



**UNIVERSIDADE FEDERAL DO CEARÁ**  
**CAMPUS DE QUIXADÁ**  
**CURSO DE GRADUAÇÃO EM SISTEMAS DE INFORMAÇÃO**

**DAVI TEIXEIRA SILVA**

**ACCESSHUB: UM GUIA DE ACESSIBILIDADE PARA DESENVOLVEDORES WEB**

**QUIXADÁ**  
**2023**

DAVI TEIXEIRA SILVA

ACCESSHUB: UM GUIA DE ACESSIBILIDADE PARA DESENVOLVEDORES WEB

Trabalho de Conclusão de Curso apresentado ao Curso de Graduação em Sistemas de Informação do Campus de Quixadá da Universidade Federal do Ceará, como requisito parcial à obtenção do grau de bacharel em Sistemas de Informação.

Orientadora: Profa. Dra. Carla Ilane Moreira Bezerra

Coorientador: Bel. José Cezar Junior de Souza Filho

QUIXADÁ

2023

Dados Internacionais de Catalogação na Publicação  
Universidade Federal do Ceará  
Sistema de Bibliotecas  
Gerada automaticamente pelo módulo Catalog, mediante os dados fornecidos pelo(a) autor(a)

---

- S579a Silva, Davi Teixeira.  
AccessHub: um guia de acessibilidade para desenvolvedores web / Davi Teixeira Silva. – 2023.  
88 f. : il. color.
- Trabalho de Conclusão de Curso (graduação) – Universidade Federal do Ceará, Campus de Quixadá,  
Curso de Sistemas de Informação, Quixadá, 2023.  
Orientação: Profa. Dra. Carla Ilane Moreira.  
Coorientação: Prof. José Cezar Junior de Souza Filho.
1. Acessibilidade Web. 2. Diretrizes de Acessibilidade. 3. Open Source. 4. Reúso de Software. 5.  
WCAG. I. Título.

CDD 005

---

DAVI TEIXEIRA SILVA

ACCESSHUB: UM GUIA DE ACESSIBILIDADE PARA DESENVOLVEDORES WEB

Trabalho de Conclusão de Curso apresentado ao Curso de Graduação em Sistemas de Informação do Campus de Quixadá da Universidade Federal do Ceará, como requisito parcial à obtenção do grau de bacharel em Sistemas de Informação.

Aprovada em: \_\_\_\_/\_\_\_\_/\_\_\_\_.

BANCA EXAMINADORA

---

Profa. Dra. Carla Ilane Moreira  
Bezerra (Orientadora)  
Universidade Federal do Ceará (UFC)

---

Bel. José Cezar Junior de Souza Filho (Coorientador)  
Universidade Federal do Amazonas (UFAM)

---

Profa. Dra. Anna Beatriz Marques  
Universidade Federal do Ceará (UFC)

---

Profa. Dra. Ingrid Teixeira Monteiro  
Universidade Federal do Ceará (UFC)

Dedico este trabalho aos meus amados pais, pela sua dedicação e apoio incondicional. Também dedico este trabalho às pessoas com deficiência, cuja força e resiliência inspiraram a criação do AccessHub. Que este guia contribua para um mundo digital mais inclusivo e acessível.

## AGRADECIMENTOS

Primeiro gostaria de agradecer a Deus, por ter dado a todos a oportunidade de desfrutar a vida. Por nos proporcionar o dom de aprender, de amar, de ter fé e de sonhar. Por expressar o seu amor e nos presentear com este mundo repleto de maravilhas. Por nos abençoar durante todos os nossos caminhos.

À minha mãe Elianne, ao meu pai Denilson, e a minha irmã Ellen, quero que saibam que esta conquista é nossa, pois a conclusão deste trabalho e desta jornada só foi possível graças a todo o apoio que vocês me deram. Por mais que eu pense sobre o que escrever, das palavras de carinho que quero expressar, nada parece ser o suficiente para descrever o quanto sou grato e amo vocês.

Também sou imensamente grato a você Aline, você é a melhor surpresa que essa jornada me trouxe, você que sempre me apoiou, que esteve comigo em todos os momentos bons e ruins, que me deu forças e não largou a minha mão. Foi o seu carinho, amor e cuidado que me deram forças. Saiba que minha vida ficou mais feliz depois que te conheci, te amo.

Agradeço à minha orientadora Carla Ilane, por ter aceitado a minha proposta de trabalho, por ter visto o potencial que esta pesquisa tem, por todo o apoio, orientação e contribuição para que esse projeto fosse desenvolvido da melhor maneira. Aprendi bastante com a senhora.

Ao meu coorientador e amigo Cezar Junior, pela sua orientação e valiosa contribuição, por dedicar o seu tempo e conhecimentos, por me fazer perceber o quanto interessante e incrível é a área de pesquisas, por sempre me tranquilizar durante todo o processo de construção deste projeto.

Gostaria de agradecer também a Universidade Federal do Ceará, por esse campus em Quixadá do qual vou carregar comigo muitas memórias e aprendizado, agradeço pela educação pública de qualidade e a todos os professores e servidores que contribuíram para a minha graduação.

Por último mas não menos importante, eu quero me agradecer. Eu quero agradecer por ter acreditado em mim, por fazer todo o trabalho duro, por não ter tirado folga e por nunca ter desistido. Eu quero me agradecer por sempre tentar dar mais do que recebo. Eu quero me agradecer por tentar fazer mais o certo do que o errado e continuar a ser eu mesmo, em todos os momentos.

“A verdadeira inclusão digital só é alcançada quando o acesso à informação e aos serviços online é garantido a todas as pessoas, independentemente de suas habilidades e deficiências.”

(Sérgio Amadeu)

## RESUMO

Visto a preocupação de garantir o desenvolvimento de conteúdos ou serviços acessíveis na *internet*, este trabalho aborda o desenvolvimento do AccessHub, um guia abrangente sobre funcionalidades de acessibilidade para desenvolvedores *web*. O guia tem como objetivo apresentar padrões, materiais de referência e um conjunto de exemplos de código *open source*, através dos conceitos de reúso de *software*, que possam ser facilmente acessados, compreendidos e compartilhados, buscando facilitar o processo de aprendizagem e simplificar a implementação de recursos acessíveis em *websites*. O AccessHub baseia-se nas diretrizes e recomendações do *Web Content Accessibility Guidelines (WCAG)* e do Modelo de Acessibilidade em Governo Eletrônico (eMAG), garantindo a conformidade com padrões internacionais e nacionais de acessibilidade. O guia apresenta uma vasta gama de recursos de acessibilidade, abordando aspectos visuais, auditivos e de interação para atender às necessidades de pessoas com diferentes deficiências. Durante o processo de avaliação do projeto, que aconteceu em duas etapas, sendo elas a auditoria individual do *website* e o desafio técnico com os participantes da pesquisa, o AccessHub obteve resultados positivos, com destaque na sua abordagem didática e na facilidade de entendimento e utilização dos recursos disponíveis, demonstrando-se como uma ferramenta promissora para o desenvolvimento de projetos *web* acessíveis. Através desse guia, espera-se que os desenvolvedores possam criar *interfaces* acessíveis, garantindo uma experiência digital igualitária para todos os usuários, independentemente de suas habilidades ou limitações, representando assim um passo significativo em direção a um ambiente *online* mais inclusivo.

**Palavras-chave:** Acessibilidade *Web*; Diretrizes de Acessibilidade; WCAG; eMAG; *Open Source*; Reúso de *Software*.



## ABSTRACT

Given the concern to ensure the development of accessible content or services on the internet, this work addresses the development of AccessHub, a comprehensive guide on accessibility features for web developers. The guide aims to present standards, reference materials and a set of examples of open source code, through the concepts of software reuse, which can be easily accessed, understood and shared, seeking to facilitate the learning process and simplify the implementation of resources accessible on websites. AccessHub is based on the guidelines and recommendations of the Web Content Accessibility Guidelines (WCAG) and the Electronic Government Accessibility Model (eMAG), ensuring compliance with international and national accessibility standards. The guide presents a wide range of accessibility resources, addressing visual, auditory and interaction aspects to meet the needs of people with different disabilities. During the project evaluation process, which took place in two stages, namely the individual audit of the website and the technical challenge with the research participants, AccessHub obtained positive results, with emphasis on its didactic approach and ease of understanding and use available resources, demonstrating itself as a promising tool for the development of accessible web projects. Through this guide, developers are expected to be able to create accessible interfaces, ensuring an equal digital experience for all users, regardless of their abilities or limitations, thus representing a significant step towards a more inclusive online environment.

**Keywords:** Web Accessibility; Accessibility Guidelines; WCAG; eMAG; Open Source; Software Reuse.

## LISTA DE FIGURAS

Figura 1 – Fluxo de atividades. . . . .	35
Figura 2 – Representação hierárquica da estrutura do site . . . . .	41
Figura 3 – Tela de Home Page . . . . .	49
Figura 4 – Tela de Introdução . . . . .	50
Figura 5 – Tela de Componentes . . . . .	50
Figura 6 – Apresentação dos diferentes tipos de resolução e Dark Mode do projeto . . .	51
Figura 7 – Visão geral dos 16 componentes de funcionalidade do AccessHub . . . . .	52
Figura 8 – Imagem do código fonte do componente para alterar o tamanho da fonte . .	58
Figura 9 – Resultados do questionário TAM . . . . .	68

## LISTA DE TABELAS

Tabela 1 – Tabela da priorização das diretrizes de acessibilidade do princípio Perceptível da WCAG. . . . .	38
Tabela 2 – Pontuações obtidas por cada página desenvolvida do AccessHub. . . . .	62
Tabela 3 – Pontuações obtidas por cada desenvolvedor no desafio técnico. . . . .	65
Tabela 4 – Percentual de aumento ou redução da pontuação obtida dos participantes . .	66

## LISTA DE QUADROS

Quadro 1 – Princípios e diretrizes da WCAG 2.1 . . . . .	22
Quadro 2 – Seções e Recomendações do eMag . . . . .	24
Quadro 3 – Abordagens de Reutilização de <i>Software</i> . . . . .	26
Quadro 4 – Principais vantagens das bibliotecas de programas. . . . .	27
Quadro 5 – Análise comparativa entre os trabalhos relacionados e este trabalho. . . . .	33
Quadro 6 – Critérios de seleção das diretrizes. . . . .	37
Quadro 7 – Afirmativas do questionário TAM adaptados ao contexto do projeto. . . . .	43
Quadro 8 – Perfil dos participantes. . . . .	64

## LISTA DE ABREVIATURAS E SIGLAS

W3C	<i>World Wide Web Consortium</i>
WAI	<i>Web Accessibility Initiative</i>
WCAG	<i>Web Content Accessibility Guidelines</i>
ATAG	<i>Authoring Tool Accessibility Guidelines</i>
UAAG	<i>User Agent Accessibility Guideline</i>
eMAG	Modelo de Acessibilidade em Governo Eletrônico
TA	Tecnologias Assistivas
DOM	<i>Document Object Model</i>
HTML	<i>Hyper Text Markup Language</i>
CSS	<i>Cascading Style Sheets</i>
NATO	<i>North Atlantic Treaty Organization</i>
OSI	<i>Open Source Initiative</i>
TEA	Transtorno do Espectro Autista
GAIA	Guia de Acessibilidade de Interfaces Web com foco em aspectos do Autismo
GAW	Guia de Acessibilidade <i>Web</i>
UX	<i>User Experience</i>
TCLE	Termo de Consentimento Livre e Esclarecido
TAM	Modelo de Aceitação de Tecnologia
UP	Utilidade Percebida
FUP	Facilidade de Uso Percebida
IC	Intenção Comportamental
SEO	<i>Search Engine Optimization</i>
SPA	<i>Single Page Application</i>
UI	<i>User Interfaces</i>
PO	<i>Product Owner</i>
QA	<i>Quality Assurance</i>

## SUMÁRIO

<b>1</b>	<b>INTRODUÇÃO</b>	<b>16</b>
<b>1.1</b>	<b>Objetivos</b>	<b>18</b>
<b>1.1.1</b>	<i>Objetivo Geral</i>	<b>18</b>
<b>1.1.2</b>	<i>Objetivos Específicos</i>	<b>18</b>
<b>1.2</b>	<b>Organização</b>	<b>18</b>
<b>2</b>	<b>FUNDAMENTAÇÃO TEÓRICA</b>	<b>20</b>
<b>2.1</b>	<b>Acessibilidade Web</b>	<b>20</b>
<b>2.1.1</b>	<i>Diretrizes de Acessibilidade de Conteúdo da Web (WCAG) 2.1</i>	<b>21</b>
<b>2.1.2</b>	<i>Modelo de Acessibilidade em Governo Eletrônico (eMAG) 3.1</i>	<b>22</b>
<b>2.2</b>	<i>Reúso de Software</i>	<b>23</b>
<b>2.3</b>	<i>Open Source</i>	<b>27</b>
<b>2.4</b>	<b>Considerações finais do capítulo</b>	<b>29</b>
<b>3</b>	<b>TRABALHOS RELACIONADOS</b>	<b>30</b>
<b>3.1</b>	<b>GAIA: uma proposta de guia de recomendações de acessibilidade <i>web</i> com foco em aspectos do autismo</b>	<b>30</b>
<b>3.2</b>	<b>Catálogo de práticas de acessibilidade: um apoio <i>online</i> voltado à acessibilidade da <i>Web</i></b>	<b>31</b>
<b>3.3</b>	<b>Guia de recomendações de acessibilidade para produção de interfaces <i>web</i></b>	<b>32</b>
<b>3.4</b>	<b>Análise Comparativa</b>	<b>33</b>
<b>4</b>	<b>METODOLOGIA</b>	<b>35</b>
<b>4.1</b>	<b>Pesquisa na literatura</b>	<b>35</b>
<b>4.2</b>	<b>Identificação das diretrizes</b>	<b>35</b>
<b>4.3</b>	<b>CrITÉRIOS de seleção</b>	<b>36</b>
<b>4.4</b>	<b>Seleção das diretrizes</b>	<b>36</b>
<b>4.5</b>	<b>Priorização das Diretrizes</b>	<b>37</b>
<b>4.6</b>	<b>Desenvolvimento dos componentes de funcionalidades de acessibilidade</b>	<b>39</b>
<b>4.7</b>	<b>Arquitetura de Navegação</b>	<b>40</b>
<b>4.8</b>	<b>Desenvolvimento do AccessHub</b>	<b>40</b>
<b>4.9</b>	<b>Publicação</b>	<b>40</b>
<b>4.10</b>	<b>Participantes e Materiais</b>	<b>40</b>

<b>4.11</b>	<b>Avaliação</b>	42
<b>5</b>	<b>ACCESSHUB</b>	45
<b>5.1</b>	<b>Apresentação estrutural do AccessHub</b>	45
<b>5.1.1</b>	<b>Front-end do projeto</b>	46
5.1.1.1	Node.js	46
5.1.1.2	Vite	46
5.1.1.3	React	46
5.1.1.4	React Router	47
5.1.1.5	Chakra-UI	47
5.1.1.6	TypeScript	47
5.1.1.7	Componente de apresentação dos códigos fontes	48
<b>5.1.2</b>	<b>Interfaces do AccesHub e as suas funções</b>	48
5.1.2.1	Tela de Home Page	48
5.1.2.2	Tela de Introdução	49
5.1.2.3	Tela de Componentes	49
5.1.2.4	Página de Créditos das Imagens	49
5.1.2.5	Página de Error 404	51
5.1.2.6	Responsividade e Dark Mode	51
<b>5.1.3</b>	<b>Utilização dos Componentes de Funcionalidades</b>	52
5.1.3.1	Componente de HTML Inicial	52
5.1.3.2	Componente de Recomendações	53
5.1.3.3	Componente de Cabeçalhos	53
5.1.3.4	Componente de Pular Navegação	54
5.1.3.5	Componente de Breadcrumbs	54
5.1.3.6	Componente de Tabelas	54
5.1.3.7	Componente de Formulários	54
5.1.3.8	Componente de Links	55
5.1.3.9	Componente de Imagens	55
5.1.3.10	Componente de Vídeos	55
5.1.3.11	Componente de Áudios	56
5.1.3.12	Componente para Mudar o Tamanho da Fonte	56
5.1.3.13	Componente para Alterar o Contraste do Site	56

5.1.3.14	<i>Componente para Ouvir a Descrição de uma Imagem</i>	57
5.1.3.15	<i>Componente para a Navegação pelo Teclado</i>	57
5.1.3.16	<i>Componente de Máscara de Leitura</i>	57
5.1.4	<b><i>Exemplo Prático de Utilização</i></b>	57
5.1.5	<b><i>Como contribuir com o projeto</i></b>	59
6	<b>AVALIAÇÃO DO ACCESSHUB</b>	61
6.1	<b>Auditoria individual do AccessHub</b>	61
6.2	<b>Perfil técnico dos participantes</b>	63
6.3	<b>Resultados do desafio técnico com os desenvolvedores</b>	63
6.4	<b>Questionário TAM</b>	67
6.5	<b>Perguntas abertas</b>	69
7	<b>CONCLUSÕES E TRABALHOS FUTUROS</b>	73
	<b>REFERÊNCIAS</b>	75
	<b>APÊNDICES</b>	79
	<b>APÊNDICE A–ROTEIRO DO DESAFIO TÉCNICO</b>	79
A.1	<b><i>Landing Page para a Venda de um Telescópio</i></b>	79
A.1.1	<b><i>Desafio</i></b>	79
A.1.2	<b><i>Passo a passo</i></b>	80
A.1.3	<b><i>Entrega do desafio</i></b>	80
A.1.4	<b><i>Feedback</i></b>	80
	<b>APÊNDICE B–TERMO DE CONSENTIMENTO</b>	81
	<b>APÊNDICE C–QUESTIONÁRIO DE PERFIL</b>	83
	<b>APÊNDICE D–PERCEPÇÃO DO PROJETO + QUESTIONÁRIO TAM</b>	85



## 1 INTRODUÇÃO

É importante garantir o desenvolvimento de conteúdos ou serviços acessíveis na internet, para que estejam disponíveis para todas as pessoas. Para isso, a adoção de práticas de acessibilidade na *Web* visa tornar os *sites* mais acessíveis e utilizáveis pelo maior número possível de pessoas, independentemente de seus conhecimentos, características ou habilidades técnicas (CAMPOVERDE-MOLINA *et al.*, 2020). Entretanto, o desenvolvimento de *sites Web* é, geralmente, realizado sem oferecer o suporte necessário para que pessoas com limitações possam acessá-los sem dificuldades. Desta forma, o *World Wide Web Consortium* (W3C)<sup>1</sup>, criou a *Web Accessibility Initiative* (WAI)<sup>2</sup>, desenvolvendo assim as *Web Content Accessibility Guidelines* (WCAG)<sup>3</sup> que descrevem os requisitos para tornar qualquer conteúdo digital acessível a diversos usuários (ALAJARMEH, 2021).

A WCAG tem como objetivo fornecer um padrão referente à acessibilidade para conteúdos na *Web*, sendo neutra em termos de tecnologia, independente da plataforma utilizada (ALAJARMEH, 2021). A WAI, além das diretrizes do WCAG, disponibiliza outros dois guias de recomendações: o *Authoring Tool Accessibility Guidelines* (ATAG)<sup>4</sup>, que busca ajudar os desenvolvedores a projetar ferramentas que produzam conteúdo *Web* acessível, fornecendo recomendações para a criação de *interfaces* de autoria acessíveis (ACOSTA *et al.*, 2020). Assim como o *User Agent Accessibility Guideline* (UAAG)<sup>5</sup>, que orienta os desenvolvedores na concepção de agentes de usuário que incluem navegadores, extensões de navegador, reprodutores de mídia, leitores e outros aplicativos que processam conteúdo da *Web* (SOUSA *et al.*, 2020).

No Brasil, foi criado o Modelo de Acessibilidade em Governo Eletrônico (eMAG)<sup>6</sup>. Este documento contém um conjunto de recomendações para garantir a acessibilidade dos *sites* e portais do governo brasileiro, auxiliando a desenvolver o conteúdo digital do governo federal para que seja acessível a todos (MARTIN; PIOVESAN, 2019).

Portanto, implementar funcionalidades de acessibilidade em projetos de desenvolvimento de *websites*, significa que pessoas com algum tipo de deficiência podem fazer uso da *Web* nas mesmas condições que o restante das pessoas. Aplicar as práticas de acessibilidade permite que os usuários possam perceber, compreender, navegar e interagir com a *Web*, assim

<sup>1</sup> Disponível em: <https://www.w3.org/>

<sup>2</sup> Disponível em: <https://www.w3.org/WAI/>

<sup>3</sup> Disponível em: <https://www.w3.org/WAI/standards-guidelines/wcag/>

<sup>4</sup> Disponível em: <https://www.w3.org/WAI/standards-guidelines/atag/>

<sup>5</sup> Disponível em: <https://www.w3.org/WAI/standards-guidelines/uaag/>

<sup>6</sup> Disponível em: <https://emag.governoeletronico.gov.br/>

como também beneficiam os idosos que tendem a observar a diminuição de suas habilidades como resultado da idade (CAMPOVERDE-MOLINA *et al.*, 2020).

Para interagir com páginas da *Web*, usuários com limitações geralmente utilizam Tecnologias Assistivas (TA), sendo estes produtos ou sistemas que os apoiam e auxiliam, melhorando a comunicação, interação, mobilidade e execução de muitas atividades diárias em casa, escola e trabalho (KRASNIQI *et al.*, 2022). Desta forma, pessoas com deficiência visual e intelectual contam com um leitor de tela, uma tecnologia assistiva de propósito especial que permite aos usuários ouvir o conteúdo da página *Web* de forma linear, navegando pelo *Document Object Model* (DOM) da página e lendo em voz alta o conteúdo em texto trazido pelo dispositivo (FERDOUS *et al.*, 2021). Já para os usuários surdos, que dependem de funcionalidades dos próprios *sites*, soluções como a conversão da fala ou do som em texto escrito para legendas e intérpretes, são algumas das recomendações sugeridas, facilitando a compreensão de determinados conteúdos (SOUSA *et al.*, 2020).

Porém, para que essas tecnologias funcionem de forma adequada é necessário que os *websites* sejam desenvolvidos seguindo os padrões semânticos disponibilizados para o *Hyper Text Markup Language* (HTML)<sup>7</sup>, sendo esta uma linguagem de marcação que apresenta funcionalidades que auxiliam na interação e navegação de seus usuários, contribuindo no suporte ao desenvolvimento de páginas *web* acessíveis (MDN Web Docs, 2022b). Da mesma forma, o *Cascading Style Sheets* (CSS)<sup>8</sup> e o *JavaScript*<sup>9</sup>, quando usados corretamente, permitem aos desenvolvedores a criação de funcionalidades que interagem com o conteúdo complexo de um *site*, permitindo que esse seja o mais acessível possível, entretanto, podendo ajudar ou prejudicar a acessibilidade, dependendo de como são utilizadas (MDN Web Docs, 2022a).

Entretanto, podemos observar que existem dificuldades de tornar os projetos de *websites* acessíveis, seja pela falta de conhecimento técnico sobre programação acessível por parte dos desenvolvedores, problemas relacionados à falta de tempo do time, um suposto aumento no orçamento do projeto ou a falta de interesse por parte dos programadores (SILVA; RODRIGUES, 2018).

Mesmo com a existência das normas e diretrizes de acessibilidade na *Web*, muitos programadores não conseguem entender como aplicá-las em seus projetos, desta forma a falta de conhecimento tem impacto direto nas demais dificuldades levantadas, sendo esses recursos mal

<sup>7</sup> Disponível em: <https://developer.mozilla.org/en-US/docs/Web/HTML>

<sup>8</sup> Disponível em: <https://developer.mozilla.org/en-US/docs/Web/CSS>

<sup>9</sup> Disponível em: <https://developer.mozilla.org/en-US/docs/Web/JavaScript>

compreendidos por parte dos desenvolvedores *Web* (SILVA; RODRIGUES, 2018).

Portanto, oferecer exemplos de implementações como referência, principalmente, para desenvolvedores inexperientes, auxilia no processo de programação de funcionalidades e recursos que estejam em conformidade com as recomendações de acessibilidade, podendo ser reutilizados, diminuindo o retrabalho e agilizando o processo de desenvolvimento.

Desta forma, considerando o contexto apresentado, este trabalho tem como objetivo desenvolver um *website* a ser usado como um guia sobre funcionalidades de acessibilidade para desenvolvedores *Web*, a fim de prover: (i) apresentação de padrões e materiais de referência para ajudar na construção de *websites* acessíveis, (ii) um conjunto de recomendações de funcionalidades, e (iii) exemplos *open source* para acessibilidade *Web* que possam ser facilmente acessados, compreendidos e compartilhados.

## 1.1 Objetivos

### 1.1.1 Objetivo Geral

Desenvolver um guia para auxiliar na implementação de funcionalidades de acessibilidade, possibilitando a apresentação de padrões e materiais de suporte para a construção de páginas *Web* acessíveis por desenvolvedores.

### 1.1.2 Objetivos Específicos

Os objetivos específicos deste trabalho são:

- a) Desenvolver material de referência para o suporte na implementação das funcionalidades de acessibilidade *Web* por desenvolvedores.
- b) Sugerir um conjunto de recomendações para apoiar a implementação das funcionalidades de acessibilidade *Web*.
- c) Disponibilizar exemplos de código *open source* para apoiar a implementação das funcionalidades de acessibilidade *Web*.

## 1.2 Organização

Desta maneira, o restante deste trabalho está dividido nos seguintes capítulos. O Capítulo 2 apresenta os conceitos necessários para esclarecimento e compreensão do trabalho. O

Capítulo 3 apresenta os trabalhos relacionados a esse projeto, além das semelhanças e diferenças com o presente trabalho. No Capítulo 4, a metodologia utilizada é apresentada de maneira mais detalhada. No Capítulo 5 temos a apresentação sobre o desenvolvimento da proposta deste trabalho e as suas funcionalidades. No Capítulo 6 são descritos os processos de avaliação deste projeto. Por fim, o Capítulo 7 apresenta os resultados, as conclusões e sugestões de trabalhos futuros, a partir deste trabalho.

## 2 FUNDAMENTAÇÃO TEÓRICA

Nesta seção são apresentados os conceitos sobre Acessibilidade *Web*, Reúso de *Software* e *Open Source*, necessários para o entendimento e desenvolvimento deste trabalho.

### 2.1 Acessibilidade Web

O principal objetivo da acessibilidade *web* é tornar as páginas *web* mais fáceis de se navegar e ler, para todos os usuários, independentemente das deficiências que estão enfrentando, são práticas inclusivas para remover as barreiras de acesso à *web*, atendendo usuários com deficiências visuais, motoras, auditivas, cognitivas ou intelectuais (ISMAIL; KUPPUSAMY, 2019).

Ou seja, durante o processo de concepção, desenvolvimento e publicação de conteúdo *web*, devem ser consideradas as necessidades das pessoas com deficiência temporária, permanente, moderada e grave, caso contrário, em alguns casos, a falta de acessibilidade nos *sites* pode privar o acesso desses usuários à informação (ACOSTA *et al.*, 2020).

A acessibilidade da *web* também oferece benefícios para pessoas sem deficiência. Esse grupo pode incluir pessoas que usam óculos, aquelas que usam dispositivos com telas pequenas, idosos, aqueles localizados em ambientes que exigem silêncio, aqueles em ambientes ruidosos e/ou aqueles que usam dispositivos com conexões de *internet* lentas ou largura de banda inadequada (Web Accessibility Initiative, 2022).

Portanto, o desenvolvimento de *sites* deve estar em conformidade com padrões de acessibilidade para alcançar o conceito de que todo *site* deve ser projetado, desenvolvido e editado corretamente, para que todos os tipos de usuários possam obter acesso igual e sem barreiras às informações e funcionalidades (ISMAIL; KUPPUSAMY, 2019).

Assim, a W3C, que é uma comunidade internacional onde as organizações-membro trabalham para a criação e desenvolvimento de padrões para a *web*, criou a WAI, que é responsável por criar diretrizes e especificações sobre acessibilidade *web*.

Desta forma, no que diz respeito à criação de conteúdo *web* acessível, o WAI propôs padrões de acessibilidade que visam fornecer a possibilidade de acesso a conteúdo da *web* para usuários com deficiências, ofertando documentações, recursos, técnicas, códigos e possíveis falhas, sendo proposto três grupos de diretrizes:

- **WCAG**: Que fornece recomendações específicas de como aumentar a acessibilidade do

- conteúdo em um *website* (NUÑEZ *et al.*, 2019);
- **ATAG**: Que fornece recomendações para a criação de ferramentas de autoria acessíveis para que criadores de conteúdo com deficiência possam usar as ferramentas de forma independente (KULKARNI, 2019);
  - **UAAG**: Que são diretrizes para a avaliação da acessibilidade das tecnologias de agentes do usuário, como navegadores da *Web*, *players* de mídia, visualizadores de documentos e ferramentas de tecnologia que pessoas com deficiência usam para acessar informações (KULKARNI, 2019).

No contexto nacional, o governo brasileiro buscando promover a inclusão digital da população, criou o eMAG, estabelecido como o guia do governo federal, a partir dos modelos internacionais, para o desenvolvimento e adaptação de conteúdo digital, garantindo que seja acessível a todos (Portal do Governo Brasileiro, 2022).

Desse modo, o trabalho proposto vai utilizar como base as diretrizes estabelecidas pela WCAG e as recomendações propostas pelo eMAG, visto que ambas propõem orientações de como desenvolver conteúdo para a *web* de forma acessível, fazendo com que os componentes desenvolvidos estejam em conformidade com os padrões estabelecidos internacionalmente, da mesma forma que atendam às necessidades brasileiras.

### **2.1.1 Diretrizes de Acessibilidade de Conteúdo da Web (WCAG) 2.1**

A WCAG 2.1 é uma documentação que explica como tornar as informações de um conteúdo na *web* mais acessíveis para pessoas com deficiência, ela destina-se principalmente a: Desenvolvedores de conteúdo da *Web*; Desenvolvedores de ferramentas de autoria da *Web*; Desenvolvedores de ferramentas de avaliação de acessibilidade na *Web* ou qualquer um que deseja ou precisa de um padrão para acessibilidade na *Web* (WAI, 2022a). O documento está dividido em 13 diretrizes agrupadas em 4 princípios: Percebido, Operável, Compreensível e Robusto, conforme apresentado no Quadro 1.

Estes princípios estabelecem a base necessária para que qualquer pessoa acesse e use o conteúdo da *Web*, que seja: Perceptível - Todos os usuários devem ser capazes de perceber o conteúdo de forma visual, sonora ou tátil; Operável - Os usuários devem ser capazes de usar e navegar pelos componentes da interface; Compreensível - Tanto o conteúdo quanto os controles da *interface* para seu gerenciamento devem ser compreensíveis para o usuário; Robusto - O conteúdo deve ser robusto para ser interpretado de forma confiável pelo número mais significativo

Quadro 1 – Princípios e diretrizes da WCAG 2.1

Princípio	Diretrizes
<b>1. Perceptível</b>	1.1. Fornecer alternativas de texto para conteúdo não textual. 1.2. Fornecer legendas e outras alternativas para multimídia. 1.3. Criar conteúdo que possa ser apresentado de diversas formas, inclusive por tecnologias assistivas, sem perder o sentido. 1.4. Tornar mais fácil para os usuários ver e ouvir o conteúdo.
<b>2. Operável</b>	2.1. Disponibilize todas as funcionalidades a partir de um teclado. 2.2. Dê aos usuários tempo suficiente para ler e usar o conteúdo. 2.3. Não use conteúdo que cause convulsões ou reações físicas. 2.4. Ajude os usuários a navegar e encontrar conteúdo. 2.5. Facilite o uso de outras entradas além do teclado.
<b>3. Compreensível</b>	3.1. Tornar o texto legível e compreensível. 3.2. Faça o conteúdo aparecer e operar de maneira previsível. 3.3. Ajude os usuários a evitar e corrigir erro.
<b>4. Robusto</b>	4.1. Maximize a compatibilidade com ferramentas de usuário atuais e futuras.

Fonte: Adaptado de (WAI, 2022b).

de usuários com tecnologias atuais e futuras (SALVADOR-ULLAURI *et al.*, 2020).

As diretrizes, ao todo, contém 87 critérios de sucesso testáveis divididos em três níveis de sucesso: A, AA e AAA. O nível A corresponde ao nível mais baixo, mas a conformidade garante acessibilidade mínima na *web*. O nível AA é um nível intermediário, supondo que os critérios de sucesso A e AA sejam atendidos. O nível avançado é AAA, e os critérios de sucesso para atingir este nível também atendem aos níveis A e AA. Desta forma, em alguns casos, vários métodos são recomendados para atender aos critérios de sucesso e o mesmo método pode ser usado para atender a vários critérios de sucesso, assim o W3C incentiva o uso da mais recente versão do WCAG ao desenvolver ou atualizar *websites* com acessibilidade.

### 2.1.2 Modelo de Acessibilidade em Governo Eletrônico (eMAG) 3.1

Como forma de garantir a acessibilidade aos *sites* governamentais, foi criado o Modelo de Acessibilidade em Governo Eletrônico (eMAG), uma versão especializada do WCAG voltado para o governo brasileiro, que busca orientar e auxiliar no desenvolvimento e adaptação do conteúdo digital para a população. As recomendações do eMAG permitem que a implementação

da acessibilidade digital seja conduzida de forma padronizada, de fácil implementação, coerente com as necessidades brasileiras e em conformidade com os padrões internacionais (Portal do Governo Brasileiro, 2022).

Para a criação do eMAG, optou-se por descartar os níveis de prioridade A, AA e AAA, pois o padrão é para páginas governamentais e não permite exceções na adesão às recomendações, incluindo assim a seção chamada “Padronização de acessibilidade nas páginas do governo federal”, com o intuito de padronizar elementos de acessibilidade que devem existir em todos os *sites* e portais do governo (Portal do Governo Brasileiro, 2022).

Porém o eMAG, conforme mostrado no Quadro 2, possui quarenta e cinco recomendações apresentando descrições e exemplos de acessibilidade divididas em seis seções: marcação, comportamento, conteúdo/informação, apresentação/*design*, multimídia e formulários (OLIVEIRA *et al.*, 2020).

O eMAG foi instituído pela portaria nº. 3 de maio de 2007 tornando-o obrigatório para os portais da Internet da Administração Pública Federal, desta forma, Souza *et al.* (2016) ressaltam que tais recomendações tratam de conteúdos visuais e alternativas de áudio que sejam claras e compreensíveis e que assegurem o controle do usuário sobre sua navegação independentemente de suas capacidades físicas/motoras, perceptivas e culturais ou sociais, torná-los mais acessíveis ao maior número de pessoas possível. Desta forma, ao incorporar o eMAG neste trabalho, é demonstrado a importância e relevância do alinhamento do AccessHub com as diretrizes e regulamentações nacionais, que buscam contribuir para a promoção de uma *Web* mais inclusiva e acessível para todos.

## 2.2 Reúso de *Software*

O reúso de *software* é o caminho para a criação de projetos a partir de componentes existentes, de acordo com Sandhu e Bath (2021), o reúso de *software* foi discutido pela primeira vez na conferência da *North Atlantic Treaty Organization* (NATO), *Software Engineering Conference* em 1968. O reúso de *software* ajuda na redução tanto no custo quanto no tempo de desenvolvimento de um *software*, aumentando a produtividade, confiabilidade e qualidade do projeto.

Além disso, a reutilização de *software* pode descrever o uso de componentes pré-existentes para desenvolver novos sistemas e manter produtos antigos (sistemas legados), de tal forma que um sistema atualizado pode ser visto como uma nova versão do sistema usado



Quadro 2 – Seções e Recomendações do eMag

Seções	Recomendações
<b>1. Marcação</b>	1.1. Respeitar os Padrões Web. 1.2. Organizar o código HTML de forma lógica e semântica. 1.3. Utilizar corretamente os níveis de cabeçalho. 1.4. Ordenar de forma lógica e intuitiva a leitura e tabulação. 1.5. Fornecer âncoras para ir direto a um bloco de conteúdo. 1.6. Não utilizar tabelas para diagramação. 1.7. Separar links adjacentes. 1.8. Dividir as áreas de informação. 1.9. Não abrir novas instâncias sem a solicitação do usuário.
<b>2. Comportamento (DOM)</b>	2.1. Disponibilizar todas as funções da página via teclado. 2.2. Garantir que os objetos programáveis sejam acessíveis. 2.3. Não criar páginas com atualização automática periódica. 2.4. Não utilizar o redirecionamento automático de páginas. 2.5. Fornecer alternativa para modificar limite de tempo. 2.6. Não incluir situações com intermitência de tela. 2.7. Assegurar o controle do usuário sobre as alterações do conteúdo.
<b>3. Conteúdo/Informação</b>	3.1. Identificar o idioma principal da página. 3.2. Informar mudança de idioma no conteúdo. 3.3. Oferecer um título descritivo e informativo à página. 3.4. Informar o usuário sobre sua localização na página. 3.5. Descrever links clara e sucintamente. 3.6. Fornecer alternativa em texto para as imagens do site. 3.7. Utilizar mapas de imagem de forma acessível. 3.8. Disponibilizar documentos em formatos acessíveis. 3.9. Em tabelas, utilizar títulos e resumos de forma apropriada. 3.10. Associar células de dados às células de cabeçalho. 3.11. Garantir a leitura e compreensão das informações. 3.12. Explicar siglas, abreviaturas e palavras incomuns.
<b>4. Apresentação/Design</b>	4.1. Valor de contraste mínimo entre plano de fundo e primeiro plano. 4.2. Não utilizar apenas cor ou outras características sensoriais para diferenciar elementos. 4.3. Permitir redimensionamento sem perda de funcionalidade. 4.4. Que o elemento com foco seja visualmente evidente.
<b>5. Multimídia</b>	5.1. Fornecer alternativa para vídeo. 5.2. Fornecer alternativa para áudio. 5.3. Oferecer audiodescrição para vídeo pré-gravado. 5.4. Fornecer controle de áudio para som. 5.5. Fornecer controle de animação.
<b>6. Formulário</b>	6.1. Alternativa em texto para os botões de imagem de formulários. 6.2. Associar etiquetas aos seus campos. 6.3. Estabelecer uma ordem lógica de navegação. 6.4. Não provocar automaticamente alteração no contexto. 6.5. Fornecer instruções para entrada de dados. 6.6. Identificar e descrever erros de entrada de dados e confirmar o envio das informações. 6.7. Agrupar campos de formulário. 6.8. Usar estratégias de segurança específicas ao invés de <i>CAPTCHA</i> .

Fonte: Adaptado de (Portal do Governo Brasileiro, 2022).

anteriormente (BARROS-JUSTO *et al.*, 2019). Deste modo, para sua implementação, existem dois grandes processos: O desenvolvimento de componentes predefinidos e o desenvolvimento para a reutilização desses componentes na criação de novos sistemas (RAJAKUMARI, 2020).

O desenvolvimento para reuso consiste na criação de componentes de *software* genéricos, flexíveis e de alta qualidade, determinando como eles vão ser construídos e a sua utilização em outros projetos (TOFFOLON; DAKHLI, 1998). Já o desenvolvimento de *software* com reuso faz parte da estratégia de reduzir o atraso no desenvolvimento de um sistema, permitindo modificações ou adaptações em vez de desenvolver as suas funcionalidades (TOFFOLON; DAKHLI, 1998).

Desta maneira, os autores Anasuoidei e Ojekudo (2021) ressaltam que, na engenharia de *software*, o método de reuso de *software* pode ser caracterizado de diferentes formas, sendo elas:

- a) **Reuso de Sistemas:** Essa abordagem é o reuso completo, que consiste em várias aplicações que podem ser reutilizadas em um sistema, assim um sistema constituído de outros sistemas é usado para gerenciar dois ou mais sistemas de componentes;
- b) **Reuso de Aplicativos:** Essa forma de reutilização envolve a adaptação do sistema de *software* para atender às necessidades de diferentes clientes para que o código-fonte não seja afetado, sendo desenvolvido principalmente para o mercado geral e não para clientes específicos;
- c) **Reuso Baseado em Componentes:** Diferentes *softwares* são divididos em componentes únicos, sendo que esta forma utiliza um repositório onde esses componentes possam ser selecionados para o desenvolvimento de um novo sistema;
- d) **Reuso de Objetos e Funções:** Componentes de *software* que executam tarefas específicas, como funções matemáticas ou objetos de classe, podem ser reutilizados.

Muitas técnicas foram desenvolvidas para apoiar o planejamento da reutilização de *software*. A partir do trabalho de Sommerville (2011), o Quadro 3 descreve as técnicas de desenvolvimento de *software* para o reuso, essas técnicas aproveitam o fato de que os sistemas dentro do mesmo domínio de aplicação são semelhantes e têm potencial de reutilização.

Contudo, vale considerar qual técnica é a mais adequada para ser utilizada em

determinado contexto, dependendo do planejamento do projeto, dos requisitos do sistema que está em desenvolvimento e das tecnologias que serão utilizadas.

Quadro 3 – Abordagens de Reutilização de *Software*.

Abordagem	Descrição
<b>Padrões de arquitetura</b>	Padrões de arquitetura de <i>software</i> que oferecem suporte a tipos comuns de sistemas de aplicação são usados como base de aplicações.
<b>Padrões de projeto</b>	Abstrações genéricas que ocorrem em todas as aplicações são representadas como padrões de projeto, mostrando os objetos abstratos e concretos e as interações.
<b>Desenvolvimento baseado em componentes</b>	Sistemas são desenvolvidos através da integração de componentes (coleções de objetos) que atendem aos padrões de modelos e componentes.
<b>Framework de aplicações</b>	Coleções de classes abstratas e concretas são adaptadas e estendidas para criar sistemas de aplicação.
<b>Empacotamento de sistemas legados</b>	Sistemas legados são ‘empacotados’ pela definição de um conjunto de <i>interfaces</i> e acesso a esses sistemas legados por meio dessas <i>interfaces</i> .
<b>Sistemas orientados a serviços</b>	Sistemas são desenvolvidos pela ligação de serviços compartilhados, que podem ser fornecidos externamente.
<b>Linhas de produtos de <i>software</i></b>	Um tipo de aplicação é generalizado em torno de uma arquitetura comum para que esta possa ser adaptada para diferentes clientes.
<b>Reúso de produto COTS</b>	Sistemas são desenvolvidos pela configuração e integração de sistemas de aplicação existentes.
<b>Sistemas de ERP</b>	Sistemas de grande porte que sintetizam a funcionalidade e as regras de negócios genéricas são configurados para uma organização.
<b>Aplicações verticais configuráveis</b>	Sistemas genéricos são projetados para poderem ser configurados para as necessidades dos clientes de sistemas específicos.
<b>Bibliotecas de programas</b>	Bibliotecas de classe e funções que implementam abstrações comumente usadas são disponibilizadas para reúso.
<b>Engenharia dirigida a modelos</b>	O <i>software</i> é representado como modelos de domínio e modelos de implementação independentes. O código é gerado a partir desses modelos.
<b>Geradores de programas</b>	Um sistema gerador incorpora o conhecimento de um tipo de aplicação, e é usado para gerar sistemas nesse domínio a partir de um modelo de sistema fornecido pelo usuário.
<b>Desenvolvimento de <i>software</i> orientado a aspectos</b>	Quando o programa é compilado, os componentes compartilhados são integrados em uma aplicação em diferentes locais.

Fonte: Adaptado de (SOMMERVILLE, 2011).

Assim sendo, a técnica que será utilizada neste trabalho é a Biblioteca de Programas, que consiste em dados e código, que busca fornecer os serviços necessários para diferentes programas, incentivando a troca e compartilhamento de código entre diversos projetos (DABHADE *et al.*, 2016). Essa técnica apresenta um conjunto de propósitos que alcança o objetivo de qual-

quer rotina específica, devendo demonstrar um design universal, confiabilidade e estabilidade com a pretensão de que seu conteúdo seja benéfico para uma comunidade de usuários em geral (AHMARO *et al.*, 2014).

As Bibliotecas de Programas apresentam código e dados para serviços essenciais a programas independentes, onde as classes e funções da biblioteca empregam abstrações comumente usadas que estão disponíveis para reutilização (AHMARO *et al.*, 2014). Desta forma, os desenvolvedores *web* podem encontrar os códigos de funções para acessibilidade prontas para o reuso, sem a necessidade de importar *frameworks* de aplicação específicos para a solução de uma demanda existente, apresentando o código *open-source* das funcionalidades e a completa implementação delas em sua base. O Quadro 4 resume as principais vantagens das bibliotecas de programas.

Quadro 4 – Principais vantagens das bibliotecas de programas.

<b>Vantagens</b>	<b>Descrição</b>
<b>Melhora na qualidade</b>	As bibliotecas permitem que os elementos sejam construídos de acordo com comandos especificados por um programa específico.
<b>Reduz os erros no sistema</b>	A biblioteca de programas resolve os problemas de otimização que são substanciais.
<b>Aumenta a reutilização</b>	Classe e função das bibliotecas que empregam abstrações comumente usadas estão disponíveis para reutilização.
<b>Aumenta o compartilhamento de código e dados</b>	As bibliotecas abrangem código e dados que oferecem serviços essenciais a programas independentes.

Fonte: Adaptado de (AHMARO *et al.*, 2014).

### 2.3 *Open Source*

Criado pela *Open Source Initiative* (OSI)<sup>1</sup>, o termo *Open Source* significa que o *software* é livre para qualquer pessoa acessar, usar, modificar e compartilhar o código-fonte do programa (Open Source Initiative, 2022). Além disso, abordagens *Open Source* permitem que a comunidade aprimore ou melhore os resultados do desenvolvimento de um projeto, podendo se comparar aos *softwares* confidenciais e aos *softwares* privados (WOLFART *et al.*, 2019).

A colaboração existente entre usuários e desenvolvedores, juntamente com a disponibilidade de código-fonte, permite que programas de código aberto sejam usados com frequência para suportar aplicativos de rede críticos, incluindo *interfaces* de *Internet* e *Web*, que podem ser continuamente modificadas e aprimoradas (PRADO, 2012).

<sup>1</sup> Disponível em: <https://opensource.org/>

Porém, *open source* não significa que apenas o acesso ao código-fonte seja aberto, de acordo com a Open Source Initiative (2022) os termos de distribuição do *software open source* devem obedecer aos seguintes critérios:

1. **Redistribuição gratuita:** A licença não restringe nenhuma das partes de vender ou distribuir o *software*, e nenhum *royalties* ou outras taxas podem ser cobrados por tais vendas;
2. **Código Fonte:** O programa deve conter código-fonte e deve permitir sua distribuição, bem como de forma compilada. A ofuscação intencional do código-fonte não é permitida. Formas intermediárias, como a saída de um pré-processador ou tradutor, não são permitidas;
3. **Obras Derivadas:** A licença deve permitir modificações e trabalhos derivados, permitindo que sejam distribuídos sob os mesmos termos de licença do *software* original;
4. **Integridade do Código Fonte do Autor:** A licença pode restringir a distribuição do código-fonte de forma modificada somente se a licença permitir a distribuição de "arquivos de correção" com o código-fonte para modificar o programa em tempo de construção;
5. **Nenhuma discriminação contra pessoas ou grupos:** A licença não deve discriminar nenhum indivíduo ou grupo;
6. **Não Discriminação Contra Campos de Esforço:** A licença não deve restringir ninguém de usar o programa em um campo de trabalho específico;
7. **Distribuição de Licença:** Os direitos que acompanham o Programa se aplicam a todos que redistribuem o Programa sem a necessidade de tais partes exigirem licenças adicionais;
8. **A licença não deve ser específica para um produto:** Direitos associados ao programa não devem depender de o programa fazer parte de uma distribuição de *software* específica;
9. **A licença não deve restringir outro software:** A licença não deve impor restrições a outro *software* distribuído com o *software* licenciado;
10. **A licença deve ser neutra em termos de tecnologia:** Qualquer disposição da licença não deve ser baseada em nenhuma tecnologia separada ou estilo de interface.

As comunidades *open source* normalmente usam repositórios de *software* para arquivar vários projetos de *software* e seu código-fonte, tópicos em listas de discussões, documentação, relatórios de *bugs* entre outras informações, fazendo com que devido à grande quantidade de conteúdo informativo, esses repositórios ofereçam inúmeras oportunidades para compartilhar informações entre projetos (KAWAGUCHI *et al.*, 2004). Desta forma, a criação deste repositório

e a disponibilização do código aberto vai permitir que o projeto esteja acessível para toda a comunidade *open source*, fornecendo a possibilidade da mesma contribuir para o projeto.

## 2.4 Considerações finais do capítulo

Este capítulo abordou os conceitos necessários para o entendimento deste trabalho. Podemos compreender o que é Acessibilidade *Web*, destacando a sua importância no desenvolvimento de *websites* e apresentando as diretrizes de acessibilidade que vão ser utilizadas. Logo depois, foi apresentado o que é o Reúso de *Software* e seus benefícios, apoiando-se dele que utilizaremos a técnica de Biblioteca de Programas no processo de desenvolvimento deste projeto. Por fim, abordamos o conceito de *Open Source* para enfatizar a estratégia de compartilhamento do conteúdo que será desenvolvido neste trabalho.

Contudo, recomendações de acessibilidade nem sempre conseguem atender as pessoas com deficiências se forem mal implementadas, desta forma o AccessHub tem o objetivo de prover um conjunto de funcionalidades de código aberto para a acessibilidade *web*, com a proposta de ser um guia de referência para qualquer projeto *web* que busque adotar as recomendações estabelecidas pela WCAG e eMAG, como também a qualquer desenvolvedor que queira contribuir com a iniciativa. Assim, as etapas de construção do AccessHub são detalhadas no Capítulo 4.

### 3 TRABALHOS RELACIONADOS

Neste capítulo, serão apresentados os trabalhos que contribuíram para o desenvolvimento desta pesquisa, contendo guias de recomendações ou catálogos de conceitos relacionados à acessibilidade *Web*. Este capítulo também apresenta as semelhanças e diferenças destes trabalhos com a solução desenvolvida no presente trabalho.

#### 3.1 GAIA: uma proposta de guia de recomendações de acessibilidade *web* com foco em aspectos do autismo

No trabalho de Britto (2016) são investigadas as ferramentas e soluções computacionais que são utilizadas de apoio para crianças com Transtorno do Espectro Autista (TEA), assim como para seus pais, terapeutas e professores. A autora ressalta que, se essas tecnologias não forem bem projetadas, elas podem levar a criação de barreiras na interação de pessoas com TEA que utilizam a tecnologia, em casos mais graves, causando desconforto e estresse desnecessário para esses usuários (BRITTO, 2016).

Deste modo, o objetivo principal do trabalho foi desenvolver o Guia de Acessibilidade de Interfaces Web com foco em aspectos do Autismo (GAIA), um *website*<sup>1</sup> que contém sugestões de *design* para acessibilidade *web*, projetado para ajudar no entendimento de como desenvolver *websites* adequados para as necessidades de usuários com TEA.

Para o desenvolvimento do GAIA, o projeto foi realizado em três etapas: a) levantamento bibliográfico exploratório, que gerou a primeira versão das diretrizes do GAIA; b) depois foi aplicada uma pesquisa *online* para mapear o conhecimento dos desenvolvedores sobre deficiências cognitivas, neuronais ou de aprendizagem; c) logo após foram realizadas entrevistas com pais de crianças com TEA para compreender empaticamente o valor da tecnologia.

Triangulando os resultados das três etapas, a autora chegou às seguintes conclusões: (i) existe a dificuldade de compreensão dos desenvolvedores por parte do material existente sobre acessibilidade *web*, possuindo desconhecimento sobre deficiências cognitivas; (ii) a partir das entrevistas, foi possível compreender o contexto de uso da tecnologia por crianças com TEA; (iii) nas entrevistas houve aspectos recorrentes de interação que não foram encontrados na literatura.

A relação do trabalho proposto com o trabalho realizado por Britto (2016) se dá pelo fato que ambos buscam desenvolver material de referência textual para desenvolvedores *web*,

---

<sup>1</sup> Disponível em: <https://gaia.wiki.br/>

a fim de oferecer suporte na construção de funcionalidades acessíveis. No entanto, enquanto o trabalho de Britto (2016) se concentra exclusivamente em sugestões textuais para o desenvolvimento de *sites* voltados para usuários com TEA, o trabalho proposto tem uma abordagem mais abrangente, apresentando exemplos *open source* de funcionalidades de acessibilidade para diferentes tipos de deficiências.

### **3.2 Catálogo de práticas de acessibilidade: um apoio *online* voltado à acessibilidade da *Web***

Neste trabalho, Hargreaves (2021) ressalta que a acessibilidade continua sendo uma questão subestimada para muitos desenvolvedores *web*, onde sua falta leva à criação de barreiras no acesso de informações e na interação com serviços que muitas vezes são necessários para um grande número de usuários. Assim, o autor buscou investigar melhores fontes de conhecimento e treinamento, necessárias para que desenvolvedores de *software* criem soluções *web* mais acessíveis.

Desta forma, foi proposto o desenvolvimento de um apoio *online*<sup>2</sup>, que promova conceitos básicos sobre acessibilidade *web*, tendo como referência literatura especializada sobre o tema, considerando como público-alvo os desenvolvedores de *software*.

Em seu trabalho, o autor definiu duas metodologias de pesquisa, sendo elas: a pesquisa bibliográfica e depois a pesquisa-ação. A pesquisa bibliográfica, segundo o autor, permite a aquisição de conhecimento, a partir de referências já analisadas e publicadas, com o objetivo de estabelecer uma base teórica sólida para o tema, já a pesquisa-ação, por sua vez, é uma forma de pesquisa baseada na ação real para a análise dos resultados, fornecendo mecanismos para corrigir e melhorar as soluções sugeridas durante a execução do trabalho (HARGREAVES, 2021).

Para a elaboração do projeto, foi necessário o aprofundamento teórico para compreender e definir com mais precisão termos importantes, as principais normas técnicas e legais relacionadas à acessibilidade na atualidade.

O trabalho proposto assemelha-se aos objetivos de Hargreaves (2021), pois ambos buscam apresentar material introdutório à acessibilidade *web*, disponibilizando exemplos de código *open-source* de acesso rápido para *websites*. Porém, o trabalho proposto tem como objetivo disponibilizar exemplos de funcionalidades em diferentes cenários de uso, onde os

---

<sup>2</sup> Disponível em: <https://hargre.github.io/a11y-catalog/>



desenvolvedores podem escolher quais se adaptam melhor aos seus projetos.

### 3.3 Guia de recomendações de acessibilidade para produção de interfaces *web*

No trabalho de Medeiros (2021) é ressaltado que apesar de os avanços tecnológicos propiciarem uma maior inclusão digital, usuários que apresentam deficiência visual ainda necessitam de atenção quando o assunto são as páginas *web*, sendo necessário haver maior preocupação por parte das empresas e desenvolvedores de *softwares* para o desenvolvimento de *interfaces* acessíveis para usuários com deficiência. Desta forma, tendo em mente que a maneira pela qual usuários com deficiência visual acessam as informações contidas nas páginas *web* é distinta dos usuários com visão normal, é possível afirmar que é necessária atenção dos desenvolvedores no momento do desenvolvimento de *softwares* para que sejam construídas *interfaces* que suportem a utilização confortável por esse público (MEDEIROS, 2021).

O objetivo geral deste trabalho foi o desenvolvimento de um guia com recomendações de acessibilidade para páginas *web*, de maneira a auxiliar profissionais da área de desenvolvimento de *software* na elaboração e adaptação de *interfaces* a partir das práticas relacionadas à acessibilidade *web* para pessoas cegas (MEDEIROS, 2021).

Em relação à metodologia escolhida por Medeiros (2021), a pesquisa bibliográfica foi aplicada, para corroborar com a fundamentação dos principais conceitos utilizados na presente monografia. O projeto foi aplicado em dois momentos: no primeiro momento, um questionário foi realizado para entender como é a realidade e as dificuldades de pessoas cegas no acesso à páginas *web*. Em um segundo momento, após a construção do guia, foi aplicado, como instrumento para a validação do guia de acessibilidade *web*, por meio de questionamentos enviados para os participantes da pesquisa. O Guia de Acessibilidade *Web* (GAW) contempla 28 recomendações, divididas em 4 categorias: multimídia computacional, funcionalidades, formulários e acesso a conteúdo, de modo a possibilitar a organização e a busca dos conteúdos (MEDEIROS, 2021).

Como é possível identificar, o trabalho realizado pelo autor tem foco nas funcionalidades de acessibilidade para usuários com deficiência visual, mas sem apresentar qualquer protótipo de *interface* ou *website*. Assim, as principais diferenças entre o trabalho realizado por Medeiros (2021) e o proposto neste trabalho são a disponibilização de código *open source* e a expansão do catálogo de recomendações para as demais deficiências, objetivando assim a criação de um guia mais completo.

### 3.4 Análise Comparativa

Esta seção apresenta as principais características encontradas, mostrando uma análise comparativa entre os trabalhos apresentados neste capítulo e a solução proposta neste trabalho. O Quadro 5 apresenta um resumo desta análise, destacando os critérios de comparação:

- a) **Desenvolvimento de material para referência:** Utilizado para identificar se o trabalho desenvolvido apresenta material de referência que auxilie no entendimento das soluções de funcionalidades para os desenvolvedores *web*.
- b) **Exemplos *open source* de código fonte:** Verifica se a solução proposta disponibiliza exemplos *open source* de funcionalidades de acessibilidade.
- c) **Sugestão de recomendações:** Identifica se a solução proposta apresenta um conjunto de recomendações de funcionalidades para acessibilidade *web*.
- d) **Apresentar um protótipo de *interface* ou *website*:** Verifica se a solução proposta apresenta algum tipo de *interface* ou *website* para a disponibilização de seu trabalho.
- e) **Sugestão de funcionalidades em diferentes cenários:** Verifica se o trabalho proposto disponibiliza exemplos de funcionalidades em diferentes cenários de uso.
- f) **Abrange tipos de deficiência:** Utilizado para identificar se o trabalho desenvolvido se propõe em apresentar conteúdo de funcionalidades para diferentes grupos de pessoas com deficiências.

Quadro 5 – Análise comparativa entre os trabalhos relacionados e este trabalho.

Critérios	Britto (2016)	Hargreaves (2021)	Medeiros (2021)	Este trabalho
Desenvolvimento de material para referência	SIM	SIM	SIM	SIM
Sugestão de recomendações	SIM	SIM	SIM	SIM
Exemplos <i>open-sources</i> de código fonte	NÃO	SIM	NÃO	SIM
Apresenta um protótipo de <i>interface</i> ou <i>website</i>	SIM	SIM	NÃO	SIM
Sugestão de funcionalidades em diferentes cenários	NÃO	NÃO	NÃO	SIM
Abrange demais tipos de deficiência	NÃO	SIM	NÃO	SIM

Fonte: Elaborado pelo autor.

Todos os trabalhos apresentados neste capítulo propõem o desenvolvimento de material para referência como também a sugestão de recomendações de funcionalidades de acessibilidade, isso demonstra a preocupação dos autores no desenvolvimento de um conteúdo

que possa auxiliar os desenvolvedores de *software* em seus projetos.

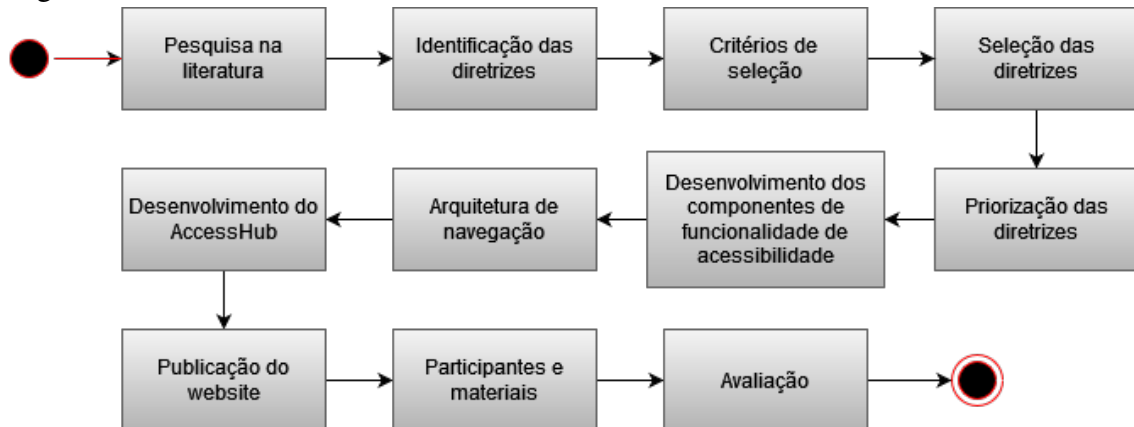
Com exceção do trabalho de Medeiros (2021), cujo foco é na criação de um guia de acessibilidade, todos os demais trabalhos, assim como o presente trabalho apresentam como solução um *website* para a disponibilização de seus respectivos conteúdos. Porém, apenas o trabalho de Hargreaves (2021) e o presente trabalho, buscam disponibilizar exemplos *open source* de código fonte sobre funcionalidades de acessibilidade *web*.

Da mesma forma, com exceção do trabalho de Medeiros (2021), assim como o presente trabalho, ambos apresentam conteúdo sobre como implementar funcionalidades que se aplicam a mais de um tipo de deficiência. Porém, diferentemente dos trabalhos de Britto (2016), Hargreaves (2021) e Medeiros (2021), este trabalho além de abranger todos os outros critérios propostos, permite que se obtenha sugestões e funcionalidades a partir de código fonte para diferentes cenários, assim os desenvolvedores podem utilizar o conteúdo e aplicar de maneira personalizada aos objetivos de seus projetos.

## 4 METODOLOGIA

Este capítulo apresenta as etapas que foram seguidas para alcançar os objetivos deste trabalho. A Figura 1 resume essas etapas e as próximas sessões descrevem cada uma delas.

Figura 1 – Fluxo de atividades.



Fonte: Elaborado pelo autor.

### 4.1 Pesquisa na literatura

Para a execução deste trabalho, foi realizada uma pesquisa exploratória que segundo Devezer *et al.* (2020), está relacionada à geração e teste de hipóteses, que podem levar a novas descobertas. Desta forma, com o intuito de identificar e selecionar as diretrizes de acessibilidade que melhor se enquadram nos objetivos deste projeto, foi realizada uma pesquisa no *google* acadêmico com os seguintes termos: “acessibilidade *web*” e “*web accessibility*”, desta forma, a partir de trabalhos nacionais e internacionais foram identificados um conjunto de diretrizes que fornecem informações sobre o tema.

Com as informações obtidas foi possível identificar os *sites* oficiais dessas recomendações, o que proporcionou uma visão mais detalhada de cada uma e a elaboração de critérios para selecionar quais diretrizes serão utilizadas no projeto.

### 4.2 Identificação das diretrizes

A partir da pesquisa realizada no *Google* Acadêmico pelos termos citados, foram selecionados 13 trabalhos que mencionam as diretrizes de acessibilidade *web*. Assim, seguindo a linha de pesquisa, nos trabalhos de Alajarmeh (2021), Campoverde-Molina *et al.* (2020), Ismail e Kuppusamy (2019), Nuñez *et al.* (2019), Salvador-Ullauri *et al.* (2020) e Sousa *et al.* (2020)

foi possível identificar o conjunto de diretrizes do WCAG e suas características.

Em seguida, após ter encontrado o site da WAI, foi possível encontrar outras duas diretrizes relacionadas, sendo elas a ATAG e UAAG, desta forma, foi possível pesquisar e encontrar os trabalhos de Acosta *et al.* (2020), Kulkarni (2019) e Souza e Almeida (2021) que forneceram informações sobre as mesmas. Por fim, estudando um pouco mais sobre o tema em âmbito nacional, foi possível identificar nos trabalhos de Martin e Piovesan (2019), Oliveira *et al.* (2020), Silva e Rodrigues (2018) e Souza *et al.* (2016) que tratam das recomendações de acessibilidade do eMAG, criado para o governo brasileiro.

Como resultado, foi identificado um conjunto de quatro diretrizes de acessibilidade, sendo elas a WCAG, ATAG, UAAG e eMAG que foram descritas em detalhes no Capítulo 2.

### **4.3 Critérios de seleção**

Após a identificação das diretrizes de acessibilidade, foi elaborado um conjunto de critérios para realizar a seleção das recomendações que se enquadram com os objetivos propostos para esse projeto, uma vez que é importante estabelecer qual será o campo de pesquisa específico para qual o trabalho será elaborado.

Desta forma, os critérios escolhidos para essa avaliação serão: diretrizes que apoiam o desenvolvimento de conteúdo acessível para a *web*; se essas recomendações identificadas oferecem uma abrangência na construção de páginas *web* de forma geral, não estabelecendo domínios específicos sobre o tema; se essas diretrizes apresentam especificações das recomendações propostas; e por fim, se essas recomendações apresentam exemplos de código de como realizar suas implementações.

### **4.4 Seleção das diretrizes**

Com os critérios já estabelecidos, o passo seguinte é a realização da seleção das diretrizes conforme apresentado no Quadro 6. Para realizar a seleção, foi feita a identificação para saber se cada diretriz está em conformidade com os critérios propostos, se ela está parcialmente em conformidade ou se ela não atende aos critérios estabelecidos.

Assim, para o desenvolvimento do AccessHub foi utilizado as diretrizes do WCAG, visto que são normativas específicas para o desenvolvimento de *websites* internacionalmente, e o eMAG, para a adoção parcial dos padrões estabelecidos pelo governo federal, não se prendendo

Quadro 6 – Critérios de seleção das diretrizes.

<b>Critérios</b>	<b>eMAG</b>	<b>WCAG</b>	<b>ATAG</b>	<b>UAAG</b>
<b>Apoia o desenvolvimento de conteúdo acessível para a web.</b>	SIM	SIM	SIM	SIM
<b>Apresenta especificações das recomendações propostas.</b>	SIM	SIM	SIM	SIM
<b>Abrange no desenvolvimento de páginas web no geral.</b>	SIM	SIM	NÃO	NÃO
<b>Apresenta exemplos de como desenvolver as recomendações.</b>	SIM	Parcialmente	NÃO	NÃO

Fonte: Elaborado pelo autor.

às suas restrições, mas buscando aproveitar melhor as suas soluções, visto que as implementações das funcionalidades serão *open source*, permitindo que os desenvolvedores utilizarem o conteúdo como desejarem em seus projetos.

Desta forma, devido a ATAG ser especificamente para ferramentas de autoria e a UAAG que é específica para tecnologias e ferramentas de agente de usuário, que auxiliam pessoas com deficiência no acesso à informação, ambas não serão adotadas para o desenvolvimento desta pesquisa uma vez que não estão em conformidade com a proposta inicial do projeto.

#### 4.5 Priorização das Diretrizes

Após a realização do estudo aprofundado das diretrizes e recomendações do WCAG e o eMAG, foi realizado um levantamento para priorizar o desenvolvimento dos componentes de funcionalidades de acessibilidade. O levantamento consistiu na divisão estrutural, em formato de tabela, das diretrizes do WCAG a partir dos seus princípios, sendo eles: (Perceptível, Operável, Compreensível e Robusto) e o eMAG a partir das suas seções, das quais são: (Marcação, Comportamento (DOM), Conteúdo/Informação, Apresentação/*Design*, Multimídia e Formulário).

Cada uma das dez tabelas<sup>1</sup> foram distribuídas em páginas a partir de um documento de planilhas do *Google*, onde foi possível observar a descrição da diretriz ou seção, além da definição de dois níveis de valores para cada, sendo estes: A dificuldade de desenvolvimento pelo autor e a dificuldade que o mesmo supõe que os desenvolvedores possam ter em utilizar essas diretrizes ou seções em seus projetos.

Desta forma, foi possível realizar a priorização das diretrizes a partir do resultado desses valores, os quais foram preenchidos de acordo com o conhecimento atual de desenvol-

<sup>1</sup> Disponível em: <https://github.com/daviteixeira-btm/accesshub/tree/main/archives>

Tabela 1 – Tabela da priorização das diretrizes de acessibilidade do princípio Perceptível da WCAG.

<b>Perceptível</b>	<b>Fazer</b>	<b>Usar</b>
(1.1.1) Conteúdo não textual	5	2
(1.2.1) Apenas áudio ou apenas vídeo (pré-gravado)	5	2
(1.2.2) Legendas (pré-gravado)	5	2
(1.2.3) Audiodescrição ou mídia alternativa (pré-gravado)	2	2
(1.3.1) Informações e Relações	5	5
(1.3.2) Sequência com significado	5	4
(1.3.3) Características sensoriais	5	4
(1.4.1) Utilização de cores	5	3
(1.4.2) Controle de áudio	5	5
(1.2.4) Legendas (ao vivo)	2	2
(1.2.5) Audiodescrição (pré-gravado)	5	2
(1.3.4) Orientação	5	4
(1.3.5) Identificar o objetivo de entrada	5	5
(1.4.3) Contraste (mínimo)	4	3
(1.4.4) Redimensionar texto	5	4
(1.4.5) Imagens de texto	5	4
(1.4.10) Refluxo	3	3
(1.4.11) Contraste Não-Textual	4	3
(1.4.12) Espaçamento de texto	3	3
(1.4.13) Conteúdo em foco por mouse ou teclado	4	3
(1.3.6) Identificar o objetivo	4	4
(1.4.6) Contraste (melhorado)	4	3
(1.4.7) Som baixo ou sem som de fundo	4	2
(1.4.8) Apresentação visual	3	2
(1.4.9) Imagens de texto (sem exceção)	5	4
(1.2.6) Língua de sinais (pré-gravado)	1	1
(1.2.7) Audiodescrição estendida (pré-gravado)	2	2
(1.2.8) - Mídia alternativa (pré-gravado)	5	2
(1.2.9) - Apenas áudio (ao vivo)	1	1
<b>Legenda</b>		
Muito Difícil	1	
Difícil	2	
Neutro	3	
Fácil	4	
Muito Fácil	5	

Fonte: Elaborado pelo autor.

vimento do autor deste projeto. Na Tabela 1 é apresentado um exemplo de como uma das dez tabelas foram organizadas:

Ao finalizar a distribuição dos valores em cada tabela, foi realizado o envio de duas cópias, uma para cada orientador da pesquisa, com o objetivo dos mesmos revisarem os valores. A partir disso foi realizado um consenso entre os mesmos, sobre os resultados. Por fim, iniciou-se

o desenvolvimento dos componentes de funcionalidades a partir dos critérios estabelecidos.

#### 4.6 Desenvolvimento dos componentes de funcionalidades de acessibilidade

No contexto deste trabalho, os recursos de acessibilidade são atributos fornecidos pela linguagem de marcação HTML para melhorar a semântica e acessibilidade dos elementos do projeto que está sendo desenvolvido. Por exemplo, o atributo “*alt*” é usado para fornecer a descrição de um texto alternativo caso a imagem não carregue e o atributo “*title*” para dar um texto à imagem quando o *mouse* descansa sobre a mesma em uma página *Web*.

Uma funcionalidade é um conjunto de instruções que são desenvolvidas por meio de uma linguagem de programação, como *JavaScript*, para que se possa fornecer determinados comportamentos na página. Por exemplo, um botão para aumentar o tamanho da fonte do texto com o objetivo de ajudar os usuários que apresentam dificuldades em relação à leitura do conteúdo.

Já um componente é um conjunto de funções de códigos pré-desenvolvidas que podem ser implementados em aplicações com o objetivo de realizar uma determinada tarefa dentro do projeto, desta forma, podemos citar como exemplo um componente que contém um conjunto de funções que são ativadas quando o usuário de um *website* escolhe uma opção no menu para ativar um esquema de cores diferente na página, melhorando assim o contraste e facilitando a compreensão dos elementos para usuários que têm problemas de visão.

Para o desenvolvimento dos componentes de funcionalidades, será feito um estudo detalhado das diretrizes escolhidas para a identificação dos requisitos que cada recomendação propõe em sua documentação. Logo após, será feito o estudo das documentações oficiais das tecnologias que serão utilizadas no trabalho: (HTML, CSS e JavaScript), sendo elas a *stack* mais utilizada no mercado de desenvolvimento de *software* para a *web* (Stack Overflow, 2022).

A proposta é que a implementação dessas funcionalidades seja feita utilizando apenas o código base, destacando o princípio fundamental do reúso de *software*. Dessa forma, as funcionalidades podem ser facilmente adaptadas e incorporadas a diferentes projetos, eliminando a necessidade de depender de bibliotecas de desenvolvimento ou *frameworks* específicos. O enfoque principal do trabalho reside na capacidade de reutilização dessas funcionalidades, proporcionando aos desenvolvedores uma abordagem flexível e versátil para a criação de *interfaces* acessíveis em suas aplicações *web*.



## 4.7 Arquitetura de Navegação

Nesta etapa, para a construção da estrutura de navegação do projeto foi desenvolvida a representação hierárquica do *site*, assim, segundo Ceci e Lanotte (2021) uma especificação explícita do conceito de *design* e organização do conhecimento, apresentando os principais fluxos de uso para os usuários, como também organizando hierarquicamente os conceitos do *site* para definir a ordem e a navegação das páginas em um projeto, como podemos observar na Figura 2. Essa representação é muito útil para o planejamento de estratégias de *User Experience* (UX) no desenvolvimento de *websites*.

## 4.8 Desenvolvimento do AccessHub

A próxima etapa foi o desenvolvimento do *website* com base nos artefatos gerados nas etapas anteriores. Desta forma, a aplicação foi desenvolvida utilizando as tecnologias HTML, CSS e JavaScript em sua base, além de outras tecnologias auxiliares que são descritas com mais detalhes no capítulo 5.

## 4.9 Publicação

A etapa de publicação consistiu no *deploy* na plataforma de hospedagem da Vercel<sup>2</sup>, que de acordo com a sua página oficial, Vercel (2023) é uma plataforma de nuvem que fornece aos desenvolvedores a hospedagem de *sites* e serviços *Web* instantaneamente, permitindo criar, implantar e dimensionar aplicativos sem servidor e *sites* estáticos. Como também a disponibilização do projeto no *Github*<sup>3</sup>, que segundo a sua página oficial, GitHub (2023) permite hospedagem de código para controle de versão e colaboração. Desta forma o *site* vai poder se estabelecer como um projeto *open source*, onde sua manutenção e atualização vai ser mantida pelo autor, como também pelos desenvolvedores interessados em contribuir com o trabalho.

## 4.10 Participantes e Materiais

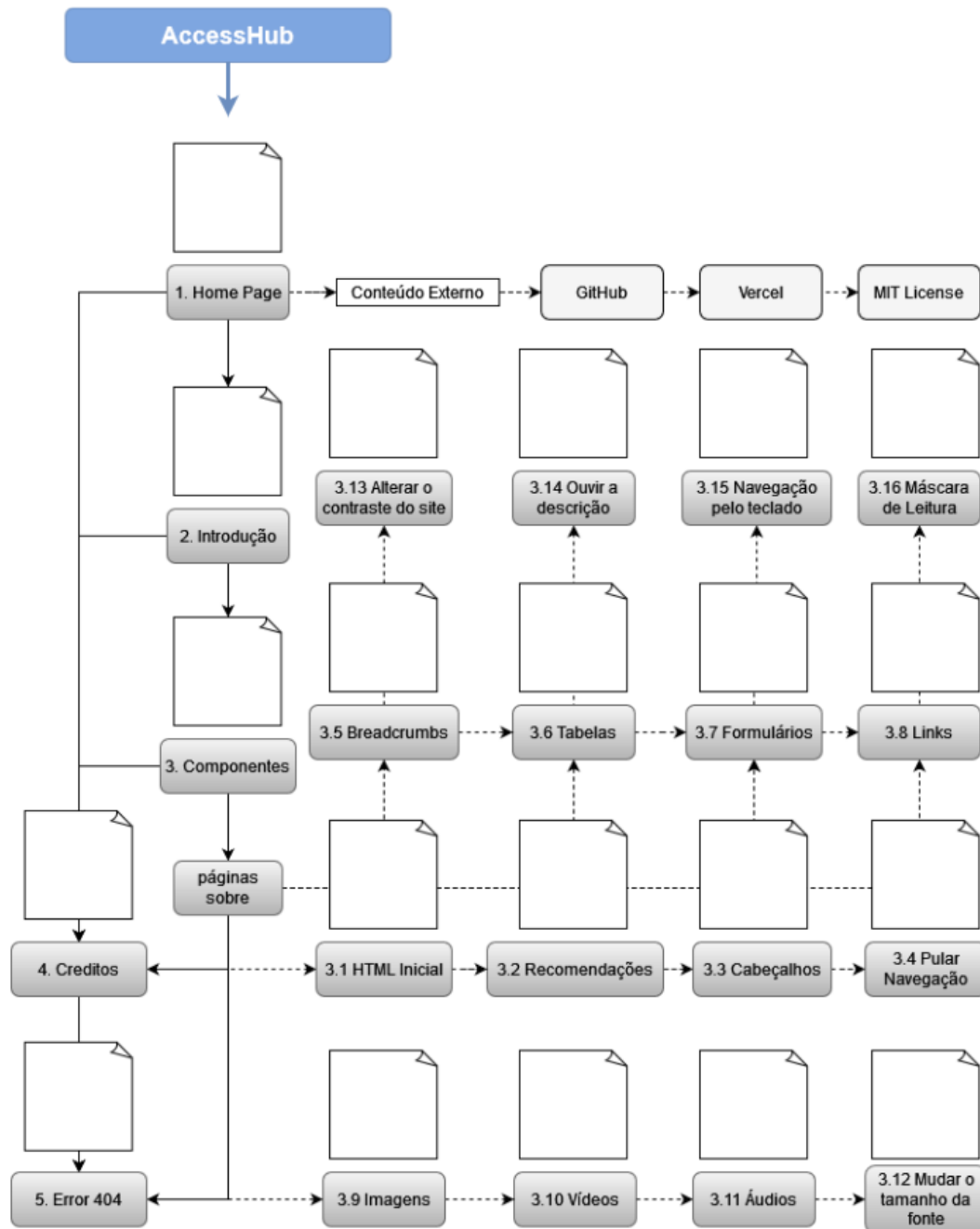
Para a realização da avaliação do AccessHub, foi conduzida uma pesquisa por meio de um desafio técnico remoto, com duração de duas semanas. A participação de oito desenvolvedores distintos (ver Quadro 8), foi selecionada levando em consideração critérios

---

<sup>2</sup> Disponível em: <https://vercel.com/>

<sup>3</sup> Disponível em: <https://github.com/>

Figura 2 – Representação hierárquica da estrutura do site



Fonte: Elaborado pelo autor.

específicos que garantiriam um perfil adequado e representativo para a realização do desafio técnico. Os participantes foram escolhidos com base em sua experiência atuando em equipes de desenvolvimento de *software*, pois esse perfil de desenvolvedor possui conhecimentos práticos e vivência no ambiente de desenvolvimento, o que lhes confere a capacidade de avaliar e aplicar as funcionalidades de acessibilidade propostas pelo AccessHub de forma significativa e realista. Essa seleção cuidadosa permitiu obter *insights* valiosos para a avaliação do AccessHub.

Foi usado os seguintes materiais para este estudo: i) um Termo de Consentimento Livre e Esclarecido (TCLE); ii) um questionário de perfil para coletar informações dos participantes sobre sua função na equipe de desenvolvimento, tempo de experiência nessa função, se está cursando alguma graduação na área de Tecnologia da Informação; se possui experiência em Acessibilidade *Web*; Se já participou de algum treinamento ou curso específico sobre Acessibilidade *Web*; se possui conhecimento nas diretrizes de acessibilidade *web* abordadas neste projeto ou alguma outra; e se a equipe que o participante trabalha adota práticas de Acessibilidade *Web* em seus projetos; iii) o plugin do *Google Lighthouse*<sup>4</sup>; iv) o projeto AccessHub; v) uma *Landing Page* desenvolvida pelo próprio autor que serviu como material para um desafio técnico; vi) um questionário pós-desafio técnico composto pelo Modelo de Aceitação de Tecnologia (TAM) com perguntas adicionais abertas para obter a opinião dos desenvolvedores participantes sobre o que foi fácil ou difícil ao usar a ferramenta e sugestões de melhoria.

O questionário TAM, do tipo *Likert* de sete pontos, foi utilizado para avaliar a aceitação do AccessHub por meio de vários construtos, como Autoeficácia do Computador, Prazer Percebido, Usabilidade Objetiva e outros. Entre eles, focamos nos três construtos centrais: Utilidade Percebida (UP), Facilidade de Uso Percebida (FUP) e Intenção Comportamental (IC). O questionário foi adaptado para se adequar ao contexto da pesquisa, alterando algumas palavras-chave de cada afirmação, conforme apresentado no Quadro 7.

#### 4.11 Avaliação

Com a conclusão da primeira versão do AccessHub, a validação do projeto aconteceu conforme as seguintes etapas:

A primeira etapa consistiu na análise do *site* através do *plugin* de avaliação do *Google Lighthouse*. Este *plugin*, de acordo com a sua página oficial Lighthouse (2023), é uma ferramenta de código aberto desenvolvida pelo *Google* que ajuda a avaliar e melhorar a qualidade das páginas da *web*, permitindo que os desenvolvedores executem auditorias nas páginas da *web* diretamente no navegador.

Desta forma, segundo Chan-Jong-Chu *et al.* (2020) o *plugin* realiza auditorias automatizadas em vários aspectos de uma página da *web*, incluindo desempenho, acessibilidade, práticas recomendadas de desenvolvimento, *Search Engine Optimization* (SEO) e muito mais, fornecendo uma pontuação geral, bem como detalhes específicos sobre os problemas encontrados

<sup>4</sup> Disponível em: <https://developer.chrome.com/docs/lighthouse/>

Quadro 7 – Afirmativas do questionário TAM adaptados ao contexto do projeto.

Item	Declaração
UP1	Usar o AccessHub melhora o meu desempenho para implementar recursos de Acessibilidade Web.
UP2	Usar o AccessHub melhora a minha produtividade para implementar recursos de Acessibilidade Web.
UP3	Usar o AccessHub melhora a minha efetividade para implementar recursos de Acessibilidade Web.
UP4	Eu considero o AccessHub útil para implementar recursos de Acessibilidade Web.
FUP1	O AccessHub foi claro e fácil de entender.
FUP2	Usar o AccessHub não exigiu muito esforço mental.
FUP3	Eu considero o AccessHub fácil de usar.
FUP4	Eu acho fácil implementar recursos de Acessibilidade Web usando o AccessHub.
IC1	Assumindo que eu tenha acesso ao AccessHub, pretendo usá-lo para implementar recursos de Acessibilidade Web.
IC2	Dado que eu tenha acesso ao AccessHub, prevejo que vou utilizá-lo para implementar recursos de Acessibilidade Web.
IC3	Pretendo usar o AccessHub para implementar recursos de Acessibilidade Web nos próximos meses.

Fonte: Elaborado pelo autor.

e sugestões de otimização. Depois de executar a análise, o *Lighthouse* produz um relatório com pontuações referentes a cada tipo de auditoria (por exemplo, desempenho ou acessibilidade) e fornece conselhos acionáveis para resolver os problemas levantados.

A segunda etapa foi a avaliação do AccessHub com desenvolvedores *Web*, como o objetivo de verificar se o projeto está atendendo os objetivos idealizados a partir da percepção de profissionais. Os desenvolvedores participantes primeiro foram solicitados a preencher o TCLE em conjunto com o questionário de perfil, a fim de obter informações sobre suas habilidades e experiência relacionadas ao desenvolvimento *Web* e acessibilidade.

Em seguida, os desenvolvedores foram convidados a participar da realização de um desafio técnico. Para o desafio, foi desenvolvido uma *Landing Page* para a venda de um produto fictício, porém a página apresentava problemas relacionados à acessibilidade *Web*. O processo de desenvolvimento desta *Landing Page* foi dividido em duas etapas distintas.

Na primeira etapa, focamos na criação da página original, buscando atingir a pontuação máxima em todos os critérios do *Google Lighthouse*. Em seguida, foi realizada uma cópia dessa página, na qual foram feitas modificações, como a remoção de *tags* e a alteração da semântica dos elementos, com o objetivo de reduzir a pontuação obtida. Ao ter essas duas versões da página como referência, tornou-se possível realizar uma análise comparativa e identificar as correções necessárias nos projetos enviados pelos desenvolvedores.

Para solucionar esse desafio, foi utilizado o AccessHub que oferece recursos e funcionalidades de acessibilidade prontos para uso. A partir desse *site*, os desenvolvedores enfrentaram o desafio de implementar e integrar uma lista de sete funcionalidades pré-estabelecidas na *Landing Page*, tornando-a mais acessível e a fim de proporcionar uma experiência inclusiva para todos os usuários seguindo as explicações e código fonte fornecidos pelo AccessHub.

Ao concluir a implementação, os desenvolvedores foram instruídos a enviar seus projetos e a pontuação obtida no *Google Lighthouse*, permitindo a comparação dos dados com a pontuação anterior registrada da *Landing Page* antes do desafio técnico.

Além disso, os desenvolvedores foram convidados a preencher um formulário de pesquisa para compartilhar sua percepção e *feedback* sobre o AccessHub, bem como responder ao questionário do TAM, fornecendo *insights* sobre a aceitação e adoção da ferramenta.

## 5 ACCESSHUB

O AccessHub<sup>1</sup> surge como uma resposta à dificuldade enfrentada pelos desenvolvedores na implementação de funcionalidades de acessibilidade em projetos de *websites*. Muitas vezes, essa dificuldade é resultado da falta de conhecimento sobre o assunto ou da falta de tempo para priorizar o planejamento adequado para a acessibilidade *web* em seus projetos.

Com o objetivo de superar esses obstáculos, o projeto AccessHub foi idealizado e desenvolvido. O AccessHub é um *website* guia projetado para auxiliar os desenvolvedores na criação de páginas *web* acessíveis, fornecendo materiais de suporte e exemplos práticos de implementações de funcionalidades de acessibilidade.

A proposta do AccessHub é oferecer suporte e orientação abrangentes para os desenvolvedores, agilizando o processo de aprendizagem e desenvolvimento de *websites* acessíveis. Ao disponibilizar uma estrutura organizada e recursos práticos, o AccessHub visa capacitar os desenvolvedores a criar páginas *web* que atendam aos critérios de acessibilidade estabelecidos, garantindo uma experiência inclusiva para todos os usuários.

Ao longo deste trabalho, serão apresentados os principais aspectos do AccessHub, incluindo a estrutura do site, os componentes disponíveis, exemplos de implementação e as formas como os desenvolvedores podem fazer uso dessas ferramentas. Além disso, serão explorados os resultados da avaliação realizada com os desenvolvedores, analisando tanto a pontuação obtida no *Google Lighthouse* quanto a percepção geral sobre o AccessHub.

Por meio dessa abordagem, espera-se evidenciar a importância do AccessHub como uma solução viável e eficiente para a implementação de acessibilidade em projetos *web*, oferecendo suporte prático e recursos indispensáveis para os desenvolvedores.

### 5.1 Apresentação estrutural do AccessHub

A seguir, descrevo detalhadamente a estrutura do *website*, onde poderemos observar quais tecnologias foram usadas para o desenvolvimento do AccessHub, além de mostrar como elas se relacionam e contribuem para o objetivo geral do projeto. Logo após, será apresentado as páginas que mostram o conteúdo desenvolvido, como também os componentes de funcionalidades de acessibilidade e as suas funções na construção de páginas *web* acessíveis, desenvolvidas para este projeto.

---

<sup>1</sup> Disponível em: <https://accesshub.vercel.app/>

### 5.1.1 *Front-end do projeto*

Na construção do *Front-end* do projeto AccessHub, foi adotada a abordagem de desenvolvimento *Single Page Application* (SPA). Essa abordagem, que se assemelha ao modelo tradicional de desenvolvimento, mantém a presença de um *Front-end*, mas trabalha com componentes que separam a camada de dados da camada de apresentação. Para a implementação do AccessHub, foram utilizadas as seguintes tecnologias:

#### 5.1.1.1 *Node.js*

O primeiro passo consistiu na instalação do Node.js<sup>2</sup>, um ambiente de tempo de execução de código aberto. Conforme mencionado por Li *et al.* (2022), o Node.js permite a execução de código JavaScript no lado do servidor, fornecendo um vasto ecossistema de pacotes e módulos disponíveis no npm (Node Package Manager). Essa escolha facilita o desenvolvimento e a integração de bibliotecas de terceiros no projeto do AccessHub.

#### 5.1.1.2 *Vite*

Em seguida, utilizou-se o Vite<sup>3</sup> como ferramenta de compilação do projeto. Segundo Françoise *et al.* (2021), o Vite é um *boilerplate* (um conjunto de códigos que devem ser incluídos em muitos lugares com pouca ou nenhuma alteração) com configurações pré-estabelecidas para a construção de projetos React, oferecendo uma série de funcionalidades prontas para uso. Além disso, a ferramenta é facilmente configurável, suportada por diversas aplicações e fornece um servidor de desenvolvimento altamente performático, tornando-a ideal para o projeto AccessHub.

#### 5.1.1.3 *React*

O React<sup>4</sup> foi escolhido como biblioteca JavaScript para a criação das *interfaces* de usuário do AccessHub. Conforme destacado por Madsen *et al.* (2020), o React é conhecido por sua performance e é amplamente utilizado para a criação de *User Interfaces* (UI). As aplicações desenvolvidas com React seguem uma abordagem baseada em componentização, onde cada componente possui um conjunto de propriedades, um estado que pode variar ao longo do tempo e um método de renderização que especifica declarativamente os subcomponentes. A escolha do

---

<sup>2</sup> Disponível em: <https://nodejs.org/en>

<sup>3</sup> Disponível em: <https://vitejs.dev/>

<sup>4</sup> Disponível em: <https://react.dev/>

React garante uma *interface* ágil para o AccessHub, sem a necessidade de esforços adicionais para otimização de performance.

#### 5.1.1.4 *React Router*

Para a navegação entre as páginas do AccessHub, foi adotada a biblioteca React Router<sup>5</sup>. De acordo com a página oficial do React Router (Remix, 2023), essa biblioteca fornece recursos completos de roteamento para aplicações React, permitindo a navegação baseada em URLs em aplicações de página única (SPA), onde diferentes componentes são renderizados com base na URL atual.

#### 5.1.1.5 *Chakra-UI*

O Chakra-UI<sup>6</sup> foi escolhido como biblioteca de componentes para a criação dos blocos de construção do *site* do AccessHub. Conforme documentação oficial disponível no Github (CHAKRA UI, 2023), o Chakra-UI é uma biblioteca de componentes React acessíveis, reutilizáveis e combináveis, que agiliza o desenvolvimento de aplicativos e *sites* modernos. Ele oferece uma ampla variedade de componentes que podem ser facilmente combinados para criar *interfaces* de usuário complexas, seguindo as melhores práticas de acessibilidade *web*. A utilização do Chakra-UI permitiu o desenvolvimento de uma *interface* intuitiva, responsiva e alinhada com os padrões de acessibilidade *web* no projeto.

#### 5.1.1.6 *TypeScript*

Para garantir uma maior segurança e prevenir erros no desenvolvimento dos componentes do AccessHub, foi adotado o TypeScript<sup>7</sup>. De acordo com Bogner e Merkel (2022), o JavaScript, devido à sua natureza dinâmica e flexível, pode apresentar desafios em relação à qualidade do código, desta forma o TypeScript, sendo um superconjunto de JavaScript, permite a utilização de tipagem estática, abordando e corrigindo problemas comuns de codificação. Ao compilar o código TypeScript, ele é convertido novamente para JavaScript, fornecendo um desenvolvimento mais robusto e confiável no projeto do AccessHub.

<sup>5</sup> Disponível em: <https://reactrouter.com/en/main>

<sup>6</sup> Disponível em: <https://chakra-ui.com/>

<sup>7</sup> Disponível em: <https://www.typescriptlang.org/>



### 5.1.1.7 Componente de apresentação dos códigos fontes

Para apresentar os códigos fonte das funcionalidades *open source* do AccessHub, optou-se por uma abordagem interna utilizando Markdown. Os exemplos de códigos são escritos em arquivos distintos e carregados em um componente interno por meio do pacote `markdown-to-jsx`<sup>8</sup>, que converte o texto em formato Markdown para JSX, facilitando a apresentação dos códigos já formatados.

Essa abordagem elimina a necessidade de um *Back-end* para armazenar o código, evitando o uso de serviços externos de armazenamento *web*. Além disso, permite que os usuários tenham acesso imediato aos exemplos de código, sem a necessidade de esperar por requisições de dados, melhorando significativamente a performance do *website*.

Alguns pacotes foram utilizados para a apresentação dos exemplos de código:

1. **react-syntax-highlighter**<sup>9</sup>: fornece recursos de destaque de sintaxe para texto, especialmente para códigos-fonte, destacando palavras-chave, identificadores, comentários, entre outros elementos, facilitando a leitura e reconhecimento dos aspectos do código.
2. **react-copy-to-clipboard**<sup>10</sup>: oferece um componente simples para copiar o conteúdo para a área de transferência. Esse pacote não renderiza nenhuma *tag* e requer a presença de um único elemento filho, que é usado para capturar cliques e acionar a cópia do conteúdo.

Essa combinação de pacotes permite a apresentação dos exemplos de código de forma eficiente e amigável no AccessHub.

## 5.1.2 Interfaces do AccessHub e as suas funções

Para a criação do AccessHub, foram desenvolvidas diversas páginas com o objetivo de apresentar o projeto e os componentes de funcionalidades desenvolvidos de forma acessível. A seguir, serão apresentados exemplos concretos das páginas que compõe o projeto e suas respectivas funções:

### 5.1.2.1 Tela de Home Page

A *Home Page* do AccessHub é a primeira tela que os usuários encontram ao acessar o *site*. Nessa página, é apresentado o *slogan*, como também os objetivos do projeto, além de *links*

<sup>8</sup> Disponível em: <https://www.npmjs.com/package/markdown-to-jsx>

<sup>9</sup> Disponível em: <https://www.npmjs.com/package/react-syntax-highlighter>

<sup>10</sup> Disponível em: <https://www.npmjs.com/package/react-copy-to-clipboard>

de navegação para direcionar os usuários para o repositório do projeto, disponível no GitHub, como também para outras páginas do *site*.

Figura 3 – Tela de Home Page



Fonte: Elaborado pelo autor.

#### 5.1.2.2 Tela de Introdução

A Tela de Introdução tem como objetivo fornecer aos usuários uma visão geral do projeto AccessHub e explicar como eles podem contribuir com ele. Nessa página, é utilizado textos descritivos para apresentar os objetivos do projeto além do *link* para o mecanismo de contribuição do GitHub, para os interessados em participar do projeto.

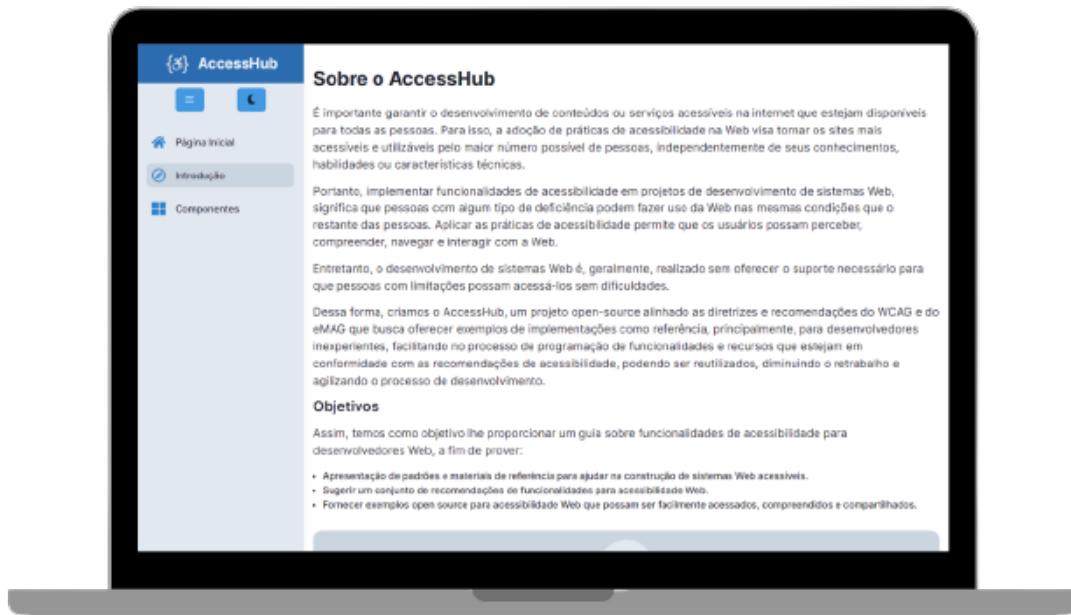
#### 5.1.2.3 Tela de Componentes

Na Tela de Componentes, listamos os 16 componentes de funcionalidades disponíveis no AccessHub. Cada componente, apresentado em formato de *card*, possui um *link* para sua própria página, que contém explicações textuais e exemplos de código fonte.

#### 5.1.2.4 Página de Créditos das Imagens

A Página de Créditos das Imagens é dedicada a fornecer informações sobre as imagens utilizadas no *site* do AccessHub. Nessa página, utilizamos listas e *links* para listar as

Figura 4 – Tela de Introdução



Fonte: Elaborado pelo autor.

Figura 5 – Tela de Componentes



Fonte: Elaborado pelo autor.

imagens utilizadas, juntamente com os respectivos créditos e direitos autorais. Essa abordagem garante a devida atribuição aos criadores das imagens e demonstra respeito às normas de direitos autorais.

### 5.1.2.5 Página de Error 404

A Página de Error 404 é exibida quando um usuário tenta acessar uma página que não existe no *site*. Essa página utiliza um componente de erro personalizado, que exibe uma mensagem amigável informando que a página solicitada não foi encontrada. Além disso, incluímos um botão para redirecionar o usuário de volta à página inicial do projeto.

### 5.1.2.6 Responsividade e Dark Mode

O projeto foi desenvolvido para que possa funcionar em diferentes dispositivos, sejam eles computadores, *laptops*, *tablets* ou *smartphones*, expandindo assim para que os usuários possam acessar o conteúdo em diferentes formatos de exibição. O projeto também inclui o formato de exibição *Dark Mode* (modo escuro), sendo este utilizado em *sites* como uma opção que permite aos usuários visualizar o conteúdo da tela em tons de preto e branco, geralmente ativado em ambientes com pouca luz, possibilitando aos usuários uma melhor leitura do conteúdo da página.

Figura 6 – Apresentação dos diferentes tipos de resolução e Dark Mode do projeto



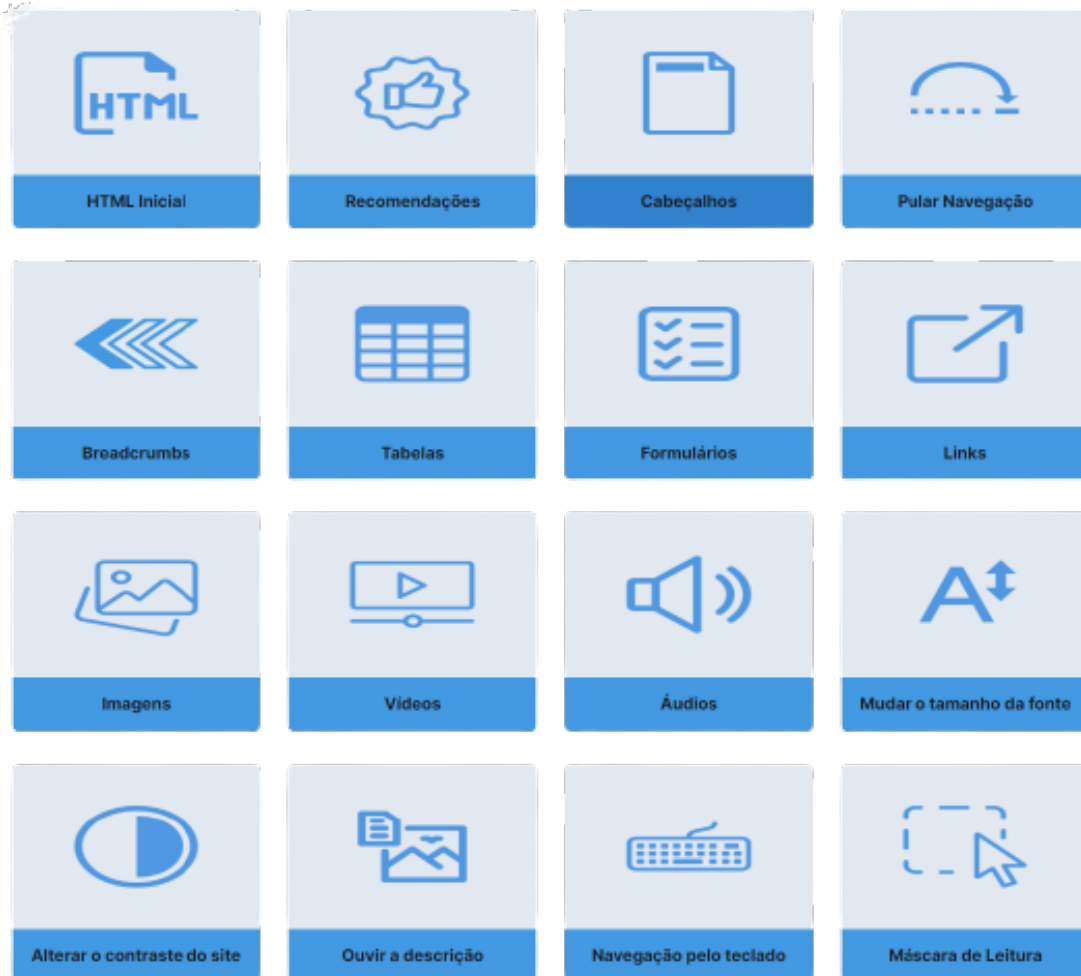
Fonte: Elaborado pelo autor.

Cada página desempenha um papel importante na experiência do usuário, garantindo que as informações sejam apresentadas de forma clara, organizada e acessível, permitindo que os usuários interajam e naveguem pelo *site* com facilidade.

### 5.1.3 Utilização dos Componentes de Funcionalidades

Os componentes de funcionalidades desenvolvidos no AccessHub desempenham um papel fundamental na criação de um ambiente digital acessível. Nesta seção, serão apresentados os 16 componentes de funcionalidades disponíveis no projeto, fornecendo uma visão geral de cada um deles, bem como exemplos práticos de sua utilização.

Figura 7 – Visão geral dos 16 componentes de funcionalidade do AccessHub



Fonte: Elaborado pelo autor.

#### 5.1.3.1 Componente de HTML Inicial

Componente de HTML Inicial: Este componente permite aprender sobre a estrutura inicial do HTML5, que serve como base para o desenvolvimento de *sites* acessíveis. O componente aborda os seguintes pontos:

- A importância de seguir uma ordem compreensível nos elementos HTML, utilizando-os de acordo com sua finalidade específica;

- Identificação do idioma principal da página;
- Informar sobre mudanças de idioma no conteúdo textual da página;
- Utilização de títulos descritivos e informativos para cada página;
- Divisão das áreas de informação do conteúdo de uma página *web*;
- Utilização do HTML5 com o ARIA;
- Evitar a criação de páginas com atualizações automáticas periódicas;
- Evitar redirecionamento automático de páginas;
- Fornecer alternativas para modificar o limite de tempo em uma página *web*.

Oferecendo exemplos de implementação para que os desenvolvedores possam iniciar um projeto de página web da maneira correta.

#### 5.1.3.2 *Componente de Recomendações*

Esse componente é responsável por apresentar recomendações para o desenvolvimento de *websites* acessíveis, abordando os seguintes pontos:

- Respeitar os padrões *web* e boas práticas de desenvolvimento *web*;
- A importância dos cabeçalhos e rótulos para a localização de informações pelos usuários;
- Ordenação lógica e intuitiva das informações para leitura e tabulação;
- Não utilizar apenas cores para diferenciar elementos em uma página *web*;
- Evitar intermitência de tela;
- Assegurar o controle do usuário sobre alterações temporais do conteúdo;
- Disponibilizar documentos em formatos acessíveis;
- Garantir a leitura e compreensão das informações em uma página *web*;
- Exibir explicações para siglas, abreviaturas e palavras incomuns;
- Fornecer controle de animações em um *website*;
- Garantir a acessibilidade dos objetos programáveis.

Oferecendo exemplos de código para que os desenvolvedores possam utilizar as recomendações disponíveis.

#### 5.1.3.3 *Componente de Cabeçalhos*

Com este componente, os desenvolvedores têm acesso a exemplos de código fonte para aprender a utilizar corretamente os níveis de cabeçalho no desenvolvimento de suas páginas *web*. Também é abordada a criação de títulos de seções significativos e bem estruturados, pois os

cabeçalhos desempenham um papel fundamental na organização e acessibilidade do conteúdo de um *site*.

#### 5.1.3.4 *Componente de Pular Navegação*

Utilizando este componente, é possível aprender, por meio de um exemplo de código, como criar âncoras para acessar diretamente blocos de conteúdo definidos pelos desenvolvedores. Isso permite que usuários com deficiências ou limitações acessem diretamente o conteúdo principal de uma página *web* sem a necessidade de percorrer todos os elementos de navegação e cabeçalhos.

#### 5.1.3.5 *Componente de Breadcrumbs*

Este componente ensina, por meio de um exemplo de código, como informar aos usuários a sua localização atual nas páginas *web* de forma acessível. Ele fornece um mecanismo que permite a orientação dentro de um conjunto de páginas por meio de *links* navegáveis dispostos em uma lista, permitindo entender o caminho percorrido até a página atual.

#### 5.1.3.6 *Componente de Tabelas*

Este componente, por meio de exemplos de código, ensina como tornar as tabelas de dados simples acessíveis por meio das *tags* semânticas das tabelas. Isso permite a utilização de títulos e resumos adequados nas tabelas, além do agrupamento das linhas de cabeçalho, corpo e rodapé.

#### 5.1.3.7 *Componente de Formulários*

Através deste componente, é possível aprender técnicas para o desenvolvimento e exibição de formulários de forma acessível. Dentre as quais:

- Agrupamento de campos de formulários;
- Identificação de erros na entrada de dados e confirmação do envio das informações;
- Fornecimento de instruções para a entrada de dados;
- Não provocar alterações no contexto da página de forma automática;
- Estabelecer uma ordem lógica de navegação nos formulários;
- Identificar os objetos de entrada através de rótulos descritivos para cada campo;

- Associar etiquetas aos campos de dados;
- Oferecer alternativa em texto para botões de imagem em formulários;
- Fornecer estratégias de segurança específicas em vez de *CAPTCHA* em formulários.

Assim, por meio de exemplos de código fonte, os desenvolvedores *web* podem entender e implementar as estratégias propostas.

#### 5.1.3.8 *Componente de Links*

Com este componente, é possível aprender a utilizar as âncoras em HTML de forma acessível, abordando os seguintes pontos:

- Separação de *links* adjacentes por meio de listas, evitando confusão para usuários que utilizam leitores de tela;
- Definir abertura de novas abas apenas com solicitação do usuário;
- Apresentar uma descrição clara e sucinta dos *links*;
- Evidenciar visualmente os *links* em foco.

Assim, através do conteúdo e exemplos de código, é possível mostrar aos desenvolvedores a importância dessas técnicas ao desenvolver elementos de âncora em seus *sites*.

#### 5.1.3.9 *Componente de Imagens*

A partir deste componente, é possível aprender como desenvolver e apresentar imagens em um *website* de forma mais acessível, abordando as seguintes práticas:

- Alternativas em texto para as imagens do *site* por meio do atributo "alt";
- Fornecimento de descrições complexas por meio de *links*;
- Introdução de elementos que permitam agrupar uma imagem com sua legenda;
- Utilização acessível de mapas de imagem.

Desta forma, os desenvolvedores podem trabalhar melhor com as imagens utilizadas em seus projetos, utilizando os exemplos de código fornecidos.

#### 5.1.3.10 *Componente de Vídeos*

É essencial que os vídeos disponibilizados em um *site* possuam controles, especialmente para reproduzir, pausar e parar. Além disso, é importante fornecer alternativas sonoras ou textuais para usuários com deficiências auditivas ou visuais. Caso seja necessário desenvolver as



funcionalidades de forma isolada, são fornecidos exemplos de código para:

- Reprodução/Pausa de um vídeo;
- Modo tela cheia;
- Aumentar, diminuir e mutar o volume;
- Escolher entre diferentes opções de legenda;
- Fazer o *download* do vídeo;
- Escolher a velocidade de reprodução do vídeo;
- Fornecer audiodescrição ou transcrição descritiva em texto.

Desta forma, é possível aprender a desenvolver as funcionalidades de vídeo e torná-las acessíveis para os usuários.

#### 5.1.3.11 *Componente de Áudios*

Com este componente, o desenvolvedor pode aprender como fornecer alternativas para áudios em projetos de *websites*. Isso inclui transcrições do conteúdo apresentado, beneficiando usuários com deficiência auditiva ou aqueles que preferem apenas ler o material. Além disso, são fornecidos mecanismos que permitem aos usuários controlar o áudio reproduzido.

#### 5.1.3.12 *Componente para Mudar o Tamanho da Fonte*

Com esse componente de funcionalidade, os desenvolvedores podem oferecer aos usuários a possibilidade de alterar o tamanho das fontes do *site*. Isso permite o redimensionamento sem perda de funcionalidade, adaptando o texto à resolução de tela do dispositivo utilizado para acessá-lo.

#### 5.1.3.13 *Componente para Alterar o Contraste do Site*

Este componente de funcionalidade permite que os desenvolvedores aprendam como oferecer um contraste mínimo adequado entre plano de fundo e primeiro plano de um *website*. Isso garante a visualização por pessoas com deficiências visuais, como baixa visão, cromodeficiências ou que utilizam monitores de vídeo monocromático.

#### 5.1.3.14 *Componente para Ouvir a Descrição de uma Imagem*

A partir desse componente de funcionalidade, é possível que o desenvolvedor aprenda como fornecer um recurso de acessibilidade que disponibilize a inclusão de descrições em áudio para imagens e elementos gráficos.

#### 5.1.3.15 *Componente para a Navegação pelo Teclado*

Existem situações em que os desenvolvedores precisam criar algo único que talvez não tenha um equivalente nativo no navegador. Portanto, é necessário criar um controle personalizado e adicionar suporte ao teclado. Este componente ensina, por meio de exemplos com código, como tornar as funcionalidades desenvolvidas compatíveis com a navegação via teclado.

#### 5.1.3.16 *Componente de Máscara de Leitura*

O último componente proporciona aos desenvolvedores a aprendizagem do desenvolvimento da funcionalidade de máscara de leitura. Essa funcionalidade é útil para melhorar a experiência de leitura de usuários com dificuldades visuais. A máscara de leitura consiste em um destaque móvel que acompanha o cursor do mouse, auxiliando na focalização do texto durante a leitura.

### 5.1.4 *Exemplo Prático de Utilização*

A fim de ilustrar a utilização dos componentes de funcionalidades, apresentaremos um exemplo prático envolvendo o componente de Alteração do Tamanho da Fonte. Neste exemplo, demonstraremos como incorporar este componente a um projeto de construção de uma página *web* acessível.

É fundamental permitir o redimensionamento dos textos em um *website* sem perda de funcionalidade, garantindo que a página permaneça legível e funcional mesmo quando redimensionada em até 200%. Isso evita a sobreposição de elementos e a aparição de barras de rolagem horizontais ao redimensionar a página.

Abaixo, é apresentado um exemplo de código que permite aumentar, diminuir e redefinir o tamanho da fonte em um *site*.

Nesse código, é possível observar na parte do JavaScript que podemos adicionar limites para o aumento e a diminuição do tamanho da fonte, inserindo condições dentro dos

Figura 8 – Imagem do código fonte do componente para alterar o tamanho da fonte

```

<!-- Código HTML -->
<button title="Aumentar fonte" onclick="increaseFont()">(A+)</button>
<button title="Tamanho de fonte padrão" onclick="resetFont()">(A)</button>
<button title="Diminuir fonte" onclick="decreaseFont()">(A-)</button>

<p class="font-size">Esse é um texto de exemplo para você observar a funcionalidade.</p>

<!-- Código JavaScript -->
<script>
  //Variáveis para armazenar os valores originais da fonte

  let fontSizeElements = document.querySelectorAll(".font-size");
  let originalFontSizes = {};

  for(let i = 0; i < fontSizeElements.length; i++){
    // Armazena o tamanho original da fonte
    originalFontSizes[i] = getComputedStyle(fontSizeElements[i]).fontSize;
  }

  // Função para aumentar o tamanho da fonte
  function increaseFont(){
    // Seleciona todos os elementos com a classe "font-size"
    let elements = document.querySelectorAll(".font-size");

    for(let i = 0; i < elements.length; i++){
      let fontSize = parseInt(getComputedStyle(elements[i]).fontSize);

      if(fontSize < 30){ // Limite máximo de aumento
        fontSize += 2;
        elements[i].style.fontSize = fontSize + "px";
      }
    }
  }

  // Função para diminuir o tamanho da fonte
  function decreaseFont(){
    // Seleciona todos os elementos com a classe "font-size"
    let elements = document.querySelectorAll(".font-size");

    for(let i = 0; i < elements.length; i++){
      let fontSize = parseInt(getComputedStyle(elements[i]).fontSize);

      if(fontSize > 12){ // Limite mínimo de diminuição
        fontSize -= 2;
        elements[i].style.fontSize = fontSize + "px";
      }
    }
  }

  // Função para voltar ao tamanho original da fonte
  function resetFont(){
    for(let i = 0; i < fontSizeElements.length; i++){
      fontSizeElements[i].style.fontSize = originalFontSizes[i];
    }
  }
</script>

```

Fonte: Elaborado pelo autor.

*event listeners* dos botões. Os limites são definidos em *'if (fontSize < 30)'* e *'if (fontSize > 12)'*, em que 30px é o limite máximo para o aumento e 12px é o limite mínimo para a diminuição.

Esses limites podem ser ajustados de acordo com as necessidades do seu *site*.

É possível armazenar o tamanho atual da fonte de cada elemento individualmente, em vez de utilizar uma variável global. Dessa forma, cada elemento terá seu próprio tamanho de fonte e não haverá interferência entre eles.

Dentro do *loop* *for*, utilizamos a função `getComputedStyle(elements[i]).fontSize` para obter o tamanho atual da fonte do elemento e armazenamos esse valor na variável `fontSize`. Em seguida, verificamos se esse tamanho é menor que 30 *pixels* e, se for, adicionamos 2 *pixels* a ele. Assim, cada elemento com a classe `font-size` terá seu próprio tamanho de fonte, aumentado ou diminuído em 2 *pixels* a cada clique nos botões.

Para garantir que os valores originais sejam restaurados corretamente, armazenamos os valores originais da fonte de cada elemento individualmente quando a página é carregada e, em seguida, podemos utilizá-los para restaurar a fonte quando a função `resetFont` é chamada.

Ao carregar a página, armazenamos os valores originais da fonte de cada elemento com a classe `font-size` em uma variável chamada `originalFontSizes`. Em seguida, na função `resetFont`, percorremos cada elemento e definimos seu tamanho de fonte como o valor armazenado na variável `originalFontSizes`. Dessa forma, cada elemento terá seu tamanho de fonte original restaurado, independentemente de ter um tamanho diferente antes de ser modificado.

Ao seguir essa abordagem, os desenvolvedores podem integrar efetivamente os componentes de funcionalidades do AccessHub em seus projetos, aumentando a acessibilidade das páginas da *web* e oferecendo uma experiência mais inclusiva aos usuários.

### **5.1.5 Como contribuir com o projeto**

Além de utilizar os componentes de funcionalidade fornecidos pelo AccessHub, os desenvolvedores têm a oportunidade de contribuir diretamente com o projeto, auxiliando no aprimoramento contínuo da acessibilidade na *web*. Uma maneira de fazer isso é por meio do mecanismo de contribuição do GitHub.

O AccessHub mantém um repositório *open source* no GitHub onde os desenvolvedores podem colaborar, compartilhando ideias, soluções e contribuindo com código. Ao se envolver nesse processo, é possível ajudar a identificar e corrigir problemas de acessibilidade, propor novos recursos e aprimorar a usabilidade dos componentes existentes.

Para contribuir com o projeto do AccessHub no GitHub, os desenvolvedores podem seguir os seguintes passos:

1. Acesse o repositório do AccessHub no GitHub<sup>11</sup>.
2. Explore os problemas (*issues*) existentes para identificar aqueles em que você pode contribuir ou resolver.
3. Caso encontre um problema que deseje abordar, verifique se ele já não está sendo trabalhado por outro colaborador.
4. Se o problema ainda estiver aberto, você pode comentar na respectiva *issue* para expressar seu interesse em trabalhar nele.
5. Caso não exista um problema relacionado à sua contribuição, você pode abrir uma nova *issue* descrevendo detalhadamente o que gostaria de abordar.
6. Caso deseje contribuir com código, você pode fazer um *fork* do repositório do AccessHub, criar um *branch* para a sua modificação e, em seguida, submeter um *pull request* com suas alterações propostas.
7. Certifique-se de seguir as diretrizes de contribuição do projeto, respeitando a estrutura do código, as boas práticas e a documentação.

Ao contribuir com o AccessHub, os desenvolvedores não apenas auxiliam na melhoria do projeto, mas também passam a se envolver e participar de uma comunidade dedicada a tornar a *web* mais inclusiva e acessível para todos.

---

<sup>11</sup> Disponível em: <https://github.com/daviteixeira-btm/accesshub>

## 6 AVALIAÇÃO DO ACCESSHUB

Nesta seção, apresentamos os resultados da avaliação do AccessHub, que inclui a avaliação individual de suas páginas por meio dos relatórios do *Google Lighthouse*, bem como a avaliação do AccessHub e seus componentes de funcionalidade, realizada por meio de um desafio técnico envolvendo desenvolvedores *web*.

### 6.1 Auditoria individual do AccessHub

Conforme mencionado na seção 4.11 do capítulo 4, a primeira etapa da avaliação do projeto foi a realização de auditorias individuais em cada uma das páginas que compõem o AccessHub. Essas auditorias permitiram avaliar o desempenho, a acessibilidade, às boas práticas de desenvolvimento e o SEO de cada página. O *Google Lighthouse*, seguindo as diretrizes e recomendações do WCAG, foi utilizado para gerar relatórios detalhados, contendo métricas e recomendações de melhoria no código, identificando problemas e erros ao longo do desenvolvimento do projeto. Dessa forma, buscou-se garantir a conformidade com os padrões, identificar possíveis gargalos de desempenho e melhorar a acessibilidade e o SEO do AccessHub. A tabela a seguir (Tabela 2) apresenta a pontuação final de cada página do AccessHub:

A avaliação individual do projeto teve como objetivo principal proporcionar uma melhor experiência aos usuários do AccessHub, assegurando um desempenho otimizado e a conformidade com as diretrizes de acessibilidade. Analisando a Tabela 2, destacamos os seguintes pontos:

1. **SEO:** Todas as páginas receberam a pontuação máxima em SEO, o que contribui para o aumento da visibilidade do projeto entre a comunidade de desenvolvedores e seu posicionamento nas principais pesquisas do *Google*.
2. **Boas práticas de desenvolvimento:** A maioria das páginas obteve a pontuação máxima em boas práticas de desenvolvimento *web*. No entanto, a página de apresentação dos componentes obteve uma pontuação ligeiramente menor (92) devido a um problema na renderização das imagens que representam os componentes, apresentando proporções de tamanho incorretas em suas resoluções. Embora esse problema não afete significativamente a primeira versão do projeto, será estudada uma solução para futuras atualizações.
3. **Acessibilidade:** No geral, todas as páginas alcançaram resultados excelentes em acessibilidade, seguindo as recomendações de acessibilidade *web*. A única exceção são as páginas

Tabela 2 – Pontuações obtidas por cada página desenvolvida do AccessHub.

Telas	Performance	Acessibilidade	Boas Práticas	SEO
Home Page	95	100	100	100
Introdução	95	100	100	100
Componentes	96	100	92	100
Créditos	97	100	100	100
Error 404	96	100	100	100
HTML Inicial	67	90	100	100
Recomendações	93	90	100	100
Cabeçalhos	92	90	100	100
Pular Navegação	78	90	100	100
Breadcrumbs	77	90	100	100
Tabelas	61	90	100	100
Formulários	82	90	100	100
Links	87	90	100	100
Imagens	68	90	100	100
Vídeos	77	90	100	100
Áudios	73	90	100	100
Mudar o Tamanho da Fonte	73	90	100	100
Alterar o Contraste do Site	91	90	100	100
Ouvir a Descrição	79	90	100	100
Navegação pelo Teclado	75	90	100	100
Máscara de Leitura	69	90	100	100
<b>Legenda</b>	<b>(0-49) Ruim</b>	<b>(50-89) Precisa Melhorar</b>	<b>(90-100) Bom</b>	

Fonte: Elaborado pelo autor.

de apresentação dos componentes, que não receberam pontuação máxima devido à forma como os exemplos de código exibem as linhas de comentários, com um contraste insuficiente em relação ao plano de fundo. Esse aspecto será considerado para aprimoramentos futuros.

4. **Performance:** A maioria das páginas de apresentação dos componentes de funcionalidades de acessibilidade obteve pontuações medianas em performance. Isso se deve ao tempo de carregamento das informações, evidenciando um tamanho excessivo na apresentação do conteúdo pela DOM. Será realizado um estudo para melhorar a forma de apresentação do conteúdo nas futuras atualizações do projeto.

Em resumo, a avaliação individual do AccessHub revelou que o projeto está bem estruturado e que as informações apresentadas proporcionam uma boa experiência aos usuários.

## 6.2 Perfil técnico dos participantes

Antes da realização do desafio técnico, cada desenvolvedor foi convidado a responder um formulário contendo TCLE e algumas perguntas para conhecer melhor o perfil técnico de cada um. Participaram do desafio um total de oito desenvolvedores, cujos perfis foram analisados a fim de obter informações relevantes, das quais são apresentadas no Quadro 8.

- Todos os participantes têm uma experiência profissional entre um e três anos.
- Em relação aos cargos atuais, os desenvolvedores P1, P2, P3, P4, P7 e P8 são desenvolvedores *FullStack*, o participante P5 é desenvolvedor *web Back-end* e o participante P6 é desenvolvedor *web Front-end*.
- Quanto ao nível acadêmico, o participante P1 não possui graduação em áreas relacionadas à Tecnologia da Informação; o participante P2 está cursando Sistemas de Informação no 8º semestre; o participante P3 está cursando Engenharia de Software no 7º semestre; o participante P4 está cursando Sistemas de Informação no 8º semestre; o participante P5 está cursando Análise e Desenvolvimento de Sistemas no 5º semestre; o participante P6 é formado em Sistemas de Informação; o participante P7 está cursando Ciências da Computação no 9º semestre; e o participante P8 está cursando Engenharia de Software no 7º semestre.
- Quanto ao conhecimento específico em Acessibilidade *Web*, nenhum dos participantes havia realizado treinamentos ou cursos sobre o assunto antes do desafio.

Além disso, foram feitas perguntas sobre o nível de experiência em Acessibilidade *Web*, a familiaridade com diretrizes de acessibilidade e se suas equipes atuais adotam práticas de Acessibilidade *Web* em seus projetos.

Com essas informações, foi possível obter uma visão geral do perfil técnico dos desenvolvedores participantes e identificar os conhecimentos prévios sobre acessibilidade. As respostas fornecidas serão levadas em consideração na análise dos resultados do desafio técnico, a fim de avaliar o impacto da participação no AccessHub na percepção e compreensão dos conceitos de acessibilidade *web* pelos desenvolvedores.

## 6.3 Resultados do desafio técnico com os desenvolvedores

O desafio técnico proposto consistiu na resolução de problemas relacionados à acessibilidade *web*, semântica, boas práticas de desenvolvimento e na implementação de componentes



Quadro 8 – Perfil dos participantes.

Participantes	Experiência em Acessibilidade Web	Possui familiaridade com alguma diretriz de Acessibilidade Web	Equipe no qual trabalha adota práticas de Acessibilidade Web em seus projetos
P1	Estou aprendendo sobre o tema, mas nunca implementei recursos de Acessibilidade Web	Não	Parcialmente
P2	Não tenho conhecimento sobre Acessibilidade Web	Não	Não
P3	Tenho algum conhecimento e implementei recursos de Acessibilidade Web em pelo menos um projeto	Não	Parcialmente
P4	Implementei recursos de Acessibilidade Web em vários projetos, mas não de forma constante	WCAG	Parcialmente
P5	Não tenho conhecimento sobre Acessibilidade Web	Não	Parcialmente
P6	Implemento recursos de Acessibilidade Web constantemente como parte do meu trabalho	WCAG e eMAG	Sim
P7	Implementei recursos de Acessibilidade Web em vários projetos, mas não de forma constante	WCAG	Não
P8	Estou aprendendo sobre o tema, mas nunca implementei recursos de Acessibilidade Web	Não	Não

Fonte: Elaborado pelo autor.

de funcionalidades de acessibilidade em uma *Landing Page* para a venda de um produto fictício. Os desenvolvedores foram instruídos a implementar sete componentes de funcionalidade no projeto, visando aprimorar a experiência do usuário e tornar o conteúdo acessível a todos. Os componentes incluíam funcionalidades para pular navegação para o conteúdo principal, mudar o tamanho da fonte do texto, ajustar o contraste da página, ouvir a descrição da primeira imagem, acionar a máscara de leitura, possibilitar navegação pelo teclado em uma tabela e estruturar corretamente o vídeo da página.

Para concluir esse desafio, os desenvolvedores foram orientados a utilizar o *AccessHub*, que oferece recursos e funcionalidades de acessibilidade prontos para uso, e o *plugin do Google Lighthouse*, com o objetivo de melhorar a pontuação de Acessibilidade e Boas Práticas de desenvolvimento. Assim, o desafio buscou destacar a importância desses aspectos na criação de aplicações *web* e demonstrar como é possível implementar soluções eficazes para garantir a acessibilidade do conteúdo a todos os usuários.

Após a conclusão do desafio, os desenvolvedores enviaram seus projetos e a pontuação obtida no *Google Lighthouse* em pastas individuais no *Google Drive*, permitindo a comparação dos dados com a pontuação anterior registrada da página do produto antes do desafio

Tabela 3 – Pontuações obtidas por cada desenvolvedor no desafio técnico.

<b>Landing Page</b>	<b>Performance</b>	<b>Acessibilidade</b>	<b>Boas Práticas</b>	<b>SEO</b>
Pontuação anterior	93	59	83	70
<b>Participantes</b>	<b>Performance</b>	<b>Acessibilidade</b>	<b>Boas Práticas</b>	<b>SEO</b>
P1	89	98	100	100
P2	83	100	83	80
P3	92	92	92	100
P4	97	93	92	80
P5	92	87	83	80
P6	97	100	92	100
P7	94	100	100	100
P8	69	91	83	80
<b>Legenda</b>	<b>(0-49) Ruim</b>	<b>(50-89) Precisa Melhorar</b>	<b>(90-100) Bom</b>	

Fonte: Elaborado pelo autor.

técnico. Dessa forma, foi possível identificar o percentual de melhoria alcançado por cada desenvolvedor. Os resultados estão apresentados na Tabela 3.

Analisando os dados obtidos das pontuações que cada desenvolvedor obteve no desafio técnico, foi possível calcular o aumento ou a redução percentual em cada uma das categorias apresentadas na avaliação do *Google Lighthouse* com o seguinte cálculo:

$$X = \text{Novo Valor da Categoria} - \text{Valor Original da Categoria}$$

$$Y = X / \text{Valor Original da Categoria}$$

$$Y * 100 = \text{Valor Percentual}$$

Desta forma, podemos determinar se os participantes obtiveram resultados positivos ou negativos na realização do desafio técnico. A Tabela 4 apresenta o resultado final aproximado que cada participante obteve de acordo com os cálculos apresentados.

Com base nos resultados da tabela, podemos fazer uma análise geral do desempenho de cada participante. Observamos que todos os participantes conseguiram melhorar as pontuações de Acessibilidade e SEO de seus projetos. Além disso, cinco dos participantes conseguiram melhorar a pontuação em Boas Práticas de Desenvolvimento.

No entanto, é importante ressaltar que cinco participantes tiveram uma queda na pontuação de performance do projeto. Essa queda pode ser atribuída a fatores como a forma como as funcionalidades foram implementadas em cada projeto. Vale mencionar que, na pontuação anterior da *Landing Page* para o desafio técnico, não foram considerados os componentes de

Tabela 4 – Percentual de aumento ou redução da pontuação obtida dos participantes

Participantes	Performance	Acessibilidade	Boas Práticas	SEO
P1	- 4,3%	66,10%	20,48%	42,85%
P2	- 10,75%	69,49%	0%	14,28%
P3	- 1,07%	55,93%	10,84%	42,85%
P4	4,3%	57,62%	10,84%	14,28%
P5	- 1,07%	47,45%	0%	14,28%
P6	4,3%	69,49%	10,84%	42,85%
P7	1,07%	69,49%	20,48%	42,85%
P8	- 25,80%	54,23%	0%	14,28%

Fonte: Elaborado pelo autor.

funcionalidade de acessibilidade que foram solicitados. Portanto, é possível que essas mudanças na performance tenham sido influenciadas pela inclusão desses componentes ou pela forma que o participante realizou a codificação do desafio.

Em seguida, foi realizada a verificação dos projetos enviados por cada participante no *Google Drive*<sup>1</sup> para avaliar se eles conseguiram completar o principal objetivo do desafio, que era a implementação dos sete componentes de funcionalidades de acessibilidade. Os resultados destacaram o seguinte:

- O participante P1 conseguiu implementar seis dos sete componentes de funcionalidades de acessibilidade, mas não conseguiu implementar a Máscara de Leitura. Além disso, das seis funcionalidades implementadas, apenas quatro delas funcionaram conforme o esperado. A funcionalidade de pular navegação não utilizou o código fornecido, optando apenas pela *tag* de âncora para saltar ao conteúdo principal. Além disso, a funcionalidade para mudar o tamanho das fontes da página não está funcionando adequadamente, pois está aplicando funcionalidades de *zoom* na página. É interessante observar que o participante P1 refatorou os componentes de funcionalidades de acordo com sua abordagem, modificando os códigos quase que totalmente. Embora o AccessHub incentive o reúso de código, essas modificações causaram pequenas alterações no comportamento das funcionalidades, fazendo com que elas não atendessem totalmente aos objetivos propostos.
- O participante P2 conseguiu implementar todos os componentes de funcionalidades propostos, exceto o vídeo. Em vez de implementar os botões que realizam as funcionalidades do vídeo, o participante optou pelo atributo "*controls*" e implementou um botão para baixar o arquivo de transcrição textual do vídeo. Isso sugere que talvez o texto do desafio nessa parte não tenha sido claro o suficiente sobre a implementação dessas funcionalidades sem

<sup>1</sup> Disponível em: <https://www.google.com/intl/pt-br/drive/about.html>

- utilizar o que já vem nativamente da *tag* de vídeo.
- O participante P3 conseguiu implementar seis dos sete componentes de funcionalidades propostos, faltando somente a navegação pelo teclado na tabela. Além disso, o mesmo ofereceu sugestões de melhoria em trechos dos códigos fontes, sugestões essas que não causaram impacto negativo no comportamento do código, mostrando que o participante buscou se aprofundar no conteúdo disponível no projeto e em outras fontes.
  - Analisando o projeto do participante P4, notamos que o mesmo implementou cinco dos sete componentes de funcionalidade, faltando apenas o botão para saltar para o conteúdo principal e a navegação pelo teclado na tabela.
  - O participante P5 conseguiu realizar a implementação de todos os componentes de funcionalidades de acessibilidade propostos, demonstrando um ótimo desempenho no desafio.
  - O participante P6 conseguiu implementar seis dos sete componentes de funcionalidades de acessibilidade, mas três deles funcionaram parcialmente, contendo erros no código, possivelmente por falta de atenção do participante. Além disso, o componente de vídeo não foi implementado.
  - Analisando o projeto enviado pelo participante P7, foi constatado que o mesmo não implementou nenhum componente de funcionalidade de acessibilidade. Ao que parece, o participante focou apenas em solucionar os erros de código presentes no desafio, sem considerar as funcionalidades de acessibilidade. Portanto, foi necessário desqualificá-lo no desafio, pois sua opinião final não teria relevância nas respostas do segundo questionário em relação à validação do AccessHub.
  - Por fim, o participante P8 implementou somente as funcionalidades de pular navegação, alterar o tamanho da fonte dos textos e os controles básicos do vídeo.

A análise dos projetos dos participantes permitiu avaliar o desempenho individual de cada um em relação à implementação dos componentes de funcionalidades de acessibilidade propostos no desafio técnico. Embora a maioria dos participantes tenha alcançado um bom nível de sucesso na implementação dessas funcionalidades, foram identificadas algumas oportunidades de melhoria.

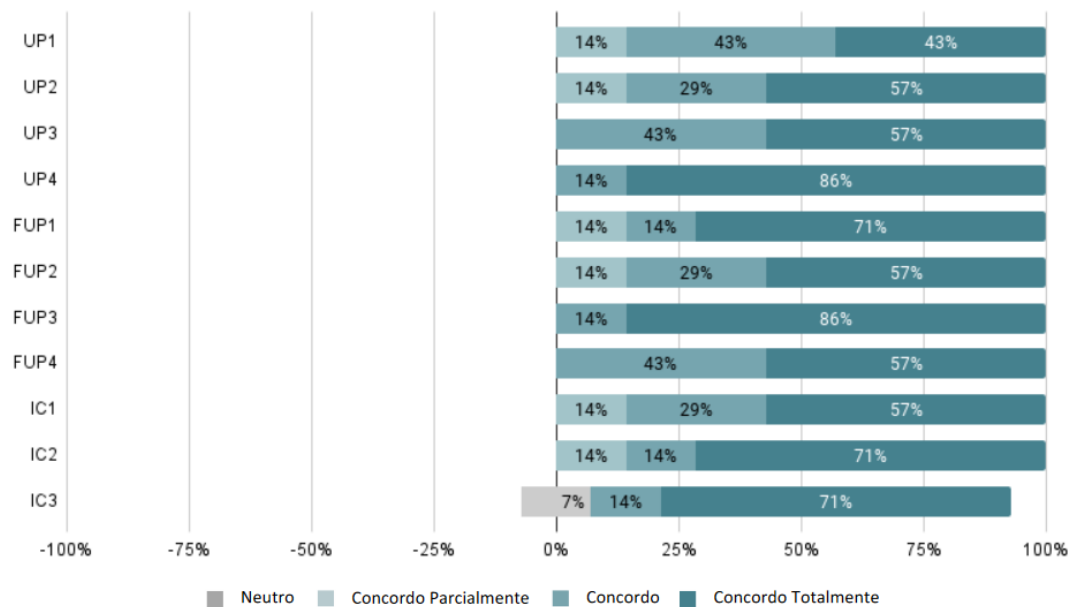
#### **6.4 Questionário TAM**

Dado que o questionário TAM foi adaptado ao contexto do projeto, foi avaliado a credibilidade das escalas calculando o *Alpha* de Cronbach, que de acordo com Taber (2018)

é uma estatística que fornece um valor entre 0 e 1 que indica a consistência interna da escala. Embora não exista uma regra para o valor desejado de alfa, geralmente os pesquisadores, como Iacobucci e Duhachek (2003), consideram 0,70 como um nível aceitável de consistência. Os resultados indicaram alta consistência interna em todos os três construtos, como segue: UP = 0,950, FUP = 0,857 e IC = 1,0.

No geral, a maioria dos participantes concordou com todas as afirmações da TAM. As medianas para Utilidade Percebida (UP), Facilidade de Uso Percebida (FUP) e Intenção Comportamental (IC) foram 7, 7 e 7 respectivamente, indicando uma percepção positiva dos participantes em relação à proposta do AccessHub (ver Figura 9).

Figura 9 – Resultados do questionário TAM



Fonte: Elaborado pelo autor.

Em relação às UP, a maioria dos participantes considerou o projeto útil. Os itens UP3 e UP4 foram os que apresentaram maior índice de concordância entre os participantes. Eles consideraram que usar o AccessHub melhora a efetividade para implementar recursos de *Acessibilidade Web* (UP3) e é, em geral, útil para apoiar essa tarefa (UP4). No entanto, o participante P8 concorda parcialmente com essa afirmativa que usar o AccessHub melhora o desempenho para implementar recursos de *Acessibilidade Web*. Isso indica que eles podem ter enfrentado alguma dificuldade em utilizar os exemplos de código.

Em relação às FUP, as opiniões dos participantes foram mais positivas. Todos os participantes concordaram com todas as quatro afirmações, principalmente o item FUP3,

considerando que o projeto é fácil de usar. O participante P8, no entanto, concordou parcialmente em que o AccessHub foi claro e fácil de entender (FUP1) e usar o AccessHub não exige muito esforço mental (FUP2), mostrando que o mesmo precisou de uma atenção maior para aprender.

Por fim, em relação às IC, somente o participante P2 que concordou parcialmente que assumindo que ele tenha acesso ao AccessHub, pretende usá-lo para implementar recursos de Acessibilidade *Web* (IC1), como também prevê que vai utilizá-lo para implementar recursos de Acessibilidade *Web* (IC2), além disso o mesmo manteve-se neutro em relação se vai utilizar o projeto nos próximos meses (IC3). Entretanto, os resultados indicam uma aceitação positiva do projeto por parte do restante dos participantes.

## 6.5 Perguntas abertas

Quando questionados se tiveram alguma dificuldade para entender e utilizar os recursos de acessibilidade fornecidos pelo AccessHub, ficou evidente que a maioria dos desenvolvedores não enfrentou grandes desafios nessa área. Em geral, eles consideraram o conteúdo fornecido pelo AccessHub claro e de fácil compreensão.

Os participantes P2, P3, P4, P5 e P6 expressaram que não tiveram dificuldades significativas em entender e utilizar os recursos de acessibilidade disponibilizados pelo AccessHub. Eles destacaram que a documentação fornecida era bem clara e que os exemplos apresentados foram úteis para orientá-los na implementação das funcionalidades de acessibilidade.

Por outro lado, o participante P1 relatou que “Senti falta de clareza nos passos a serem feitos na máscara de leitura”. Essa observação ressalta a importância de garantir que todas as informações sejam apresentadas de forma clara, para que os desenvolvedores possam compreender melhor a aplicação das funcionalidades propostas.

Por fim, o participante P8 enfatizou que o conteúdo do AccessHub estava organizado de forma fluída e que “cada componente estava em seu devido tópico, fácil de entender com uma boa descrição”. Essa percepção, reforça o impacto positivo da organização adequada das informações, facilitando a compreensão e utilização dos recursos de acessibilidade.

Ao questionar os participantes sobre o que foi fácil ao utilizar o AccessHub, percebeu-se que vários aspectos do projeto foram elogiados por sua praticidade e eficácia, revelando os seguintes pontos positivos.

O participante P1 destacou que “foi fácil de utilizar a manipulação de eventos de vídeo, a criação de navegação pelo teclado e a estruturação correta de *hiperlinks*, assim como os

exemplos que deixam ainda mais clara a explicação". Destacando que a presença de exemplos claros contribuiu significativamente para o entendimento das explicações.

O participante P2 apreciou o site de forma geral, mencionando que o mesmo “tem um *design* bem simplista e organizado”, como fatores que facilitaram a localização e compreensão das informações necessárias para a realização do desafio técnico.

Para o participante P3, a “leitura e entendimento dos conceitos disponíveis pelo *site*, bem como de manuseio do sistema”, foram fáceis de assimilar. Essa facilidade de compreensão demonstra a eficácia da apresentação das informações no AccessHub.

O participante P4 elogiou a descrição detalhada de cada componente e ação na documentação do AccessHub. Além disso, a apresentação de diferentes abordagens e suas finalidades foi considerada de acordo com o mesmo “bem interessante no dia a dia do *dev*”.

De forma similar, o participante P5 ressaltou “a clareza nos detalhes da explicação de cada um dos componentes” e a importância dos exemplos fornecidos para auxiliar na realização do desafio. O participante P6 destacou a “forma didática e autoexplicativa do conteúdo”, o que contribuiu para a facilidade de compreensão dos tópicos abordados.

Por fim, o participante P8 mencionou que a facilidade de localização dos componentes e a boa descrição das informações disponíveis no *site* contribuíram para a boa experiência do usuário ao utilizar o AccessHub.

Ao questionar os participantes sobre o que foi difícil ao utilizar o AccessHub, obteve-se um conjunto de respostas bastante variadas, que forneceu *insights* valiosos para possíveis melhorias no projeto. O participante P1 mencionou que, com exceção da clareza do componente “máscara de leitura”, não encontrou dificuldades significativas durante a utilização do AccessHub. O participante P2 relatou que não teve dificuldades específicas ao utilizar a plataforma, indicando uma boa experiência de uso.

No entanto, o participante P3 identificou um ponto que poderia ser aprimorado. Ele mencionou a dificuldade em localizar implementações com focos distintos reunidos em uma mesma página, especialmente no componente de vídeo. A sugestão do participante é que “códigos para objetivos diferentes que poderiam ser separados em outras abas no menu para facilitar a localização”, proporcionando uma experiência mais fluida ao navegar pelos recursos. Por outro lado, o participante P4 expressou que a plataforma é objetiva e clara em relação a cada recurso disponível, não enfrentando dificuldades durante a utilização.

Similarmente, o participante P5 não teve dificuldades em relação ao uso do Acces-

sHub, o que indica que a plataforma conseguiu atender de forma satisfatória às necessidades desse participante. O participante P6 destacou que a documentação está bem organizada e “os exemplos apresentados auxiliam bastante na implementação”, indicando uma boa experiência geral.

Por fim, o participante P8 mencionou a dificuldade em entender visualmente como o componente ficaria disposto na tela, além de um ponto de melhoria na facilidade de poder copiar trechos de códigos distintos. O participante sugeriu que um botão de copiar para cada bloco distinto de código (HTML, CSS e *JavaScript*), poderia proporcionar mais comodidade ao selecionar e copiar os códigos necessários.

Ao questionar os participantes sobre possíveis melhorias no AccessHub, foi possível obter *feedback* valioso, que permitiu identificar áreas em que o projeto pode ser aprimorado para atender melhor às necessidades dos desenvolvedores.

O participante P1 sugeriu a adição de uma barra de pesquisa para “facilitar o encontro de certos elementos acessíveis” específicos. Além disso, o participante destacou a importância de organizar os recursos por tópicos, como "acessibilidades visuais", "acessibilidades auditivas" e "acessibilidades para o navegador", o que tornaria mais fácil para os usuários encontrarem os recursos adequados às suas necessidades.

Os participantes P2 e P4 sugeriram que inserir fotos ou *prints* em alguns pontos da documentação poderia agilizar o entendimento de como seria a aparência dos elementos descritos. Essa adição visual poderia ajudar os desenvolvedores a visualizarem melhor o resultado final das implementações.

O participante P3 também enfatizou a importância de exemplos práticos no *site*. Ele sugeriu que ter exemplos visuais para cada código seria uma maneira eficiente de fornecer uma explicação visual para os usuários, permitindo que eles compreendam melhor como cada componente de acessibilidade se comportaria na prática.

O participante P5 ofereceu uma sugestão relevante sobre a criação de uma “biblioteca com os componentes pré-prontos do AccessHub para outros *frameworks frontend*”. Essa adição permitiria que os desenvolvedores incorporassem facilmente os recursos de acessibilidade em seus projetos utilizando seus *frameworks* preferidos, o que seria especialmente útil em um cenário em que o uso de *frameworks* é bastante comum.

Outra sugestão interessante veio do participante P6, que propôs a implementação de um painel de visualização para executar o código de exemplo. Essa funcionalidade permitiria



aos desenvolvedores verem o resultado prático dos códigos e compreenderem seu funcionamento de forma mais interativa.

Por fim, o participante P8 sugeriu melhorias na disposição das descrições, destacando que os textos estão muito extensos e podem levar à perda de foco durante a leitura. Ele também sugeriu dividir os códigos em categorias (HTML, CSS e *JavaScript*) e adicionar botões de cópia individuais para cada um deles, o que tornaria mais fácil para os desenvolvedores copiarem o código necessário. Além disso, a inclusão de exemplos visuais de como os componentes seriam dispostos na página seria benéfica para uma melhor compreensão das implementações.

## 7 CONCLUSÕES E TRABALHOS FUTUROS

Com o propósito de auxiliar no desenvolvimento de *websites* que proporcionem uma experiência digital mais acessível, este trabalho se dedicou na criação de um guia abrangente sobre funcionalidades de acessibilidade para desenvolvedores, abordando padrões, materiais de referência e disponibilizando um conjunto de exemplos de código *open source* que possam ser facilmente acessados, compreendidos e compartilhados. O AccessHub, em conformidade com as diretrizes e recomendações do WCAG e do eMAG, busca simplificar o processo de implementação de recursos acessíveis em *websites*, capacitando os desenvolvedores na criação de seus projetos, permitindo que pessoas com diversas necessidades tenham acesso inclusivo ao conteúdo digital.

Inicialmente, é apresentado, por meio de pesquisas na literatura, os conceitos fundamentais sobre acessibilidade na *web*, reúso de *software* e projetos *open source*, sendo estes os pilares deste trabalho, permitindo a compreensão do leitor sobre a proposta geral do projeto. Em seguida, os critérios de seleção das diretrizes, o desenvolvimento do AccessHub com os seus componentes de funcionalidades e as metodologias de validação são mostradas e discutidas no decorrer do trabalho.

Logo após, é apresentado uma visão geral do AccessHub, descrevendo as tecnologias utilizadas para o seu desenvolvimento e explicando em detalhes sobre os componentes de funcionalidades desenvolvidos, fornecendo um exemplo prático de como utilizá-los.

A partir da avaliação individual e do desafio técnico proposto, os resultados obtidos foram extremamente valiosos, revelando que o projeto está bem estruturado, e constatando que, em geral, a experiência dos desenvolvedores com o AccessHub foi positiva, de tal forma que a maioria deles encontraram facilidade em entender e utilizar os recursos de acessibilidade disponibilizados pela projeto.

Um aspecto destacado pelos participantes foi a clareza das informações presentes no AccessHub. Eles relataram que a documentação fornecida era bem explicativa e contava com exemplos práticos que auxiliaram no processo de implementação dos recursos de acessibilidade. Além disso, apreciaram a organização dos componentes em tópicos específicos, facilitando a localização das informações relevantes.

Considerando as sugestões apresentadas pelos participantes, é identificado algumas áreas que podem ser melhoradas em trabalhos futuros, podendo contribuir para aprimorar ainda mais o projeto. Sendo essas, a inclusão de exemplos visuais para cada código que permitam

uma melhor visualização do resultado final das implementações. A divisão dos códigos em categorias, permitindo a adição de botões de copiar individuais, que oferecem a possibilidade dos desenvolvedores escolherem facilmente os códigos necessários. Como também, a criação de uma biblioteca com os componentes pré-prontos para outros *frameworks Front-end*, considerando o cenário atual do desenvolvimento *web*.

Diante dessas conclusões, pode-se afirmar que o AccessHub é uma ferramenta promissora para o desenvolvimento de projetos *web* acessíveis. Sua capacidade de fornecer recursos prontos para uso e a facilidade de implementação foram elogiadas pelos desenvolvedores, tornando-a uma opção viável de ferramenta para garantir a acessibilidade digital.

Espera-se que este estudo contribua para a conscientização sobre a importância da acessibilidade digital e para a disseminação de práticas inclusivas no desenvolvimento de projetos *web*. Com a colaboração contínua de desenvolvedores e da comunidade em geral, podemos tornar a *web* um ambiente mais acessível para todos os usuários, independentemente de suas habilidades ou limitações.

## REFERÊNCIAS

- ACOSTA, T.; ZAMBRANO-MIRANDA, J.; LUJÁN-MORA, S. Techniques for the publication of accessible multimedia content on the web. **IEEE access**, IEEE, v. 8, p. 55300–55322, 2020.
- AHMARO, I.; ABUALKISHIK, A. M.; YUSOFF, M. Z. M. Taxonomy, definition, approaches, benefits, reusability levels, factors and adaptation of software reusability: a review of the research literature. **Journal of Applied Sciences**, Asian Network for Scientific Information (ANSINET), v. 14, n. 20, p. 2396, 2014.
- ALAJARMEH, N. The extent of mobile accessibility coverage in wcag 2.1: Sufficiency of success criteria and appropriateness of relevant conformance levels pertaining to accessibility problems encountered by users who are visually impaired. **Universal Access in the Information Society**, Springer, p. 1–26, 2021.
- ANASUODEI, M.; OJEKUDO, N. A. Software reusability: Approaches and challenges. **International Journal of Research and Innovation in Applied Science**, RSIS, v. 6, p. 142–146, 2021.
- BARROS-JUSTO, J. L.; BENITTI, F. B.; MATALONGA, S. Trends in software reuse research: A tertiary study. **Computer Standards & Interfaces**, Elsevier, v. 66, p. 103352, 2019.
- BOGNER, J.; MERKEL, M. To type or not to type? a systematic comparison of the software quality of javascript and typescript applications on github. In: **Proceedings of the 19th International Conference on Mining Software Repositories**. [S. l.: s. n.], 2022. p. 658–669.
- BRITTO, T. C. P. **GAIA**: uma proposta de guia de recomendações de acessibilidade web com foco em aspectos do autismo. Universidade Federal de São Carlos, 2016.
- CAMPOVERDE-MOLINA, M.; LUJAN-MORA, S.; GARCIA, L. V. Empirical studies on web accessibility of educational websites: A systematic literature review. **IEEE Access**, IEEE, v. 8, p. 91676–91700, 2020.
- CECI, M.; LANOTTE, P. F. Closed sequential pattern mining for sitemap generation. **World Wide Web**, Springer, v. 24, n. 1, p. 175–203, 2021.
- CHAKRA UI. **Chakra UI - Build Accessible React Apps with Speed**. 2023. Disponível em: <https://github.com/chakra-ui/chakra-ui/>. Acesso em: 19 jun. 2023.
- CHAN-JONG-CHU, K.; ISLAM, T.; EXPOSITO, M. M.; SHEOMBAR, S.; VALLADARES, C.; PHILIPPOT, O.; GRUA, E. M.; MALAVOLTA, I. Investigating the correlation between performance scores and energy consumption of mobile web apps. In: **Proceedings of the evaluation and assessment in software engineering**. [S. l.: s. n.], 2020. p. 190–199.
- DABHADE, M.; SURYAWANSHI, S.; MANJULA, R. A systematic review of software reuse using domain engineering paradigms. In: IEEE. **2016 Online International Conference on Green Engineering and Technologies (IC-GET)**. [S. l.], 2016. p. 1–6.
- DEVEZER, B.; NAVARRO, D. J.; VANDEKERCKHOVE, J.; BUZBAS, E. O. The case for formal methodology in scientific reform. **Royal Society open science**, The Royal Society, v. 8, n. 3, p. 200805, 2020.

FERDOUS, J.; UDDIN, S.; ASHOK, V. Semantic table-of-contents for efficient web screen reading. In: **Proceedings of the 36th Annual ACM Symposium on Applied Computing**. [S. l.: s. n.], 2021. p. 1941–1949.

FRANÇOISE, J.; CARAMIAUX, B.; SANCHEZ, T. Marcelle: composing interactive machine learning workflows and interfaces. In: **The 34th Annual ACM Symposium on User Interface Software and Technology**. [S. l.: s. n.], 2021. p. 39–53.

GitHub. **Hello World**. 2023. Disponível em: <https://docs.github.com/en/get-started/quickstart/hello-world>. Acesso em: 30 jun. 2023.

HARGREAVES, F. d. O. **Catálogo de práticas de acessibilidade**: um apoio online voltado à acessibilidade da web. 2021.

IACOBUCCI, D.; DUHACHEK, A. Advancing alpha: Measuring reliability with confidence. **Journal of consumer psychology**, Elsevier, v. 13, n. 4, p. 478–487, 2003.

ISMAIL, A.; KUPPUSAMY, K. Web accessibility investigation and identification of major issues of higher education websites with statistical measures: A case study of college websites. **Journal of King Saud University-Computer and Information Sciences**, Elsevier, 2019.

KAWAGUCHI, S.; GARG, P. K.; MATSUSHITA, M.; INOUE, K. Mudablue: An automatic categorization system for open source repositories. In: IEEE. **11th Asia-Pacific Software Engineering Conference**. [S. l.], 2004. p. 184–193.

KRASNIQI, V.; ZDRAVKOVA, K.; DALIPI, F. Impact of assistive technologies to inclusive education and independent life of down syndrome persons: A systematic literature review and research agenda. **Sustainability**, MDPI, v. 14, n. 8, p. 4630, 2022.

KULKARNI, M. Digital accessibility: Challenges and opportunities. **IIMB Management Review**, Elsevier, v. 31, n. 1, p. 91–98, 2019.

LI, S.; KANG, M.; HOU, J.; CAO, Y. Mining node.js vulnerabilities via object dependence graph and query. In: **31st USENIX Security Symposium (USENIX Security 22)**. [S. l.: s. n.], 2022. p. 143–160.

Lighthouse. **Overview**. 2023. Disponível em: <https://developer.chrome.com/docs/lighthouse/overview/>. Acesso em: 26 jun. 2023.

MADSEN, M.; LHOTAK, O.; TIP, F. A semantics for the essence of react. In: **European Conference on Object-Oriented Programming**. [S. l.: s. n.], 2020.

MARTIN, P. C.; PIOVESAN, S. D. Análise da acessibilidade nos moocs das universidades federais do brasil em conformidade com os requisitos do w3c e emag. **RENOTE**, v. 17, n. 3, p. 51–60, 2019.

MDN Web Docs. **CSS and JavaScript accessibility best practices**. 2022. Disponível em: [https://developer.mozilla.org/en-US/docs/Learn/Accessibility/CSS\\_and\\_JavaScript](https://developer.mozilla.org/en-US/docs/Learn/Accessibility/CSS_and_JavaScript). Acesso em: 04 jul. 2022.

MDN Web Docs. **HTML: A good basis for accessibility**. 2022. Disponível em: <https://developer.mozilla.org/en-US/docs/Learn/Accessibility/HTML>. Acesso em: 03 jul. 2022.

MEDEIROS, G. F. **Guia de recomendações de acessibilidade para produção de interfaces web**. 2021.

NUÑEZ, A.; MOQUILLAZA, A.; PAZ, F. Web accessibility evaluation methods: a systematic review. In: SPRINGER. **International Conference on Human-Computer Interaction**. [S. l.], 2019. p. 226–237.

OLIVEIRA, A. C.; SILVA, L. F. da; ELER, M. M.; FREIRE, A. P. Do brazilian federal agencies specify accessibility requirements for the development of their mobile apps? In: **XVI Brazilian Symposium on Information Systems**. [S. l.: s. n.], 2020. p. 1–8.

Open Source Initiative. **Basics of Open Source**. 2022. Disponível em: <https://opensource.org/faq>. Acesso em: 24 jun. 2022.

Portal do Governo Brasileiro. **eMAG - Modelo de Acessibilidade em Governo Eletrônico**. 2022. Disponível em: <https://emag.governoeletronico.gov.br/>. Acesso em: 11 jun. 2022.

PRADO, E. P. V. An analysis of open source software adoption in a brazilian nonprofit university. In: SBC. **Anais do VIII Simpósio Brasileiro de Sistemas de Informação**. [S. l.], 2012. p. 320–330.

RAJAKUMARI, K. E. Towards a novel conceptual framework for analyzing code clones to assist in software development and software reuse. In: IEEE. **2020 4th International Conference on Intelligent Computing and Control Systems (ICICCS)**. [S. l.], 2020. p. 105–111.

Remix. **React Router v6**. 2023. Disponível em: <https://reactrouter.com/en/main>. Acesso em: 20 jun. 2023.

SALVADOR-ULLAURI, L.; ACOSTA-VARGAS, P.; GONZALEZ, M.; LUJÁN-MORA, S. Combined method for evaluating accessibility in serious games. **Applied Sciences**, Multidisciplinary Digital Publishing Institute, v. 10, n. 18, p. 6324, 2020.

SANDHU, A. K.; BATH, R. S. Integration of artificial intelligence into software reuse: An overview of software intelligence. In: IEEE. **2021 2nd International Conference on Computation, Automation and Knowledge Management (ICCAKM)**. [S. l.], 2021. p. 357–362.

SILVA, B. G. da; RODRIGUES, K. R. H. Accessibility challenges in web systems implementation. In: SBC. **Anais do IX Workshop sobre Aspectos da Interação Humano-Computador para a Web Social**. [S. l.], 2018. p. 105–116.

SOMMERVILLE, I. **Engenharia de Software**. [S. l.]: Pearson Prentice Hall, 2011. v. 9.

SOUSA, C. C.; OLIVEIRA, L. M.; RODRIGUES, C. L.; BULCÃO-NETO, R. F.; FERREIRA, D. J. Web accessibility testing for deaf: Requirements and approaches for automation. In: IEEE. **2020 IEEE International Conference on Systems, Man, and Cybernetics (SMC)**. [S. l.], 2020. p. 2734–2739.

SOUZA, I. M. de; MACIEL, C.; CAPPELLI, C. The model of accessibility to electronic government: applicability in dataprev. In: **Proceedings of the 17th International Digital Government Research Conference on Digital Government Research**. [S. l.: s. n.], 2016. p. 287–292.

SOUZA, M. de; ALMEIDA, F. G. Acessibilidade web dos sites das bibliotecas das universidades federais do estado de minas gerais. **RDBCI: Revista Digital de Biblioteconomia e Ciência da Informação**, v. 19, p. e021027–e021027, 2021.

Stack Overflow. **2022 Developer Survey**. 2022. Disponível em: <https://survey.stackoverflow.co/2022/>. Acesso em: 25 junho 2022.

TABER, K. S. The use of cronbach’s alpha when developing and reporting research instruments in science education. **Research in science education**, Springer, v. 48, p. 1273–1296, 2018.

TOFFOLON, C.; DAKHLI, S. Software artifacts reuse and maintenance: An organizational framework. In: IEEE. **Proceedings of the Second Euromicro Conference on Software Maintenance and Reengineering**. [S. l.], 1998. p. 228–233.

Vercel. **What does Vercel do?** 2023. Disponível em: <https://vercel.com/blog/what-is-vercel>. Acesso em: 30 jun. 2023.

WAI. **WCAG 2 Overview**. 2022. Disponível em: <https://www.w3.org/WAI/standards-guidelines/wcag/>. Acesso em: 08 jun. 2022.

WAI. **WCAG 2.1 at a Glance**. 2022. Disponível em: <https://www.w3.org/WAI/standards-guidelines/wcag/glance/>. Acesso em: 08 jun. 2022.

Web Accessibility Initiative. **Introduction to Web Accessibility**. 2022. Disponível em: <https://www.w3.org/WAI/fundamentals/accessibility-intro/>. Acesso em: 05 jun. 2022.

WOLFART, D.; ASSUNÇÃO, W. K. G.; MARTINEZ, J. Open source software on the research of extractive adoption of software product lines. In: SBC. **Anais do XVI Congresso Latino-Americano de Software Livre e Tecnologias Abertas**. [S. l.], 2019. p. 142–145.

## APÊNDICE A – ROTEIRO DO DESAFIO TÉCNICO

### A.1 *Landing Page* para a Venda de um Telescópio

Este projeto consiste no desenvolvimento de uma *Landing Page* para a venda de um Telescópio Refratário Astronômico Profissional. A página tem como objetivo apresentar e comercializar esse equipamento astronômico de alta qualidade, fornecendo aos entusiastas e profissionais da astronomia uma ferramenta poderosa para a exploração do céu noturno.

#### A.1.1 *Desafio*

A acessibilidade desempenha um papel fundamental no desenvolvimento de aplicações *web*, pois visa garantir que pessoas com diferentes habilidades e necessidades possam interagir e utilizar os recursos da página de forma efetiva. O desafio dessa pesquisa consiste em abordar e solucionar as questões de acessibilidade desta página que não atende aos requisitos básicos de acessibilidade, o que pode dificultar ou até mesmo impossibilitar o acesso e a compreensão do conteúdo para pessoas com deficiências visuais, auditivas, motoras ou cognitivas. Para abordar esse desafio, será utilizado o *website* do AccessHub que oferece recursos e funcionalidades de acessibilidade prontos para uso. A partir desse *site*, o desenvolvedor enfrentará o desafio de implementar e integrar essas funcionalidades na página do telescópio, tornando-a mais acessível e proporcionando uma experiência inclusiva para todos os usuários.

*Link para a Página do AccessHub*<sup>1</sup>

Além disso, espera-se também melhorar a pontuação de acessibilidade no *Google Lighthouse*, refletindo os esforços para atender aos padrões e diretrizes de acessibilidade.

*Link para o Plugin Google Lighthouse*<sup>2</sup>

Através desse desafio, busca-se destacar a importância da acessibilidade na criação de aplicações *web* e demonstrar como é possível implementar soluções eficazes para tornar o conteúdo acessível a todos os usuários. Espera-se que os resultados dessa pesquisa contribuam não apenas para a melhoria da página do telescópio, mas também para conscientizar e incentivar outros desenvolvedores a priorizarem a acessibilidade em seus projetos.

<sup>1</sup> Disponível em: <https://accesshub.vercel.app/>

<sup>2</sup> Disponível em: <https://chrome.google.com/webstore/detail/lighthouse/blipmdconlknpinefehnmjammfjpmphjk>



### A.1.2 *Passo a passo*

1. Clone este repositório para fazer o *download* do projeto.
2. Baixe o *plugin* do *Google Lighthouse*, disponível no *Google Chrome*.
3. Execute o projeto, e em seguida utilize o *plugin* para ver a pontuação da página. (OBS: caso o *plugin* não funcione, tente utilizar ele por meio do *Dev Tools* do *Google* apertando F12 em seu teclado).
4. A partir disso, visite o *site* do AccessHub para aprender mais sobre acessibilidade *Web*, você vai precisar dele para implementar as seguintes funcionalidades ao projeto:
  - Funcionalidade para pular navegação para o conteúdo principal;
  - Funcionalidade para mudar o tamanho da fonte dos texto da página;
  - Funcionalidade para mudar o contraste da página;
  - Funcionalidade para ouvir a descrição da primeira imagem da página;
  - Funcionalidade para acionar a máscara de leitura;
  - Navegação pelo teclado na tabela;
  - Estruturar corretamente o vídeo da página.
5. Existem alguns problemas de semântica e detalhes de boas práticas de desenvolvimento faltando no projeto, tente resolve-los. O ideal é que a pontuação obtida no *Google Lighthouse* esteja entre 90 - 100 nos requisitos de Acessibilidade e Boas Práticas para a entrega do desafio.

### A.1.3 *Entrega do desafio*

A partir da conclusão de todas as etapas do desafio, utilize o *Google Lighthouse* para ver sua pontuação total e nos envie um *print* juntamente com o código fonte do seu projeto em formato *.zip* no *link* do *Drive* disponibilizado pelo autor da pesquisa. Será fornecido um ID de usuário exclusivo para que possamos controlar os envios em pastas individuais, garantindo assim a organização e confidencialidade dos participantes da pesquisa.

### A.1.4 *Feedback*

Responda o seguinte questionário para a realização dessa pesquisa.

*Link* para o questionario de Percepção sobre o projeto + Questionário TAM<sup>3</sup>

<sup>3</sup> Disponível em: <https://forms.gle/XuEUcFgB8YAIRM5h8>

## APÊNDICE B – TERMO DE CONSENTIMENTO

Prezado Senhor (a),

Você está sendo convidado a participar de uma pesquisa intitulada "AccessHub: Guia de Acessibilidade para Desenvolvedores *Web*", cujo principal objetivo é desenvolver um guia para auxiliar na implementação de funcionalidades de acessibilidade, possibilitando a apresentação de padrões e materiais de suporte para a construção de páginas *Web* acessíveis por desenvolvedores.

### 1) Procedimentos

Você responderá a um questionário com perguntas relacionadas à percepção sobre a utilização do artefato proposto. Todos os questionários respondidos serão descartados após a transcrição. Quando os dados forem transcritos, seu nome será removido das transcrições e não será utilizado em nenhum momento durante a análise ou apresentação dos resultados. Sua imagem não será utilizada e nem divulgada, de forma a manter sua identidade em sigilo.

### 2) Tratamento de possíveis riscos e desconfortos

A sua participação consiste tão somente em utilizar o artefato proposto e avaliá-lo, com duração aproximada de 30 minutos, seguida do preenchimento de um questionário. Ainda assim, serão tomadas todas as providências durante a coleta desses dados de forma a garantir a sua privacidade e seu anonimato. Os dados coletados durante o estudo destinam-se estritamente a atividades da pesquisa. Desta forma, não serão utilizados para qualquer forma de avaliação profissional ou pessoal.

### 3) Confidencialidade da Pesquisa

Toda informação coletada neste estudo é confidencial, e seu nome e o da organização não serão identificados de modo algum, a não ser em caso de autorização explícita para esse fim.

### 4) Participação

Sua participação neste estudo é muito importante e voluntária. Você tem o direito de não querer participar ou de sair deste estudo a qualquer momento, sem penalidades. Você também

tem o direito de se recusar a responder a qualquer questão do questionário. Para participar deste estudo você deverá possuir idade maior ou igual a 18 anos.

Toda a dúvida a respeito desta pesquisa, poderá ser perguntada diretamente aos pesquisadores envolvidos: Davi Teixeira Silva, através do e-mail: daviteixeira.btm@gmail.com, José Cezar Junior de Souza Filho, através do e-mail: cezarfilho@ufc.br, e Carla Ilane Moreira Bezerra, através do e-mail: carlailane@ufc.br.

### **Declaração de consentimento**

- Li ou alguém leu para mim as informações contidas neste documento antes de assinar este termo de consentimento. Declaro que toda a linguagem técnica utilizada na descrição deste estudo de pesquisa foi explicada satisfatoriamente e que recebi respostas para todas as minhas dúvidas. Confirmo também que recebi uma cópia deste Termo de Consentimento Livre e Esclarecido. Compreendo que sou livre para me retirar do estudo em qualquer momento, sem qualquer penalidade. Declaro ter mais de 18 anos e dou meu consentimento de livre e espontânea vontade para participar deste estudo.

## APÊNDICE C – QUESTIONÁRIO DE PERFIL

### Questionário de perfil

Inicialmente, faremos algumas perguntas para conhecer a sua experiência profissional em desenvolvimento de *software*.

1. **Qual foi o seu último papel (ou papel atual) na sua equipe de desenvolvimento de *software*?**
  - Desenvolvedor *Web Front-end*
  - Desenvolvedor *Web Back-end*
  - Desenvolvedor *Web FullStack*
  - Engenheiro de Requisitos / *Product Owner* (PO)
  - UI/UX Designer*
  - Projetista / Arquiteto de *Software*
  - Testador
  - Analista de *Quality Assurance* (QA)
  - Gerente de Projeto
  - Líder Técnico
  - Outros...
2. **Quanto tempo de experiência profissional você possui nesse papel?**
  - Menos de 1 ano
  - Entre 1 e 3 anos
  - Entre 4 e 6 anos
  - Entre 7 e 9 anos
  - Mais de 10 anos
3. **Você cursa alguma graduação na área de Tecnologia da Informação? Se sim, qual o seu curso e semestre? (Exemplo: Sistemas de Informação / 5º semestre).**
4. **Qual é o seu nível de experiência em Acessibilidade Web?**
  - Não tenho conhecimento sobre Acessibilidade *Web*.
  - Estou aprendendo sobre o tema, mas nunca implementei recursos de Acessibilidade *Web*.

- Tenho algum conhecimento e implementei recursos de Acessibilidade *Web* em pelo menos 1 projeto.
  - Implementei recursos de Acessibilidade *Web* em vários projetos, mas não de forma constante.
  - Implemento recursos de Acessibilidade *Web* constantemente como parte do meu trabalho.
5. **Você já participou de algum treinamento ou curso específico sobre Acessibilidade *Web*? Se sim, poderia informar qual o nome do treinamento/curso?**
6. **Qual das seguintes diretrizes de Acessibilidade *Web* você possui familiaridade?**
- Não tenho familiaridade com diretrizes de Acessibilidade *Web*
  - WCAG - *Web Content Accessibility Guidelines*
  - eMAG - Modelo de Acessibilidade em Governo Eletrônico
  - Outros...
7. **A equipe em que você trabalha adota práticas de Acessibilidade *Web*?**
- Sim - Nossa equipe possui um compromisso forte com a Acessibilidade *Web* e implementa práticas consistentes em nossos projetos.
  - Parcialmente - Nossa equipe tem algum nível de consciência sobre Acessibilidade *Web*, mas ainda temos espaço para melhorias.
  - Não - A Acessibilidade *Web* não é uma prioridade em nossa equipe ou não temos recursos para implementá-la adequadamente.

Agradeço sinceramente por dedicar seu tempo para responder a este questionário.

Sua contribuição é extremamente valiosa para o nosso projeto. Muito obrigado!





**2. Dado que eu tenha acesso ao AccessHub, prevejo que vou utilizá-lo para implementar recursos de Acessibilidade Web**

Discordo totalmente    1    2    3    4    5    6    7    Concordo totalmente  
○    ○    ○    ○    ○    ○    ○

**3. Pretendo usar o AccessHub para implementar recursos de Acessibilidade Web nos próximos meses**

Discordo totalmente    1    2    3    4    5    6    7    Concordo totalmente  
○    ○    ○    ○    ○    ○    ○