Hindawi
Advances in Human-Computer Interaction
Volume 2019, Article ID 3271475, 14 pages
https://doi.org/10.1155/2019/3271475

Research Article

What Methods Software Teams Prefer When Testing Web Accessibility

Aleksander Bai,[1] **Viktoria Stray**,[2] **and Heidi Mork**[3]

[1]*Norwegian Computing Center, Oslo, Norway*
[2]*University of Oslo, Norway*
[3]*NRK, Oslo, Norway*

Correspondence should be addressed to Aleksander Bai; aleksander.bai@nr.no

Received 2 November 2018; Revised 1 April 2019; Accepted 14 May 2019; Published 10 June 2019

Academic Editor: Antonio Piccinno

Accessibility has become an important focus in software development; the goal is to allow as many people as possible, regardless of their capabilities, to use software. We have investigated the methods that software teams prefer when testing the accessibility of their software. We conducted a large-scale study to evaluate six methods, using a sample of 53 people who work on various software teams. We present a detailed breakdown of the results for each testing method and analyze the differences between the methods. Our findings show that there are statistically significant differences in team members' preferences, particularly for those with different roles. This implies that a software team should not choose a single method for all team members.

1. Introduction

Most software-development teams focus on making software with good usability, and both the industry as a whole and individual team members understand that products and services with poor usability will fail due to poor user experiences [1]. However, there is a need for software teams to pay more attention to accessibility testing [2]. Many countries have legislation that requires digital solutions to be accessible and universally designed; a reliance on these laws is not sufficient to raise awareness and increase responsibility within the industry [3].

Today's education for information and communication technology (ICT) specialists does not provide proper training with regard to accessibility concepts and practices [4]. As a consequence, members of software-development teams—the people who develop tomorrow's ICT solutions—gain very little knowledge and even less training in how to test for accessibility during the course of their education [5]. In many scenarios, these team members only know a few testing methods; some are only aware of the de facto method, which involves checking the WCAG [6]. This leads these workers to neglect accessibility testing because they do not have complete knowledge of the available testing methods and cannot choose the most appropriate method for a given situation.

In this article, we present the results of a study in which we gathered feedback for six methods of testing accessibility. We evaluated these methods with a sample of 53 people who are involved in software development, including representatives of the typical roles in the software process: developers, testers, designers, and managers (team and project leaders). Our motivation for this study is to compare how other, lesser-known methods compare to the de facto method (the WCAG walk-through). Earlier, we have reported the findings from seven of these participants evaluating three of the tools [7].

The remainder of the paper is organized as follows: Section 2 contains an overview of the related work for accessibility testing. We then present the method we used to gather the data in Section 3. We describe the evaluation process in Section 4. After that, we present the results from the evaluations in Section 5 and discuss the findings in Section 6. We present the limitations of our work in Section 7 and then highlight potential research directions in Section 8.

Table 1: Overview of selected methods.

Name	Type	Disabilities
WCAG walkthrough	Checklist/Guidelines	Multiple
SiteImprove	Automatic checker	Multiple
Cambridge Glasses	Simulation (physical)	Visual
Screen reader	Assistive Technology/Simulation	Visual
Dyslexia simulator	Simulation (browser plugin)	Cognitive
Personas	Heuristic	Cognitive

2. Background and Related Work

The goal of accessibility testing is to evaluate whether a given solution is accessible for a wide range of people, including people with various types of disabilities. Accessibility testing is an integral part of achieving universally designed solutions [8, 9], and companies want a greater focus on accessibility [10].

Because the developers control the code behind the solutions, they need both knowledge and engagement to ensure that those solutions are accessible [11]. An essential aspect of ensuring accessibility is finding tools and methods that engage developers, designers, testers, and other team members; greater engagement increases the likelihood that those tools and methods will be used frequently. However, software teams have limited knowledge of which techniques are most appropriate for finding and addressing accessibility issues [12, 13]. Kane [14] highlighted the lack of usability testing in agile development and proposed techniques for incorporating more such testing into already established agile practices. Other scholars have suggested ways for various testing methods to be integrated into software development [15], as well as ways for usability testing to be part of an agile sprint [16]. In addition, usability testing in iterative environments can lead to increased testing as part of a development strategy that is integrated into the work process [17].

The correct comparison of tools is not trivial, and Brajnik [18] proposed the following comparison criteria: completeness, correctness, and specificity. The comparison of methods and tools depends heavily on the accessibility model, and multiple definitions exist [19]. Other scholars have focused on often-neglected aspects of the various testing methods, such as their inherent benefits and drawbacks [20]. Bai et al. [21] previously evaluated many methods for testing accessibility in order to determine which methods are best for identifying various types of problems.

3. Accessibility Testing Methods

Many methods exist for accessibility testing [22]. For instance, the W3C [23] listed over 100 tools for checking web accessibility. It is challenging to provide a good overview of these testing methods, as there are so many of them and as they are often complex; it is also difficult to determine which types of issues each method is best at identifying [21].

In this study, we evaluate methods that test visual or cognitive aspects, in addition to methods that cover a broad range of disabilities. Users with visual disabilities often have difficulties using ICT solutions, since these solutions often are based on visual information and navigation. Users with cognitive disabilities can also have difficulties using digital solutions because those solutions contain significant complex information. Cognitive disabilities are also challenging to test for [24]. Many other disabilities and impairments could be included, but because of limited scope and time, we decided to focus on these two disability groups, as well as on methods that cover all disabilities and impairments.

Based on Bai et al. [25], we selected the de facto methods for each of the categories, resulting in the methods listed in Table 1. We also focused on selecting testing methods that are intended to be simple to install and use. Both the screen reader and the Cambridge Simulation Glasses can test for visual accessibility, whereas the Dyslexia Simulator and personas can test for cognitive accessibility. Personas can also be virtually used for any disability, but the ones that we used targeted cognitive disabilities. Both the WCAG walk-through and the SiteImprove method cover multiple disabilities and impairments.

It is important to select tools that are complementary and that can be used together in order to provide a new perspective Fuglerud [9]. For instance, the Cambridge Simulation Glasses and screen reader can be used together, even though they both cover the visual range. This is because a screen reader gives a more technical view of the issues, whereas the Cambridge Simulation Glasses give a more personal view of the issues. There are also multiple ways to combine the methods in various orders [26].

3.1. WCAG Walk-Through. We chose to include a WCAG walk-through, as it is the de facto standard for testing accessibility. Several countries legally enforce part or all of the WCAG 2.0 standard [6].

To conduct a WCAG walk-through, we created an Excel document with all 61 of the criteria from WCAG 2.0. We sent the document after the in-person evaluation and reminded the participants to complete it twice (after two days and after two weeks). For each criterion, the participants could select *pass, fail, or not applicable*, depending on their evaluation. We used a local translation of the WCAG criteria and also provided links to the W3C's WCAG 2.0 standard.

3.2. SiteImprove. We assessed multiple automatic checkers (including Wave; [27]) during the pilot session, and we chose SiteImprove [28] because it had the best feedback regarding the user interface and layout. Bai et al. [25] also recommended SiteImprove.

For SiteImprove, we gave the participants instructions on how to use the Chrome extension, and we explained how to install it. We also informed the participants that automatic checkers have limitations in what they can practically check [29]. For instance, they can check whether an image has an alternative text, but they cannot determine whether the text itself is accurate or helpful for those who use screen readers.

3.3. Cambridge Simulation Glasses. The Cambridge Simulation Glasses [30] do not simulate a particular eye condition or disease. Its effects are representative of an inability to achieve the correct focus, a reduced sensitivity in the retinal cells, and cloudiness in the internal parts of the eye. These problems typically are the result of aging, one of several eye conditions or diseases, or a failure to wear appropriate corrective glasses. We further explained that the simulation of general visual impairments can reveal issues with contrast and text sizes.

By stacking multiple pairs of glasses, wearers can increasingly degrade their visual conditions. Before starting, we calibrated the number of glasses that each participant needed. We used the 1% test, for which 99% of the population must be able to use the solution for it to be acceptable. Most participants used two glasses, but some required three glasses.

3.4. Screen Reader. We informed the participants that blind or visually impaired individuals cannot use vision to read the content of a web page, so they instead use a screen reader to parse the content. The content is either spoken aloud or sent to a braille reader as the user navigates the page. We used NVDA [31] for the participants who used Windows and VoiceOver [32] for those who used macOS.

Before the evaluations, we sent an email with a link to some resources and instructions, which we asked the participants to read. We sent different emails for Windows and macOS users. We also asked the participants to watch a short movie that explained how to use a screen reader as a sighted developer or tester. The tips in the video included disabling speech and turning on the speech viewer plugin, which shows a textual output from the screen reader. These tips make testing more manageable for those that are not proficient with screen readers.

3.5. Dyslexia Simulator. To simulate the experience of dyslexia, we used the Dyslexia Simulation Chrome extension [33]. This extension was developed in collaboration with dyslexic people and is meant to help the users understand what many dyslexic people experience when reading. We explained to the participants that this does not necessarily provide a good simulation of dyslexia but that it does provide insight into what people with dyslexia experience when they visit web pages. The extension constantly moves the letters within words around, so the users need to use considerable concentration in order to recognize words and read the content of a web page.

We stressed that although this is not exactly how dyslexics experience reading, it does help nondyslexic people to experience text that is approximately as difficult to read as normal text is for dyslexic people. For dyslexic people, unstructured information and long words are particularly difficult to read, as they require more concentration. We explained that this effect is what helps a simulation reveal the consequences of dyslexia. We also provided the participants with some background information about the dyslexic population and dyslexia itself.

3.6. Personas. The use of personas is a well-known method for becoming more familiar with a user group [34]. Personas are often used in ICT to represent users in target groups. They are typically used during the specification process, as well as during development, testing, and marketing. We gave the participants instructions, such as how to map users' needs (which is essential when creating personas). We selected a persona with cognitive challenges and one with challenges associated with attention deficit hyperactivity disorder (ADHD). However, in the latter case, instead of focusing on the diagnosis, we focused on the challenges that come with this disorder because this makes it easier for people to visualize the persona.

We used publicly available personas that were developed for a separate project [35].

4. Evaluation Process

4.1. Setup. We conducted two pilot studies before we started the evaluation. In the pilot studies, we sought to verify that the scenarios and selected methods were appropriate; we adjusted the final study based on the feedback from the pilot study. The most significant adjustments involved making the time control stricter and using an external, publicly available website. We wanted the participants to be able to use their own solutions, but this caused trouble for some because of login restrictions. The participants also seemed to be more relaxed when they used an external website than when they were restricted to an internal page.

After the pilot sessions, we conducted the main evaluations over a period of 4 months. We included seven software teams across six public and private companies in the evaluations. In the end, we gathered feedback from 53 participants. Each participant evaluated two or three methods, depending on how much time they had available. On average, the participants spent between 60 and 70 minutes on the evaluation sessions. The participants used their own laptops throughout the sessions, as this made logistics easier and as we wanted to ensure that the participants were already familiar with the equipment.

We used a well-known public website and asked the participants to consider five scenarios using the methods. We used the same setup for all participants. The participants had 10 minutes to test each method before filling out the evaluation form for that method.

The participants spent on average 3 minutes filling out each of the evaluation forms. We told participants to focus on the method itself rather than on the scenarios. Even though we instructed the participants thoroughly in advance about how to complete the survey, two participants misunderstood the process and did not master the concepts involved until the second or third evaluation. For those participants, we asked them to reevaluate the previous methods and deleted

their original evaluations for those methods. To reduce the influence of the order of the evaluations, we put the methods in different orders for different participants.

We collected all the responses anonymously through Google Forms [36]. We removed all the incomplete evaluations from the results, resulting in 176 total method evaluations. We used the USE questionnaire [37], which consists of 30 questions across four categories: Usefulness, Satisfaction, Ease of Use and Ease of Learning. We also considered other questionnaires, such as SUS [38], but we chose the USE questionnaire because we wanted to compare methods across various categories.

The USE questionnaire is constructed such that the respondents use a 7-point Likert scale (from 1, strongly disagree, to 7, strongly agree) to rate their agreement with various statements. We used the original (English) versions of the questions to avoid confusing the respondents with incorrect translations. During the pilot study, we verified that the participants understood all the questions. We used all 30 original questions even though factor analysis has shown that the questionnaire could be reduced to 21 questions [39].

The four categories of the USE questionnaire provide valuable information about the participants' evaluations of each method. Usefulness can be interpreted as how effective and efficient a method is when testing accessibility. Satisfaction relates to how pleasing a method is to use, including whether the participants want to use it again. Ease of Use relates to how simple the method is to use on an occasional basis. Ease of Learning involves how simple the method is to understand. Together, these four categories provide a good overview of the advantages and disadvantages of each method.

We interviewed all the participants after their evaluations. We used an interview guide, and the interviews lasted 10 minutes, on average.

4.2. Participants. The sample comprises 53 participants who evaluated one or more methods. Team leaders or project owners recruited all of these participants. We asked each participant to fill out a background survey, which was anonymous. Unfortunately, not everyone did this, so we only have full background data on 42 participants.

In this paper, we use standard notations for mean, standard deviation, and standard error (μ, σ and $\sigma_{\bar{x}}$, respectively). The participants' ages are well-distributed: $\mu = 37.2$ ($\sigma = 10.4$) years. The data have a higher representation of men (74%) than women (26%), which is expected, as the ICT industry is dominated by men.

The target population comprises the members of web-development projects; we thus recruited all participants from web-development teams. The participants' amount of experience also had a good distribution: $\mu = 11.7$ ($\sigma = 8.8$) years.

We asked the participants about the domains they work in and their roles. They could use predefined choices or input their own answers. For domains, the majority (58.6%) of the members work in the front-end domain with user interfaces, whether as interaction designers or as developers (or both); another 9.9% worked in the back-end domain, which indicates that they have no direct connection with the end users. However, the back-end developers can still influence user interfaces in numerous ways. It is not uncommon for back-end services to generate graphics, tables, content, and so on; thus, the back-end developers must also know about accessibility and usability. The last large group of 19.6% comprises full-stack developers, which involves both back-end and front-end developers. Designers made up 7% of the sample; finally, those who did not specify a domain comprised 4.9% of the sample.

Table 2: Overview of participants.

Role	Count	Avg. age	Avg. experience
Designers	12	32.5	8.3
Developers	27	36.5	11.8
Managers	4	51	25.0
Testers	10	41.2	13.5
Total	53	37.2	11.7

For the distribution of roles, a plurality (40.5%) are developers; another 26.2% are designers (either graphical or interaction designers). In addition, 31.0% are testers. Finally, a few members (2.3%) have a management role. This population of participants is typical of agile teams. An overview of the participants is provided in Table 2.

We also asked the participants to self-evaluate their knowledge of universal design (UD) [8] by asking "How do you rate your own competence within UD?" We used a Likert scale from 1 (very poor) to 7 (very good). We asked about UD and not accessibility, as the former is the broader term Henry [40]. As shown in Figure 1, the participants' scores are a bit below average ($\mu = 3.7$, $\sigma = 1.3$, $\sigma_{\bar{x}} = 0.15$).

Next, we asked the participants about tools that they have used recently ("Which of the following tools or methods have you used for testing UD in the last 6 months?"). The choices included six methods, as well as a "None of the above" option; the participants could also add a method that was not on the list. As shown in Figure 2, a high number of participants (35.7%) had recently used WCAG or other checklists; even more (45.2%) had recently used a screen reader. Another 38.1% had not recently used any methods for testing UD; this was quite surprising because almost 60% of the participants work in the front-end domain. One explanation is that some of the participants rely on knowledge and experience, so they feel that they do not need to use any other tools.

Both the age and experience distributions are wide, and all the common professional roles (e.g., developers, testers, and designers) are well-represented. Based on the fact that most of the participants experienced no accessibility training during their education [41], we assumed that most of the testing methods would be unfamiliar to many of the participants. This is supported by the answers shown in Figure 2.

We thus can be reasonably sure that the study's population is typical of the wider population of software-development team members. For more background details regarding the participants and setup, see [42]

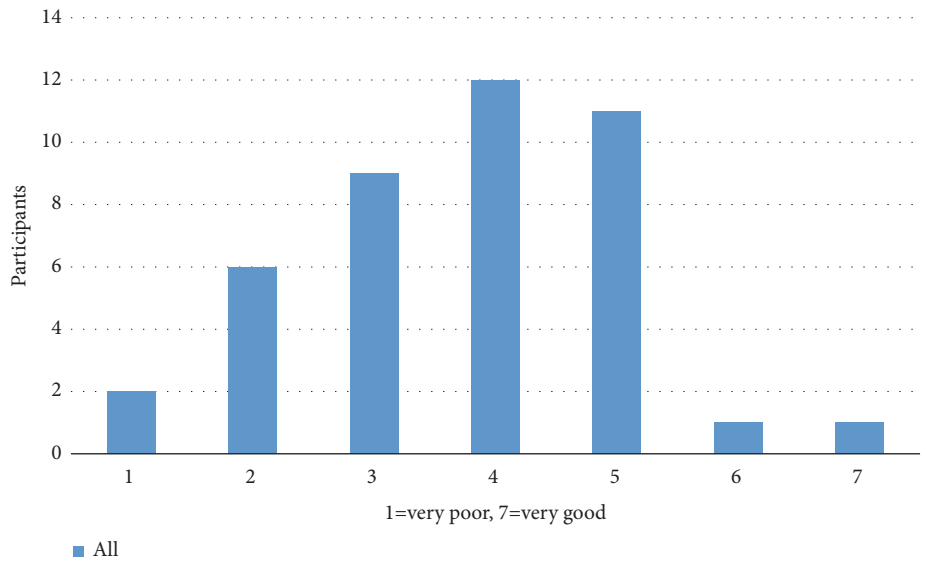

Figure 1: Self-evaluation.

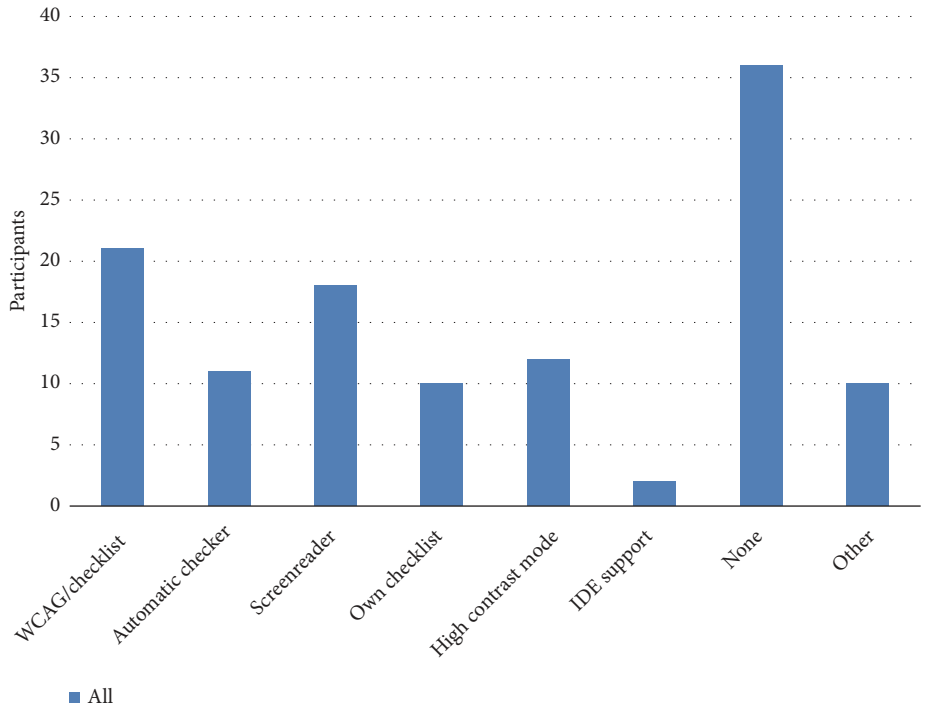

Figure 2: Which tools have been used.

5. Results

All participants managed to install the tools in a short time (less than 5 minutes).

5.1. WCAG Walk-Through. A subsample of 19 participants evaluated the WCAG walk-through method. This is the fewest number of evaluations in the study, despite the reminders that we sent to complete the evaluations for this method. The participants did not have much motivation to conduct a WCAG walk-through, and this is reflected in the evaluations.

As Figure 3 and Table 3 show, this method scores poorly. A score of 4 is average. All the categories for the WCAG walk-through have scores below 4, and the participants stated the method is especially hard to use (Ease of Use: $\mu = 2.89$; $\sigma = 0.50$).

On the positive side, the participants stated that the WCAG walk-through is a reasonably useful method ($\mu = 3.68$; $\sigma = 0.54$). One of the questions in the Usefulness category stood out from the rest: Question 3, "Is it useful?" ($\mu = 5.00$; $\sigma = 1.34$). This might imply that most participants think the WCAG walk-through is useful in itself but that the other attributes of the method make it tedious to use.

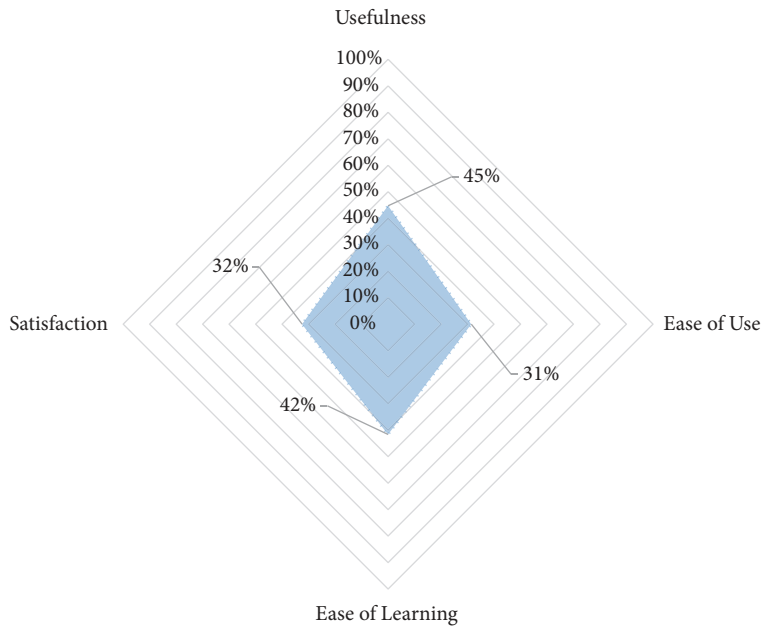

Figure 3: WCAG walk-through evaluation results.

Table 3: WCAG walk-through evaluation results.

Category	Avg.	Std. dev.	Std. err.
Usefulness	3.68	0.54	0.12
Ease of Use	2.89	0.50	0.11
Ease of Learning	3.50	0.29	0.07
Satisfaction	2.95	0.52	0.12
Total	3.19	0.61	0.14

Table 4: Dyslexia Simulator evaluation results.

Category	Avg.	Std. dev.	Std. err.
Usefulness	4.97	0.41	0.08
Ease of Use	4.98	0.49	0.09
Ease of Learning	6.41	0.18	0.03
Satisfaction	4.73	0.38	0.07
Total	5.11	0.66	0.12

This is also reflected in its low satisfaction score ($\mu = 2.95$; $\sigma = 0.52$), which indicates that most participants do not think the WCAG walk-through is satisfying to work with.

The WCAG walk-through had very low scores overall ($\mu = 3.19$; $\sigma = 0.61$); these scores are worse than we anticipated. The 95% confidence interval for the WCAG walk-through is between $\mu = 2.92$ and $\sigma = 3.5$.

We detected a slightly hostile attitude among the interviewees toward the WCAG walk-throughs, often because they had been forced to conduct WCAG walk-throughs. We do not know if this is representative of all users; we suspect some of these negative results are due to bad previous experiences and do not represent objective views of the method itself.

5.2. Dyslexia Simulator. A subsample of 29 participants evaluated the Dyslexia Simulator, and as Figure 4 shows, they regarded this method highly. This is one of the best methods overall; it has the second-highest evaluation in the Ease of Learning category: $\mu = 6.41$ ($\sigma = 0.18$). This is expected because the method only requires the push of a button in the browser to enable. Both Usefulness and Ease of Use also have high scores: $\mu = 4.97$ ($\sigma = 0.41$) and $\mu = 4.98$ ($\sigma = 0.49$), respectively.

Overall, the Dyslexia Simulator method scores highly, particularly in the category of Ease of Learning, as Table 4 shows. The method itself scores well above neutral (4), and it might have got even better scores if not for some bugs in the extension. This is also reflected in a subquestion for Ease of Use ("I don't notice any inconsistencies as I use it"; $\mu = 3.86$; $\sigma = 1.74$), which has a score well below that of all the other subquestions. This is also probably connected to the fact that the extension does not adapt all the text on a web page.

The Dyslexia Simulator scores very highly overall ($\mu = 5.11$; $\sigma = 0.66$), with a 95% confidence interval between $\mu = 4.87$ and $\mu = 5.35$. This is higher than we had foreseen, particularly because we expected more participants to misunderstand the idea behind the method.

In the interviews, many participants said that this method opened their eyes to other experiences. The participants also liked, first, that this method made it easy to visualize problems such as there being too much text on a web page and, second, that the method put emphasis on writing clear and readable text. One participant put it like this: *"It is a good reminder to not write complicated [sentences], but instead [to use] simple and good language."* On the negative side, several noted that it can be difficult to understand how effective the method is at finding issues for people with dyslexia. One

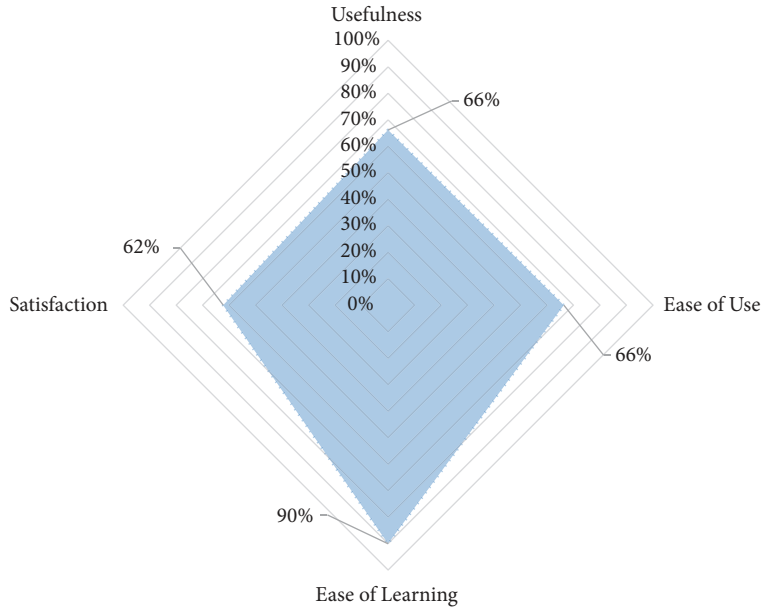

Figure 4: Dyslexia Simulator evaluation results.

participant commented, *"It is difficult to understand when something is wrong,"* and another participant stated, *"The dyslexia plugin is less useful than the other tools."*

This eye-opening experience and the fact that there are few other methods for testing readability help explain why the Dyslexia Simulator has good evaluations. However, even though the method has high scores, several participants did not understand the difficulties of having dyslexia or how the challenges associated with dyslexia should be solved. Some also had difficulty understanding that the method simulates the experience of dyslexia rather than dyslexia itself. On the other hand, several participants said that this would be a welcome tool for organizations to visualize challenges with significant complex text on their website.

5.3. Cambridge Simulation Glasses. This method had the most participant evaluations, as nearly every participant (50 in all) completed the evaluation. Not surprisingly, this method scores very well on both Ease of Learning ($\mu = 6.80$; $\sigma = 0.10$) and Ease of Use ($\mu = 6.12$; $\sigma = 0.43$), as it only requires wearing a pair of glasses. The Ease of Use scores for the Cambridge Simulation Glasses are probably reduced somewhat because of the need for calibration before use. For the Usefulness category, the method scores highly ($\mu = 5.41$; $\sigma = 0.54$); the score for the subquestion "Is it useful?" ($\mu = 6.38$; $\sigma = 0.94$) was also well above the mean.

For the Satisfaction category, this method scores the highest of all the methods ($\mu = 5.54$; $\sigma = 0.69$); this is also reflected in the interviews. Many participants mentioned that this method is pleasant to use, and many also tried the glasses on their own after the session ended. During the interviews, many participants also mentioned that the method increased their awareness of the challenges associated with poor contrasts and small fonts.

As Figure 5 and Table 5 show, the Cambridge Simulation Glasses has high scores in all categories and had the highest overall score ($\mu = 5.88$; $\sigma = 0.69$) of all the methods. The 95% confidence interval for the method is between $\mu = 5.69$ and $\mu = 6.07$.

Table 5: Cambridge Simulation Glasses evaluation results.

Category	Avg.	Std. dev.	Std. err.
Usefulness	5.41	0.54	0.08
Ease of Use	6.12	0.43	0.06
Ease of Learning	6.80	0.10	0.01
Satisfaction	5.53	0.69	0.10
Total	5.88	0.69	0.10

In the interviews, the most frequently mentioned positive aspects of this method are its simplicity, ease of use, and speed. One tester stated, *"The glasses were very easy to use – it was a very low-threshold at [a] low cost. I got a sense of what it was like to have those eye challenges."* Almost all participants liked being able to keep the glasses on their desks and only put them on to do quick verifications when needed. Some participants mentioned that the glasses are tiresome to use, and some also had problems getting the glasses to sit comfortably. In general, the designers rated this method most highly, perhaps because they regularly use digital tools to check for contrast errors. One tester commented that designers often are tempted to use gray for a professional look and noted that the glasses reveal contrast errors. In general, everyone seemed to gain empathy for users who faces challenges associated with low vision. One tester stated, *"Honestly, I did not expect those glasses to help that much."*

5.4. Personas. The personas method received evaluations from 21 participants, making it one of the methods with the fewest evaluations. This method has good scores in all categories, as shown in Figure 6, but it scores a little lower on

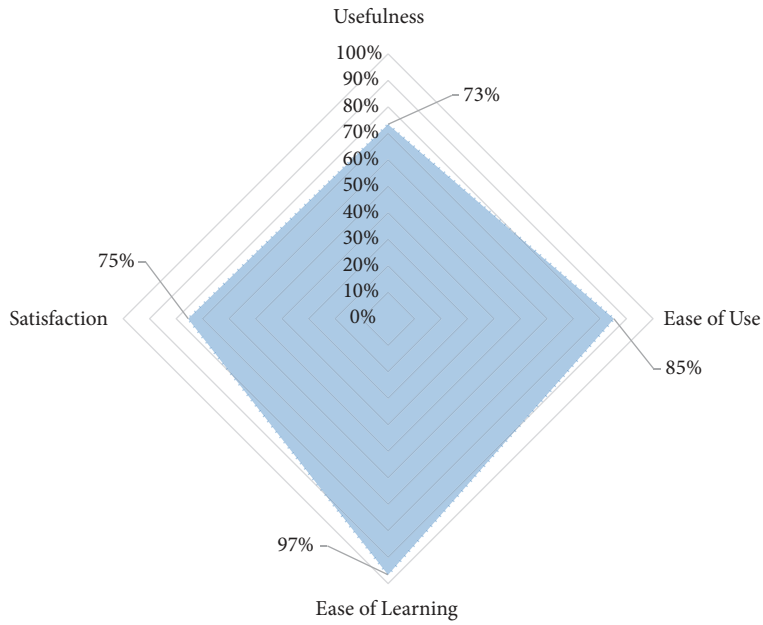

Figure 5: Cambridge Simulation Glasses evaluation results.

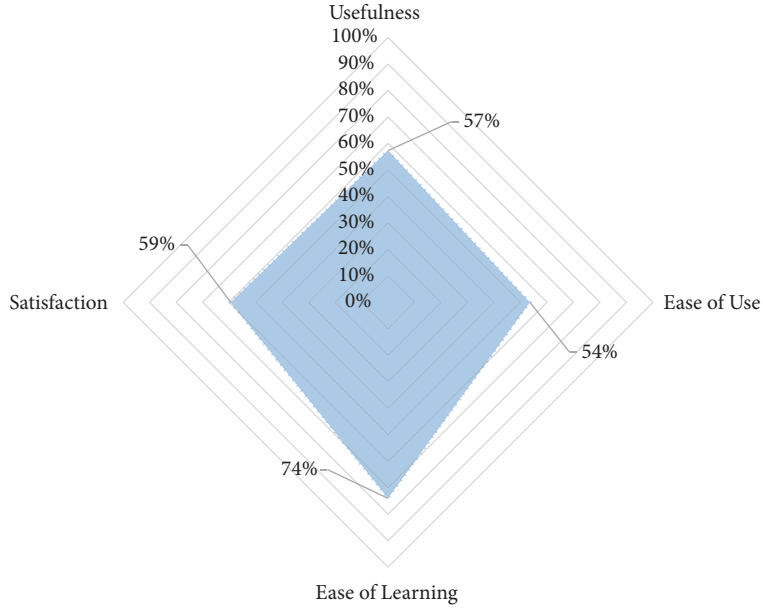

Figure 6: Personas evaluation results.

the Usefulness category than most of the other methods do. This is not a bad score ($\mu = 4.44$; $\sigma = 0.39$), but it is still lower than the score for all the other methods except the WCAG walk-through.

We are surprised to see such high evaluations in the Ease of Learning category ($\mu = 5.44$; $\sigma = 0.32$), as this method is difficult to learn and understand. However, in the Ease of Use category, the method has a relatively low score ($\mu = 4.21$; $\sigma = 0.41$), which indicates that the participants think the method is easier to learn than to use in practice. We are not surprised to see close-to-average scores on Ease of Use. The personas method has the second-lowest score in the Ease of Use category, which indicates the complexity of understanding the persona and gaining empathy for other viewpoints.

Table 6 shows that the personas method does quite well, with scores above average. Several of the participants had experience with the method and used it actively in their work, and we found that to be very positive. Not all participants used the method as designed, but we do not think that is a problem as long as it creates awareness around the various challenges that users can have. However, almost nobody had

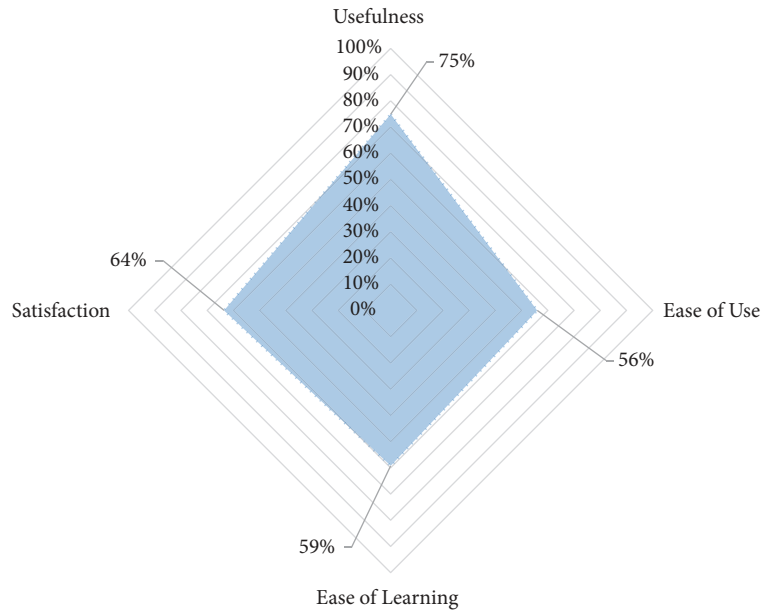

Figure 7: Screen reader evaluation results.

Table 6: Personas evaluation results.

Category	Avg.	Std. dev.	Std. err.
Usefulness	4.44	0.39	0.09
Ease of Use	4.21	0.41	0.09
Ease of Learning	5.44	0.32	0.07
Satisfaction	4.56	0.53	0.12
Total	4.52	0.57	0.13

Table 7: Screen reader evaluation results.

Category	Avg.	Std. dev.	Std. err.
Usefulness	5.49	0.43	0.09
Ease of Use	4.36	0.39	0.08
Ease of Learning	4.56	0.25	0.05
Satisfaction	4.81	0.46	0.10
Total	4.79	0.61	0.13

personas with some cognitive challenges; several participants commented on this in the interviews. We got the impression that they wanted to have more personas with cognitive challenges to increase understanding and awareness.

The personas method scores well overall ($\mu = 4.52$; $\sigma = 0.57$), with a 95% confidence interval between $\mu = 4.28$ and $\mu = 4.76$.

During the interviews, many participants expressed appreciation that this method can be easily combined with other methods, such as the Cambridge Simulation Glasses. A surprisingly large number also liked the role-playing part of the persona; we suspect that this might be because it is quite different from how they work on a regular basis. However, several noted that the method is hard to master and expressed that disregarding their own habits and experience was difficult. A participant commented by stating that personas requiring thinking all the time and in a different way than usual. Personas offer a creative way to work, but the users often forgot the roles that they were playing. Some people also thought the method was too subjective to interpretation, and several were unsure that they had interpreted the personas correctly.

5.5. Screen Reader. In all, 22 participants evaluated the screen reader: 15 using the NVDA screen reader and seven using the VoiceOver screen reader. The VoiceOver version has marginally better scores than the NVDA version, but the difference are not statistically significant. We therefore present the combined scores for both screen readers in Figure 7 and Table 7.

As expected, the screen reader method has a high score ($\mu = 5.49$; $\sigma = 0.43$) in the Usefulness category. More surprisingly, the method also scores well in Ease of Learning ($\mu = 4.56$; $\sigma = 0.25$) and Ease of Use ($\mu = 4.36$; $\sigma = 0.39$). We had expected lower scores because this is a very complex method to learn and to use. We probably influenced the results by providing some introductions in the email that we sent in advance of the study; this should be taken into account, particularly for the Ease of Use category. We recommended using the speech viewer plugin, which likely helped the novice users.

Even though the method scores high in the Usefulness category, it has even higher scores for the subquestion "Is it useful?": $\mu = 6.32$ ($\sigma = 0.76$). This is almost as high as the Cambridge Simulation Glasses method has for the same subquestion. In the Ease of Use category, the subquestion "I can use it without written instructions" has a score that is well below average for the category ($\mu = 3.77$; $\sigma = 1.70$). None of these results are surprising, and they confirm our expectations that screen readers are useful but complicated.

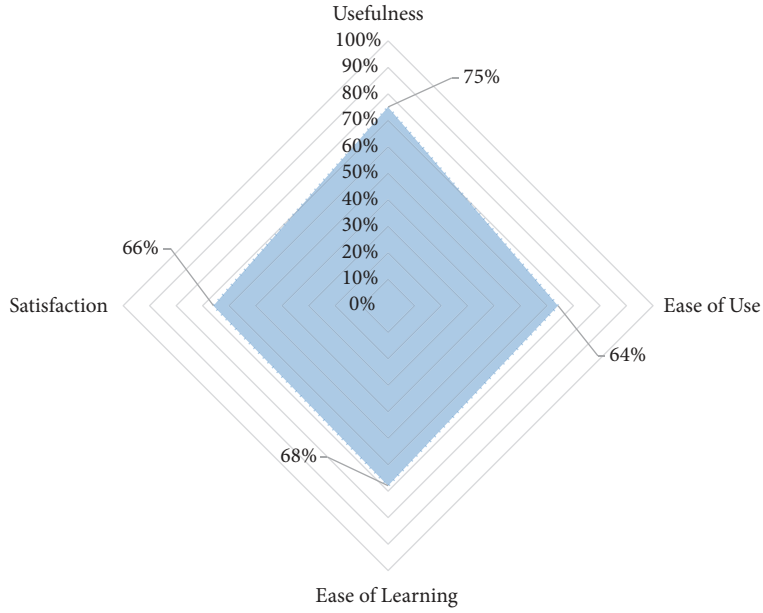

Figure 8: SiteImprove evaluation results.

As shown in Table 7, this method does well overall (μ = 4.79; σ = 0.61). The 95% confidence interval is between μ = 4.53 and μ = 5.04.

Many participants noted that this method gave an eye-opening experience similar to that of the Cambridge Simulation Glasses. Many said during the interviews that they gained a new awareness for people who use screen readers or keyboard navigation. Most participants also felt that this method uncovered most of the critical issues, which is probably because an incorrect encoding makes the screen reader impossible to use in many cases. The negative aspects that the participants mentioned are expected because the screen reader method is difficult to both install and master. Another disadvantage some participants mentioned is that the method requires more than just a prototype of a product or service to be used. They argued that prototypes often are not focused on structure or accessibility. We have seen this argument before, as it is often used to postpone testing. Many scholars have shown that this is not a good approach, however, because it increases both the risk and the cost of making changes late in the development phase [5, 43].

5.6. SiteImprove. A subsample of 35 participants evaluated the SiteImprove method, which made this the second-most-evaluated method. As Figure 8 shows, the SiteImprove method has a high score for all categories. It scores very highly in the Usefulness category (μ = 5.50; σ = 0.30), which is the highest category score for any of the methods. It is not surprising that the participants gave this method good evaluation for Ease of Learning (μ = 5.08; σ = 0.24) because most of its operations are provided automatically after starting the extension.

The method also scores highly in Ease of Use: μ = 4.85 (σ = 0.28); we expected a lower score here because the method uses many complicated terms and advanced terminology.

Table 8: SiteImprove evaluation results.

Category	Avg.	Std. dev.	Std. err.
Usefulness	5.50	0.30	0.05
Ease of Use	4.85	0.28	0.05
Ease of Learning	5.08	0.24	0.04
Satisfaction	4.96	0.30	0.05
Total	5.08	0.39	0.07

The SiteImprove method did very well overall, as Table 8 shows (μ = 5.08; σ = 0.39). The 95% confidence interval is between μ = 4.95 and μ = 5.21.

During the interviews, many participants mentioned that they liked the fact that they received their results immediately and that the information was well-structured and objective. The participants also liked that the method revealed many issues in a short time. However, many had problems understanding the exact locations and nature of the issues. One tester commented, *"I think it is difficult sometimes to understand what they mean."* Some mentioned that it is easy to ignore repeating information, but the most prominent objection for most was that this method requires prior knowledge of WCAG. The SiteImprove method also requires a working prototype and cannot be used on design sketches.

6. Discussion

In Figure 9, we plot all the methods for easier comparison. The WCAG walk-through method stands out as having the lowest scores. The Usefulness and Satisfaction categories have

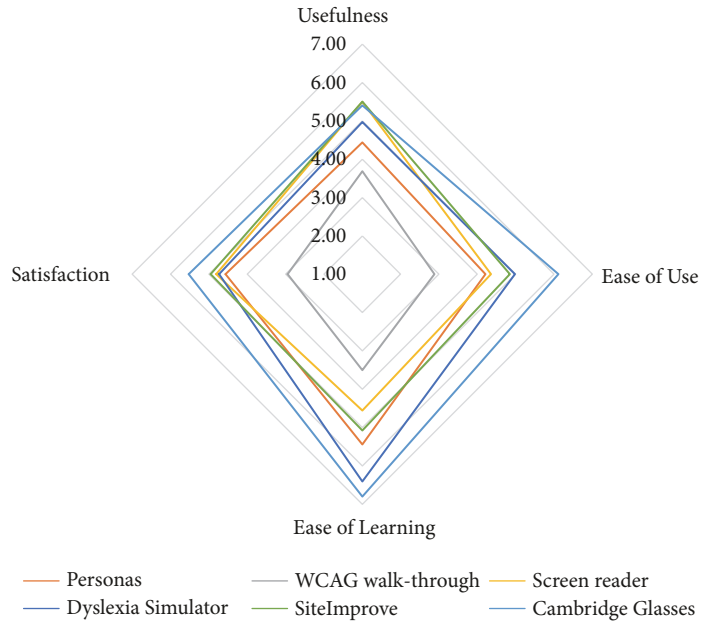

Figure 9: All evaluation results.

Table 9: All evaluation results.

Method	Usefulness	Ease of Use	Ease of Learning	Satisfaction	Total
Cambridge Glasses	5.41	6.12	6.80	5.53	5.88 ± 0.19
Personas	4.44	4.21	5.44	4.56	4.52 ± 0.24
WCAG walk-through	3.68	2.89	3.50	2.95	3.19 ± 0.27
Screen reader	5.49	4.36	4.56	4.81	4.79 ± 0.25
Dyslexia Simulator	4.97	4.98	6.41	4.73	5.11 ± 0.24
SiteImprove	5.50	4.85	5.08	4.96	5.08 ± 0.13

relatively low levels of variation between methods, whereas the Ease of Learning and Ease of Use categories have relatively high levels of variation.

We find it very promising that almost all of the methods have a high overall score, as shown in Table 9. We are surprised that the screen reader scores so highly because it is often considered to be complex and difficult to use. The Cambridge Simulation Glasses also have a high score, which corresponds well with the impression we gained during the testing sessions. On the other hand, although we expected that the WCAG walk-through would get low scores, we are disappointed that the method has such a significant difference from the others.

We conducted a two tailed test with an expected mean of 4.0, and all the results are statistically significant ($p < 0.0001$). This means that we can be sure that the results are generic and representative.

Brajnik [44] pointed out that evaluation methods should be valid, reliable, useful, and efficient. Vigo et al. [45] discussed how to select a tool based on its strengths and weaknesses with respect to coverage, completeness, and correctness. However, we strongly believe that a testing method must be user-friendly so that the practitioners will be motivated to use it. This is also supported by results showing that developers will not use a method if they do not like it [46].

Several participants noted that the methods overlap and complement each other. We find this to be very interesting because we had hoped that the participants would experience different challenges for different impairments. This is also supported by other studies' results showing that each testing method works best when included at a particular stage of software development Bai et al. [15].

Figure 10 shows how the overall scores for all the methods are distributed based on the participants' roles. We only show the results for developers and testers because the other roles have too few evaluations for some of the methods. However, Figure 10 clearly shows that there are differences between the roles. In general, the developers have more positive evaluations of the methods than the testers, particularly for the WCAG walk-through method. The developers also have more positive views than the testers and the total population for three methods: WCAG walk-through, personas, and SiteImprove.

We are not surprised that the developers have positive views of the more technical methods such as SiteImprove because they use tools such as these on a daily basis. However, we find it very interesting that developers regard

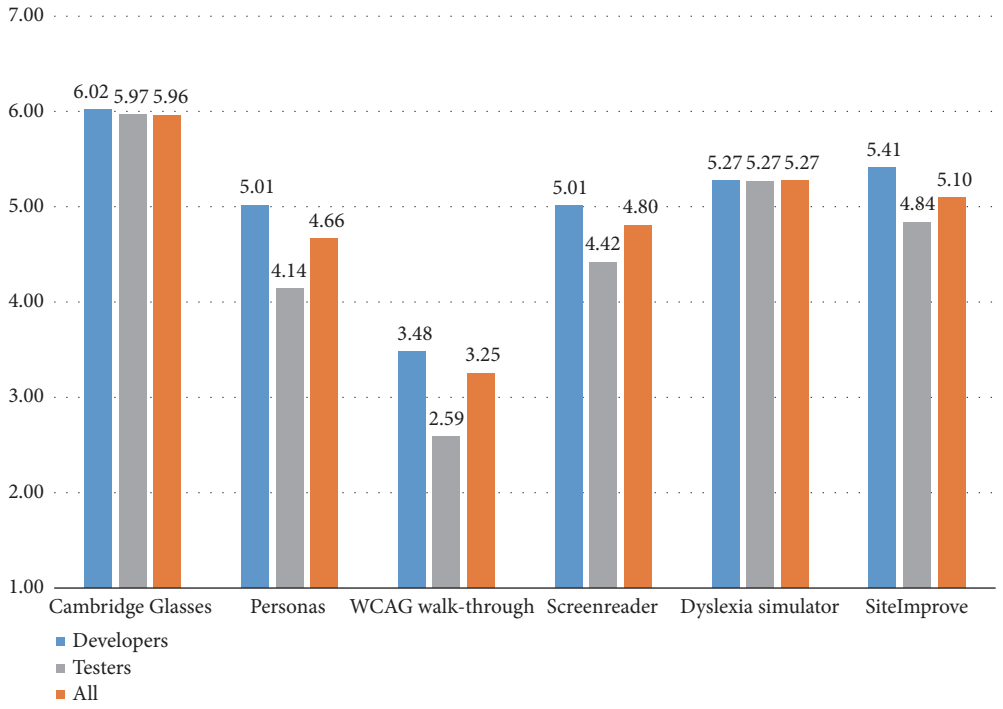

Figure 10: Developers vs Testers evaluation results.

personas much more highly than either the testers or the full population. We did not expect that the testers would rate the WCAG walk-through the lowest of all roles, as this is the only group that uses that method on a regular basis. Perhaps the testers find this method to be tiresome and see the other methods as refreshing. Alternatively, the WCAG walk-through could be difficult to use, even for experts [47].

7. Limitation

As with all empirical studies, there are some limitations that we need to consider. First there is the possibility of bias in the data collection and analysis. Therefore, the general criticisms such as uniqueness, may apply to our study. However, we triangulated from multiple sources of evidence such as observations, interviews, and questionnaires to increase the quality of the study. We also had participants with various roles from seven different teams in six different companies. Although we tested only six accessibility testing methods, these methods are generally representative of the various testing types, and they cover various aspects and impairments Bai et al. [25].

Second, the participants filled out evaluation forms. When using self-reported data to measure usability of tools, one might have the "social desirability bias" [48] where respondents respond more positively to rating questions because they unconsciously want to satisfy the facilitator giving them the questionnaire. To reduce this bias, following the advice of Albert and Tullis [49], we collected the responses anonymously on the web, and we did not look at the results until afterward. In general, we assume that the majority of the participants are more motivated than average individuals, as they chose to participate. That might have resulted in overall higher scores for the evaluations in our study. However, the background information indicates that there is notable variation among the participants when it comes to work experience, roles, and previous experience with accessibility testing.

Third, one may argue that 10 minutes is too little time to evaluate a method properly. However, our intention in this study was to have methods that are easy to understand and use, and therefore we wanted to limit the time frame for the evaluations. If the participants had more time, their evaluations of all the methods would probably be more positive, as they would have more experience with each. On the other hand, methods that are too complex to understand and conduct within our specified 10 minute time frame are out of scope for our evaluation. We suspect that the WCAG walk-through method achieved low evaluation scores because it requires more than 10 minutes to conduct. However, since it is the de facto method it must be included in the evaluation on similar terms as the other methods.

We probably influenced the results of the screen reader by providing some introductions in the email that we sent in advance of the study; we recommended using the speech viewer plugin, which likely helped the novice users. We suspect that if we had not given some tips in advanced, then the method would have received similar scores as the WCAG walk-through method.

Lastly, the participants tested several tools and methods in the same test session, so the scores for a given method may have been colored by the scores for the previously tested methods. For instance, if a participant was very happy with a previous method, the score for the next method

might be lower than if the order had been reversed (as the participant in that case would not have had another method for comparison). To reduce this limitation, we should have randomized the order in which the participants evaluated the different methods. We did not randomize the order, but some reordering of the methods was done for around half the evaluations, and this should reduce the impact.

We had a good number of participant and they were also from several different project and companies. Even though all the results were statistically significant ($p < 0.0001$), it would be nice to see even larger studies for each of the roles. In particular it would be interesting to have more data from manager, since we had quite few participants with that role as shown in Table 2. It would also be very interesting to have more studies for other testing methods.

8. Conclusion

Almost all the participants had a positive experience when evaluating the methods, and almost all said that they acquired more empathy for users and more awareness of the various challenges. Many of the participants also said that it was pleasant and valuable to utilize testing methods, except for the WCAG walk-through. Many participants found it difficult to test accessibility, and they found some methods to be more subjective and open to interpretation than others.

The project members stated that, when testing accessibility, their choice of method would depend on many factors, including preference, development phase, role, and context. However, the results of this study show that there are significant differences in how software members of different roles (e.g., developers vs. testers) regard testing methods, which implies that software teams should not choose a single method for all members. We also conclude that the other methods are better liked than the WCAG walk-through and that the software teams need to focus on finding methods that complement the WCAG walk-through. Several participants suggested that these tools should be part of their weekly routines and included in existing checklists.

Data Availability

The supporting data can be found in a report written in Norwegian: "Holdninger rundt universell utforming i smidige team" with ISBN 978-82-539-0546-4. However, the individual data points are not accessible because of privacy concerns.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This work has been supported by the ITS project funded by the Norwegian Directorate for Children, Youth and Family Affairs in the UnIKT program, grant number 13566.

References

[1] R. G. Bias and C.-M. Karat, "Justifying cost-justifying usability," in *Cost-Justifying Usability*, pp. 1–16, Elsevier, 2nd edition, 2005.

[2] C. Ardito, P. Buono, D. Caivano, M. F. Costabile, and R. Lanzilotti, "Investigating and promoting UX practice in industry: an experimental study," *International Journal of Human-Computer Studies*, vol. 72, no. 6, pp. 542–551, 2014.

[3] T. Halbach and K. S. Fuglerud, "On assessing the costs and benefits of universal design of ICT," *Studies in Health Technology and Informatics*, vol. 229, pp. 662–672, 2016.

[4] C. Putnam, M. Dahman, E. Rose, J. Cheng, and G. Bradford, "Best practices for teaching accessibility in university classrooms: cultivating awareness, understanding, and appreciation for diverse users," *ACM Transactions on Accessible Computing (TACCESS)*, vol. 8, no. 4, p. 13, 2016.

[5] M.-L. Sánchez-Gordón and L. Moreno, "Toward an integration of web accessibility into testing processes," *Procedia Computer Science*, vol. 27, pp. 281–291, 2014.

[6] W3C, "Web Content Accessibility Guidelines," https://www.w3.org/TR/WCAG20/.

[7] V. Stray, A. Bai, N. Sverdrup, and H. Mork, "Empowering agile project members with accessibility testing tools: a case study," in *Agile Processes in Software Engineering and Extreme Programming*, P. Kruchten, S. Fraser, and F. Coallier, Eds., vol. 355 of *Lecture Notes in Business Information Processing*, pp. 86–101, Springer International Publishing, Cham, Germany, 2019.

[8] R. Mace, *What Is Universal Design?* The Center for Universal Design at North Carolina State University, 1997.

[9] K. S. Fuglerud, *Inclusive design of ICT: The challenge of diversity [[Dissertation for the Degree of PhD]]*, University of Oslo, Faculty of Humanitites, 2014.

[10] L. Goldberg, *Teaching Accessibility: A Call to Action from the Tech Industry*, 2015.

[11] C. Law, J. Jacko, and P. Edwards, "Programmer-focused website accessibility evaluations," in *Proceedings of the ASSETS 2005 - The Seventh International ACM SIGACCESS Conference on Computers and Accessibility*, pp. 20–27, ACM, USA, October 2005.

[12] A. P. Freire, C. M. Russo, and R. P. M. Fortes, "The perception of accessibility in Web development by academy, industry and government: A survey of the Brazilian scenario," *New Review of Hypermedia and Multimedia*, vol. 14, no. 2, pp. 149–175, 2008.

[13] S. Trewin, B. Cragun, C. Swart, J. Brezin, and J. Richards, "Accessibility challenges and tool features: An IBM Web developer perspective," in *Proceedings of the International Cross Disciplinary Conference on Web Accessibility, W4A 2010*, p. 32, ACM, USA, 2010.

[14] D. Kane, "Finding a place for discount usability engineering in agile development: Throwing down the gauntlet," in *Proceedings of the Agile Development Conference, ADC 2003*, pp. 40–46, USA, June 2003.

[15] A. Bai, H. C. Mork, and V. Stray, "A cost-benefit analysis of accessibility testing in agile software development results from a multiple case study," *International Journal on Advances in Software*, vol. 10, no. 1 & 2, 2017.

[16] S. Kieffer, A. Ghouti, and B. Macq, *The Agile UX Development Lifecycle: Combining Formative Usability and Agile Methods*, 2017.

[17] J. Ferreira, J. Noble, and R. Biddle, "Agile development iterations and UI design," in *Proceedings of the Agile Conference (AGILE)*, pp. 50–58, IEEE, USA, 2007.

[18] G. Brajnik, "Comparing accessibility evaluation tools: a method for tool effectiveness," *Universal Access in the Information Society*, vol. 3, no. 3-4, pp. 252–263, 2004.

[19] G. Brajnik, "Beyond conformance: the role of accessibility evaluation methods," in *Proceedings of the International Conference on Web Information Systems Engineering*, pp. 63–80, Springer, 2008.

[20] T. H. Røssvoll and K. S. Fuglerud, "Best practice for efficient development of inclusive ICT," in *Universal Access in Human-Computer Interaction. Design Methods, Tools, and Interaction Techniques for eInclusion*, C. Stephanidis and M. Antona, Eds., vol. 8009 of *Lecture Notes in Computer Science*, pp. 97–106, Springer, Berlin, Germany, 2013.

[21] A. Bai, H. C. Mork, T. Schulzand, and K. S. Fuglerud, "Evaluation of accessibility testing methods. which methods uncover what type of problems?" *Studies in Health Technology and Informatics*, vol. 229, pp. 506–516, 2016.

[22] F. Paz and J. A. Pow-Sang, "A systematic mapping review of usability evaluation methods for software development process," *International journal of Software Engineering & Applications*, vol. 10, no. 1, pp. 165–178, 2016.

[23] W3C, "Web Accessibility Evaluation Tools List," https://www.w3.org/WAI/ER/tools/.

[24] J. Borg, A. Lantz, and J. Gulliksen, "Accessibility to electronic communication for people with cognitive disabilities: a systematic search and review of empirical evidence," *Universal Access in the Information Society*, vol. 14, no. 4, pp. 547–562, 2015.

[25] A. Bai, K. Fuglerud, R. Skjerve, and T. Halbach, "Categorization and comparison of accessibility testing methods for software development," *Studies in Health Technology and Informatics*, vol. 256, pp. 821–831, 2018.

[26] T. Halbach and K. S. Fuglerud, "On assessing the costs and benefits of universal design of ICT," in *Universal Design 2016: Learning from the Past, Designing for the Future*, H. Petrie, J. Darzentas, T. Walsh et al., Eds., York (GB). IOS Press, 2016.

[27] WebAim, *Wave Extension*, 2018, https://wave.webaim.org/extension/.

[28] SiteImprove, *SiteImprove Chrome Plugin*, 2018, https://chrome.google.com/webstore/detail/siteimprove-accessibility/efcfolpjihicnikpmhnmphjhhpiclljc?hl.

[29] T. Halbach and W. Lyszkiewicz, "Accessibility checkers for the web: How reliable are they, actually?" in *Proceedings of the 14th International Conference WWW/Internet*, vol. 2015, pp. 3–10, Ireland, 2015.

[30] University of Cambridge, *Cambridge Simulation Glasses*, 2018, http://www.inclusivedesigntoolkit.com/csg/csg.html.

[31] NV Access, *NVDA*, 2018, https://www.nvaccess.org/.

[32] Apple, *VoiceOver*, 2018, https://www.apple.com/lae/accessibility/mac/vision/.

[33] jontonsoup4, *Dyslexia Simulation*, 2018, https://chrome.google.com/webstore/detail/dyslexia-simulation/cnobhbaaijmbcbfdiakhllckiemjigac?hl=en.

[34] J. Pruitt and J. Grudin, "Personas: practice and theory," in *Proceedings of the 2003 Conference on Designing for User Experiences, DUX '03*, pp. 1–15, ACM, USA, 2003.

[35] IKT for alle, *Persona Examples*, 2018.

[36] Google, *Google Forms*, 2012, https://docs.google.com/forms/.

[37] A. M. Lund, "Measuring usability with the use questionnaire12," *Usability Interface*, vol. 8, no. 2, pp. 3–6, 2001.

[38] J. Brooke, "SUS-A quick and dirty usability scale," *Usability Evaluation in Industry*, vol. 189, no. 194, pp. 4–7, 1996.

[39] T. Tullis and B. Albert, *Measuring The User Experience. Collecting, Analyzing, and Presenting Usability Metrics*, 2008.

[40] S. L. Henry, "Understanding web accessibility," in *Constructing Accessible Web Sites*, pp. 6–31, Springer, 2002.

[41] S. Kawas, L. Vonessen, and A. J. Ko, *Teaching Accessibility: A Design Exploration of Faculty Professional Development at Scale*, 2019.

[42] A. Bai, H. Mork, and V. Stray, "How agile teams regard and practice universal design during software development," *Studies in Health Technology and Informatics*, vol. 256, pp. 171–184, 2018.

[43] B. Haskins, J. Stecklein, B. Dick, G. Moroney, R. Lovell, and J. Dabney, "Error cost escalation through the project life cycle," *Incose -Annual Conference Symposium Proceedings- Cd Rom Edition*, 2004.

[44] G. Brajnik, "Web accessibility testing: when the method is the culprit," in *Proceedings of the International Conference on Computers for Handicapped Persons*, vol. 4061, pp. 156–163, Springer, 2006.

[45] M. Vigo, J. Brown, and V. Conway, "Benchmarking web accessibility evaluation tools," in *Proceedings of the 10th International Cross-Disciplinary Conference onWeb Accessibility, W4A 2013*, Brazil, 2013.

[46] B. Johnson, Y. Song, E. Murphy-Hill, and R. Bowdidge, "Why don't software developers use static analysis tools to find bugs?" in *Proceedings of the 2013 35th International Conference on Software Engineering, ICSE 2013*, pp. 672–681, IEEE Press, USA, May 2013.

[47] G. Brajnik, Y. Yesilada, and S. Harper, "Testability and validity of wcag 2.0: the expertise effect," in *Proceedings of the 12th International ACM SIGACCESS Conference on Computers and Accessibility, ASSETS'10*, pp. 43–50, ACM, USA, 2010.

[48] C. Nancarrow and I. Brace, "Saying the right thing: coping with social desirability bias in marketing research," *Bristol Business School Teaching and Research Review*, vol. 3, no. 11, pp. 1–11, 2000.

[49] W. Albert and T. Tullis, *Measuring the User Experience: Collecting, Analyzing, and Presenting Usability Metrics*, Newnes, 2013.