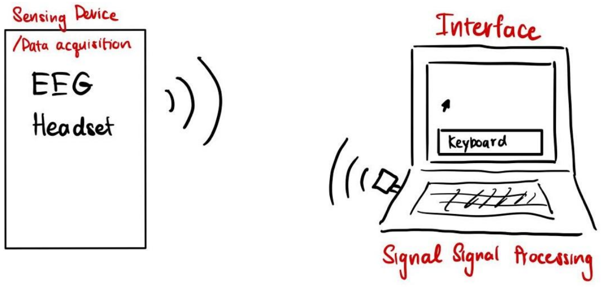
NeuroController

Team A0: Jean, Jonathan, Wendy

Application Area

Goal: Increase accessibility of computer interfaces for amputees and individuals with difficulty controlling muscle functions.

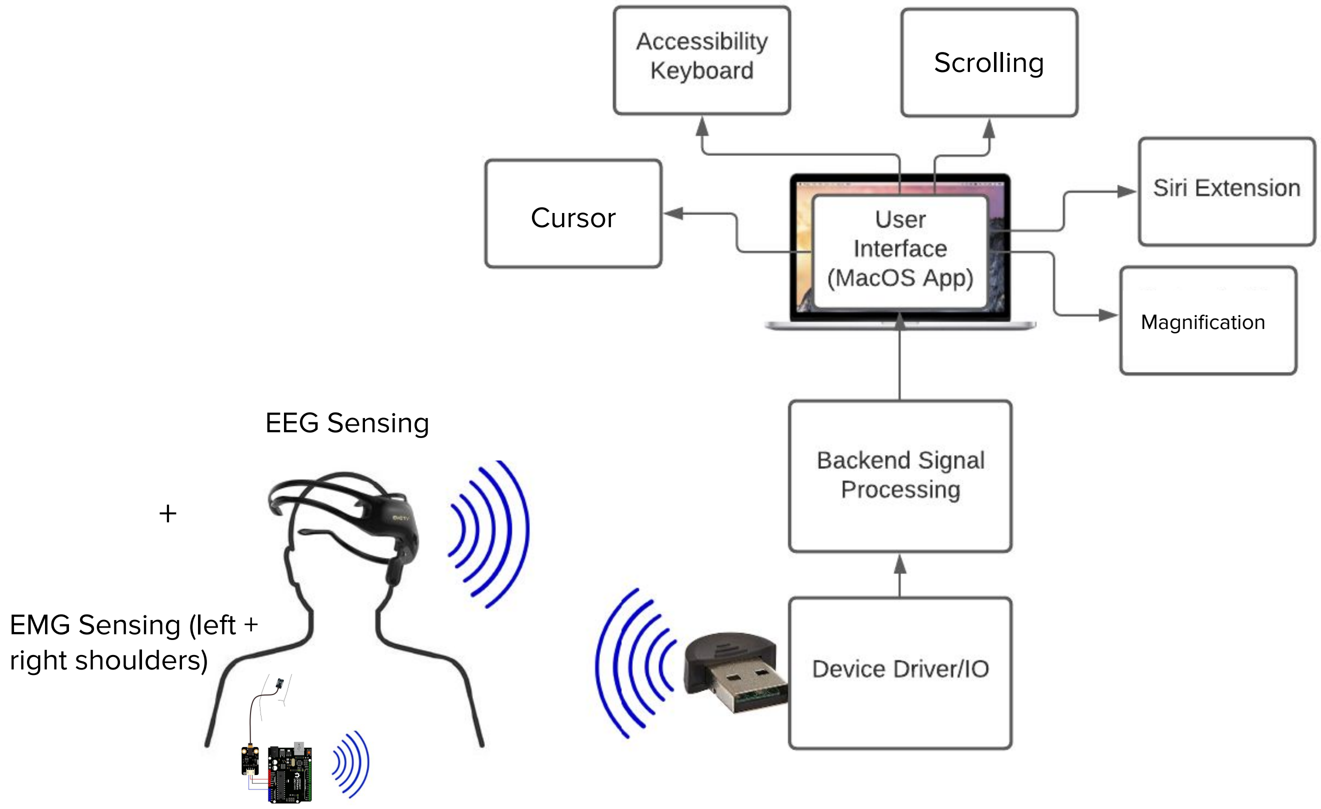
Our solution: Create an application that serves as the mouse and keyboard interface that is controlled through EEG (electroencephalogram) and EMG (electromyography) signals.



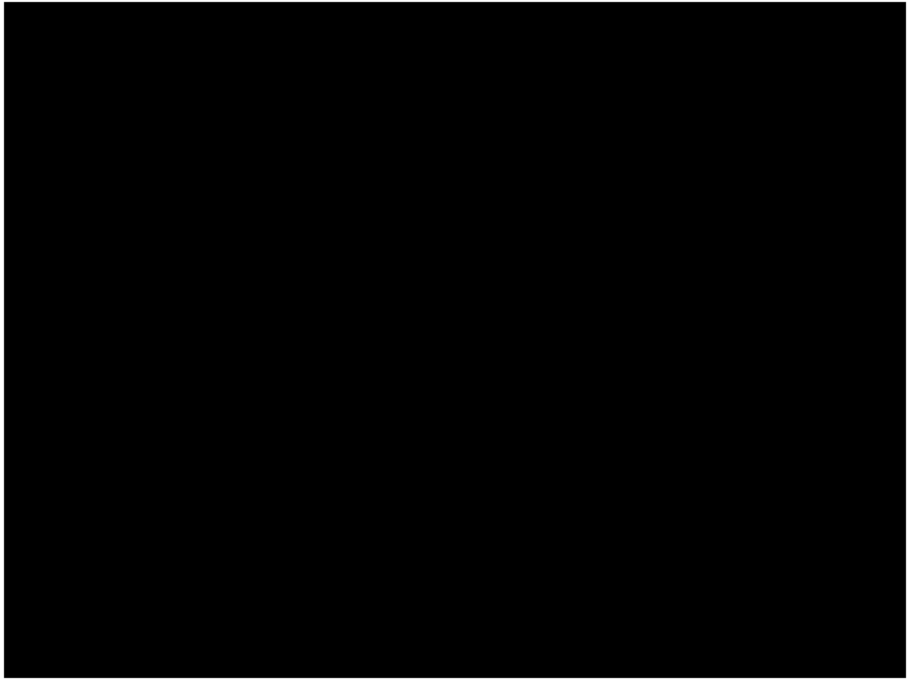
Use Case Requirements

- Aim for above 90% accuracy in converting user intentions into the corresponding output within the interface
- Desire for a user to on average score 1s on reaction time in the human benchmark test using our device
- Require that our user hit three targets within 60 seconds

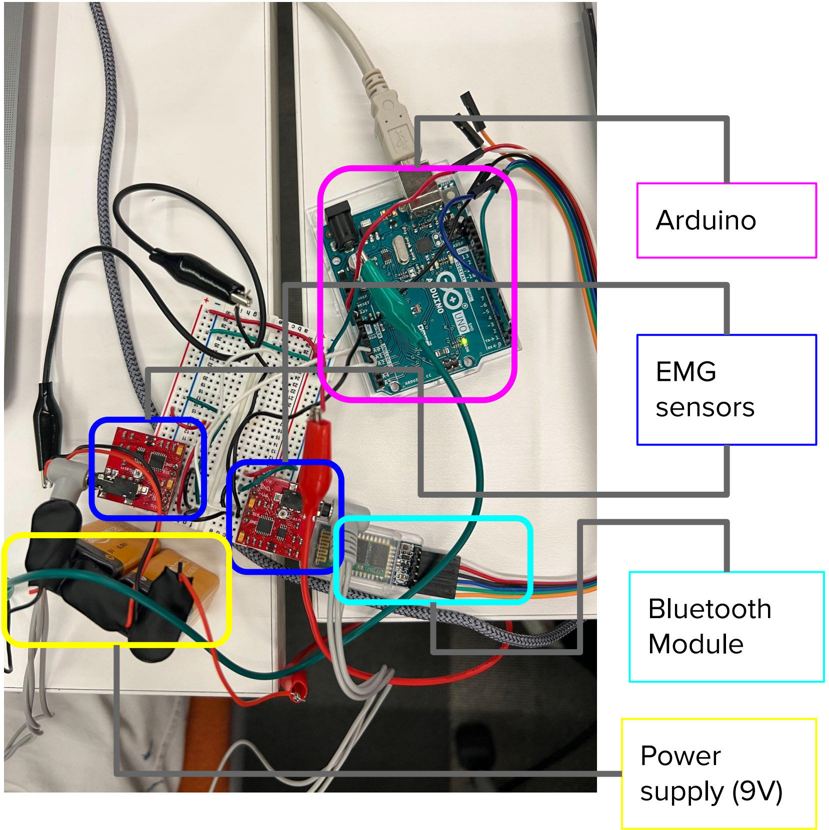
Solution Approach



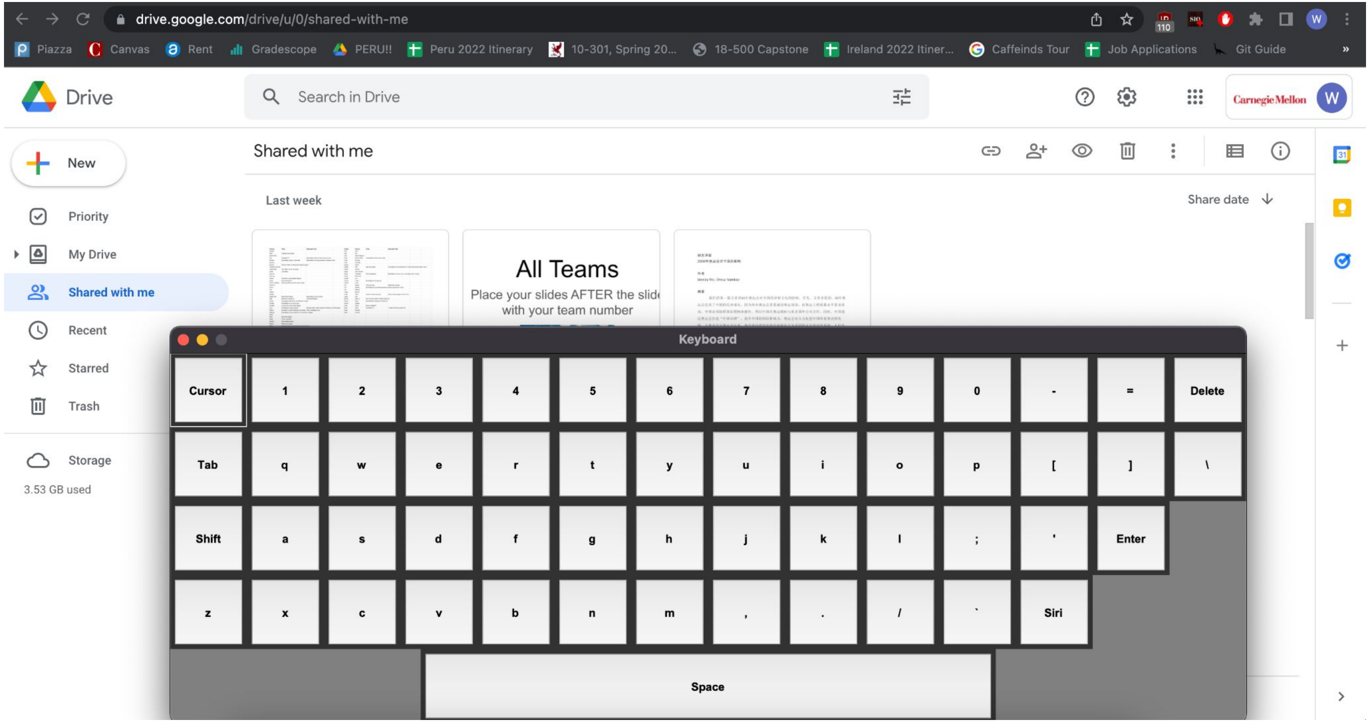
Complete Solution - EMG



EMG controlling cursor



Complete Solution - Frontend



Human Action	Application Event
Double blink	- Left click in cursor (mouse) mode - Key press in keyboard mode
Right wink	Toggle between horizontal and vertical controls
Left EMG	- Moves cursor left or down depending on which toggle is set in cursor (mouse) mode - Moves focus key on keyboard left or down depending on which toggle is set in keyboard mode

Metrics and Validation

Requirement	Testing Strategy	Quantitative Metric
User Latency	Human benchmark test	Time user reaction to a stimulus and display reaction within 1s (clicking)
User Accuracy & Speed	Point and click test	User can click 3 randomly spaced static targets within 60 seconds
Overall Intention Accuracy	Task test	User follow the procedures of a given task (eg. Typing “Hello”, closing window)
EEG Signal Classification Accuracy	Signal pattern recognition test	90% accuracy on identifying labeled test data
User Experience	Ask users who have tried this product if they would recommend this to someone in need	Over 70% recommendation out of 10 people

Metrics and Validation Results

EEG

1. User Latency
   a. 1627 ms reaction time
2. User Accuracy and Speed
   a. Takes about 30 seconds to reach a target location and click
   b. Only using one-directional movement and winking to move

Integrated (EEG + EMG)

1. User Accuracy and Speed
   a. Type two words: ‘cmu’, ‘dog’
   b. Average time per word: 72 seconds

Results - EEG Signal Processing

- Models used in final design each individually predict with error between 0% to 20% on validation data
- On average, prediction error is below average 10% across all models.

Per Model prediction error:

Event Occured (LR)	Noisy (LR)	Blink (LR)	Right Wink (LR)	Left Wink (RF)	Double Blink (RF)	Left vs Right (LR)
6.31%	1%	2.53%	2.61%	20%	9.5%	0%

Live Predictor Error (from live sampling):

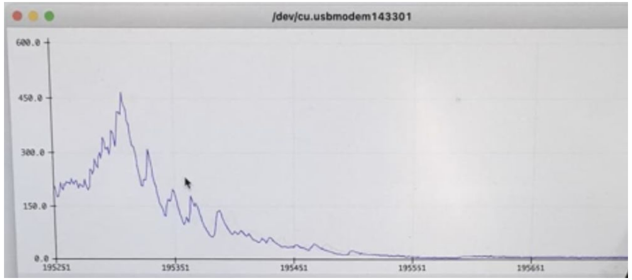
Left Wink	Right Wink	Double Blink
10%	15%	30%

Results - EMG

- Raw EMG data are rectified, amplified, and smoothed by EMG sensor
- Arduino digitized and mapped the EMG values to between **0 to 1023**
- Reasonable values for the resting state is from **~ 0 - 200**

Calibration criterion

- 1000 data values (~ 4.7 sec)
- standard deviation < 20



Movement event trigger :

- Data buffer of 50 data points in a buffer –> move 10 pixels
- Standard deviation: > 20
- Threshold ratio = $\frac{\text{average of values in real-time data buffer}}{\text{average value from calibration}} > 1.5$

Design Trade-offs

- **Three second sliding window**
  - Three seconds allows the model to collect enough data to predict
  - Easy to generate a sample in 3 seconds for training
- **Removing RPi offshoring of ML and signal processing**
  - Speed of processing
  - Minimized intensive compute
- **Removal of FFT processing**
  - The features that we had chosen had produced a satisfying result of 90%
  - Increase the speed of EEG classification
- **Integrating EMG on top of EEG**
  - Allows continuous control data that is not physically exhausting for the user
  - Have to implement on the bluetooth module
- **Python application vs. Chrome extension or iOS Application**
  - Greatest experience
  - Chrome extension limits our interface to one application and there are challenges with porting existing Python code to Javascript
  - Allows users to use the application for any app (eg. Notes, Chrome)

Difficulties Faced and Solutions

- **Difficulty** - EEG headset
  - Distinguishment of features with the sliding window
  - Generalization to multiple users
- **Solution** - create an FSM to control when to use which predictor so we have more reliable real time prediction and collected more data so models can be better generalized
- **Difficulty** - EMG baseline variance
  - Difficulties obtaining steady EMG signals necessary to hit our threshold from different people
- **Solution** - Calibrate by averaging the EMG baseline values for 5 seconds with standard deviation as the criteria to ensure the resting state

Work Remaining

- Tuning
  - Test and tune on appropriate cursor movement vs. event detection for a user-friendly interface
- Test with more subjects
  - Test EEG and EMG integrated interface with more subjects
  - Test on overall intention accuracy
- Frontend
  - Add a scrolling feature
- Final Poster/Video/Report

	3/28/22	4/4/22	4/11/22	4/18/22	4/25/22	5/2/22
Deliverables				SLACK	SLACK	
Proposal Presentation Slides						
Proposal Presentation						
Design Presentation Slides						
Design Presentation						
Final Presentation Slides				All		
Final Presentation				WM		
Public Demo						All
Final Report						All
Logistics						
Ordering all components						
Ethics Section						
Weekly Status Reports	All					
Individual Status Reports	All					
Signals						
Design & implement signal processing algorithm + calibration algorithm	JK					
Test on EMG signals acquisition between sensor-arduino-pc						
EMG calibration	JU					
Hardware Drivers						
Create a device driver/acquisition system for OS/Pi	JK					
Set up Bluetooth communication between computer and Arduino		JU	JU			
Interface (Software)						
Setting of RPi (expansion)						
Testing communiciations between EMG and interface (integration & debugging)		WM, JU				
Research how to integrate Python application to desktop screen	WM					
Move to integrating a keyboard to any desktop app		WM (implement)				
Integrate EMG and EEG with desktop app			All (integrate)			

Most updated schedule