Master Thesis Research Proposal

Assessing Web Accessibility of Generative Code

Alp K. Türedi

June 12, 2025

Supervisor: [TBD]

Tallinn University
aturedi@tlu.ee

Abstract

This thesis aims to identify and analyze accessibility barriers present in the outputs of generative AI tools for web development, specifically Claude Sonnet 3.5 model used by StackBlitz's Bolt open-source repository. The methodology involves compiling a set of prompts, encompassing both general and accessibility-focused instructions, and using these prompts to generate code repositories with the chosen AI tool and model. Subsequently, each generated repository will undergo accessibility evaluation using the WAVE tool. Finally, the study will investigate the capacity of the chosen model to automatically remediate identified accessibility issues and will report on the effectiveness of these attempts. The findings of this research will contribute to a deeper understanding of the current state of accessibility considerations within generative AI for web development and highlight areas for future improvement.

Keywords: Web accessibility, WCAG, Large Language Models, Generative Code

1 Introduction

The rapid advancement of generative artificial intelligence is transforming numerous domains, and web development is no exception. Tools like Vercel's v0, StackBlitz's Bolt and Anysphere's Cursor are empowering developers to generate web repositories with unprecedented speed and efficiency. However, this accelerated development raises critical questions about the accessibility of the resulting web content. Accessibility, which ensures that websites are usable by everyone, including individuals with disabilities, is paramount. This study investigates the accessibility implications of AI-generated web content, examining the common accessibility issues that arise, the AI tool's capacity for remediation, and the potential of prompt engineering to mitigate these issues.

2 Acronyms

LLM Large Language Model

SC Success Criteria

WAI Web Accessibility Initiative

WCAG Web Content Accessibility Guidelines

3 Problem Statement

While AI design tools like v0, Bolt and Cursor promise rapid UI generation, their ability to produce accessible interfaces for users with disabilities is largely unknown. This poses a potential risk of increasing accessibility barriers in digital products, hindering inclusivity and equitable access. Existing evaluation methods may not fully capture the nuanced challenges presented by AI-generated UIs, especially considering the variability and potential inconsistencies in their output.

3.1 Research Motivation and Purpose

My motivation is rooted in the belief that everyone deserves equal access to the digital world. The purpose of highlighting the accessibility risks associated with rapidly advancing generative AI in code creation is to prevent the unintentional creation of inaccessible web pages. This is crucial ethically, legally, for broader audience reach, and to ensure that technological progress fosters inclusivity rather than creating new barriers.

4 Literature Review

The investigation encompasses two key phases: site generation and accessibility assessment. The site generation phase will be evaluated based on the prompt selection methodology, the specific Large Language Model (LLM) employed, and any iterative prompting strategies implemented to address potential issues. The subsequent assessment phase will focus on the methodology used to evaluate the generated websites against the Web Content Accessibility Guidelines (WCAG).

A singular study employing a comparable methodology is that of Aljedaani et al. [2], who utilized ChatGPT for a related purpose. In their work, 88 developers generated websites, and ChatGPT was subsequently prompted to remediate these websites. Their findings indicated that a majority of the generated websites contained accessibility violations. However, ChatGPT demonstrated a 70% success rate in rectifying these violations within its own generated source code and a 73% success rate in addressing accessibility issues present in third-party open-source project code [2].

The reviewed studies on generative AI reveal two key approaches to prompt selection. The first involves a participant-driven methodology, where researchers recruit individuals and directly observe their interaction with the tools, capturing the prompts they naturally generate and employ. Conversely, the second approach relies on researcher-defined prompts, which are formulated without explicit reasoning and are presented as inherently obvious or logical choices for the intended task.

The second aspect of this study concerns assessment. WCAG 2.2 is a guidelines document and there needs to be a methodology to check those front-ends against those guidelines. There are three Success Criteria (SC) levels defined, level-A, AA, and AAA. Abu Doush et al. [1] conducted an evaluation of 34 automated web accessibility assessment tools, including WAVE, to determine their effectiveness in measuring WCAG compliance. Their results showed that "44% of all WCAG SC can be automatically checked offering conformance details directly from web content. [...] The other 43% of WCAG SC "all levels" cannot be easily checked automatically. This means that there is no direct mapping from SC to results using existing technologies and requires an expert to double-check. These include things like non-text content video, no keyboard trap, focus order (level A), headings and labels and error suggestions (level AA), timeouts, interruptions, and re-authentication (level AAA). The remaining 13% of WCAG 2.1 SC have no direct automatic way to be checked." [1]

Ara et al. [3] conducted a literature review of different existing solutions for web accessibility testing to identify their challenges and limitations. They reviewed different web accessibility assessing studies. They also talked about the limitations of algorithmic evaluations. Aligned with the previous study they also said "most of the accessibility testing tools only check a specific number of WCAG success criteria which is around 50% of the total guidelines." And they suggested user testing and expert testing for validations. [3]

Lastly López-Gil & Pereira [4] compared WAVE with ChatGPT for WCAG evaluation. In their study "the LLM-based scripts successfully identified accessibility issues that automatic accessibility evaluators missed or labelled as warnings, achieving an overall 87.18% detection across applicable test cases." This was to evaluate web accessibility success criteria. [4]

5 Research Goal and Research Questions

To evaluate the accessibility of user interfaces generated AI code generation tools by using Claude Sonnet through Bolt by StackBlitz. Identifying potential accessibility barriers and proposing recommendations for improving the inclusivity of these AI design tools.

5.1 Research Questions

1. *To what extent do the user interfaces generated by v0, Bolt and Cursor comply with established accessibility guidelines WCAG 2.2?*
2. *What are the common accessibility barriers present in the UIs generated by these AI design tools?*
3. *What specific recommendations can be provided to developers and designers to improve the accessibility of UIs generated by AI design tools?*

6 Conceptual or Theoretical Framework and Methodology

"Web Content Accessibility Guidelines (WCAG) 2.2 covers a wide range of recommendations for making web content more accessible. Following these guidelines will make content more accessible to a wider range of people with disabilities, including accommodations for blindness and low vision, deafness and hearing loss, limited movement, speech disabilities, photosensitivity, and combinations of these, and some accommodation for learning disabilities and cognitive limitations; but will not address every user need for people with these disabilities. These guidelines address accessibility of web content on any kind of device (including desktops, laptops, kiosks, and mobile devices). Following these guidelines will also often make web content more usable to users in general.

WCAG 2.2 success criteria are written as testable statements that are not technology-specific. Guidance about satisfying the success criteria in specific technologies, as well as general information about interpreting the success criteria, is provided in separate documents. See Web Content Accessibility Guidelines (WCAG) Overview for an introduction and links to WCAG technical and educational material." [6]

There are 45 automated listing evaluation tools for websites on WAI's website (Initiative, n.d.). Some of them are specific like WCAG Color Contrast Checker that checks only for specific items. Many of the studies use the WAVE Web Accessibility Evaluation Tool by WebAIM. WAVE gives a visual representation of the websites and highlights errors with yellow and on the sidebar gives the list of the problems. It has a Chromium plugin where you can initiate the evaluation from. It is up to date to access WCAG 2.2. [5]

7 Research Plan

7.1 Phase 1: Compiling Real-World Prompts for Accessibility Evaluation

This initial phase focuses on gathering authentic prompts that reflect how designers and developers would naturally interact with generative AI tools in their workflows. We will conduct a targeted workshop that will involve a diverse group of designers and developers with varying levels of experience. The workshop will center around specific, real-world design and development scenarios (e.g., generating a landing page section, creating a component library, implementing a data visualization). For each scenario, participants will be asked to formulate prompts as they would in a practical setting. Crucially, for each scenario, we will explicitly guide participants to create two distinct prompt sets: Standard Prompt: A prompt formulated without specific consideration for accessibility requirements. This will represent a baseline user interaction. Accessibility-Considered Prompt: A prompt explicitly incorporating accessibility-related keywords, constraints, or considerations (e.g., "create a form with clear labels and keyboard navigation," "generate an image carousel with alt text for screen readers").

This approach will allow for a direct comparison of the outputs generated from standard versus accessibility-aware prompting, providing valuable insights into the tools' inherent accessibility considerations and their responsiveness to explicit accessibility instructions. The selection of participants will aim for a balance of expertise and backgrounds to ensure a broad range of prompting styles and perspectives.

7.2 Phase 2: Generating Web Repositories with Target AI Tool

The second phase involves utilizing the curated prompts to generate repositories using bolt.new with Claude Sonnet 3.5 model. This method is chosen because bolt.new is an open-source repository

and Claude Sonnet 3.5 is a widely recognized LLM for code generation. We will systematically input each of the collected prompts (both standard and accessibility-considered variations for each scenario) into the tool. This process will result in a collection of generated web repositories, one with each a normal and an accessibility-focused prompt.

7.3 Phase 3: Accessibility Evaluation using WAVE Tool

The third phase will involve an accessibility audit of all the repositories generated. We will employ the WAVE (Web Accessibility Evaluation Tool) browser extension, a widely recognized tool for identifying accessibility issues based on the Web Content Accessibility Guidelines (WCAG). For each generated website, a WAVE report will be generated. This report will detail all instances where WCAG guidelines are violated, categorizing the severity and nature of the accessibility barriers present (e.g., missing alternative text for images, insufficient color contrast, lack of keyboard navigation). This evaluation will provide an understanding of the inherent accessibility of the outputs produced by the AI tool under different prompting conditions. The identified violations will be documented, noting the specific WCAG guideline(s) violated and the context within the generated code or UI.

7.4 Phase 4: Assessing LLM Remediation Capabilities

The final phase aims to investigate the ability of llm to address the accessibility issues identified in the previous stage. We will take the specific accessibility violations flagged by the WAVE tool and formulate targeted prompts instructing the AI tool to rectify these issues. For example, if an image is flagged as missing alternative text, we will prompt the tool to "add descriptive alt text to the image." Similarly, for color contrast issues, we will prompt for adjustments to meet WCAG contrast ratio requirements. By feeding these problem-specific prompts back into the target AI tool, we can assess their capacity to understand and implement accessibility fixes. This phase will provide critical insights into the current state of AI-driven accessibility remediation, highlighting both the potential and the limitations of these tools in automatically addressing accessibility barriers in generated web content. The success and nature of the fixes implemented by each tool will be analyzed and compared.

8 Expected Outcomes

The study is expected to produce a catalog of accessibility violations and an analysis of the AI's remediation success. The research also aims to provide recommendations for prompt selection to minimize accessibility barriers, demonstrating -if there is any- the impact of accessibility-focused prompting.

References

[1] Abu Doush, I., Sultan, K., Al-Betar, M. A., Almeraj, Z., Alyasseri, Z. A. A., and Awadallah, M. A. Web accessibility automatic evaluation tools: to what extent can they be automated? *CCF Transactions on Pervasive Computing and Interaction 5*, 3 (Sept. 2023), 288–320. Company: Springer Distributor: Springer Institution: Springer Label: Springer Number: 3 Publisher: Springer Nature Singapore.

[2] Aljedaani, W., Habib, A., Aljohani, A., Eler, M., and Feng, Y. Does ChatGPT Generate Accessible Code? Investigating Accessibility Challenges in LLM-Generated Source Code. In *Proceedings of the 21st International Web for All Conference* (New York, NY, USA, Oct. 2024), W4A '24, Association for Computing Machinery, pp. 165–176.

[3] Ara, J., Sik-Lanyi, C., Kelemen, A., and Guzsvinecz, T. An inclusive framework for automated web content accessibility evaluation. *Universal Access in the Information Society* (Oct. 2024), 1–27. Company: Springer Distributor: Springer Institution: Springer Label: Springer Publisher: Springer Berlin Heidelberg.

[4] López-Gil, J.-M., and Pereira, J. Turning manual web accessibility success criteria into automatic: an LLM-based approach. *Universal Access in the Information Society 24*, 1 (Mar. 2025), 837–852. Company: Springer Distributor: Springer Institution: Springer Label: Springer Number: 1 Publisher: Springer Berlin Heidelberg.

[5] WebAIM. Wave Web Accessibility Evaluation Tool. https://wave.webaim.org/, 2024. Accessed: 2025-05-08.

[6] World Wide Web Consortium (W3C). Web Content Accessibility Guidelines (WCAG) 2.2. https://www.w3.org/TR/WCAG22/, 2024. W3C Recommendation.