Master's thesis

Master's Programme in Computer Science

# Capabilities of Large Language Models in Web Accessibility Evaluation: A Design Science Approach

Sebastian Sergelius

October 4, 2024

Faculty of Science

University of Helsinki

**Contact information**

P. O. Box 68 (Pietari Kalmin katu 5)
00014 University of Helsinki,Finland

Email address: info@cs.helsinki.fi
URL: http://www.cs.helsinki.fi/

HELSINGIN YLIOPISTO – HELSINGFORS UNIVERSITET – UNIVERSITY OF HELSINKI

| Tiedekunta — Fakultet — Faculty | Koulutusohjelma — Utbildningsprogram — Study programme |
|---|---|
| Faculty of Science | Master's Programme in Computer Science |

Tekijä — Författare — Author

Sebastian Sergelius

Työn nimi — Arbetets titel — Title

Capabilities of Large Language Models in Web Accessibility Evaluation: A Design Science Approach

Ohjaajat — Handledare — Supervisors

Prof. Tomi Männistö

| Työn laji — Arbetets art — Level | Aika — Datum — Month and year | Sivumäärä — Sidoantal — Number of pages |
|---|---|---|
| Master's thesis | October 4, 2024 | 45 pages, 5 appendix pages |

Tiivistelmä — Referat — Abstract

Web accessibility is an issue on many websites, hindering people with disabilities to perceive information on the web. Accessibility evaluation based on existing accessibility guidelines helps to find the barriers that affect these users. A comprehensive accessibility conformance review requires manual labor. However, Accessibility Evaluation Tools (AET) help find the most common accessibility barriers. This thesis explores the potential of Large Language Models (LLM) in evaluating the Web Content Accessibility Guideline (WCAG) success criterion 2.4.2, Page Titled, using pre-made HTML test cases provided by the WCAG Accessibility Conformance Testing rules. Utilizing a Design Science Research method, the input prompt is iterated to evaluate whether the LLM can accurately assess if the HTML code contains a title, whether the title describes the page content, and whether the title identifies the page. The findings reveal that the LLM can perform these evaluations, suggesting that it can be used as an assistant in conformance reviews, potentially speeding up the conformance review process and reducing the need to understand the website's subject matter. However, the study acknowledges the simplicity of the test cases and the non-deterministic nature of LLMs. Future research on LLMs evaluating web accessibility should address language diversity. In addition, implementing LLMs into AETs would provide insight into how LLMs affect the efficiency of the accessibility evaluation process and the capabilities of LLMs on more complex websites.

**ACM Computing Classification System (CCS)**
Human-centered computing → Accessibility → Accessibility systems and tools
Human-centered computing → Accessibility → Accessibility technologies

Avainsanat — Nyckelord — Keywords

web accessibility, accessibility evaluation

Säilytyspaikka — Förvaringsställe — Where deposited

Helsinki University Library

Muita tietoja — övriga uppgifter — Additional information

Software study track

# Contents

# 1 Introduction

Within the past decade, web accessibility legislation in the EU has enforced the public sector websites to comply with web accessibility guidelines (Directive 2016/2102, 2016). Additionally, the European Accessibility Act will force some private sector websites to comply after the second half of 2025 (Directive 2019/882, 2019). The legislation aims to ensure equal access to digital information and services for people with disabilities. However, more than 95% of the top million websites have accessibility issues that are detectable with Accessibility Evaluation Tools (AETs) (WebAIM, 2024).

There are multiple AETs to help assess accessibility compliance (Duran et al., 2017). However, these tools cover reliably around 20–30% of the 86 success criteria (Duran et al., 2018; Deque Systems, 2021; WebAIM, 2024). Therefore, human evaluation and expertise in accessibility are necessary to uncover barriers that affect people with disabilities (Lange et al., 2024). An accessibility evaluation tool can detect if a website has a title, but a human has to evaluate if the title describes the content of the page.

This thesis explores web accessibility evaluation and the potential of Large Language Models (LLMs) in web accessibility evaluation, emphasizing 2.4.2 Page Titled Success Criterion that requires human evaluation. A design science research approach is taken to build an input prompt for an LLM to help evaluate if the title describes the content. The thesis is structured as follows.

Chapter 2 introduction to web accessibility, covering definitions, common disabilities affecting web use, implementation, guidelines, and legislation in the EU. Chapter 3 focuses on accessibility evaluation methodologies, accessibility evaluation tools, manual evaluation, and the potential role of AI within accessibility. Evaluation methodologies are presented, following the capabilities of AETs, detailing which criteria they cover, how the effectiveness and reliability of AETs have been studied, and what has been done to standardize the AET's development.

Chapter 4 details the selected research methods, research questions, problem identification, selected tools and scope, and the two iteration phases of the input prompt created in the study, along with the evaluation metrics used. The results of each iteration are presented in Chapter 5, highlighting the accuracy, consistency, and perceived usefulness of the LLM's output.

Chapter 6 summarizes the main findings, discusses the implications, and addresses study limitations. The thesis concludes by emphasizing the importance of web accessibility evaluation through AI-assisted approaches.

# 2 Web accessibility

Web accessibility aims to ensure an equitable online experience for users of diverse abilities. According to the World Health Organization, an estimated of over 1.3 billion people worldwide face some form of significant disability (WHO, 2023b). Persons with low vision, hard of hearing, or learning disability might have difficulties perceiving information on the web. The significance of web accessibility is increasingly apparent. This chapter delves into web accessibility foundations.

Section 2.1 defines web accessibility. In section 2.2 we describe commonly known disabilities affecting the use of web services. Section 2.3 briefly describes how to implement web accessibility and what are web standards used to overcome barriers. Section 2.4 delves into the international standard for web accessibility called Web Content Accessibility Guidelines (WCAG). The last section of this chapter is the legal requirements for web accessibility in the European Union.

## 2.1 Definition

Web accessibility is defined as an equal opportunity to perceive, operate, understand, and interact with an online service (Regional State Administrative Agency, 2023). Regardless of disability a person faces, the same information and functionality should be made available. Web accessibility considers the technical and content implementation. Technical implementation of web accessibility ensures that barriers to assistive technologies, such as screen readers or speech input software, are removed. Structuring content with headings, using short paragraphs and simple understandable language makes the content accessible. Additionally, content accessibility is providing content in multiple forms, such as videos, sound, images, and text.

## 2.2 Disabilities affecting the use of web

Spanning from visual impairments to situational limitations, each condition presents barriers that hinder the ability to interact and comprehend information on the web. Under-

standing these disabilities is crucial when designing online services that prioritize inclusion and ensure an equal experience for all users. Some disabilities that affect web use are visual, physical, auditory, and cognitive impairments (Abou-Zahra, 2017).

Among the most common disabilities affecting web accessibility are vision impairment and blindness, with more than 2.2 billion people impacted worldwide (Yeliz and Simon, 2019; WHO, 2023a, Chapter 1). Individuals with visual impairments depend on functionality for formatting the content to their needs and on content accessibility. Barriers encountered by individuals with visual impairment are a lack of text alternatives for images, videos, and controls, insufficient contrast between text color and background color, lack of keyboard navigation support, and missing descriptions for links and inputs.

Accessibility is not only targeted at individuals with impairments. Individuals who have lost their glasses, have a broken arm, or have a hard time reading the screen when the sun glares also benefit from web accessibility (Abou-Zahra, 2017). In addition to temporal disabilities and situational limitations, also aging people benefit. Most of the people with low vision are above the age of 50 (WHO, 2023a).

Since 2019, WebAIM (2024) has conducted annual research analyzing the accessibility of the top million websites. In their research, they used their own WAVE automated accessibility evaluation tool to check for accessibility barriers on the landing page. Results show that 95.9% of landing pages have accessibility barriers. In addition, the amount of elements and ARIA tags on websites has grown each year indicating that websites are more complex than before. Moreover, the five most found accessibility barriers in the past 5 years all affect people with low vision or blindness using assistive technologies and keyboard only (WebAIM, 2024).

## 2.3 Implementation

Web accessibility strives for an equal opportunity for persons with any sort of impairment to interact with web services on any device with the help of assistive technologies. In web accessibility, the goal is to implement a website in a specific way so that assistive technologies can help remove the barrier to accessing the information provided. Commonly known assistive technologies are screen readers, screen magnifiers, and speech input software that are incorporated into major operating systems.

Web designers, developers, and content creators implement web accessibility. Accessibility

is built upon these three topics: technical implementation, ease of use, and comprehensibility of content (Regional State Administrative Agency, 2023). The technical implementation provides functionality for assistive technologies in the web source code based on web standards. Ease of use involves having a user experience with a structured frame for navigation and content placement. Comprehensibility of content is the structure of the content itself, meaning the content is written in short paragraphs and plain language.

In web development, the technical implementation is mainly done by web developers by following the HTML and CSS standards. Accessible Rich Internet Application (ARIA) is a standard used within HTML for advanced web applications with dynamic content to help screen readers interpret the content. For example, removing the barrier for individuals with low vision to perceive images requires an alternative text in the image tag in HTML. In addition, having a progress bar on a web application requires ARIA tags for screen readers to communicate the status of an action.

However, there are overlaps in responsibilities of implementing accessibility, as building and managing a website has become easier due to website builders and content management systems. In the EU, the implementation details for web accessibility have been standardized based on existing guidelines.

## 2.4 Guidelines

The World Wide Web Consortium creates accessibility guidelines through the Web Accessibility Initiative (Henry, 2023). The international standard is named Web Accessibility Content Guidelines (WCAG). Version 2 of WCAG was published in 2008. Version 2.1 came out in 2018 and is the version referenced in the European Standard. Version 2.2 was published, and made a recommendation by W3C in October 2023. The W3C recommends always using the latest version of WCAG, as it is backward compatible.

The guidelines aim to ensure that digital content is accessible to a broad audience and adaptable to various forms, accommodating diverse sensory, physical, and cognitive abilities of individuals (Campbell et al., 2023). The WCAG documents are written in a technology-agnostic way. It is based on four principles: *Perceivable, Operable, Understandable, and Robust*. In each principle, there are guidelines, and under each guideline are success criteria. There are a total of 13 guidelines and 86 success criteria. In total, there are three levels of accessibility; A being the lowest, AA, and AAA being the highest, see Table 2.1.

**Table 2.1:** WCAG 2.2 principles, guidelines and success criteria divided with the three accessibility levels. Adapted from Campbell et al. (2023)

| **Principle** | **Guideline** | **Success Criterion Level A** | **Success Criterion Level AA** | **Success Criterion Level AAA** |
|---|---|---|---|---|
| **Perceivable** | 1.1 Text alternatives | 1.1.1 Non-text content | | |
| | 1.2 Time-based media (Prerecorded, unless stated otherwise) | 1.2.1 Audio-only and Video-only<br>1.2.2 Captions<br>1.2.3 Audio description or media alternative | 1.2.4 Captions (Live)<br>1.2.5 Audio description | 1.2.6 Sign language<br>1.2.7 Extended audio description<br>1.2.8 Media alternative<br>1.2.9 Audio-only (Live) |
| | 1.3 Adaptable | 1.3.1 Info and relationships<br>1.3.2 Meaningful sequence<br>1.3.3 Sensory characteristics | 1.3.4 Orientation<br>1.3.5 Identify input purpose | 1.3.6 Identify purpose |
| | 1.4 Distinguishable | 1.4.1 Use of color<br>1.4.2 Audio control | 1.4.3 Contrast (Minimum)<br>1.4.4 Resize text<br>1.4.5 Images of text<br>1.4.10 Reflow<br>1.4.11 Non-text contrast<br>1.4.12 Text spacing<br>1.4.13 Content on hover or focus | 1.4.6 Contrast (Enhanced)<br>1.4.7 Low or no background audio<br>1.4.8 Visual presentation<br>1.4.9 Images of text (No exception) |
| **Operable** | 2.1 Keyboard accessible | 2.1.1 Keyboard<br>2.1.2 No keyboard trap<br>2.1.4 Character key shortcuts | | 2.1.3 Keyboard (No exception) |
| | 2.2 Enough time | 2.2.1 Timing adjustable<br>2.2.2 Pause, stop, hide | | 2.2.3 No timing<br>2.2.4 Interruptions<br>2.2.5 Re-authenticating<br>2.2.6 Timeouts |
| | 2.3 Seizures and physical reactions | 2.3.1 Three flashes or below threshold | | 2.3.2 Three flashes<br>2.3.3 Animation from interactions |
| | 2.4 Navigable | 2.4.1 Bypass block<br>2.4.2 Page titled<br>2.4.3 Focus order<br>2.4.4 Link purpose (In context) | 2.4.5 Multiple ways<br>2.4.6 Headings and labels<br>2.4.7 Focus visible<br>2.4.11 Focus not obscured [New] | 2.4.8 Location<br>2.4.9 Link purpose (Link only)<br>2.4.10 Section headings<br>2.4.12 Focus not obscured (Enhanced) [New]<br>2.4.13 Focus appearance [New] |
| | 2.5 Input modalities | 2.5.1 Pointer gestures<br>2.5.2 Pointer cancellation<br>2.5.3 Label in name<br>2.5.4 Motion actuation | 2.5.7 Dragging movements [New]<br>2.5.8 Target size (Minimum) [New] | 2.5.5 Target size (Enhanced)<br>2.5.6 Concurrent input mechanisms |
| **Understandable** | 3.1 Readable | 3.1.1 Language of page | 3.1.2 Language of parts | 3.1.3 Unusual words<br>3.1.4 Abbrevations<br>3.1.5 Reading level<br>3.1.6 Pronunciation |
| | 3.2 Predictable | 3.2.1 On focus<br>3.2.2 On input<br>3.2.6 Consistent help [New] | 3.2.3 Consistent navigation<br>3.2.4 Consistent identification | 3.2.5 Change on request |
| | 3.3 Input Assistance | 3.3.1 Error identification<br>3.3.2 Labels or instructions<br>3.3.7 Redundant entry [New] | 3.3.3 Error suggestions<br>3.3.4 Error prevention (Legal, Financial, Data)<br>3.3.8 Accessible authentication (Minimum) [New] | 3.3.5 Help<br>3.3.6 Error prevention (All)<br>3.3.9 Accessible authentication (Enhanced) [New] |
| **Robust** | 4.1 Compatible | 4.1.1 Parsing (Obsolete and removed)<br>4.1.2 Name, role, value | 4.1.3 Status messages | |

Each success criterion is part of one level of accessibility. Version 2.2 removes one and adds nine new success criteria. Success criterion 4.1.1 is made obsolete, as assistive technologies do not need to directly parse the HTML code anymore, and problems that occur by this criterion are addressed by other criteria. The new nine added success criteria are: 2.4.11, 2.4.12, 2.4.13, 2.5.7, 2.5.8, 3.2.6, 3.3.7, 3.3.8 and 3.3.9 as shown in Table 2.1.

Perceivable means that information and elements can be perceived by all users (Campbell et al., 2023). Non-textual content should have a textual representation, such as alternative text for images or textual description for prerecorded audio and video (guidelines 1.1 and 1.2). Adaptable describes that content is structured so that computer programs can convey the information to users regardless of device screen size or orientation. Distinguishable main purpose is for users to be able to distinguish between foreground and background information.

The operable principle ensures that the page can be used with different input peripherals, such as a keyboard, mouse, or touch screen (Campbell et al., 2023). Keyboard accessible guideline 2.1 defines that the same functionality should be available by only using the keyboard. Guideline 2.2 (Enough time) requires time-based content, such as notifications, to be available and controllable by the user. Seizure and physical reactions (2.3) ask designers not to create flashing content and limit the flashing of a page to three. Navigable (2.4) provides users with information where they are on the page and quick access to relevant content. A known accessibility feature for navigating is the "Skip to content" link that is provided on pages for screen readers to quickly jump to the content block. The last guideline 2.5 is input modalities that guide developers to ensure functionality beyond keyboard input.

The understandable principle is about the accessibility of the content. Readable (3.1) requires web developers to set the language for the page. Predictable expects that there are no sudden changes on the page, such as a "Request for help" button would change position. Input assistance guideline ensures that input elements have labels, that the same information would not be asked twice from the user in the same session, and that input requirements and errors are displayed correctly.

Robust requires the page to be created in such a way that it can be interpreted by assistive technologies now and in the future (Campbell et al., 2023). Compatible is the sole guideline defining that developers should apply roles and names for elements and that status messages can be read by assistive technologies without focusing on the element.

The success criteria under the guidelines are written as testable statements (Campbell

et al., 2023). WCAG also offers guidance and best practices for each success criterion by providing informative techniques for web content developers. The techniques are categorized as sufficient and advisory, where advisory techniques go beyond the requirements for the specific criterion. Common failures for conformance are also documented. The techniques describing sufficient conformance are more technical and contain examples of how to implement a success criterion with HTML and CSS techniques.

There are in total 55 success criteria to cover for conformance with the AA level of web accessibility in WCAG 2.2. Eggert (2023) provides a comprehensive checklist of tips and techniques to achieve conformance with AA level of accessibility. The AA level of web accessibility is what legislation requires from the public sector and in the future for some of the private sector. In comparison to WCAG 2.1, six out of the nine new success criteria added in WCAG 2.2, are new requirements to check for to achieve the AA level of conformance.

## 2.5 Legislation

The emergence of directives and legislative measures plays a role in the goal of developing best practices for web accessibility. The European Union (EU) Directive 2016/2102 was adopted in 2016 to ratify part of the United Nations Convention on the Rights of Persons with Disabilities (UN CRPD). The Directive enforces EU Member States to ensure that their public sector organizations implement and monitor accessibility on their websites and mobile applications (Directive 2016/2102, 2016).

Directive 2016/2102 (2016) gave a clear timeline for the member states. At the latest of September 2018 each member state should have transposed this directive into national legislation. Since September 2019 all new websites must conform. All public sector websites had to comply after September 2020. In June 2021 mobile applications had to conform and in December 2021 the monitoring and reporting started.

The EU Directive adopted in 2016 refers to the European Standard EN 301 549 which specifies the accessibility requirements for ICT products and services. In the first version of the Standard, the guidelines and requirements were based on Web Content Accessibility Guidelines provided by the World Wide Web Consortium (Abou-Zahra, 2018). During the same time as member states had to transpose this into a national law in 2018, the standard was changed to be a direct reference to the WCAG making the WCAG an accessibility standard that EU member states should follow. The EU Commission is in the process

of harmonizing the EN 301 549 Standard with the latest WCAG version with a planned release of 2025 (ETSI, 2024).

Monitoring of accessibility in the public sector should be done every three years (The European Commission, 2018). The first monitoring and accessibility report from member states to the EU Commission was given in 2021. The next reports should be delivered in December 2024. This monitoring is divided into two review categories, in-depth and simplified. Simplified review requires accessibility testing using only automated accessibility evaluation tools, whereas in-depth review requires both tools and manual review by accessibility specialists.

Based on the public sector directive, the EU Commission has created a similar directive affecting some private sector bodies. The European Accessibility Act (EAA) was adopted by the EU in 2019 to complement the 2016 directive and UN CRPD (Directive 2019/882, 2019). The EAA will require medium and large-sized private sector companies in the fields of banking, travel, and e-commerce, to start following the accessibility guidelines after the end of June 2025.

# 3 Web accessibility evaluation

As web services undergo rapid development and continuous deployment, a continuous accessibility evaluation is crucial, mandated by both legislative organizations and end-users. The primary goal of assessing web accessibility is to promote digital inclusion by identifying and eliminating barriers, thereby expanding access to a wider audience. Multiple accessibility evaluation tools are available to help web content developers in ensuring inclusivity and conformance to accessibility standards.

## 3.1 Methodologies

Web accessibility evaluation is a process of assessing and determining to which extent a website is accessible to people with disabilities (Yeliz and Simon, 2019, Chapter 26.2). The evaluation of web accessibility involves analyzing each web page on the site against established standards and guidelines. The process of web accessibility evaluation can be categorized into three categories: automated testing, manual inspection, and user testing.

Automated testing is done by accessibility evaluation tools that are programmed to automatically parse and evaluate the source code of a web page based on guidelines (Yeliz and Simon, 2019, Chapter 26.2). Adopting automated testing early ensures that potential accessibility issues are identified and addressed with immediate feedback during the development cycle. However, automated testing tools can only cover those guidelines that are assessable through machine-based analysis.

Manual inspection is required to ensure that a web page conforms to accessibility standards (Yeliz and Simon, 2019, Chapter 26.2). A conformance review is the most frequently used methodology that involves evaluators checking if the web page meets criteria based on a checklist. Brajnik (2008) introduced and compared his Barrier Walkthrough methodology where evaluation is done in a more systematic way taking into consideration the context of the scenario and goal. An example of the scenario and goal is a person with vision impairment, using a specific assistive technology trying to fill in a form. Results show that the Barrier Walkthrough method is better at finding critical problems more accurately, but fails in finding all the accessibility problems (Brajnik, 2008). Nevertheless, a manual inspection conducted by different evaluators may result in different outcomes regardless

of the methodology used (Brajnik et al., 2009; Brajnik et al., 2010).

User testing is based on empirical usability testing. User testing is the most reliable accessibility evaluation method involving actual users with disabilities performing tasks on a web page (Yeliz and Simon, 2019, Chapter 26.2). Reliability is the consistency of the outcome regardless of whom performed the test. However, user testing is slow and costly. In addition, it is complex to set up the testing session and to take into account a diverse range of users (Brajnik, 2008)

Multiple evaluation methodologies have been proposed combining the three categories described above (Yeliz and Simon, 2019, Chapter 26.2.1). It is recommended to always use more than one evaluation tool, as the technical implementation of each tool might differ ending up in different results. Manual inspection and user testing should always be done to ensure conformance. However, there is no consensus on how to combine these methods for a comprehensive accessibility study (Yeliz and Simon, 2019, Chapter 26.2.1).

## 3.2 Automated tools

Automated Accessibility Evaluation Tools (AET) should be used to check if a web page conforms to accessibility guidelines. An accessibility evaluation tool parses the HTML and CSS code of a web page and checks that the web page conforms to the accessibility standards. These tools are helpful for quick accessibility evaluation. However, human evaluation is always required to check for full conformance as these tools might produce misleading results (Lange et al., 2024). Therefore, accessibility evaluation tools should be used as assistive evaluators when evaluating the conformance of a website.

There are dozens of tools to choose from, differing in functionality, coverage, and features (WAI, 2024). Studies show that the amount of errors found on a page varies largely between tools (Frazão and Duarte, 2020; Chadli et al., 2023; Rajh and Debevc, 2023). Where one tool can find a dozen errors, another tool can find thousands on the same page. Rajh and Debevc (2023) reviewed the monitoring reports from member states and did a comparative analysis on 10 free-to-use tools. Results show that almost all tools link found problems to WCAG success criteria. However, tools differed in the amount of criteria they checked and in the final representation of the coverage report. Furthermore, the transparency of tools on which success criteria they cover is not clear to users (Rajh and Debevc, 2023).

A method to compare and measure tool effectiveness was proposed by Brajnik (2004). The effectiveness of a tool can be measured with correctness, completeness, and specificity. Correctness measures how tools report non-existing problems (false positives). Completeness measures how tools fail to find problems that exist (false negatives). A tool can be considered highly specific if it can detect a wide range of distinct accessibility issues, offering detailed insights into each problem. Vigo et al. (2013) studied the coverage, completeness, and correctness of six popular tools showing that a higher amount of accessibility issues on a page gives a higher completeness score, and in contradiction, on highly accessible pages the completeness score drops. In addition, the study shows that tools with high completeness scores report more easily false positives, reducing the correctness (Vigo et al., 2013). Parvin et al. (2021) evaluated four tools on their specificity and transparency. Results show that three of the tools share what success criteria and techniques they cover. However, differences were reported on how tools display the result to the user.

To mitigate the uncertainty and instability of accessibility evaluation tools the W3C has created a task force working on Accessibility Conformance Testing (ACT) rules (Abou-Zahra and Henry, 2020). The purpose of ACT rules is to standardize the interpretation of WCAG documents by creating technology-specific test cases that accessibility tool developers and accessibility methodology developers can use to test their outcomes. For unified reporting, the W3C has developed a resource description framework named Evaluation and Report Language (EARL). EARL is used to report conformance to the ACT rules. In addition, it can be used to combine results from different tools. However, currently, there is no standard on how to report the output of an automatic evaluation tool, and current tools differ in the reporting style (Rajh and Debevc, 2023).

## 3.3 Tool coverage

Automated Accessibility Evaluation Tools (AET's) cover only around 20–30% of all success criteria in the WCAG (Duran et al., 2018; Deque Systems, 2021; WebAIM, 2024). In a study conducted by Duran et al. (2018) they created a page with 142 accessibility issues and analyzed this page with 12 different tools (Nu HTML Checker excluded, not an AET). The lowest score was 17 % by Google Accessibility Developer Tool and the highest scoring tool was SortSite with 40 %. In addition, the coverage of all tools combined managed to find only 100 out of the 142 accessibility barriers, which supports the need to use multiple tools together when evaluating accessibility (Duran, 2017). Frazão and Duarte (2020)

conducted similar research where they found out that evaluating a page with multiple tools raises the coverage by 10–40 %. However, Frazão and Duarte (2020) did not study how to combine and remove duplicates from the results between each tool. One large developer and vendor of accessibility testing, Deque Systems, believes that the apprehension on the coverage should be based on real findings rather than which success criteria are covered (Deque Systems, 2021).

Deque states that up to 57 % of accessibility issues can be found when considering that there are usually multiple violations for one success criterion on a page (Deque Systems, 2021). From over 2000 in-depth accessibility evaluations they showed that in most cases automation finds more issues than manual review. Data in this study is based on A and AA levels of accessibility when WCAG 2.1 was the recommendation by W3C. The obsolete success criterion 4.1.1 Parsing in WCAG 2.2 was one of the most detected by automation with a proportion of 90.28 % being found automatically. The six types of issues encountered with a significantly high proportion (percentage in brackets) found by automation are the following:

- 3.1.1 Language of page (91.81 %, 1 995 issues)
- 1.4.3 Contrast (Minimum) (83.11 %, 73 733 issues)
- 2.4.1 Bypass blocks (79 %, 2 001 issues)
- 1.1.1 Non-text context (67.57 %, 16 014 issues)
- 4.1.2 Name, role, value (54.42 %, 26 276 issues)
- 1.3.1 Info and relationship (45.17 %, 16 432 issues)

These six success criteria account for up to 52 % of all accessibility issues found by automation. By removing the obsolete success criterion 4.1.1 from the data the total amount of found issues by automation is 53 %. However, this study does not account for the six newly added success criteria in WCAG 2.2 Level AA. Moreover, Fiers (2023) writes in Deque's blog that the only identified success criterion to be testable in WCAG 2.2 according to their promise, that is not returning false positives, is the success criterion 2.5.8 Target size (Fiers, 2023). Deque's target is to ensure that the axe-core engine reports zero false positives to ensure that the results can be trusted by developers and accessibility evaluators (Deque Systems, 2021). Axe-core is an open-source accessibility testing engine

for web browsers. It is used by millions of Github projects and it also works as the core for Deque's tools and Google Lighthouse.

As WCAG version 2.2 is the new standard, according to the coverage report by Deque Systems (2021) there would be 17 out of 86 success criteria that automation can discover with certainty. An automated accessibility evaluator can determine that a page has a title or an image has an alternative text from the web page source code. However, these tools can not determine if a page title or alternative text for non-text content is descriptive (Eggert, 2023).

## 3.4 Manual evaluation

Automated accessibility evaluation tools are the first step in finding accessibility problems on a web page. To increase the coverage, manual inspection is required. Manual evaluation is a costly and time-consuming process as each web page on the whole site has to be evaluated separately. Most of the success criteria in the WCAG can only be determined properly by human evaluation, such as criteria related to context. For example, test automation can detect if the alternative (alt) attribute is set for image tags in the HTML code. However, it can not determine if the alternative text describes the image correctly to users (Frazão and Duarte, 2020). Additionally, expertise does matter when evaluating a page for conformance (Brajnik et al., 2010; Brajnik, 2008). In comparison to a novice, an expert in the field of web accessibility is more capable of finding accessibility barriers on a web page.

As there is no standard methodology for manual evaluation, each evaluator has a toolset that they use to evaluate individual pages manually. Using a semi-automated test tool is a great way to increase coverage of found barriers and to guide the evaluator in particular with wizard-based steps guiding the manual evaluation. Semi-automated test tools are used to evaluate a single page in a specific state for accessibility barriers. These tools run automated accessibility evaluations combined with, for example, user input wizards to cover more possible manually evaluated accessibility barriers.

Two large accessibility evaluation consultant companies, Deque and Siteimprove, have their own paid semi-automated accessibility evaluation tools. Deque Systems (2022) extended the same in-depth coverage study mentioned in Section 3.3 on their semi-automated accessibility testing tool. They discovered that the coverage of found accessibility barriers is increased by 23% when using their wizard-based semi-automated accessibility evaluation

tool (Deque Systems, 2022). The guided testing is incorporated into their test automation to help evaluators with a more systematic conformance evaluation. For example, when trying out Deque's tool free version subscription, the guided testing prompts to check if an alternative text for an image or page title is descriptive in a wizard to the evaluator. The page title wizard was the following question: "The page title is 'Understanding WCAG 2.2 | WAI | W3C'. Does it accurately describe the purpose of the page?". This question leaves the decision up to the evaluator to answer. To answer the question reliably, the evaluator needs to read the content of the page and form an understanding of the entire page.

## 3.5 Use of AI

With the emergence of multiple Generative AI tools for different contexts, such as music, image, or text generation, the use of AI and machine learning models in web accessibility has sparked discussion and gained focus in the field of accessibility. Companies within the accessibility evaluation industry, such as AudioEye and Deque, have incorporated AI features in their tools (Deque, 2024; Bureau of Internet Accessibility, 2023a). In addition, the current investment in AI has started a cautious discussion within the W3C WCAG Working Group where they are following the progress and possible use-cases of AI in web accessibility (Campbell, 2024).

One of the most commonly found issues is a missing descriptive alternative text for images (WebAIM, 2024; Deque Systems, 2021). Therefore, the potential of AI in generating alternative text for images is an interesting topic with multiple viewpoints (Campbell, 2024; Bureau of Internet Accessibility, 2023b; Gustafson, 2024). Mozilla is experimenting with a machine learning model that runs locally on the user's computer to generate missing image alternatives when viewing PDF files within the browser (Ziadé, 2024). Even though there are tools to generate automatic alternative descriptions to images, the descriptions are not always considered appropriate as they do not take into account the context around the image (Dolson, 2023; Bureau of Internet Accessibility, 2023b). Besides, within the accessibility community, some say that AI should work as an assistant that should not make automatic decisions regarding accessibility (Campbell, 2024; Dolson, 2023).

A case study conducted by Othman et al. (2023) showed that the use of ChatGPT in solving accessibility issues was able to provide correction to code and speed up the process of solving accessibility barriers. However, the study was conducted on two websites and

the LLM had problems solving subjective issues, such as the color contrast between the background color and text color. López and Varela (2024) conducted an experiment where they tested if an LLM would be able to evaluate three different WCAG success criteria that require manual evaluation. In total, they had 39 ACT cases and a LLM was able to successfully evaluate 34 of them. However, they had to modify the prompt manually for more complex test cases to specify where to look for the correct information in the HTML tags.

# 4 Methods

This chapter will describe the research goal, questions, and methods used. Additionally, the selected tools and scope of the research are specified. The process and iterations of the artifact are provided in this chapter which is used to collect data to be evaluated. Furthermore, the evaluation process is presented in this chapter.

## 4.1 Research goals

The objective of this research was to find answers to the research questions below. The research questions will help to understand the current situation of web accessibility and how web accessibility is evaluated. Based on the findings of the literature review we can identify if there is potential in Generative AI when assessing accessibility of a web page.

The research questions for the thesis are the following:

- **RQ1.** What are the limitations of web accessibility evaluation tools in assessing compliance with Web Content Accessibility Guidelines (WCAG)?
- **RQ2.** How does Generative AI assist in addressing these limitations?

The primary goal is to evaluate the capabilities of Generative AI in the scope of web accessibility evaluation, forming an input prompt, that is the demonstrated and empirically evaluated artifact in this study. Research Question 1 (RQ1) pinpoints relevant challenges within web accessibility evaluation. Subsequently, to answer Research Question 2 (RQ2) we create an artifact in the form of an input prompt. A prompt is a set of instructions given to the Large Language Model (LLM) to generate an output. The artifact is planned to be improved in iterations. For each iteration, the artifact is demonstrated in use and evaluated accordingly.

## 4.2 Research methods

This thesis will use a qualitative research method that is based on an iterative manner of the Design Science Research Methodology (DSRM) (Hevner et al., 2004; Peffers et al.,

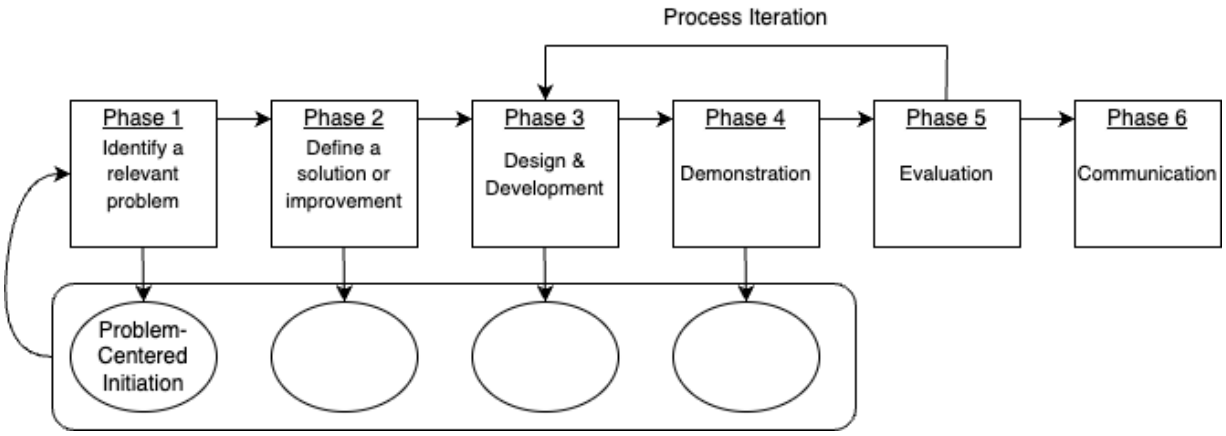


**Figure 4.1:** DSRM Iterative Process, modelled from Peffers et al. (2007)

2007). DSRM aims to identify a problem and create an artifact to increase productiveness. Peffers et al. (2007) presented the DSRM that consists of six phases as shown in Figure 4.1.

Phase 1 and Phase 2, the problem identification and objective of the solution, are conducted as part of the RQ1. RQ2 is answered in the Process Iteration cycle, which consists of the design & development, demonstration, and evaluation phases. Phase 6 is presented in the Result and Discussion chapters.

The process entry can start from any of the first four phases (Peffers et al., 2007). In this study, the initiation of the research is problem-centered. The problem identification and motivation started by asking; Why are so many websites inaccessible even though legislation is moving forward? What needs to be done to make the web more accessible?

With these questions in mind and very limited knowledge of web accessibility, a multivocal literature review was conducted in Chapter 2 from a legislative perspective in the EU. The review was done by delving into the Directive 2016/2102 (2016) and documents it referred to. In addition, Google was used to search for information on web accessibility.

## 4.3 Problem identification

RQ1 is used to identify a problem and define the scope of the artifact. A multivocal literature review was conducted to figure out the current state of web accessibility evaluation methodologies and tools in Chapter 3. The ACM Digital Library is chosen as the main source as they are reputable source in the field of computer science. In addition, they

have a journal on Human-Computer Interaction that contains articles on web accessibility. Taking into account the groundwork in Chapter 2 from the legislative perspective, to answer RQ1 the following simple search string was used in the ACM Digital Library:

```
"accessibility evaluation"
```

Filters applied were: Research Article and Publication Date between 2019–2024 to find currently relevant publications within the field.

A total of 164 research articles were found in the ACM Digital Library with these search filters. Found papers were selected based on their title and abstracts. An inclusion criterion was used based on either the title or abstract mentioned comparing tools, the EU Directive, or the WCAG guidelines. Exclusion criteria were used if the paper title or abstract had mentioned one of the following: specific disability, specific technology, mobile accessibility, or the study was conducted in a country or targeted towards an entity. A snowballing method was used to identify the commonly referred papers that are also used in this thesis as references. Additionally, researchers searched individually by name to find relevant literature and Google was used to find government reports, books, and other grey literature regarding web accessibility.

## 4.4 Selected tools and scope

The Generative AI selected for this thesis is a Large Language Model (LLM) ChatGPT 3.5 by OpenAI. Selection criteria for ChatGPT are based on the popularity of the tool in research and the tool is free to use (Ouyang et al., 2023; White et al., 2023). In this thesis, ChatGPT will be used to test the 2.4.2 Page Titled success criterion. The Page Titled Success Criterion was chosen for this thesis as the success criterion requires manual evaluation from an accessibility evaluator to achieve sufficient conformance based on the context of the web page. Additionally, the success criterion has ACT rules provided by the W3C that can be used as test cases and it is categorized in the lowest A-level of accessibility.

ChatGPT 3.5 will be prompted through their website user interface, chat.openai.com, which has been trained with data available in early 2022 (OpenAI, 2024). Each prompt will be opened as a new chat to ensure that the conversation feature in ChatGPT does not affect the outcome.

The Page Titled success criterion is helpful for users using screen readers to identify the page without the need to delve into the page content (Campbell et al., 2024b). To meet the success criterion on a web page a descriptive title has to be provided in the HTML source code.

There are two helpful techniques described in the WCAG documentation for success criterion 2.4.2 to help achieve conformance, H25 and G88. Technique H25 is HTML-specific and requires the <title> tag to be in place. The G88 technique is informal on how to provide a descriptive title that should describe the content of the page (Campbell et al., 2024a). To test for this technique, the page has to have a title, the title has to be relevant to the content and the page content should be identifiable solely based on the title.

To help accessibility evaluation tools and methodology developers, an ACT rule has been created with examples on how to evaluate if a page title is descriptive (Nørregaard and O'Connor, 2023). The ACT rule contains test cases on how tools should interpret pass, fail, and inapplicable criteria when checking conformance. There are in total seven examples of ACT cases provided by WCAG, of which three should pass, three should fail and one is inapplicable.

Passed test cases are referred to as P1, P2, and P3, as shown in Figure 4.2. P1 should pass as the title describes the content of the web page. P2 takes into account the assumption that web browsers only pick the first title element the browser encounters. P3 also assumes that the browser can parse the erroneous placement of the title element within the body (Nørregaard and O'Connor, 2023).

Failing test cases are F1, F2, and F3, as shown in Figure 4.3. The F1 test case title does not describe the content of the page. In the F2 test case, the first title is incorrect, but the second is correct. However, as browsers typically use the first title element found, the test should fail. The title in F3 is too generic and does not describe the content of the page (Nørregaard and O'Connor, 2023).

Inapplicable I1, see Figure 4.4, test example is defined that there are no applicable HTML nodes that contain a title element for the page. The title element is part of a Scalable Vector Graphic (SVG) and provides an accessible name for the SVG rather than the whole web page.

P1

```
<html lang="en">
 <head>
  <title>Clementine
      harvesting season
     </title>
 </head>
 <body>
  <p>Clementines will
      be ready to
      harvest from late
       October through
      February.</p>
 </body>
</html>
```

P2

```
<html lang="en">
 <head>
  <title>Clementine
      harvesting season
     </title>
  <title>Second title
      is ignored</title
     >
 </head>
 <body>
  <p>Clementines will
      be ready to
      harvest from late
       October through
      February.</p>
 </body>
</html>
```

P3

```
<html lang="en">
 <head> </head>
 <body>
  <title>Clementine
      harvesting season
     </title>
  <p>Clementines will
      be ready to
      harvest from late
       October through
      February.</p>
 </body>
</html>
```

**Figure 4.2:** Passed test cases P1, P2 and P3 copied from Nørregaard and O'Connor (2023)

F1

```
<html lang="en">
 <head>
  <title>Apple
      harvesting season
     </title>
 </head>
 <body>
  <p>
   Clementines will be
       ready to harvest
        from late
       October through
       February.
  </p>
 </body>
</html>
```

F2

```
<html lang="en">
 <head>
  <title>First title is
       incorrect</title
      >
  <title>Clementine
      harvesting season
     </title>
 </head>
 <body>
  <p>
   Clementines will be
       ready to harvest
        from late
       October through
       February.
  </p>
 </body>
</html>
```

F3

```
<html lang="en">
 <head>
  <title>University of
      Arkham</title>
 </head>
 <body>
  <h1>Search results
      for "
      accessibility" at
       the University
      of Arkham</h1>
  <p>None</p>
 </body>
</html>
```

**Figure 4.3:** Failed test cases F1, F2 and F3 copied from Nørregaard and O'Connor (2023)

I1

```
<html lang="en">
 <head>
  <title>University of
      Arkham</title>
 </head>
 <body>
  <h1>Search results
      for "
      accessibility" at
       the University
      of Arkham</h1>
  <p>None</p>
 </body>
</html>
```

**Figure 4.4:** Inapplicable test case I1 copied from Nørregaard and O'Connor (2023)

## 4.5 Iteration phases

The design & development, demonstration, and evaluation phases are used to answer RQ2. Details found in Chapter 3 are used to identify limitations in current accessibility evaluation tools and methodologies. The goal is to find out if the LLM could be prompted in a way that could recognize accessibility issues.

As a basis for the prompt, a Zero-Shot prompting method will be used. Zero-shot prompting means that no task-specific examples are provided within the prompt that would guide the LLM on how to accomplish the task correctly, instead the instructions are given manually in the input prompt (Kojima et al., 2022).

As a basis, the persona pattern and context manager pattern techniques from White et al. (2023) on AI prompting will be used when constructing the artifact. A prompt is a set of conditions given to the LLM. The persona pattern is used to emphasize the topic of discussion, while the context pattern is used to specify the context of the input to take into account. As generalization is important for the artifact (Peffers et al., 2018), techniques of the template pattern are utilized to some extent to ensure that the artifact could be used for multiple success criteria in the WCAG related to context evaluation. Template

pattern will guide the manual prompt building by trying to use possible placeholders in the artifact that could be replaced with information from other ACT rules provided by the WCAG.

### 4.5.1 Initial iteration

The design of the first iteration prompt uses the persona and context pattern. The procedures in Campbell et al. (2024a) under the section "Tests" are used as base rules. These rules are given in the prompt to specify the context and actions for the LLM. The expected result is that all of these three rules are fulfilled in each of the ACT cases. In addition, the assumption of the first title element being the one recognized by browsers in HTML code by Nørregaard and O'Connor (2023), is added to the ruleset as a fourth condition to check for.

As WCAG is technology agnostic (Campbell et al., 2023), the wording "document" will be changed to "web page" reflecting the style of other rules provided in the prompt. In the persona pattern, a descriptive title is required to be provided to the LLM. Therefore, the "Web accessibility specialist" will be used to set the persona for the LLM, as it is a title used by International Association of Accessibility Professionals (2024) which is a known accessibility organization providing certificates in web accessibility. The first iteration of the prompt is shown in Figure 4.5.

### 4.5.2 Second iteration

A second iteration, see Figure 4.6, is done based on the perceived problems found when evaluating the output of the first version of the artifact. However, ACT cases will not be changed, even though the outcome indicates that the P2 and F2 test cases with multiple titles stating that *"First title is incorrect"* and *"Second title is ignored"* may affect the outcome.

To improve the accuracy of the output the Zero Shot Chain of Thought prompt method presented by Kojima et al. (2022) will be used. As the fourth conditional rule that guides the LLM to pick the first title element encountered in the HTML code appeared to affect the conformance outcome of the F2 test case, therefore the fourth rule is shortened and added as a separate sentence before the three main ACT rules for the success criterion. Moreover, the LLM is explicitly given a task to evaluate the web page based on the

2.4.2 Page Titled Success Criterion and given specific instructions for this, therefore the question at the end of the prompt will be removed and the wording "Let's go step by step" demonstrated in Kojima et al. (2022) will be added the aim to increase the usefulness, accuracy, and consistency.

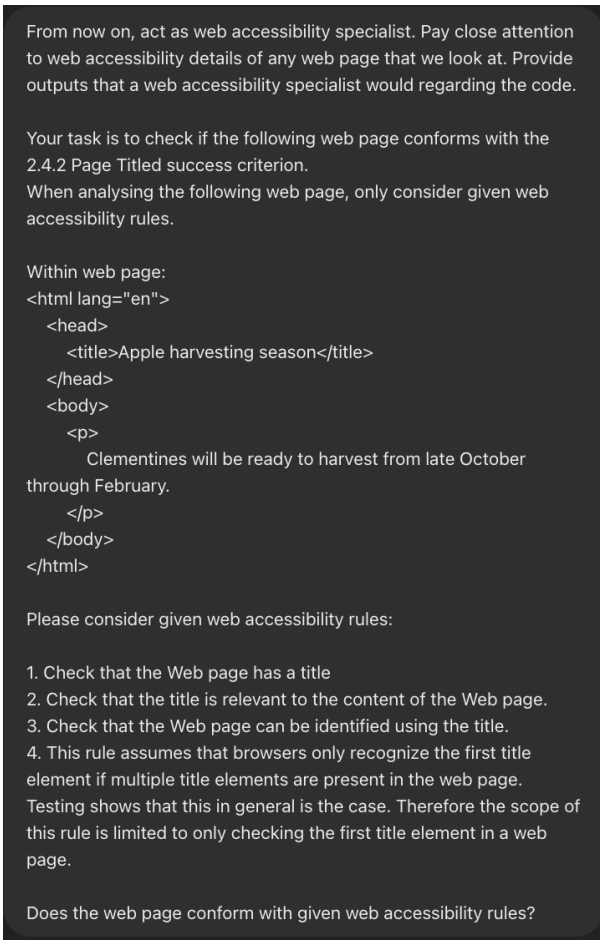


**Figure 4.5:** The first iteration of the input for the LLM with F1 test case.

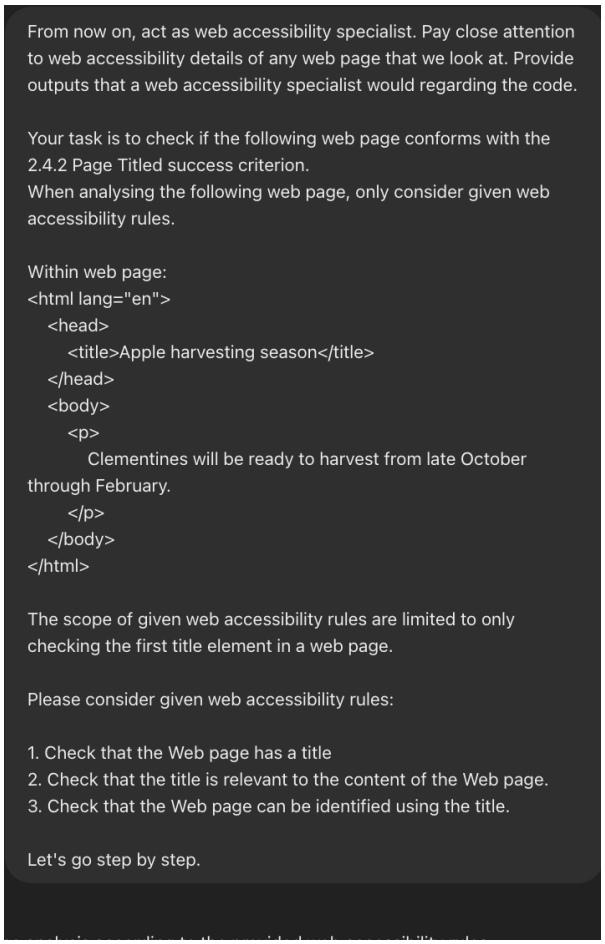


**Figure 4.6:** The second iteration of the input for the LLM with F1 test case.

## 4.6 Evaluation metrics

Evaluation in DSRM is outcome-based (Peffers et al., 2018). The output of the LLM is artificially produced text that will be evaluated as an entirety, taking into account the perceived usefulness of the output. Perceived usefulness as an evaluation metric is used in user acceptance of systems (Davis, 1989). It is based on a person's subjective opinion that using the system would enhance the effectiveness and productivity of the person's task. Subsequently, on each iteration, the LLM output will be evaluated for accuracy and

consistency. Accuracy and consistency of the output need to be evaluated as LLMs are non-deterministic (Ouyang et al., 2023; Power, 2021).

The artifact will be evaluated based on generalization at the end of the last iteration in Chapter 5. The generalization of the artifact will be evaluated to see if the artifact could be utilized for other ACT rules that evaluate similar context-based success criteria. For example, currently proposed ACT rules are "Heading descriptive" or "Link descriptive".

The evaluation of perceived usefulness will consist of how relevant and helpful the output is to the user. The output of the LLM will be evaluated with the following characteristics:

- provides suggestions for improvement when applicable (F1, F2 and F3 test cases)
- assesses whether the LLM output accurately reflects the test case with the rules provided

Instructions provided to the LLM will not explicitly ask for improvements in accessibility as the final evaluation for conformance should be done by the evaluator. The perceived usefulness will be evaluated by the thesis writer.

Accuracy is evaluated based on the seven pre-defined test cases provided by W3C in Nørregaard and O'Connor (2023). An outcome of the overall compliance is expected from the LLM based on the rules provided in the input artifact. A passed outcome means that the LLM meets all provided rules and a failed means that it partially meets the rules (Nørregaard and O'Connor, 2023). In regards to the inapplicable test case, the accuracy can not be determined.

In addition, consistency is evaluated by executing the same input and evaluating how often it provides perceivable identical answers in the form of overall accuracy. In other words, the output text states conformance with the success criterion in some form.

# 5 Results

This chapter presents the results and answers the research questions. Section 5.1 answers the first research question. In Section 5.2 the outcome of the LLM is analyzed for each iteration based on the evaluation criteria. In the last section, the artifact is studied from the perspective of how it could be used in other possible success criteria and ACT rules provided by W3C.

## 5.1 Limitations of web accessibility evaluation tools

The literature review done in Chapter 3 answers RQ1. In total, there are 86 success criteria in the WCAG 2.2 documents. Out of these 86 success criteria, test automation covers only 17 reliably (zero false positives). Using manual evaluation with semi-automated web accessibility evaluation tools increases the coverage, and adds to the sufficiency of these 17 found by automation (Deque Systems, 2022).

Research on Automated AETs shows that there is a significant distribution of the amount of found accessibility barriers on the same page (Frazão and Duarte, 2020; Rajh and Debevc, 2023). Studies conducted point out that even experts evaluating the same page can end up with different results (Brajnik et al., 2010). Additionally, the transparency of results by AETs to accessibility evaluators even further adds to the complexity of interpreting which success criteria have been sufficiently evaluated.

Semi-automated web accessibility evaluation tools help in conformance evaluation by guiding the evaluator with wizards to further increase coverage (Deque Systems, 2022). However, the outcome of the final evaluation is subjective to the evaluator (Brajnik et al., 2010). To fully conform with some of the success criteria requires knowledge of the context of the web page. Therefore, expertise matters when evaluating web pages for conformance using semi-automated AETs.

**Table 5.1:** Table summarizing the overall result of whether the LLM succeeded in the conformance evaluation.

| Test case | P1 | | P2 | | P3 | | F1 | | F2 | | F3 | | I1 | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Iteration | 1 | 2 | 1 | 2 | 1 | 2 | 1 | 2 | 1 | 2 | 1 | 2 | 1 | 2 |
| Sequence 1 | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | | ✓ | | ✓ | | ✓ | ✓ | |
| Sequence 2 | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | | ✓ |
| Sequence 3 | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |
| Sequence 4 | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | | ✓ | ✓ | ✓ | | |
| Sequence 5 | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | | ✓ | | ✓ | ✓ | ✓ | | |

## 5.2 Capabilities of LLM's for Page Titled evaluation

This section describes the outcomes of the artifact and answers RQ2. Some of the success criteria have context-based evaluation metrics, such as 2.4.2 Page Titled success criterion. The results show that the Large Language Model is capable of assisting in a conformance evaluation when evaluating the context-based success criterion 2.4.2 Page Titled. However, a more detailed look at how it ended up with the correct outcome shows a discrepancy.

In both iterations of the artifact, the passing test cases P1, P2, and P3 were the most consistent, accurate, and useful. The LLM did not suggest any improvements for the ACT passing test cases, as each sequence was correctly evaluated and already had a descriptive title present.

The LLM had issues evaluating the test cases F1 and F2 in the first iteration. The second iteration outcome showed improvements in the conformance evaluation, see Table 5.1. However, even though the second iteration explicitly states partial conformance for the F1 and F2 test cases in the outcome, how the LLM ended up to this conclusion had fallacies. In both iterations, out of the 30 sequences, there were only two output sequences (F1-1-1 and F1-1-5) that did not suggest an improved title for the page. The F3 test case did not cause major problems for the LLM.

In both iterations of the artifact, only two inapplicable sequences were evaluated correctly, therefore the evaluation outcome can not be relied upon.

Each iteration is analyzed in its subsection in more detail. Iteration results are evaluated more deeply based on usefulness, accuracy, and consistency. See Appendix A for actual outputs.

### 5.2.1 First iteration

The LLM was capable of evaluating if the title was descriptive based on the content of the web page. However, problems occurred in failing and inapplicable test cases. Passing test cases had a successful outcome based on our evaluation criteria, whereas failing and inapplicable test cases had a variance in consistency and fallacies in reasoning.

#### Passed test cases

When assessing the output in detail for usefulness, the LLM provides reasoning for each four of the rules in the input. However, the P2 test case's fourth sequence (P2-1-4) states that the code snippet has only one title element, even though the test case has two.

In the P3 test case, it can be noted that the LLM is not capable of identifying correctly where the title element is present, and in three out of five sequences, it falsely states that the title element is within the head section of the HTML document, even though it is in the body section. However, this is a minor mistake, as most browsers can fix and set the title correctly.

The accuracy of the accessibility evaluation is described in a short paragraph at the end of the output for all the test cases. Each test case and its sequences have the same outcome from a contextual perspective, therefore the LLM is accurate and consistent when evaluating these passed test cases.

#### Failing test cases

Inconsistency and inaccuracy are observed in the failing test cases F1 and F2 sequences. Additionally, the usefulness of the outcome is not at an appropriate leave as the reasoning has fallacies. However, the F3 test case sequence outcomes did not have similar problems as in test cases F1 and F2.

#### F1

F1-1-2, F1-1-3 and F1-1-4 have the correct outcome where the LLM states partial conformance. Additionally, those sequences that succeed in evaluating conformance correctly also provide suggestions for improvement.

The first sequence (F1-1-1) has problems understanding that the whole web page code is

provided. In the outcome, there is no mention of clementines being the correct context of the page. Additionally, the same uncertainty of what the LLM is evaluating can be understood from the last paragraph which seems to state conformance:

> Overall, based on the provided HTML code and the given web accessibility rules, the web page appears to conform with the criteria. However, further evaluation within the context of the entire website and user experience might be necessary for a comprehensive assessment.

In the fifth sequence (F1-1-5), the LLM explicitly indicates that the page conforms with the 2.4.2 Page Titled success criterion. In addition, the LLM seems to provide an irrational conclusion for the second rule *Check that the title is relevant to the content of the Web page* with an output of:

"The title of the web page is "Apple harvesting season". This title seems relevant to the content of the page, which discusses the harvest timing of clementines, a type of fruit."

**F2**

The F2 test case with two titles, see Figure 4.3, where the first one is incorrect and the second one is correct causes issues with the conformance evaluation made by the LLM. In regards to accuracy, the LLM states conformance in three out of the five sequences. Therefore, three out of five fail to provide an accurate conformance evaluation as F2 is a test case that has partial conformance.

A closer look at the three not correctly evaluated test sequences indicates that the existence of two titles on the web page code, and the fourth rule pointing out to use the first encountered title causes problems for reasoning, see Figure 4.5 for the four rules. The LLM finds the second title on the web page and uses it to evaluate conformance for the first three rules. This ordering of rules causes problems for the LLM, indicating that the scope of the fourth rule is not in a logical order.

Below is an example from F2-1-4 output from the LLM on the fourth rule reasoning, which is similar in all three not correctly evaluated cases (F2-1-1, F2-1-4, and F2-1-5): "Since browsers typically recognize only the first title element, and in this case, the relevant title is the second one, it aligns with the rule.". The LLM evaluated the conformance using the second title even though the rule explicitly said to do the evaluation based on the first title found in the HTML code.

Due to the possible problems described above, the accuracy for this test case is not on a sufficient level, nor the consistency. Therefore, the output that the LLM provided for evaluating conformance has fallacies. However, in each sequence, the LLM points out that the second title would be more relevant.

#### F3

All, except the first sequences (F3-1-1), the answer is consistent and all outcomes accurately state partial conformance and provide suggestions on how to improve the title description. The partial conformance from each outcome is based on the first and fourth rule of the input artifact. In other words, the web page has a title and there is only one title present.

However, the first sequence ends with "Overall, while the web page meets the basic requirement of having a title, it could improve its accessibility by making the title more descriptive and directly relevant to the content, such as "Search results for 'accessibility' at University of Arkham".", indicating that the test case passed the conformance evaluation, but the title could be improved.

#### Inapplicable test case (I1)

In only two of the sequences, it was able to indicate that the page should not be evaluated as it does not have a proper HTML structure as the title tag is used to describe the SVG element. On other test sequences, it satisfied the success criterion. However, as this is a zero-shot prompt there are no examples of how to determine applicability or inapplicability. Therefore, the outcome of the sequences that try to evaluate this inapplicable test case is of limited use if the accuracy can not be improved.

### 5.2.2 Second iteration

The improvements done in the second iteration of the artifact had a significant effect on the outcome of failing test cases. With the removal of the fourth rule and using it as a separate instruction in the artifact, the usefulness, accuracy, and consistency increased.

### Passed test cases

In regards to accuracy, all sequences within the passing test cases were evaluated correctly. Therefore, each test case sequence was consistent regarding accuracy.

Logical reasoning in each outcome on how it evaluated conformance is provided by the LLM. Therefore, the artifact and the outcome are useful for the evaluator. As in the first iteration of the artifact, the LLM still had issues identifying in the P3 test case that the title element is not within the head section. As earlier stated in the first iteration results, this is a minor flaw that does not affect the result when the descriptive title is within the body tag.

### Failing test cases

In comparison to the first iteration of the artifact, significant improvements in overall accuracy can be identified for the failing test cases. Accuracy and consistency improvements are distinguished from the outputs, as each sequence outcome is perceived as partial conformance.

#### F1

Each of the five sequences ends up in partial conformance. Partial conformance is based on the first rule passing, that is, the web page has a title. One sequence (F1-2-2) stood out in the results, where there is a difference from the other sequences. The F1-2-2 states that it conforms with the third rule where it should check that the web page can be identified by the title "Apple harvesting season", even though the content is about clementines. However, all except the second sequence (F1-2-2) have similar contextual output, where the first rule is met, but the second and third rule fails, ending up in partial conformance.

#### F2

The expected result is that partial conformance should come out of the second and third conditions to check for, as the first title is not relevant to the page content and the title does not identify the web page.

The input artifact explicitly states the following: "The scope of given web accessibility rules are limited to only checking the first title element in a web page.". However, within

each of the outcomes, the LLM has used the second title "Clementine harvesting season" in assessing conformance. However, all of them end up in partial conformance due to either the second or third condition being evaluated based on the first title found. Additionally, it can be observed that the second title is always mentioned as a better option for the page title from the LLM output detail of the third rule.

Overall, in terms of accuracy and consistency, each outcome states partial conformance that is described in the last paragraph of each outcome. However, the LLM neglects the information that it should only be evaluating the first found title.

#### F3

A direct quote from one of the sequences summarizes the results very well: "Overall, while the page meets the first criterion by having a title, it falls short of the other two criteria as the title isn't directly relevant to the content and may not effectively identify the page.". Therefore, consistency and accuracy are spot on. Additionally, in each sequence, the reasoning for the outcome is similar and the LLM suggests either explicitly or implicitly a more descriptive title for the test case.

#### Inapplicable test case (I1)

Only the second and third sequence indicates successful evaluation for inapplicability as there is no title element for the whole web page, rather the title element is for the SVG element. For example, the first check that the web page has a title is evaluated in the following way: "The web page contains a <title> element within the <svg> tag. However, the <title> element is intended for providing a title for the SVG graphic for accessibility purposes, not for the entire web page".

### 5.2.3 Generalization

In Design Science Research Methodologies, the generalization of the artifact is important. Generalization is evaluated on how well the artifact works in other similar contexts (Peffers et al., 2018). This section evaluates the possibility of using the artifact for other conformance-checking techniques provided in the WCAG documents.

In its current form, as a proof of concept, the possibility of using the artifact for other ACT rules is limited because the input prompt is specifically tailored to test if the web page

title is descriptive. The artifact is a proof of concept on how LLM evaluates accessibility using the technique provided by Campbell et al. (2024a) and modified accessibility support information from Nørregaard and O'Connor (2023).

By incorporating the input prompt in a semi-automated accessibility evaluation tool, placeholders in the input prompt could be utilized, therefore improving the generalization of the artifact. However, the artifact works as "the code" for the LLM, and a small change can have a significant change in the quality of the output, as demonstrated in this study. Therefore, placeholders, choice of words, and small changes in the input prompt require thorough testing and evaluation.

Furthermore, currently, the artifact is not tied to any specific accessibility evaluation tool. The input artifact could be used by any organization and in other Large Language Models, as the input prompt was the artifact iterated upon.

To conclude the generalization, the artifact has the potential to be used in other contexts, as the rules could be easily swapped with other ACT rules for other success criteria, and the custom sentence on picking the first title found could be an additional placeholder.

# 6 Discussion

This chapter contains a discussion of the study, implications, limitations, and potential future research. Section 6.1 is a summary of the research questions. Section 6.2 a discussion on the study result and implications are presented. Section 6.3 presents threats to the validity of the study. Section 6.4 promotes potential future research.

## 6.1 Summary of main findings

Below is a recap of the research questions and a summary of the answers.

- **RQ1.** What are the limitations of web accessibility evaluation tools in assessing compliance with Web Content Accessibility Guidelines (WCAG)?

A multivocal literature review answered the current limitations of web accessibility evaluation tools. Test automation covers 17 out of the 86 success criteria in the WCAG 2.2. However, test automation tools are not capable of thorough evaluation that would meet the WCAG set standard for some of these 17 success criteria.

Sufficiently evaluating conformance requires an accessibility specialist evaluation which is a tedious process, as web pages are more complicated than ever. Semi-automated accessibility evaluation tools help evaluators by guiding them through the most common accessibility barriers found on web pages. However, manual evaluation can end up with a different outcome depending on the evaluator's knowledge and workmanship.

- **RQ2.** How does Generative AI assist in addressing these limitations?

Generative AI can be utilized to address these limitations. This study shows that Large Language Models (LLM) can assist evaluators in conformance checks on the page title that require a thorough understanding of the content. However, a more detailed observation shows that the LLM had fallacies in how it ended up with the correct outcome when conducting a conformance evaluation for the failing ACT cases. On pages with more content, the outcome of the LLM could speed up the conformance evaluation process. However, as results show, with a zero-shot chain of thought prompting, LLMs are not capable of reliably determining inapplicability.

## 6.2 Study result analysis

This study shows that even though legislation is moving forward in regards to accessibility, the nature of accessibility and accessibility evaluation is complex, and requires expertise from designers, developers, quality assurance, and accessibility reviewers. An accessibility specialist needs to have a thorough knowledge of the WCAG documents, as no automated or semi-automated accessibility evaluation tool has 100% coverage. In addition, web developers, designers, and content creators ought to study the same WCAG documents.

An accessibility evaluation tool developer has to understand the ACT rules and sufficient techniques used to check for conformance, as well as how browsers work, to develop a reliable and robust tool for accessibility evaluators. Transparency of evaluation tools helps the evaluator understand which success criteria it covers and to what extent. The study shows that Large Language Models can be utilized to improve the sufficiency of the accessibility evaluation and help ease the conformance evaluation process when using semi-automated accessibility evaluation tools. However, the output of the LLM should be evaluated by a human on how it ended up to the conclusion. Therefore, an LLM would be an additional asset in AETs when conducting conformance evaluations.

### 6.2.1 Prompt iteration

These results build on existing evidence that LLMs are good zero-shot reasoners (Kojima et al., 2022). With rigorous prompt iteration, the accuracy and quality of the outcome improved. By evaluating the outcome of the LLM, patterns can be detected where the LLM fails to provide reasoning for checks it performed, giving possible directions for improvements. An example observed in this study between the iterations is the order of the conditional checks the LLM should take into account. Therefore, an imperative approach to how the LLM should operate step by step combined with the zero-shot chain of thought improved the quality of the outcome. However, as the second iteration of the artifact went through multiple changes, it remains unknown if Kojima et al. (2022) zero-shot chain of thought change, or moving the fourth rule as a separate sentence, affected the accuracy and consistency in this study.

## 6.3 Limitations

The lack of a thorough evaluation, such as surveys or interviews with potential users of this artifact, is a concern of the validity of this study. The evaluation of usefulness is solely based on the observations of the thesis writer. The artifact is a proof of concept and has not been evaluated by accessibility evaluators for usefulness in accessibility conformance reviews. Therefore, a proof of suitability, evaluating whether the LLM would assist and speed up conformance reviews, is yet to be evaluated that would support the findings of the study.

Two concerns regarding the chosen LLM are that this study was solely done using the ChatGPT 3.5 user interface due to it being free to use and that the LLM is a closed-source tool. Therefore, between the iterations, there is no knowledge if there have been any improvements made by OpenAI to the language model. However, according to OpenAI, the ChatGPT 3.5 model was trained only with data available in early 2022 (OpenAI, 2024). Therefore, it can be assumed that the test cases available at Nørregaard and O'Connor (2023) should not be part of the LLM knowledge base. The replication of this study should be straightforward with different LLMs, such as the new GPT-4o, which was announced in September 2024 and is now the main model provided to free-tire users of OpenAI.

Additionally, the characteristics of LLMs, such as non-deterministic output given the same input, or the limited amount of characters that you can input, are a threat to the external validity of the study. This study was conducted using very short code snippets, therefore no input limits were hit. Even though, in the second iteration all the passing and failing test case sequences were correctly evaluated by the LLM, this does not guarantee that the LLM would always correctly evaluate due to the characteristics of LLM being non-deterministic. However, a better design could reduce the input limitation threat. For example, in this study setting, parsing the HTML code, picking the first title element encountered, and the content within the HTML body or main tag would significantly reduce the number of characters sent to the LLM. In addition, the non-deterministic behavior would cause problems if your accessibility evaluation tool promises zero false positives when evaluating for accessibility.

It is beyond the scope of this study to evaluate how the LLM would work if the website language were other than English, or with other large language models available, either provided by some entity online or running the models locally.

## 6.4 Future research

Future studies should take into account the language used on the website, as accessibility barriers are not only limited to English, specifically, to low-resource languages that have less data available to train the LLMs. In addition, as this study was limited to the short ACT cases, a case study where the artifact would be implemented into a semi-automatic accessibility evaluation tool with a feedback loop from the evaluator would provide insight into how the LLM evaluates more complex websites with more content, and how LLMs could improve the efficiency of the conformance review. Additionally, as conformance reviews are a tedious process, automated ways to measure and observe the output of an LLM's effectiveness and trustworthiness for success criteria evaluations would grow the coverage of automated accessibility evaluation tools.

# 7 Conclusions

The goal of this study was to find out how sufficiently current accessibility evaluation tools test the Web Content Accessibility Guideline version 2.2 success criteria and to assess the potential of large language models evaluating a context-based accessibility criterion. It is suggested to use multiple accessibility evaluation tools during conformance reviews to improve the coverage of found accessibility barriers. The findings show that large language models, when given conditions to check for in the prompt, can assist in that the HTML code has a title, the title is relevant to the page content, and the title identifies the page. However, a human evaluator must assess the accuracy of the output, particularly by examining how the large language model reached its conclusions based on the specified conditions. Additionally, an accessibility specialist does not necessarily need to be a subject matter expert regarding website content, as a large language model would help evaluate the context-based criteria. Integrating Generative AI, such as large language models, into accessibility evaluation tools could enhance the accuracy and efficiency of conformance evaluation, enabling accessibility reviewers to carry out more comprehensive accessibility assessments.

# Bibliography


Abou-Zahra, S. (2017). *Diverse Abilities and Barriers*. Accessed: 16.04.2024. URL: https://www.w3.org/WAI/people-use-web/abilities-barriers/.

– (2018). *WCAG 2.1 Adoption in Europe*. Accessed: 07.02.2024. URL: https://www.w3.org/blog/2018/wcag-2-1-adoption-in-europe/.

Abou-Zahra, S. and Henry, S. L. (2020). *Accessibility Conformance Testing (ACT) Overview*. Accessed: 18.03.2024. URL: https://www.w3.org/WAI/standards-guidelines/act/.

Brajnik, G. (Oct. 2004). "Comparing accessibility evaluation tools: a method for tool effectiveness". In: *Universal Access in the Information Society* 3.3, pp. 252–263. ISSN: 1615-5297. DOI: 10.1007/s10209-004-0105-y. URL: https://doi.org/10.1007/s10209-004-0105-y.

– (2008). "A comparative test of web accessibility evaluation methods". In: *Proceedings of the 10th International ACM SIGACCESS Conference on Computers and Accessibility*. Assets '08. Halifax, Nova Scotia, Canada: Association for Computing Machinery, pp. 113–120. ISBN: 9781595939760. DOI: 10.1145/1414471.1414494. URL: https://doi.org/10.1145/1414471.1414494.

Brajnik, G., Harper, S., and Yesilada, Y. (2009). *How much does expertise matter?: a barrier walkthrough study with experts and non-experts*. Accessed: 11.03.2024. URL: https://doi.org/10.1145/1639642.1639678.

Brajnik, G., Yesilada, Y., and Harper, S. (2010). "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. Orlando, Florida, USA: Association for Computing Machinery, pp. 43–50. ISBN: 9781605588810. DOI: 10.1145/1878803.1878813. URL: https://doi.org/10.1145/1878803.1878813.

Bureau of Internet Accessibility (2023a). *3 Ways That Artificial Intelligence Can Improve Web Accessibility*. Accessed: 02.05.2024. URL: https://www.boia.org/blog/3-ways-that-artificial-intelligence-can-improve-web-accessibility.

– (2023b). *Be Careful When Using A.I. for Alternative Text*. Accessed: 02.05.2024. URL: https://www.boia.org/blog/be-careful-when-using-ai-for-alternative-text.

Campbell, A. (Apr. 2024). *Re: AI and the future of Web accessibility Guidelines from Alastair Campbell on 2024-04-19 (w3c-wai-gl@w3.org from April to June 2024)*. Ac-

cessed: 02.05.2024. URL: https://lists.w3.org/Archives/Public/w3c-wai-gl/2024AprJun/0043.html.

Campbell, A., Adams, C., and Montgomery, R. B. (2024a). *G88: Providing descriptive titles for Web pages | WAI | W3C*. Accessed: 29.04.2024. URL: https://www.w3.org/WAI/WCAG22/Techniques/general/G88.

– (2024b). *Understanding Success Criterion 2.4.2: Page Titled | WAI | W3C*. Accessed: 29.04.2024. URL: https://www.w3.org/WAI/WCAG22/Understanding/page-titled.html.

Campbell, A., Adams, C., Montgomery, R. B., Cooper, M., and Kirkpatrick, A. (2023). *Web Content Accessibility Guidelines (WCAG) 2.2*. Accessed: 27.02.2024. URL: https://www.w3.org/TR/WCAG22/.

Chadli, F. E., Gretete, D., and Moumen, A. (2023). "Comparison of Free and Open Source WCAG Accessibility Evaluation Tools". In: *Proceedings of the 6th International Conference on Networking, Intelligent Systems & Security*. NISS '23. Larache, Morocco: Association for Computing Machinery. DOI: 10.1145/3607720.3607722. URL: https://doi.org/10.1145/3607720.3607722.

Davis, F. (Sept. 1989). "Perceived Usefulness, Perceived Ease of Use, and User Acceptance of Information Technology". In: *MIS Quarterly* 13, pp. 319–. DOI: 10.2307/249008.

Deque (2024). *Intelligent Guided Tests | Deque Docs*. Accessed: 02.05.2024. URL: https://docs.deque.com/devtools-for-web/4/en/devtools-igt.

Deque Systems (2021). *The Automated Accessibility Coverage Report*. Accessed: 06.03.2024. URL: https://accessibility.deque.com/hubfs/Accessibility-Coverage-Report.pdf.

– (2022). *The Semi-Automated Accessibility Testing Coverage Report*. Accessed: 10.04.2024. URL: https://accessibility.deque.com/hubfs/Semi-Automated-Accessibility-Testing-Coverage-Report.pdf.

Directive 2016/2102 (2016). *Directive (EU) 2016/2102 of the European Parliament and of the Council of 26 October 2016 on the accessibility of the websites and mobile applications of public sector bodies*. Accessed: 07.02.2024. URL: https://eur-lex.europa.eu/eli/dir/2x016/2102/oj.

Directive 2019/882 (2019). *Directive (EU) 2019/882 of the European Parliament and of the Council of 17 April 2019 on the accessibility requirements for products and services*. Accessed: 06.02.2024. URL: https://eur-lex.europa.eu/eli/dir/2019/882/oj.

Dolson, J. (June 2023). *Accessibility and Artificial Intelligence - Joe Dolson Web Accessibility*. Accessed: 02.05.2024. URL: https://www.joedolson.com/2023/06/accessibility-and-artificial-intelligence/.

Duran, M. (2017). *What we found when we tested tools on the world's least-accessible webpage*. Accessed: 07.04.2024. URL: https://accessibility.blog.gov.uk/2017/02/24/what-we-found-when-we-tested-tools-on-the-worlds-least-accessible-webpage/.

Duran, M., Duggin, A., and Morton, R. (2017). *What we found when we tested tools on the world's least-accessible webpage*. Accessed: 05.03.2024. URL: https://accessibility.blog.gov.uk/2017/02/24/what-we-found-when-we-tested-tools-on-the-worlds-least-accessible-webpage/.

– (2018). *Accessibility tools audit results - Results - GDS accessibility team*. Accessed: 06.03.2024. URL: https://alphagov.github.io/accessibility-tool-audit/index.html.

Eggert, E. (2023). *How to Meet WCAG (Quick Reference)*. Accessed: 07.04.2024. URL: https://www.w3.org/WAI/WCAG22/quickref/?currentsidebar=%23col_customize&showtechniques=111&levels=aaa&techniques=advisory&technologies=js%2Cserver%2Csmil%2Cpdf.

ETSI (2024). *EN 301 549 V3 the harmonized European Standard for ICT Accessibility*. Accessed: 07.04.2024. URL: https://www.etsi.org/human-factors-accessibility/en-301-549-v3-the-harmonized-european-standard-for-ict-accessibility.

Fiers, W. (2023). *Axe-core 4.5: First WCAG 2.2 Support and More*. Accessed: 07.03.2024. URL: https://www.deque.com/blog/axe-core-4-5-first-wcag-2-2-support-and-more/.

Frazão, T. and Duarte, C. (2020). "Comparing accessibility evaluation plug-ins". In: *Proceedings of the 17th International Web for All Conference*. W4A '20. Taipei, Taiwan: Association for Computing Machinery. ISBN: 9781450370561. DOI: 10.1145/3371300.3383346. URL: https://doi.org/10.1145/3371300.3383346.

Gustafson, A. (Feb. 2024). *Opportunities for AI in Accessibility – A List Apart*. Accessed: 02.05.2024. URL: https://alistapart.com/article/opportunities-for-ai-in-accessibility/.

Henry, S. L. (2023). *WCAG 2 Overview*. Accessed: 26.02.2024. URL: https://www.w3.org/WAI/standards-guidelines/wcag/.

Hevner, A. R., March, S. T., Park, J., and Ram, S. (2004). "Design Science in Information Systems Research". In: *MIS Quarterly* 28.1, pp. 75–105. ISSN: 02767783. URL: http://www.jstor.org/stable/25148625 (visited on Mar. 15, 2024).

International Association of Accessibility Professionals (2024). *Web Accessibility Specialist | International Association of Accessibility Professionals*. Accessed: 11.05.2024. URL: https://www.accessibilityassociation.org/s/wascertification.

Kojima, T., Gu, S. S., Reid, M., Matsuo, Y., and Iwasawa, Y. (2022). "Large language models are zero-shot reasoners". In: *Advances in neural information processing systems* 35, pp. 22199–22213.

Lange, V., White, K., and Hansma, M. (2024). *Selecting Web Accessibility Evaluation Tools*. Accessed: 05.03.2024. URL: https://www.w3.org/WAI/test-evaluate/tools/selecting/.

López, J. M. and Varela, J. P. (Mar. 2024). "Turning manual web accessibility success criteria into automatic: an LLM-based approach". In: *Universal Access in the Information Society*. DOI: 10.1007/s10209-024-01108-z.

Nørregaard, A. T. and O'Connor, C. (2023). *HTML page title is descriptive | ACT Rule | WAI | W3C*. Accessed: 29.04.2024. URL: https://www.w3.org/WAI/standards-guidelines/act/rules/c4a8a4/.

OpenAI (2024). *Introducing ChatGPT*. Accessed: 04.09.2024. URL: https://openai.com/index/chatgpt/.

Othman, A., Dhouib, A., and Nasser Al Jabor, A. (2023). "Fostering websites accessibility: A case study on the use of the Large Language Models ChatGPT for automatic remediation". In: *Proceedings of the 16th International Conference on PErvasive Technologies Related to Assistive Environments*. PETRA '23. Corfu, Greece: Association for Computing Machinery, pp. 707–713. DOI: 10.1145/3594806.3596542. URL: https://doi.org/10.1145/3594806.3596542.

Ouyang, S., Zhang, J. M., Harman, M., and Wang, M. (2023). "LLM is Like a Box of Chocolates: the Non-determinism of ChatGPT in Code Generation". In: *arXiv preprint arXiv:2308.02828*.

Parvin, P., Palumbo, V., Manca, M., and Paternò, F. (2021). "The transparency of automatic accessibility evaluation tools". In: *Proceedings of the 18th International Web for All Conference*. W4A '21. Ljubljana, Slovenia: Association for Computing Machinery. ISBN: 9781450382120. DOI: 10.1145/3430263.3452436. URL: https://doi.org/10.1145/3430263.3452436.

Peffers, K., Tuunanen, T., and Niehaves, B. (2018). "Design science research genres: introduction to the special issue on exemplars and criteria for applicable design science research". In: *European Journal of Information Systems* 27.2, pp. 129–139. DOI: 10.1080/0960085X.2018.1458066. URL: https://doi.org/10.1080/0960085X.2018.1458066.

Peffers, K., Tuunanen, T., Rothenberger, M., and Chatterjee, S. (Dec. 2007). "A Design Science Research Methodology for Information Systems Research". In: *J. Manage. Inf. Syst.* 24.3, pp. 45–77. ISSN: 0742-1222. DOI: 10.2753/MIS0742-1222240302. URL: https://doi.org/10.2753/MIS0742-1222240302.

Power, B. (Aug. 2021). *A question on determinism - API - OpenAI Developer Forum*. Accessed: 06.05.2024. URL: https://community.openai.com/t/a-question-on-determinism/8185.

Rajh, N. and Debevc, M. (2023). "Analysis of web accessibility evaluation tools and guidelines for monitoring according to the Directive (EU) 2016/2102". In: *Proceedings of the 10th International Conference on Software Development and Technologies for Enhancing Accessibility and Fighting Info-Exclusion*. DSAI '22. Lisbon, Portugal: Association for Computing Machinery, pp. 195–202. ISBN: 9781450398077. DOI: 10.1145/3563137.3563148. URL: https://doi.org/10.1145/3563137.3563148.

Regional State Administrative Agency (2023). *Accessibility overview*. Accessed: 05.02.2024. URL: https://www.webaccessibility.fi/accessibility-overview/.

The European Commission (2018). *establishing a monitoring methodology and the arrangements for reporting by Member States in accordance with Directive (EU) 2016/2102 of the European Parliament and of the Council on the accessibility of the websites and mobile applications of public sector bodies*. Accessed: 05.02.2024. URL: https://eur-lex.europa.eu/legal-content/EN/TXT/HTML/?uri=CELEX:32018D1524.

Vigo, M., Brown, J., and Conway, V. (2013). "Benchmarking web accessibility evaluation tools: measuring the harm of sole reliance on automated tests". In: *Proceedings of the 10th International Cross-Disciplinary Conference on Web Accessibility*. W4A '13. Rio de Janeiro, Brazil: Association for Computing Machinery. ISBN: 9781450318440. DOI: 10.1145/2461121.2461124. URL: https://doi.org/10.1145/2461121.2461124.

WAI, W. W. A. I. (2024). *Web Accessibility Evaluation Tools List*. Accessed: 18.03.2024. URL: https://www.w3.org/WAI/test-evaluate/tools/list/.

WebAIM (2024). *The 2024 report on the accessibility of the top 1,000,000 home pages*. Accessed: 17.04.2024. URL: https://webaim.org/projects/million/.

White, J., Fu, Q., Hays, S., Sandborn, M., Olea, C., Gilbert, H., Elnashar, A., Spencer-Smith, J., and Schmidt, D. C. (2023). "A prompt pattern catalog to enhance prompt engineering with chatgpt". In: *arXiv preprint arXiv:2302.11382*.

WHO, W. H. O. (2023a). *Blindness and vision impairment*. URL: https://www.who.int/news-room/fact-sheets/detail/blindness-and-visual-impairment (visited on Apr. 16, 2024).

– (2023b). *Disability*. URL: https://www.who.int/news-room/fact-sheets/detail/disability-and-health (visited on Feb. 5, 2024).

Yeliz, Y. and Simon, H. (2019). *Web Accessibility : A Foundation for Research*. Vol. 2nd ed. 2019. Human–Computer Interaction Series. Springer. ISBN: 9781447174394. URL: https://search.ebscohost.com/login.aspx?direct=true&db=nlebk&AN=2153923&site=ehost-live&scope=site.

Ziadé, T. (2024). *Experimenting with local alt text generation in Firefox Nightly*. Accessed: 10.09.2024. URL: https://hacks.mozilla.org/2024/05/experimenting-with-local-alt-text-generation-in-firefox-nightly/.

# Appendix A Test case outputs

The conversation is renamed in form of P1-1-1, where P1 refers to the test case in the Nørregaard and O'Connor (2023). First number describes the iteration, in other words, which version of the artifact and the last number is the sequence number ran for that iteration.

## A.1 P1

### A.1.1 Iteration 1

1. P1-1-1: https://chat.openai.com/share/5589d0f3-a3e3-4aee-bd09-63fb87503f4e
2. P1-1-2: https://chat.openai.com/share/c9cbb56f-9da3-42f4-928c-7931e1f6d6c6
3. P1-1-3: https://chat.openai.com/share/73173b78-220d-49de-85fd-a659f38ce09d
4. P1-1-4: https://chat.openai.com/share/b262027f-b852-4ffb-8691-bbff11a7b660
5. P1-1-5: https://chat.openai.com/share/e0c1b60f-b309-4996-a17b-21a3020ee029

### A.1.2 Iteration 2

1. P1-2-1: https://chatgpt.com/share/10e0e7c2-924f-4c94-b5d3-50fa3ce8e990
2. P1-2-2: https://chatgpt.com/share/22b4c9da-ad47-4bed-869b-e36e97d51faf
3. P1-2-3: https://chatgpt.com/share/1ffb4ae6-c998-4d38-b935-ac5b603ab482
4. P1-2-4: https://chatgpt.com/share/c42f634e-80a1-44d5-84c2-6ad533804760
5. P1-2-5: https://chatgpt.com/share/d70a12a0-a3c4-410d-86ee-859128972a83

## A.2 P2

### A.2.1 Iteration 1

1. P2-1-1: https://chat.openai.com/share/fa9ad3e2-ad93-4429-b6ee-38a7ef6bad0e
2. P2-1-2: https://chat.openai.com/share/79e7dd08-a79e-4594-80ef-87b7a07b64b2
3. P2-1-3: https://chat.openai.com/share/aa354068-bc22-4fdf-9ac3-4a783be9b5d6
4. P2-1-4: https://chat.openai.com/share/fb200c6f-a83d-4363-81e0-3e869ed52f48
5. P2-1-5: https://chat.openai.com/share/af4efc55-de9b-4b0b-b742-074798a1ba0c

### A.2.2 Iteration 2

1. P2-2-1: https://chatgpt.com/share/d180c25d-b1da-41eb-8b22-77f1e1b22ac6
2. P2-2-2: https://chatgpt.com/share/1ef1453a-0349-4aeb-8204-34b65e68a24f
3. P2-2-3: https://chatgpt.com/share/c707d684-a4cd-4ff0-94b2-6189302ee5db
4. P2-2-4: https://chatgpt.com/share/8e93bbe0-c85e-4a1e-9c3b-5f7aa944e375
5. P2-2-5: https://chatgpt.com/share/9ecfab41-ae15-4ca8-bda3-fdfc969d1842

## A.3 P3

### A.3.1 Iteration 1

1. P3-1-1: https://chat.openai.com/share/d2af5c6b-9489-4836-998f-c8af4d99213c
2. P3-1-2: https://chat.openai.com/share/a7e9bb5a-20ea-4eac-820c-dede05d34279
3. P3-1-3: https://chat.openai.com/share/18227ab9-65a6-4eaa-868e-b50774e96199
4. P3-1-4: https://chat.openai.com/share/dabac1ef-8ca3-4765-9366-e9296f9ddd7d
5. P3-1-5: https://chat.openai.com/share/9dac3e30-e2f0-4ae5-9f9a-20e3d7775d6c

### A.3.2 Iteration 2

1. P3-2-1: https://chatgpt.com/share/4b52f5f3-2da9-425a-91b7-aef5713bd04f
2. P3-2-2: https://chatgpt.com/share/1c3957f0-aa3c-47cd-9a1e-11d12436b25c
3. P3-2-3: https://chatgpt.com/share/6731c4b8-9d85-4905-a666-1a8810aafc23
4. P3-2-4: https://chatgpt.com/share/be4a8fbc-82cc-4408-8507-67187a1b99cf
5. P3-2-5: https://chatgpt.com/share/164a9db4-14d8-4cff-a751-a0729b4d4c67

## A.4 F1

### A.4.1 Iteration 1

1. F1-1-1: https://chat.openai.com/share/87a9d162-b402-40fc-8bfa-664c6a5ba3a2
2. F1-1-2: https://chat.openai.com/share/34809a27-8665-4f6c-84ab-7eb5b243f065
3. F1-1-3: https://chat.openai.com/share/ccb471b1-522e-4b85-9fec-b405e823086b
4. F1-1-4: https://chat.openai.com/share/af1997ef-ef04-46fc-8cfe-a1f8bba6daab
5. F1-1-5: https://chat.openai.com/share/c571e9ed-b9ab-4e54-9389-67c456776708

### A.4.2 Iteration 2

1. F1-2-1: https://chatgpt.com/share/07592414-6d62-4929-a9c5-9e51b100f92d
2. F1-2-2: https://chatgpt.com/share/0bbd3995-de1c-407e-a3d2-2f5c730faba3
3. F1-2-3: https://chatgpt.com/share/5d414e15-ea1f-4d5d-900a-612c16f824c1
4. F1-2-4: https://chatgpt.com/share/9b7eea10-1ae3-4904-a539-54781329d218
5. F1-2-5: https://chatgpt.com/share/b042e503-cd44-488f-ab45-619d63098c1e

## A.5 F2

### A.5.1 Iteration 1

1. F2-1-1: https://chat.openai.com/share/866af13f-2653-4d60-99e5-234f67a2be93
2. F2-1-2: https://chat.openai.com/share/1da36f2b-f112-4881-89ad-af37c4ee6aaf
3. F2-1-3: https://chat.openai.com/share/e33b7b58-a43c-4c0b-be08-4e05d31af6f2
4. F2-1-4: https://chat.openai.com/share/acec840d-8e8b-4a84-8499-4157e62efc41
5. F2-1-5: https://chat.openai.com/share/4bc77120-396e-43a3-8292-8594259b331b

### A.5.2 Iteration 2

1. F2-2-1: https://chatgpt.com/share/e3e048af-b623-439d-b267-0710f9b0c049
2. F2-2-2: https://chatgpt.com/share/fd8d7aab-6447-45f6-aef5-c90fd9171404
3. F2-2-3: https://chatgpt.com/share/d5fd65a8-da11-44e5-a065-42ca9455602d
4. F2-2-4: https://chatgpt.com/share/f05d4764-8b8a-41ab-82b6-a7d83b6bcc85
5. F2-2-5: https://chatgpt.com/share/c60d397a-d994-401d-9dc7-89bf3ff00118

## A.6 F3

### A.6.1 Iteration 1

1. F3-1-1: https://chat.openai.com/share/86f2fb56-fd98-4791-a764-0d61b13c920f
2. F3-1-2: https://chat.openai.com/share/f1814074-d83d-429e-ae03-1c6843d4aac0
3. F3-1-3: https://chat.openai.com/share/08684ef8-34e2-4b72-9d00-5215b0d90e69
4. F3-1-4: https://chat.openai.com/share/8cb6a180-8ff0-48e2-aa3c-d5f73f3993e6
5. F3-1-5: https://chat.openai.com/share/7e2f1b2a-a6c7-45f4-9901-7f3cba63c24e

### A.6.2 Iteration 2

1. F3-2-1: https://chatgpt.com/share/2c460c56-930a-45c6-9a9b-9f7e4823fd27
2. F3-2-2: https://chatgpt.com/share/423833b1-db6b-462d-b111-d6872495a113
3. F3-2-3: https://chatgpt.com/share/a3f0e2de-78c6-4205-8275-203514ef4717
4. F3-2-4: https://chatgpt.com/share/ae1d8b0a-738e-491f-921b-5168d8e73bdc
5. F3-2-5: https://chatgpt.com/share/4a04107e-5da9-435d-a61b-416798c929b5

## A.7 I1

### A.7.1 Iteration 1

1. I1-1-1: https://chat.openai.com/share/d6c95007-b965-4ca9-b176-4e3f9899bb0d
2. I1-1-2: https://chat.openai.com/share/a2789f63-0ad6-4e56-9a3f-0fc2de56fdae
3. I1-1-3: https://chat.openai.com/share/63d5dc58-6c52-4055-9357-b6e97fbbeeb0
4. I1-1-4: https://chat.openai.com/share/20b42730-76de-4345-a6db-ef130d43b1a9
5. I1-1-5: https://chat.openai.com/share/9ddfd254-5917-430f-8984-8610e6608c70

### A.7.2 Iteration 2

1. I1-2-1: https://chatgpt.com/share/c0148e6e-1121-4f12-94b1-d2a0ae89a37c
2. I1-2-2: https://chatgpt.com/share/6c7f11fa-59b5-4ed7-915a-379f22cca3a2
3. I1-2-3: https://chatgpt.com/share/71dfe5ee-5af6-4729-8fe5-09b436f76fb6
4. I1-2-4: https://chatgpt.com/share/d2f6f76b-47b6-4937-bdb3-a626f9c7b84c
5. I1-2-5: https://chatgpt.com/share/312537cb-477d-47f1-8701-dad5a0fbe6ce