

“Game Changer” or “Overenthusiastic Drunk Acquaintance”? Generative AI Use by Blind and Low Vision Software Professionals in the Workplace

Yoonha Cha
yoonha.cha@uci.edu
University of California, Irvine
Irvine, CA, USA

Victoria Jackson
v.jackson@soton.ac.uk
University of Southampton
Southampton, UK

Lauren Shu
lpshu@uci.edu
University of California, Irvine
Irvine, CA, USA

Stacy M. Branham
sbranham@uci.edu
University of California, Irvine
Irvine, CA, USA

André van der Hoek
andre@ics.uci.edu
University of California, Irvine
Irvine, CA, USA

Abstract

The software development workplace poses numerous technical and collaborative accessibility challenges for blind and low vision software professionals (BLVSPs). Though Generative AI (GenAI) is increasingly adopted within the software development industry and has been a rapidly growing topic of interest in research, to date, the unique perspectives of BLVSPs have yet to be consulted. We report on a qualitative study involving 39 semi-structured interviews with BLVSPs about what the introduction of GenAI has meant for their work. We found that BLVSPs used GenAI for many software development tasks, resulting in benefits such as increased productivity and accessibility. However, significant costs were also accompanied by GenAI use as they were more vulnerable to hallucinations than their sighted colleagues. Sometimes, organizational policies prevented use. Based on our findings, we discuss the higher-risks and higher-returns that BLVSPs had to carefully weigh when deciding whether and when to use GenAI tools for work.

CCS Concepts

• **Human-centered computing** → **Empirical studies in HCI**;
Empirical studies in accessibility; • **Software and its engineering** → **Programming teams**.

Keywords

software development, accessibility, generative AI, workplace accessibility, blind and low vision

ACM Reference Format:

Yoonha Cha, Victoria Jackson, Lauren Shu, Stacy M. Branham, and André van der Hoek. 2026. “Game Changer” or “Overenthusiastic Drunk Acquaintance”? Generative AI Use by Blind and Low Vision Software Professionals in the Workplace. In *2026 IEEE/ACM 48th International Conference on Software Engineering (ICSE '26)*, April 12–18, 2026, Rio de Janeiro, Brazil. ACM, New York, NY, USA, 13 pages. <https://doi.org/10.1145/3744916.3773172>



This work is licensed under a Creative Commons Attribution 4.0 International License. *ICSE '26, Rio de Janeiro, Brazil*

© 2026 Copyright held by the owner/author(s).
ACM ISBN 979-8-4007-2025-3/2026/04
<https://doi.org/10.1145/3744916.3773172>

1 Introduction

According to the most recent Stack Overflow survey that includes disability demographics (2022) [73], 1.7% percent of the total respondents reported being blind or having low vision (BLV), comprising 44.8% of the respondents who identified as having a disability. Existing scholarship regarding blind and low vision software professionals (BLVSPs) documents the significant labor they invest into gaining access at work, as many tools and processes used for daily job tasks (e.g., IDEs, project management tools) remain inaccessible [3, 4, 25, 30, 59]. This additional, “invisible work” [11] materializes in many different forms, including procuring assistance from sighted coworkers [3], identifying alternative accessible solutions [15, 37, 60], and even building do-it-yourself (DIY) tools outside of regular work hours [18]. As a result, BLVSPs experience inequitable barriers to lateral and hierarchical career progression [17].

In the software development profession, Generative Artificial Intelligence (GenAI) has gained significant attention and is rapidly becoming integrated into professionals’ everyday workflows [6]. 76% of all respondents to Stack Overflow’s 2024 survey indicated an intention to use GenAI tools for software development, and 62% reported already utilizing them; this marks a 6% and 18% increase, respectively, from the previous year [58]. Much research has focused on the use of GenAI in programming [36], maintenance [36], and testing [36], how to efficiently integrate GenAI tools into software development [67], and its impact on the job market [46, 48]. However, while GenAI is being taken up by software professionals to support productivity and efficiency, little is known about the experiences of underrepresented groups in software development with GenAI tools and how they may benefit or hinder them. In particular, the experiences of BLVSPs with GenAI and the potential of this technology to reduce the burden of invisible work and overcome career barriers have yet to be explored. To address this gap, we investigated the following research questions:

- RQ1. How do BLVSPs perceive and utilize generative AI in the workplace?
- RQ2. How do generative AI tools impact BLVSPs’ work and career prospects?

We report the results from in-depth, semi-structured interviews with 39 BLVSPs who identified as being blind or having low vision, are working or have worked in a software development position either in a corporate setting or as a freelancer, and have at least one year of work experience. Through thematic analysis [12], we identified three main themes: (1) benefits of using GenAI, (2) barriers and risks associated with using GenAI for work, and (3) the wider impact that GenAI has on BLVSPs in the workplace. We discuss BLVSPs' rationales for deciding whether and when to use GenAI tools at work, and suggest implications for GenAI tool developers, researchers, and employers.

Through this study, we contribute the following:

- A comprehensive understanding of BLVSPs' decisions about the (non) use of Generative AI tools in the workplace.
- Impact of GenAI tools on overall workplace accessibility and career paths for BLVSPs.
- A set of implications for GenAI tool developers, researchers, and employers.

2 Background and Related Work

2.1 Assistive Technology

As per the World Health Organization, "Assistive Technology (AT) enables and promotes inclusion and participation, especially of persons with disability, aging populations, and people with non-communicable diseases" [57]. Popular AT products used by BLV individuals include screen readers (e.g., NVDA [1], JAWS [71]) that speak the content of a webpage or application out loud and are navigable with key combinations, screen magnifiers that zoom in and out to help read the contents of a screen, and braille displays that enable access to information by raising and lowering pins in refreshable, tactile braille character cells.

2.2 BLVSPs in the Workplace

BLVSPs face challenges in equitable participation in the workplace due to inaccessible developer tools [60], work environments [15, 17], and ableist colleagues [17, 24]. Overcoming these barriers requires additional access labor of negotiating practices, and developing workarounds and bespoke tools to be able to complete job tasks [15, 17, 18]. Legislation and policies in many countries require companies to provide reasonable accommodations (e.g., the Americans with Disabilities Act (ADA) in the USA [55]). Such accommodations include the provision of adequate ATs and flexible work practices. Yet, accommodations are not always made in a timely manner, leading to workarounds and frustrations [49]. The workplace continues to remain challenging for many BLVSPs [17], leading to career advancement concerns [17, 25].

While software development is a collaborative endeavor [53], it can be challenging for BLVSPs to fully participate in collaborative work (e.g., [37, 59]). Remote meetings introduce difficulties such as the need to juggle the use of a screen reader to interpret a presentation while simultaneously listening to the presenter [77]. Ideating with colleagues is tricky, as popular digital whiteboarding tools (e.g., Miro) do not work well with screen readers or braille displays [25]. Collaborative authoring imposes additional work as collaborative document editors, while accessible, are often not usable for BLVSPs [24]. BLVSPs thus rely on workarounds such

as assistance from sighted colleagues (e.g., [24, 25, 77]) or custom tools [18] to collaborate effectively. Deciding whether to ask colleagues for assistance is often stressful for BLVSPs as asking for help discloses their disability with worries about how this will be perceived and potential negative consequences [15].

A key area of research focus has been the accessibility of technical tasks such as authoring code (e.g., [3, 8, 59, 61, 66]), pair-programming (e.g., [59]), and debugging (e.g. [66, 74]). Among the challenges caused by accessibility issues are difficulties in navigating code [3], reading code in an IDE [61], and limited pair programming contributions [59]. To improve the accessibility of technical tasks, various IDE plugins have been proposed by researchers (e.g., [8, 65]). For example, CodeTalk [66] summarizes information within an IDE screen as an audio feed. More recently, tools that incorporate AI to improve accessibility, such as a command-line debugging tool [68], have been explored.

To support the completion of technical tasks, developers (including BLVSPs) seek information from forums, blogs, tutorials, and official documentation websites. Many of these sources pose accessibility issues that require workarounds, such as custom scripts or asking sighted colleagues for assistance [18, 75].

2.3 GenAI Use in Software Development

Developers have rapidly integrated GenAI [58] into numerous development tasks, including code generation [62], authoring test cases [72], and debugging [44]. One of the primary benefits of adopting GenAI is increased developer productivity (e.g., [10, 62, 82]). This increase is due to reasons such as GenAI enabling users to quickly search for pertinent information [26], automating boilerplate code [42], and auto-completing code [51]. Use of GenAI has also been noted to increase job satisfaction [72]. However, GenAI can generate poor quality code [47], insecure code [63], and inaccessible code [5], for which rework is needed to fix the mistakes [45]. Trust in GenAI tools is a significant concern among developers (e.g., [44, 79]) as well as skepticism about how well GenAI can assist with more complex software development tasks [43].

GenAI has shown potential to improve software accessibility for BLV people by assisting developers in authoring accessible applications (e.g., [5, 14, 54, 76, 81]). One study found that GenAI-generated code has fewer accessibility errors in some accessibility categories than human-generated code, but in other categories it introduces more accessibility errors than humans [76]. Another noted that GenAI was able to fix some, but not all, accessibility errors in both GenAI and human-generated code [5]. To better support developers, researchers have explored methods to facilitate the creation of accessible code. For example, MS VS Code plugins use GenAI to automatically detect accessibility issues in the code [54], alert developers of accessibility issues [54], and suggest fixes to identified issues [14]. While the plugins result in fewer accessibility errors, more complex issues still require developer expertise to address [54]. While GenAI shows promise, developer education and training remain vital for developing accessible applications [54].

As GenAI use becomes commonplace and an increasing amount of coding is automated [6], it is unclear what impact GenAI will have on the developer profession [29] and the nature of development work. Some project that the demand for AI-based products will

increase job opportunities for developers [32, 56], so it is critical for developers to upskill by learning how to use GenAI effectively [40]. Early indications are that development work is becoming more rote with the potential to reduce job satisfaction [70]. Although others believe developers will need to focus on the more complex tasks AI struggles with [29, 41], or the more interesting, valuable creative tasks [39]. Finally, the impact of GenAI adoption on the software profession may not be equally distributed, since developers in minority groups may be more negatively impacted [35].

2.4 GenAI Use by BLV People

GenAI tools are used by BLV people to describe visual information contained within images or photos [2, 78]. The rich descriptions provided are often superior to those offered by sighted individuals. However, hallucinations cause BLV people to contemplate how much they can trust the response [78] and whether verification through other sources (e.g., using Google, prompting multiple models, asking another person) is required [78]. This decision to verify depends on the context of use, how high the stakes are, whether it can be verified, and relying on perceived “believability” of the response [2], which requires additional work from BLV people. Furthermore, to even use GenAI tools, individuals often need to overcome accessibility issues present in GenAI tools. For example, buttons are unlabeled, or headings and landmarks are not present in ChatGPT, making it difficult to use the tool with a screen reader [2]. This inaccessibility in GenAI applications leads to concerns that people with disabilities (PWD)—including BLV people—will be left behind and unable to reap the many benefits promised by GenAI [7].

Beyond the multi-purpose GenAI tools used for both general work and development work, such as ChatGPT and Claude, developers also utilize specific developer tools like GitHub Copilot, which leverages AI to suggest code as they author it. A recent study [31] on the accessibility of GitHub Copilot found that BLV programmers experienced similar benefits as sighted developers, including improved productivity. However, Copilot also posed challenges and frustrations to BLVSPs, such as constant context switching between tabs, breaking the coding flow, and requiring multiple keystrokes to navigate through the application.

This study contributes to existing scholarship on the experiences of BLVSPs at work by investigating their experiences with using various GenAI tools beyond GitHub Copilot to support better workplace accessibility.

3 Study Design

This study builds on an existing study examining how BLVSPs overcome accessibility issues at work by building and using do-it-yourself (DIY) tools [18], in which we asked several initial questions about their general engagement with GenAI. This provided us with some early insights, though these did not cover all aspects of interest for the researchers. Thus, the researchers decided to conduct a further set of interviews to probe more specifically into BLVSPs’ GenAI use at work.

3.1 Semi-Structured Interviews

We prepared a new interview protocol, where we kept some GenAI-related questions from the previous protocol, and added new questions of interest. These questions were informed by prior studies on the adoption of GenAI by developers (e.g., [26, 62]) and the researchers’ prior accessibility-related studies. The original protocol (questions related to GenAI are highlighted) and the new interview protocol are available in supplementary materials [16].

3.2 Participants

We sought to recruit software professionals who: (1) self-identified as being blind or having low vision, (2) working or have worked as a software professional (e.g., software engineer, accessibility specialist, product manager, designer), and (3) have at least one year of professional work experience. We employed several non-probabilistic approaches [9] to recruitment. Firstly, we used the researchers’ professional networks to connect with existing contacts and potential participants, resulting in 14 recruits. We also sent recruitment emails to mailing lists tailored to BLVSPs such as Program-L [33], which resulted in the recruitment of 16 participants. Finally, 9 participants were recruited via snowball sampling [9].

Our participants had diverse demographics. First, they reported a spectrum of vision abilities. 20 self-reported as being totally or completely blind (51%), 6 as having light and / or color perceptions (15%), and 13 participants (33%) as having varying degrees of residual vision, such as tunnel vision and 2% useful vision. Participants were located in various countries across continents, with 67% from North America (n = 26) and the rest being from Europe (n = 6), Asia (n = 4), and South America (n = 3). Participants had a wide range of work experiences; 17 had 1 to 5 years, 10 had 6 to 10 years, 5 had 11 to 20 years, 3 had 21 to 30 years, and 4 had 31 years or more years of experience working. Company sizes also varied, from startups to multinational corporations. 34 participants identified as men, 4 as women, and 1 as non-binary. Participants had various job positions, the most common being software engineers (n = 21), followed by accessibility specialists (n = 5) and other positions such as UX specialists and technical executives. Participant IDs were randomly assigned, and we selectively omit participant IDs in places where disclosure may lead to potential participant identification. Further participant information is specified in supplementary materials [16].

3.3 Procedure

We conducted audio-recorded, semi-structured interviews with 39 BLVSPs over Zoom. 30 participants were interviewed as part of the prior study on DIY tools [18], between May 2024 and July 2024. 9 additional participants were recruited specifically for this study on the use of GenAI at work, interviewed between January 2025 and February 2025. During this period, we also re-interviewed 5 of the 30 prior participants who responded to a request for additional interviews. While there was a difference of approximately six months between the times of the first and second set of interviews, the user interfaces (UI) of the tools used did not change radically and hallucination was a well-known concept throughout. This paper presents the results from the collective interviews with 39 BLVSPs. 37 interviews were conducted in English. Two interviews were

Higher-Level Themes	Themes	RQs	Section
Benefits of GenAI	Productivity Gains and Well-Being	RQ1, RQ2	4.2
	Less Reliance on Sighted Colleagues		
	Reduced Access Labor - Creation of or Dispensation with DIY Tools		
	Better Participation in Professional Conversations		
	Maintaining Professional Images		
Costs and Risks of GenAI	Suboptimal UX of GenAI Tools	RQ1, RQ2	4.3
	Meticulous Validations		
	Increased Vulnerability		
	Non-Use vs. Proceeding Despite the Risks		
Wider Impact of GenAI on Workplace Equity	Benefiting or Harming Accessibility (Awareness) in the Workplace	RQ2	4.4
	Creating More Positions for BLVSPs		
	Supporting BLV People’s Pursuit of Software Development		
	Being Recognized as a Competitive Candidate for Promotions		
	Organizational Policies around GenAI Use		

Table 1: The higher-level themes and themes along with corresponding research questions and findings section.

conducted in Portuguese as preferred by the participants, by researchers whose native language was Portuguese.

Interviews ranged from 43 minutes to 182 minutes, with an average of 72 minutes. Participants were emailed a study information sheet for review and completed a form collecting demographic information. Researchers acquired verbal consent at the beginning of each interview. Questions asked during the interviews covered multiple topics, including the GenAI tools used, the tasks for which they are used, and their opinions on using GenAI tools for work. Interview protocols are available in supplementary materials [16]. Participants were compensated at a rate of \$40 per hour via an Amazon gift card or a similar method.

3.4 Ethics and Data Collection

This study was approved by the researchers’ Institutional Review Board (IRB). All audio recordings and transcripts were stored on a secure drive accessible only to the researchers. All personally identifiable information, such as participant names, was removed from the transcripts before data analysis.

3.5 Data Analysis

All recorded audio files were transcribed and verified for accuracy by the research team. The two audio recordings in Portuguese were transcribed and translated into English by the researcher undertaking the interview, who is a native Portuguese and a proficient English speaker. We first performed thematic analysis [12] to analyze the data. The first two authors analyzed the same five randomly selected transcripts, performed open coding, and frequently met to discuss and identify similarities and differences between themes. After discussions with the identified and refined themes and reaching an agreement, the first author revisited the five transcripts to analyze them for new themes. Then, the first author coded the remaining 34 transcripts to identify themes. The identified themes were presented to the entire research team in regular meetings and further discussed. Higher-level themes were identified by the team through this process. These themes are presented in Table 1.

We did not adopt coding reliability measures such as inter-rater reliability (IRR). IRR is a statistical measure aiming to ensure “objective” and “unbiased” coding, rooted in positivism [13, 50]. However, in reflexive and interpretivist approaches to qualitative research, which we ascribe to, introducing coding reliability is “illogical” [13], as reflexive thematic analysis considers researcher subjectivity and interpretation as valuable analytical resources [13]. Therefore, we did not calculate IRR. This is also seen in other research in software engineering [23, 64].

Additionally, two researchers independently reviewed all transcripts to identify the various GenAI tools used and the tasks they supported (see subsection 4.1). In multiple half-hour meetings, two researchers reviewed and discussed results and met with a third researcher who has extensive experience in the software engineering industry, to discuss alignments and divergence, and collectively make adjustments.

We did not seek complete saturation [69] in conducting interviews. Recruiting more participants for saturation was challenging, as (1) the population size of BLVSPs is relatively small compared to sighted software professionals, (2) PWD and marginalized groups are already overburdened, with the small population already being constantly asked by researchers to participate in studies [27], and (3) this population is already shouldering significant access labor [11, 15, 17] to make their work accessible.

4 Findings

In each subsection, each high-level theme identified by the researchers is addressed. Table 1 lists the research questions each higher-level theme answers. Section 4.1 describes the use cases, and sections 4.2 and 4.3 describe the benefits and drawbacks of GenAI use. We note that in some use cases, we observed both benefits and drawbacks. Lastly, subsection 4.4 documents the impact of GenAI tools on BLVSPs’ work experiences on a higher level.

4.1 GenAI Usage by BLVSPs

Just like their sighted counterparts, BLVSPs were enthusiastic about GenAI, which “*ma[de] [their] life much easier*” (P33), enabling them

Tasks	Examples
Ideation / Brainstorming	Image, diagram, slide generation Rubber-ducking solutions
Architecture / Design	Identifying suitable libraries Making architectural decisions
Producing Code	Writing unit test cases Writing code in familiar languages (e.g., Python, C++) Writing code in an unfamiliar language (e.g., YAML, PHP)
Understanding Code	Learning an unfamiliar programming language Navigating inaccessible Maven files in the IDE Code review (e.g., Reviewing pull requests and “diff” files) Skimming lengthy legacy / unfamiliar codebases (e.g., onboarding) Debugging (e.g., log file analysis, pinpointing errors on certain lines) Asking theoretical algorithmic questions (e.g., Dynamic Programming or Greedy Solution)
Visual Descriptions	Describing screenshots of code or product UI sent by colleagues Describing technical visual materials (e.g., UML diagrams, flow charts) Describing inaccessibly formatted tables, slides, and images Verifying rendered GUI during front-end development
Creating and Delivering Presentations	Checking that presentation is visually accessible to sighted colleagues Facing directly into the camera during video conferencing or video recording
General Information Seeking	Recalling best practices, naming practices, software principles, etc. Summarizing information that would otherwise require multiple (inaccessible) web searches Skimming printed material documents to assess need for full access Summarizing WCAG guidelines for colleagues Querying shortcuts for software development tools such as GitHub and GitLab
Locating Information within Codebases and Documentations	Generating python code that summarizes educational materials without traversing each page Detecting trends and anomalies in the data file Data analysis - finding a specific key-value pair in a large JSON file
Miscellaneous	Troubleshooting when AT shuts down

Table 2: Tasks BLVSPs used GenAI tools for and example cases.

to “do things [tasks] more efficiently, and not flail around” (P16). Most participants praised these tools as a “super competitive” (P35) workaround, bringing them closer to equitable workplace experiences. Participants were fond of GenAI: “I feel it’s probably the most reliable, consistent and affordable solution compared to whatever I’ve used before” (P35).

Participants used a range of GenAI tools for work, with varying frequency of use. All participants answered having used GenAI in some capacity. 62.5% (n = 25) used it daily, sometimes multiple times a day, while the remaining 37.5% (n = 15) only occasionally used it when deemed necessary for work or out of “general curiosity” (P11). Frequently used GenAI tools included Claude (3.5 Sonnet, 3.5 Haiku) [20], DeepSeek [28], and Midjourney [52], with ChatGPT (GPT-4, GPT-4o) [19], Gemini [34], and Copilot [21] being the most used. Notably, BLVSPs actively used GenAI tools “specifically designed for... visually impaired people... [that] help me work with [existing software]” (P37) such as “Be My AI” and “Picture Smart” (JAWS feature). Five participants even created their own GenAI-based DIY tools to meet specific needs.

GenAI assisted BLVSPs with numerous software development tasks. Like their sighted co-workers, they used it to write code and boost their productivity in doing so. In addition, GenAI tools helped with tasks that are often “pretty visual” (P7), including programming

tasks like debugging and assisting with ingesting or creating visual materials like flow charts, tables, diagrams, and presentations. P1 described a major shift in his approach for handling inaccessible visuals: “Recently I dropped everything else and now I’m using LLMs to take pictures and get image descriptions.” Table 2 is a comprehensive list of tasks our participants reported to use GenAI for. Due to limited space, we only elaborate on a few select tasks below.

4.1.1 Debugging. GenAI tools greatly improved the accessibility of debugging, which traditionally poses accessibility challenges due to highly-visual error indicators and the complex structure of terminal output [66]. For instance, P38 described using ChatGPT to interpret the “confusing” output of the Linux command “psdash”: “It [psdash]’s going to show you the output in tabular format. So it looks like a table... but a screen reader cannot detect which is the process ID or which is the running time and so on... it’s hard to figure [that] out.” Before the introduction of GenAI tools, he “had to remember each column name, number of columns, and then the order, and then try to read it word by word.” To tackle this inconvenience, he decided to simply “copy the terminal output, paste it into the GPT, and then... ask a specific question that I have.” Similarly, BLVSPs used ChatGPT to avoid traversing lines of code with a screen reader to locate an

error or “write a program to find it[error]” (P13), when “a sighted person can just look at the screen and find it” (P13):

“Sometimes it’s 1000 lines. [I have to go through] character by character... It would take a whole day just to check that code. Then I paste it in ChatGPT and it says, ‘There are missing parentheses on line 998.’ I fix it and it works.” (P21)

4.1.2 Front-End Development Verification. GenAI tools empowered BLVSPs to perform visual development tasks: particularly, front-end UI development and review. For example, P13 described utilizing GenAI to validate whether his output was “visually attractive.” These tools were also used to enable participation in UI design review sessions, which rely on highly visual tools such as Figma, “a staple in every company in the SDLC” (P7):

“Before [Gen]AI, my option in the design review was to say, ‘You’re going to need to describe to me everything that’s there, and everything that’s not there.’ Which is, a huge gotcha. You can’t explain to me something that’s not there if you don’t know what’s supposed to be there. So, I can use [Gen]AI to criticize designs that people have, and ask if things exist.” (P7)

4.1.3 Interpreting Documentation. GenAI also helped our participants understand inaccessible materials, such as API documentation in improperly tagged PDFs. Several participants chose to use GenAI to ask for a description of the PDF (P35) or to restructure it into more accessible HTML or Markdown (P13).

4.1.4 Interpreting Technical Visuals. Almost all of our participants used GenAI to help interpret flow charts, diagrams, tables, and even screenshots of code, acknowledging its convenience and speed: “You just click two buttons; open the targeting menu, choose the model, and it gives you a description... A two-key combination is pretty fast and pretty convenient” (P15). For straightforward visuals such as tables and bar charts, GenAI proved highly effective, with our participants’ confidence level in using GenAI for these being “80% or 90%, something like that” (P13). However, BLVSPs mostly found GenAI less helpful for relatively complex visuals like flow charts, especially when they did not have enough context beforehand: “Flowchart is the worst one, like a process chart... If I have no [prior] idea what that is, then it’s very hard to understand” (P33).

GenAI was also “good for recognizing screenshots that [sighted colleagues] send me” (P15), usually of product UI and code snippets. P32 utilized ChatGPT to interpret the screenshots of code without alternative text and “press my magical key combination” to retrieve the code on the image. Another perk of GenAI tools that other workarounds did not have, was being able to “dig deeper” (P34) about responses when asking for such visual descriptions, without taking up too much of their colleagues’ time or feeling pressured to ask all the questions within a limited time frame.

4.2 Benefits of Using GenAI

By using GenAI tools in many aspects of their work, participants’ reliance on sighted colleagues greatly decreased **when compared with prior workflows**, while maintaining a professional image and experiencing a positive impact on their mental well-being.

4.2.1 Productivity Gains and Well-Being. Our participants believed that “GenAI makes blind engineers much more productive” (P34), noting additional productivity boosts and enhanced mental well-being. Traditionally, inaccessible tools “impacted [their] performance negatively” (P12) and “definitely takes a mental toll on you... a really bad one” (P12). BLVSPs reported substantial improvements in their work efficiency and stress levels after using GenAI: “I’m faster and I use less resources. I’m not as stressed” (P10). For many, the benefits justified paying for premium subscriptions because “it just takes me so much longer to do my job without [Gen]AI” (P14).

4.2.2 Less Reliance on Sighted Colleagues. Despite needing sighted assistance at times, participants felt uneasy burdening their colleagues. Sometimes, this discomfort stemmed from the desire to maintain independence and the fear that their accessibility requests may be misconstrued as excessive reliance. They were also conscious of the cumulative impact of repeated requests on their sighted colleagues’ productivity:

“If you’re having to repeatedly, constantly ask sighted colleagues, you can take the amount of time that you are both spending to solve this problem, you can add your rate and the person’s rate that you’re asking the question to... And the more times that that happens, the more difficult it becomes to ask for help.” (P7)

Thus, participants perceived GenAI as “the colleague that doesn’t get annoyed at me” (P7) and used GenAI tools in doing their due diligence before seeking human intervention—getting a general grasp of the task, doing their best to work around most accessibility issues—and only going to colleagues for support when absolutely necessary so the colleagues are “more inclined to help” (P13). Even when assistance from “human beings can do a much better job” (P13), participants described actively trying to “minimize it” (P13).

4.2.3 Reduced Access Labor - Creation of or Replacement for DIY Tools. GenAI streamlined the DIY process and in some cases removed the need to DIY altogether. This substantially lightened the burden on our participants who previously invested considerable personal time and energy—often outside of regular work hours—crafting custom tools to navigate workplace accessibility challenges.

Many BLVSPs recognized the benefits of DIY-ing accessible solutions in their careers, but also emphasized the exhaustion the process caused, as “it is a commitment to build the tools” (P16). They also preferred to keep work and personal lives separate and invest in personal growth and passion projects outside of work. Also commonly shared among participants was the fatigue coming from extra work: “I’m very busy and tired... I’m just trying to get my baby fed and find litter for my cat” (P16).

GenAI tools proved particularly useful in assisting BLVSPs with the ideation, brainstorming, and coding process of the DIY tool. In addition, when participants were not familiar with a particular language to be coding the DIY solution from scratch, oftentimes GenAI tools supported the swift authoring of the required solution. P10 shared an instance of utilizing ChatGPT to write a JavaScript code snippet—a language he “really did suck at”—to improve usability of Skype in supporting seamless alternating between the chat and the main call was troublesome.

While GenAI could not resolve every use case for the need to DIY, participants largely shared the notion that GenAI proved a useful addition to their toolbox that decreased the “*unwanted access labor*” (P38) often required in their work: “*Before GenAI it was just the two things, right? Ask a sighted person or build your own tool... It is still semi-frequent. But it’s significantly less than before*” (P13).

4.2.4 Affordable Alternative to Unfeasible AT. The paid subscription plans also turned out to be a much more affordable alternative to accessible tools like Braille Displays that were “*extremely unfeasible options*” (P21), which P21 highlighted can cost “*around R\$ 20,000 in Brazil (around USD \$4,000)... It would make my life easier... [But] I’m not going to spend R\$20,000*” (P21).

4.2.5 Better Participation in Professional Interactions. Participants acknowledged that descriptions of work materials, such as diagrams and flowcharts obtained through GenAI tools, led to more engaging and comprehensive professional interactions with colleagues compared to their interactions prior to the introduction of GenAI tools. P24 shared: “*I get to have conversations that I’ve never had with people before... that feels quite liberating*” (P24). Additionally, P35 was able to “*talk a little bit more visually*” at work with information gained from GenAI tools about UIs and screen layouts:

“I’ve never seen a computer screen... Before, I would be super linear as to how a screen reader would [describe it] to me... And now I’m able to speak to it a little more confidently, in terminology that they [sighted colleagues] are more familiar with.” (P35)

4.2.6 Maintaining Professional Image. As GenAI increased productivity and well being, and reduced sighted assistance and excessive access labor and provided affordability, BLVSPs were able to better manage their professional image and participate better in professional interactions. P24 shared, “*on paper, it looks like I’m a lot better because the tester just doesn’t get any incomplete pieces of work from me anymore.*” While BLVSPs used to work overtime in concern that colleagues may be “*inclined to think that I’m not giving this project the time that it deserves*” (P13) if they did not, P13 felt that “*AI has definitely helped*” maintain his professional image with less access labor.

Notably, GenAI provided protection against unwanted disability disclosure in professional settings. P7 recounted having clients who “*don’t even know that I’m blind... I prefer to keep it that way because I don’t think it detracts from the quality of my work.*” Further, P13 shared his strong preference for not disclosing his disability due to numerous prior experiences where he had “*lost projects just by sharing the fact [that I’m blind]. Many, many, many times... someone said, ‘Since you’re visually impaired, you won’t be able to [work],’ and they just terminated my contract in a few days.*” Both participants praised GenAI’s safeguarding from inadvertent disclosure: “*Now it’s easier for that [my blindness] not to even be brought up in the first place because if there’s something that’s visual in nature, OK, let me just have AI do it for me*” (P7). P13 took a step further in being extra careful to not even inadvertently give away hints at his disability by using AI to reformat code in a way that other sighted developers would do.

4.3 Barriers and Risks of Using GenAI

While we saw benefits of using GenAI tools, BLVSPs also took significant risks. In addition to general concerns like hallucinations—which took longer for BLVSPs to detect and correct—additional costs such as inaccessibility and low usability further eroded GenAI’s potential advantages. These risks and costs compounded, and often led BLVSPs to either abandon GenAI tools for already established workarounds and DIY solutions or meticulously weigh benefits against these considerable costs.

4.3.1 Suboptimal UX of GenAI Tools. BLVSPs were stressed about the dissatisfactory UX and accessibility of GenAI tools, reported feeling left behind, and shared coping strategies.

Suboptimal UX of GenAI tools for Programming. GitHub Copilot proved useful in many ways, such as obtaining summaries of legacy code or unfamiliar codebases (see section 4.2 for more). For some, the “*experience I have with GitHub Copilot is very good*” (P37) or “*bearable*” (P32). However, a number of BLVSPs reported that GitHub Copilot provided “*awful*” (P24) user experience (UX) with auditory overloads, despite the robust quality of the code it suggested. Unlike sighted users who can visually scan code suggestions and simply keep typing to ignore it, screen reader users receive Copilot’s suggestions through audio. An audible cue (e.g., a beep) is followed by the screen reader reading the suggestion aloud. This adds yet another layer of auditory load on top of the screen reader’s continuous narration of the interface. Thus, its default settings were “*really, really, chatty... They’re not sensible defaults to [BLVSPs] at all*” (P24):

“You’re writing code and it’s constantly giving you suggestions, but the default settings are for it to beep every time it gives you a suggestion, plus then it interrupts speech to tell you what the suggestion is.” (P24)

This constant aural communication, while intuitive for sighted professionals who could “*easily go back and forth.. just click and go wherever*” (P13), significantly cut down on the productivity of BLVSPs and took them “*out of the zone*” (P24). The speech interruption from Copilot extensions added another layer of aural information on top of screen reader outputs, making swift context switching challenging: “*It’s not easy, figuring out the best way [to] easily switch between that and talk to it in the same window with a Copilot extension*” (P13). P13 even preferred to pair program with a colleague for this very reason, comparing GenAI to an “*unpaid intern... it does the job well. But relying on it fully is not a good idea.*”

To cope with these issues, our participants had to either “*customize and disable certain things, or just train yourself to ignore a bunch of it*” (P13). Some felt that “*there’s no reason to install the [IDE] extension*” (P34), ignored it, and used web-interfaces to get answers “*on-demand*” (P7) and “*get the information when I want it, not when the AI thinks I want it*” (P7). P12 also “*almost always*” used Copilot’s web interface for the same reason.

Suboptimal UX of Other GenAI Tools. Many GenAI tools (e.g., Claude, Gemini), despite their capabilities, were “*not the most convenient interface to use*” (P7) for BLVSPs. ChatGPT was the tool that disappointed our participants the most, as “*screen reader navigation is not really well thought-out*” (P15). In contrast to Gemini which

provided relatively straightforward navigation “with headings for each answer” (P6), many BLVSPs lamented the difficulty of navigating between responses without headings and the absence of button labels: “You always have to reread your own prompt before you get to the reply of ChatGPT” (P10). P33 pointed out that difficulty of navigation skyrocketed when different elements were mixed together without proper distinction: “There’s a code segment that you can indicate in HTML that code is code. But those gen AI tools don’t indicate those things correctly.”

Further compounding the challenges, frequent updates of GenAI tools many times broke accessibility. Buttons shifted back and forth between being labeled and unlabeled, and graphics were inconsistently accompanied by alternative text, which forced users to constantly “re-learn how to use the tools with your assistive technology” (P9) with every update. Additionally, OpenAI lacked robust technical support; P15 described that “they don’t actually have a contact form for reporting issues with their website,” and P7 shared, “It’s really hard to actually get through to them.” Disappointing navigation experiences, along with inconsistent updates breaking accessibility and the absence of proper feedback systems created a lose-lose situation for both the company and our participants: “I feel that they are missing out on their design principles... It’s bad for me, it’s bad for the company as well” (P9).

Distressed by the inaccessibility of these tools, BLVSPs sometimes chose to DIY their own scripts and applications “that are just using the API of [AI]” (P13) so they “don’t have to use their stupid interface” (P16). Participants also felt left behind their sighted colleagues with new advances in GenAI tools (e.g., Cursor code editor) that has “so much potential” (P7), but is not yet equipped with accessibility: “I’ve seen sighted developers who really know what they’re doing crank out so much code with Cursor [code editor]. It’s easily better than Copilot. Of course accessibility is not there. And so part of me does wonder from that perspective, ‘What am I missing?’” (P7) Moreover, when blind people encounter inaccessibility in certain technologies, they may not go back: “Maybe they fixed it... But I can’t really be bothered to try... The initial experience was so poor that I just have zero interest in going back and seeing if they’ve made any changes” (P24).

4.3.2 Costs of Using GenAI. Though GenAI offered benefits like greater productivity and enhanced professional images, these also became risks for BLVSPs who were unable to quickly identify and fix errors as easily as their sighted colleagues.

Meticulous Validation. While our participants acknowledged that “there’s a magical quality to [GenAI]” (P16), persistent hallucinations bred frustration. Frequent hallucinations led our participants to liken GenAI to “a very arrogant piece of metal... its opinion about its work is excellent” (P10), and compared their interactions with GenAI to “talking to an overly confident drunken acquaintance” (P26), highlighting the need to “always, always... laterally research” (P26).

As opposed to sighted professionals, for BLVSPs detecting hallucinations was “really taxing” (P14), as it took many tabs with a screen reader, or sometimes even impossible (e.g., descriptions of technical visuals). When hallucinations persisted, participants had to “go and do extra work to fix it” (P39), which “somewhat defeats the purpose” (P27): “You struggle hard to understand what it did and

to fix it, and end up spending much more time than if you took the other path and just wrote the damn thing yourself” (P10).

Increased Vulnerability. BLVSPs’ perceived that their use of GenAI tools carried higher risks than their sighted colleagues’. They felt that “as blind people we stand to benefit the most from them” (P24), but at the same time felt uniquely more “vulnerable” (P13) to hallucinations because they were “the group that’s probably least able to tell whether they’re hallucinating or not” (P24). Consequently, BLVSPs still had to verify GenAI outputs with their colleagues, even though one of their main reasons for turning to GenAI was to avoid repeatedly asking for sighted assistance, which could be detrimental to their professional reputations. P7 described:

“Without asking another human and being the annoying colleague who’s like, ‘Hey, can you look at this?’, there’s absolutely no way for me to know if it is hallucinating. And I think that’s one of the things that does result in an unequal playing field.” (P7)

In addition, some believed that they used GenAI for certain “riskier” (P24) tasks that their colleagues wouldn’t use it for, such as analyzing log files when debugging. While participants reported that GenAI tools greatly improved the accessibility of debugging activities which are known to be inaccessible due to the amount of noise in logs, P24 “worr[ie]d that sighted people aren’t going to be using [GenAI] to analyze the logs” and an unspotted error could make make him “look stupid.” In addition, P35 worried about inconsistent formatting of GenAI-generated content when combined with his own work, fearing he “might miss something and... actually ship that out [to colleagues],” potentially impacting how their work appeared to others if even a small portion looked “different in terms of fonts and stuff like that.”

P7 articulated the difficult position BLVSPs are stuck in: while GenAI can’t always be trusted, they “have to trust it, that what it’s saying is actually true” to “get the work done” quicker and avoid constantly asking for sighted assistance. This reliance, coupled with the lack of “safety net” options (P35) available to our participants amplified the risks that BLVSPs took in using GenAI for work: “they [sighted people] have so many options to fall back on, like tutorial videos and stuff which is, again, much harder for blind folks. And it could many times be a case where I am either learning [programming] through this [AI] or I am not learning it at all or something like that. So, either generating code or not generating at all” (P35).

4.3.3 Non-Use vs. Deciding to Take the Risk. Faced with unreliable GenAI outputs and higher risks compared to their colleagues, some users deliberately chose non-use in professional settings and relied on existing workarounds or developed DIY tools, BLVSPs carefully contemplated whether to find another option or still proceed with using GenAI for completing the task at hand. Some participants “ask[ed] a couple questions” (P7) to themselves on whether GenAI would be a suitable solution. P38 noted developing an intuition: “I know what kind of questions I should be asking and what kind of questions I shouldn’t be asking,” recognizing GenAI’s “limited capacity to remember and think.” As such, they used GenAI in “low-stakes” tasks such as generating boilerplate code and brainstorming possible ideas and solutions, rather than tasks with high complexity or those that required definitiveness

While some BLVSPs chose to abandon GenAI tools and opted for already established workarounds, others continued using GenAI tools when “clearly the reward outweighs the risk” (P24).

4.4 GenAI’s Wider Impact on Workplace Equity

Many participants believed GenAI was a “game-changer” (P5, P17). GenAI “open[s] a huge space of possibilities... It is extremely promising to make the lives of people with disabilities more independent, and more equal with people without disabilities” (P21). However, they simultaneously believed GenAI had the potential to set them further back if such tools remain inaccessible or not allowed for use at work.

4.4.1 GenAI and Accessibility (Awareness) in the Workplace.

GenAI supported general workplace accessibility, but at the same time raised concerns about how general awareness about the importance of accessibility among software professionals would decrease, which could potentially perpetuate inaccessible developer tools.

Participants expressed concerns of a potential “scary trend” (P32) that those without much experience in accessibility will assume GenAI to be the “magic powder” (P15) that solves all accessibility problems: “People just believe that AI will do it somehow, and they’re pretty cool with that” (P15). Many worried that developers will entirely trust GenAI to take care of accessibility, thereby losing motivation, and absolving them of the need to attend to accessibility in tools they create:

“They just won’t bother including best accessibility practices... They won’t bother with writing descriptions, labeling buttons properly. They’ll just believe that they won’t have to pay the price for making this accessible and they will just externalize [to GenAI] the cost of using their product.” (P15)

Participants expressed frustrations about GenAI being used as an alternative for human professionals with accessibility knowledge to “cut some costs” (P15), as currently humans were far more superior: “It’s not the same. I’m sorry, soy milk is not the same as dairy milk” (P32). P35 pointed out that when GenAI produces inaccessible code and software, “it [would be] very... hard to fix” and pinpoint the accessibility problem since GenAI took over so much of the process.

Notably, one participant thought otherwise; P10 believed that accessibility being GenAI’s entire responsibility may be better, as the average developer’s knowledge about accessibility in software products is very poor: “I think that [Gen]AI knows more accessibility than other [sighted] developers... Because many developers don’t know anything at all about accessibility... I saw enough inaccessible things [built by sighted developers].” (P10)

4.4.2 GenAI’s Impact on Perceived Career Outlook. Although our participants echoed the concern that “GenAI gets way better than it’s now and starts replacing developers” (P10), they also felt that GenAI would make software engineering training materials more accessible, giving them access to a wider range of job positions, and potentially leading to recognition from superiors. For example, P37 recounted watching YouTube tutorials that used ample ç language when learning programming: “They were not very accessible...: ‘Click on this. Go to that. Go on that and do this and do that’... I was not getting it... I struggled a lot while learning programming.” However, GenAI provided an easier entry point to coding: “You

can ask [Gen]AI to list down the concepts in simpler language or in the way you understand the most.” P31 also shared a story of his acquaintance who used GenAI tools to learn programming, and that “he was more confident in wanting to go into the tech industry to do tech-type work.” As such, P38 believed that “more visually impaired people will be participating in the IT industry because of GenAI... people would be using it to learn programming a lot.”

Further, participants envisioned GenAI opening up more career opportunities and positions (e.g., UI/UX designer, prompt engineer), some of which were previously considered unattainable for BLVSPs because of the visual elements:

“Could you be a UX designer if you were blind, using GenAI?... Maybe they wouldn’t be able to use Photoshop to design a button, but maybe if they understood what they wanted, they’d have enough sight to feed that into a model and then make sure that it made the right thing.” (P24)

Some even saw “more streams opening up” (P9) for professionals with disabilities in the technical track, instead of pigeonholing them into accessibility roles against their interests.

Several participants believed the time efficiencies gained by GenAI would enable them to produce faster, higher-quality deliverables that make them “more likely...to get noticed” and “help me climb the ladder” (P7). Not only would this advance their careers, but it would help them demonstrate to superiors that they are as productive and “also equally [as] capable as a sighted person” (P5), thus contributing towards their career success.

4.4.3 Organizational Policies around GenAI Use. Our participants’ organizations had different policies around using GenAI in the workplace, which impacted them in various ways.

Permission and Encouragement. Many companies either encouraged the use of GenAI at work or did not restrict it, as long as no security or privacy was breached. P24 reported that their organization changed policies into allowing select models for work, and shared that “it was a challenge resisting the urge to use them when I knew how helpful they could be.” Notably, some companies even built internal LLMs for more convenience, but P10 was blocked from using it due to its inaccessibility: “As a company policy, they actually built a local ChatGPT for us to use, with some customizations. I’m not sure what it does, because it’s inaccessible.”

Limited Use, Selective Disclosure. Four participants’ companies only allowed select models that passed internal checking. One participant who experienced remarkable benefits from GenAI, in fear that going too much into detail about their specific use case would result in restrictions around their GenAI use, kept their use case “quite generic” and only selectively disclosed how they used GenAI to their organization: “I’m saying, ‘...I’m going to be writing some code that uses these models. It’s not audience facing. It’s just for me.’ And I don’t think that was a lie, but it was a beginner-friendly version of what I was going to be doing with them.”

Total Ban, Surreptitious Use. Our participants¹ speculated “quite a bit of unauthorized use of models that were happening internally” in companies where GenAI use was prohibited altogether. Five out

¹Here, we omit all participant IDs due to risks of identification.

of eight participants whose companies disallowed use disclosed clandestine use. Even those who were currently allowed to use GenAI emphasized that *“I won’t work at a company that doesn’t use GenAI”* or further, *“would happily use it under the radar.”* BLVSPs described the tension that they experienced as they were not willing to give up the productivity boost granted by GenAI, especially as a BLVSP: *“Do I use it behind the scenes and not tell it and keep my job? Or do I just not use it and risk lower productivity and being slower?”* Three participants cleverly *“use[d] it anyway on my personal laptop”* while not using it on corporate devices to *“save myself”*.

Participants were also *“very alert to not leave AI-generated materials that get me in trouble.”* Sometimes, they *“d[id]n’t have a choice”: “I will try my best to not use it... unless I’m absolutely forced into it... [If] they have something inaccessible...”* Another BLVSP further explained the rationale behind such decisions: *“I wouldn’t even say it’s malicious intent... at the end of the day... I can’t be pressing a bunch of buttons, to go back to the window, to see the message that someone sent in the chat while people are waiting for me [in a meeting].”* Interestingly, one, whom other employees occasionally report to, shared that they have *“turned a blind eye to it [GenAI use] a few times, and will probably continue to do so,”* as long as *“I see that they’re using AI responsibly and... their work does seem to be better in certain ways.”*

5 Discussion

Existing scholarship focuses on how software developers use GenAI to code [10, 26, 31] and how BLV individuals use GenAI for accessibility [2, 78]. To the best of our knowledge, this is the first research study to investigate how BLV professionals who work in software development utilize GenAI for everyday job tasks, including but not limited to writing code at the workplace.

5.1 To Use or Not to Use? The Higher-Risk, Higher-Return Calculus of GenAI

The *“game-changing”* returns of using GenAI were evident in several ways: beyond its support for programming, GenAI enabled independent engagement in highly visual job tasks (e.g., debugging, UI verification, interpreting technical diagrams) that previously demanded either significant manual effort (e.g., DIY tooling [18]) or routine reliance on sighted colleagues [25]. BLVSPs were also able to avoid unwanted disability disclosure as well as excessive, unpaid work dealing with inaccessible tools and tasks, which participants reported enhanced their mental well-being. These benefits in turn had a positive impact on their outlook for their future careers, as they were able to take on more responsibilities and diverse roles and be perceived by their colleagues as more competent, productive, and self-sufficient. GenAI also expanded their problem-solving toolbox, perhaps more so than it did for their sighted counterparts.

However, accompanying these substantial benefits were substantial risks, often exceeding those faced by BLVSPs’ sighted counterparts. A primary source of risk was the suboptimal accessibility and user experience of GenAI tools themselves. Our work confirms previously documented factors that inhibit the usability of GenAI tools (e.g., inaccessible UI [2], cognitive overload [31]). We document additional factors (e.g., inability to distinguish code and URL output from plain text, frequent updates breaking accessibility) as well as situating them within the workplace context. These accessibility

and usability challenges forced BLVSPs to perform *“access labor”* in re-learning how to use a given tool, building DIY tools to avoid these issues altogether, or finding other, manual workarounds, all of which their sighted colleagues did not experience. This led to lower productivity and decreased well-being, in some cases negating the gains of GenAI.

Further compounding this risk was what we refer to as the *“verification catch-22.”* While hallucinations were a general concern among both sighted [62] and blind software professionals, BLVSPs reported feeling uniquely more vulnerable. Sighted software professionals have multiple ways to rapidly confirm an answer (e.g., eyeballing the response, cross-checking with documentation, browsing Stack Overflow). However, because of the linear, aural information presentation of screen readers, BLVSPs are unable to quickly scan results as well as quickly switch back-and-forth between tools and documents. Further, referential sources (official documentation) are not always fully accessible [75]. As such, detecting errors was highly demanding or impossible, which amplified the risks BLVSPs took, and threatened to nullify the returns of GenAI tools when verification labor took too long or was unsuccessful. This created a significant irony: one of the main reasons BLVSPs chose to use GenAI was to avoid asking for sighted assistance to maintain a professional image, yet using GenAI meant they frequently had to either consult a sighted colleague or trust unverified output, thus risking their professional image.

Based on the higher-risks and higher-returns of GenAI tools identified, our participants carefully contemplated the decision to adopt or forgo GenAI on a case-by-case basis. On one hand, *using GenAI* helped them compete with sighted colleagues for recognition and promotions, but it came with increased risks of lost productivity and credibility, and threats to their job security if caught with unauthorized use. On the other hand, *not using GenAI* carried the potential to widen the gap between them and their sighted colleagues and hinder their careers, as tasks would still remain inaccessible, additional labor to compensate for accessibility was needed, and sighted colleagues would be moving forward with their own use of GenAI tools. In navigating these tradeoffs, BLVSPs often chose to bear the risk in pursuit of the benefits GenAI brought them.

Furthermore, while GenAI had the potential to reduce the gap with their colleagues, we note that the *“equal playing field”* may be a moving target, as sighted professionals are also rapidly adopting and leveraging GenAI to further their own productivity and capabilities (e.g., [44, 82]). BLVSPs thus might find themselves in a continuous race to catch up, especially in a world where demand for skills and jobs related to AI are expected to see rapid growth [32, 56].

5.2 Towards a Low-Risk, Higher-Return GenAI

GenAI tool *developers* must consider inclusive design to support non-visual access, such as making sure that GenAI tools comply with digital accessibility guidelines (e.g., Web Content Accessibility Guidelines (WCAG) [38]). One key principle in the WCAG is *“perceivability”*—*“information and user interface components must be presentable to users in ways they can perceive”* [38]— and this could be achieved through ensuring seamless screen reader compatibility and navigation, non-overwhelming auditory cues [31],

and accurate labeling of all elements (e.g., code, URLs) so they are distinguishable. Also included in this would be consistency and stability with updates, to prevent regressions in accessibility that keep BLVSPs trapped in a cycle of constant re-learning.

Researchers should study and formulate best practices in building accessible GenAI tools, so tool developers can use these guidelines to inform their work. This requires longitudinal studies investigating persistent and newly emerging effects of GenAI tools on BLVSPs (e.g., productivity, well-being, career progression) and the design of robust, non-visual verification methods to support BLVSPs in detecting hallucinations. Feasible ideas include providing internal self-validation methods [2], high-quality alternative texts for visual descriptions, and transparent confidence metrics (e.g., percentages), and third-party remote verification options.

Finally, we urge *employers* to develop clear and inclusive GenAI use policies that recognize the needs of BLVSPs. Rather than imposing uniform restrictions, policies should consider each BLVSP's needs and allow GenAI to be an accommodation with flexibility. In addition, procurement departments should ensure GenAI tools that are brought in for employee use—either commercially available or internally built GenAI tools—are accessible so BLVSPs can access and utilize them effectively and efficiently. Organizations should be cognizant that GenAI tools are not a panacea and will not address all issues; therefore accessibility awareness must be retained and inclusive work environments should continue to be promoted.

6 Limitations and Threats to Validity

In this section we discuss the credibility, reliability, and transferability of our findings as is common in qualitative research [22].

Credibility. The research team consists of scholars in multiple fields: expertise in accessibility research, working with people with disabilities, SE research experience, including one researcher with extensive industry experience. The team thus brought different perspectives to the research project. Such diversity helped the study in several ways such as minimizing risks arising from individual researchers' biases, ensuring the questions in the interview protocol were relevant to software professionals who were BLV, and interpreting the interview data through the lenses of accessibility and software engineering.

Reliability. To address risks related to inconsistencies in capturing and analyzing data, we ensured consistent data collection and analysis processes, as detailed in section 3. We provide the interview protocols and participant information for inspection in supplementary materials [16].

While we did not triangulate with additional data sources, we note that some of the identified themes (e.g., Copilot accessibility problems, using AI for image descriptions) resonate with and extend prior research into the GenAI experiences of BLV people more broadly (e.g., [2, 31]), although not in specific workplace settings, thereby increasing confidence in the accuracy of these findings.

We note that we did not always probe which specific GenAI tools or models participants used, thus parts of our findings do not specify the exact tool referenced.

Transferability. Our participants held a wide range of job roles and tenures, and were employed in various industry sectors and

across different company sizes (see section 3). Our sample had a gender skew towards men (see subsection 3.2), which is not uncommon in studies of software professionals given the overall demographic. The majority of participants resided in the USA, with other regions represented, including Europe, Asia, and South America. Thus, our findings do not represent all countries and cultures.

Finally, a particular threat to acknowledge is that new LLMs and GenAI-based tools are being released frequently; therefore we acknowledge that noted accessibility issues with specific GenAI tools may have been addressed already. If any of the noted issues have been addressed, this illustrates the observation that the accessibility of tools is often considered an afterthought [80], leaving BLVSPs unable to take immediate advantage of novel, game-changing tools when they are first made available.

7 Conclusion

GenAI tools such as ChatGPT and GitHub Copilot are rapidly becoming an integral part of software development. Yet, little is known about their impact on the workplace experiences of minority groups, such as blind and low vision software professionals (BLVSPs). This qualitative study with 39 BLVSPs explored the benefits and risks of using GenAI and its impact on career prospects. We found that GenAI was seen as a game-changer for BLVSPs, anticipation of making significant progress in leveling the playing field with sighted software professionals (for reasons such as increased productivity and reduced needs for workarounds). However, we also found that BLVSPs faced risks in using GenAI, beyond those faced by sighted developers (e.g., being more vulnerable to hallucinations, struggling with poor UX, leaving poor impressions on colleagues). The thorny choice to use or not to use GenAI posed outsized tradeoffs to BLVSPs, leaving them more vulnerable to reputational damage and career setbacks. Recommendations include GenAI tool developers to adhere to digital accessibility guidelines and for researchers to investigate robust non-visual verification systems.

Acknowledgments

We would like to thank our participants for their involvement in our study and for providing valuable insights and perspectives. This work was supported by the National Science Foundation (NSF) awards #2211790, #2326023, #2210812, and #2326489.

References

- [1] NV Access. 2025. NV Access | About NVDA. <https://www.nvaccess.org/about-nvda/>.
- [2] Rudaiba Adnin and Maitraye Das. 2024. "I look at it as the king of knowledge": How Blind People Use and Understand Generative AI Tools. In *Proceedings of the 26th International ACM SIGACCESS Conference on Computers and Accessibility* (St. John's, NL, Canada) (ASSETS '24). ACM, New York, NY, USA, Article 64, 14 pages. doi:10.1145/3663548.3675631
- [3] Khaled Albusays, Stephanie Ludi, and Matt Huenerfauth. 2017. Interviews and Observation of Blind Software Developers at Work to Understand Code Navigation Challenges. In *Proceedings of the 19th International ACM SIGACCESS Conference on Computers and Accessibility* (Baltimore, Maryland, USA) (ASSETS '17). ACM, New York, NY, USA, 91–100. doi:10.1145/3132525.3132550
- [4] Rahaf Alharbi, John Tang, and Karl Henderson. 2023. Accessibility Barriers, Conflicts, and Repairs: Understanding the Experience of Professionals with Disabilities in Hybrid Meetings. In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems* (CHI '23). ACM, New York, NY, USA, 1–15. doi:10.1145/3544548.3581541
- [5] Wajdi Aljedaani, Abdulrahman Habib, Ahmed Aljohani, Marcelo Eler, and Yunhe Feng. 2024. Does ChatGPT Generate Accessible Code? Investigating Accessibility

- Challenges in LLM-Generated Source Code. In *Proceedings of the 21st International Web for All Conference* (Singapore, Singapore) (W4A '24). ACM, New York, NY, USA, 165–176. doi:10.1145/3677846.3677854
- [6] Alphabet. 2024. Alphabet Q3 earnings call: CEO's remarks. <https://blog.google/inside-google/message-ceo/alphabet-earnings-q3-2024/>. Accessed: 2024-10-29.
- [7] Bedour Alshaigy and Virginia Grande. 2024. Forgotten Again: Addressing Accessibility Challenges of Generative AI Tools for People with Disabilities. In *Adjunct Proceedings of the 2024 Nordic Conference on Human-Computer Interaction* (Uppsala, Sweden) (NordCHI '24 Adjunct). ACM, New York, NY, USA, Article 68, 6 pages. doi:10.1145/3677045.3685493
- [8] Ameer Armary, Paige Rodeghero, and Collin McMillan. 2018. AudioHighlight: Code Skimming for Blind Programmers. In *2018 IEEE International Conference on Software Maintenance and Evolution (ICSME)*. IEEE, 206–216. doi:10.1109/ICSME.2018.00030
- [9] Sebastian Baltes and Paul Ralph. 2022. Sampling in software engineering research: A critical review and guidelines. *Empirical Software Engineering* 27, 4 (2022), 94.
- [10] Christian Bird, Denae Ford, Thomas Zimmermann, Nicole Forsgren, Eirini Kalliamvakou, Travis Lowdermilk, and Idan Gazit. 2023. Taking Flight with Copilot: Early insights and opportunities of AI-powered pair-programming tools. *Queue* 20, 6 (Jan. 2023), 35–57. doi:10.1145/3582083
- [11] Stacy M. Branham and Shaun K. Kane. 2015. The Invisible Work of Accessibility: How Blind Employees Manage Accessibility in Mixed-Ability Workplaces. In *Proceedings of the 17th International ACM SIGACCESS Conference on Computers & Accessibility* (Lisbon, Portugal) (ASSETS '15). ACM, New York, NY, USA, 163–171. doi:10.1145/2700648.2809864
- [12] Virginia Braun and Victoria Clarke. 2006. Using Thematic Analysis in Psychology. *Qualitative Research in Psychology* 3, 2 (2006), 77–101. doi:10.1191/1478088706qp0630a
- [13] Virginia Braun and Victoria Clarke. 2021. One size fits all? What counts as quality practice in (reflexive) thematic analysis? *Qualitative research in psychology* 18, 3 (2021), 328–352.
- [14] Elisa Cali, Tommaso Fulcini, Riccardo Coppola, Lorenzo Laudadio, and Marco Torchiano. 2025. A Prototype VS Code Extension to Improve Web Accessible Development. In *2025 IEEE/ACM Second IDE Workshop (IDE)*. 52–57. doi:10.1109/IDE66625.2025.00015
- [15] Yoonha Cha, Isabela Figueira, Jessy Ayala, Emory James Edwards, Joshua Garcia, André van der Hoek, and Stacy Marie Branham. 2024. "Do You Want Me to Participate or Not?": Investigating the Accessibility of Software Development Meetings for Blind and Low Vision Professionals. In *Proceedings of the 2024 CHI Conference on Human Factors in Computing Systems* (Honolulu, HI, USA) (CHI '24). ACM, New York, NY, USA, Article 933, 17 pages. doi:10.1145/3613904.3642130
- [16] Yoonha Cha, Victoria Jackson, Lauren Chu, André van der Hoek, and Stacy Branham. 2025. Supplementary material. <https://doi.org/10.5061/dryad.gmsbcc313>. Accessed: 2025-20-11.
- [17] Yoonha Cha, Victoria Jackson, Isabela Figueira, Stacy Marie Branham, and André van der Hoek. 2024. Understanding the Career Mobility of Blind and Low Vision Software Professionals. In *Proceedings of the 2024 IEEE/ACM 17th International Conference on Cooperative and Human Aspects of Software Engineering* (Lisbon, Portugal) (CHASE '24). ACM, New York, NY, USA, 170–181. doi:10.1145/3641822.3641872
- [18] Yoonha Cha, Victoria Jackson, Karina Kohl, Rafael Prikladnicki, André van der Hoek, and Stacy Branham. 2025. The Dilemma of Building Do-It-Yourself (DIY) Solutions for Workplace Accessibility. In *Proceedings of the 2025 CHI Conference on Human Factors in Computing Systems* (CHI '25). Association for Computing Machinery, New York, NY, USA, Article 47, 17 pages. doi:10.1145/3706598.3713302
- [19] Chatgpt 2025. ChatGPT. <https://chatgpt.com/>.
- [20] Claude 2025. Claude. <https://claude.ai/>.
- [21] Copilot 2025. GitHub Copilot - Your AI pair programmer. <https://github.com/features/copilot>.
- [22] John W. Creswell and Cheryl N. Poth. 2018. *Qualitative Inquiry & Research Design* (4 ed.). SAGE Publications, Los Angeles.
- [23] Faith Culas, Amisha Singh, Atharva Arankalle, Priyanka Dhopade, and Kelly Blincoe. 2025. Newcomers' experiences during debugging: A cognitive inclusivity perspective using GenderMag. *Information and Software Technology* (2025), 107932.
- [24] Maitraye Das, Darren Gergle, and Anne Marie Piper. 2019. "It doesn't win you friends": Understanding Accessibility in Collaborative Writing for People with Vision Impairments. *Proc. ACM Hum.-Comput. Interact.* 3, CSCW, Article 191 (Nov. 2019), 26 pages. doi:10.1145/3359293
- [25] Maitraye Das, Abigale Stangl, and Leah Findlater. 2024. "That comes with a huge career cost": Understanding Collaborative Ideation Experiences of Disabled Professionals. *Proc. ACM Hum.-Comput. Interact.* 8, CSCW1, Article 179 (April 2024), 28 pages. doi:10.1145/3641018
- [26] Nicole Davila, Igor Wiese, Igor Steinmacher, Lucas Lucio da Silva, Andre Kawamoto, Gilson Jose Peres Favaro, and Ingrid Nunes. 2024. An Industry Case Study on Adoption of AI-based Programming Assistants. In *Proceedings of the 46th International Conference on Software Engineering: Software Engineering in Practice* (2024-05-31) (ICSE-SEIP '24). ACM, New York, NY, USA, 92–102. doi:10.1145/3639477.3643648
- [27] Marianne Dee and Vicki L. Hanson. 2016. A Pool of Representative Users for Accessibility Research: Seeing through the Eyes of the Users. *ACM Trans. Access. Comput.* 8, 1, Article 4 (Jan. 2016), 31 pages. doi:10.1145/2845088
- [28] DeepSeek 2025. DeepSeek. <https://deepseek.com/>.
- [29] The Economist. 2024. AI and globalisation are shaking up software developers' world. <https://www.economist.com/business/2024/09/29/ai-and-globalisation-are-shaking-up-software-developers-world>
- [30] Avelino Ferreira Gomes Filho and Rodrigo De Toledo. 2015. Visual Management and Blind Software Developers. In *2015 Agile Conference*. IEEE, Washington, DC, 31–39. doi:10.1109/Agile.2015.14
- [31] Claudia Flores-Saviaga, Benjamin V. Hanrahan, Kashif Imteyaz, Steven Clarke, and Saiph Savage*. 2025. The Impact of Generative AI Coding Assistants on Developers Who Are Visually Impaired. In *Proceedings of the 2025 CHI Conference on Human Factors in Computing Systems* (Yokohama, Japan) (CHI '25). ACM, New York, NY, USA, Article 1164, 17 pages. doi:10.1145/3706598.3714008
- [32] World Economics Forum. 2025. The Future of Jobs Report 2025. <https://www.weforum.org/publications/the-future-of-jobs-report-2025/digest/>. Accessed: 2025-06-25.
- [33] Freelists. 2024. program-1. <https://www.freelists.org/list/program-1>. Online; accessed 10-Sept-2024.
- [34] Gemini 2025. Google Gemini. <https://gemini.google.com/>.
- [35] Catherine M Hicks, Carol S Lee, and Kristen Foster-Marks. 2024. The New Developer: AI Skill Threat, Identity Change & Developer Thriving in the Transition to AI-Assisted Software Development. (2024).
- [36] Xinyi Hou, Yanjie Zhao, Yue Liu, Zhou Yang, Kailong Wang, Li Li, Xiapu Luo, David Lo, John Grundy, and Haoyu Wang. 2024. Large Language Models for Software Engineering: A Systematic Literature Review. *ACM Trans. Softw. Eng. Methodol.* 33, 8, Article 220 (Dec. 2024), 79 pages. doi:10.1145/3695988
- [37] Earl W. Huff, Kwajo Boateng, Makayla Moser, Paige Rodeghero, and Julian Brinkley. 2020. Examining The Work Experience of Programmers with Visual Impairments. In *2020 IEEE International Conference on Software Maintenance and Evolution (ICSME)*. IEEE, 707–711. doi:10.1109/ICSME46990.2020.00077
- [38] Initiative (WAI), W3C Web Accessibility. [n.d.]. WCAG 2 Overview. <https://www.w3.org/WAI/standards-guidelines/wcag/>
- [39] Victoria Jackson, Bogdan Vasilescu, Daniel Russo, Paul Ralph, Rafael Prikladnicki, Maliheh Izadi, Sarah D'Angelo, Sarah Inman, Anielle Andrade, and André van der Hoek. 2025. The Impact of Generative AI on Creativity in Software Development: A Research Agenda. *ACM Trans. Softw. Eng. Methodol.* 34, 5, Article 133 (May 2025), 28 pages. doi:10.1145/3708523
- [40] Matthew Kam, Cody Miller, Miaoxin Wang, Abey Tidwell, Irene A. Lee, Joyce Malyn-Smith, Beatriz Perret, Vikram Tiwari, Joshua Kenitzer, Andrew Macvean, and Erin Barrar. 2025. What do professional software developers need to know to succeed in an age of Artificial Intelligence?. In *Proceedings of the 33rd ACM International Conference on the Foundations of Software Engineering* (Clarion Hotel Trondheim, Trondheim, Norway) (FSE Companion '25). Association for Computing Machinery, New York, NY, USA, 947–958. doi:10.1145/3696630.3727251
- [41] Eunsook Kang and Mary Shaw. 2024. I;dr: Chill, y'all: AI Will Not Devour SE. In *Proceedings of the 2024 ACM SIGPLAN International Symposium on New Ideas, New Paradigms, and Reflections on Programming and Software* (Pasadena, CA, USA) (Onward! '24). ACM, New York, NY, USA, 303–315. doi:10.1145/3689492.3689816
- [42] Kai-Kristian Kemell, Matti Saarikallio, Anh Nguyen-Duc, and Pekka Abrahamson. 2025. Still just personal assistants? – A multiple case study of generative AI adoption in software organizations. *Information and Software Technology* (2025), 107805. doi:10.1016/j.infsof.2025.107805
- [43] Mansi Khemka and Brian Houck. 2024. Toward Effective AI Support for Developers: A survey of desires and concerns. *Commun. ACM* 67, 11 (Oct. 2024), 42–49. doi:10.1145/3690928
- [44] Ranim Khojah, Mazen Mohamad, Philipp Leitner, and Francisco Gomes De Oliveira Neto. 2024. Beyond Code Generation: An Observational Study of ChatGPT Usage in Software Engineering Practice. *Proc. ACM Softw. Eng.* 1, FSE (July 2024), 1819–1840. doi:10.1145/3660788
- [45] Jenny T. Liang, Chenyang Yang, and Brad A. Myers. 2024. A Large-Scale Survey on the Usability of AI Programming Assistants: Successes and Challenges. In *Proceedings of the IEEE/ACM 46th International Conference on Software Engineering* (New York, NY, USA, 2024-02-06) (ICSE '24). ACM, New York, NY, USA, 1–13. doi:10.1145/3597503.3608128
- [46] Caroline Da Conceição Lima, Herbert Salazar, Yuri Lima, Carlos Eduardo Barbosa, Matheus Argôlo, Alan Lyra, and Jano Moreira de Souza. 2024. Generative AI Impact on the Future of Work: Insights from Software Development. In *2024 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*. 3887–3892. doi:10.1109/SMC54092.2024.10831787
- [47] Yue Liu, Thanh Le-Cong, Ratnadira Widayarsi, Chakkrit Tantithamthavorn, Li Li, Xuan-Bach D. Le, and David Lo. 2024. Refining ChatGPT-Generated Code: Characterizing and Mitigating Code Quality Issues. *ACM Trans. Softw. Eng. Methodol.* 33, 5, Article 116 (June 2024), 26 pages. doi:10.1145/3643674
- [48] Phelipe Silva Malheiros, Rayfran Rocha Lima, and Ana Carolina Oran. 2024. Impact of Generative AI Technologies on Software Development Professionals'

- Perceptions of Job Security. In *Proceedings of the XXIII Brazilian Symposium on Software Quality*. ACM, Salvador Bahia Brazil, 169–178. doi:10.1145/3701625.3701656
- [49] Aparajita S Marathe and Anne Marie Piper. 2025. The Accessibility Paradox: How Blind and Low Vision Employees Experience and Negotiate Accessibility in the Technology Industry. *Proc. ACM Hum.-Comput. Interact.* 9, 7, Article CSCW485 (Oct. 2025), 25 pages. doi:10.1145/3757666
- [50] Nora McDonald, Sarita Schoenebeck, and Andrea Forte. 2019. Reliability and Inter-rater Reliability in Qualitative Research: Norms and Guidelines for CSCW and HCI Practice. *Proc. ACM Hum.-Comput. Interact.* 3, CSCW, Article 72 (Nov. 2019), 23 pages. doi:10.1145/3359174
- [51] Wendy Mendes, Samara Souza, and Cleidson De Souza. 2024. "You're on a bicycle with a little motor": Benefits and Challenges of Using AI Code Assistants. In *Proceedings of the 2024 IEEE/ACM 17th International Conference on Cooperative and Human Aspects of Software Engineering* (Lisbon, Portugal) (CHASE '24). ACM, New York, NY, USA, 144–152. doi:10.1145/3641822.3641882
- [52] MidJourney 2025. Midjourney. <https://midjourney.com/>.
- [53] Ivan Mistrík, John Grundy, André Van der Hoek, and Jim Whitehead (Eds.). 2010. *Collaborative software engineering*. Springer, Springer Berlin, Heidelberg.
- [54] Peya Mowar, Yi-Hao Peng, Jason Wu, Aaron Steinfeld, and Jeffrey P Bigham. 2025. CodeA11y: Making AI Coding Assistants Useful for Accessible Web Development. In *Proceedings of the 2025 CHI Conference on Human Factors in Computing Systems* (CHI '25). ACM, New York, NY, USA, Article 45, 15 pages. doi:10.1145/3706598.3713335
- [55] U.S. Department of Justice Civil Rights Division. 1990. The Americans with Disabilities Act | ADA.gov. <https://www.ada.gov/>.
- [56] United States Department of Labor. 2025. Incorporating AI impact in BLS employment projections: occupational case studies. <https://www.bls.gov/opub/mlr/2025/article/incorporating-ai-impacts-in-bls-employment-projections.htm>.
- [57] World Health Organization. 2025. Assistive technology. <https://www.who.int/health-topics/assistive-technology>. Accessed: 2025-06-25.
- [58] Stack Overflow. 2024. 2024 Stack Overflow Developer Survey. <https://survey.stackoverflow.co/2024/>.
- [59] Maulishree Pandey, Vaishnav Kameswaran, Hrishikesh V. Rao, Sile O'Modhrain, and Steve Oney. 2021. Understanding Accessibility and Collaboration in Programming for People with Visual Impairments. *Proc. ACM Hum.-Comput. Interact.* 5, CSCW1, Article 129 (April 2021), 30 pages. doi:10.1145/3449203
- [60] Maulishree Pandey, Vaishnav Kameswaran, Hrishikesh V. Rao, Sile O'Modhrain, and Steve Oney. 2021. Understanding Accessibility and Collaboration in Programming for People with Visual Impairments. *Proceedings of the ACM on Human-Computer Interaction* 5, CSCW1 (April 2021), 129:1–129:30. doi:10.1145/3449203
- [61] Maulishree Pandey, Steve Oney, and Andrew Begel. 2024. Towards Inclusive Source Code Readability Based on the Preferences of Programmers with Visual Impairments. In *Proceedings of the CHI Conference on Human Factors in Computing Systems* (CHI '24). ACM, New York, NY, USA, 1–18. doi:10.1145/3613904.3642512
- [62] Guilherme Vaz Pereira, Victoria Jackson, Rafael Prikladnicki, André van der Hoek, Luciane Fortes, Carolina Araújo, André Coelho, Ligia Chelli, and Diego Ramos. 2025. Exploring GenAI in Software Development: Insights from a Case Study in a Large Brazilian Company. In *2025 IEEE/ACM 47th International Conference on Software Engineering: Software Engineering in Practice (ICSE-SEIP)*. 330–341. doi:10.1109/ICSE-SEIP66354.2025.00035
- [63] Neil Perry, Megha Srivastava, Deepak Kumar, and Dan Boneh. 2023. Do Users Write More Insecure Code with AI Assistants?. In *Proceedings of the 2023 ACM SIGSAC Conference on Computer and Communications Security* (Copenhagen, Denmark) (CCS '23). ACM, New York, NY, USA, 2785–2799. doi:10.1145/3576915.3623157
- [64] Veronica Pimenova, Sarah Fakhoury, Christian Bird, Margaret-Anne Storey, and Madeline Endres. 2025. Good Vibrations? A Qualitative Study of Co-Creation, Communication, Flow, and Trust in Vibe Coding. *arXiv preprint arXiv:2509.12491* (2025).
- [65] Venkatesh Potluri, Maulishree Pandey, Andrew Begel, Michael Barnett, and Scott Reitherman. 2022. CodeWalk: Facilitating Shared Awareness in Mixed-Ability Collaborative Software Development. In *Proceedings of the 24th International ACM SIGACCESS Conference on Computers and Accessibility (ASSETS '22)*. ACM, New York, NY, USA, 1–16. doi:10.1145/3517428.3544812
- [66] Venkatesh Potluri, Priyan Vaithilingam, Suresh Iyengar, Y. Vidya, Manohar Swaminathan, and Gopal Srinivasa. 2018. CodeTalk: Improving Programming Environment Accessibility for Visually Impaired Developers. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems* (Montreal QC, Canada) (CHI '18). ACM, New York, NY, USA, 1–11. doi:10.1145/3173574.3174192
- [67] Daniel Russo. 2024. Navigating the complexity of generative ai adoption in software engineering. *ACM Transactions on Software Engineering and Methodology* 33, 5 (2024), 1–50.
- [68] Clark Saben and Prashant Chandrasekar. 2024. Enabling BLV Developers with LLM-driven Code Debugging. doi:10.48550/arXiv.2401.16654
- [69] Benjamin Saunders, Julius Sim, Tom Kingstone, Shula Baker, Jackie Waterfield, Bernadette Bartlam, Heather Burroughs, and Clare Jinks. 2018. Saturation in qualitative research: exploring its conceptualization and operationalization. *Quality & Quantity* 52, 4 (2018), 1893–1907. doi:10.1007/s11135-017-0574-8
- [70] Noam Scheiber. 2025. At Amazon, Some Coders Say Their Jobs Have Begun To Resemble Warehouse Work. <https://www.nytimes.com/2025/05/25/business/amazon-ai-coders.html>.
- [71] Freedom Scientific. 2025. JAWS® – Freedom Scientific. <https://www.freedomscientific.com/products/software/jaws/>
- [72] Agnia Sergeyuk, Yaroslav Golubev, Timofey Bryksin, and Iftekhar Ahmed. 2025. Using AI-based coding assistants in practice: State of affairs, perceptions, and ways forward. *Information and Software Technology* 178 (2025), 107610.
- [73] Stack Overflow. 2022. Stack Overflow Developer Survey 2022. <https://survey.stackoverflow.co/2022/>. Accessed: 2025-06-20.
- [74] Andrew Stefik, Andrew Haywood, Shahzada Mansoor, Brock Dunda, and Daniel Garcia. 2009. SODBeans. In *2009 IEEE 17th International Conference on Program Comprehension*. IEEE, 293–294. doi:10.1109/ICPC.2009.5090064
- [75] Kevin M Storer, Harini Sampath, and M. Alice Alice Merrick. 2021. "It's Just Everything Outside of the IDE that's the Problem": Information Seeking by Software Developers with Visual Impairments. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems* (Yokohama, Japan) (CHI '21). ACM, New York, NY, USA, Article 487, 12 pages. doi:10.1145/3411764.3445090
- [76] Hyunjae Suh, Mahan Tafreshipour, Sam Malek, and Iftekhar Ahmed. 2025. Human or LLM? A Comparative Study on Accessible Code Generation Capability. *arXiv preprint arXiv:2503.15885* (2025).
- [77] John Tang. 2021. Understanding the Telework Experience of People with Disabilities. *Proc. ACM Hum.-Comput. Interact.* 5, CSCW1, Article 30 (April 2021), 27 pages. doi:10.1145/3449104
- [78] Xinru Tang, Ali Abdolrahmani, Darren Gergle, and Anne Marie Piper. 2025. Everyday Uncertainty: How Blind People Use GenAI Tools for Information Access. In *Proceedings of the 2025 CHI Conference on Human Factors in Computing Systems* (CHI '25). ACM, New York, NY, USA, Article 63, 17 pages. doi:10.1145/3706598.3713433
- [79] Ruotong Wang, Ruijia Cheng, Denae Ford, and Thomas Zimmermann. 2024. Investigating and Designing for Trust in AI-powered Code Generation Tools. In *Proceedings of the 2024 ACM Conference on Fairness, Accountability, and Transparency* (Rio de Janeiro, Brazil) (FAccT '24). ACM, New York, NY, USA, 1475–1493. doi:10.1145/3630106.3658984
- [80] Brian Wentz, Paul T Jaeger, and Jonathan Lazar. 2011. Retrofitting accessibility: The legal inequality of after-the-fact online access for persons with disabilities in the United States. *First Monday* (2011).
- [81] Yaman Yu, Bektur Ryskeldiev, Ayaka Tsutsui, Matthew Gillingham, and Yang Wang. 2025. LLM-Driven Optimization of HTML Structure to Support Screen Reader Navigation. *arXiv preprint arXiv:2502.18701* (2025).
- [82] Albert Ziegler, Eirini Kalliamvakou, X Alice Li, Andrew Rice, Devon Rifkin, Shawn Simister, Ganesh Sittampalam, and Edward Aftandilian. 2024. Measuring GitHub Copilot's impact on productivity. *Commun. ACM* 67, 3 (2024), 54–63.