5-2023

Culture in Computing: The Importance of Developing Gender-Inclusive Software

Creighton France

Culture in Computing: The Importance of Developing Gender-Inclusive Software

An Undergraduate Honors College Thesis

in the

Department of Computer Science and Computer Engineering
College of Engineering
University of Arkansas
Fayetteville, AR
April, 2023

by

Creighton France
csfrance@uark.edu

Abstract

The field of computing as we know it today exists because of the contributions of numerous female mathematicians, computer scientists, and programmers. While working with hardware was viewed as “a man’s job” during the mid-20th century, computing and programming was viewed as a noble and high-paying field for women to occupy. However, as time has progressed, the U.S. has seen a decrease in the number of women pursuing computer science. The idea that computing is a masculine discipline is common in the U.S. today for reasons such as male-centered marketing of electronics and gadgets, an inaccurate representation of what it means to be a “programmer” in media, and the fact that math confidence has been found to be lower in young girls compared to their male counterparts. Encouraging gender minorities to pursue computing is a complex task, and there are many proposed solutions. Many of these plans focus on changing attributes of those in the minority instead of working with the dominant group.

As there are trends in the way people of a certain gender interact with technology, there are trends in the barriers some users may face when interacting with software products. Software developers can use existing frameworks such as GenderMag to investigate gender accessibility for specific functions and workflows. Improving gender accessibility is not only useful from a software design standpoint, but it can be positioned as a useful tool to increase the presence of gender minorities in computing. This project discusses the connection between gender and computing culture and also provides an application of the core principles of the GenderMag framework to gain insights on potential pain points of a software product with respect to gender.

Contents

Acknowledgements

I would first like to thank my mentor, Dr. Susan Gauch, for working on this project with me and assisting with everything from experimental design to IRB approval. Even though this project addresses a completely different research topic than I had originally proposed, she helped this project reach the finish line.

I would also like to thank the University of Arkansas Honors College for funding this project through the Honors College Research Grant.

Introduction

Motivation

Computing and gender have been coupled since the beginning of computing itself, and women have made some of the most crucial contributions to the field [1]. Over the last few decades, the field of computer science has been perceived as a “masculine” field by many [2], which has created a barrier for women and non-binary people in the discipline [3]. The lack of gender diversity in computing education has been a topic of particular interest in recent years [4]. There have been many efforts to increase gender diversity in computing. For example, Girls Who Code is an organization that aims to encourage and teach girls how to code and feel more confident and capable in STEM fields. However, Girls Who Code and other organizations geared towards women have received criticism for centering their solution on changing women rather than changing STEM culture itself [5].

Because one’s gender can have an influence on the way in which they solve problems and implement solutions [6], software products that have been designed with a typically “masculine” way of thinking could cause frustration for gender minorities and discourage them from developing or exploring an interest in computing. There are already methods to rectify these disparities such as the GenderMag framework [7] that developers can use to improve gender accessibility in software workflows. Facilitating positive interactions between gender minorities and technology has been shown to increase the presence of women and queer people in computing [8]. Because of this, intentionally creating gender-accessible software products is a promising approach to improve gender diversity in computing.

Intentionally improving the usability of software products could increase the number of positive interactions between gender minorities and technology, but it could also alter culture in the field of computing by encouraging developers to consciously consider the experiences of many different groups of people. Because there is not typically a strong focus on gender accessibility in traditional computing education [3], developers may not know where to begin when attempting to improve their products and make them more accessible and usable by everybody.

Goals

In this paper, I primarily aim to discuss why it is important to be knowledgeable of the relationship between gender and computing. I hope this will encourage software developers to consider the ways in which their own perspective may be limited and how these limitations could prevent their software from being used to its full potential by everyone. I will discuss GenderMag, an existing framework for improving gender accessibility in software, and show how its core principles could be applied to a software accessibility survey to improve software workflows with gender in mind. I believe that this real-world study could serve as an example of a feasible method to evaluate gender accessibility during the software development process.

Gender in Computing

Gender, Sex, and Intersectionality

As this project will be discussing gender, I believe it wise to clarify the distinction between sex and gender. “Gender” and “sex” are both socially constructed concepts that relate to a person’s identity. Sex is used to describe physical attributes that a society has determined are distinctive for a subset of the population, while gender is a concept of individual identification that may or may not manifest itself outwardly or align with characteristics of an individual's sex [9]. In this project, I have chosen to discuss gender rather than sex. This is due to the fact that several concepts in this project such as self-efficacy, attitude toward risk, and information processing style strongly relate to an individual's internal thoughts, feelings and emotions.

Intersectionality, or how an individual's combined attributes/identities relate to their experiences, opportunities, and outcomes [10] is also very important to consider. Race and sexual orientation as well as national origin are important factors when discussing the characteristics listed above. While I will not be focusing on the impact of racial or cultural factors in this study, I would like to point out that sexual orientation is heavily linked to gender. People whose gender identity does not align with the sex they were assigned at birth are described as 'transgender', while people whose gender identity and sex assigned at birth align are described as 'cisgender'. Here, I will be using the term 'queer' to group people who are not cisgender (transgender) and/or not heterosexual (bisexual, asexual, etc.). I will be using the term 'gender minority' to encompass individuals who find themselves outside of the gender majority. In computing, the current gender majority consists of heterosexual cisgender men [11], so the gender minority consists of people who are women and/or queer.

Gender Dominance in Computing

Some of the first 'computers' were women that were tasked with performing calculations during World War 2 while men were contributing physically to the war effort [1]. As the field of computing grew, women moved with the field and transitioned from performing calculations to working on the computers themselves. Women and queer people were responsible for many breakthroughs in computing during this time, with one of the most well-known being Alan

Turing [12]. The presence of women in computing grew into the 80's, but decreased after this point. There are several reasons for this (hiring bias, lack of recognition, shifting social roles) [13]. On top of these factors, the rise in male dominance in computing has been attributed to video game consoles being marketed toward young boys [14], the public presence of male dotcom millionaires [15], and even a 'toxic geek culture' that has been found to create unwelcoming spaces for gender minorities [16].

Previous research has found that while women have a lower feeling of belonging than men in computing, queer women often have an even lower sense of belonging than non-queer women [11]. The same pattern is seen when comparing queer men and heterosexual men, with the former having a lower sense of belonging than the latter.

Binary and Hyper-feminine Solutions

There have been many efforts to increase gender diversity in computing. For example, Girls Who Code is an organization that aims to encourage and teach girls how to code and feel more confident and capable in STEM fields. However, Girls Who Code and other organizations geared towards improving the presence of women in computing have received criticism for centering their solution on changing gender minorities themselves as opposed to working on the dominant cultures in computing. The idea that increasing a person's computer self-efficacy would make them more likely to pursue computing is not without merit. However, centering the discussion of gender diversity on the confidence and aptitude of women and queer people does a disservice to them by insinuating that the problem is a fault of their own.

Other projects [17] try to entice women and girls into joining computing by making computing seem 'pink' and 'feminine'. This too has received criticism as it plays off of stereotypical ideas of what women are interested in as opposed to encouraging them that their contributions to the field of computing as it stands can be extremely valuable and important. Overall, trying to solve this diversity problem with solutions that focus on changing a particular gender may not be optimal.

Positive Interactions with Technology

Improving gender accessibility of software products can improve the experience of all people while centering the discussion on existing computer scientists instead of a specific gender minority. Software products and their associated workflows aim to improve the process by which people complete certain tasks with regard to technology. For example, video editing software products such as DaVinci Resolve, Final Cut Pro, and Open Broadcaster Software are designed to help a user improve their video production workflow by providing efficient ways to crop, edit, color grade, and render video files as well as perform other related operations. Completing all of these operations on a single video would be dramatically more difficult without the use of software products, or even with the use of multiple single-function software products. While software products such as those listed above are generally useful for everybody, gender bias in the software development process can disproportionately reduce accessibility and performance for women and non-binary people when workflows favor a typically ‘male’ style of thinking, as is often the case when teams predominantly consist of men. Previous research suggests that software can be designed in a way that consciously considers and improves overall gender accessibility without reducing usability for users of any one gender, even when a group is relatively homogenous [18].

GenderMag

GenderMag is a research-centered framework that "aims at how society and education can impact the success and retention of women in computer science." The creators of GenderMag define it as "a process and set of materials" that can be used to better consider gender differences when designing software. GenderMag aims to consider how different thinking styles that might, on average, align with specific gender identities can impact the usability of a software product. GenderMag refrains from explicitly assigning any one thinking style to any one gender. Instead, the framework outlines 'personas' that have certain characteristics in three areas:

- Background Knowledge/Skills
- Motivations and Strategies
- Attitude To Technology

The base framework has information on creating new personas, and itself contains three that can be summarized as follows:

- Abi (prefers to stick to reliable technologies and avoid mistakes altogether)
- Pat (methodical and open to learning new technologies for practical purposes)
- Tim ("techie" who always enjoys learning about unfamiliar technology)

GenderMag suggests using these personas to assess the usability of a software product. A developer would complete small tasks while sort of "role playing" as a persona to try and make decisions rooted in the persona's thinking style. Though I will likely not be able to use this product to examine an individual's background or motivations for using technology, I aim to gather insights on their attitude toward technology, information processing style, and computer self-efficacy. The motivation to do so is not to reproduce the findings of previous research, but rather to document a new instance of accessing gender accessibility in software development.

Methods

Loopz Software Workflow

I created a simple web-app using PHP, HTML, CSS, and Javascript and hosted it on the University of Arkansas's Turing Server (Apache). The web-app is called 'Loopz' and is advertised as a multipurpose file converter that can be used to perform batch conversions in a multi-stage process. The app is explained to use 'Storage Spheres' and 'Conversion Cubes'. Storage Spheres are said to hold multiple files of a specific type, and 'Conversion Cubes' are said to convert files from one type to another. Upon loading the page a user is presented with the following view:

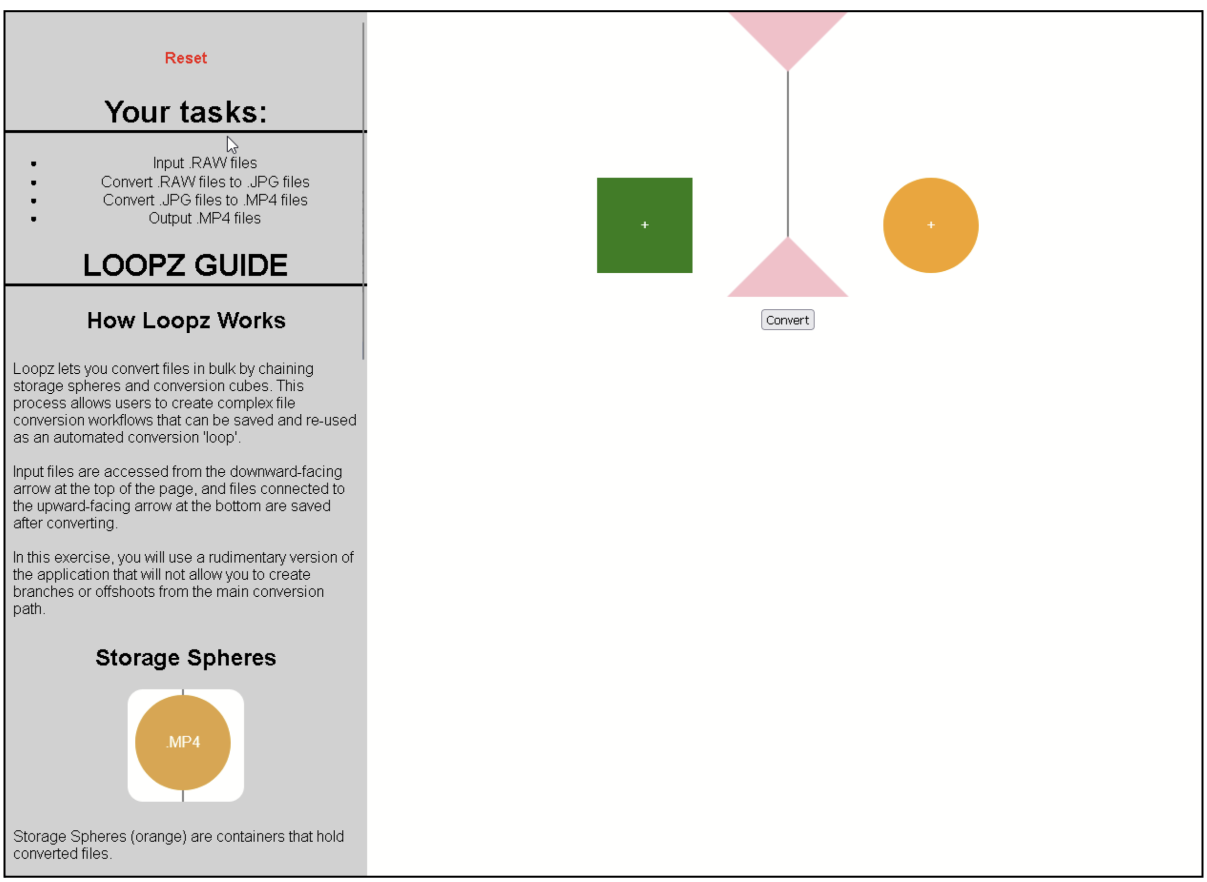

Figure 1 - Loopz Workflow Landing Page

Ideally, participants will infer that this is a semi-functional prototype for a real piece of software that is being tested for accessibility. If this software were to truly be implemented, there would be many more sections prompting the user for input. For example, converting a batch of images to a video should ask for parameters such as image display time, transition type, and perhaps even an input for audio. The workflow is intentionally clunky and rudimentary to encourage users to 'think' to complete their task. For instance, conversion cubes require a mouse click before their display text will reflect the file conversion they would be performing. However, if the user adds a conversion cube without connecting it to storage spheres, the cube would read 'Undefined to Undefined', which does not provide much information as to what is going wrong. The cube can also be clicked more than once without resetting the page, but the 'Click to Read' text disappears after the first click. The workflow includes a guide that users may reference if they are unsure of how exactly to use the application.

In order to measure how well a user utilizes the workflow, I prompted the users with the following scenario:

You are in charge of uploading marketing photos for your company website. But, there's a problem: All of the photos you took are in RAW format (the file names have the .RAW extension at the end) but they need to be in PNG format (the file names have the .PNG extension at the end) in order to be turned into a video that your company can use as the background of their website. Your job is to convert .RAW photos to .PNG files and then turn these into .MP4 files. You will not need to upload or download any files, as these parts of the application are being simulated for the purposes of this assessment.

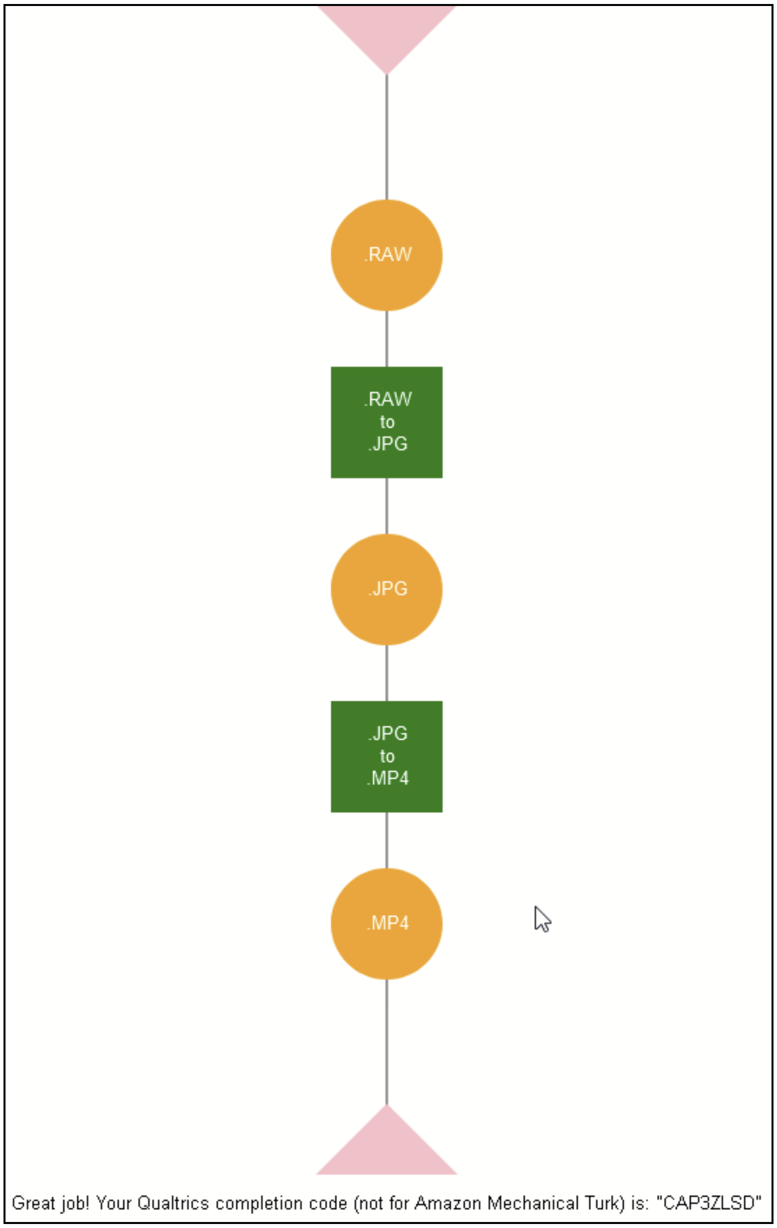

Figure 2 - Expected Solution for Loopz Scenario

The key tasks of the scenario are outlined above the guide section on the application. A user who likes to tinker will likely try multiple configurations to intuit the inner workings of the software, while a user who is more process-oriented will likely read the help text first. The app allows the former to try many different configurations and reset them, and it also provides the latter with a more concrete explanation to follow.

Data Collection

I deployed a survey link on Amazon Mechanical Turk, where “workers” can complete tasks to earn money (or just for fun). Workers can be made to fit certain criteria such as age, income, and gender, but I felt adding these restrictions was unnecessary in this example. The Mechanical Turk Survey links to a Qualtrics survey that contains various questions related to demographic information, confidence/self-efficacy, and a user’s experience with the workflow. While the respondents were told that this study was related to gender in computing, I still wanted to add this user experience question to simulate a “real” usability survey.

Results

Though 150 responses were recorded, 25 were discarded for a variety of reasons (such as incomplete responses from participants who opted out after reading the informed consent statement. The demographic questions were text input fields and allowed respondents to enter any answer. The racial and gender demographic information for the remaining respondents is show here:

	Male	**Female**	
White	70	43	113
Asian	7	0	7
Black	5	0	5
	82	43	125

Figure 3 - Respondent Demographic Data (Race and Gender)

In order to effectively compare the responses of the male and female subgroups, I converted each response count to a percentage of the total respondents in each gender subgroup. Gender-normalized response comparisons for all questions are shown below.

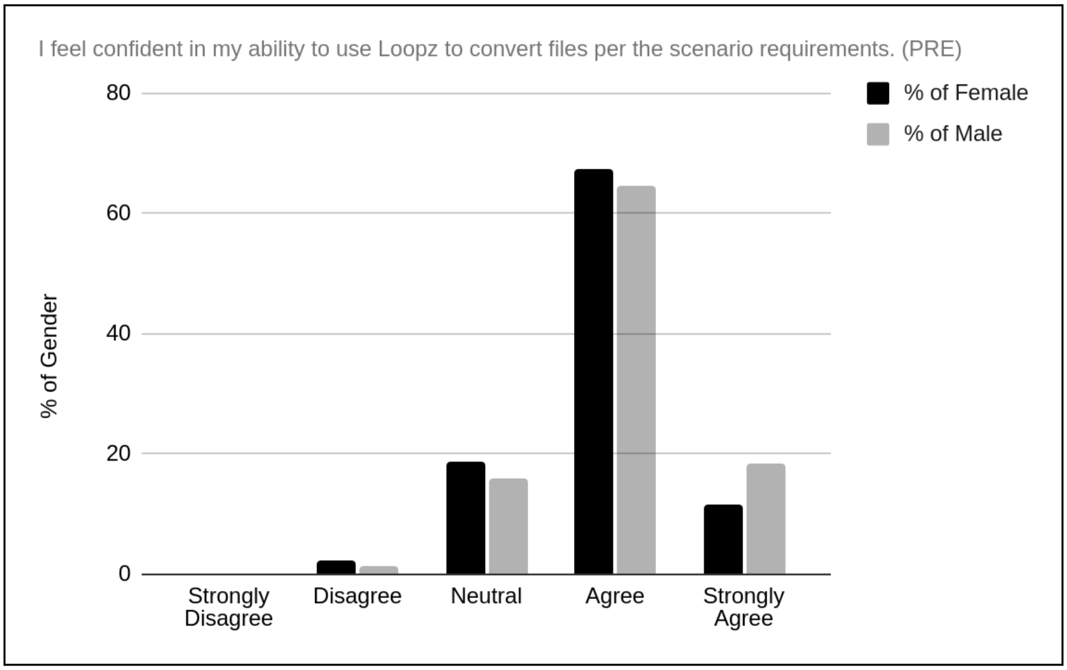

Figure 4A - Pre-workflow Question 1 Responses (Proportion by Gender)

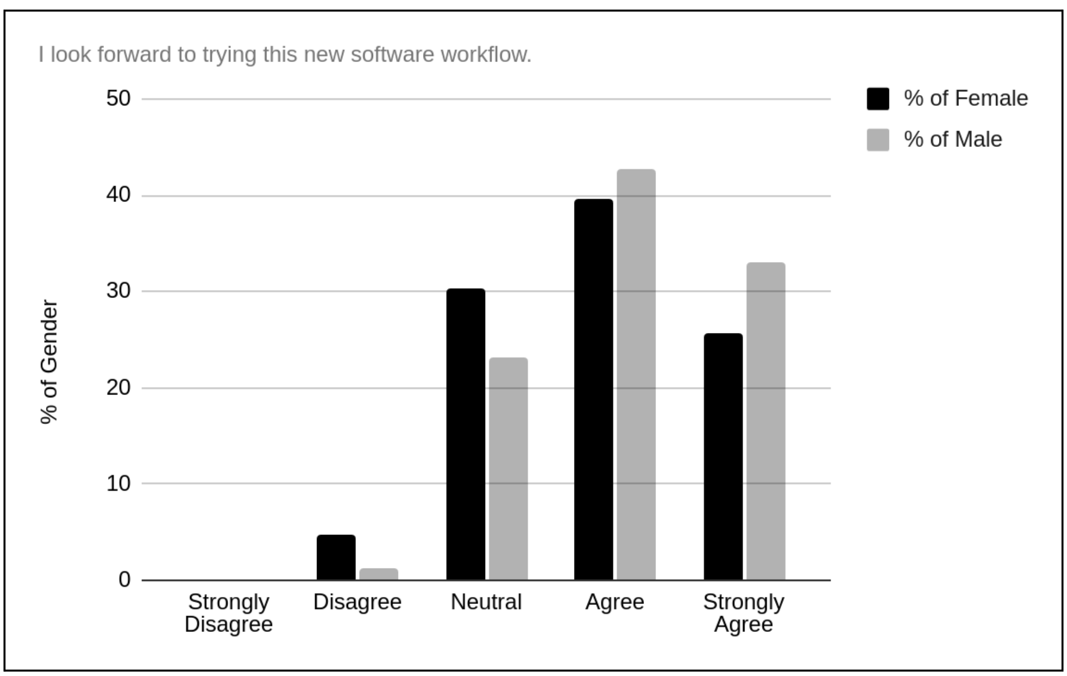

Figure 4B - Pre-workflow Question 2 Responses (Proportion by Gender)

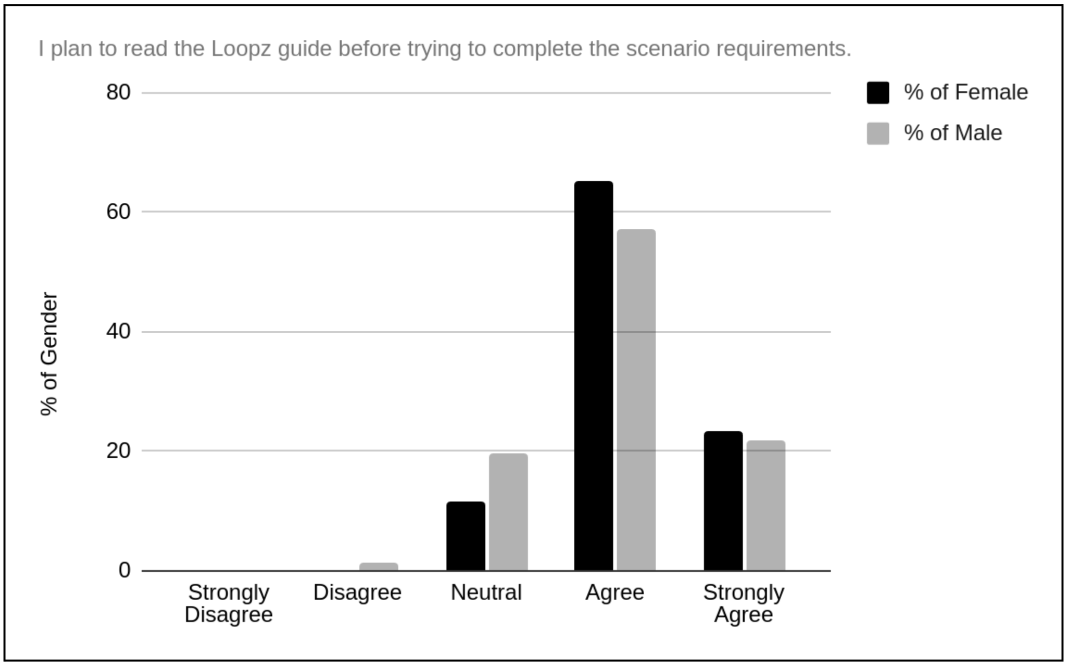

Figure 4C - Pre-workflow Question 3 Responses (Proportion by Gender)

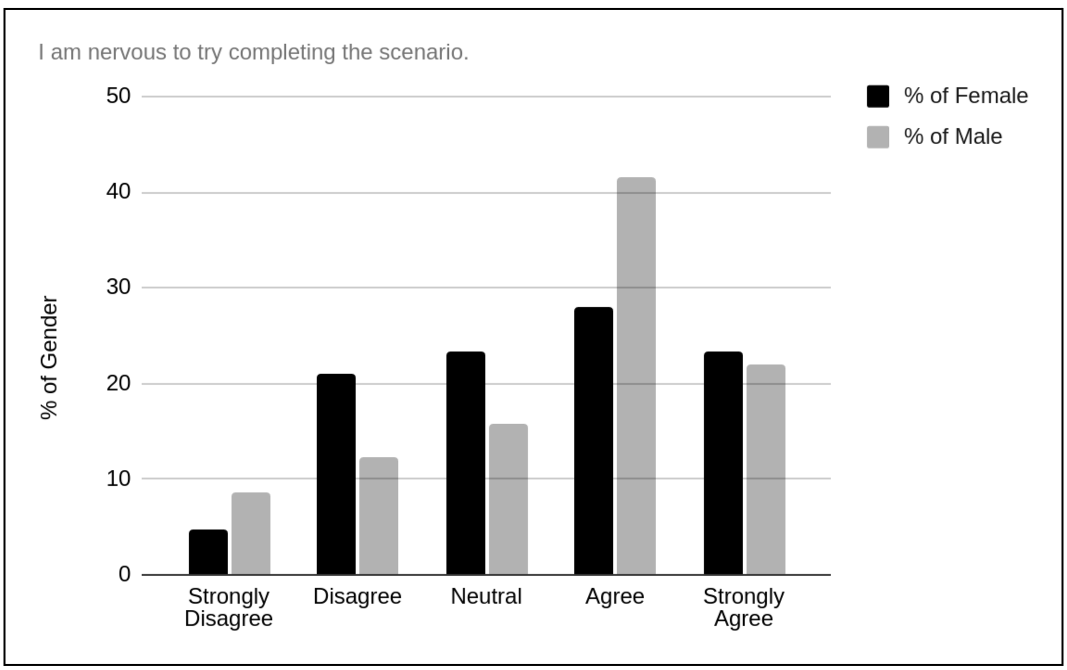

Figure 4D - Pre-workflow Question 4 Responses (Proportion by Gender)

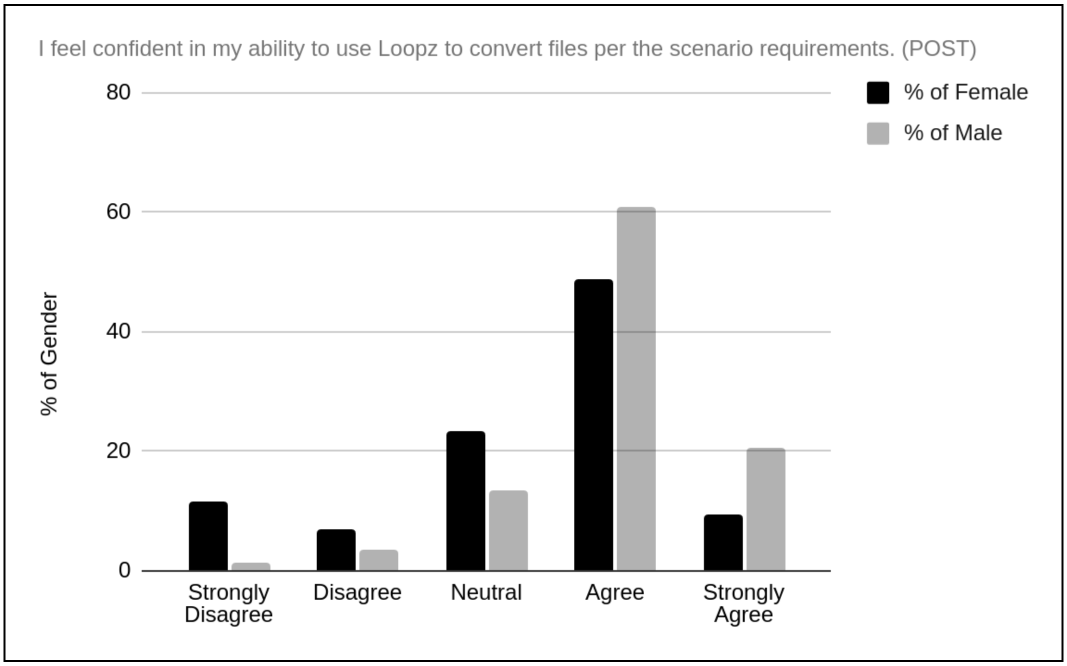

Figure 5A - Post-workflow Question 1 Responses (Proportion by Gender)

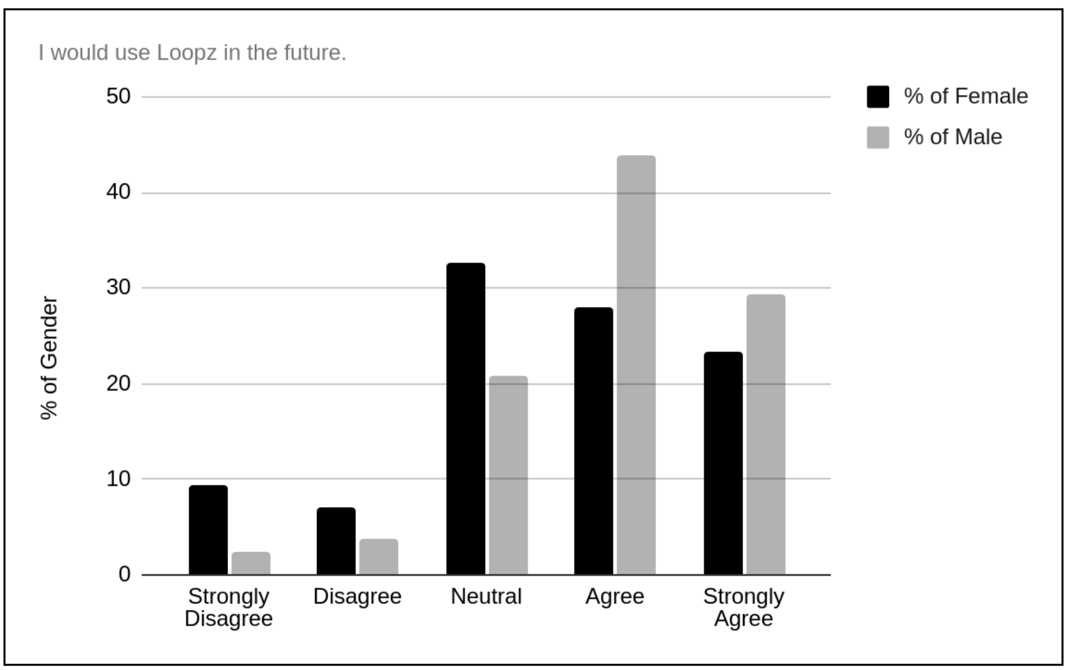

Figure 5B - Post-workflow Question 2 Responses (Proportion by Gender)

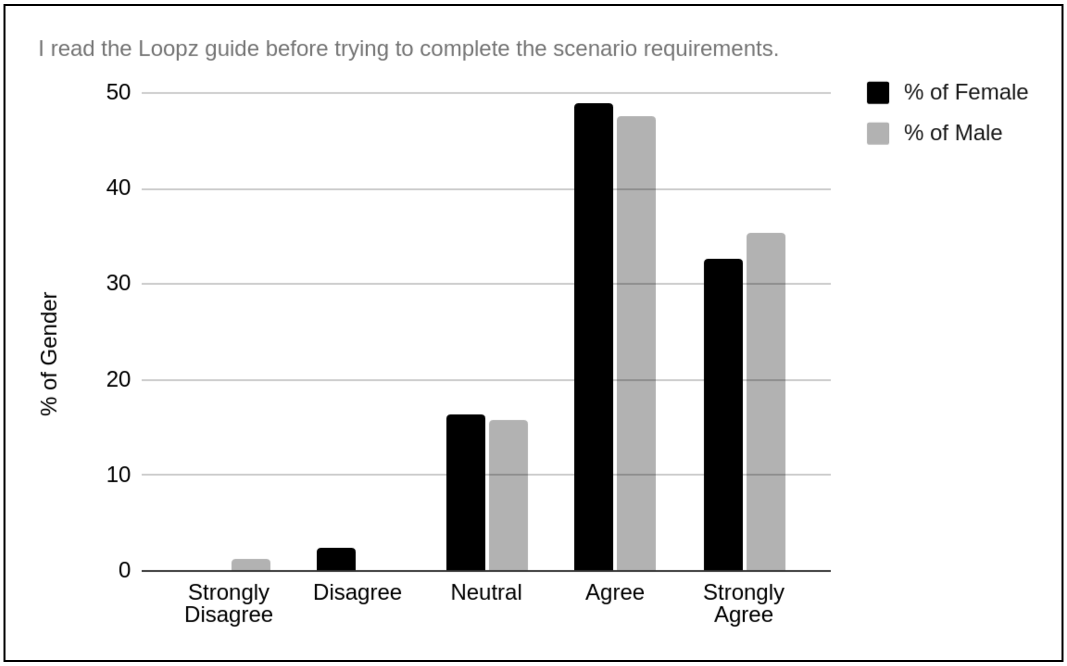

Figure 5C - Post-workflow Question 3 Responses (Proportion by Gender)

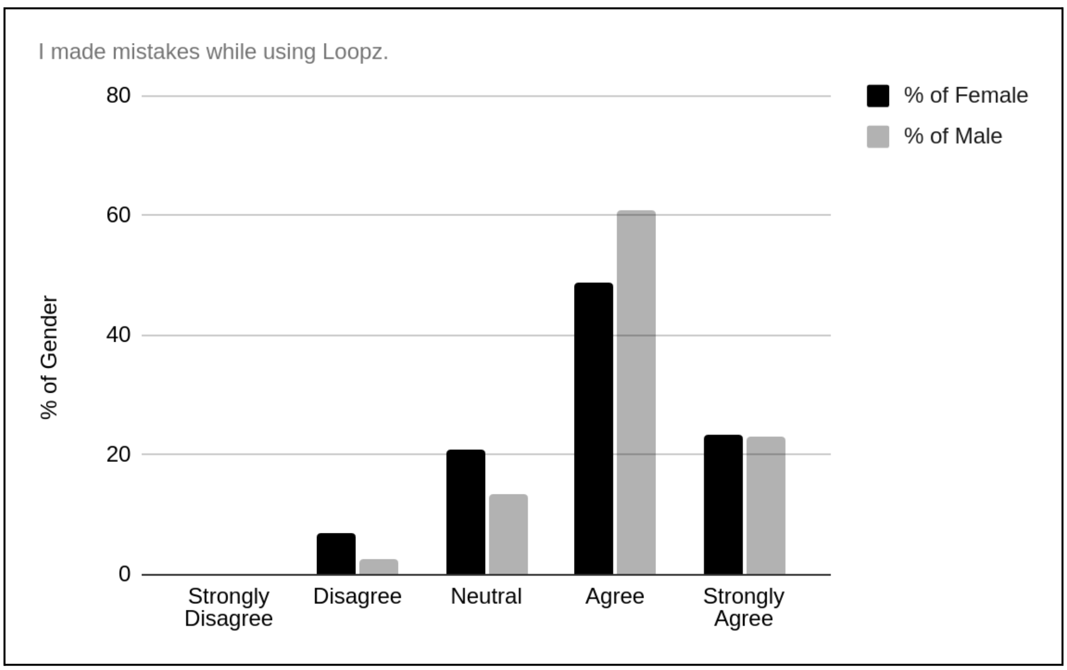

Figure 5D - Post-workflow Question 4 Responses (Proportion by Gender)

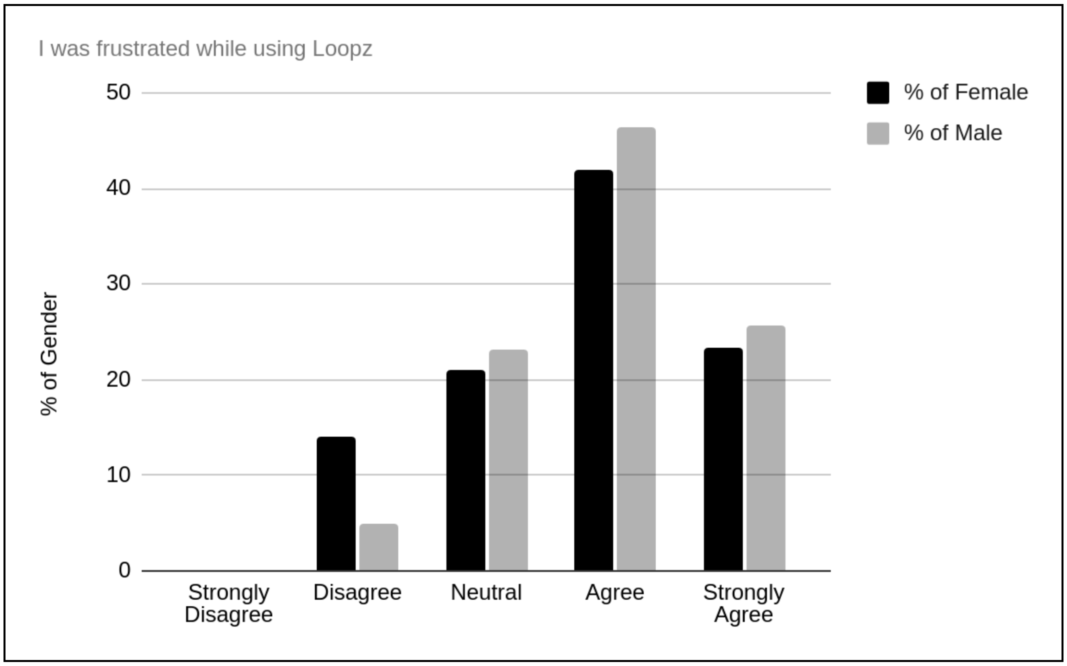

Figure 5E - Post-workflow Question 5 Responses (Proportion by Gender)

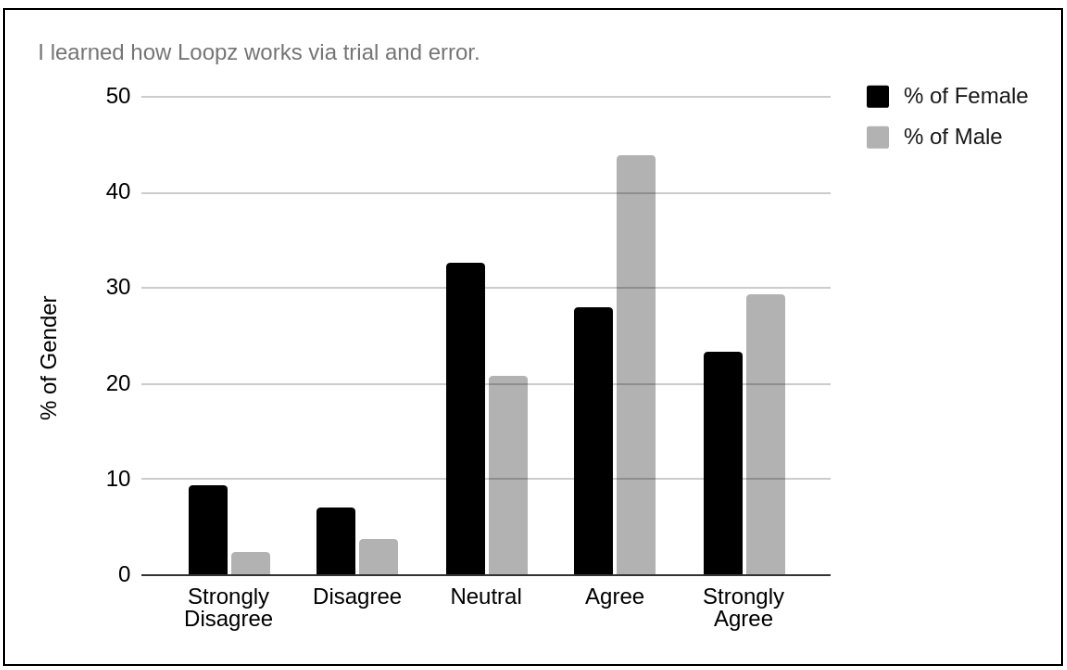

Figure 5F - Post-workflow Question 6 Responses (Proportion by Gender)

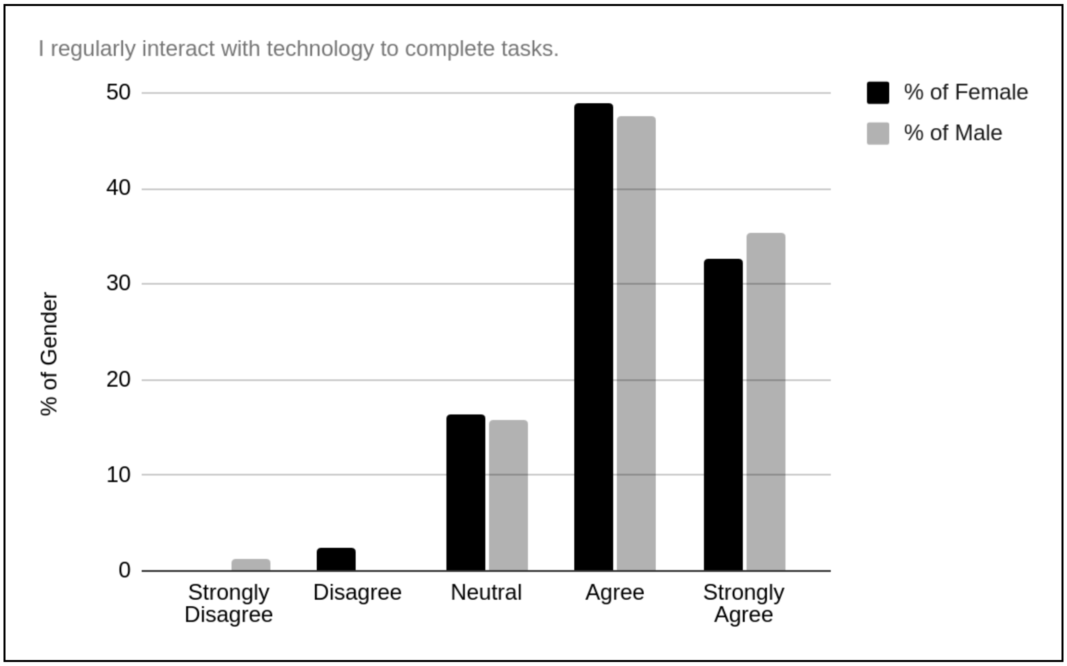

Figure 5G - Post-workflow Question 7 Responses (Proportion by Gender)

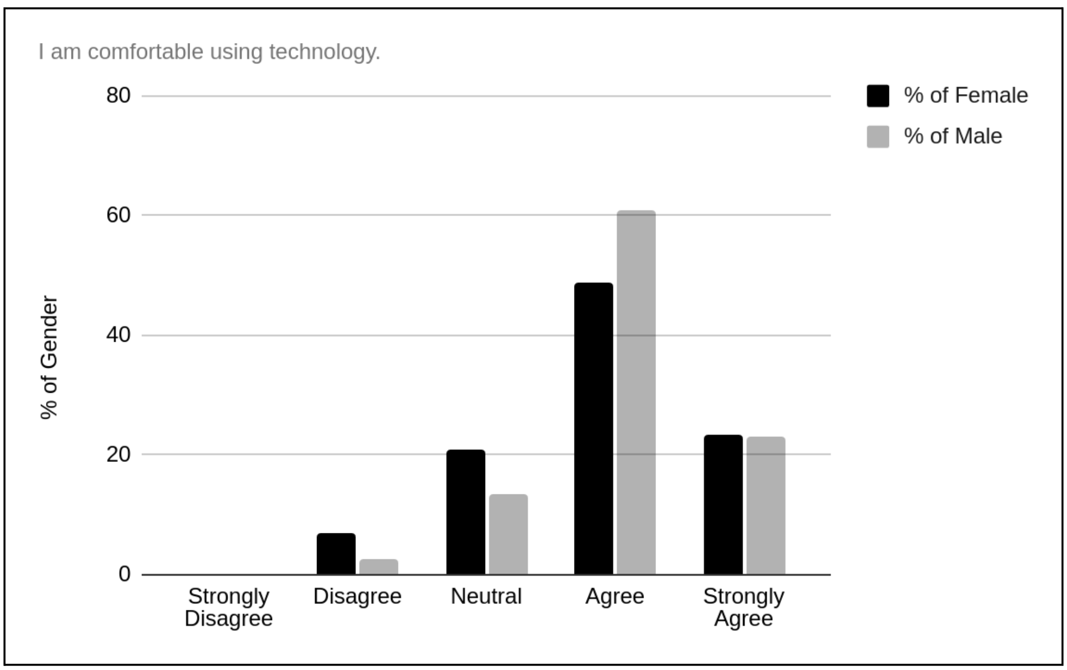

Figure 5H - Post-workflow Question 8 Responses (Proportion by Gender)

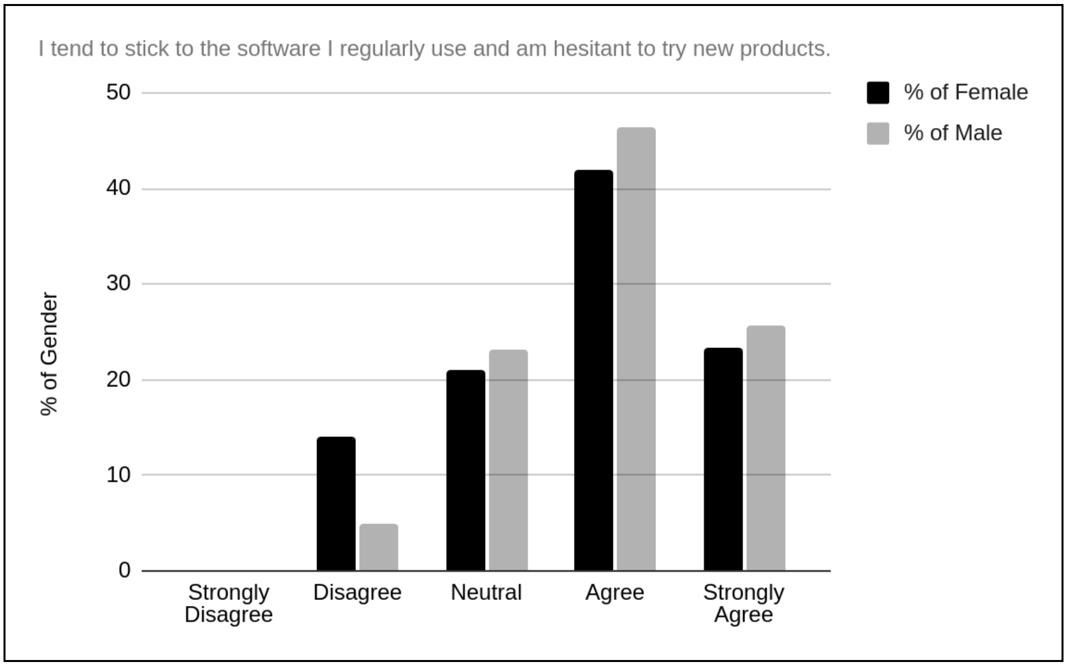

Figure 5I - Post-workflow Question 9 Responses (Proportion by Gender)

Loopz scenario completion percentages are shown here:

	Completed	Not Completed
Male	23.17%	76.83%
Female	32.56%	67.44%
Total	26.40%	73.60%

Figure 6 - Loopz Scenario Completion Percentages (As Proportion of Gender Subgroup)

Users provided their agreement to the statement "I feel confident in my ability to use Loopz to convert files per the scenario requirements" before and after using the workflow. The percent change for each option by gender is shown below:

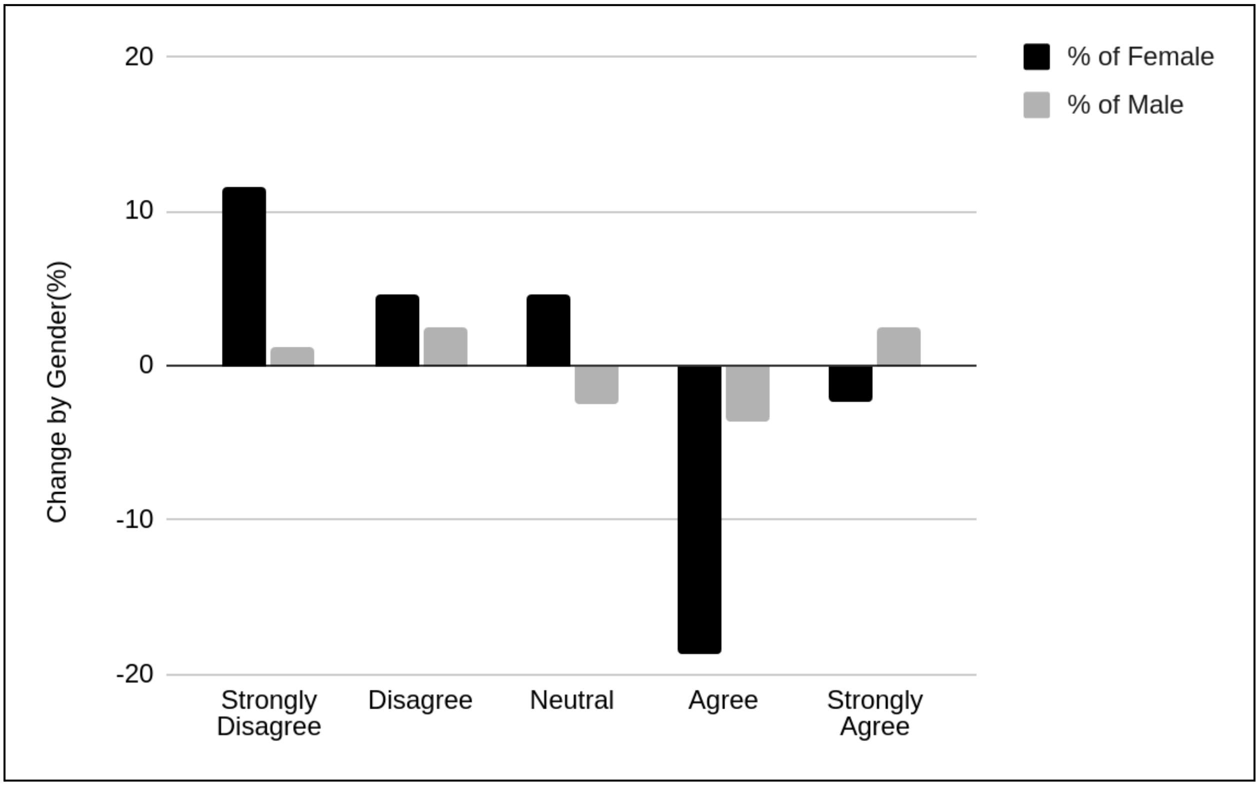

Figure 7 - Absolute Change in Confidence Question Response Proportions by Gender Subgroup

Discussion and Future Work

There are a lot of interesting trends in the above graphs that not only provide insight on potential improvement for the Loopz software workflow, but also confirm the findings of previous research referenced in this project. One of the most interesting graphs is **Figure 7**, which illustrates the change in confidence after using the Loopz workflow. There was a drastic reduction in how many female respondents 'agreed' that they were confident in their ability to use Loopz to complete the provided scenario. While increased confidence before using the workflow could be attributed to a lack of information on how Loopz was built, it is interesting that the average drop in confidence was much more substantial for women than men. This is even more interesting when looking at **Figure 6**, which shows that the female subgroup was actually more likely to complete the scenario. Even though the female subgroup used Loopz more effectively, they were less comfortable and confident working with it. If one chooses to think that the difference in self-efficacy after using the workflow is related to Loopz, one would likely conclude that Loopz has some accessibility issues related to gender. If one chooses to attribute this difference to some factors outside of Loopz, it would seem that our sample set aligns with previous research on the relationship between software and gender minorities.

While the confidence question was the only question asked before and after using the workflow, there were still some interesting trends in the data. As shown in **Figure 5H**, the male subgroup of the respondents seemed to report a slightly higher comfort surrounding technology. **Figure 5F** shows that this group reported experimenting with Loopz through trial and error more frequently than the female subgroup. On top of this, **Figure 4B** shows that the male respondents were, on average, more likely to report that they were looking forward to trying the new software workflow. All of these trends are consistent with the findings of previous research.

I believe that these findings are relevant due to a key aspect of this example: the Loopz workflow was created by a person who identifies as male. Even though the male subgroup reported more mistakes and frustration while using Loopz (**Figures 5D, 5E**), they seemed more likely to use Loopz again than the female subgroup (**Figure 5B**). While this could be related to the idea that straightforward tinkering is a more 'masculine' processing style than calculated exploration, the fact that the gender identity of the software creator aligns with the identities of the male subgroup of respondents is likely a contributing factor to these trends. Though I hoped that adding a rudimentary guide to Loopz would support the learning styles of more process-oriented users, it seems that this was not enough to make Loopz accessible to users of all genders. But without considering the possibility that Loopz would not be as accessible to female users as it would male users, this survey would not have been deployed and this insight may not have been gained.

The conversation surrounding gender and computing is not new. The ways we discuss gender, the way gender manifests itself through technology, and technology itself are in a constant state of change. I believe I have shown that assessing the way technology is received by all subgroups of the population is a key part of ensuring that technology continues to become more user-friendly, and also that doing so can be a useful tool in improving the presence of gender minorities in computing. Discussing gender accessibility in software in educational and professional settings could improve the experiences of today's users, improve the output of today's coders, and improve the outlook of tomorrow's computer scientists and engineers. While the problem of gender accessibility in computing will not be solved overnight, evaluating the usability of existing software products will turn the field of computing towards a hopeful direction.

Appendix

	Strongly Disagree		Disagree		Neutral		Agree		Strongly Agree	
	M	**F**	**M**	**F**	**M**	**F**	**M**	**F**	**M**	**F**
I feel confident in my ability to use Loopz to convert files per the scenario requirements.	0	0	1	1	13	8	53	29	15	5
I look forward to trying this new software workflow.	0	0	1	2	19	13	35	17	27	11
I plan to read the Loopz guide before trying to complete the scenario requirements.	0	0	1	0	16	5	47	28	18	10
I am nervous to try completing the scenario.	7	2	10	9	13	10	34	12	18	10

Figure i - Raw Pre-Workflow Response Count by Gender Subgroup

	Strongly Disagree		Disagree		Neutral		Agree		Strongly Agree	
	M	**F**	**M**	**F**	**M**	**F**	**M**	**F**	**M**	**F**
I feel confident in my ability to use Loopz to convert files per the scenario requirements.	1	5	3	3	11	10	50	21	17	4
I would use Loopz in the future	2	4	3	3	17	14	36	12	24	10
read the Loopz guide before trying to complete the scenario requirements.	1	0	0	1	13	7	39	21	29	14
I made mistakes when using Loopz.	0	0	2	3	11	9	50	21	19	10
I was frustrated while using Loopz	0	0	4	6	19	9	38	18	21	10
I learned how Loopz works via trial and error.	2	4	3	3	17	14	36	12	24	10
I regularly interact with technology to complete tasks.	1	0	0	1	13	7	39	21	29	14
I am comfortable using technology.	0	0	2	3	11	9	50	21	19	10
I tend to stick to the software I regularly use and am hesitant to try new products.	0	0	4	6	19	9	38	18	21	10

Figure ii - Raw Post-Workflow Response Count by Gender Subgroup

	Completed	Did Not Complete	
Male	19	63	82
Female	14	29	43
	33	92	125

Figure iii - Raw Completion Count by Gender Subgroup

References

[1] Denise Gürer. 2002. Women in computing history. SIGCSE Bull. 34, 2 (June 2002), 116–120. https://doi.org/10.1145/543812.543843

[2] Sapna Cheryan, Victoria C. Plaut, Paul G. Davies, and Claude M. Steele. 2009. Ambient belonging: How stereotypical cues impact gender participation in computer science. *Journal of Personality and Social Psychology* 97, 6 (2009), 1045–1060.

[3] Amanda Menier, Rebecca Zarch, and Stacey Sexton. 2021. Broadening Gender in Computing for Transgender and Nonbinary Learners. In *2021 Conference on Research in Equitable and Sustained Participation in Engineering, Computing, and Technology (RESPECT)*, IEEE. Retrieved September 21, 2022 from http://dx.doi.org/10.1109/respect51740.2021.9620612

[4]Sylvia Beyer, Kristina Rynes, Julie Perrault, Kelly Hay, and Susan Haller. 2003. Gender differences in computer science students. In *Proceedings of the 34th SIGCSE technical symposium on Computer science education - SIGCSE '03*, ACM Press, New York, New York, USA. Retrieved September 21, 2022 from http://dx.doi.org/10.1145/611892.611930

[5] Z. Smits. 2021. Analyzing the discourse of Girls Who Code concerning gender inequality. Retrieved from https://studenttheses.uu.nl/handle/20.500.12932/1443

[6] Simone Stumpf et al. 2020. Gender-inclusive HCI research and design: A conceptual review. *Foundations and Trends® in Human–Computer Interaction* 13, 1 (2020), 1–69. DOI:http://dx.doi.org/10.1561/1100000056

[7] Margaret Burnett et al. 2016. GenderMag: A method for evaluating software's gender inclusiveness. *Interacting with Computers* 28, 6 (2016), 760–787. DOI:http://dx.doi.org/10.1093/iwc/iwv046

[8] Timothy J Weston, Wendy M Dubow, and Alexis Kaminsky. 2019. Predicting women's persistence in computer science-and technology-related majors from high school to college. *ACM Transactions on Computing Education (TOCE)* 20, 1 (2019), 1–16.

[9] Os Keyes, Chandler May, and Annabelle Carrell. 2021. You keep using that word: Ways of thinking about gender in computing research. *Proceedings of the ACM on Human-Computer Interaction* 5, CSCW1 (2021), 1–23. DOI:https://doi.org/10.1145/3449113

[10] Jane G Stout and Heather M Wright. 2016. Lesbian, gay, bisexual, transgender, and queer students' sense of belonging in computing: An Intersectional approach. *Computing in Science & Engineering* 18, 3 (2016), 24–30.

[11] Monica Babeş-Vroman, Thuytien N Nguyen, and Thu D Nguyen. 2021. Gender Diversity in Computer Science at a Large Public R1 Research University: Reporting on a Self-study. *ACM Transactions on Computing Education (TOCE)* 22, 2 (2021), 1–31.

[12] Patricia Fancher. 2017. Embodying Turing's Machine: Queer, Embodied Rhetorics in the History of Digital Computation. *Rhetoric Review* 37, 1 (December 2017), 90–104. DOI:https://doi.org/10.1080/07350198.2018.1395268

[13] Elizabeth Patitsas, Michelle Craig, and Steve Easterbrook. 2014. A historical examination of the social factors affecting female participation in computing. In *Proceedings of the 2014 conference on Innovation & technology in computer science education - ITiCSE '14*, ACM Press, New York, New York, USA. Retrieved April 14, 2023 from http://dx.doi.org/10.1145/2591708.2591731

[14] Roger Smith and Adrienne Decker. 2016. Understanding the Impact of QPOC Representation in Video Games. *2016 Research on Equity and Sustained Participation in Engineering, Computing, and Technology (RESPECT)* (2016), 1–8. DOI:https://doi.org/10.1109/RESPECT.2016.7836164

[15] J. Abbate. 2003. Guest Editor's Introduction: Women and Gender in the History of Computing. *IEEE Annals of the History of Computing* 25, 04 (2003), 4–8. DOI:https://doi.org/10.1109/MAHC.2003.1253885

[16] Anastasia Salter and Bridget Blodgett. 2017. *Toxic geek masculinity in media: Sexism, trolling, and identity policing*. Springer.

[17] Anne Weibert, Andrea Marshall, Konstantin Aal, Kai Schubert, and Jennifer Rode. 2014. Sewing interest in E-textiles: analyzing making from a gendered perspective. In *Proceedings of the 2014 conference on Designing interactive systems*, 15–24.

[18] Margaret Burnett, Robin Counts, Ronette Lawrence, and Hannah Hanson. 2017. Gender HCl and microsoft: Highlights from a longitudinal study. In *2017 IEEE Symposium on Visual*

Languages and Human-Centric Computing (VL/HCC), IEEE. Retrieved April 14, 2023 from http://dx.doi.org/10.1109/vlhcc.2017.8103461