**The P3N**

**Project Overview**

[Inspiration](https://www.amazon.com/Huion-H420-Graphics-Drawing-Tablet/dp/B00MGLD3E6/ref=asc_df_B00MGLD3E6/?tag=hyprod-20&linkCode=df0&hvadid=309777534894&hvpos=1o1&hvnetw=g&hvrand=6472774712652528500&hvpone=&hvptwo=&hvqmt=&hvdev=c&hvdvcmdl=&hvlocint=&hvlocphy=9002003&hvtargid=pla-420374441173&psc=1&tag=&ref=&adgrpid=58425267301&hvpone=&hvptwo=&hvadid=309777534894&hvpos=1o1&hvnetw=g&hvrand=6472774712652528500&hvqmt=&hvdev=c&hvdvcmdl=&hvlocint=&hvlocphy=9002003&hvtargid=pla-420374441173)

Currently, drawing digitally without a mouse requires an iPad or some other sensing device that is hooked up to a program that understands 2D gestures. These systems depend on touchpads and actually drawing on it, but we are proposing to make a 3D drawing pen that can draw

1) without a pad and

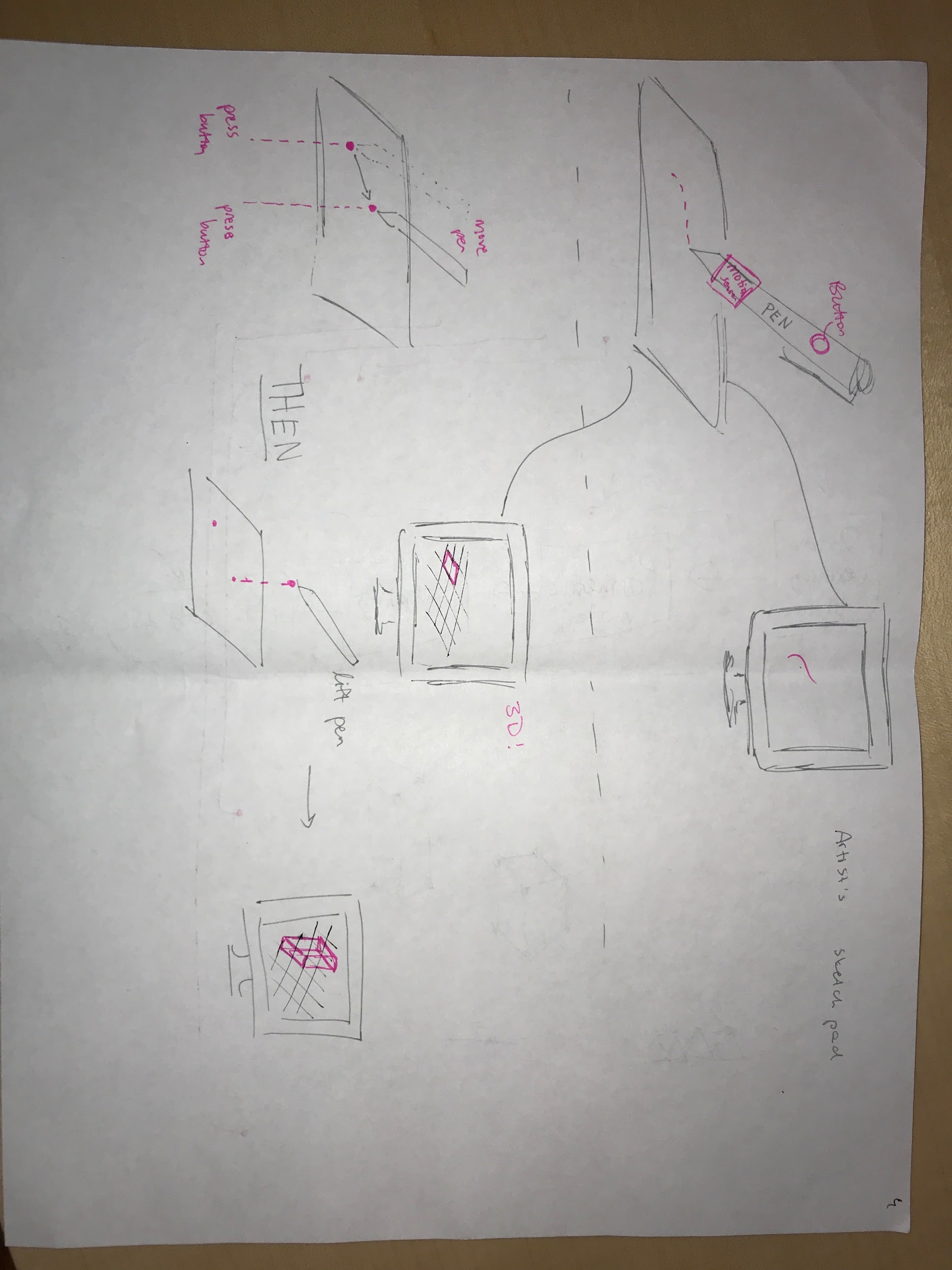
2) draw in three dimensions.

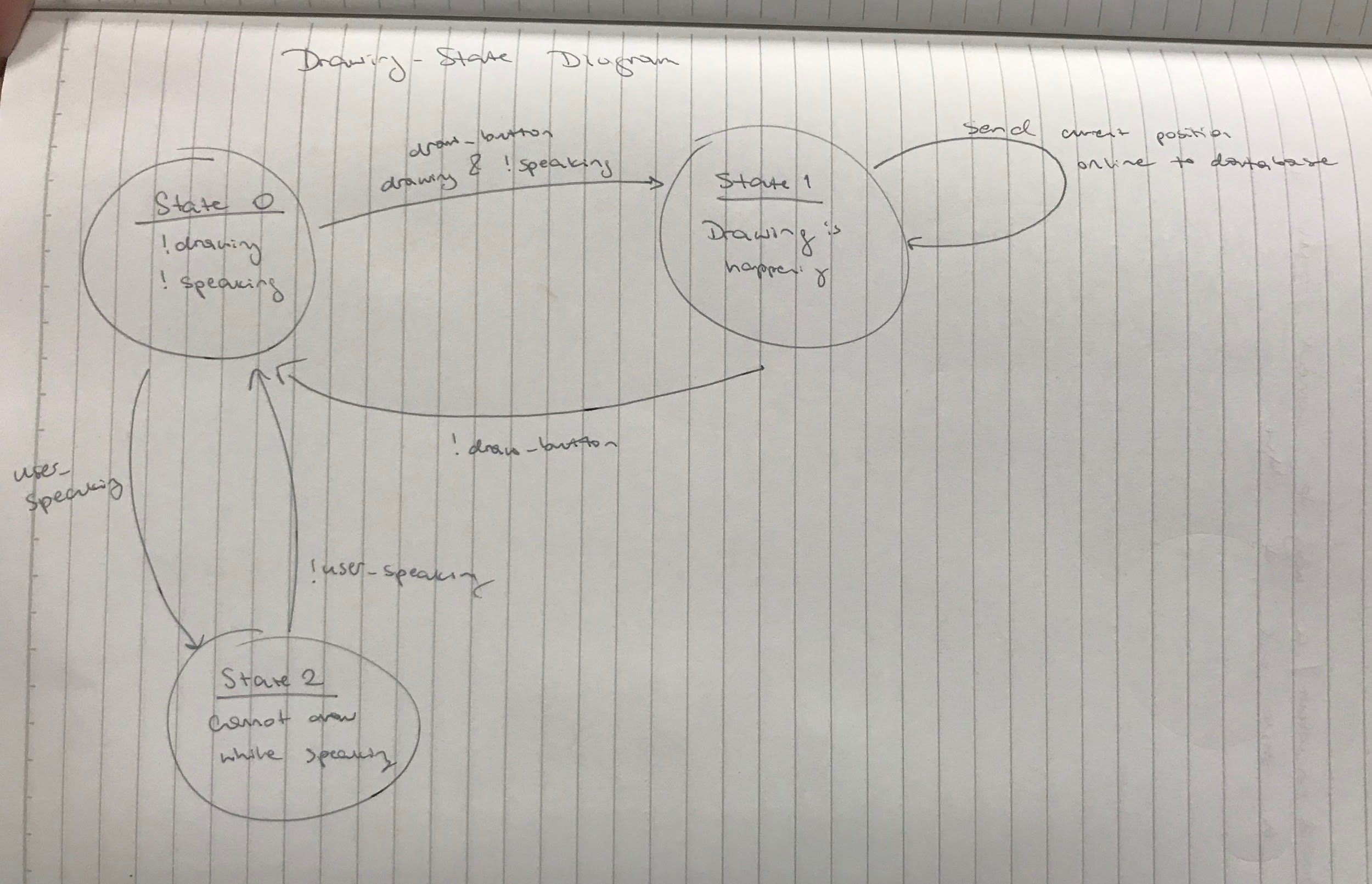
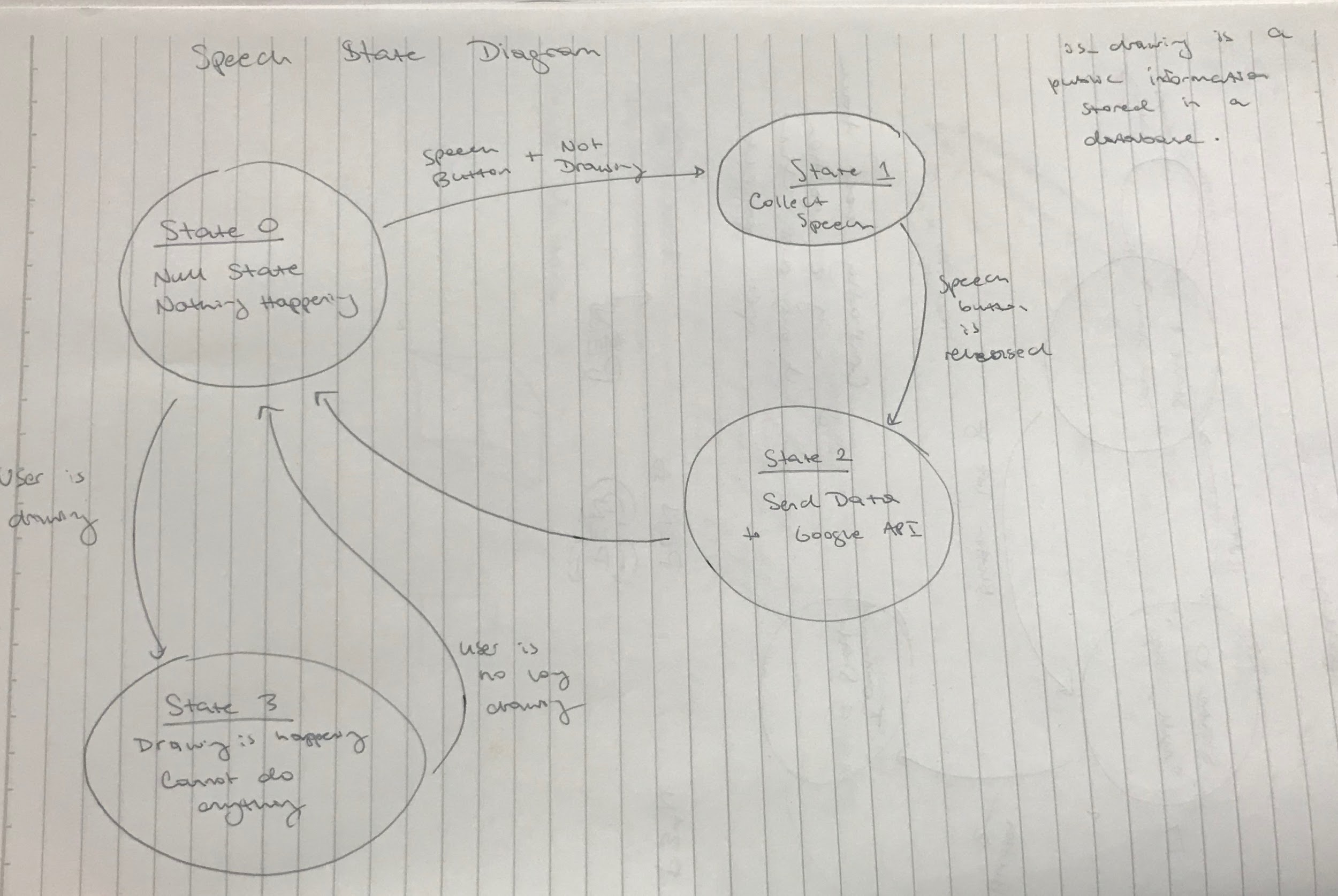
For the scope of this project we would focus on simple geometries. We are considering that the user would first draw a square or other simple shape on any flat surface (just to keep the z position consistent) and when they are done they lift the pen to extrude the square into a cube. The user would hold down the drawing button as they are drawing a line. There will also be another button that the user can hold down that will activate the microphone, with which they can make commands that will determine the color of the “ink” of the pen or the width of the lines they will draw. This concept can be applied to other geometries.

This would be useful for urban planners who want to design neighborhoods and cities quickly, or other designers in architecture and product design that are tired of having to click around and would instead like to make digital drawings with a pen.

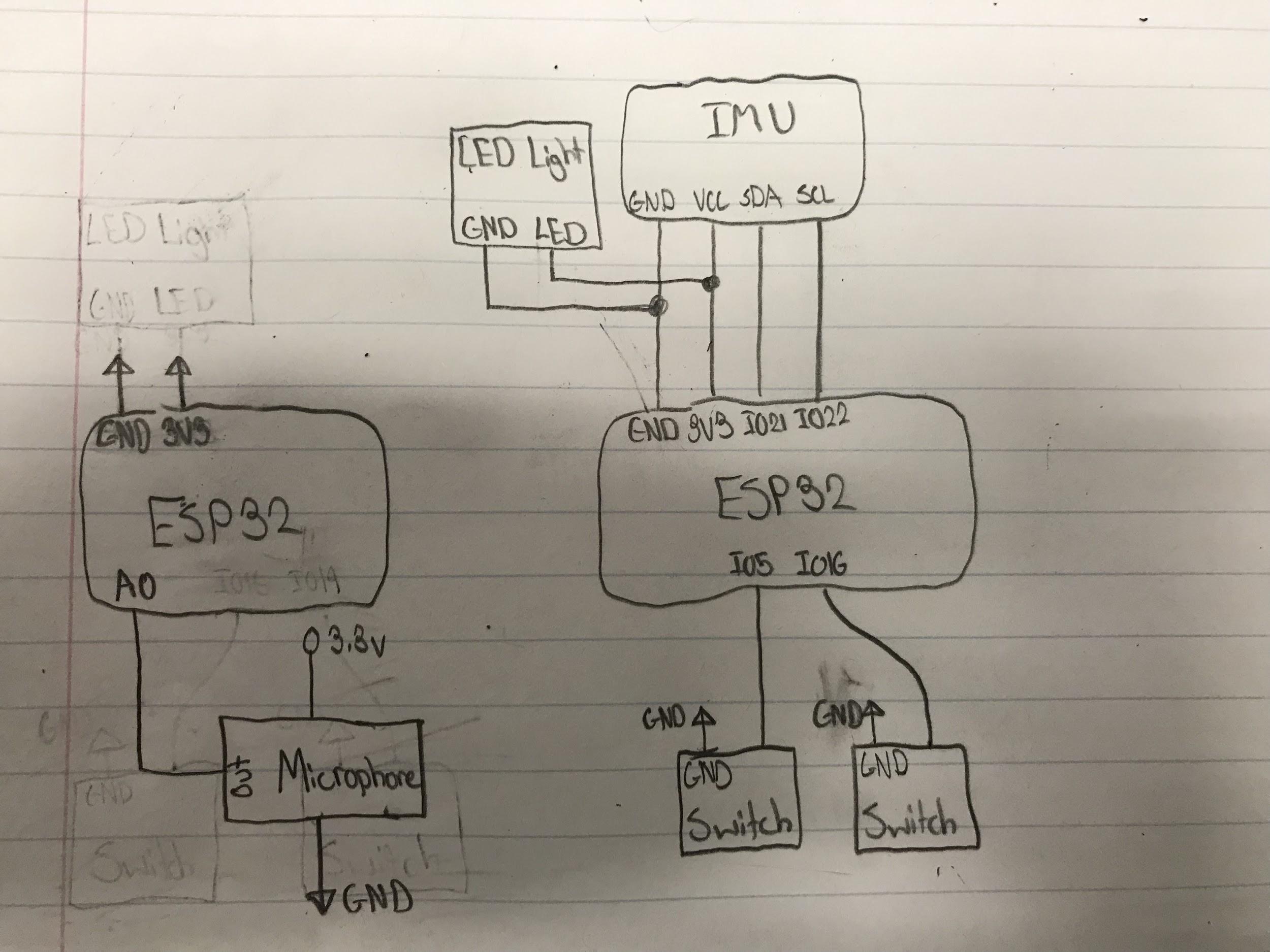
**Intended Functionality**

We will have two ESP32s, one for the drawing side of things (see state machine 1 below) and another for the speech commands (see state machine 2). This will prevent any blocking code that will potentially arise from these conflicting processes. We will use an IMU to measure the coordinates of the tip of our pen, the two aforementioned buttons, and 2 led lights that will indicate whether the user is in the drawing or speech state. These components will be packaged into a pen component that will be wired to a breadboard that will hold the ESP32s and microphones and connect to power. The board will also have an empty working space that will act as the drawing pad.

*The P3N* 

*State machine 1**State machine 2*

*System Block Diagram showing the two ESP32s*



**Technical Challenges**

The greatest technical challenge we expect to face during this project is rendering 3D objects on our website. We want to build a simple website that will act as the canvas for the project. However, this means that we must invest a lot of time and work in figuring out ways to render the 3D drawings. Many JavaScript libraries exist that can aid in rendering 3D objects, but naturally this will require a lot of work in post-processing data in order to correctly render an object being drawn by the pen.

We also expect some difficulty with keeping track of the position of the pen. Since we’re drawing images in three dimensions, we’ll need to keep track of the X, Y, and Z positions. We plan on using the IMU’s acceleration data in order to calculate this positional data; however, this can lead to some issues with either our calculations not being accurate enough, a lot of noise if someone is drawing shakily, or even issues in the acceleration data since the pen will be held at an angle.

As we said earlier in this proposal, we plan on using two ESP32’s for this project, one which controls the pad - which will contain speech control - and the other for the pen - which will be how the user draws on the canvas. We want to ensure that there is correct communication between these two microcontrollers - either through the API to each other, or through the API to the website. We also plan on expanding this to include multiple users, which we expect to have some technical difficulty due to the fact that we have to check for conflicts among user drawings as well as handle lots of incoming data and a large database.

**Parts List**

From the 6.08 lab kit provided to us throughout the course, we will be using two   
ESP32 microcontrollers, the breadboard, the IMU, and the microphone. In addition to those items, we will be purchasing 4 buttons (two are needed, and two will serve as replacements), a long and skinny breadboard to be used for the pen, and an LED light which will notify the user of the different states that the pen is in.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Part Name | Description | Item Number | Vendor | Price |
| Skinny Breadboard | The skinny breadboard will act as the breadboard inside the Pen and will hold the IMU, two buttons, and an LED light. | 65  <https://www.adafruit.com/product/65> | Adafruit | $4.00 |
| Led Light | The LED light will serve as a system of notifying the user of the multiple different states that the system can be in. | 297, 301  <https://www.adafruit.com/product/297>,  <https://www.adafruit.com/product/301> | Adafruit | $16.00 |
| Buttons | Two of these buttons are needed, and two will serve as replacements. The buttons act as better buttons (more consistent) than those provided in class | a14111800ux0250  <https://www.amazon.com/Uxcell-Momentary-Tactile-Button-Switch/dp/B00VG702AY/ref=asc_df_B00VG702AY/?tag=hyprod-20&linkCode=df0&hvadid=216516247812&hvpos=1o2&hvnetw=g&hvrand=7229800172744920435&hvpone=&hvptwo=&hvqmt=&hvdev=c&hvdvcmdl=&hvlocint=&hvlocphy=9002000&hvtargid=pla-367172054382&psc=1> | Amazon | $3.25  (Quantity: 2) |