prof. Greg Lee

Submitted for EECS 398 in partial fulfillment for the degree of Bachelor of Science of

Electrical Engineering[1]

Systems & Control Engineering[2]

Case Western Reserve University

Fall 2020

## Academic Integrity Policy

All students in this course are expected to adhere to University standards of academic integrity. Cheating, plagiarism, misrepresentation, and other forms of academic dishonesty will not be tolerated. This includes, but is not limited to, consulting with another person during an exam, turning in written work that was prepared by someone other than you, making minor modifications to the work of someone else and turning it in as your own, or engaging in misrepresentation in seeking a postponement or extension. Ignorance will not be accepted as an excuse. If you are not sure whether something you plan to submit would be considered either cheating or plagiarism, it is your responsibility to ask for clarification. For complete information, please go to https://students.case.edu/community/conduct/aiboard/policy.html.

The individual(s) submitting this document affirm compliance with the above statement

Caroline Broady

# Executive Summary

3D models are growing in importance for the artistic community especially given the COVID-19 pandemic with limited ability for gallery exhibition. Education and conservation staff at the Cleveland Museum of Art (CMA) have expressed the need for an intuitive 3D model controller. An intuitive 3D model controller is presented below to fulfill the stated need. 9-axis absolute orientation data pulled from an absolute orientation sensor will be passed through to a web server and converted into 3D model orientation transformations. 3D model viewer libraries are implemented on a locally hosted website to display uploaded STL models of CMA artwork. The device is housed in an acrylic box to provide shape and structure for CMA users.

# Table of Contents

## Problem Statement

Museum exhibitions, product demos, and other presentations use digital 3D models to display objects when in-person interaction is not feasible. The traditional methods for orienting the digital object use a mouse or trackpad. While readily available, these products disrupt the user experience as they feel unnatural to operate. There is a gap in technology available today for a product that the user can operate intuitively, in particular, a handheld device that the user can orient to rotate the 3D model in real time.

Virtual models are increasingly used to augment reality or replace in-person interaction when it is not feasible, especially given the COVID-19 pandemic and social distancing guidelines. The Cleveland Museum of Art (CMA) presents a stakeholder in this shift to virtual communication. Given the COVID-19 pandemic, museums are forced to operate on a low number of visitors and have been forced to close during periods when the outbreak worsens. This is especially difficult for the art industry, as so much of the experience relies on in-person interaction and seeing the artwork in person, both for a visitor and a staff member doing research or conservation work. However, the use of 3D models and the creation of 3D model galleries allows for individuals to remain engaged with the CMA's content without going physically to the museum. Slingstone allows the user to manipulate the model with a handheld device in a user-friendly environment instead of the conventional mouse and trackpad manipulation which can be imprecise and difficult to use. Additionally, we have designed a housing for the electronics that can fit comfortably in the hand of a user.

## Context

The Slingstone device will fit within the CMA within the following three categories: Education, Conservation, and Curation. Throughout the design process, numerous staff members at the CMA have been consulted to fully develop the use-cases for each of the categories.

*Education*: The CMA ArtLens Gallery provides a unique opportunity for interactive education; the Slingstone project will allow for another means for a general member of the CMA to interact with 3D art virtually. Existing ArtLens interactive exhibits such as the "Strike a Pose", "Make a Face", and "Body Language" already allow users to connect physically with displayed artwork. The Slingstone device will be another means for users to connect physically with artwork to support the CMA's overarching theme of "What Can Art Be?" and will allow users to access and manipulate 3D models to have a more hands-on experience at the museum. This device can also be used in a virtual gallery setting, where text and models can be linked together to provide the users the experience of seeing the artwork and reading details about it, even if such an in-person experience is not possible. This device allows a museum visitor to more easily manipulate 3D models as compared to conventional mice and joysticks.

*Conservation*: Conservators within the CMA can share information on different museum pieces virtually and emphasize sections of the relevant artwork quickly and effortlessly. Expert consultants brought in virtually to assist with conservation projects can develop a more intimate understanding of the artwork using the Slingstone device as the model will allow for the museum staff member to point out artwork details, even when it might be difficult to reach on the physical piece itself. The Slingstone device would supersede existing forms of media sharing such as photographs and written documentation when 3D models are available for existing artwork.

*Curation*: Curators can highlight different pieces during lecture series and manipulate 3D artwork without sacrificing time and effort controlling the model. Curators can be assured virtual audience members have a strong understanding of the artwork before providing cultural context for the art pieces. Additionally, in a large conference or meeting, the presenter could manipulate a work of art on their own device and the audience members will be able to see the movements on their own devices, allowing the audience to more easily follow along with a presentation. The Slingstone device will be used to supplement photographs and written documentation of the artwork as it provides an easy way to access and manipulate the CMA's 3D model collection.

## Success Criteria

The outlined need leads into the following success criteria centered around the creation of a device that can be used to manipulate 3D models in a seamless manner has not changed. However, feedback provided from CMA staff during user testing has led to the modification of the success criteria. Modifications to the success criteria were made since the start of the project to provide greater specificity in the following three areas; defining successful device and displayed model interaction, defining successful user interaction with the GUI, and defining successful device startup operation. The final definitions for successful system operation are as follows:

1. The displayed model will mimic the device orientation within 10 degrees
2. The device can be held in one hand
3. The system startup operation can be completed within 30 seconds
4. The device volume will be less than 300cm^3
5. Communication between the sensor and MCU/single-board computer will be conducted in under 50ms using a 3ft long wired connection
6. CMA staff are able to upload and use different 3D models with the GUI website open using less than 3 clicks
7. Scrolling and model display latency for the GUI website is under 100ms

The success criteria meets the stated need of an intuitive 3D model controller with a lightweight frontend. Low latency, small device volumes, and clearly outlined device-model interactions require a full-stack approach to prototyping.

## Background

The Cleveland Museum of Art stores its 3D model collection on its website using Sketchfab, a popular site for storing and creating 3D model collections. It has a powerful API that can be used to embed models in other sites and make modifications to the model viewer. The State Darwin Museum in Moscow, Russia created an application built on top of Sketchfab by modifying the Viewer API [1]. Sketchfab allows for models to be embedded into websites with only a few clicks. This ease of use and access for 3D models is essential to allow users of Slingstone to change the model being used.

The State Darwin Museum developed an embedded joystick device in 2019 to connect a physical model with the movements of a 3D model. They developed a physical 3D printed skull model and placed the electronic components within the skull model, allowing the user's exact movements of the physical model to correspond to the movements of the 3D model. The State Darwin Museum used an ESP8266 microcontroller for control and communication and GY-85 chip, a 9-axis gyroscope, magnetom3. eter, and accelerometer, for the sensor input to the model. These two devices were connected via i2C. The researchers at the State Darwin Museum developed a Javascript-based model viewer as the GUI and used SciVi for firmware development. They noticed that as the sensor information was sent over WiFi, a user elsewhere in the building could also see the 3D model manipulation. However, as the Darwin Museum model relied on having a physical version, it cannot be used to manipulate a variety of models, only those that also have a 3D printed version [1].

Another similar application was created by an individual user screen-named “dimtass” as a demonstration of current 3D modelling abilities using open source software. They used an Arduino Teensy 3.2 microcontroller and a MPU-6050 6-axis sensor consisting of a gyroscope and accelerometer to control a 3D model in Unity. However, this 6-axis sensor required complex calibration as it used raw HID protocol. Additionally, a buffer was required to get the raw sensor data from the device into Unity3D. This solution, while highly functional on the user’s computer, was not usable cross-platform [2].

The Adafruit BNO55 9-axis absolute orientation sensor was determined to be the best sensor for this application. The BNO55 contains a gyroscope, magnetometer, and accelerometer. Produced by Bosch, the BNO55 is one of the first chips that has built-in sensor fusion algorithms instead of having to translate raw data from the sensors into information about absolute orientation. The BNO55 can output the absolute orientation information across three axes as an Euler vector or in a four-dimensional quaternion format [3]. The housing has 28 pins, allowing for the BNO55 to be integrated into more complex devices and connected with other components such as in Slingstone. The BNO55 is built to work with the HID protocol via I2C [4].  In a proof-of-concept on the Adafruit website, the BNO55 is connected to an Arduino which allows for 3D model manipulations [3]. However, this demonstration does not include a housing or a user-friendly interface which is not helpful for a user who would want to change the 3D model or have a housing to protect the electronics.

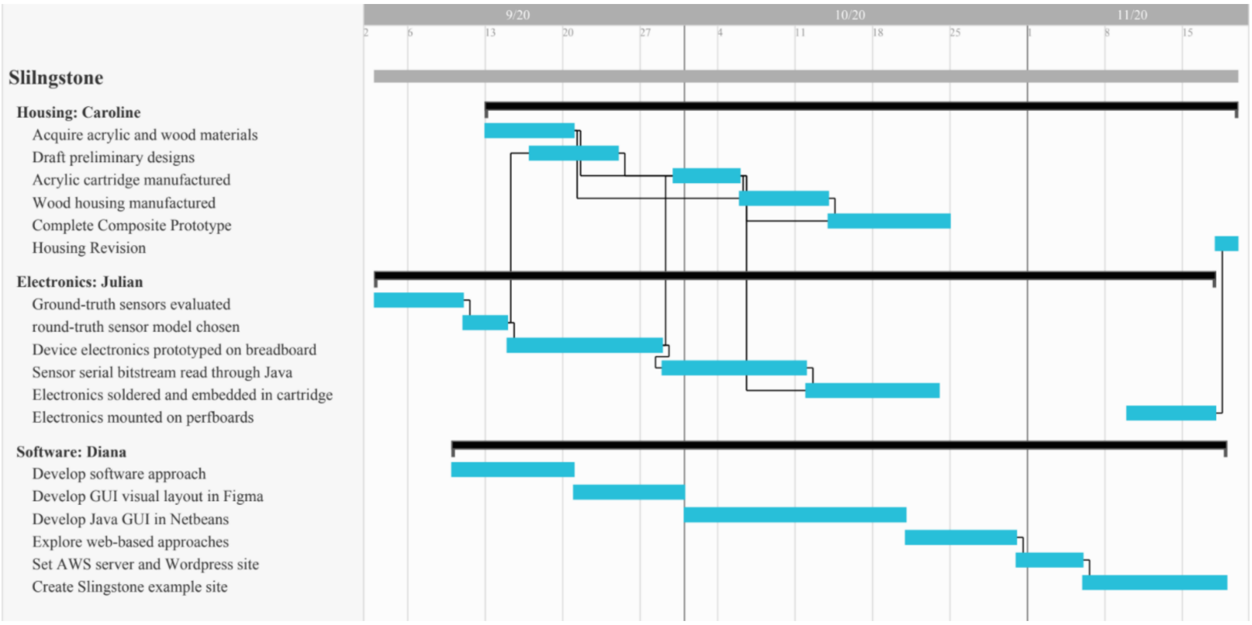
Through this project, two prototypes were built. The first prototype was built with the HID standard as had been done in the previous work by Adafruit and the Darwin Museum [5]. However, the HID approach did not meet all of the success criteria, and a BeagleBone Black (BBB) single-board computer was used in the second iteration. Existing examples are provided through Adafruit of 3D model manipulation using a BNO055 sensor and BBB computer [6].

## Project Management

| Proposed | Final |
|---|---|
| **1. Housing**<br>Caroline Broady served as physical design lead, designing housing for circuitry based around constraints of hardware size. She developed multiple prototypes and drafted conceptual designs for future application. | |
| 1.1 Acquire acrylic and wood materials | 1.1 Acquire acrylic and wood materials |
| 1.2 Housing design modeled in CAD and meets size criteria | 1.2 Design manufacturing vector files to fit proposed electronics design dimensions |
| 1.3 Acrylic cartridge manufactured that can fit all electronics components | 1.3 Build concept prototype from manufacturing files |
| 1.4 Wood housing manufactured meets size success criteria | 1.4 Acrylic cartridge manufactured that fits all electronics with high margin for error **(Milestone 1)** |
| 1.5 Complete composite prototype meeting size, weight, and IP54 criteria **(Milestone 1)** | 1.5 Propose redesign that improves adherence to size, weight, and IP54 criteria. |

| | |
|---|---|
| **2. Electronics**<br>Julian Narvaez acted as Hardware Engineer, responsible for the design and build of the project circuitry. He evaluated different options for configuration, conducted troubleshooting, and made component selection for the device. | |
| 2.1 Ground-truth sensors evaluated | 2.1 Ground-truth sensors evaluated |
| 2.2 Ground-truth sensor module chosen | 2.2 Ground-truth sensor module chosen |
| 2.3 Device electronics prototyped on breadboard and read in sensor bitstream using Arduino IDE | 2.3 Device electronics prototyped on breadboard and read in sensor bitstream using Arduino IDE |
| 2.4 Sensor serial bitstream read through Java | 2.4 Sensor information passed to web server and outputting to a javascript console |
| 2.5 Functional electronics package mounted on perfboard and embedded within cartridge **(Milestone 2)** | 2.5 Functional electronics package mounted on perfboard **(Milestone 2)** |
| **3. Software**<br>Diana Zavela was responsible for software and system design, creating the frontend and backend development to host the project. She examined different options for software approach and developed a solution that implements multiple software products. | |
| 3.1 Develop Java program to print incoming device bitstream on Windows 10 | 3.1 Determine software approach |
| 3.2 Develop GUI visual layout with buttons for uploading and displaying 3D models | 3.2 Develop GUI visual layout with buttons for uploading and displaying 3D models |
| 3.3 Write sensor serial bitstream to GUI | 3.3 Develop Java GUI in Netbeans |
| 3.4 Display 3D model on Java GUI that can be rotated 360 degrees on all axes **(Milestone 3)** | 3.4 Explore alternative web-based approaches |
| 3.5 Create Java executable displays 3D model that can run on a Linux OS, Windows 10 OS, and MacOS **(Milestone 4)** | 3.5 Set up Amazon Web Services remote server and Wordpress test site (Milestone 3) |
| | 3.6 Create Slingstone example exhibit site (Milestone 4) |
| **4. CMA Deployment** | |
| 4.1 Create folder with at least 5 CMA 3D model files | 4.1 Document design changes and publish code online |
| 4.2 Statement of Work for CMA approved by all CMA and CWRU advisors **(Milestone 5)** | 4.2 Perform user testing with CMA staff **(Milestone 5)** |
| 4.3 Documentation Tasks | |

The Gantt chart below illustrates the timeline of project developments:




## Proposed Budget:

| Item | Cost | Note |
|---|---|---|
| 1/8" wood | $20 | Option for prototyping housing |
| Acrylic Sheets | $22 | Material for prototyping cartridge |
| Adafruit 9-DOF Absolute Orientation IMU Fusion Breakout - BNO055 | $40 | Critical sensor |
| Arduino Uno | $23 | Microcontroller |
| Arduino Nano | $21 | Alternative Microcontroller |
| Long Arduino Cable | $8 | To allow moving away from laptop |
| Adjustable Strain Relief Gaskets | $15 | Strain Relief |
| **Total** | **$149** | |

## Final Budget:

| Item | Cost | Note |
|---|---|---|
| 1/8" wood | $20 | Option for prototyping housing |
| Acrylic Sheets | $22 | Material for prototyping cartridge |
| Adafruit 9-DOF Absolute Orientation IMU Fusion Breakout - BNO055 | $40 | Critical sensor |

| | | |
|---|---|---|
| Arduino Micro | $17 | Final Microcontroller |
| Long Arduino Cable | $8 | To allow moving away from laptop |
| BeagleBone Black | $55 | Hosting local web server and receiving sensor data |
| 32 GB SD Card | $7 | Expanding existing BBB flash memory |
| Ethernet Cable | $13 | Connecting BBB to local area network |
| Small Breadboards | $3 | Interfacing BBB with BNO055 sensor |
| **Total** | **$185** | |

## Approach and Methedology

The approach and design methodology were centered around iterating upon prior work to create a device that entirely meets the defined success criteria. Two distinct prototypes have been developed each meeting different parts of the defined success criteria. For the rest of the report, the prototype developed using the HID standard will be referred to as the HID prototype. The prototype developed using the BeagleBone Black (BBB) single-board computer will be referred to as the BBB prototype.

Both prototypes relied on the BNO055 sensor to provide absolute orientation data over a two-wire serial stream. Firmware libraries were freely available for the BNO055 for use with a Raspberry Pi and Arduino which made the BNO055 an attractive option for rapid prototyping [7][8]. In addition, the merged quaternion and euler angle outputs sent directly from the BNO055 removed the need for multiple IMU, magnetometer sensors, and complex transforms to merge the data on the MCU [4]. Multiple open-source projects have been developed using the BNO055 sensor which provide multiple references for firmware and software integration projects.

Passing the serial data collected from the BNO055 sensor into a native web application presented the first of many technical constraints. Serial port read and write privileges are understandably locked or unimplemented in multiple web browsers such as Safari, Mozilla Firefox, and Microsoft Edge for security. Mozilla Firefox for example, has responded with a “potentially harmful” tag to the integration of a Serial API on their browsing platform [9]. Safari has responded similarly with a statement mentioning Serial API as included in “features we have decided to not yet implement due to fingerprinting, security, and other concerns” [10].

Serial data cannot be passed directly from a serial port into a website’s front end directly using the aforementioned browsers. Google Chrome is the sole browser which allows for serial communication from a website’s frontend site to a serial port [11]. Chrome offers the Web Serial API as a solution for integrating serial data into a web browser. However, the Web Serial API is still flagged as an experimental feature in Chrome; the Web Serial API is implemented differently in different versions of Chrome browsers. Examples of BNO055 integration with the Web Serial API are present; however, browser versions, operating system differences, and differences in serial port drivers lead to a non-functioning WebSerial Visualizer on different machines [12].

The aforementioned technical constraints restricting serial communication directly to a website’s front end have led to the following firmware approach.

## HID Electronics Approach/Methodology

HID protocols were chosen for the initial prototype as a means of circumventing the need to collect and manipulate sensor data using a virtual serial port. In addition, HID protocols allowed the HID prototype to interface with a website without needing to implement a "Web Serial"-like solution. In addition, the HID device could be applied generically to whatever frontend is developed for the CMA. The HID standards are implemented across operating systems which offer a cross-platform firmware solution; HID device standards also allow for "plug and play" operation [5]. The HID device can be plugged into a variety of operating systems and works immediately. Cross-platform compatibility is especially important for CMA staff given the lack of hardware standardization used by conservators. Conservators using MacOS or Windows operating systems can integrate the HID device directly into their workflows without needing to download or install firmware drivers.

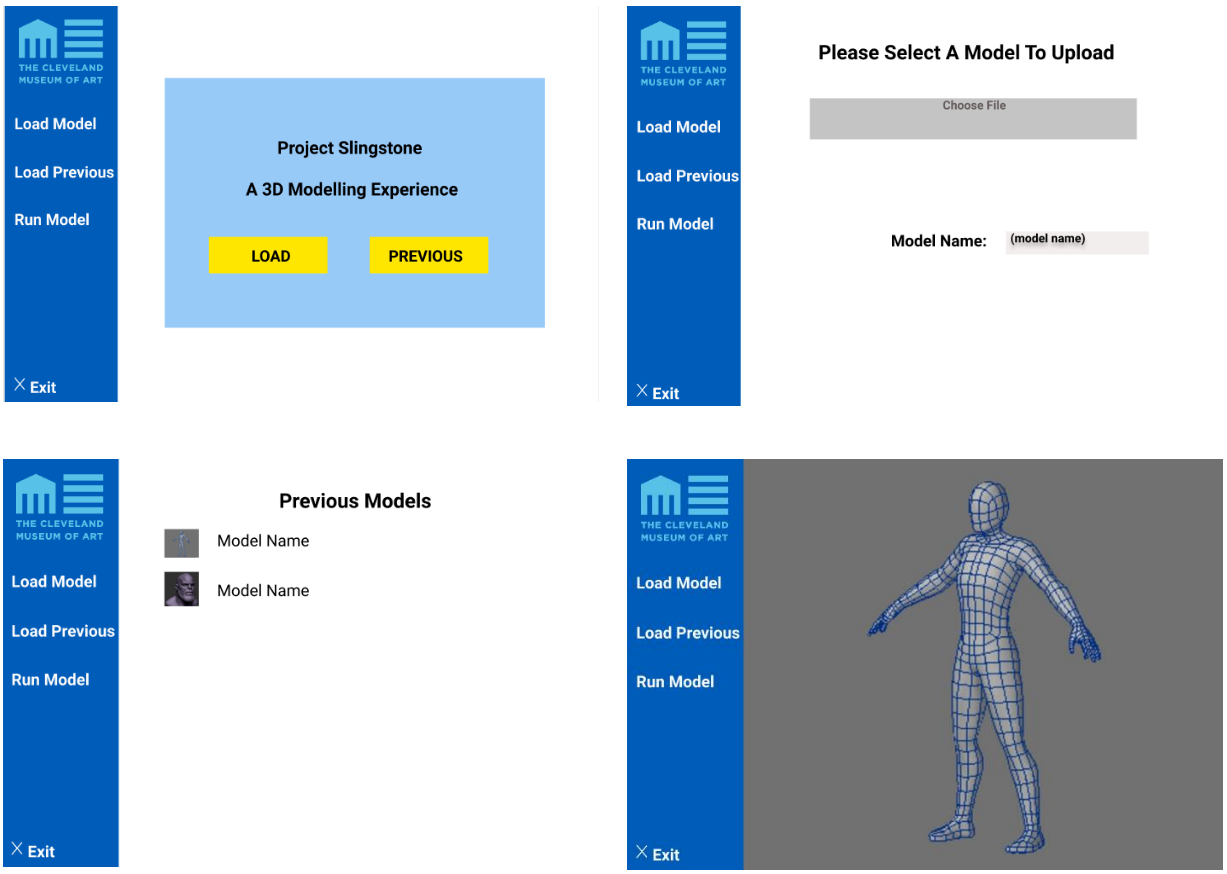
Implementation of an HID device was simplified by the availability of multiple Arduino HID libraries [13]. The HID compatibility and size of an Arduino Micro presented the Micro as a viable microcontroller for the application. Developing with an Arduino Micro allowed for usage of Arduino libraries while maintaining the sought "plug and play" functionality [14]. HID device protocols present in the HID libraries presented options for HID interfaces such as USB Keyboard, USB Mice, and USB Joysticks; in addition, a "Raw HID" example in the Arduino HID library presented an opportunity to pass the sensor information directly into the desktop or laptop without needing to open a virtual COM port. Three options are available for HID implementation; either simulate an existing HID device such as a USB mouse or USB keyboard, create a driver for a new HID device, or pass HID data directly into the 3D model application. The complexity of creating a new HID driver and locking in compatibility with the 3D model viewer software made creating a new HID driver an unattractive option.

Passing HID data directly from the device to the laptop or desktop would allow for unfiltered BNO055 sensor data to be passed through without needing to open a virtual COM port or install serial drivers. However, the USB HID Report packet would still need to be configured for the type of information passed through. For example, the size of the USB HID Report packet would need to be large enough to encode all three axes of motion data coming from the BNO055 sensor to limit latency [15]. Reading in and manipulating incoming USB HID Report packets requires software such as "HID Manager" [16]. Unfortunately, cross-platform "HID Manager"-like software requires outside installation for device operation. Passing packets from the HID Manager software into the 3D model viewer into web-native software presents similar implementation challenges as seen with passing serial information into a website. Browser plugins are available for integrating HID packets into websites but are not supported across browsers.

The technical challenges of creating an HID device or passing HID information directly to a website make both options unfeasible with the given project timeframe. Therefore, simulating an existing HID device using BNO055 sensor data is the recommended avenue. Simulating a USB mouse was chosen because of the existing mouse manipulation interface already present in Sketchfab's 3D model viewer. BNO055 sensor data would be filtered and used to manipulate a mouse cursor. The specific USB Mouse HID implementation is available in the project's GitHub repository located at https://github.com/juliannarvaez/slingstone.

## HID Software Approach/Methodology

The HID software approach has been an iterative process with the goal of having low latency and allowing the user to easily upload various models. The first software approach for Slingstone was creating a Java GUI for users to upload and manipulate 3D models. Figma mock-ups of the GUI to plan what screens were needed and how to integrate functionality.




*Figures 1-4*: Figma model screen design for Java GUI. Top left: Entry screen, allowing the user to select a model. Top right: Model upload screen. Bottom left: Previous model screen. Bottom Right: Model rendering screen.

However, the local Java GUI approach was flawed for a few reasons, leading to a pivot from a locally hosted applet to a web-based approach. Because the CMA's current 3D models are all online on their website in Sketchfab, the creation of a Java applet would be a very large departure from the current systems that the CMA uses currently, and the goal of Slingstone was to integrate with the CMA's already existing infrastructure. Additionally, a locally hosted applet would only allow the user with the applet to use the device and manipulate the 3D models. Given these factors, a web-based approach was determined to be ideal to remain in line with the CMA's current 3D model system.

Ultimately, the team determined that an online example exhibit would be the best way to deliver the functionality of this product to the CMA. This was accomplished through an Amazon Web Services remote server and through an AWS/Wordpress site. This site was connected to the CMA's Sketchfab models to create a virtual exhibition as could be used for visitors of the museum in the future.

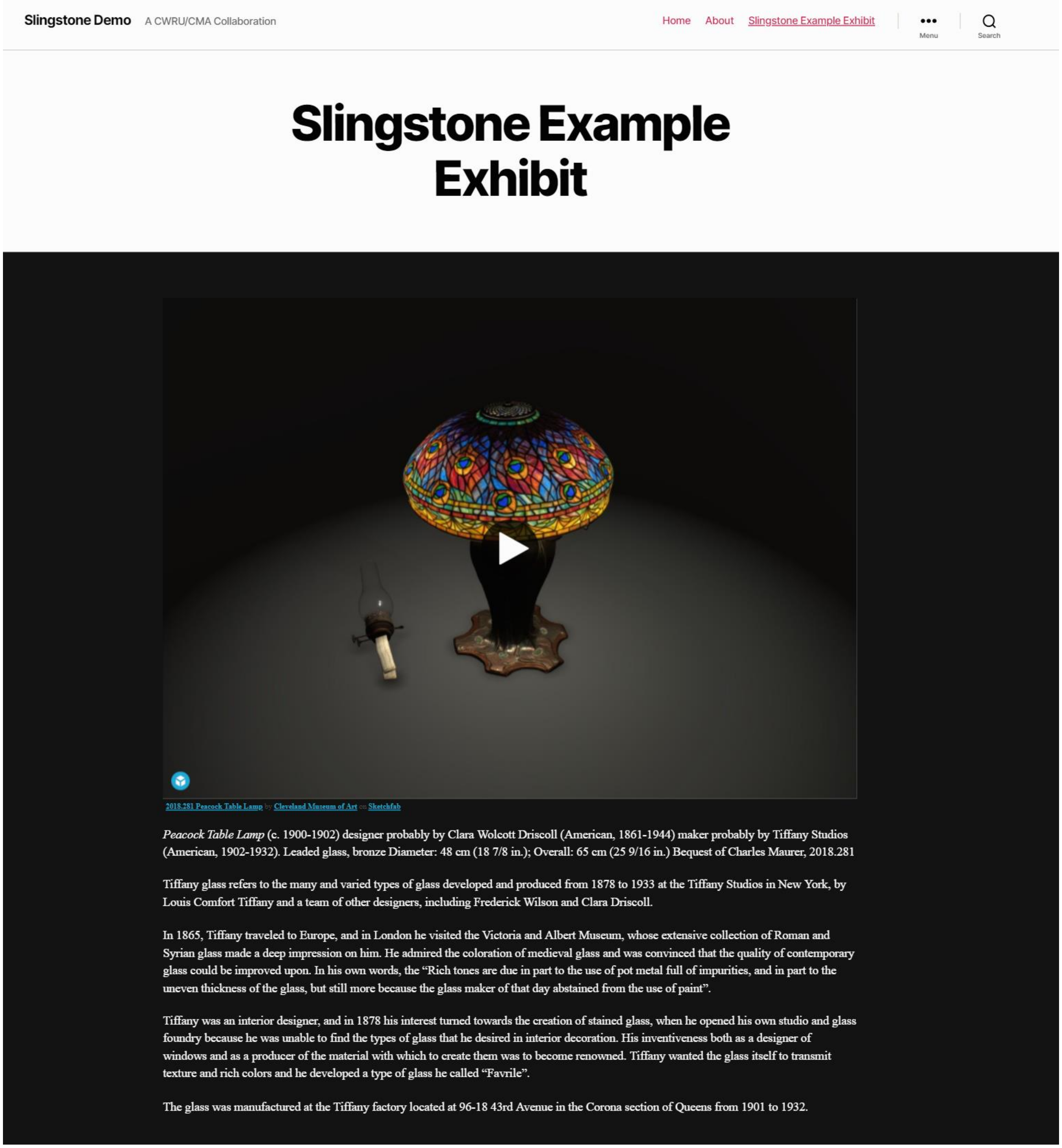



*Figure 5: Excerpt from Slingstone host site*

There were several technical constraints that affected the software approach for the HID prototype. Although Sketchfab is an open-source site, much of the API is locked and only accessible to users with the expensive Pro plans. This meant that many of the viewer modifications that would allow the model to be seamlessly integrated into the website for the viewer was not possible. This was another factor that led to the transition to the BBB prototype.

## Housing & Physical Design Approach/Methodology

The housing approach was designed based on what manufacturing methods were available with quick turn-around time and low volume. A natural method (and the one chosen) is laser cutting, which offers high precision in 2 dimensions and rapid prototyping. 3D Printing was also considered as an option for its similar accessibility and flexibility of design, but it was ultimately rejected as 3D design software had poor performance operating in a remote environment (high latency and frequent crashing). By contrast, laser cutting design files can be developed locally in a 2D environment. CorelDRAW, a vector-based editor software, was chosen for housing development as it did not exhibit the same latency issues as Solidworks.

Our design approach was a composite structure consisting of an acrylic cartridge in a wooden case. The acrylic cartridge would be a simple rectangular box that would fit around the electronics. Notches in the walls of the box would maintain alignment and provide strength, and the walls would be glued together. The wooden case would be a more organic, handheld structure that would hold the acrylic box securely. An organic structure can be manufactured from stacking multiple thin layers, each one serving as a cross section.

Based on dimensions provided by electronics of 55 mm x 35 mm x 30 mm for the HID prototype, a prototype was designed to fit these dimensions. The prototype was sent to CMA, who provided feedback that it was too large, and it was not clear what way to orient it.

An additional issue with the prototype is that it did not fully encapsulate all the components, as mounting them on a perfboard had increased footprint. The perforated breadboard construction additionally had durability risks, and it contributed to the failure of one BNO055 sensor. For the final prototype, we took a more conservative approach in order to mitigate risk of device failure or breakage. For the final prototype, all components were mounted on a network of miniature breadboards, which increased the footprint of the electronics.

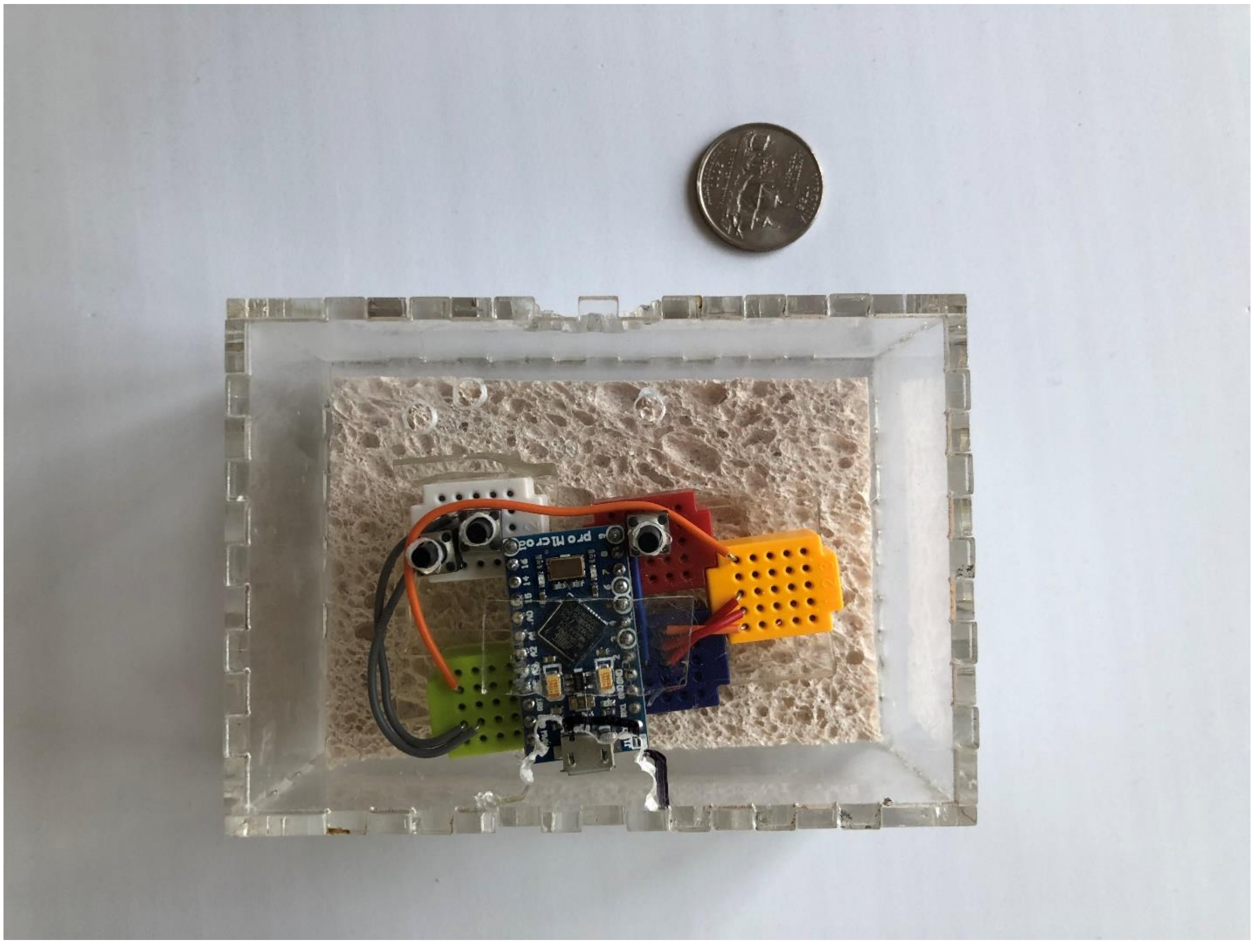


*Figure 6: Final housing developed for prototype*

For the final version of the prototype, it was critical that all components be contained in the cartridge. Extremely conservative interior dimensions of 100 mm x 70 mm x 40 mm were specified to guarantee all components would fit. Holes were machined out of the material in order to accommodate device buttons and the USB. In the final design, the wood case was abandoned as the addition of buttons reduced its utility.

The most prevalent constraint affecting housing design is size. The design must be large enough to hold all the components securely yet compact enough to fit in the hand. Electronics specifications required a 100 mm x 70 mm x 40 mm box.

The chosen material also presents a design constraint. While a laser cutter can operate at high precision in two dimensions, the precision along the z-axis is limited by the thickness of the stratifying material. Material thickness of 3mm was chosen in order to maximize smoothness while adhering to standard-width materials.

## HID Results

The HID device operation can be seen in the linked video. Per the video, it can be seen that the HID device operates similarly to a joystick; tilting the device further from the initial position causes the mouse cursor to move more quickly in the tilted direction. Simulating a mouse presents usability issues since the user cannot restrict the cursor movement to the 3D model viewer itself.

Feedback from the CMA centered around the challenges with adequately viewing the 3D model when the HID device acted as a joystick instead of the 3D model mimicking the device orientation. Uploading and switching between different 3D models also proved challenging with the HID device given the design of the website frontend. The performance of the HID device was not adequate for continued usage by the CMA because of the challenges with clicking on the model viewer while manipulating the HID device acting as a joystick. The difficulties manipulating the models with the HID device can be seen in the aforementioned video.

Size, system startup, and latency success criteria were met. Size success criteria were verified by measuring the final HID device which measured at 100mm x 70mm x 40mm which measures under the $300cm^3$ volume listed in the success criteria . System startup was verified by timing the time between HID device USB plugin and cursor manipulation; the time needed for startup was less than 5 seconds. Verifying latency performance between sensor movement and cursor movement was more difficult given the challenge of polling physical movement and pairing movement with cursor events. However, surveyed CMA staff and team members did not report observing any noticeable latency between sensor movement and cursor movement. Presented below is a tabular comparison of the HID device performance compared with the final success criterion.

| Success Criteria | HID |
|---|---|
| The displayed model will mimic the device orientation within 10 degrees | |
| The device can be held in one hand | Yes |
| The system startup operation can be completed within 30 seconds | Yes |
| The device volume will be less than $300cm^3$ | Yes |
| Communication between the sensor and MCU/single-board computer will be conducted in under 50ms using a 3ft long wired connection | Yes |
| CMA staff are able to upload and use different 3D models with with the GUI website open using less than 3 clicks | |
| Model display latency for the GUI website is under 100ms | Yes |

## BBB Software/Electronics Approach/Methodology

The failure in meeting the specified success criteria with the HID device paired with technical constraints in HID device development encouraged a transition to a different software and firmware stack. Existing model manipulation examples using the BeagleBone Black single-board computer presented a viable alternative. Developing with the BBB allowed for an integrated software, firmware, and hardware solution using Adafruit’s CircuitPython library and Flask [17][18].

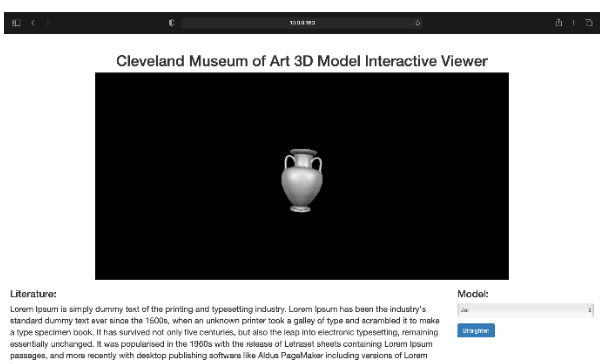
Developing on the BBB requires first interfacing with the single-board computer; using SSH to interface with the BBB is recommended [19]. Remoting into the BBB requires an active local area network, ethernet cable, and separate laptop or desktop. From there, installation of required CircuitPython and Flask frameworks is recommended. Sensor specific CircuitPython libraries are available through Adafruit which provide a means of interfacing with the BNO055 sensor without needing to develop custom firmware [7].

Installing and downloading the aforementioned libraries requires at least 2GB of available memory. The onboard flash memory of the BBB is constrained to 4GB, ~3.2 GB of which are allocated immediately to the BBB operating system. The memory limitations of the BBB onboard flash memory require memory expansion in the form of an external SD card or USB flash drive. This technical constraint increases the cost of the final product because of the need to purchase external memory storage.

The installed libraries convert the BBB into a local web server displaying 3D models using an HTML site. The Adafruit CircuitPython library includes an example named bno055_webgl_demo which interfaces output BNO055 sensor to the frontend of a local web server [7]. Modifications to the bno055_webgl_demo were made following the installation of the CircuitPython library. The bno055_webgl_demo leveraged the javascript three.js library for the display and manipulation of 3D models which is then integrated into a HTML5 site. Three.js is a javascript tool which allows developers to embed 3D model viewers into a website [20]. The three.js javascript code provided in the demo allows the user to select between multiple models stored in the webserver.

The 3D models provided in the demo were either .obj, .stl, or .htm formats and used the respective three.js file loaders. Textured and colored 3D models available on the CMA's Sketchfab account use the GLTF file format; the GLTF file format allows the model to have multiple textures applied using PNG images. Modifications made to the existing demo code were made to attempt to incorporate the necessary three.js GLTF file loader library. However, the structure of the Flask web server and CircuitPython made installing the GLTF file loader library difficult. The decision was made to convert the CMA's GLTF files into STL format in order to maintain functionality while displaying CMA specific artwork. The STL format does not allow for colorization of the 3D models but retains all spatial information.

The HTML frontend of the demo website was modified to include CMA specific information and remove extraneous information; modifications were made with the intention of facilitating the use of the BBB prototype into existing CMA workflows. Screenshots of the demo website and final website detail the changes made





*Figures 7-8: Before and after of website development*

Local web server hosting using Flask allows for users on the same network to access the HTML site and observe the same 3D model manipulations across devices is illustrated here.

The size and form factor of the BBB are significant technical constraints in creating a single-hand device; the BBB's 87mm x 54mm footprint constrain the device housing [22]. In addition, the wiring between the BBB and BNO055 sensor limits the movement of the sensor independently of the BBB computer. Lastly, the wired ethernet connection to the BBB to a router keeps the prototype tethered to an external router, inhibiting the movement of the prototype in the user's hands.

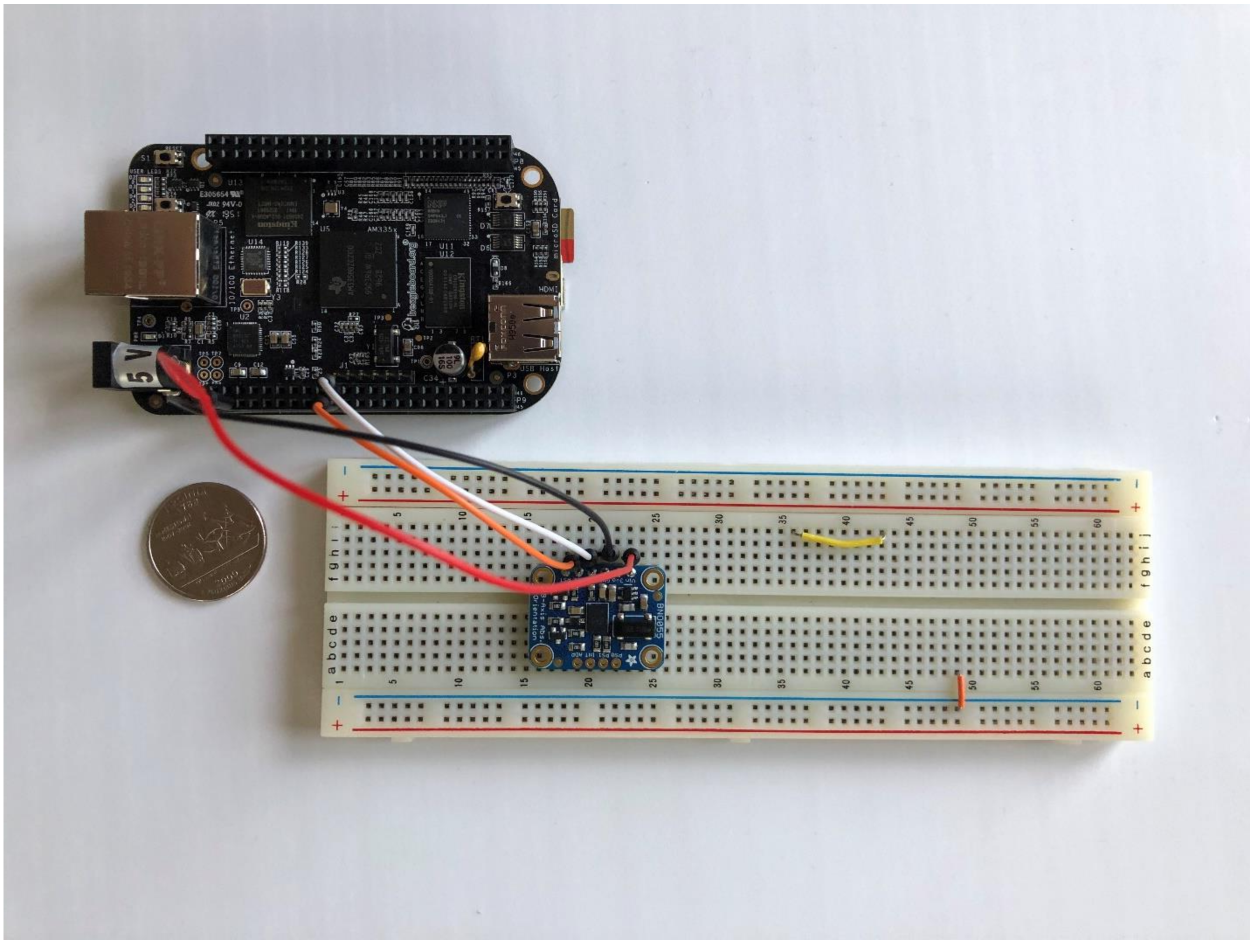


*Figure 9: Electronics developed on breadboard*

## BBB Results and Verification

The BBB prototype meets different success criteria when compared to the original HID prototype. Most significantly is the displayed model mimicking the BNO055 sensor orientation within 10 degrees. Special emphasis was placed on meeting the orientation mimicking success criteria after receiving feedback from CMA staff. These results can be verified by placing the 3D model website side by side with the BNO055 sensor and observing any significant changes in orientation between the sensor and 3D model.

The BBB prototype meets success criteria items size but fails when considering user grasp. Specifically the size of the prototyping breadboard (55mm x 170mm x 10mm) paired with the BBB computer size combine to be less than 200cm$^3$ when measured. However, difficulties holding the breadboard and BBB together without disconnecting the wiring between both lead to users requiring both hands to hold the device up. The reliance on the existing BBB and breadboard have led to the success criteria being missed.

Changes to the website frontend allowed users to switch between desired 3D models but did not allow users to upload models specifically. These results can be verified by viewing screencaps for the final website frontend. Difficulty in developing the website frontend and integrating the three.js GLTF loader has led to the success criteria being missed.

Latency once again was difficult to measure between sensor movement and 3D model movement. However, from side by side observation of the BNO055 sensor and 3D model it can be seen that the latency is noticeable between sensor movement and the displayed 3D model. Hosting the 3D model viewer site on the BBB while taking in BNO055 sensor information creates the latency shown in the demonstration video. Creating a prototype which passes the sensor information to an external web server would allow for decreased latency between sensor and 3D model movement.

| Success Criteria | BBB |
|---|---|
| The displayed model will mimic the device orientation within 10 degrees | Yes |
| The device can be held in one hand | |
| The system startup operation can be completed within 30 seconds | Yes |
| The device volume will be less than 300cm$^3$ | Yes |
| Communication between the sensor and MCU/single-board computer will be conducted in under 50ms using a 3ft long wired connection | Yes |
| CMA staff are able to upload and use different 3D models with with the GUI website open using less than 3 clicks | |
| Model display latency for the GUI website is under 100ms | |

# Bibliography and Standards Used

[1] A. Akhtamzyan, “Creating an Interactive Museum Exhibit using an IoT Joystick,” Sketchfab Community Blog, 21-May-2019. [Online]. Available: https://sketchfab.com/blogs/community/creating-an-interactive-museum-exhibit-using-an-iot-joystick/. [Accessed: 07-Dec-2020].

[2] dimtass, “Controlling a 3D object in Unity3D with teensy and MPU-6050,” Stupid Projects, 08-Aug-2019. [Online]. Available: https://www.stupid-projects.com/controlling-a-3d-object-in-unity3d-with-teensy-and-mpu-6050. [Accessed: 07-Dec-2020].

[3] K. Townsend, “Adafruit BNO055 Absolute Orientation Sensor,” Adafruit Learning System. [Online]. Available: https://learn.adafruit.com/adafruit-bno055-absolute-orientation-sensor. [Accessed: 07-Dec-2020].

[4] Bosch Sensortec, “Data sheet BNO055 Intelligent 9-axis absolute orientation sensor,” Adafruit Learning System, Nov-2014. [Online]. Available: https://cdn-shop.adafruit.com/datasheets/BST_BNO055_DS000_12.pdf. [Accessed: 07-Dec-2020].

[5] *Device Class Definition for Human Interface Devices (HID),* USB HID Standard 1.11, 2001

The HID standard, an extension of the USB standard, provides guidelines for devices that are used for human-computer interaction, including keyboards, mouses, joysticks, buttons, and more. The project is a HID device as its goal is to manipulate computer models by mimicking the user's movement. The first prototype relied heavily on this standard, as it mimicked the movement of a mouse. Although for the second prototype the user is not clicking the model directly the way a mouse would be used, as the user is still impacting the model's movement it is considered a HID device.

*Device Class Definition for HID accessed digitally from* *https://www.usb.org/sites/default/files/hid1_11.pdf*

[6] K. Townsend, "Adafruit BNO055 Absolute Orientation Sensor: Python & CircuitPython," Adafruit Learning System. [Online]. Available: https://learn.adafruit.com/adafruit-bno055-absolute-orientation-sensor. [Accessed: 07-Dec-2020].

[7] R. Dopieralski, "Adafruit_CircuitPython_BNO055," GitHub, Feb-2017. [Online]. Available: https://github.com/adafruit/Adafruit_CircuitPython_BNO055. [Accessed: 07-Dec-2020].

[8] Adafruit, " Adafruit_BNO055," GitHub, 2018. [Online]. Available: https://github.com/adafruit/Adafruit_CircuitPython_BNO055. [Accessed: 07-Dec-2020]

Adafruit_BNO055

[9] Mozilla, "Specification Positions," Mozilla Specification Positions. [Online]. Available: https://mozilla.github.io/standards-positions/#webseria. [Accessed: 07-Dec-2020].

[10] Apple, "Tracking Prevention in WebKit," WebKit. [Online]. Available: https://webkit.org/tracking-prevention/. [Accessed: 07-Dec-2020].

[11] Google, "Getting started with the Web Serial API | Google Codelabs," Google. [Online]. Available: https://codelabs.developers.google.com/codelabs/web-serial/?hl=ro#0. [Accessed: 07-Dec-2020].

[12] K. Townsend, "Adafruit BNO055 Absolute Orientation Sensor: WebSerial Visualizer," Adafruit Learning System. [Online]. Available: https://learn.adafruit.com/adafruit-bno055-absolute-orientation-sensor. [Accessed: 07-Dec-2020].

[13] NicoHood, " HID," GitHub, 2019. [Online]. Available: https://github.com/NicoHood/HID. [Accessed: 07-Dec-2020]

[14] Jimblom, "Turn your ProMicro into a USB Keyboard/Mouse - SparkFun Electronics," Sparkfun, 31-Jan-2012. [Online]. Available: https://www.sparkfun.com/tutorials/337. [Accessed: 08-Dec-2020].

[15] B. Miller, "A Closer Look at HID Class," TraceSystems, Inc. [Online]. Available: https://www.tracesystemsinc.com/USB_Tutorials_web/USB/B1_USB_Classes/Books/A3_A_Closer_Look_at_HID_Class/sbook2.htm. [Accessed: 08-Dec-2020].

[16] D. Merill and O. Matthes, "APIs for Human Input Devices," Working with USB HIDs, 2007. [Online]. Available: https://at.or.at/hans/research/nime/hid/apis.html. [Accessed: 08-Dec-2020].

[17] "CircuitPython," CircuitPython. [Online]. Available: https://circuitpython.org/. [Accessed: 08-Dec-2020].

[18] Pallets, “Welcome to Flask,” Flask Documentation (1.1.x). [Online]. Available: https://flask.palletsprojects.com/en/1.1.x/. [Accessed: 08-Dec-2020].

[19] “Getting Started,” Beagle Board. [Online]. Available: https://beagleboard.org/getting-started. [Accessed: 08-Dec-2020].

[20] three.js. [Online]. Available: https://threejs.org/. [Accessed: 08-Dec-2020].

[21] C. Nelson, “Adafruit BNO55 Absolute Orientation Demo,” Adafruit Learning System, 17-Jul-2015. [Online]. Available: https://learn.adafruit.com/assets/75568. [Accessed: 08-Dec-2020].

[22] G. Coley, “BeagleBone Black System Reference Manual,” Adafruit Learning System, 11-Apr-2013. [Online]. Available: https://cdn-shop.adafruit.com/datasheets/BBB_SRM.pdf. [Accessed: 07-Dec-2020].

[23] *I2C-bus specification and user manual,* NXP Semiconductors v.6, 2014.

The I2C standard is a two-dimensional bus that allows for communication between integrated circuits, either on the same board or on different boards. I2C is used widely throughout the field of electrical engineering in complex designs in which components need to communicate with each other. The benefit of this protocol is that it ensures that only one IC is communicating at a time while using the same wiring, allowing the device to be smaller and simpler in design. In this project, the BNO55 sensor passed its data regarding hand movements through a serial port using i2c as well as the information from the Arduino Ethernet shield. I2C was one of the main communication protocols used in this project.

*I2C-bus specification and user manual accessed digitally from* *https://www.nxp.com/docs/en/user-guide/UM10204.pdf*

[24] *Standard for Ethernet,* IEEE Standard P802.3, 2018

The Ethernet standard provides guidelines for communication over Local Area Networks (LANs) and Metropolitan Area Networks (MANs). Ethernet is used universally for computers that are physically connected to each other although it varies in speed and the size of the network. This protocol allows for fast communication and direct communication for devices within the same network. In this project, the handheld device has a wired Ethernet connection to connect to the computer it is communicating with, ensuring that latency is as low as possible.

*Standard for Ethernet accessed digitally from* *https://ieeexplore.ieee.org/document/8457469/*

[25] *USB 2.0 Specification,* USB Standard 2.0, 2000

The USB standard is one of the most widely used standards. It provides guidance for devices that are physically connected to communicate with a computer and ensures that producers use the same requirements when creating a device that connects via USB. The USB standard was used in constructing Slingstone, as the computer and the device are connected through a USB cable. On the device a USB micro-b was used, and on the computer a USB 2.0 was used.

*USB 2.0 Specification accessed digitally from* *https://www.usb.org/document-library/usb-20-specification*

[26] *glTF Specification*, Khronos Group Standard 2.0, 2017

The GL Transmission Format (glTF) standard is developed by the Khronos group and provides specifications for a common 3D model filetype that can be used across programs and devices. glTF provides the significant benefit that it is optimized for speed at both download and runtime while still preserving the architecture of the model. This specification was used to download 3D models from Sketchfab, as this is Sketchfab's standard filetype.

*glTF Specification accessed digitally from [https://github.com/KhronosGroup/glTF/blob/master/specification/2.0/README.md](https://github.com/KhronosGroup/glTF/blob/master/specification/2.0/README.md)*

[27] *STL (STereoLithography) File Format Family*, 3D Systems Standard 2.0, 1989

The STL format is a file type used for 3D models which breaks the model into small triangles. Created in 1989 and coming into widespread use in the 1990s, it is one of the earliest and still widely used file formats for 3D manipulation including 3D printing and modelling. The original standard documentation was not able to be found by the Library of Congress, but secondary documentation is still available to ensure that the specifications are met. This is a filetype used often throughout the design process, both when producing the device housing and for accessing the CMA 3D model collection.

*STL File Format information from [https://www.loc.gov/preservation/digital/formats/fdd/fdd000504.shtml](https://www.loc.gov/preservation/digital/formats/fdd/fdd000504.shtml)*

## Relevant Courses

*Table 1: Relevant courses for Diana Zavela*

| Course | Description |
|---|---|
| EECS 325<br><br>Computer Networks | Computer networks and remote servers were introduced, which helped to set up the AWS/Wordpress site. |
| EECS 223 Introduction to Data Structures | Project-based applications of coding, specifically regarding managing large amounts of information, helped with coding. |
| ENGR 210 Introduction to Circuits | Introductory circuits course. Knowledge from this course assisted with providing feedback on work completed by team members |

*Table 2: Relevant courses for Julian Narvaez*

| Course | Description |
|---|---|
| EECS 223 Introduction to Data Structures | Proper usage of Git was introduced which assisted with software/firmware version control |

| | |
|---|---|
| ENGR 210 Introduction to Circuits | Introductory circuit models were introduced which assisted during sensor and MCU/single-board computer troubleshooting |
| EECS 313<br><br>Digital Signal Processing | Digital signal processing tools were introduced which assisted with development of digital low-pass filters for smoother model behavior |

*Table 3: Relevant courses for Caroline Broady*

| Course | Description |
|---|---|
| ENGR 210<br><br>Introduction to Circuits | Basic circuits were introduced, which is fundamental to any electronics project. |
| EECS 275<br><br>Fundamentals of Robotics | Arduino development and GIT usage was introduced, which assisted in using such systems for the project |
| EECS 371<br><br>Applied Circuit Design | Provided deep experience designing and troubleshooting circuits, which assisted with debugging issues on device |