



Renan Paixão de Lemos

# **Acessibilidade: Analisando a aptidão do LLM Claude em melhorar a inclusão em páginas web**

Recife

2025

Renan Paixão de Lemos

## **Acessibilidade: Analisando a aptidão do LLM Claude em melhorar a inclusão em páginas web**

Trabalho de conclusão de curso apresentado ao Curso de Bacharelado em Ciências da Computação da Universidade Federal Rural de Pernambuco, como requisito parcial para obtenção do título de Bacharel em Ciências da Computação.

Universidade Federal Rural de Pernambuco – UFRPE

Departamento de Computação

Curso de Bacharelado em Ciências da Computação

Orientador: Paulo Anselmo da Mota Silveira Neto

Recife

2025

Dados Internacionais de Catalogação na Publicação  
Sistema Integrado de Bibliotecas da UFRPE  
Bibliotecário(a): Suely Manzi – CRB-4 809

L555a Lemos, Renan Paixão de.  
Acessibilidade: analisando a aptidão do LLM Claude em melhorar a inclusão em páginas web / Renan Paixão de Lemos. – Recife, 2025.  
34 f.; il.

Orientador(a): Paulo Anselmo da Mota Silveira Neto.

Trabalho de Conclusão de Curso (Graduação) – Universidade Federal Rural de Pernambuco, Bacharelado em Ciência da Computação, Recife, BR-PE, 2025.

Inclui referências.

1. WEB (Linguagem de programação). 2. Linguagem de programação (Computadores). 3. Educação inclusiva. 4. Tecnologia educacional 5. Inteligência artificial . I. Silveira Neto, Paulo Anselmo da Mota, orient. II. Título

CDD 004

Renan Paixão de Lemos

## **Acessibilidade: Analisando a aptidão do LLM Claude em melhorar a inclusão em páginas web**

Trabalho de conclusão de curso apresentado ao Curso de Bacharelado em Ciências da Computação da Universidade Federal Rural de Pernambuco, como requisito parcial para obtenção do título de Bacharel em Ciências da Computação.

Aprovado em: 17 de dezembro de 2025.

### **BANCA EXAMINADORA**

---

**Prof. Dr. Paulo Anselmo da Mota  
Silveira Neto**  
Orientador  
Universidade Federal Rural de  
Pernambuco (UFRPE)

---

**Prof. Dr. Walter Felipe Dos Santos**  
Universidade Federal do Vale do São  
Francisco (UNIVASF)

# Agradecimentos

Agradeço aos meus pais, por todo o incentivo e apoio em cada decisão ao longo dessa jornada. À minha esposa, que esteve ao meu lado desde o início. Sua contribuição e presença tornaram este momento possível.

Aos professores, que direta e indiretamente me fizeram crescer como pessoa e profissional de forma exponencial.

Por fim, aos amigos que caminharam comigo nessa jornada, em especial ao Eduardo, o qual a amizade e companheirismo tornaram essa caminhada muito mais leve.

# Resumo

Atualmente, a acessibilidade web é um desafio constante, com milhões de páginas falhando em atender as diretrizes Web Content Accessibility Guidelines ([WCAG](#)). Este estudo investiga a capacidade do Claude Code, um modelo de linguagem em larga escala desenvolvido pela Anthropic — empresa fundada por ex-integrantes da OpenAI — de identificar e corrigir problemas de acessibilidade no desenvolvimento web. Três experimentos foram conduzidos: o primeiro testou 53 critérios WCAG dos níveis A e AA, revelando altas taxas de sucesso para problemas estruturais, visuais e de navegação, mas claras limitações com conteúdo de áudio e vídeo. No segundo, uma página foi construída do zero utilizando diretrizes de acessibilidade, alcançando zero erros em uma análise usando a ferramenta Wave após ajustes mínimos. Já no terceiro e último, uma análise das páginas iniciais de 27 universidades brasileiras revelou 1.128 erros de acessibilidade, grande parte sendo de contraste (96% das instituições) e rótulos ausentes (74%). O cruzamento desses dados mostrou que a maioria dos erros encontrados nos sites das universidades seriam possivelmente corrigidos com o uso do Claude Code. Os resultados sugerem que ferramentas baseadas em Modelo de linguagem de grande escala ([LLM](#)) podem ajudar a preencher a lacuna de conhecimento em acessibilidade entre desenvolvedores, contribuindo para experiências web mais inclusivas.

**Palavras-chave:** acessibilidade web, WCAG, modelos de linguagem de grande escala, Claude Code, LLM, tecnologia assistiva.

# Abstract

Currently, web accessibility is a constant challenge, with millions of pages failing to meet [WCAG](#) guidelines. This study investigates the ability of Claude Code, a large-scale language model developed by Anthropic — a company founded by former OpenAI members — to identify and correct accessibility issues in web development. Three experiments were conducted: the first tested 53 WCAG criteria at levels A and AA, revealing high success rates for structural, visual, and navigation issues, but clear limitations with audio and video content. In the second, a page was built from scratch using accessibility guidelines, achieving zero errors in an analysis using the Wave tool after minimal adjustments. In the third and last, an analysis of 27 Brazilian university homepages uncovered 1,128 accessibility errors, largely contrast issues (96% of institutions) and missing labels (74%). Cross-referencing these findings showed that most errors found in university websites would likely be fixed with the use of Claude Code. The results suggest that [LLM](#)-based tools can help bridge the accessibility knowledge gap among developers, contributing to more inclusive web experiences.

**Keywords:** web accessibility, WCAG, large language models, Claude Code, LLM, assistive technology.

# Lista de ilustrações

Figura 1 – Fluxograma do processo de iteração para correção de problemas de acessibilidade . . . . .	21
Figura 2 – Fluxograma do processo de criação da página acessível . . . . .	27
Figura 3 – Página criada pelo Claude Code dividida em duas partes . . . . .	28
Figura 4 – Página criada pelo Claude code sem erros após ajustes . . . . .	30

# Lista de tabelas

Tabela 1 – Resultados do Princípio 1 — Perceptível . . . . .	22
Tabela 2 – Resultados do Princípio 2 — Operável . . . . .	23
Tabela 3 – Resultados do Princípio 3 — Entendível . . . . .	24
Tabela 4 – Resultados do Princípio 4 — Robusto . . . . .	25
Tabela 5 – Correções de acessibilidade realizadas pelo Claude Code na landing page . . . . .	30
Tabela 6 – Quantitativo de erros de acessibilidade e diversidade de tipos em websites de universidades brasileiras . . . . .	33
Tabela 7 – Erros de acessibilidade mais frequentes nas universidades analisadas	33
Tabela 8 – Paralelo entre erros nas universidades e correções realizadas pelo Claude Code . . . . .	34

# Lista de abreviaturas e siglas

**WCAG** Web Content Accessibility Guidelines

**LLM** Modelo de linguagem de grande escala

**HTML** Linguagem de Marcação de Hipertexto

**eMAG** Modelo de Acessibilidade em Governo Eletrônico

**W3C** World Wide Web Consortium

**ADA** Americans with Disabilities Act

**ACA** Accessible Canada Act

**SIGAA** Sistema Integrado de Gestão de Atividades Acadêmicas

# Sumário

	<b>Lista de ilustrações</b> . . . . .	<b>6</b>
<b>1</b>	<b>INTRODUÇÃO</b> . . . . .	<b>10</b>
<b>1.1</b>	<b>Objetivos</b> . . . . .	<b>11</b>
1.1.1	Objetivo Geral . . . . .	11
1.1.2	Objetivos Específicos . . . . .	11
<b>1.2</b>	<b>Estrutura do Trabalho</b> . . . . .	<b>11</b>
<b>2</b>	<b>FUNDAMENTAÇÃO TEÓRICA</b> . . . . .	<b>12</b>
<b>2.1</b>	<b>Acessibilidade na Web</b> . . . . .	<b>12</b>
<b>2.2</b>	<b>Diretrizes adotadas no Brasil e no mundo</b> . . . . .	<b>12</b>
<b>2.3</b>	<b>Desenvolvimento de software, inteligência artificial generativa e acessibilidade</b> . . . . .	<b>14</b>
<b>2.4</b>	<b>Modelos de linguagem de grande escala - LLMs</b> . . . . .	<b>14</b>
<b>2.5</b>	<b>Trabalhos relacionados</b> . . . . .	<b>15</b>
2.5.1	Quadro comparativo . . . . .	16
2.5.2	Originalidade e contribuição deste trabalho . . . . .	17
<b>3</b>	<b>MÉTODO</b> . . . . .	<b>18</b>
<b>3.1</b>	<b>Enquadramento metodológico</b> . . . . .	<b>18</b>
3.1.1	Abordagem da pesquisa . . . . .	18
3.1.2	Procedimentos . . . . .	18
3.1.3	Ferramentas utilizadas . . . . .	18
<b>3.2</b>	<b>Utilização do Claude code para resolver inconformidade com critérios da WCAG</b> . . . . .	<b>19</b>
3.2.1	<b>1. Perceptível</b> . . . . .	22
3.2.2	<b>2. Operável</b> . . . . .	23
3.2.3	<b>3. Entendível</b> . . . . .	24
3.2.4	<b>4. Robusto</b> . . . . .	25
<b>3.3</b>	<b>Capacidade do Claude Code em gerar código acessível</b> . . . . .	<b>25</b>
<b>3.4</b>	<b>Analisando páginas iniciais de Universidades pelo Brasil</b> . . . . .	<b>31</b>
<b>4</b>	<b>CONCLUSÃO</b> . . . . .	<b>36</b>
	<b>REFERÊNCIAS</b> . . . . .	<b>38</b>

# 1 Introdução

Globalmente, mais de 1,3 bilhões de pessoas vivem com alguma forma de deficiência, o que representa aproximadamente uma em cada seis pessoas no mundo (WHO, 2023). No Brasil, esse número chega a 18,6 milhões (FEDERAL, 2025). Com o avanço da digitalização, a inacessibilidade em páginas web se torna um fator de exclusão social para uma parcela significativa da população (DIGITAL, 2014).

Diretrizes como a WCAG, desenvolvida pelo World Wide Web Consortium (W3C), e o Modelo de Acessibilidade em Governo Eletrônico (eMAG), versão brasileira adaptada às necessidades nacionais, existem para orientar o desenvolvimento de conteúdo acessível (CONSORTIUM, 2023; DIGITAL, 2014). Apesar disso, dados do WebAIM Million revelam que, em 2025, páginas web apresentam em média 51 erros de acessibilidade (WEBAIM, 2024), evidenciando que as diretrizes existentes não estão sendo amplamente seguidas.

Estudos indicam que a acessibilidade raramente é considerada nas etapas iniciais do desenvolvimento de software, sendo frequentemente tratada como requisito secundário (BI et al., 2022; BI et al., 2021). Entre as causas apontadas estão a complexidade das diretrizes, a falta de padrões educacionais na indústria e a percepção de custos adicionais (MOWAR, 2024; GURIŁŁ; VATAVU, 2025).

Paralelamente, a adoção de ferramentas de inteligência artificial para auxiliar o desenvolvimento de software cresceu expressivamente, sendo utilizadas por cerca de 85% dos desenvolvedores (JETBRAINS, 2025; OVERFLOW, 2025). Modelos de linguagem de grande escala (LLMs) como o Claude, considerado o mais admirado pelos desenvolvedores em 2025 Overflow (2025), demonstram capacidade de gerar código funcional em diversas linguagens (SOBO et al., 2025). O Claude Code, versão integrada do Claude ao terminal, é capaz de compreender projetos inteiros e gerar código contextualizado (ANTHROPIC, 2025b).

Diante desse cenário, surge a oportunidade de investigar se ferramentas baseadas em LLMs, especificamente o Claude Code dada sua grande adoção e admiração por desenvolvedores Overflow (2025), podem contribuir para a melhoria da acessibilidade web, suprimindo a lacuna de conhecimento especializado entre desenvolvedores (ANDRUCCIOLI et al., 2025; GURIŁŁ; VATAVU, 2025).

## 1.1 Objetivos

### 1.1.1 Objetivo Geral

Analisar a capacidade do Claude Code em identificar e corrigir problemas de acessibilidade web baseados nas diretrizes [WCAG](#).

### 1.1.2 Objetivos Específicos

1. Avaliar a efetividade do Claude Code na correção de problemas de acessibilidade web;
2. Identificar quais critérios da [WCAG](#) apresentam maior dificuldade de correção pelo Claude Code;
3. Analisar a capacidade do Claude Code em corrigir os principais problemas de acessibilidade encontrados em sites de universidades brasileiras.

## 1.2 Estrutura do Trabalho

Este trabalho está organizado da seguinte forma: o Capítulo 2 apresenta a fundamentação teórica, abordando conceitos de acessibilidade web, diretrizes [WCAG](#) e [eMAG](#), além de uma visão geral sobre [LLMs](#) e trabalhos relacionados. O Capítulo 3 fala sobre a metodologia adotada e apresenta os resultados dos experimentos realizados. Por fim, o Capítulo 4 traz as conclusões e sugestões para trabalhos futuros.

## 2 Fundamentação teórica

### 2.1 Acessibilidade na Web

Acessibilidade na web se refere ao design, desenvolvimento e produtos que garantem que pessoas com diferentes habilidades e deficiências possam interagir, entender, navegar e contribuir para o meio digital (WEGENER et al., 2024). O objetivo é simples e direto, fazer com que a maior quantidade de pessoas, independente de qualquer limitação seja ela permanente, temporária ou situacional, possam usar tecnologia (WEGENER et al., 2024; BRITTO; PIZZOLATO, 2021).

Globalmente, mais de 1,3 bilhões de pessoas vivem com alguma forma de deficiência, proporcionalmente, 1 a cada 6 pessoas. Esse número tende a crescer nos próximos anos, devido ao envelhecimento da população e ao aumento na prevalência de doenças não transmissíveis WHO (2023). No Brasil esse número chega a 18.6 milhões de pessoas. Federal (2025)

Com isso em vista, a inacessibilidade em sites na web exclui uma parcela significativa da população do acesso às informações veiculadas na internet Digital (2014). Com tamanha relevância, o governo brasileiro investe no uso adequado e coordenado da tecnologia, pois compreende que a inclusão digital é o caminho para a inclusão social Digital (2014).

### 2.2 Diretrizes adotadas no Brasil e no mundo

Leis e medidas para viabilizar o meio digital para o maior número de pessoas já existem. O maior exemplo é a WCAG, um conjunto de diretrizes que visam tornar o mundo online acessível (CONSORTIUM, 2023). Em âmbito nacional temos o eMAG, uma versão especializada da WCAG voltada as necessidades do nosso país, que tem como objetivo nortear o desenvolvimento e a adaptação de conteúdos do governo federal, garantindo acesso a todos (DIGITAL, 2014).

A WCAG é o padrão mais amplamente adotado pelo mundo quando se fala em acessibilidade na web. Desenvolvida pelo W3C, teve sua primeira versão lançada em 1999 (WAI) (1999). Desde então, novas versões são indicadas sempre que publicadas como padrões (CONSORTIUM, 2023). A mais recente versão é a 2.2, publicada em 5 de outubro de 2023 e tendo a última atualização em 12 de dezembro de 2024 (CONSORTIUM, 2023).

Suas diretrizes se baseiam e são organizadas em quatro princípios ([Consortium \(2023\)](#), [WAI \(2023b\)](#)):

### 1. **Perceptível**

As informações e os componentes da interface do usuário devem ser apresentados aos usuários de maneiras que eles possam perceber.

### 2. **Operável**

Os componentes da interface e a navegação devem permitir que os usuários os operem sem barreiras.

### 3. **Entendível**

Os usuários devem conseguir entender claramente tanto as informações quanto como operar a interface.

### 4. **Robusto**

O conteúdo deve ser robusto o suficiente para ser interpretado de forma confiável por uma grande variedade de agentes de usuário, incluindo tecnologias assistivas.

Dentro dos princípios, existem os critérios, que são disponibilizados para possibilitar os testes de acessibilidade onde forem necessários.

Para abranger um grande número de grupos e diferentes situações, três níveis de conformidade foram definidos: **A**, **AA** e **AAA** ([CONSORTIUM, 2023](#); [WAI, 2023a](#)).

Uma página estar em conformidade com um desses níveis significa que ([WAI, 2023a](#)):

- Para o nível **A**, a página atende a todos os critérios mínimos.
- Para o **AA**, a página atende a todo o nível **A** e também aos critérios **AA**.
- Similar para o nível **AAA**, a página atende a todos os critérios do nível **A**, **AA** e **AAA**.

Existem ainda leis como a Americans with Disabilities Act ([ADA](#)) e Accessible Canada Act ([ACA](#)), que propõem um prazo limite para que as diretrizes de acessibilidade sejam atendidas em âmbito nacional [Justice \(2024\)](#), [Canada \(2019\)](#).

## 2.3 Desenvolvimento de software, inteligência artificial generativa e acessibilidade

Na maioria dos casos, a engenharia de software não tem acessibilidade como primeiro plano. Tais tipos de problema normalmente são considerados apenas em estágios avançados da esteira de desenvolvimento (BI et al., 2022).

É válido mencionar que dificuldades em incorporar acessibilidade nos projetos também se tornam presentes, devido a necessidade de conhecimentos específicos sobre deficiências e padrões de acessibilidade (BI et al., 2022).

Isso se mostra refletido nos números coletados pelo WEB-AIM Million (WebAIM (2024)), um relatório que usa a ferramenta de acessibilidade WAVE (WebAIM (2025b)) para avaliar um milhão de páginas iniciais pela internet.

Os dados levantados mostram que, em 2025, foram detectados uma média de 51 erros por página WebAIM (2024).

Baseado na grande adoção de ferramentas de inteligência artificial para desenvolvimento de software, cerca de 85% dos desenvolvedores JetBrains (2025), Overflow (2025), é notável o potencial de uso dessas ferramentas para suprir a lacuna de conhecimentos específicos requeridos dos desenvolvedores e assim, melhorar a acessibilidade na web.

## 2.4 Modelos de linguagem de grande escala - LLMs

Os modelos de linguagem de grande escala representam uma mudança de paradigma no processamento de linguagem natural, migrando de modelos projetados para tarefas específicas para sistemas mais diversificados (ANDRUCCIOLI et al., 2025; SOBO et al., 2025).

São treinados com uma imensa quantidade de dados em texto, que incluem código e discussões relacionadas à programação. Essa exposição permite que os modelos aprendam os padrões e estruturas de várias linguagens de programação (SOBO et al., 2025; STRYKER; IBM, 2025). Assim, quando lhes é apresentada uma instrução ou mesmo parte de um código, o modelo consegue prever continuações mais prováveis baseado nos padrões que observou no treinamento. A partir disto, a capacidade de geração de código é estabelecida (SOBO et al., 2025).

Dentre os modelos de linguagem da atualidade, o LLM Claude se destaca entre os desenvolvedores, sendo o mais usado atualmente entre profissionais Overflow (2025). Foi criado pela empresa Anthropic, uma startup voltada para pesquisas em inteligência artificial fundada por ex-membros da OpenAI, a pioneira no mundo das IAs

generativas com o criação do ChatGPT [Editors \(2023\)](#).

O modelo tem grande potencial de democratizar o desenvolvimento web para pessoas com pouco ou nenhum conhecimento em programação ([GURI et al.; VATAVU, 2025](#)).

Em uma pesquisa recente, o Claude se destacou diante de outros grandes LLMs do mercado como ChatGPT e Gemini, demonstrando uma taxa de sucesso de 95% na geração de código Java funcional para tarefas de robótica ([SOBO et al., 2025](#)), o que indica grande capacidade técnica do LLM.

O Claude dispõe de uma versão que pode ser integrada diretamente em qualquer terminal (aplicativo de linha de comando), o Claude Code ([ANTHROPIC, 2025b](#)). A ferramenta é otimizada especificamente para entender e gerar código usando o modelo Sonnet 4.5, também desenvolvido pela Anthropic. É capaz de usar "busca agêntica", possibilitando entender todo o projeto em que está inserido [Anthropic \(2025b\)](#).

## 2.5 Trabalhos relacionados

Diversos estudos e testes foram realizados a fim de entender o estado atual da acessibilidade web, bem como ferramentas capazes de avaliar e corrigir problemas nesse aspecto. Um dos levantamentos mais abrangentes é o *WebAIM Million* ([WEBAIM, 2024](#)), onde os resultados demonstram de forma consistente ao longo dos anos grande falta de adequação à [WCAG](#), com erros principalmente relacionados à contraste, rótulos e textos alternativos.

Um estudo investigou a acessibilidade no Sistema Integrado de Gestão de Atividades Acadêmicas ([SIGAA](#)), software utilizado por diversas instituições de ensino no Brasil. Este estudo foi voltado a usuários com deficiência visual, e o mesmo constatou a não conformidade com as diretrizes [WCAG](#) e [eMAG](#) ([CORREA; VITORIANO; LLANOS, 2025](#)). As descobertas evidenciaram a falta de conhecimento dos profissionais responsáveis sobre a prática de acessibilidade de software em universidades públicas ([CORREA; VITORIANO; LLANOS, 2025](#)).

Outros estudos investigaram as causas da limitação da implementação da acessibilidade por parte dos desenvolvedores ([BI et al., 2021](#)). Esses estudos demonstraram que a acessibilidade geralmente é tratada tardiamente em projetos, devido a falta de padrões educacionais na indústria, complexidade das diretrizes e a percepção de custos e esforços adicionais. ([MOWAR, 2024; GURI et al.; VATAVU, 2025; BI et al., 2021](#))

Ainda, estudos focaram em avaliar o desempenho de LLMs para a validação de acessibilidade, buscando diminuir as barreiras existentes. Tais estudos sugerem que os LLMs podem ajudar desenvolvedores no cotidiano, mas que limitações que

incluem baixa confiabilidade na avaliação dos modelos e geração de falsos positivos se fazem presentes. (ANDRUCCIOLI et al., 2025; BRITTO; PIZZOLATO, 2021; GURIÃO; VATAVU, 2025)

Por fim, documentos como a WCAG e o eMAG brasileiro, bem como leis internacionais como a ADA e ACA fornecem uma base para guiar soluções automatizadas. Tais diretrizes são essenciais para analisar ferramentas modernas, incluindo os LLMs, no que se diz respeito à atender os critérios esperados.

### 2.5.1 Quadro comparativo

Esta subseção apresenta uma síntese comparativa dos trabalhos relacionados. Podemos observar que a maioria dos estudos foca na identificação de problemas de acessibilidade, seja por meio de avaliações de sistemas específicos (CORREA; VITORIANO; LLANOS, 2025) ou investigações sobre as barreiras enfrentadas por desenvolvedores (BI et al., 2021; BI et al., 2022).

Trabalho	Foco	Metodologia	Uso de LLM	Validação Prática
(CORREA; VITORIANO; LLANOS, 2025)	Acessibilidade no SIGAA para deficientes visuais	Avaliação com WCAG e eMAG	Não	Identificação de problemas específicos
(BI et al., 2021)	Perspectiva dos desenvolvedores sobre acessibilidade	Entrevistas e surveys com 365 profissionais	Não	Análise qualitativa
(MOWAR, 2024)	Impacto de assistentes de IA no código acessível	Estudo com desenvolvedores usando AI	Sim (Copilot)	Código gerado por desenvolvedores
(GURIÃO; VATAVU, 2025)	Barreiras de acessibilidade em código gerado por LLM	Análise de código gerado	Sim (diversos)	Identificação de problemas
(ANDRUCCIOLI et al., 2025)	LLMs para validação de acessibilidade	Análise de código-fonte	Sim (diversos)	Validação, não correção
<b>Este trabalho</b>	<b>Correção de acessibilidade via Claude Code</b>	<b>Experimentos com 53 critérios WCAG + páginas reais</b>	<b>Sim (Claude Code)</b>	<b>Correção validada + criação de página acessível</b>

Fonte: Elaborado pelo autor (2025)

### 2.5.2 Originalidade e contribuição deste trabalho

Os trabalhos mais recentes sobre LLMs se concentram principalmente em avaliar se esses modelos perpetuam barreiras de acessibilidade (GURIŁŁ; VATAVU, 2025) ou se podem antecipar violações durante a validação de código (ANDRUCCIOLI et al., 2025). No entanto, poucos estudos investigam a capacidade dos LLMs de efetivamente corrigir problemas de acessibilidade. Este trabalho preenche essa lacuna ao avaliar a capacidade do Claude Code de corrigir 53 critérios da WCAG níveis A e AA, com diferentes níveis de dificuldade, demonstrando se a ferramenta é capaz de corrigir os critérios avaliados.

## 3 Método

Este trabalho tem por objetivo analisar o potencial da [LLM Claude code](#) no que se diz respeito a correção de problemas de acessibilidade baseados nas diretrizes [WCAG](#), bem como se o mesmo é capaz de não introduzir problemas junto com a solução.

### 3.1 Enquadramento metodológico

Esta seção apresenta a estrutura metodológica adotada para investigar a capacidade do Claude Code em identificar e corrigir problemas de acessibilidade web.

#### 3.1.1 Abordagem da pesquisa

Este trabalho adota uma abordagem mista, qualitativa e quantitativa. A parte quantitativa está na contagem de iterações necessárias para correção, taxas de sucesso por critério e número de erros encontrados em sites reais. Já a qualitativa está presente na análise do comportamento do modelo frente a diferentes tipos de problemas e na avaliação da qualidade das correções realizadas.

#### 3.1.2 Procedimentos

O estudo foi conduzido em três experimentos complementares:

1. **Avaliação critério a critério:** Testes controlados com 53 critérios da [WCAG](#) níveis A e AA, utilizando um repositório próprio com exemplos de falhas;
2. **Criação de página acessível:** Geração de uma landing page completa pelo Claude Code, com validação posterior pela ferramenta WAVE;
3. **Auditoria de sites reais:** Análise de 27 páginas iniciais de universidades federais brasileiras, ligando os erros encontrados com os resultados do primeiro experimento.

#### 3.1.3 Ferramentas utilizadas

- **Claude Code:** Ferramenta de linha de comando da Anthropic, utilizando o modelo Sonnet 4.5;

- **WAVE:** Ferramenta de avaliação de acessibilidade web da WebAIM, organização responsável pelo relatório ([WEBAIM, 2024](#)), baseada na [WCAG 2.2](#);
- **Repositório de testes:** Repositório próprio contendo exemplos de falhas para cada critério avaliado.

A seguir, o enquadramento metodológico deste trabalho está representado:

Elemento	Descrição
Abordagem	Mista
Tipo de pesquisa	Exploratória e experimental
Objeto de estudo	Capacidade do Claude Code em corrigir problemas de acessibilidade web
Dados analisados	53 critérios WCAG (A e AA) + 27 sites de universidades
Instrumentos de coleta	Repositório de testes, ferramenta WAVE
Técnica de análise	Análise de taxa de sucesso, contagem de iterações e regressões

Fonte: Elaborado pelo autor (2025)

## 3.2 Utilização do Claude code para resolver inconformidade com critérios da [WCAG](#)

Para a análise, foi utilizado um repositório ([Paixão \(2025b\)](#)) com todos os critérios com nível de conformidade A e AA presentes na [WCAG](#), exceto o critério 1.2.4 Captions (live). Este critério se refere ao uso de legendas em conteúdos ao vivo. O motivo foi a ausência de formas adequadas para replicar o problema.

A escolha dos níveis A e AA se justifica por serem os níveis de conformidade mais amplamente adotados e exigidos por legislações de acessibilidade ao redor do mundo, como a [ADA](#) nos Estados Unidos e [ACA](#) no Canadá. O nível AAA é desejável, mas não é recomendado como meta geral, pois nem sempre é possível satisfazer todos os critérios desse nível para alguns tipos de conteúdo ([WAI, 2023a](#)).

Ao todo, 53 critérios de 86 existentes foram considerados neste estudo. A maioria dos exemplos para testes foram retirados do guia rápido para como atender às diretrizes da [WCAG WAI \(2024\)](#). Quando um exemplo de falha não estava disponível, um novo foi criado com base na descrição do problema presente no próprio guia.

O experimento foi realizado da seguinte forma:

Baseado nas iterações com o Claude code, foram definidos níveis de dificuldade para cada uma das soluções encontradas. Cada iteração adiciona mais contexto sobre o problema em específico.

- **Iteração 1 - Nível fácil (sem contexto sobre o problema)**

Caso este arquivo contenha problemas de acessibilidade, corrija-os.

- **Iteração 2 - Nível intermediário (prover id do critério)**

O problema presente nesse arquivo é relacionado ao critério 1.3.1 da [WCAG](#)

- **Iteração 3 - Nível difícil (prover exemplos)**

Demonstração de exemplo com contexto sobre o problema:

O exemplo a seguir mostra como um ponto de referência genérico de "região" pode ser adicionado a um elemento. Não há texto na página que possa ser referenciado como rótulo, então ele é rotulado com `aria-label`.

```
<div role="region" aria-label="weather portlet">  
  ...  
</div>
```

- **Iteração 4 ou mais - Nível Muito difícil (iterações com ajustes)**

Providenciar feedbacks de acordo com os últimos ajustes do Claude code, a fim de solucionar o problema baseado em constantes feedbacks sobre o estado atual do código após cada nova iteração. Por Exemplo:

**Usuário:** Envia um trecho de código em que um certo critério falha

**Assistente:** Retorna o código com modificações relacionadas a outro critério, mas não o esperado pelo usuário

**Usuário:** Usuário pede a correção, indicando o tipo problema que está acontecendo, podendo inclusive mencionar o critério em questão

**Assistente:** Tenta resolver, mas não consegue resolver todas as ocorrências do problema.

**Usuário:** Envia novos direcionamentos, baseado nas mudanças mais recentes retornadas pelo [LLM](#)...

A Figura 1 resume visualmente o processo de iteração descrito acima.

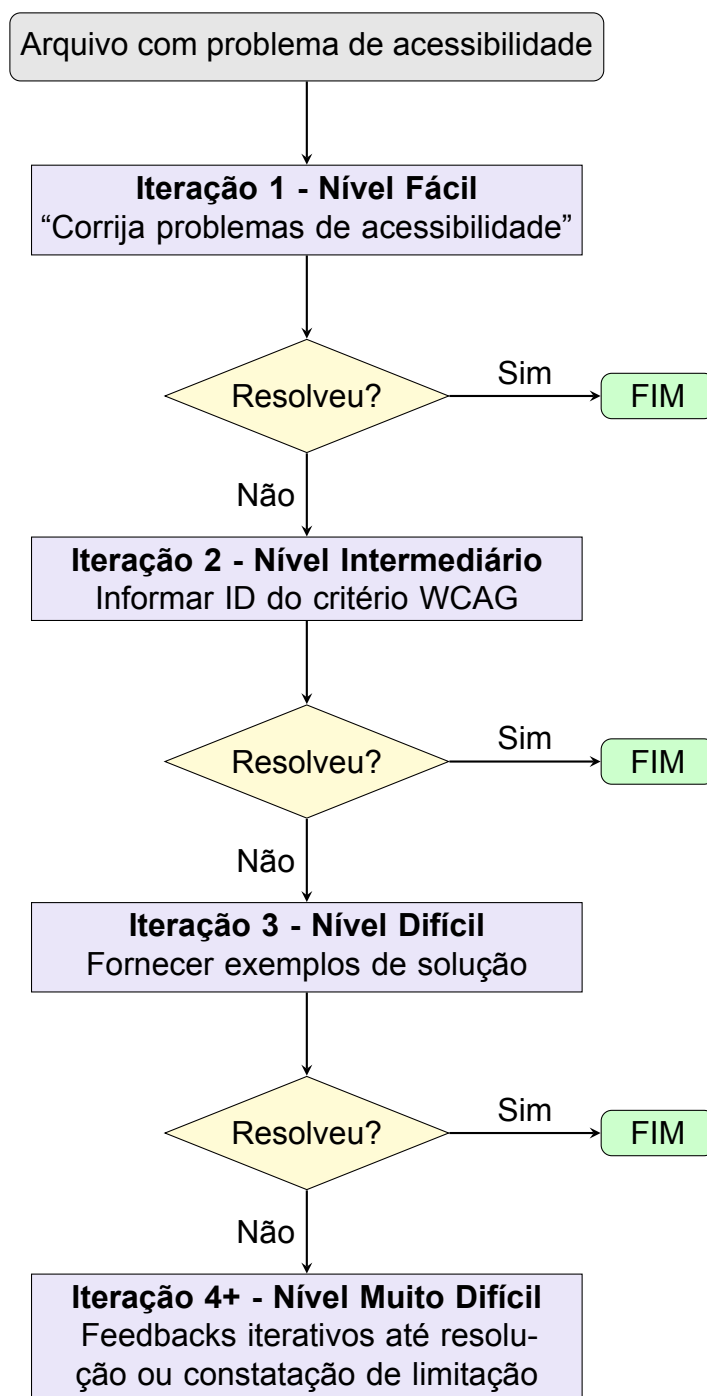


Figura 1 – Fluxograma do processo de iteração para correção de problemas de acessibilidade

Sobre o Claude Code, o modelo usado para os testes foi o Sonnet 4.5. Todos os testes foram reproduzidos manualmente, sem o auxílio de ferramentas para avaliação de acessibilidade.

Para facilitar a análise, os resultados foram divididos em uma tabela por grupo de princípios da WCAG 2.2: Perceptível (1.x.x), Operável (2.x.x), Entendível (3.x.x) e

Robusto (4.x.x).

Cada tabela mostra os critérios de cada princípio, junto a um veredito, que pode variar entre "Corrigiu", "Corrigiu parcialmente" e "Não corrigiu", além do número de iterações necessárias e possíveis problemas introduzidos durante as correções.

### 3.2.1 1. Perceptível

<b>Critério/Nome do arquivo</b>	<b>Iterações</b>	<b>Veredito</b>	<b>Introduziu novos problemas?</b>
1.1.1-non-text-content	1	Corrigiu totalmente	Não
1.2.1-audio-only-and-video-only	4+	Não corrigiu	Não
1.2.2-audio-video-only	4+	Não corrigiu	Sim
1.2.3-audio-description	4+	Não corrigiu	Sim
1.2.5-audio-description-prerecorded	4+	Não corrigiu	Sim
1.3.1-info-and-relationships	4+	Corrigiu parcialmente	Não
1.3.2-meaningful-sequence	1	Corrigiu totalmente	Não
1.3.3-sensory-characteristics	3	Corrigiu totalmente	Não
1.3.4-orientation	2	Corrigiu totalmente	Não
1.3.5-input-purpose	1	Corrigiu totalmente	Não
1.4.1-use-of-color	1	Corrigiu totalmente	Não
1.4.2-audio-control	1	Corrigiu totalmente	Não
1.4.3-contrast-minimum	1	Corrigiu totalmente	Não
1.4.4-resize-text	1	Corrigiu totalmente	Não
1.4.5-images-of-text	2	Corrigiu parcialmente	Não
1.4.10-reflow	1	Corrigiu totalmente	Não
1.4.11-non-text-contrast	2	Corrigiu totalmente	Sim
1.4.12-text-spacing	1	Corrigiu totalmente	Não
1.4.13-content-on-hover-or-focus	4+	Corrigiu parcialmente	Não

Tabela 1 – Resultados do Princípio 1 — Perceptível

Como podemos observar, a grande maioria dos critérios foram resolvidos com apenas uma iteração, especialmente os relacionados a estrutura de página, uso de cor e contraste. Isso indica que o modelo lida bem com problemas mais diretos, relacionados a estrutura do Linguagem de Marcação de Hipertexto ([HTML](#)).

Porém, em critérios relacionados a mídia, como áudio, vídeo e imagens, o modelo apresentou dificuldades ou total incapacidade de resolver esse tipo de problema,

como em quatro dos critérios avaliados. Os problemas eram relacionados a falta de transcrição ou legenda em áudios e vídeos. Por sua vez, vale ressaltar que o LLM demonstrou entender o problema, pois adicionou arquivos para legenda ou transcrição, mas não conseguiu resolver o problema em sua totalidade, visto que as legendas não tinham relação com o conteúdo das mídias.

Outro ponto relevante é que apenas quatro critérios resultaram em novos problemas sendo introduzidos. Este resultado já considera todos os critérios presentes nesse teste, visto que retrocessos aconteceram apenas no princípio 1, o que mostra que regressões em correção de problemas de acessibilidade são raras.

De forma geral, os resultados do princípio 1 mostram que o modelo tem um bom desempenho no ajuste de problemas estruturais e visuais, mas encontra limitações quando o assunto são mídias.

### 3.2.2 2. Operável

<b>Critério/Nome do arquivo</b>	<b>Iterações</b>	<b>Veredito</b>	<b>Introduziu novos problemas?</b>
2.1.1-keyboard	1	Corrigiu totalmente	Não
2.1.2-no-keyboard-trap	1	Corrigiu totalmente	Não
2.1.4-character-key-shortcuts	2	Corrigiu totalmente	Não
2.2.1-timing-adjustable	1	Corrigiu totalmente	Não
2.2.2-pause-stop-hide	1	Corrigiu totalmente	Não
2.3.1-three-flashes-or-below	1	Corrigiu totalmente	Não
2.4.1-bypass-blocks	1	Corrigiu totalmente	Não
2.4.2-page-titled	1	Corrigiu totalmente	Não
2.4.3-focus-order	1	Corrigiu totalmente	Não
2.4.4-link-purpose-in-context	1	Corrigiu totalmente	Não
2.4.5-multiple-ways	1	Corrigiu totalmente	Não
2.4.6-headings-and-labels	1	Corrigiu totalmente	Não
2.4.7-focus-visible	1	Corrigiu totalmente	Não
2.4.11-focus-not-obscured	1	Corrigiu totalmente	Não
2.5.1-pointer-gestures	3	Corrigiu totalmente	Não
2.5.2-pointer-cancellation	1	Corrigiu totalmente	Não
2.5.3-label-in-name	1	Corrigiu totalmente	Não
2.5.4-motion-activation	1	Corrigiu totalmente	Não
2.5.7-dragging-movements	1	Corrigiu totalmente	Não
2.5.8-target-size-minimum	1	Corrigiu totalmente	Não

Tabela 2 – Resultados do Princípio 2 — Operável

Sobre o princípio "Operável", é nítida a facilidade em resolver problemas relacionados a navegação, teclado, foco e interações do usuário, visto apenas dois dos pontos foram resolvidos com mais de uma iteração.

Além disso, é importante ressaltar que nenhuma regressão foi introduzida ao resolver os problemas presentes neste princípio, o que indica forte estabilidade do LLM.

### 3.2.3 3. Entendível

<b>Critério/Nome do arquivo</b>	<b>Iterações</b>	<b>Veredito</b>	<b>Introduziu novos problemas?</b>
3.1.1-language-of-page	1	Corrigiu totalmente	Não
3.1.2-language-of-parts	1	Corrigiu totalmente	Não
3.2.1-on-focus	1	Corrigiu totalmente	Não
3.2.3-consistent-navigation	1	Corrigiu totalmente	Não
3.2.4-consistent-identification	1	Corrigiu totalmente	Não
3.2.6-consistent-help	2	Corrigiu totalmente	Não
3.3.1-error-identification	1	Corrigiu totalmente	Não
3.3.2-labels-or-instructions	1	Corrigiu totalmente	Não
3.3.3-error-suggestion	1	Corrigiu totalmente	Não
3.3.4-error-prevention	1	Corrigiu totalmente	Não
3.3.7-redundancy-entry	1	Corrigiu totalmente	Não
3.3.8-accessible-authentication-minimum	2	Corrigiu totalmente	Não

Tabela 3 – Resultados do Princípio 3 — Entendível

Agora falando sobre o princípio "Entendível", se observa o melhor desempenho dentre todos os princípios apresentados até agora, com praticamente todos os problemas sendo corrigidos com apenas uma iteração e sem regressões.

Critérios relacionados a linguagem (3.1.1 e 3.1.2) foram facilmente resolvidos, indicando boa capacidade em lidar com diferentes idiomas.

O mesmo aconteceu para critérios relacionados a previsibilidade (3.2.1 ao 3.2.6), onde os problemas foram facilmente resolvidos sem efeitos colaterais.

Ao final com critérios voltados a assistência no preenchimento de dados (3.3.1 ao 3.3.8), o modelo desempenhou de forma igualmente positiva.

Com esse resultado, podemos perceber que o modelo é eficaz nas correções e também estável, cobrindo desde problemas linguísticos até dificuldades em usabili-

dade de forma consistente.

### 3.2.4 4. Robusto

<b>Critério/Nome do arquivo</b>	<b>Iterações</b>	<b>Veredito</b>	<b>Introduziu novos problemas?</b>
4.1.2-name-role-value	1	Corrigiu totalmente	Não
4.1.3-status-messages	1	Corrigiu totalmente	Não

Tabela 4 – Resultados do Princípio 4 — Robusto

Sobre o princípio "Robusto", de longe é o menor de todos, o modelo se manteve constante e sem maiores desafios, conseguiu resolver os problemas sem introduzir novos.

Tal resultado demonstra que o modelo é capaz de garantir que o conteúdo seja entendido corretamente por leitores de tela, que são essenciais para pessoas cegas ([BORODIN et al., 2010](#)), e tecnologias assistivas.

Em termos gerais, o modelo se mantém consistente na solução da grande maioria dos problemas de acessibilidade. Das suas limitações, é notória a incapacidade em lidar com mídias como vídeos e áudios.

## 3.3 Capacidade do Claude Code em gerar código acessível

Com a alta taxa de erros encontrados em páginas iniciais de sites ao redor do mundo ([WebAIM \(2024\)](#)), analisou-se como o Claude Code se sairia ao criar uma página inteira, sem intervenção humana.

Para isso, utilizou-se uma técnica comum em projetos que usam o Claude code, que é a criação de um contexto base usando o arquivo "CLAUDE.md".

O "CLAUDE.md" é um arquivo especial que automaticamente adiciona contexto quando uma nova conversa é iniciada. Com isso, se torna um lugar ideal para documentar diretrizes, arquivos essenciais dentro do projeto, entre outros. ([ANTHROPIC, 2025a](#))

Com isso, o arquivo foi preenchido com regras, exemplos de códigos e diretrizes esperadas para o código.

A seguir apresenta-se um trecho presente no arquivo:

Todo desenvolvimento deve seguir os quatro princípios fundamentais da acessibilidade (WCAG):

### 1. Perceptível

O conteúdo deve ser apresentado de formas que todos possam perceber.

#### Regras obrigatórias:

- Toda imagem **deve** possuir *alt text* descritivo.
- Vídeos **devem** conter legendas (captions).
- Arquivos de áudio **devem** ter transcrição ou audiodescrição.
- A informação **não pode** depender exclusivamente de cor.
- Contraste mínimo: **4.5:1** para texto normal e **3:1** para texto grande.

Como o arquivo é muito extenso, não foi adicionado neste trabalho. Ele pode ser encontrado no mesmo repositório onde foram adicionadas as correções para as diretrizes [Paixão \(2025a\)](#).

O *prompt* utilizado para a criação da página foi a seguinte:

Voce deve criar uma *landing page* sobre esportes, com componentes para cada seção, sendo elas:

- Header com logo a esquerda e links de navegação a direita
- Hero section
- Section com conteúdo
- Section com testimonials
- Section para inscrição em newsletter
- Footer com links

Após a criação de cada seção, voce irá adicioná-la ao arquivo `app.tsx`, bem como qualquer estilo criado para a seção. As seções devem ter boa aparência e seguir a mesma linha de cores, fonte, espaçamento caso necessário, etc.

Voce deve criar uma seção por vez e me perguntar se precisa de mais algum ajuste ou pode seguir para a próxima seção.

A figura a seguir apresenta o processo completo de criação e validação da página:

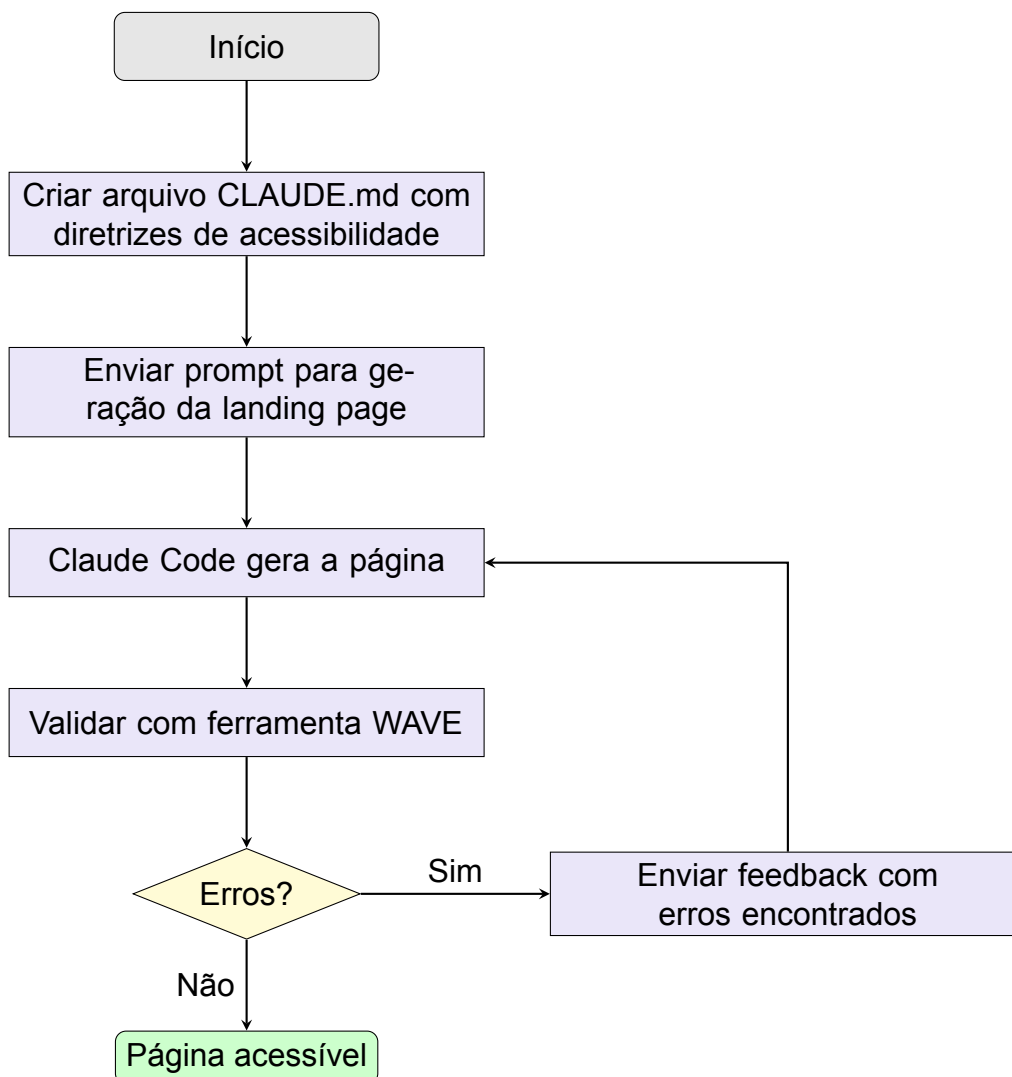


Figura 2 – Fluxograma do processo de criação da página acessível

Com isso, esta página inicial foi criada:

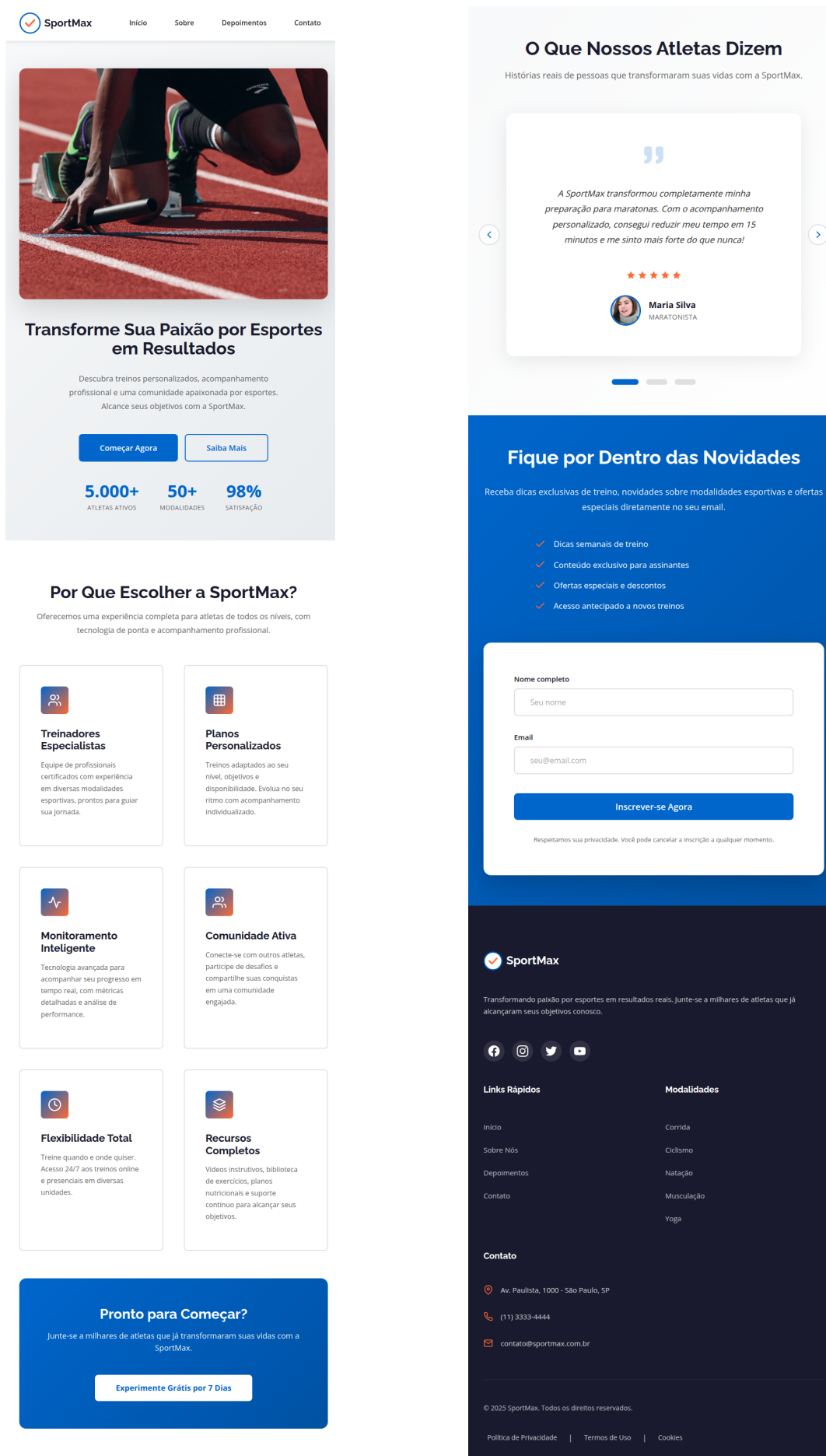


Figura 3 – Página criada pelo Claude Code dividida em duas partes

Para validar o nível de acessibilidade da página gerada pelo Claude Code, foi utilizada a ferramenta Wave, capaz de analisar websites estaticamente com base na [WCAG](#) versão 2.2 de forma gratuita [WebAIM \(2025a\)](#).

Ao usar a ferramenta, pudemos encontrar os seguintes erros:

### **Hero**

- 1 alerta para aria label muito longa, o que não necessariamente se traduz em problema

### **Footer**

- 2 erros de contraste
- 8 alertas de link levando para a mesma página, porém para um item inexistente.

O seguinte ajuste foi pedido:

Ao usar a ferramenta de análise de acessibilidade Wave, foram apontados alguns problemas:

Na section Hero, um alerta é mostrado devido ao longo texto alternativo associado à imagem.

Já no Footer, 8 alertas são mostrados para os links da parte de modalidades, termos de uso, politica de privacidade e cookies, devido a utilização de links que levam a mesma pagina, mas que estão quebrados (nao existe um elemento associado ao id).

Ainda no footer, problemas de contraste sao apontados para os separadores dos links termos de uso, politica de privacidade e cookies;

Pode resolver esses problemas?

Após esse ajuste, um novo teste usando a Wave foi feito e nenhum erro de acessibilidade foi encontrado. A tabela a seguir resume as correções realizadas:

Seção	Problema identificado	Correção realizada
Hero	Texto alternativo muito longo na imagem	Redução do texto alternativo para uma descrição mais concisa
Footer	Erro de contraste nos separadores dos links	Ajuste das cores dos separadores para atender aos níveis de contraste necessários
Footer	Links quebrados apontando para IDs inexistentes	Uso de texto simples

Tabela 5 – Correções de acessibilidade realizadas pelo Claude Code na landing page

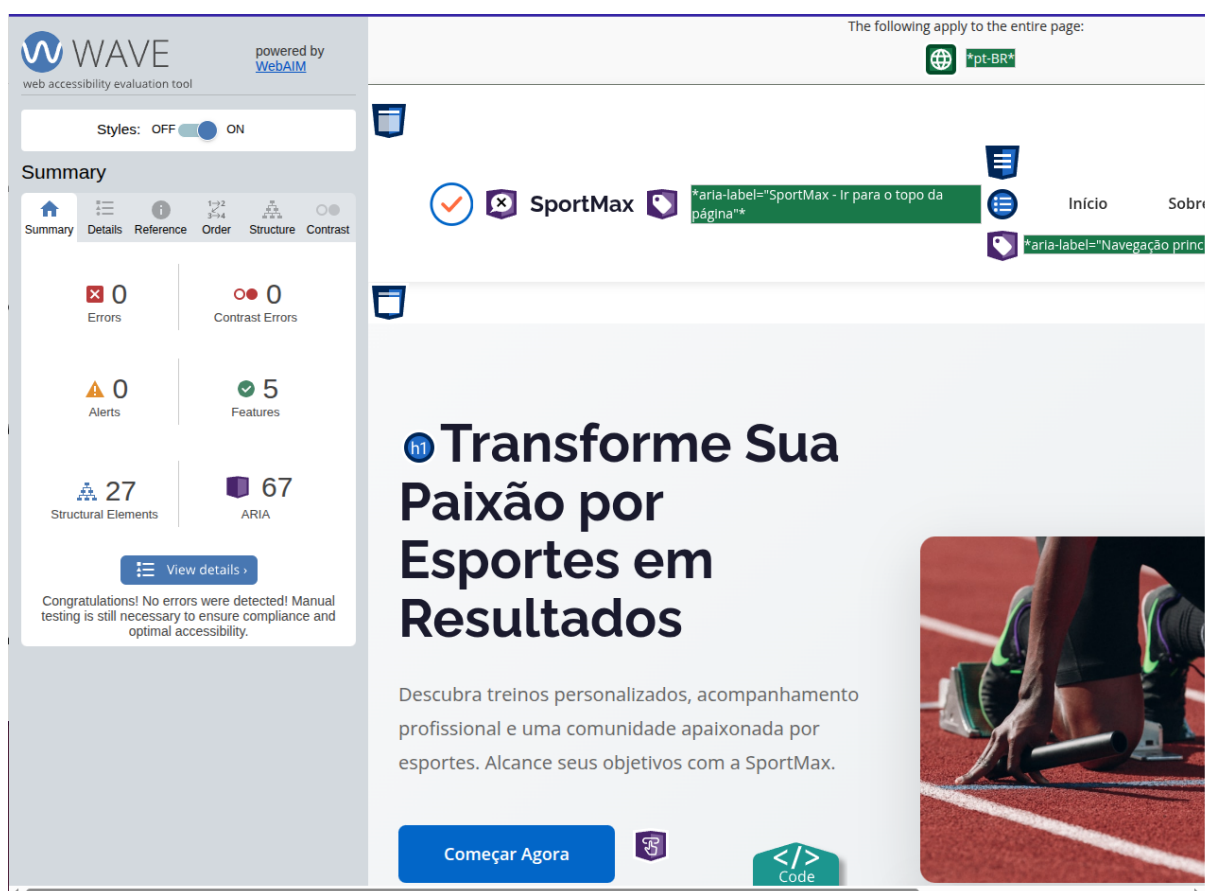


Figura 4 – Página criada pelo Claude code sem erros após ajustes

Assim, podemos observar que é possível criar páginas acessíveis usando o **LLM** como apoio no desenvolvimento diário.

Vale mencionar que, embora a ferramenta Wave consiga identificar problemas de acessibilidade, testes manuais ainda se fazem necessários, visto que alguns tipos de erros são indetectáveis para esse tipo de ferramenta, como elementos piscantes sem aviso ou o uso de linguagem complexa no conteúdo das páginas (**GDS**) (2018).

### 3.4 Analisando páginas iniciais de Universidades pelo Brasil

Seguindo com a análise de páginas iniciais, sites de universidades brasileiras foram testados a fim de identificar se os erros encontrados seriam corrigidos com a ajuda do Claude.

Com base nisso, as páginas iniciais de 27 universidades foram testadas usando a mesma ferramenta usada anteriormente (Wave).

Os resultados foram dispostos em duas tabelas complementares. Uma apresenta os erros de cada universidade de forma geral, e outra destrincha os erros e os separa, a fim de identificar os tipos de erros mais frequentes.

<b>Universidade</b>	<b>URL</b>	<b>Total de erros</b>	<b>Tipos de erro distintos</b>
Universidade Federal Rural de Pernambuco (UFRPE)	<a href="https://portal.ufrpe.br/">https://portal.ufrpe.br/</a>	30	5
Universidade Federal de Pernambuco (UFPE)	<a href="https://ufpe.br">https://ufpe.br</a>	23	6
Universidade de Pernambuco (UPE)	<a href="https://www.upe.br/">https://www.upe.br/</a>	27	5
Universidade de São Paulo (USP)	<a href="https://www5.usp.br/">https://www5.usp.br/</a>	117	4
Universidade Estadual de Campinas (Unicamp)	<a href="https://unicamp.br/">https://unicamp.br/</a>	45	3
Universidade Federal do Rio de Janeiro (UFRJ)	<a href="https://ufrj.br/">https://ufrj.br/</a>	18	4
Universidade Federal de Minas Gerais (UFMG)	<a href="https://ufmg.br/">https://ufmg.br/</a>	73	4
Universidade Federal do Rio Grande do Sul (UFRGS)	<a href="https://www.ufrgs.br/site/">https://www.ufrgs.br/site/</a>	39	7
Universidade Federal de Santa Catarina (UFSC)	<a href="https://ufsc.br/">https://ufsc.br/</a>	14	6
Universidade de Brasília (UnB)	<a href="https://unb.br/">https://unb.br/</a>	16	7
Universidade Federal da Bahia (UFBA)	<a href="https://www.ufba.br/">https://www.ufba.br/</a>	27	6

*Continua na próxima página*

<b>Universidade</b>	<b>URL</b>	<b>Total de erros</b>	<b>Tipos de erro distintos</b>
Universidade Federal do Paraná (UFPR)	<a href="https://ufpr.br/">https://ufpr.br/</a>	55	5
Universidade Federal do Ceará (UFC)	<a href="https://www.ufc.br/">https://www.ufc.br/</a>	10	4
Universidade Federal de Goiás (UFG)	<a href="https://ufg.br/">https://ufg.br/</a>	115	6
Universidade Federal de Mato Grosso do Sul (UFMS)	<a href="https://www.ufms.br/">https://www.ufms.br/</a>	54	6
Universidade Federal do Espírito Santo (UFES)	<a href="https://ufes.br/">https://ufes.br/</a>	5	1
Universidade Federal de Mato Grosso (UFMT)	<a href="https://ufmt.br/">https://ufmt.br/</a>	35	4
Universidade Federal da Paraíba (UFPB)	<a href="https://ufpb.br/">https://ufpb.br/</a>	42	4
Universidade Federal do Pará (UFPA)	<a href="https://ufpa.br/">https://ufpa.br/</a>	32	5
Universidade Federal de Ouro Preto (UFOP)	<a href="https://ufop.br/">https://ufop.br/</a>	45	5
Universidade Federal de Pelotas (UFPEl)	<a href="https://portal.ufpel.edu.br">https://portal.ufpel.edu.br</a>	129	6
Universidade Federal de Santa Maria (UFSM)	<a href="https://www.ufsm.br/">https://www.ufsm.br/</a>	9	2
Universidade Federal do ABC (UFABC)	<a href="https://www.ufabc.edu.br">https://www.ufabc.edu.br</a>	45	5
Universidade Federal do Acre (UFAC)	<a href="https://www.ufac.br/">https://www.ufac.br/</a>	42	4
Universidade Federal do Amazonas (UFAM)	<a href="https://www.ufam.edu.br/">https://www.ufam.edu.br/</a>	36	5
Universidade Federal do Piauí (UFPI)	<a href="https://www.ufpi.br/">https://www.ufpi.br/</a>	20	1

Universidade Federal de Juiz de Fora (UFJF)	<a href="https://www2.ufjf.br/ufjf/">https://www2.ufjf.br/ufjf/</a>	25	5
---	---	----	---

Tabela 6 – Quantitativo de erros de acessibilidade e diversidade de tipos em websites de universidades brasileiras

Dos erros encontrados temos a seguinte distribuição:

Tipo de erro	Critério	Nº de ocorrências	Nº de universidades afetadas	% de universidades afetadas
Erro de contraste	1.4.3	612	26	96%
Rótulo de formulário ausente	1.1.1 – 1.3.1 – 2.4.6 – 3.3.2	68	20	74%
Imagem com link sem texto alternativo	1.1.1 – 2.4.4	229	20	74%
Link vazio	2.4.4	126	17	63%
Texto alternativo ausente	1.1.1	42	14	52%
Botão vazio	1.1.1 – 2.4.4	20	9	33%
Tag de cabeçalho vazia	1.3.1 – 2.4.1 – 2.4.6	17	4	15%
Aria-menu quebrado	2.1.1 – 4.1.2	6	3	11%
Linguagem faltando ou inválida	3.1.1	3	3	11%
Cabeçalho de tabela vazio	1.3.1	2	1	4%
Rótulo de formulário vazio	1.1.1 – 1.3.1 – 2.4.6 – 3.3.2	2	1	4%
Múltiplos rótulos de formulários	1.1.1 – 1.3.1 – 2.4.6 – 3.3.2	1	1	4%

Tabela 7 – Erros de acessibilidade mais frequentes nas universidades analisadas

Com esses dados, podemos observar um número elevado de problemas de acessibilidade em universidades brasileiras. O erro mais comum foi o de contraste, encontrado em 96% das instituições, ocorrendo 612 vezes. A partir disso, pode-se esperar que parte do conteúdo está excluído da população com baixa visão, sendo um problema impactante na experiência dos usuários.

Ainda, elementos sem rótulo de identificação e imagens sem texto alternativo também tiveram um número expressivo, estando presente em 74% dos casos e somando juntos 297 ocorrências.

Mesmo erros de navegação chamam atenção, estando presente em 63% das instituições, mostrando inconsistências que prejudicam a compreensão e navegação por teclado ou tecnologia assistiva.

Outros problemas também tiveram aparições, mesmo que menos frequentes indicam que há espaço para melhorias, como é o caso de erros como botões vazios, tags de cabeçalho vazias, texto alternativo ausente, entre outros.

Com isso, pode-se fazer um paralelo entre os erros que foram identificados nas universidades e erros resolvidos pelo Claude code identificados na primeira parte deste experimento:

<b>Tipo de erro</b>	<b>Critérios</b>	<b>Status de correção pelo Claude Code</b>
Erro de contraste	1.4.3	Corrigiu totalmente
Rótulo de formulário ausente	1.1.1, 1.3.1, 2.4.6, 3.3.2	1.1.1 – total 1.3.1 – parcial 2.4.6 – total 3.3.2 – total
Imagem com link sem texto alternativo	1.1.1, 2.4.4	1.1.1 – total 2.4.4 – total
Link vazio	2.4.4	Corrigiu totalmente
Texto alternativo ausente	1.1.1	Corrigiu totalmente
Botão vazio	1.1.1, 2.4.4	1.1.1 – total 2.4.4 – total
Tag de cabeçalho vazia	1.3.1, 2.4.1, 2.4.6	1.3.1 – parcial 2.4.1 – total 2.4.6 – total
Aria-menu quebrado	2.1.1, 4.1.2	2.1.1 – total 4.1.2 – total
Linguagem faltando ou inválida	3.1.1	Corrigiu totalmente
Cabeçalho de tabela vazio	1.3.1	Corrigiu parcialmente
Rótulo de formulário vazio	1.1.1, 1.3.1, 2.4.6, 3.3.2	1.1.1 – total 1.3.1 – parcial 2.4.6 – total 3.3.2 – total
Múltiplos rótulos de formulários	1.1.1, 1.3.1, 2.4.6, 3.3.2	1.1.1 – total 1.3.1 – parcial 2.4.6 – total 3.3.2 – total

Tabela 8 – Paralelo entre erros nas universidades e correções realizadas pelo Claude Code

Pode se observar que apenas um dos critérios encontrados nas universidades

poderia não ser resolvido adequadamente, visto que nas análises de correção, o LLM corrigiu o problema parcialmente. Para todos os outros casos, o mesmo conseguiria resolver e ajudar os desenvolvedores a criar sites acessíveis no dia a dia.

De forma geral, os dados mostram que universidades apresentam um grande número de erros que poderiam ser resolvidos com a ajuda do Claude code, visto que o LLM mostrou um notório potencial para solução de problemas dessa natureza.

## 4 Conclusão

Com base nos levantamentos presentes neste trabalho, foi possível afirmar que ferramentas baseadas em **LLM**, especificamente o Claude Code, possuem capacidade técnica para atuar como aliadas no desenvolvimento de páginas web acessíveis.

Ao longo dos experimentos, ficou evidente uma desconformidade no desempenho do modelo. Enquanto questões textuais, estruturais e de interação foram tratadas com precisão, muitas vezes resolvidas em uma única tentativa, o cenário mudou quando o problema envolveu mídia. A incapacidade de produzir legendas ou transcrições de acordo com o conteúdo audiovisual mostra um limite claro na ferramenta, podendo-se concluir que o modelo opera sobre texto e código, não sobre uma ótica de áudio, vídeo ou imagens.

O levantamento realizado em 27 universidades federais brasileiras expõe problemas preocupantes. Instituições públicas de ensino superior que deveriam servir como referência em inclusão, sofrem com um número de erros alarmante em suas páginas iniciais. Como exemplos comuns temos contraste inadequado, links sem descrição e formulários sem identificação. Felizmente, o Claude Code se mostrou eficiente atuando na solução para esses tipos de falhas.

A experiência de construção de uma página do zero guiada por diretrizes mostrou que o contexto fornecido ao modelo faz diferença. Quando as regras de acessibilidade estão explícitas, o código gerado tende a respeitá-las desde o início, reduzindo a necessidade de correções.

Este trabalho aponta para uma mudança possível na dinâmica de desenvolvimento web. A falta de conhecimento especializado em acessibilidade frequentemente citada como causa pode ser melhorada substancialmente por LLMs. Não como substitutos, mas como ferramentas que melhoram as boas práticas e tornam a inclusão digital menos dependente de conhecimento específico em algum tipo de deficiência ou diretriz.

Por fim, diversas oportunidades para pesquisas futuras podem ser vistas. As principais são:

- **Modelos multimodais para mídias:** Investigar o uso de **LLMs** com capacidades multimodais, como análise de áudio e vídeo, para superar a limitação encontrada relacionada a mídias;
- **Estudo comparativo entre LLMs:** Realizar análises comparativas entre diferentes modelos (Claude, GPT, Gemini, entre outros) para identificar pontos fortes e

fracos de cada um na correção de problemas de acessibilidade;

- **Avaliação do nível AAA:** Expandir a análise para incluir os critérios do nível AAA da [WCAG](#), verificando se os [LLMs](#) conseguem atender aos requisitos mais avançados de acessibilidade;
- **Integração em pipelines de desenvolvimento:** Estudar a integração de [LLMs](#) em processos de integração e entrega contínua (CI/CD), automatizando a verificação e correção de acessibilidade durante o ciclo de desenvolvimento;
- **Adoção em contextos reais:** Investigar a adoção dessas ferramentas por equipes de desenvolvimento em projetos reais, avaliando impactos na produtividade e na qualidade do código acessível produzido;

Essas direções de pesquisa podem contribuir para aumentar o uso de inteligência artificial como ferramenta de apoio a inclusão digital, ampliando o alcance e a qualidade da acessibilidade na web.

## Referências

ANDRUCCIOLI, M. et al. Leveraging large language models for sustainable and inclusive web accessibility. *Big Data and Cognitive Computing*, v. 9, n. 10, 2025. ISSN 2504-2289. Disponível em: <<https://www.mdpi.com/2504-2289/9/10/247>>. Citado 4 vezes nas páginas 10, 14, 16 e 17.

ANTHROPIC. *Claude Code Best Practices*. 2025. Acesso em: 12 nov. 2025. Disponível em: <<https://www.anthropic.com/engineering/claude-code-best-practices>>. Citado na página 25.

ANTHROPIC. *Claude Code: Deep Coding at Terminal Velocity*. 2025. Acesso em: 22 nov. 2025. Disponível em: <<https://claude.com/product/claude-code>>. Citado 2 vezes nas páginas 10 e 15.

BI, T. et al. Accessibility in software practice: A practitioner's perspective. *arXiv:2103.08778*, 2021. ArXiv preprint, 16 Mar 2021. Disponível em: <<https://arxiv.org/abs/2103.08778>>. Citado 3 vezes nas páginas 10, 15 e 16.

BI, T. et al. Accessibility in software practice: A practitioner's perspective. *ACM Trans. Softw. Eng. Methodol.*, Association for Computing Machinery, New York, NY, USA, v. 31, n. 4, jul. 2022. ISSN 1049-331X. Disponível em: <<https://doi.org/10.1145/3503508>>. Citado 3 vezes nas páginas 10, 14 e 16.

BORODIN, Y. et al. More than meets the eye: a survey of screen-reader browsing strategies. In: *Proceedings of the 2010 International Cross Disciplinary Conference on Web Accessibility (W4A)*. New York, NY, USA: Association for Computing Machinery, 2010. (W4A '10). ISBN 9781450300452. Disponível em: <<https://doi.org/10.1145/1805986.1806005>>. Citado na página 25.

BRITTO, T.; PIZZOLATO, E. Cognitive disabilities and web accessibility: a survey into the brazilian web development community. *Journal on Interactive Systems*, p. 1, 12 2021. Citado 2 vezes nas páginas 12 e 16.

CANADA, G. of. *Accessible Canada Act (A-0.6)*. 2019. Acesso em: 21 nov. 2025. Disponível em: <<https://laws-lois.justice.gc.ca/eng/acts/A-0.6/>>. Citado na página 13.

CONSORTIUM, W. W. W. *Web Content Accessibility Guidelines*. 2023. Acesso em: 12 nov. 2025. Disponível em: <<https://www.w3.org/TR/WCAG>>. Citado 3 vezes nas páginas 10, 12 e 13.

CORREA, M.; VITORIANO, M.; LLANOS, C. Web accessibility in an academic management system in brazil: Problems and challenges for attending people with visual impairments. *Informatics*, v. 12, p. 63, 07 2025. Citado 2 vezes nas páginas 15 e 16.

DIGITAL, B. S. de G. *Modelo de Acessibilidade em Governo Eletrônico – eMAG*. 2014. Acesso em: 13 nov. 2025. Disponível em: <<https://emag.governoeletronico.gov.br/>>. Citado 2 vezes nas páginas 10 e 12.

- EDITORS, F. *Anthropic's CEO says why he quit OpenAI to start a competitor that just received billions from Amazon and Google*. 2023. Acesso em: 10 dez. 2025. Disponível em: <<https://fortune.com/2023/09/26/anthropic-ceo-interview-quit-open-ai-amazon-investment/>>. Citado na página 15.
- FEDERAL, G. *IBGE divulga censo sobre pessoas com deficiência no Brasil*. 2025. Acesso em: 11 out. 2025. Disponível em: <<https://www.gov.br/mdh/pt-br/assuntos/noticias/2025/maio/pela-primeira-vez-ibge-divulga-dados-sobre-pessoas-com-deficiencia-no-brasil>>. Citado 2 vezes nas páginas 10 e 12.
- (GDS), G. D. S. *Accessibility Tool Audit: How do automated accessibility checkers compare?* 2018. Acesso em: 12 nov. 2025. Disponível em: <<https://alphagov.github.io/accessibility-tool-audit/>>. Citado na página 30.
- GURIÑÁ, A.-E.; VATAVU, R.-D. When llm-generated code perpetuates user interface accessibility barriers, how can we break the cycle? In: *Proceedings of the 22nd International Web for All Conference (W4A '25)*. New York, NY, USA: ACM, 2025. p. 11. Citado 4 vezes nas páginas 10, 15, 16 e 17.
- JETBRAINS. *Artificial Intelligence – The State of Developer Ecosystem in 2025*. 2025. Acesso em: 22 nov. 2025. Disponível em: <<https://devecosystem-2025.jetbrains.com/artificial-intelligence>>. Citado 2 vezes nas páginas 10 e 14.
- JUSTICE, U. D. of. *Fact Sheet: New Rule on the Accessibility of Web Content and Mobile Apps Provided by State and Local Governments*. 2024. Acesso em: 21 nov. 2025. Disponível em: <<https://www.ada.gov/resources/2024-03-08-web-rule/>>. Citado na página 13.
- MOWAR, P. Accessibility in ai-assisted web development. In: . [S.l.: s.n.], 2024. p. 123–125. Citado 3 vezes nas páginas 10, 15 e 16.
- OVERFLOW, S. *2025 Stack Overflow Developer Survey*. 2025. Acesso em: 22 nov. 2025. Disponível em: <<https://survey.stackoverflow.co/2025>>. Citado 2 vezes nas páginas 10 e 14.
- PAIXÃO, R. *Accessibility Project — CLAUDE*. 2025. <[https://github.com/RenanPaixao/accessibility\\_project/blob/page-creation/CLAUDE.md](https://github.com/RenanPaixao/accessibility_project/blob/page-creation/CLAUDE.md)>. Arquivo Markdown no GitHub, acessado em 17 nov. 2025. Citado na página 26.
- PAIXÃO, R. *Projeto de Teste de Acessibilidade em React*. 2025. Acesso em: 13 nov. 2025. Disponível em: <[https://github.com/RenanPaixao/accessibility\\_project](https://github.com/RenanPaixao/accessibility_project)>. Citado na página 19.
- SOBO, A. et al. Evaluating llms for code generation in hri: A comparative study of chatgpt, gemini, and claude. *Applied Artificial Intelligence*, Taylor & Francis, v. 39, n. 1, p. 2439610, 2025. Disponível em: <<https://doi.org/10.1080/08839514.2024.2439610>>. Citado 3 vezes nas páginas 10, 14 e 15.
- STRYKER, C.; IBM. *What Are Large Language Models (LLMs)?* 2025. Acesso em: 22 nov. 2025. Disponível em: <<https://www.ibm.com/think/topics/large-language-models>>. Citado na página 14.

WAI, W. W. W. C. W. □. *Understanding WCAG 2.2: Conformance Levels*. 2023. Acesso em: 12 nov. 2025. Disponível em: <<https://www.w3.org/WAI/WCAG22/Understanding/conformance#levels>>. Citado 2 vezes nas páginas 13 e 19.

WAI, W. W. W. C. W. □. *Understanding WCAG 2.2: Introduction – The Four Principles of Accessibility*. 2023. Acesso em: 12 nov. 2025. Disponível em: <<https://www.w3.org/WAI/WCAG22/Understanding/intro#understanding-the-four-principles-of-accessibility>>. Citado na página 13.

WAI, W. W. W. C. W. □. *How to Meet WCAG 2.2 (Quick Reference)*. 2024. Acesso em: 12 nov. 2025. Disponível em: <<https://www.w3.org/WAI/WCAG22/quickref/>>. Citado na página 19.

(WAI), W. W. W. C. W. □. W. A. I. *Web Content Accessibility Guidelines 1.0*. 1999. Acesso em: 12 nov. 2025. Disponível em: <<https://www.w3.org/TR/WAI-WEBCONTENT/>>. Citado na página 12.

WEBAIM. *The WebAIM Million: The 2024 Report on the Accessibility of the Top 1,000,000 Home Pages*. 2024. Acesso em: 21 nov. 2025. Disponível em: <<https://webaim.org/projects/million/>>. Citado 5 vezes nas páginas 10, 14, 15, 19 e 25.

WEBAIM. *WAVE Help*. 2025. Acesso em: 10 dez. 2025. Disponível em: <<https://wave.webaim.org/help>>. Citado na página 29.

WEBAIM. *WAVE Web Accessibility Evaluation Tool*. 2025. Acesso em: 22 nov. 2025. Disponível em: <<https://wave.webaim.org/>>. Citado na página 14.

WEGENER, A. et al. A universal web accessibility feedback form: A participatory design study. In: *Proceedings of the 21st International Web for All Conference (W4A '24)*. New York, NY, USA: ACM, 2024. Citado na página 12.

WHO, W. H. O. *Disability*. 2023. Acesso em: 12 nov. 2025. Disponível em: <<https://www.who.int/news-room/fact-sheets/detail/disability-and-health>>. Citado 2 vezes nas páginas 10 e 12.