

Summer 2018

# EEG Experiment Scripting Tool for Novice Programmers

Sarah Walling-Bell  
swallingbell@pugetsound.edu

Follow this and additional works at: [https://soundideas.pugetsound.edu/summer\\_research](https://soundideas.pugetsound.edu/summer_research)



Part of the [Cognitive Neuroscience Commons](#), [Programming Languages and Compilers Commons](#), and the [Software Engineering Commons](#)

---

## Recommended Citation

Walling-Bell, Sarah, "EEG Experiment Scripting Tool for Novice Programmers" (2018). *Summer Research*. 334.  
[https://soundideas.pugetsound.edu/summer\\_research/334](https://soundideas.pugetsound.edu/summer_research/334)

This Article is brought to you for free and open access by Sound Ideas. It has been accepted for inclusion in Summer Research by an authorized administrator of Sound Ideas. For more information, please contact [soundideas@pugetsound.edu](mailto:soundideas@pugetsound.edu).

# EEG Experiment Scripting Tool for Novice Programmers

Sarah Walling-Bell\* David Chiu\* David Andresen†

\* Department of Mathematics and Computer Science

† Department of Psychology

University of Puget Sound, Tacoma, WA 98416

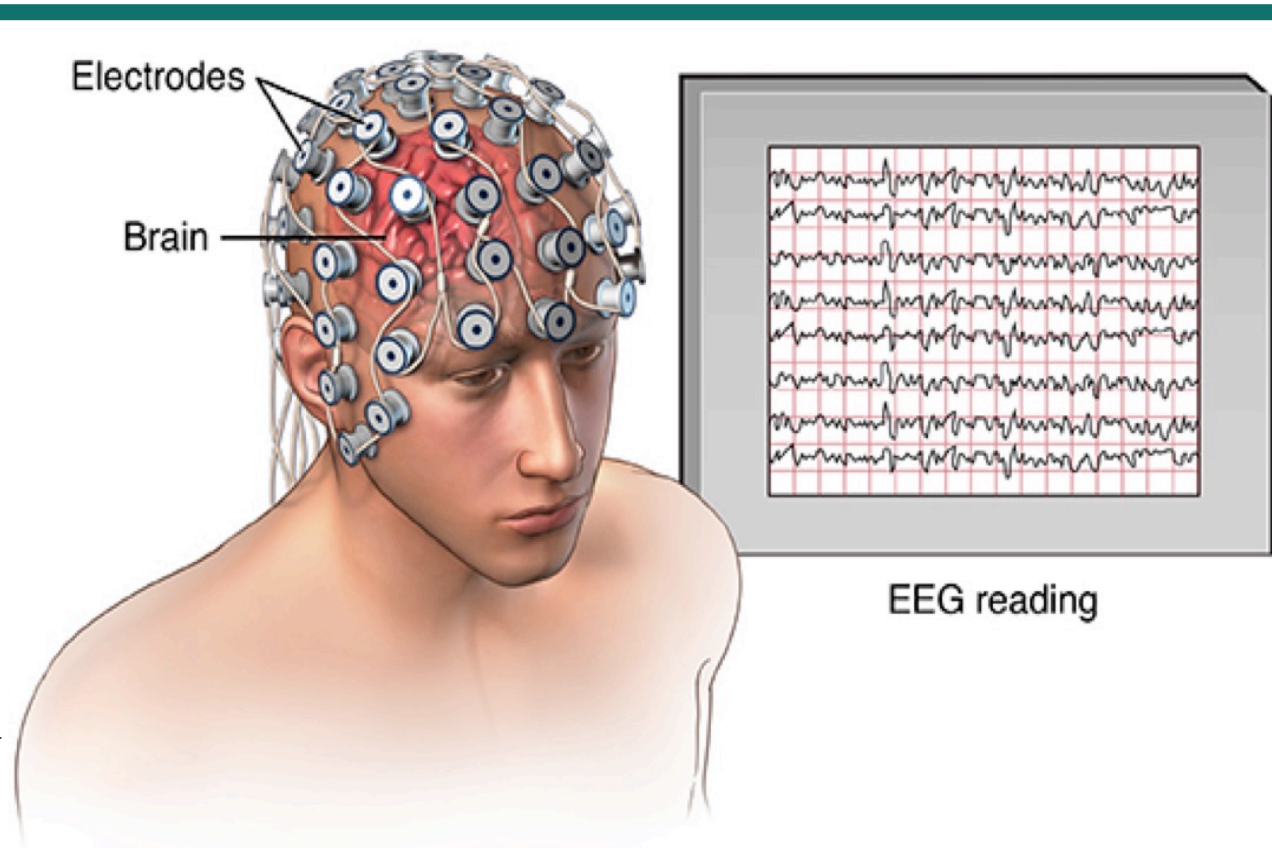
## Abstract

Most university neuroscience and psychology students are not required to learn programming in their curriculum, and so miss out on a fundamental aspect of modern brain and behavioral research: learning to organize and manipulate their stimuli algorithmically. This is the case in the University of Puget Sound (UPS) Electroencephalography (EEG) lab, where students researching how electrical activity in the brain responds to stimuli (e.g., images) are unable to program their own stimuli.

In this research I developed a software tool that enables novice programmers to build EEG experiment stimuli scripts without first learning a programming language. Future development of this tool will aim to include an educational component by teaching users basic programming concepts.

## What is an EEG?

An Electroencephalography (EEG) machine measures regional brain activity using sensitive electrodes. Researchers often measure a subject's brain activity in response to stimuli, such as sounds, images, or words.



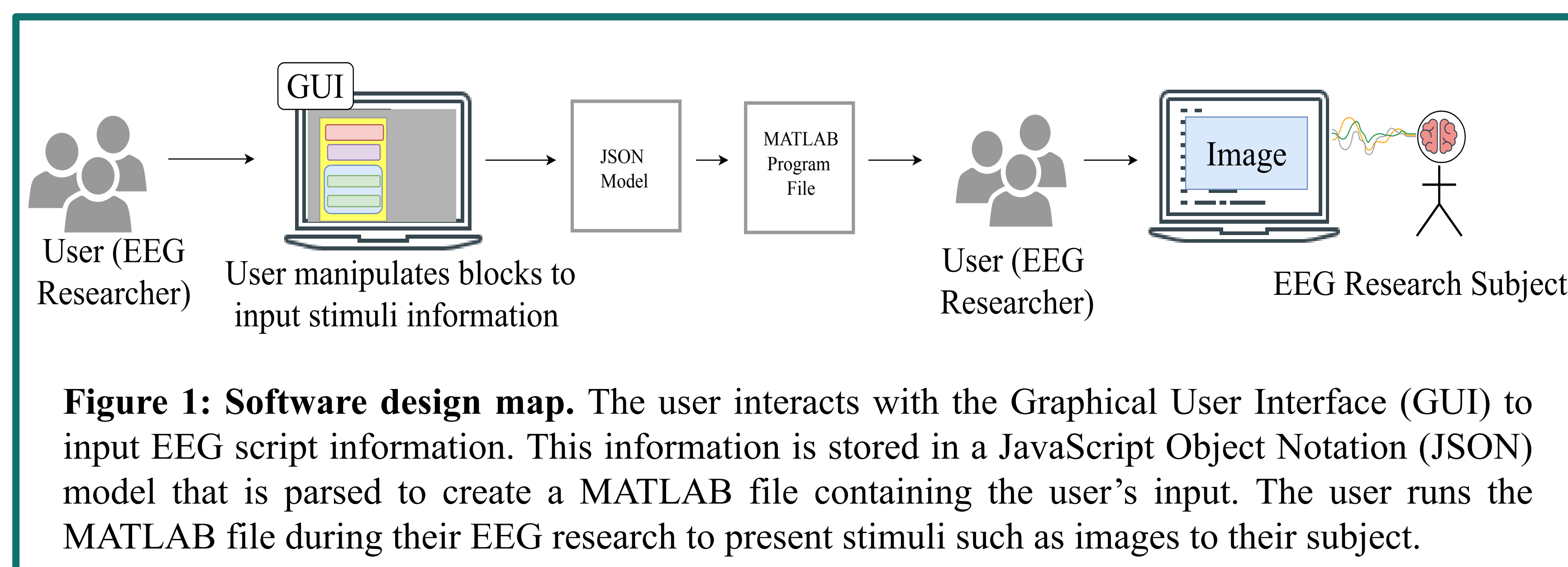
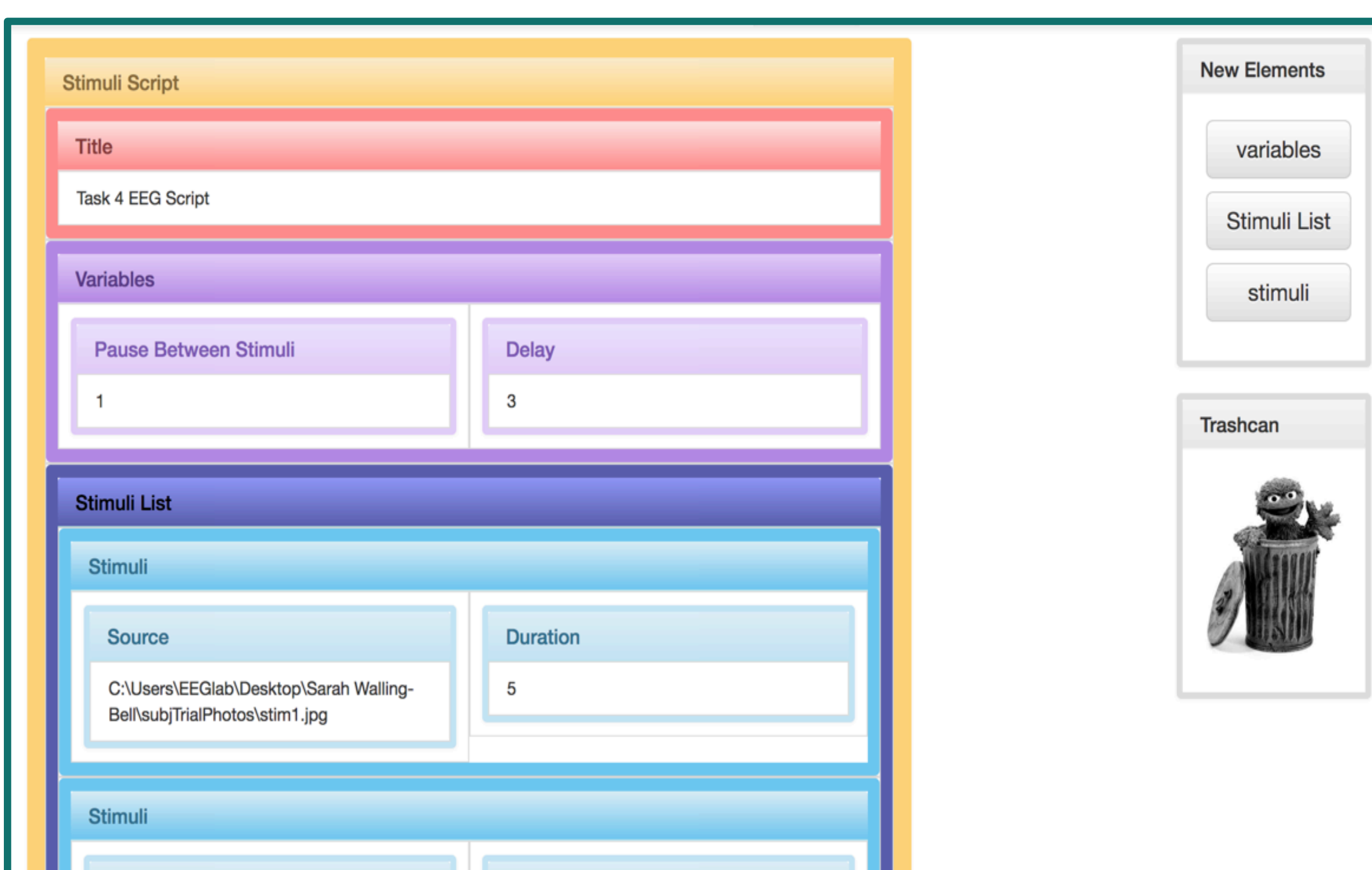
## Background

- Most undergraduate neuroscience and psychology students do not learn a programming language and so do not code their own EEG stimuli scripts.
- At UPS one psychology professor codes all student EEG scripts in the programming environment MATLAB, which takes ~8 hours per study. This problem is not unique to UPS, but rather a general and increasing trend across universities.
- I designed a software tool to address this problem, and make EEG experiment scripting accessible for novice programmers.

## Methods

The software I am developing consists of a Graphical User Interface (GUI) that the user interacts with to sculpt an EEG experiment script. The input collected from the user in the GUI is stored in a map, which is then parsed by the software to export the proper MATLAB scripts.

The GUI is written in AngularJS (JavaScript, HTML, and CSS) and exports a JavaScript Object Notation (JSON) model containing the user input held in the colorful nested blocks. This information is parsed in order to build the desired runnable MATLAB script.



## Experimental Setup

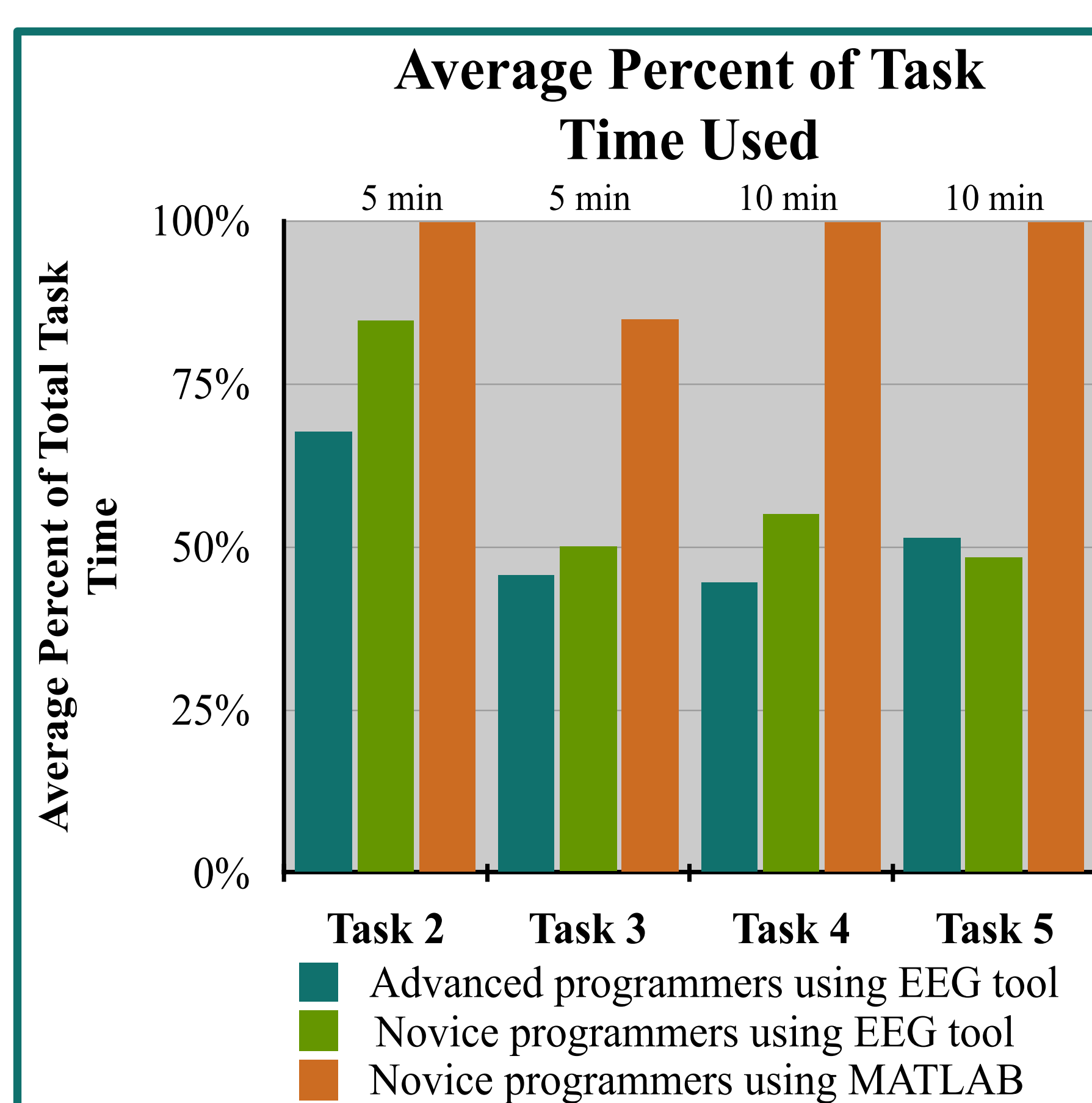
23 participants spent up to 60 minutes completing a series of 5 EEG code manipulation tasks that increased in difficulty. 20 participants (17 novice and 3 advanced programmers) used my EEG scripting software tool, while 3 made changes directly to the MATLAB script. Subjects were given time limits for each task and asked to wait for several minutes before asking questions when they were stuck. Students rated their frustration levels, and those using the EEG software tool gave user feedback. The subject's computer screen was recorded during their session, along with the amount of time they spent on each task.

## Results

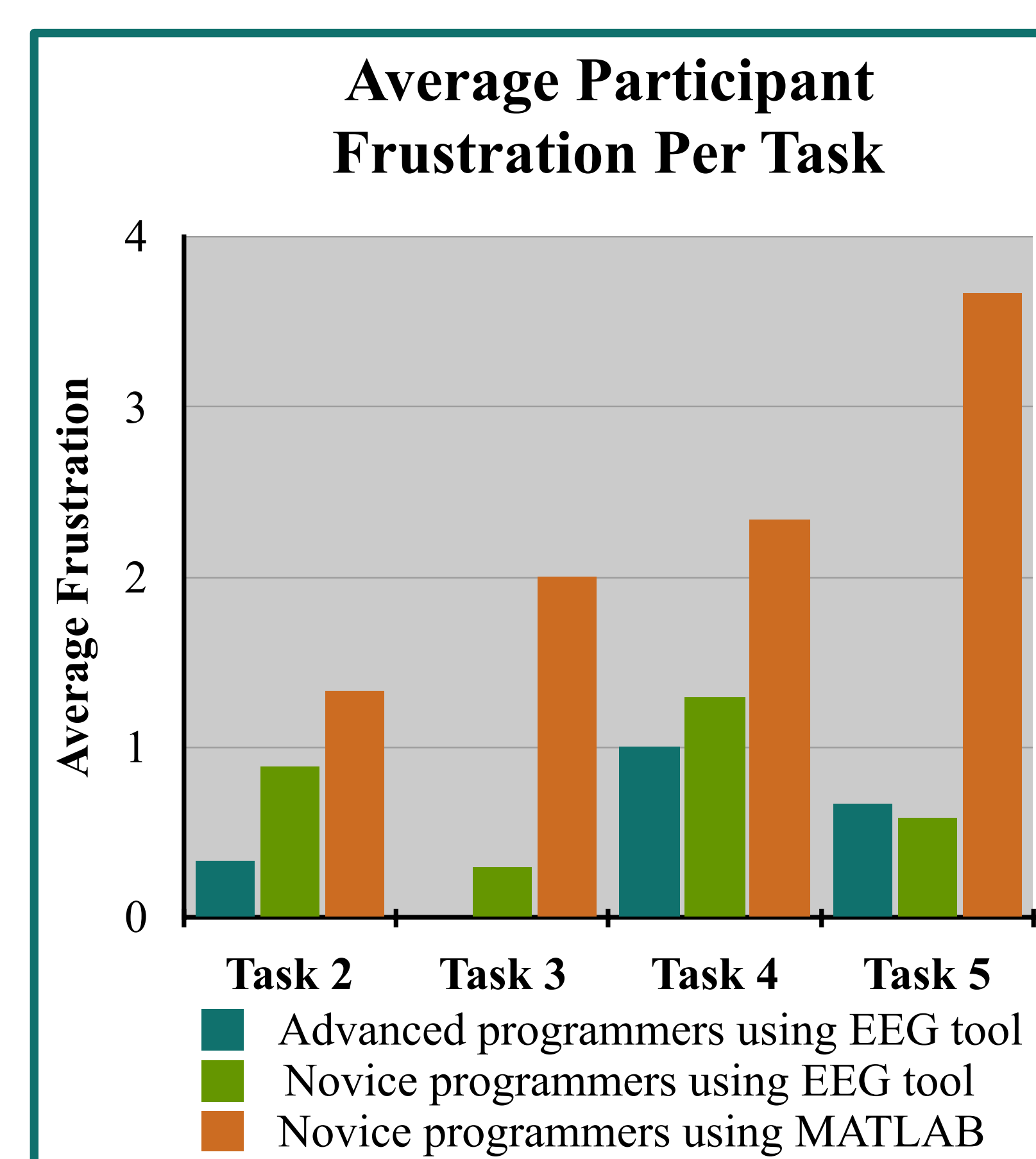
Participant Percent Success Rate Per Task

	Task 2	Task 3	Task 4	Task 5
Advanced programmers using EEG tool	100%	100%	100%	100%
Novice programmers using EEG tool	100%	100%	100%	100%
Novice programmers using MATLAB	0%	33%	33%	0%

**Figure 2.** While all participants who used the EEG Scripting software finished every task, participants who used MATLAB were routinely unable to complete the tasks.



**Figure 3.** The average participant using the EEG scripting tool often did not use the full time allotted per task, however participants working directly with MATLAB routinely ran out of time.



**Figure 4.** The average amount of frustration reported by participants during each task was highest when not using the EEG scripting software.

## Discussion

### Quantitative

- The EEG scripting tool enabled all programmers to complete all code manipulation tasks.
- No participants could build EEG scripts from scratch without using the EEG scripting tool.
- Participants using the EEG scripting tool were less frustrated while completing the task than participants who did not use the tool.
- The EEG scripting software makes coding MATLAB scripts more time efficient for novice programmers.
- Even in early development, the EEG scripting software is more user friendly and intuitive for novice programmers than MATLAB, and it enables novice programmers to build and manipulate EEG experiment scripts without first needing to learn a programming language.

### Qualitative: Common and Key Critical User Feedback

- Allow users “to manually delete the stimuli, no need to drop them in the box.”
- “It would be cool if you could edit (at once) multiple stimuli to be the same.”
- “Have the default words disappear as soon as the ‘edit value’ box is clicked so you can start typing right away.”
- “Be able to type directly into the colored boxes.”
- “A ‘copy script’ button would be really helpful and time saving.”
- “Have the ‘New Elements’ and ‘Trashcan’ boxes scroll when I did.”

## Future Development

- Utilize a Document Object Model (DOM) to increase software extensibility.
- Create a “Load-In” function for generating GUI models for many stimuli, and a “Save” function to be able to continue editing previous MATLAB scripts.
- Add “Select File” option, so that user doesn't need to know path to each stimulus. Expand stimuli types to include audio, video, text, and multimedia.
- Improve GUI graphics: limit where each item type can be dragged, create “Add Stimuli” button, create info boxes that appear when hovering over the GUI, replace “Edit Value” button with direct input into the nested blocks, & change colors to accommodate users with color blindness.

## Acknowledgments

Thank you to my research advisor Professor David Chiu for joining me in a summer of problem solving, investing so much time into researching and learning about the languages and software I utilized, and guiding me through the software development process.

Thank you to Professor David Andresen for motivating this project, providing important insight into user needs, and helping me maintain a long-term view of the research. Additionally, thank you for granting me access to computers and software for subject trial testing.

I would like to express my appreciation to the Sherman Fairchild Foundation for funding this research.