

Eduardo Pereira de Almeida

# **Wilia: Um UI kit de anotações para Documentação de acessibilidade, para leitores de tela, em projetos Web no Figma**

Salinas - Minas Gerais

9 de novembro de 2025

Eduardo Pereira de Almeida

# **Wilia: Um UI kit de anotações para Documentação de acessibilidade, para leitores de tela, em projetos Web no Figma**

Trabalho de conclusão de curso apresentado ao Curso de Bacharelado em Sistemas de Informação do Instituto Federal do Norte de Minas Gerais, como exigência para obtenção do grau de Bacharel em Sistemas de Informação.

Instituto Federal do Norte de Minas Gerais - Campus Salinas

Bacharelado em Sistemas de Informação

Orientador: Arthur Faria Porto

Salinas - Minas Gerais

9 de novembro de 2025

*À minha mãe, Maria Silvia Pereira de Almeida,  
e às minhas irmãs, Edna Pereira de Almeida e Edilaine Pereira Costa.  
Vocês foram meu pilar e a base que me manteve firme.*

*Ao meu pai, Ubaldo de Almeida Costa (in memoriam),  
que infelizmente não pôde ver esta conclusão,  
mas me acompanha no caminho de sucesso que venho trilhando.*

*Costuma-se dizer que a acessibilidade  
só se torna importante quando precisamos dela.  
Dedico este trabalho a todos que precisam dela todos os dias,  
como pessoas cegas, com baixa visão ou qualquer outra deficiência.*

# AGRADECIMENTOS

A conclusão deste trabalho representa o fim de uma longa jornada iniciada em 2018. Um caminho marcado por desafios que foram muito além da academia: enfrentei a pandemia, momentos de ansiedade e angústia, as cobranças da vida, a dor de perder alguém e, em muitas ocasiões, o desânimo e preocupação financeira me fez querer desistir. Por isso, este agradecimento é, antes de tudo, um reconhecimento a quem tornou esta conquista possível.

Agradeço imensamente ao meu orientador, professor **Arthur Faria Porto**, por acreditar neste projeto quando eu já estava desistindo do curso. Sua disposição em assumir este TCC, sua motivação, elogios e melhorias à escrita foram fundamentais para que eu mantivesse a vontade. Sua orientação foi essencial para provar a importância de um tema tão relevante e ainda tão pouco tratado.

Agradeço aos membros da banca examinadora, professora **Danielle Miranda Rodrigues** e professor **Leonardo Humberto Guimaraes Silva**, por aceitarem o convite, pelo tempo dedicado e pelas valiosas contribuições que certamente enriqueceram esta pesquisa.

Ao **Instituto Federal do Norte de Minas Gerais**, pela oportunidade de formação.

O pilar mais importante desta trajetória foi, sem dúvida, minha família.

À minha mãe, **Maira Silvia Pereira de Almeida**, que sempre me apoiou, incentivou e se manteve firme por mim, mesmo quando eu não conseguia me manter. Seu apoio incondicional foi minha base para não desistir.

Às minhas irmãs, **Edna Pereira de Almeida** e **Edilaine Pereira Costa**, por todo o apoio (com um agradecimento especial à Edna por todo o suporte estrutural e cuidados), carinho e até mesmo pelas cobranças, que, à sua maneira, nunca me deixaram esquecer o objetivo final.

A todos que, de alguma forma, acreditaram em mim, meu muito obrigado.

*“O espírito humano precisa prevalecer sobre a tecnologia.”*

— Albert Einstein

# RESUMO

A acessibilidade digital é fundamental para garantir que pessoas com deficiência possam interagir com interfaces de forma igualitária. Contudo, a ausência de especificações claras e padronizadas de requisitos de acessibilidade, especialmente relacionados a tecnologias assistivas como leitores de tela, representa um grande desafio no processo de design. Este trabalho propõe o desenvolvimento do Wilia, um *UI Kit* para Figma voltado à documentação de requisitos de acessibilidade, com foco na experiência de usuários que utilizam leitores de tela. Baseado nas diretrizes WCAG (*Web Content Accessibility Guidelines*) e WAI-ARIA (*Web Accessibility Initiative - Accessible Rich Internet Applications*), o kit organiza componentes e guias visuais em uma estrutura modular e componentizada, integrando anotações e documentação técnica diretamente no ambiente do Figma. O resultado é uma ferramenta que facilita a comunicação entre equipes multidisciplinares, reduzindo erros de implementação semântica e operacional. Além disso, o *Wilia* tem o potencial de contribuir para a criação de interfaces mais inclusivas, alinhadas às boas práticas internacionais e às políticas nacionais de acessibilidade digital.

*Palavras-chave:* Acessibilidade, UI Kit, Figma, WCAG, tecnologias assistivas, componentes reutilizáveis, atributos semânticos, design inclusivo, código acessível.

# ABSTRACT

Digital accessibility is essential to ensure that people with disabilities can interact with digital interfaces on an equal basis. However, the absence of clear and standardized specifications for accessibility requirements, especially related to assistive technologies like screen readers, represents a significant challenge in the design process. This paper proposes the development of Wilia, a *UI Kit* for Figma focused on documenting accessibility requirements, specifically for the screen reader user experience. Based on the WCAG (*Web Content Accessibility Guidelines*) and WAI-ARIA (*Web Accessibility Initiative - Accessible Rich Internet Applications*) guidelines, the kit organizes components and visual guides in a \*\*modular and component-based structure\*\*, integrating annotations and technical documentation directly within the Figma environment. The result is a tool that facilitates communication between multidisciplinary teams, reducing semantic and operational implementation errors. Furthermore, *Wilia* has the potential to contribute to the creation of more inclusive interfaces, aligned with international best practices and national digital accessibility policies.

*Keywords:* Accessibility, UI Kit, Figma, WCAG, assistive technologies, reusable components, semantic attributes, inclusive design, accessible code.

# LISTA DE ILUSTRAÇÕES

Figura 1 – Evolução histórica do WCAG . . . . .	31
Figura 2 – Fonte: (DOBBALA; LINGOLU, 2024) . . . . .	31
Figura 3 – Uso de mais um elemento visual para representar o conteúdo . . . . .	41
Figura 4 – Comparação de tags semânticas em código HTML . . . . .	44
Figura 5 – Exemplo de aplicação do kit da CVS Health . . . . .	50
Figura 6 – Exemplo de aplicação do Pencil A11Y Kit . . . . .	50
Figura 7 – Trecho da visão geral do Wilia. . . . .	61
Figura 8 – Anatomia dos componentes Wilia. . . . .	62
Figura 9 – Exemplo de alteração de uma etiqueta Título para uma Região com um componente de multiespecificação . . . . .	63
Figura 10 – Exemplo de uma incorporação técnica vista na aba de propriedades de uma etiqueta . . . . .	64
Figura 11 – Exemplo de descrição de cenários da etiqueta Região . . . . .	65
Figura 12 – Exemplo de documentação de acessibilidade com o Wilia para semântica . . . . .	66
Figura 13 – Exemplo de documentação de acessibilidade com o Wilia para ordem de leitura e de foco . . . . .	67
Figura 14 – Componente: Regiões/Landmarks . . . . .	77
Figura 15 – Componente: Cabeçalhos . . . . .	78
Figura 16 – Componente: Botões . . . . .	79
Figura 17 – Componente: Campos de Entrada . . . . .	80
Figura 18 – Componente: Imagens . . . . .	81
Figura 19 – Componente: Links . . . . .	82
Figura 20 – Componente: Ordem de Leitura . . . . .	83
Figura 21 – Componente: Ordem de Foco . . . . .	84

# LISTA DE TABELAS

Tabela 1 – Resumo comparativo dos níveis de conformidade da WCAG . . . . .	34
Tabela 2 – Comparação entre Semântica Nativa e WAI-ARIA . . . . .	36
Tabela 3 – Comparação entre WCAG, Marco Civil da Internet e ABNT NBR	38
Tabela 4 – Exemplos de boas práticas para leitores de tela . . . . .	43
Tabela 5 – Comparação entre leitores de tela populares . . . . .	45
Tabela 6 – Análise comparativa entre os principais kits de anotação de acessibilidade . . . . .	48

# LISTA DE ABREVIATURAS E SIGLAS

IFNMG	Instituto Federal do Norte de Minas Gerais
Ex.	Exemplo
ONG	Organização sem fins lucrativos
WebAIM	Web Accessibility In Mind
OMS	Organização Mundial da Saúde
WHO	World Health Organization
W3C	World Wide Web Consortium
WCAG	Web Content Accessibility Guidelines
WAI-ARIA	Accessible Rich Internet Applications Suite

# SUMÁRIO

<b>1</b>	<b>INTRODUÇÃO</b>	<b>23</b>
<b>1.1</b>	<b>Objetivos</b>	<b>24</b>
1.1.1	Objetivo Geral	24
1.1.2	Objetivos Específicos	25
<b>1.2</b>	<b>Estrutura dos capítulos</b>	<b>25</b>
<b>2</b>	<b>REVISÃO DA LITERATURA</b>	<b>27</b>
<b>2.1</b>	<b>Acessibilidade na web e exclusão digital</b>	<b>27</b>
2.1.1	Legislação	29
<b>2.2</b>	<b>Diretrizes e Padrões</b>	<b>30</b>
2.2.1	WCAG	30
2.2.2	WAI-ARIA	34
2.2.3	ATAG	36
<b>2.3</b>	<b>Leitores de tela e sua interpretação de interfaces</b>	<b>38</b>
<b>2.4</b>	<b>Design de interface e documentação no Figma</b>	<b>45</b>
<b>2.5</b>	<b>Uso de kits e anotações visuais para acessibilidade</b>	<b>47</b>
<b>3</b>	<b>METODOLOGIA</b>	<b>55</b>
<b>3.1</b>	<b>Fundamentação Teórica e Delimitação do Escopo de Atuação</b>	<b>55</b>
<b>3.2</b>	<b>Análise Comparativa de Artefatos e Modelagem Estratégica</b>	<b>56</b>
<b>3.3</b>	<b>Concepção do Kit no Figma</b>	<b>57</b>
<b>4</b>	<b>RESULTADOS</b>	<b>59</b>
<b>5</b>	<b>CONCLUSÃO</b>	<b>69</b>
	<b>REFERÊNCIAS</b>	<b>71</b>
	<b>ANEXO A – VISÃO POR COMPONENTE WILIA</b>	<b>77</b>

<b>A.1</b>	<b>Regiões</b>	<b>77</b>
<b>A.2</b>	<b>Cabeçalhos</b>	<b>78</b>
<b>A.3</b>	<b>Botões</b>	<b>79</b>
<b>A.4</b>	<b>Campos de Entrada</b>	<b>80</b>
<b>A.5</b>	<b>Imagens</b>	<b>81</b>
<b>A.6</b>	<b>Links</b>	<b>82</b>
<b>A.7</b>	<b>Ordem de Leitura</b>	<b>83</b>
<b>A.8</b>	<b>Ordem de Foco</b>	<b>84</b>

# 1 INTRODUÇÃO

A acessibilidade digital é um princípio ético e legal fundamental para garantir a inclusão de pessoas com deficiência no uso de tecnologias. No Brasil, 18,6 milhões de pessoas vivem com algum tipo de deficiência, representando 8,9% da população. Desse percentual, 3,1% são pessoas de 2 anos ou mais de idade com dificuldade para enxergar, mesmo usando óculos ou lentes de contato. Esse percentual fica somente atrás dos 3,7% que representam pessoas com dificuldade para andar ou subir degraus ([IBGE, 2023](#)).

Uma pesquisa realizada pela [BigDataCorp \(2024\)](#) em parceria com o Movimento Web para Todos ([MWPT, 2024](#)), estampou que somente 2,9% dos sites brasileiros passaram em todos os testes de acessibilidade. Além disso, somente 21% das pessoas com deficiência acreditam que os sites e aplicativos atendem suas necessidades, refletindo uma falta significativa na inclusão digital no país ([Hand Talk, 2023](#)). Segundo [Hand Talk \(2023\)](#), 72% dos componentes de interface em sistemas vão em desencontro com as diretrizes do W3C que define como tornar conteúdos e interfaces webs dinâmicas acessíveis para pessoas com deficiência, para dinamicidade. Com base nisso, pode-se questionar, qual a dificuldade em manter sites e sistemas acessíveis para todos?

Outro fator que pode contribuir para a intensificação do problema é a possível ausência de ferramentas padronizadas destinadas à especificação de requisitos de acessibilidade durante a etapa de prototipação. Visto que, existe uma dificuldade de se encontrar pesquisas e ferramentas na área. Essa lacuna pode dificultar a comunicação entre designers e desenvolvedores, resultando em entregas imprecisas e aumentando a probabilidade de erros na implementação de funcionalidades acessíveis. O “Estudo sobre o Panorama da Acessibilidade Digital no Brasil” realizado pela [Hand Talk \(2023\)](#), diz que menos de 1% dos sites nacionais analisados possuem mecanismos básicos de acessibilidade, comprometendo a experiência de usuários que dependem de tecnologias assistivas e coloca em risco o cumprimento da legislação, como o Estatuto da Pessoa com Deficiência.

Neste contexto, UI Kits (*User Interface Kits*), são conjuntos prontos de elementos gráficos e componentes de interface que facilitam o design e o desenvolvimento visual de aplicações, sites e sistemas. Essas ferramentas são amplamente usadas por designers de interface e desenvolvedores de interface para acelerar o processo de criação de produtos digitais. O destaque para o problema da acessibilidade é que elas contêm documentações inclusivas, representando um avanço estratégico no desenvolvimento web. Segundo a *Interaction Design Foundation*, esses kits colaboram para a consistência, agilidade de prototipagem e alinhamento entre design e código-fonte da aplicação, incluindo elementos essenciais para uma interface acessível por incorporarem componentes pré-configurados com contraste adequado, navegação por teclado e rotulagem semântica (FOUNDATION, 2023). Esses kits servem como exemplos técnicos eficazes para designers e desenvolvedores (CONTRIBUTORS, 2024).

Assim, o presente trabalho apresenta o **Wilia**, um Kit de Interface de Usuário para documentar especificações de acessibilidade em projetos WEB. Com o foco nos designers, a ferramenta visa promover recursos que promovem protótipos com documentação clara e tecnicamente embasada, pensando nos requisitos de acessibilidade. O presente trabalho tem foco exclusivo na acessibilidade via leitores de tela. Ao traduzir normas técnicas em componentes práticos, o UI Kit proposto serve como ponte entre a teoria da acessibilidade e sua aplicação concreta, contribuindo para a redução da dependência de desenvolvedores em relação a documentações vagas e a garantir a conformidade legal e ética, melhorando a experiência de usuários com deficiência.

## 1.1 OBJETIVOS

### 1.1.1 OBJETIVO GERAL

Criar um UI Kit para Figma que permita documentar especificações de acessibilidade em protótipos, que busca mitigar erros de implementação semântica, descritiva e operacional.

### 1.1.2 OBJETIVOS ESPECÍFICOS

- Levantar e sintetizar as diretrizes de acessibilidade da WCAG (*Web Content Accessibility Guidelines*) e WAI-ARIA (*Web Accessibility Initiative – Accessible Rich Internet Applications*).
- Desenvolver um Kit de Interface de Usuário (UI Kit) na plataforma Figma.
- Mapear componentes para documentar acessibilidade para protótipos HTML.
- Criar material educativo: Guia prático para designers sobre a aplicação dos componentes e sua relação com requisitos técnicos.
- Proporcionar um método claro e integrado ao fluxo de design

## 1.2 ESTRUTURA DOS CAPÍTULOS

Este trabalho está estruturado em cinco capítulos. O presente capítulo, **Introdução**, apresenta o tema do estudo, define os objetivos gerais e específicos e descreve a organização do texto. O segundo capítulo, **Referencial Teórico**, discute trabalhos relacionados ao tema e apresenta com maior profundidade os temas que norteiam essa pesquisa como acessibilidade digital, tecnologias assistivas, WCAG e WAI-ARIA, além da legislação a respeito e a ferramenta Figma.

O terceiro capítulo, **Metodologia**, detalha todo o processo de construção da pesquisa, incluindo a etapa de prototipação e organização no Figma. O quarto capítulo, **Resultados**, apresenta os resultados obtidos, incluindo o mapeamento dos aspectos gráficos no Figma e criação do material educativo.

Por fim, o quinto capítulo, **Conclusão**, traz as considerações finais sobre a pesquisa desenvolvida, reflexões sobre os resultados e sugestões para trabalhos futuros.

## 2 REVISÃO DA LITERATURA

A revisão da literatura aborda os tópicos em que esse trabalho está baseado, como acessibilidade na web e exclusão digital, bem como as legislações e normalizações pertinentes ao tópico.

### 2.1 ACESSIBILIDADE NA WEB E EXCLUSÃO DIGITAL

Nos últimos anos, muito se tem discutido sobre inclusão digital (([NTT Data, 2022](#)), ([WEBAIM, 2023](#)), ([Associação Brasileira de Normas Técnicas, 2023](#))), mas ainda são poucas as iniciativas que realmente se preocupam com a experiência de quem enfrenta dificuldades para acessar o conteúdo online. A acessibilidade digital, nesse contexto, deixou de ser um detalhe técnico e passou a ser parte essencial de um debate mais amplo sobre direitos e igualdade. Conforme ([FILGUEIRAS, 2024](#)), a cidadania no século XXI exige, entre outras coisas, a possibilidade de se conectar, interagir e usufruir das plataformas digitais de maneira autônoma e segura. No entanto, para uma parcela significativa da população, como por exemplo pessoas com deficiência visual, o uso da internet continua com diversas barreiras e dificuldades ([NTT Data, 2022](#)). Isso mostra que, embora muitos serviços tenham migrado para o meio digital, nem todos puderam acompanhar esse movimento em condições de igualdade, tornando a exclusão digital uma realidade marcante para pessoas com deficiência e revelando um cenário desafiador em relação à acessibilidade na web.

Segundo dados do Instituto Brasileiro de Geografia e Estatística [IBGE \(2023\)](#), existem mais de 6,5 milhões de pessoas com deficiência visual no Brasil, sendo 500 mil cegas e cerca de 6 milhões com baixa visão. Esses dados demonstram a importância de considerar a acessibilidade digital como uma questão de política pública e de direitos humanos.

Entretanto, o cenário atual parece não levar esses dados em consideração, conforme relatório [WebAIM \(2023\)](#), que avaliou a acessibilidade de um milhão

de páginas da web e concluiu que 96,3% delas apresentavam falhas básicas no atendimento das diretrizes WCAG 2.1 (*Web Content Accessibility Guidelines*) .

No Brasil, um estudo feito em 2024 pela BigDataCorp em parceria com o Movimento Web para Todos (MWPT) analisou mais de 21 milhões de sites brasileiros e revelou que menos de 3% cumpriam de forma mínima os requisitos de acessibilidade digital, conforme ([FREIRE; FERRAZ, 2024](#)).

Outra pesquisa sobre o tema foi realizada pelo de [NTT Data \(2022\)](#) demonstra que a maior deficiência pelos respondentes da pesquisa é a cegueira, e com isso também indica que, entre os sites inacessíveis para pessoas com deficiência, os elementos mais encontrados nos sites são:

- Imagens CAPTCHA apresentando texto usado para verificar que você não é um robô;
- Imagens sem descrição ou com descrições inadequadas;
- Campo de busca inacessível ou inexistente;
- Formulários complexos;
- Links ou botões que não fazem sentido;
- Telas ou partes de telas que mudam inesperadamente;
- Falta de cabeçalhos;
- Falta de links “ir para conteúdo” ou “ir para o menu”;
- Muitos elementos focados com Tab na mesma página e tabelas complexas.

Todos esses itens demonstram, que, de certa forma, que muitos sites são criados muitas vezes visando sua identidade visual, e não na acessibilidade do mesmo, problemas esses que em geral, poderiam ser resolvidos pelas etiquetas Figma propostas no trabalho.

Essa falta de acessibilidade impacta diretamente a qualidade de vida das pessoas com deficiência, pois conforme estudo realizado por [WebAIM \(2023\)](#),

revelou-se que 71% dos usuários com deficiência abandonam sites em que enfrentam dificuldades de acesso, o que não apenas restringe seu acesso a informação como também compromete sua participação no mercado de trabalho, na educação e no consumo. Em um contexto onde serviços básicos como agendamento médico, matrícula escolar, serviços bancários e pagamentos de impostos são feitos hoje em dia online, a exclusão digital representa uma forma moderna de segregação.

### 2.1.1 LEGISLAÇÃO

Do ponto de vista legal, o Brasil possui legislação específica referente a acessibilidade digital na [BRASIL \(2015\)](#), ou seja, Lei Brasileira de Inclusão da Pessoa com Deficiência (Estatuto da Pessoa com Deficiência), que estabelece em seu artigo 63, que os sites mantidos por empresas com sede ou representação no país devem assegurar acessibilidade para pessoas com deficiência, em conformidade com os padrões estabelecidos para a web. Tem-se ainda o o Modelo de Acessibilidade em Governo Eletrônico (eMAG) que foi lançado em 2007, com alinhamento as diretrizes internacionais da W3C e que deveria ser adotado por todos os órgãos da administração pública federal [Digital \(2020\)](#).

O Marco Civil da Internet, através da Lei nº 12.965/2014, também estabelece direitos fundamentais relacionados ao acesso digital, embora não seja esse diretamente seu foco. Assim, em seu artigo 3 ele inclui o acesso universal à internet como princípio, o que pressupõe que o acesso seja possível para todos, o que incluindo pessoas com deficiência; em seu artigo 7 ele trata dos direitos dos usuários, destacando o direito à acessibilidade da informação, à neutralidade da rede e à inviolabilidade da privacidade, além de incentivar políticas públicas que promovam a inclusão digital, incluindo pessoas com deficiência ([Brasil, 2014](#)).

Outra legislação brasileira que promove a acessibilidade na web é a [Associação Brasileira de Normas Técnicas \(2023\)](#), uma norma brasileira publicada pela Associação Brasileira de Normas Técnicas (ABNT) que trata especificamente da acessibilidade na internet. Ela foi criada para estabelecer diretrizes e critérios técnicos com o objetivo de garantir que websites, portais, sistemas e conteúdos digitais sejam acessíveis a todas as pessoas, incluindo aquelas com deficiência. Como norte a mesma tem a base fundamentada nas WCAG 2.1, adaptadas para o contexto

brasileiro, e aplica-se a portais públicos e privados, aplicativos móveis, plataformas educacionais, e-commerce, entre outros. Além disso, a mesma aborda elementos como o texto alternativo para imagens, navegação via teclado, contraste de cores, fontes legíveis, uso de leitores de tela para navegação e formulários acessíveis [W3C \(2023\)](#).

## 2.2 DIRETRIZES E PADRÕES

A acessibilidade na web é fundamentada em diretrizes técnicas desenvolvidas por organizações internacionais, especialmente pelo W3C (*World Wide Web Consortium*). Entre essas, destacam-se as *Web Content Accessibility Guidelines* (WCAG) e as especificações do *Accessible Rich Internet Applications* (WAI-ARIA), ambas desenvolvidas pelo grupo *Web Accessibility Initiative* (WAI).

### 2.2.1 WCAG

Já no cenário internacional, as WCAG vêm sendo atualizadas de forma regular para englobar novos desafios, por exemplo, em sua versão 2.1 atualizada em Maio de 2025, adicionou-se critérios voltados a dispositivos móveis e à acessibilidade cognitiva [W3C \(2023\)](#). Apesar disso e da boa intenção dessas técnicas, a aplicação prática de fato dessas diretrizes ainda é limitada pela falta e pela dificuldade de fiscalização, desconhecimento técnico por parte dos desenvolvedores, pressão para desenvolvimento dos produtos de forma rápida, o que faz muitas vezes com que princípios básicos de acessibilidade sejam ignorados e ausência de políticas de capacitação.

As WCAG têm como objetivo tornar o conteúdo web mais acessível para pessoas com deficiências, incluindo deficiências visuais, auditivas, motoras e cognitivas. Atualmente ela está na versão 2.2 (2023) e sua evolução histórica pode ser visualizada na Figura 1.

Figura 1 – Evolução histórica do WCAG.



Figura 2 – Fonte: ([DOBBALA; LINGOLU, 2024](#))

Conforme ([DOBBALA; LINGOLU, 2024](#)), cada nova versão tem um foco e uma evolução da anterior, sendo que a versão 1.1 de 1999 tinha como objetivo as recomendações técnicas, porém era pouco flexível, o que trazia dificuldades na sua aplicação a novos formatos e tecnologias, mesmo assim, a versão 2.0 levou quase 10 anos para ser lançada, em 2008, e já trouxe uma abordagem mais tecnológica e independente de linguagem, permitindo sua aplicação em diferentes contextos como em HTML, PDF, aplicativo, entre outros. A versão de 2018, 2.1, introduziu novos critérios voltados à acessibilidade mobile, considerando deficiências cognitivas, baixa visão e acessibilidade por toque. Por fim, a versão 2.2 lançada em 2023 acrescentou novos critérios relacionados a controles visíveis, navegação com teclado e foco, buscando melhorar a experiência de navegação para usuários com dificuldades motoras e cognitivas ([W3C, 2023](#)).

Alguns autores, como ([DOBBALA; LINGOLU, 2024](#)) tem se debruçado sobre a análise da evolução das diretrizes de acessibilidade na web, e destacam a crescente importância da conformidade para a melhoria da experiência do usuário e para a criação de ambientes digitais mais inclusivos. Dessa forma, os autores analisam os contextos históricos dos principais marcos no desenvolvimento das normas de acessibilidade, e através disso, investigam os impactos da acessibilidade indo além do cumprimento legal, e ressaltam seu papel na construção de experiências digitais acessíveis e centradas no usuário, independente de qualquer deficiência que venham a ter.

Inclusive o trabalho de ([SHAH, 2023](#)) abordou o processo de atualização da versão WCAG 2.0 para WCAG 2.1, enfatizando a necessidade de manter-se atualizado com as novas versões das normas e com as novas tecnologias, como a mobile. Com base nisso a pesquisa propõe uma estratégia para atualização para a nova versão dividida em 4 etapas principais, sendo: avaliação, planejamento estratégico, implementação e testes, e com base nisso, as empresas conseguem mensurar de forma mais assertiva o esforço para ajustar seus sites/ sistemas. Ainda é ressaltado na pesquisa a importância das equipes interdisciplinares e do envolvimento de usuários no processo, bem como as barreiras enfrentadas na migração e os aprendizados obtidos, oferecendo uma visão prática e realista dos desafios.

Já o trabalho de ([ZDRAVKOVA; DALIPI; KRASNIQI, 2022](#)) avalia a acessibilidade em sistemas de gestão de aprendizagem, aplicativos de videoconferência com áudio e vídeo, e cursos online abertos massivos (MOOCs) com base em 4 deficiências, sendo: motora, visual, auditiva e cognitiva com base nas recomendações da WCAG 2.1. Tal avaliação foi feita durante o período do covid-19, em que muitas instituições de ensino substituíram o ensino presencial pelo virtual, mostrando a fragilidade que alunos com deficiência tiveram que enfrentar para continuar estudando.

Com base nos resultados da pesquisa, são propostas recomendações para tornar o aprendizado online mais acessível a estudantes com necessidades especiais, com o objetivo de garantir uma educação ampla para todos, sem discriminação por motivo de deficiência.

Apesar das novas versões terem trazido conteúdo com aplicação em outros contextos, os princípios do WCAG continuam os mesmos que em sua primeira versão, segundo ([W3C, 2025b](#)), conforme a sigla POUR (*Perceivable - percepção, Operable - operável, Understandable - compreensível, Robust - robusto*).

Quanto a Percepção, tem-se que a informação e os componentes da interface devem ser apresentados de forma que possam ser percebidos por todos os usuários, independentemente de qualquer deficiência, o que inclui diferentes formas de navegação e acesso ao conteúdo, por exemplo. Relacionado ao princípio

Operável, seu objetivo é que os componentes da interface e a navegação devem estar disponíveis e funcionais por meio de diferentes formas de interação, como teclado ou comandos de voz. O princípio Compreensível dita que o conteúdo deve ser claro e previsível, permitindo fácil entendimento, independente do nível de conhecimento do usuário quando ao uso daquela ferramenta, e por fim, o item Robusto preza pela compatibilidade do conteúdo entre diferentes agentes de usuário, o que engloba as tecnologias assistivas.

Além disso, o WCAG trabalha com 3 níveis de conformidade, A, AA e AAA, e elas são utilizadas para mensurar o quanto um site atende aos requisitos de acessibilidade. Cada nível representa um conjunto de critérios de sucesso com exigências crescentes ([W3C, 2023](#)).

O nível A é considerado o mínimo básico de conformidade, atendendo aos requisitos básicos de acessibilidade, e para isso ele deve remover as barreiras mais graves que impedem o acesso a pessoas com deficiência, como por exemplo ter a opção de texto para os conteúdos visuais ou permitir a navegação do conteúdo via teclado.

O nível AA é o recomendado, pois fornece um melhor equilíbrio entre acessibilidade e viabilidade técnica e estética. Alguns exemplos incluem a possibilidade de redimensionar o conteúdo até 200% sem perda de conteúdo, as páginas terem títulos descritivos com a marcação correta (h1, h2, etc) e um contraste mínimo de cor entre o texto e o fundo.

Por fim, o nível AAA é considerado avançado e como o maior nível de conformidade envolvendo a acessibilidade, exigindo o cumprimento de todos os requisitos anteriores e ainda mais requisitos como por exemplo aumentar o contraste entre texto e fundo para 4:1, oferecer a definição de palavras incomuns ou jargões, fornecer legendas para conteúdo de vídeo e também a interface adaptada para diversos estilos cognitivos e de aprendizagem. A Tabela 1 apresenta um resumo comparativo entre os níveis.

Tabela 1 – Resumo comparativo dos níveis de conformidade da WCAG

Nível	Foco Principal	Aplicabilidade
A	Acessibilidade básica	Remove as barreiras críticas
AA	Acessibilidade robusta	Recomendado para a maioria dos sites
AAA	Acessibilidade avançada	Ideal, mas nem sempre viável ou exigido

**Fonte:** Do Autor, 2025.

## 2.2.2 WAI-ARIA

Já o WAI-ARIA, por sua vez, fornece um conjunto de atributos que podem ser adicionados ao HTML para descrever melhor os elementos da interface para tecnologias assistivas, como leitores de tela. Esses atributos são especialmente importantes para tornar aplicativos web dinâmicos mais compreensíveis por usuários com deficiência ([WAI-ARIA Authoring Practices Task Force, 2023](#)).

Importante diferenciar primeiramente a semântica nativa da WAI-ARIA, assim sendo, a semântica nativa refere-se ao uso apropriado dos elementos HTML que já possuem significado implícito e comportamento esperado pelos navegadores e tecnologias assistivas, como por exemplo elementos como botões, formulários, links, entre outros, e esses são automaticamente reconhecidos por leitores de tela, permitindo que usuários com deficiência naveguem e interajam com uma página sem necessidade de configurações adicionais. Esses elementos, quando utilizados corretamente, promovem acessibilidade de forma confiável, padronizada e eficiente ([Mozilla Developer Network, 2025](#)).

Por outro lado, a WAI-ARIA, como mencionado, é uma especificação do W3C que fornece atributos adicionais para melhorar a acessibilidade de interfaces ricas criadas com JavaScript e HTML dinâmico, permitindo a definição de papéis chamados de *roles*, além de estados e propriedades que descrevem o comportamento de componentes de interface que não possuem um equivalente semântico direto em HTML, como menus personalizados, carrossel de imagens, abas ou diálogos modais, conforme ([W3C, 2025a](#)).

Apesar da utilidade do ARIA em casos específicos, sua aplicação deve ser feita com cuidado, pois uma das diretrizes mais importantes da acessibilidade

moderna menciona o seguinte: "*First Rule of ARIA: Don't use ARIA*" ([WAI-ARIA Authoring Practices Task Force, 2023](#)). Assim, a regra afirma que sempre que existir um elemento HTML com semântica nativa apropriada, ele deve ser preferido em vez do uso de ARIA, pois os elementos nativos são automaticamente compreendidos por navegadores, leitores de tela e demais tecnologias assistivas, enquanto o ARIA depende de implementação correta, o que frequentemente não acontece.

Um exemplo do uso incorreto de ARIA seria por exemplo aplicar a tag de botão `role="button"` a uma `<div>` sem suporte ao teclado, o que poderia comprometer seriamente a acessibilidade da interface. Com base nisso, o uso do ARIA deve ser reservado apenas para situações em que a semântica nativa não é suficiente ou não existe, como no desenvolvimento de *widgets* customizados, e nesse caso, seria necessário o uso dos atributos ARIA com precisão técnica, seguindo as especificações do W3C e sendo testados com tecnologias assistivas reais, como por exemplo em leitores de tela.

Exemplos de como o uso incorreto do padrão podem impactar negativamente na navegação é apresentado por ([PARK; LIM, 2017](#)), em que os autores avaliaram a acessibilidade de conteúdos em 50 sites da Coreia do Sul e 50 sites internacionais, e como resultados obtiveram que muitos sites não utilizam o WAI-ARIA ou o fazem de forma incorreta. No entanto, na Coreia do Sul, observou-se um aumento gradual no uso correto da especificação.

A [Tabela 2](#) apresenta um resumo da diferença de quando usar a web semântica e ARIA com exemplos.

Tabela 2 – Comparação entre Semântica Nativa e WAI-ARIA

Aspecto	Semântica (HTML)	Nativa	WAI-ARIA
<b>Definição</b>	Elementos HTML com significado e comportamento acessível embutidos.	Atributos que descrevem papéis, estados e propriedades para componentes personalizados.	
<b>Compatibilidade</b>	Altamente compatível com navegadores e leitores de tela.	Depende de implementação correta e testes com tecnologias assistivas.	
<b>Facilidade de uso</b>	Simples: comportamento acessível já incluso.	Complexo: requer conhecimento técnico e implementação adicional.	
<b>Necessidade de script</b>	Funciona nativamente, sem necessidade de JavaScript.	Geralmente precisa ser combinado com JavaScript para funcionar corretamente.	
<b>Quando usar</b>	Sempre que possível.	Somente quando não há equivalente nativo.	
<b>Exemplo</b>	<button> (já acessível e com suporte ao teclado).	<div role="button"> (precisa de ARIA + JS + eventos de teclado).	

**Fonte:** Do Autor, 2025.

Por fim, tem-se que a semântica nativa deve ser a base da acessibilidade web e o uso da WAI-ARIA deve ser tratada como uma extensão, tendo seu uso somente quando preciso, pois seu uso inadequado pode tornar a aplicação menos acessível.

### 2.2.3 ATAG

A ATAG (Accessibility Authoring Tools Guidelines), conforme ([W3C Web Accessibility Initiative, 2024](#)) é uma diretriz desenvolvida pelo W3C, cujo objetivo é garantir que as ferramentas como editores de sites, sistemas de gerenciamento de conteúdo (CMS) ou construtores de interfaces sejam acessíveis tanto para todos os perfis de usuários. Atualmente está na versão 2.0, tendo como base os princípios da versão 1.0, que são tornar as ferramentas de autoria acessíveis e ajudar os autores

a criarem um conteúdo acessível.

Para tornar as ferramentas acessíveis busca-se que elas também sejam utilizadas por pessoas com deficiência, e quanto ao auxílio a criação de conteúdo acessível, o mesmo tem o foco orientativo, ou seja, facilitar a criação de produtos digitais acessíveis por padrão, através de sugestões automáticas, validações, modelos inclusivos e boas práticas embutidas nos kits de design ([W3C Web Accessibility Initiative, 2024](#)).

Com relação a versão 2.0 lançada em 2015, manteve-se todas as diretrizes da versão anterior, adicionando os níveis de conformidade A, AA e AAA propostos no modelo WCAG, além de uma maior integração com esse modelo, de forma a promover um conteúdo com maior acessibilidade por padrão ([RICHARDS; SPELLMAN; TREVIRANUS, 2015](#)).

O trabalho de ([BALDIRIS et al., 2022](#)) por exemplo, fez a avaliação de quatro ferramentas gratuitas e de código aberto, de autoria para criação de conteúdos educacionais, com base nas diretrizes da ATAG, avaliando a acessibilidade do conteúdo educacional gerado por essas ferramentas, com base nas recomendações da WCAG. Como conclusão estabeleceram uma série de recomendações com o objetivo de ajudar a reduzir algumas lacunas relacionadas à acessibilidade. . Assim, através da utilização dos kits que seguem a ATAG garante-se que o conteúdo acessível não seja algo que deixe para ser feito posteriormente ao desenvolvimento padrão, mas sim que seja algo feito de forma integrada, evitando-se o retrabalho e também a adequação às conformidades legais, além de também ampliar o alcance das ferramentas.

A tabela compara três normativas centrais para a acessibilidade digital no Brasil: WCAG, Marco Civil da Internet e ABNT NBR 17225:2023. A WCAG, de origem internacional, serve como base técnica para acessibilidade web, sendo amplamente adotada, embora não obrigatória. O Marco Civil, lei nacional, garante o acesso universal à internet e fundamenta juridicamente a inclusão digital. Já a ABNT NBR 17225:2023 traduz essas diretrizes e princípios legais em requisitos técnicos aplicáveis a sites e aplicativos no contexto brasileiro. Juntas, essas normativas se complementam ao combinar diretrizes técnicas, garantias legais e orientações

Tabela 3 – Comparação entre WCAG, Marco Civil da Internet e ABNT NBR

Critério	WCAG	Marco Civil da Internet	ABNT NBR 17225:2023
Origem	Internacional (W3C)	Nacional (Brasil)	Nacional (Brasil)
Tipo	Diretriz técnica internacional	Lei geral sobre uso da internet	Norma técnica (ABNT) e legislação de inclusão
Foco principal	Acessibilidade de conteúdo web (HTML, interfaces, apps)	Direitos digitais, privacidade e acesso universal à internet	Acessibilidade digital para pessoas com deficiência
Aplicação	Referência global para acessibilidade web	Garantia de acesso universal e equitativo à internet	Aplicação técnica para sites, apps e portais públicos e privados
Obrigatoriedade legal	Não obrigatória, mas recomendada e usada como base	Obrigatória para provedores e governo	Obrigatória quando citada por leis ou exigida em contratos públicos
Relacionamento entre si	Base para a ABNT NBR 17225:2023	Prevê o acesso universal, apoiando a inclusão digital	A NBR 17225 detalha tecnicamente as exigências legais da LBI
Direitos abordados	Acesso a conteúdos digitais por pessoas com deficiência	Privacidade, neutralidade da rede, liberdade de expressão	Acesso igualitário, não discriminação, adaptação de plataformas digitais

**Fonte:** Do Autor, 2025

práticas para promover a acessibilidade digital.

## 2.3 LEITORES DE TELA E SUA INTERPRETAÇÃO DE INTERFACES

De acordo com Fernandes, Ferreira e Nunes (2023), os leitores de tela são usados majoritariamente por pessoas com maiores graus de deficiência visual,

sendo muitas vezes a principal forma de acesso aos conteúdos digitais, e sua função é basicamente ler ou traduzir o conteúdo que está na interface para uma saída em áudio ou linhas braille. Alguns dos modelos mais conhecidos são o NVDA (*NonVisual Desktop Access*), o JAWS (*Job Access With Speech*) e o VoiceOver que está presente nos dispositivos Apple.

Dessa forma, e conforme Matuzović (2024), os leitores de tela trabalham com o conceito de árvores de acessibilidade, que é uma estrutura paralela à árvore DOM (*Document Object Model*), que é gerada pelos navegadores e representa os elementos de uma página da web com as informações relevantes para tecnologias assistivas. Porém, enquanto a árvore DOM contém todos os elementos HTML, a árvore de acessibilidade filtra e transforma apenas os elementos visíveis e semanticamente relevantes, incluindo informações como labels, texto alternativo nas imagens, função dos elementos (botão, imagem, título, link), estado atual (ativo, selecionado, expandido), rótulos em campos de formulário, entre outros.

A geração da árvore de acessibilidade é baseada na semântica nativa do HTML, atributos ARIA e estilos CSS que muitas vezes afetam a visibilidade dos elementos na tela, e nisso começa o problema do uso dos leitores de tela, visto que eles fazem a análise da estrutura do código HTML e transmitem ao usuário, de forma linear, as informações contidas na página, assim, o correto funcionamento desses leitores depende de forma significativa como o código HTML foi criado e se ele possui os elementos que permitem sua leitura. Um exemplo claro é o uso da tag alt dentro das imagens, que faz a descrição visual do conteúdo da imagem, e que só é utilizado por esses leitores de tela e também caso a imagem não tenha sido carregada por algum problema de conexão ou por não existir.

Outros elementos como `<header>`, `<nav>`, `<main>`, `<article>` e `<footer>` informam ao leitor de tela qual a função daquela parte da página, permitindo que o usuário navegue por seções com maior fluidez e entendimento, visto que essas ferramentas funcionam de forma linear, o que também pode ser um tanto quanto demorado para ler páginas grandes ou com muito conteúdo, como publicidades em geral. Da mesma forma, utilizar `<button>` no lugar de `<div onclick="...">` ou `<label>` vinculado a campos de formulário são práticas que ajudam o leitor de tela a identificar corretamente ações e entradas de dados.

O trabalho de Fernandes, Ferreira e Nunes (2023) e Web Accessibility Initiative (WAI) (2017) apresenta uma série de dicas e boas práticas para a correta interpretação do código por leitores de tela, sendo as principais citadas abaixo, e além disso, a Tabela 4 apresenta o equivalente em HTML considerando essas boas práticas de desenvolvimento para o uso de leitores de tela.

- Evite links genéricos como “clique aqui”. Use descrições completas e, quando necessário, adicione o atributo `title` para fornecer informações complementares.
- Não faça uso de tabelas com células mescladas.
- Evite usar imagens para representar textos que poderiam ser escritos.
- Gerencie corretamente o foco com `tabindex`, permitindo navegação fluida por teclado. Use `tabindex=-1` para elementos que devem receber foco programaticamente, mas não na ordem natural de tabulação.
- Em apresentações (PowerPoint, por exemplo), organize a ordem de tabulação dos elementos, utilize fontes sem serifa de tamanho mínimo 32pt e mantenha bom contraste entre texto e fundo.
- O texto deve ser alinhado à esquerda (sem justificação) para facilitar a leitura por pessoas com deficiência visual.
- Hiperlinks devem ser visualmente distintos do texto comum e apresentar indicadores visuais ao foco (por mouse ou teclado).
- Títulos de páginas devem ser curtos, objetivos e iniciar com as informações mais importantes, bem como utilizar corretamente os estilos de títulos (h5, h4, etc) para que leitores de tela reconheçam a hierarquia da informação.
- Estruture o conteúdo com títulos e subtítulos marcados com os estilos nativos do editor (ex.: Título 1, Título 2).
- Evite espaços em branco excessivos para simular mudança de página. Use comandos de quebra de página apropriados (como `CTRL + Enter`).

- Insira links clicáveis no sumário de documentos digitais para facilitar a navegação.
- Sempre teste seus conteúdos com leitores de tela populares, como NVDA (Windows), VoiceOver (dispositivos Apple) ou Orca (Linux), para garantir que estão acessíveis na prática.

Por fim, os autores Fernandes, Ferreira e Nunes (2023) ainda sugerem que não se utilize apenas de um meio visual para passar uma informação, como no caso da imagem na Figura 3, em que se utiliza o elemento visual das cores, mas também o elemento textual dos números.

Figura 3 – Uso de mais um elemento visual para representar o conteúdo



Fonte: (FERNANDES; FERREIRA; NUNES, 2023)

Abaixo também apresenta-se a descrição de como ocorre a leitura dos elementos HTML, de forma geral, conforme (MATUZOVIć, 2024):

- Títulos (`<h1>`, `<h2>`): Os leitores de tela informam a hierarquia dos títulos e isso permite que os usuários tenham uma visão estruturada da página e possam navegar entre seções com atalhos.

Exemplo verbalizado:

*“Heading level 1: Bem-vindo ao site”*

- Imagens (`<img>`)

Se houver atributo alt:

“Image: logotipo da empresa”

Se alt estiver vazio (alt=):

*A imagem é ignorada*

Se alt estiver ausente:

*Leitores podem ler o nome do arquivo (o que não é recomendável).*

- Regiões e marcos (<nav>, <main>, <aside>): são elementos que permitem a navegação rápida por partes da página  
“Navigation region”, “Main content”, “Complementary content”

- Formulários sem rótulo/ label (<input> sem <label> ou aria-label): são usados para que os usuários entendam o propósito do campo  
“Edit text” (sem contexto)  
Com rótulo:  
“Search, edit text”

Ainda, é possível executar a leitura contínua, onde o leitor de tela percorre sequencialmente os elementos da árvore de acessibilidade, sendo similar à leitura linear de um documento, o que é ideal para leituras de textos longos.

Entretanto, a má utilização de ARIA ou a ausência de estrutura semântica adequada pode causar várias consequências, como ao invés de facilitar a leitura, pode gerar confusão e fazer com que elementos "invisíveis" não sejam encontrados, ou até mesmo fazer com que menus sejam lidos simplesmente como blocos de texto, sem as opções de link e navegação que um menu deveria possuir.

Devido a esses problemas de sua má utilização, um de seus princípios básicos é “não use ARIA se puder usar HTML nativo”, pois os elementos semânticos da linguagem já trazem embutidos os comportamentos e a acessibilidade esperados pela maioria dos leitores de tela (WAI-ARIA Authoring Practices Task Force, 2023).

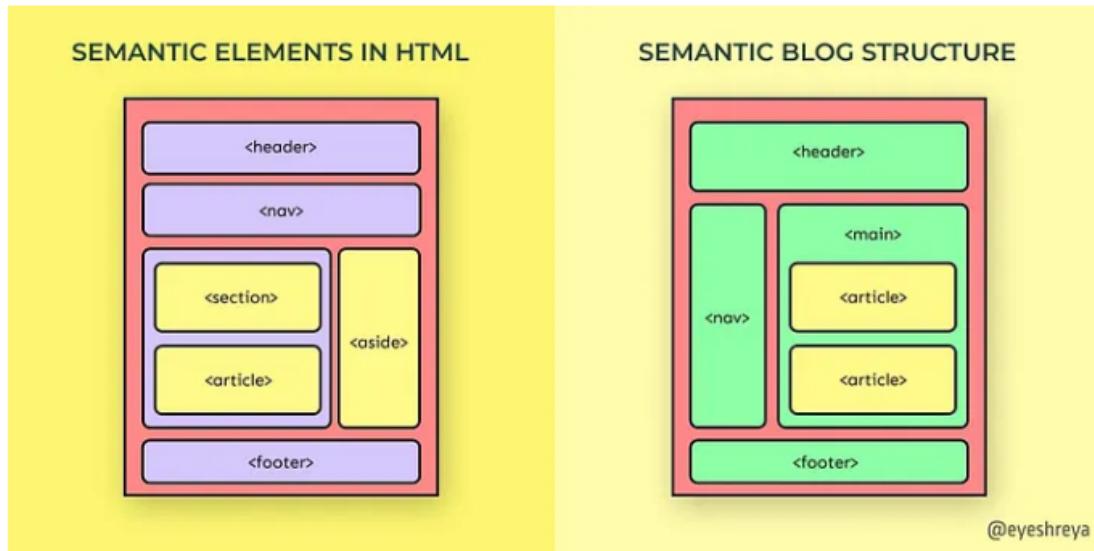
Tabela 4 – Exemplos de boas práticas para leitores de tela

Código comum em HTML	Melhoria com acessibilidade
<div onclick="enviar()> Enviar </div>	<button onclick="enviar()> Enviar </button>  Uso do elemento nativo <button> que permite foco por teclado e melhor interpretação por leitores de tela.
	  O atributo alt fornece descrição textual da imagem, essencial para leitores de tela e para quando a imagem, por algum motivo, não puder ser carregada.
<div>Menu principal</div>	<nav aria-label="Menu principal">...</nav>  Uso do elemento semântico <nav> com atributo aria-label para melhorar a navegação e semântica da página.
<span>Erro!</span>	<div role="alert">Erro ao enviar o formulário.</div>  O atributo role="alert" faz com que o leitor de tela anuncie automaticamente mensagens importantes.
<ul><li>...</li></ul> (sem título)	<h2 id="produtos">Produtos</h2>  <ul aria-labelledby="produtos">...</ul>  O atributo aria-labelledby associa o título à lista, facilitando a contextualização para leitores de tela.
<a href="doc.pdf">Clique aqui</a>  <div>Rodapé</div>	<a href="doc.pdf" title="Baixar regulamento em PDF">Regulamento (PDF)</a>  O atributo title torna o propósito do link mais claro.  <footer>Rodapé</footer>  O uso de <footer> identifica semanticamente a seção de rodapé da página.
<input type="text">	<label for="nome">Nome:</label> <input id="nome" type="text">  O uso de <label> vinculado ao input permite que o leitor de tela anuncie o campo corretamente.
<i class="fa fa-star"></i> (ícone decorativo)	<i class="fa fa-star" aria-hidden="true"></i>  Ícones puramente decorativos devem ser ocultos dos leitores com aria-hidden="true".
<div tabindex="0">...</div> sem função clara	<section tabindex=-1>...</section>  O uso adequado de tabindex permite navegação controlada por teclado, quando necessário.

Fonte: Do Autor, 2025.

Além disso, os leitores de tela percorrem o conteúdo de maneira sequencial, o que significa que a ordem dos elementos no código influencia diretamente na experiência do usuário, o que reforça a importância de se planejar a estrutura do HTML como uma narrativa acessível, onde títulos, listas, botões e formulários seguem uma ordem lógica e consistente, tal como demonstrado na Figura 4, onde antes mesmo de visualizar o resultado em tela, faz-se ideia de o que significa, aonde estará posicionado e o objetivo do trecho.

Figura 4 – Comparação de tags semânticas em código HTML



Fonte: ([ABNOVATO, 2021](#))

Por fim, a pesquisa de [NTT Data \(2022\)](#), realizada pela NTT Data em 2022 fez um levantamento dos principais leitores de tela utilizados no Brasil, obtendo 564 respostas válidas, sendo que a maior parte dos usuários de desktop utilizam o leitor de tela NVDA com Chrome ou NVDA com Firefox, e quanto aos mobile, os leitores mais utilizados são TalkBack com Chrome, VoiceOver com Safari e JieShuo (chinês) com Chrome. A pesquisa ainda fez o levantamento demográfico por região, sexo, nível de escolaridade e faixa etária, além de como o leitor de tela é de fato utilizado (navegação por região da página, links, cabeçalhos, entre outros) e cruzamento das informações, permitindo um panorama geral do uso desses leitores no Brasil.

De forma complementar, a Tabela 5 faz um resumo dos principais leitores de tela, sendo NVDA, JAWS e TalkBack.

Tabela 5 – Comparaçāo entre leitores de tela populares

Característica	NVDA	JAWS	TalkBack	VoiceOver
Plataforma	Windows	Windows	Android	macOS / iOS
Custo	Gratuito	Pago (licença)	Gratuito	Gratuito
Voz padrão	<i>Microsoft Speech Platform</i>	Várias opções, voz humana	Google TTS	Siri
Personalização	Alta	Muito alta	Moderada	Alta
Suporte a idiomas	Multilíngue	Multilíngue	Multilíngue	Multilíngue
Popularidade	Muito usado	Muito usado (profissional)	Muito usado (mobile)	Muito usado (usuários Apple)

Fonte: Do Autor, 2025.

## 2.4 DESIGN DE INTERFACE E DOCUMENTAÇÃO NO FIGMA

Segundo [Garrett \(2011\)](#) o design de interfaces é um componente essencial no desenvolvimento de produtos e serviços digitais que atendam às necessidades dos usuários de forma eficiente e inclusiva. Nos últimos anos, o Figma destacou-se como líder design de interfaces e prototipação colaborativa, com destaque pela sua plataforma baseada em nuvem que permite múltiplos usuários trabalharem de forma conjunta e colaborativa em um projeto ([IBRAHIM et al., 2023](#)). Essa característica revolucionou o processo de criação, facilitando a comunicação entre equipes multidisciplinares e acelerando o ciclo de desenvolvimento.

No contexto de *handoff*, o Figma facilita a entrega técnica por meio da exposição de propriedades CSS, medidas, cores e exportação de ativos. Por outro lado, esse processo apresenta limitações importantes quando se trata de acessibilidade.

Assim, uma das funcionalidades utilizadas do Figma é o suporte ao uso de *UI Kits*, ou seja, coleções de componentes visuais padronizados, como botões, formulários, ícones e layouts, que garantem consistência e coerência no design ([RYHUS, 2024](#)). A utilização desses kits permite uma maior produtividade e a padronização visual independente da plataforma utilizada, o que contribui para que as interfaces sejam reconhecíveis e intuitivas para os usuários, e acordo com

Garrett (2011). Além disso, eles tornam o desenvolvimento mais produtivo visto que não é necessário retrabalho na manutenção e atualização dos projetos.

Apesar das vantagens mencionadas, o Figma também enfrenta desafios e possui limitações quanto a usabilidade, pois embora existam diversos *plugins* que ajudam na análise do contraste de cores ou na validação da tabulação através do teclado, por exemplo, elas ainda não estão completamente integradas ao fluxo natural de uso da ferramenta, de acordo com Oliveira e Monteiro (2023). Por causa disso, algumas atualizações na documentação dos requisitos de acessibilidade muitas vezes precisam ser feitas manualmente, o que pode contribuir para que fiquem desatualizadas ou inconsistentes.

Além disso, Lindholm (2023) informa sobre a ausência de suporte nativo a práticas inclusivas. A ferramenta também não realiza validações automáticas de acessibilidade, como verificação de contraste ou estrutura de navegação por teclado, o que exige atenção redobrada dos designers. Além disso, é comum a ausência de rótulos textuais adequados em botões e elementos interativos, o que compromete o uso por leitores de tela, especialmente em fluxos onde não há documentação paralela (LINDHOLM, 2023).

Apesar disso, os pesquisadores Kokate, Shinohara e Tigwell (2022) realizaram a avaliação de 10 ferramentas digitais de prototipação, estando o Figma incluso, de forma a entender os recursos de acessibilidade que disponibilizavam e descobriu-se que a acessibilidade dava-se basicamente através de *plugins* terceiros, demonstrando o potencial de melhora nas mesmas.

Nesse cenário, os UI Kits desempenham um papel importante, pois através dos componentes já padronizados mantêm-se a consistência visual e os erros durante a prototipação são reduzidos. Mais importante ainda, kits bem projetados podem incorporar práticas acessíveis desde sua concepção, promovendo decisões mais responsáveis e inclusivas (LINDHOLM, 2023). No entanto, a eficácia desses kits depende diretamente de sua correta adoção e da clareza de suas documentações.

Já a ausência de documentação de acessibilidade no *handoff* pode gerar falhas críticas, onde os desenvolvedores podem frequentemente se deparar com componentes sem rótulos acessíveis, hierarquias de foco desorganizadas e estruturas

não semânticas, o que resulta em interfaces mal adaptadas para usuários com deficiências ([LINDHOLM, 2023](#)). Com base nisso, a inserção de anotações e guias visuais dentro do próprio arquivo Figma, ou o uso de kits com diretrizes explícitas, torna-se fundamental para garantir a continuidade da intenção de design ao longo do ciclo de desenvolvimento.

## 2.5 USO DE KITS E ANOTAÇÕES VISUAIS PARA ACESSIBILIDADE

Segundo [Cooper et al. \(2014\)](#), a acessibilidade digital apresentou avanço com as diretrizes propostas pelo WCAG e W3C, porém, ainda assim, sua completa e eficaz aplicação apresenta desafios e barreiras, principalmente na etapa da prototipação e do design das interfaces. Nesse cenário, os kits e anotações visuais aparecem como soluções eficazes para integrar práticas acessíveis ao fluxo de trabalho cotidiano de design, reforçando também a acessibilidade como uma etapa do projeto.

Os kits de anotação de acessibilidade (*A11y Annotation Kits*), amplamente difundidos na comunidade Figma, representam uma ferramenta crucial para integrar especificações de acessibilidade diretamente nos artefatos de design, como *wireframes* e protótipos. Esses kits funcionam como uma camada de documentação visual, onde anotações indicam atributos essenciais para a experiência de usuários de tecnologias assistivas, tais como a ordem de foco do teclado, a hierarquia de leitura, o contraste de cores e o uso de atributos ARIA (*Accessible Rich Internet Applications*). Ao tornar essas especificações explícitas na fase de design, os kits não apenas facilitam a comunicação entre designers e desenvolvedores, mas também servem como um recurso de aprendizado para equipes multidisciplinares. Essa necessidade é corroborada pela pesquisa da [Hand Talk \(2023\)](#), que aponta a falta de experiência prévia em acessibilidade como um desafio recorrente entre os profissionais da área.

A descoberta de tais recursos na comunidade Figma é facilitada por buscas com termos como "A11y", "Accessibility" ou "Accessibility kit". Entre os mais pro-

minentes, destacam-se o *A11y Annotation Kit Indeed Design* e *Stephanie Hagadorn (2020)*, focado na clareza da comunicação técnica; o *Web Accessibility Annotation Kit CVS Health (2023)*, orientado à estrutura macro da página; o *Intopia's Accessibility Annotation Kit Intopia (2023)*, que se aprofunda na conformidade com as diretrizes WCAG; e o *Pencil A11Y Kit Brainly (2022)*, que adota uma abordagem visual distinta para fases iniciais de ideação. Esses kits, embora com abordagens diferentes, compartilham o objetivo de traduzir os requisitos abstratos de acessibilidade em especificações concretas, como demonstrado na [Tabela 6](#) e nas [Figura 5](#) e [Figura 6](#), que ilustram a utilidade de alguns desses kits.

Tabela 6 – Análise comparativa entre os principais kits de anotação de acessibilidade

Kit Figma	Escopo das Anotações	Público-Alvo Principal	Ponto Forte Estratégico
A11y Annotation Kit	Estrutura semântica de componentes (cabecalhos, botões, links) e fluxo de interação (ordenem de foco).	Designers e desenvolvedores Front-end.	<b>Simplicidade e Clareza:</b> Foco em traduzir visualmente a estrutura HTML e os atributos ARIA mais comuns, agilizando o <i>handoff</i> .
Web Accessibility Annotation Kit	Estrutura macro da página, incluindo <i>landmarks</i> (cabecalho, navegação, conteúdo principal), e hierarquia de leitura.	Designers, Arquitetos de Informação e Desenvolvedores.	<b>Visão Estrutural:</b> Garante que a navegação e a compreensão global da página sejam lógicas para usuários de leitores de tela.

*Continua na próxima página...*

Tabela 6 – (Continuação)

Kit Figma	Escopo das Anotações	Público-Alvo Principal	Ponto Forte Estratégico
Intopia's Accessibility Annotation Kit	Anotações detalhadas em nível de componente, com referências explícitas aos critérios de sucesso das WCAG.	Times de Produto, QAs e Desenvolvedores que necessitam de alta conformidade técnica.	<b>Rigor Técnico e Educacional:</b> Atua como um guia de conformidade, conectando o design às regras das WCAG e educando a equipe.
Pencil A11Y Kit	Anotações com estilo visual "desenhado à mão" ( <i>sketch</i> ), ideal para fases de ideação e <i>wireframing</i> de baixa fidelidade.	Designers de UX/UI em fases iniciais de projeto.	<b>Comunicação Rápida e Informal:</b> Reduz a formalidade das anotações, incentivando a inclusão de considerações de acessibilidade desde o brainstorming.

**Fonte:** Do Autor, 2025.

Figura 5 – Exemplo de aplicação do kit da CVS Health

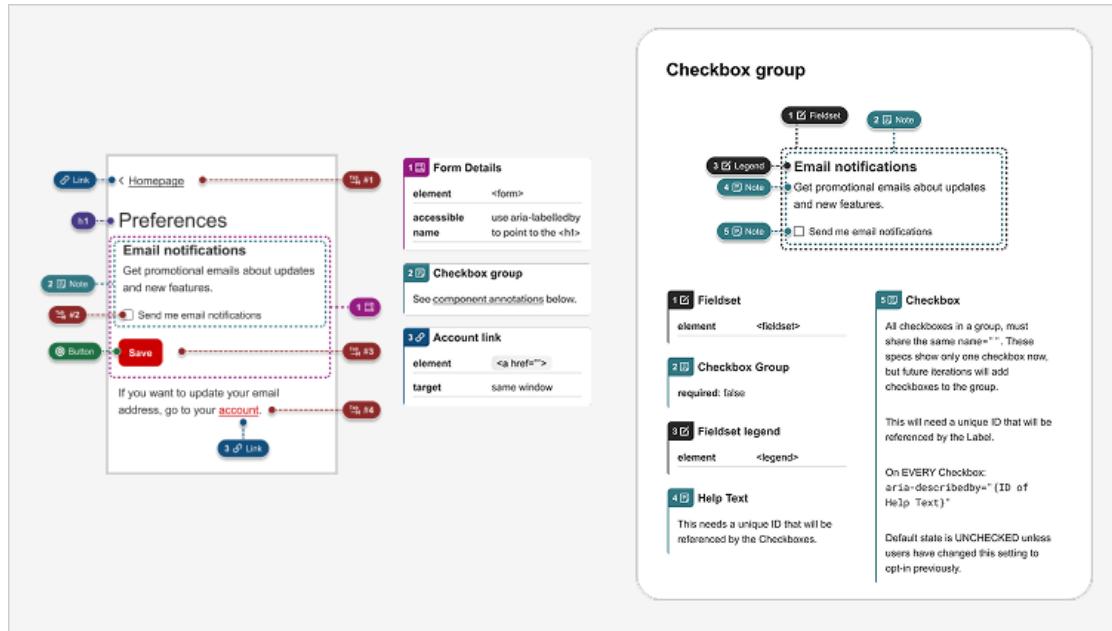
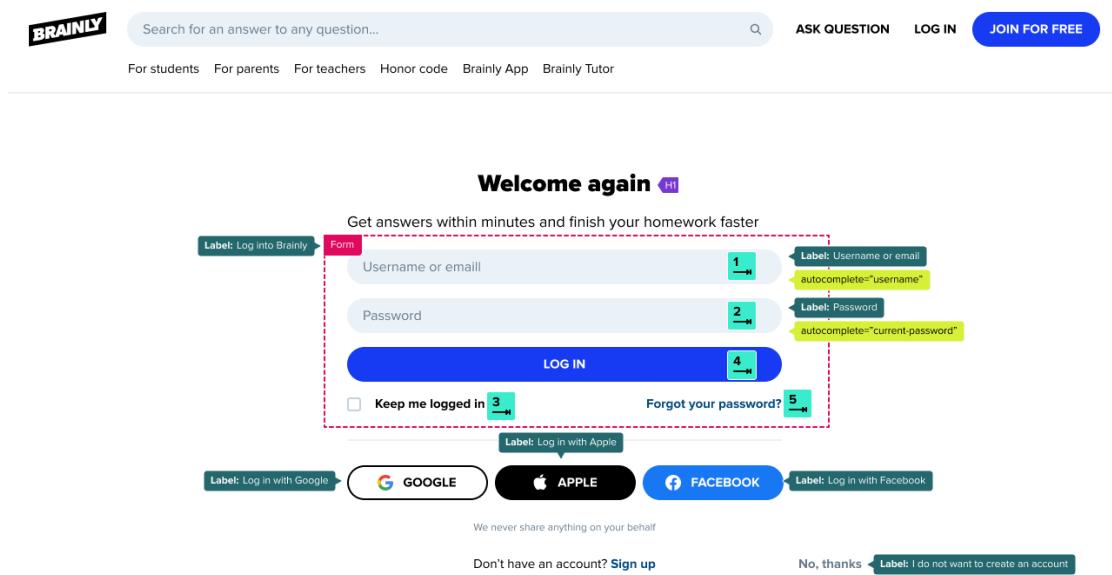
Fonte: [CVS Health \(2023\)](#)

Figura 6 – Exemplo de aplicação do Pencil A11Y Kit

Fonte: [Brainly \(2022\)](#)

De forma complementar, a Microsoft também tem investido no design inclusivo, através de seus *toolkits*, que são kits que orientam o designer na construção de soluções que considerem diferentes capacidades físicas, cognitivas e sensoriais (Microsoft, 2020; GILBERT, 2016).

Nesse teor o trabalho de Lindholm (2023) explora como um kit de anotações para acessibilidade pode apoiar designers e comunicar especificações de acessibilidade para desenvolvedores, indo além de apenas cumprir as diretrizes e legislações vigentes. Assim, utilizando-se métodos de pesquisa qualitativa e um processo de design centrado no usuário, por meio de atividades de design como entrevistas, ideação, prototipagem e testes de usabilidade, explora-se uma ferramenta de anotação de forma a avaliar como ela poderia contribuir para o suporte e a comunicação. Como conclusão, o projeto indica que os kits de anotações têm potencial para melhorar a comunicação e apoiar os designers em seu processo de criação.

Um outro fator de inclusão considerado pelos pesquisadores Viera et al. (2020) é a criação e modelagens de produtos que sejam verdadeira inclusivos, considerando questões como gênero, idade, valores éticos ou deficiências, propondo assim o desenvolvimento de um kit para tal.

Tem-se então que os benefícios desses kits não se restringem à etapa de design, mas se estendem até mesmo na fase de implementação, transformando-se em requisitos de acessibilidade que precisam ser de fato desenvolvidos e testados. Com base nisso, os kits atuam como facilitadores no processo de *handoff* entre design e desenvolvimento (BENNETT; ROSNER, 2019).

No presente contexto, os *handoff* podem ser entendidos como momentos de transição de responsabilidade ou de entrega de materiais entre diferentes equipes ou profissionais, como por exemplo da fase de design para a de desenvolvimento, ou do desenvolvimento para a etapa de testes, conforme Cooper et al. (2014).

Por outro lado, Bennett e Rosner (2019) menciona que os *handoffs* tem pouca documentação clara, sendo então propensos a erros e mal entendidos por parte da equipe, principalmente caso ela seja muito grande, podendo gerar também entregas incompletas ou ambíguas e uso de ferramentas incompatíveis entre os

times.

O estudo de [Bennett e Rosner \(2019\)](#) então aponta que as falhas de comunicação entre as equipes são uma das principais causas de não conformidade com critérios de acessibilidade, mesmo quando estes foram considerados no design inicial, e uma forma de melhorar os *handoffs* é justamente o uso de ferramentas como o Figma, Jira, Notion, Zeplin, entre outros, além de constantes reuniões de alinhamento, checklists para entrega e documentação enxuta e clara sobre os entregáveis esperados.

Nota-se que, conforme mencionado por ([BOTELHO, 2021](#)), os kits auxiliam em diversos aspectos da acessibilidade, porém ainda possuem limitações quanto aos aspectos subjetivos ou contextuais da acessibilidade, sendo necessário um esforço consciente e sistêmico para assegurar que o potencial das tecnologias digitais para a inclusão seja realizado, como por exemplo, *smartphones* podem ser incompatíveis com aparelhos auditivos necessários para pessoas surdas, telas sensíveis demais para quem tem deficiências motoras, e páginas web frequentemente carecem dos rótulos de texto necessários para softwares leitores de tela usados por pessoas cegas, e mesmo que cada um desses exemplos seja corrigido, a acessibilidade pode ser de curta duração se o processo de produção por trás desse hardware ou software não for ajustado, visto que o mundo digital evoluiu de forma ágil e constante. Assim, considerando o uso dos kits no início do projeto, busca-se atuar de madeira preventiva, reduzindo as correções e melhorando a qualidade final o produto, bem como reduzindo custos com retrabalho.

Além disso, ([ROSENFELD; MORVILLE; ARANGO, 2020](#)) menciona que esses kits também permite o envolvimento de *stakeholders* não técnicos no processo de validação e discussão a respeito da acessibilidade. Assim, a consideração de requisitos que afetam a interface diretamente torna-se mais fácil a compreensão do impacto do design sobre elas, o que é especialmente relevante em equipes grandes e multidisciplinares, em que mais áreas contribuem no projeto, como negócio, marketing ou o setor jurídico.

Outro aspecto importante é a ausência de uma cultura organizacional voltada à inclusão e acessibilidade, conforme pesquisa realizada por ([PARTHASASA-](#)

RATHY; JOSHI, 2023), que após estudo realizado com desenvolvedores web na Índia, descobriu-se que cerca de 70% deles nunca tinham recebido nenhum tipo de treinamento formal sobre acessibilidade, o que confirma que tal tópico muitas vezes é visto como secundário, e por falta de tempo na construção das ferramentas, é muitas vezes ignorado.

Designers e desenvolvedores de sistemas frequentemente enfrentam desafios para implementar requisitos de acessibilidade de forma clara e eficaz, especialmente quando há pouca experiência ou ferramentas específicas para auxiliar nesse processo Hand Talk (2023).

Finalmente, soluções como o Wilia destacam-se ao oferecer funcionalidades específicas aliadas a padronização visual, documentação acessível e facilidade de exportação de dados para plataformas de desenvolvimento. Com isso, essas ferramentas representam a convergência entre o design acessível e a engenharia prática, o que visa promover uma cultura de acessibilidade como prática contínua e não apenas como etapa final ou obrigatoriedade legal.

# 3 METODOLOGIA

O presente capítulo descreve o percurso metodológico empregue na conceção, desenvolvimento e estruturação do Wilia, um kit de anotação de acessibilidade para a ferramenta Figma. A pesquisa classifica-se como Aplicada, por se focar na criação de um artefacto destinado à solução de um problema prático no fluxo de trabalho de produtos digitais. A abordagem utilizada foi Qualitativa, centrada na análise de diretrizes, na construção de um kit de componentes de documentação e na produção de conteúdo técnico de apoio. O processo foi segmentado em quatro fases sequenciais.

## 3.1 FUNDAMENTAÇÃO TEÓRICA E DELIMITAÇÃO DO ESCOPO DE ATUAÇÃO

O ponto de partida da investigação foi a identificação de uma lacuna comunicacional e ferramental entre as disciplinas de design e desenvolvimento no que tange à especificação de requisitos de acessibilidade. Para contextualizar a relevância deste problema no cenário brasileiro, a etapa inicial incluiu a análise de pesquisas e relatórios de organizações nacionais proeminentes na área. Foram consultados os levantamentos periódicos realizados pelo [MWPT \(2024\)](#) (Movimento Web para Todos) em parceria com a BigData Corp, bem como a pesquisa “Panorama da Acessibilidade Digital no Brasil”, publicada pela [Hand Talk \(2023\)](#). Ambos os estudos serviram para evidenciar a dimensão e a persistência das barreiras digitais no ecossistema web nacional, reforçando a relevância e a urgência do problema abordado por este trabalho.

Esta análise de cenário nacional, somada ao estudo das fontes normativas globais, compôs o embasamento teórico do projeto. A investigação aprofundada foi direcionada por três pilares:

1. As Diretrizes de Acessibilidade para Conteúdo Web (WCAG), do World Wide

Web Consortium (W3C), que serviram como a base normativa para todos os critérios técnicos.

2. Os relatórios anuais da WebAIM (Web Accessibility in Mind), cuja análise dos erros mais comuns na web permitiu um diagnóstico preciso dos pontos de maior fragilidade em interfaces digitais.
3. A documentação técnica do MDN Web Docs (Mozilla) e as práticas de autoria do WAI-ARIA (W3C), que forneceram os subsídios para os exemplos de implementação e a tradução das diretrizes em código prático.

A síntese deste levantamento resultou na delimitação do escopo do artefato. Foram selecionados para compor o kit os componentes de anotação que correspondem às áreas de maior incidência de falhas de acessibilidade, a saber: Regiões (Landmarks), Cabeçalhos, Botões, Campos de Entrada, Imagens, Hiperlinks, Ordem de Leitura, Ordem de Foco, Anotação Geral e a instrução para Ignorar elementos decorativos.

### **3.2 ANÁLISE COMPARATIVA DE ARTEFATOS E MODELAGEM ESTRATÉGICA**

Uma vez definido o escopo, a segunda fase concentrou-se na análise de soluções correlatas por meio de um processo de benchmarking conduzido diretamente na plataforma Figma Community, sendo a biblioteca de conteúdos da comunidade. A pesquisa foi realizada utilizando termos como “accessibility”, “a11y”, “Acessibilidade” e “Acessibility kit”, o que levou à seleção de artefatos de alto impacto e ampla utilização pela comunidade de design, estabelecendo-os como referências para análise.

A seleção teve como critérios o quanto bem ranqueados os artefatos estavam nos resultados da busca, o número de duplicações, que remete ao número de usuários que utilizam ou utilizaram o artefato, o número de marcações como “gostei” que é uma métrica de dizer se quem o usou gostou ou não e, principalmente, se o artefato abrangia o material de estudo dessa pesquisa que trata sobre leitores de

tela e sistemas e sites web. Visto isso, foram selecionados os artefatos previamente citados:

- **A11y Annotation Kit:** Com uma Utilização pela comunidade de mais de 14900 suários e marcado como gostei por mais de 1100 usuários.
- **Web Accessibility Annotation Kit:** Com uma Utilização pela comunidade de mais de 9800 suários e marcado como gostei por mais de 1000 usuários.
- **Intopia's Accessibility Annotation Kit:** Com uma Utilização pela comunidade de mais de 2000 suários e marcado como gostei por mais de 135 usuários.
- **Pencil A11Y Kit:** Com uma Utilização pela comunidade de mais de 1200 suários e marcado como gostei por mais de 80 usuários.

### 3.3 CONCEPÇÃO DO KIT NO FIGMA

Esse último ponto consistiu na construção efetiva do projeto Wilia no Figma, concebido como uma solução de design final para documentação. A construção seguiu uma sequência rigorosa de etapas processuais: primeiramente, a definição da arquitetura visual e nomenclatura do kit; em seguida, a construção da base de componentes fundamentais; posteriormente, a estruturação destes elementos em anotações complexas; na sequência, o desenvolvimento técnico do “Multicomponente” centralizador; e, por fim, a curadoria e integração do conteúdo educacional na ferramenta.

O primeiro passo foi projetar a identidade visual do próprio kit de anotação, definindo a Arquitetura Visual e Nomenclatura. Foi desenvolvida uma paleta de cores, tipografia e iconografia distintas, para garantir que as anotações do Wilia fossem sempre legíveis e visualmente apartadas do design que estivessem documentando. Simultaneamente, foi estabelecida uma convenção de nomenclatura para todos os componentes e suas camadas, assegurando a escalabilidade e manutenibilidade do arquivo.

A construção da base de componentes foi guiada por um princípio de componentização modular. Esta etapa iniciou-se pela criação dos elementos mais fundamentais e indivisíveis. Estes elementos-base, como a Etiqueta (o rótulo visual), o contêiner de Nota (a área de texto) e os Conectores (as linhas de ligação), foram criados como componentes mestres no Figma.

Na etapa de estruturação das anotações, esses elementos-base foram então combinados para formar estruturas mais complexas e funcionais. O componente Nota, por exemplo, foi estruturado com a ferramenta Auto Layout do Figma para se adaptar dinamicamente ao conteúdo. Foram criados campos de texto específicos para “Título da anotação”, “Descrição da nota”, e “Código de exemplo”, conforme a anatomia do kit. Esta estruturação garantiu que cada anotação fosse um conjunto coeso e responsivo.

O desenvolvimento do “Multicomponente” com Component Properties foi a solução técnica implementada para otimizar a usabilidade do kit, conforme planejado na estratégia. Sua implementação técnica no Figma envolveu a criação de um componente base contendo todas as possíveis anotações; a configuração de uma propriedade de variante chamada “Tipo de Anotação”, com opções para cada um dos dez tipos de especificação (Cabeçalho, Botão, Imagem, etc.); e o uso de propriedades de texto e booleanas (Component Properties) para permitir ao usuário final a edição de textos e a alternância da visibilidade de seções diretamente no painel de inspeção do Figma.

Finalmente, a etapa de curadoria e integração do conteúdo educacional foi o que efetivamente transformou a coleção de elementos visuais em um sistema de conhecimento integrado. Com a estrutura técnica finalizada, foi realizado o processo de pesquisa, redação e integração do conteúdo de apoio. Para cada variante dentro do “Multicomponente”, o texto correspondente à sua documentação (Definição, Impacto no Usuário, Diret

## 4 RESULTADOS

O principal resultado deste trabalho é o *Wilia* (acrônimo para Web Accessibility Annotation Kit), um sistema de especificação visual desenvolvido na plataforma Figma. O *Wilia* está disponível publicamente em <<https://www.figma.com/design/2hN2ajzPoqDtcGcsAl0gxN/Wilia--Web-Accessibility-Annotation-Kit>> para consulta.

O nome **Wilia** foi cuidadosamente selecionado para refletir a missão central do artefato. Trata-se de um acrônimo para **Web Accessibility Annotation Kit**, destacando em sua própria nomenclatura a tríade de elementos que o compõem: a Web como ambiente de aplicação, a Acessibilidade (*Accessibility*) como valor fundamental e o Kit de Anotação como formato. Essa escolha nominal visa não apenas fornecer uma identidade única e memorável, mas também encapsular o propósito de que o kit atua como um sistema dedicado a traduzir diretrizes complexas de inclusão digital em ferramentas de design de uso prático e diário.

A concepção do kit está rigorosamente baseada nos padrões técnicos da WCAG 2.2 e WAI-ARIA, e enriquecida por boas práticas de mercado.

Sobre os artefatos selecionados como comparativo e modelagem que basearam o *Wilia*, foi realizada uma análise comparativa a respeito da estrutura, utilização e disponibilidade dos recursos. A análise revelou que, embora fossem funcionalmente competentes, apresentavam lacunas estratégicas:

**Documentação Dissociada:** A fundamentação teórica (o "porquê" de uma anotação) não estava muito bem difundida nos kits, exigindo que o usuário buscassem conhecimento em fontes externas e interrompesse seu fluxo de trabalho.

**Foco na Ferramenta, Não no Conhecimento:** Os kits forneciam o meio para rotular um elemento, mas não educavam o designer sobre o impacto daquela decisão na experiência de um usuário de tecnologia assistiva.

**Fluxo de Trabalho Fragmentado:** A aplicação das anotações frequentemente

exigia que o usuário procurasse e gerenciasse os componentes individuais dispersos na biblioteca, tornando o processo lento e suscetível a inconsistências.

Essas lacunas contribuíram para a modelagem estratégica do *Wilia*, possibilitando, como seu grande diferencial, a funcionalidade de atuar como uma ponte entre a teoria e a prática. O *Wilia* traduz normas de acessibilidade em componentes visuais e diretrizes acionáveis, criando uma ferramenta de aplicação direta e intuitiva que pode se integrar ao fluxo de trabalho de designers e desenvolvedores, capacitando-os a criar interfaces mais inclusivas desde a sua concepção.

A construção do *Wilia* passou por uma série de etapas interdependentes, sendo o primeiro resultado a elaboração de uma página introdutória no projeto no Figma. Dessa forma, na seção inicial apresentam-se os objetivos do kit, o público-alvo, e o contexto de sua criação, bem como links para as documentações e diretrizes mencionados acima. A introdução foi planejada como um diálogo com o usuário que irá usar o kit, buscando com isso repassar também o valor social da acessibilidade e o que motivou a criação do kit.

A página de *Visão geral*, conforme trecho recortado e demonstrado na [Figura 7](#), que oferece uma visão global de todos os componentes criados, permitindo otimizar o fluxo de trabalho ao possibilitar que o usuário visualize, selecione e compreenda rapidamente a função de cada item disponível. Essa centralização facilita tanto a navegação quanto o entendimento do conjunto como um sistema coeso. Além disso, a [Figura 8](#) demonstra a anatomia de cada componente, projetados em diferentes formatos, tamanhos e direções justamente para deixar a ferramenta flexível e com isso, permitir um maior uso por parte dos desenvolvedores e designers.

Figura 7 – Trecho da visão geral do Wilia.

### Regiões (Landmark)

Para demarcar as seções funcionais mais importantes de uma página, como header, main e nav, criando um mapa do layout.

[Leia mais sobre](#)



### Cabeçalho (Heading)

Para definir os níveis semânticos na hierarquia de títulos de uma página (h1-h6).

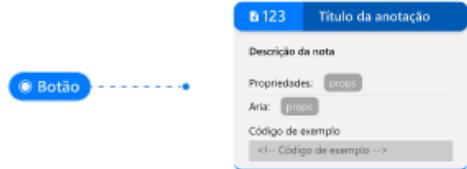
[Leia mais sobre](#)



### Botão (Button)

Para elementos interativos que executam uma ação, como enviar um formulário ou abrir um diálogo.

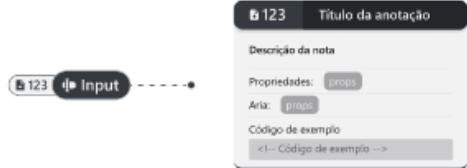
[Leia mais sobre](#)



### Campos de Entrada (Input)

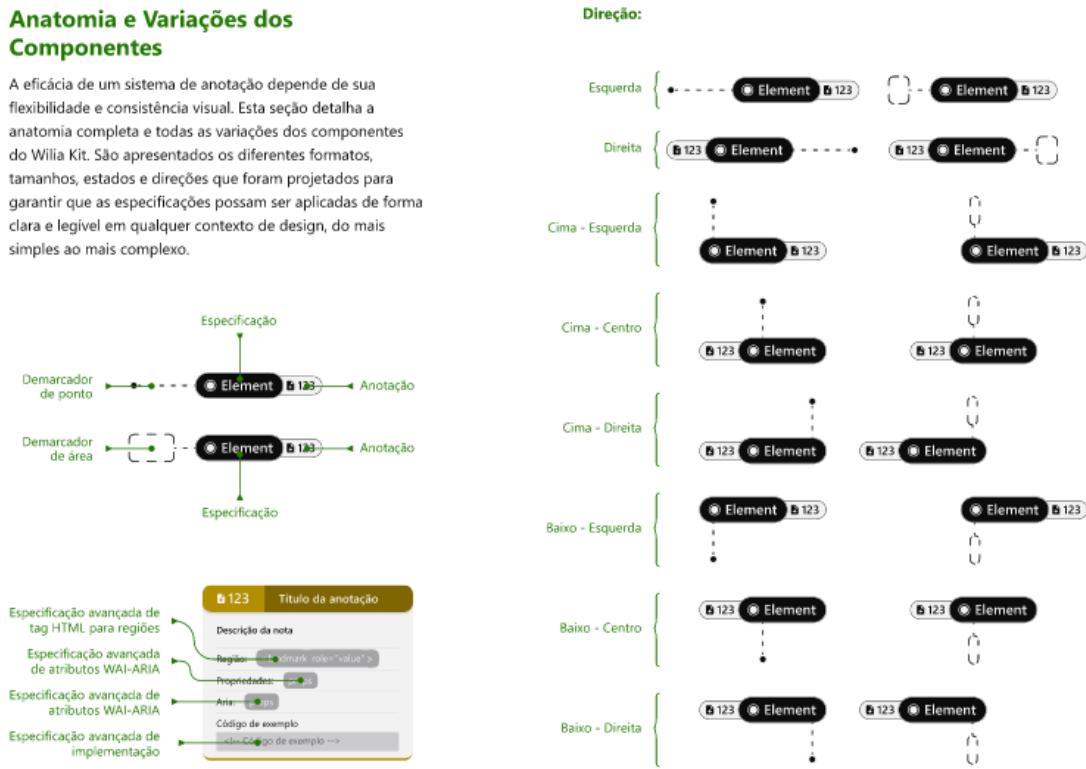
Para campos e controles de formulários que aceitam dados dos usuários, como textos, seleções, datas, etc.

[Leia mais sobre](#)



Fonte: do Autor, 2025

Figura 8 – Anatomia dos componentes Wilia.



Fonte: do Autor, 2025

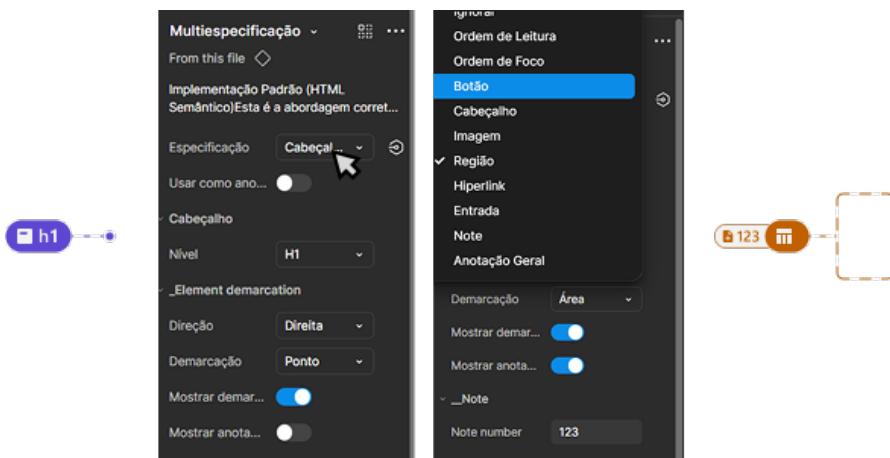
Partindo da visão geral, o kit é estruturado em páginas dedicadas a cada categoria de componente (regiões, cabeçalhos, botões, campos de entrada, imagens, hiperlinks, ordem de leitura e ordem de foco). Cada página contém suas respectivas etiquetas de acessibilidade, que funcionam como anotações visuais aplicadas diretamente sobre os protótipos.

O propósito dessas etiquetas é destacar elementos que requerem documentação de acessibilidade específica, como o uso semântico ideal, a estrutura para leitores de tela e as interações esperadas via teclado. Essa abordagem estratégica proporciona um aprendizado contextualizado e incentiva a aplicação de boas práticas desde a fase de concepção visual. A apresentação visual destas páginas, com suas diretrizes, encontra-se no ([Apêndice A](#)) deste trabalho.

Um resultado significativo deste projeto é a concepção do componente

Multiespecificação. Trata-se de um componente mestre, construído com variantes no Figma, que encapsula todos os tipos de etiquetas disponíveis no Wilia Kit. Na Figura 9 é possível ver sua arquitetura que permite que o usuário, através do painel de propriedades, alterne dinamicamente uma única instância para a especificação desejada. Isso possibilita que um ativo sirva a múltiplos contextos de documentação, representando um avanço em eficiência e usabilidade para o sistema de anotação.

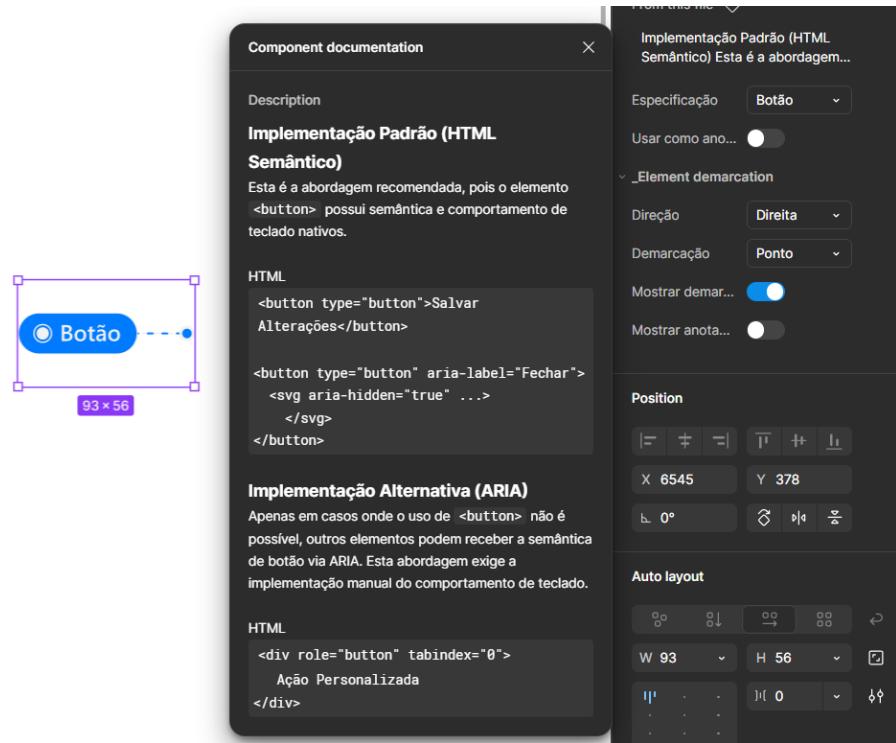
Figura 9 – Exemplo de alteração de uma etiqueta Título para uma Região com um componente de multiespecificação



Fonte: do Autor, 2025

Cada componente foi acompanhado de uma documentação técnica incorporada ao próprio Figma, aparecendo em sua aba lateral contendo orientações práticas, exemplos de marcação HTML, explicações de comportamento esperados em tecnologias assistivas e referências às diretrizes específicas da WCAG que são atendidas. Essa abordagem pode ser vista na Figura 10 onde trouxemos um entendimento prévio sobre a especificação que pode agilizar o processo de desenvolvimento com acessibilidade.

Figura 10 – Exemplo de uma incorporação técnica vista na aba de propriedades de uma etiqueta



Fonte: do Autor, 2025

Todas as páginas possuem uma descrição textual indicando o cenário onde o Wilia é utilizado, e o cenário onde o mesmo não é usado, para indicar aos designers e desenvolvedores o ganho em acessibilidade com o uso do mesmo. Demonstrativamente a Figura 11 possui os cenários mencionados referentes ao item Região.

Figura 11 – Exemplo de descrição de cenários da etiqueta Região

**Cenário de Uso de um Leitor de Tela:**

**Cenário Positivo (Com Landmarks)**

O usuário aciona um atalho e ouve uma lista como: "Banner", "Navegação", "Conteúdo Principal", "Busca", "Rodapé". Ele pode então comandar "Ir para Conteúdo Principal" e pular todo o cabeçalho e menu, começando a consumir a informação relevante imediatamente. Isso economiza um tempo e esforço consideráveis em cada página visitada.

**Cenário Negativo (Sem Landmarks)**

A página é uma "cidade sem placas". O usuário não tem um mapa geral. Para chegar ao conteúdo principal, ele é forçado a navegar linearmente por cada link do cabeçalho e do menu em todas as páginas do site, uma experiência repetitiva e altamente ineficiente.

Fonte: do Autor, 2025

Com essa estrutura, o Wilia consolida-se como um recurso funcional e dinâmico para equipes multidisciplinares, promovendo uma integração mais fluida entre design e desenvolvimento, além de contribuir para a disseminação de uma cultura de acessibilidade aplicada, prática e tecnicamente fundamentada.

Por fim, para ilustrar sua aplicação prática, a seguir apresenta-se dois exemplos de uso do Wilia na documentação de uma tela acessível, sendo essas a [Figura 12](#) e [Figura 13](#). A ferramenta permite descrever visualmente e tecnicamente os padrões de acessibilidade necessários, tornando o processo de implementação mais claro e replicável por toda a equipe.

Figura 12 – Exemplo de documentação de acessibilidade com o Wilia para semântica



Fonte: do Autor, 2025

Figura 13 – Exemplo de documentação de acessibilidade com o Wilia para ordem de leitura e de foco



Fonte: do Autor, 2025

## 5 CONCLUSÃO

A implementação da acessibilidade digital é, simultaneamente, uma obrigação ética, legal e técnica, e um dos maiores desafios para as áreas de design e desenvolvimento web no cenário global. A prática corrente demonstra uma desconexão preocupante entre a existência de diretrizes internacionais e sua efetiva implementação. Análises quantitativas de larga escala confirmam que a grande maioria das aplicações web não atende aos critérios básicos de conformidade. Como consequência direta, perpetua-se um ambiente digital não inclusivo, que impõe barreiras significativas a pessoas com deficiência.

A baixa disponibilidade de ferramentas específicas que integrem os requisitos de acessibilidade já na fase de prototipação contribui de forma significativa para esse cenário, dificultando a transição entre design e desenvolvimento técnico e comprometendo a entrega de produtos acessíveis.

Diante dessa lacuna, o presente trabalho propôs o desenvolvimento do *Wilia*, um UI Kit para documentação de acessibilidade desenvolvido e utilizado no Figma, com foco específico na experiência de usuários que utilizam leitores de tela. O kit foi concebido como uma solução prática e tecnicamente embasada, capaz de guiar designers na criação de protótipos acessíveis e de fornecer aos desenvolvedores a documentação necessária para a implementação correta dos elementos. A estrutura do kit foi baseada no conceito de Atomic Design, nas diretrizes da WCAG 2.2 e da WAI-ARIA, bem como nas más práticas apontadas por estudos como os da WebAIM, HandTalk e o Movimento Web Para Todos, permitindo a criação de um kit de componentes robusto, escalável e funcional.

Os resultados obtidos com o desenvolvimento do kit demonstram a viabilidade da aplicação de normas técnicas de acessibilidade de forma integrada ao processo de design visual, com uma documentação que alinha parte visual com textual, flexível e com isso, facilmente aplicável em diversos projetos.

O kit então organizado em páginas temáticas, como Regiões, Cabeçalhos,

Imagens, entre outras, com a criação de etiquetas visuais explicativas e a incorporação de documentação lateral no próprio ambiente do Figma, facilitando o entendimento e a aplicação dos princípios de acessibilidade tanto para designers quanto para desenvolvedores.

Ao traduzir padrões técnicos, que muitas vezes podem ser complexos, em componentes visuais interativos, o *Wilia* atua como uma conexão entre teoria e prática, alinhando-se as diretrizes do Plano Nacional de Direitos da Pessoa com Deficiência ao incentivar o uso de ferramentas que democratizam a acessibilidade digital desde as fases iniciais do projeto.

Mesmo que este kit, por enquanto, foque apenas em leitores de tela, ele foi feito pensando em crescer. Essa limitação atual já mostra o que podemos fazer no futuro.

Como próximos passos, o mais importante é validar o *Wilia* na prática. Queremos fazer uma pesquisa com designers e desenvolvedores para entender como eles usam o kit e medir se ele realmente ajuda a criar projetos mais acessíveis. O feedback deles será usado para melhorar o material.

Além disso, o plano é expandir o kit. Primeiro, queremos adicionar especificações para outras deficiências, como regras para acessibilidade motora ou cognitiva. Outra ideia é fazer a tradução do *Wilia* para outras línguas, como inglês e espanhol, para que mais usuários possam usar.

Por fim, uma ideia mais complexa para o futuro seria a criação de um plugin (extensão) do *Wilia* para o Figma. Um plugin poderia automatizar parte da documentação ou sugerir atributos de acessibilidade, o que seria um grande avanço para a ferramenta.

## REFERÊNCIAS

ABNOVATO, D. *Html e Table*. 2021. Medium. Acesso em 28 jun. 2025. Disponível em: <<https://douglasabnovato.medium.com/html-e-table-8fed85e21cde>>. Citado na página 44.

Associação Brasileira de Normas Técnicas. *ABNT NBR 17225:2023 — Acessibilidade na Web*. [S.l.]: ABNT, 2023. Citado 2 vezes nas páginas 27 e 29.

BALDIRIS, S. et al. Evaluation of authoring tools under atag and wcag recommendations. *Universal Access in the Information Society*, v. 22, n. 3, 2022. Acesso em: 14 jul. 2025. Disponível em: <<https://doi.org/10.1007/s10209-022-00904-9>>. Citado na página 37.

BENNETT, C. L.; ROSNER, D. K. The promise of empathy: Design, disability, and knowing the “other”. In: *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*. [S.l.]: ACM, 2019. Citado 2 vezes nas páginas 51 e 52.

BigDataCorp. *Apenas 2,9% dos sites brasileiros foram aprovados em todos os testes de acessibilidade*. 2024. Acesso em: 13 fev. 2025. Disponível em: <<https://blog.bigdatacorp.com.br/acessibilidade-nos-sites-brasileiros-2024/>>. Citado na página 23.

BOTELHO, F. Accessibility to digital technology: Virtual barriers, real opportunities. *Assistive Technology*, v. 33, p. 27–34, 12 2021. Citado na página 52.

BRAINLY. *Pencil A11Y Kit*. 2022. <<https://www.figma.com/community/file/1169567323075701693/pencil-a11y-kit>>. Acesso em: 11 Mai. 2025. Citado 2 vezes nas páginas 48 e 50.

Brasil. *Lei nº 12.965, de 23 de abril de 2014*. 2014. <[http://www.planalto.gov.br/ccivil\\_03/\\_ato2011-2014/2014/lei/l12965.htm](http://www.planalto.gov.br/ccivil_03/_ato2011-2014/2014/lei/l12965.htm)>. Estabelece princípios, garantias, direitos e deveres para o uso da Internet no Brasil. Citado na página 29.

BRASIL. *Lei Brasileira de Inclusão da Pessoa com Deficiência (Estatuto da Pessoa com Deficiência)*. [s.n.], 2015. Lei nº 13.146, de 6 de julho de 2015. Disponível em: <[http://www.planalto.gov.br/ccivil\\_03/\\_ato2015-2018/2015/lei/l13146.htm](http://www.planalto.gov.br/ccivil_03/_ato2015-2018/2015/lei/l13146.htm)>. Citado na página 29.

CONTRIBUTORS, U. *Accessibility*. 2024. Acesso em: 10 jun. 2025. Disponível em: <<https://getuikit.com/docs/accessibility>>. Citado na página 24.

- COOPER, A. et al. *About Face: The Essentials of Interaction Design*. Indianapolis: Wiley, 2014. Citado 2 vezes nas páginas 47 e 51.
- CVS Health. *Web Accessibility Annotation Kit*. 2023. <<https://www.figma.com/community/file/1311421011482282592/web-accessibility-annotation-kit?fuid=984228111988630726>>. Acesso em: 11 Set. 2025. Citado 2 vezes nas páginas 48 e 50.
- DIGITAL, B. C. de G. *Modelo de Acessibilidade em Governo Eletrônico – eMAG*. [S.l.]: Ministério da Economia, 2020. Disponível em: <<https://www.gov.br/governodigital/pt-br/acessibilidade>>. Acesso em: 29 jul. 2025. Citado na página 29.
- DOBBALA, M. K.; LINGOLU, M. S. S. Enhancing usability for everyone through web accessibility compliance. *JCSSD*, v. 3, p. 1–13, 05 2024. Citado 2 vezes nas páginas 13 e 31.
- FERNANDES, P. d. S.; FERREIRA, A. D.; NUNES, E. d. G. R. *O leitor de tela e a criação de materiais digitais acessíveis a pessoas com deficiência visual*. 2023. Boletim de Pesquisa e Desenvolvimento, Instituto Capixaba de Pesquisa, Assistência Técnica e Extensão Rural (INCAPER). Boletim Técnico n. 05. Disponível em: <<https://biblioteca.incaper.es.gov.br/digital/bitstream/123456789/3508/1/BRT-oleitordetelaacriacaodemateriaisacessiveis.pdf>>. Citado 3 vezes nas páginas 38, 40 e 41.
- FILGUEIRAS, I. *Inclusão digital no Brasil: uma necessidade*. 2024. Seção “Na Mídia”, NIC.br. [gif]. Disponível em: <<https://nic.br/noticia/na-midia/inclusao-digital-no-brasil-uma-necessidade/>>. Citado na página 27.
- FOUNDATION, I. D. *UI Kits*. 2023. Acesso em: 10 jun. 2025. Disponível em: <<https://www.interaction-design.org/literature/topics/ui-kits>>. Citado na página 24.
- FREIRE, S.; FERRAZ, R. *Acessibilidade digital no Brasil: dados 2024*. 2024. Relatório do Movimento Web para Todos e BigDataCorp. Disponível em: <<https://mwpt.org.br>>. Citado na página 28.
- GARRETT, J. J. *The Elements of User Experience*. 2. ed. Berkeley: New Riders, 2011. Citado 2 vezes nas páginas 45 e 46.
- GILBERT, R. M. *Inclusive Design for a Digital World: Designing with Accessibility in Mind*. New York, NY: Apress, 2016. Citado na página 51.

Hand Talk. *Panorama da Acessibilidade Digital para Pessoas com Deficiência*. 2023. <<https://acessibilidade.handtalk.me/e-book-estudo-panorama-acessibilidade-digital-pcds>>. Acesso em: 2 jul. 2025. Citado 4 vezes nas páginas 23, 47, 53 e 55.

IBGE. *Pesquisa Nacional por Amostra de Domicílios Contínua: Pessoas com deficiência 2022 — Divulgação dos Resultados Gerais (07/07/2023)*. 2023. <[https://agenciadenoticias.ibge.gov.br/media/com\\_mediaibge/arquivos/0a9afaed04d79830f73a16136dba23b9.pdf](https://agenciadenoticias.ibge.gov.br/media/com_mediaibge/arquivos/0a9afaed04d79830f73a16136dba23b9.pdf)>. Acesso em: 13 fev. 2025. Citado 2 vezes nas páginas 23 e 27.

IBRAHIM, N. et al. The effectiveness of web 2.0 tools training workshop using canva and figma in developing creative visual content. *Asian Journal of Assessment in Teaching and Learning*, v. 13, n. 2, p. 35–45, 2023. Disponível em: <<https://doi.org/10.37134/ajatel.vol13.2.4.2023>>. Citado na página 45.

Indeed Design e Stephanie Hagadorn. *A11y Annotation Kit*. 2020. <<https://www.figma.com/community/file/953682768192596304>>. Acesso em: 22 Mai. 2025. Citado na página 48.

INTOPIA. *Intopia's accessibility annotation kit*. 2023. <<https://www.figma.com/community/file/1022394680250523675/intopias-accessibility-annotation-kit>>. Acesso em: 11 Mai. 2025. Citado na página 48.

KOKATE, U.; SHINOHARA, K.; TIGWELL, G. W. *Exploring Accessibility Features and Plug-ins for Digital Prototyping Tools*. [S.l.], 2022. Acesso em: 1 jul. 2025. Disponível em: <<https://repository.rit.edu/cgi/viewcontent.cgi?article=3140&context=article>>. Citado na página 46.

LINDHOLM, S. Bachelor's thesis, *Accessibility Annotations: An Interaction Design Tool Supporting Designers to Design for Accessibility in Graphical User Interfaces*. Malmö, Sweden: [s.n.], 2023. 52 p. Disponível em: <<https://urn.kb.se/resolve?urn=urn:nbn:se:mau:diva-61727>>. Citado 3 vezes nas páginas 46, 47 e 51.

MATUZOVIĆ, M. *Web Accessibility Cookbook: Creating Inclusive Experiences*. [S.l.]: O'Reilly Media, 2024. ISBN 9781098145606. Citado 2 vezes nas páginas 39 e 41.

Microsoft. *Inclusive Design Toolkit*. 2020. <<https://www.microsoft.com/design/inclusive/>>. Acesso em: 22 jul. 2025. Citado na página 51.

Mozilla Developer Network. *HTML elements reference*. 2025. Acesso em: 14 jul. 2025. Disponível em: <<https://developer.mozilla.org/en-US/docs/Web/HTML/Element>>. Citado na página 34.

MWPT. *Quinta edição da pesquisa sobre a experiência de uso de sites por pessoas com deficiência no Brasil*. 2024. <<https://mwpt.com.br/numero-de-sites-brasileiros-aprovados-em-todos-os-testes-de-acessibilidade-tem-queda-em-relacao>>. Acesso em: 13 fev. 2025. Citado 2 vezes nas páginas 23 e 55.

NTT Data. *Resultados da Pesquisa LDT*. 2022. <<https://estudoinclusivo.com.br/pesquisa-ldt/resultados3>>. Acesso em: 12 jul. 2025. Citado 3 vezes nas páginas 27, 28 e 44.

OLIVEIRA, G. M. D.; MONTEIRO, I. T. Development and evaluation of the plugin for figma for accessibility documentation for interfaces - dai. In: *Anais do XXII Simpósio Brasileiro sobre Fatores Humanos em Sistemas Computacionais*. Maceió, AL: Sociedade Brasileira de Computação, 2023. Disponível em: <<https://sol.sbc.org.br/index.php/ihc/article/view/27586>>. Citado na página 46.

PARK, E.; LIM, H. A study on wai-aria compliance rate of the domestic and foreign websites with many users. *Advanced Science Letters*, v. 23, p. 9639–9642, 10 2017. Citado na página 35.

PARTHASARATHY, P. D.; JOSHI, S. Exploring the need of accessibility education in the software industry: Insights from a survey of software professionals in india. *arXiv preprint arXiv:2401.00451*, 2023. Disponível em: <<https://arxiv.org/abs/2401.00451>>. Citado na página 53.

RICHARDS, J.; SPELLMAN, J.; REVIRANUS, J. *Authoring Tool Accessibility Guidelines (ATAG) 2.0*. 2015. (W3C Recommendation). Acesso em: 14 jul. 2025. Disponível em: <<https://www.w3.org/TR/ATAG20/>>. Citado na página 37.

ROSENFIELD, L.; MORVILLE, P.; ARANGO, J. *Information Architecture: For the Web and Beyond*. [S.l.]: O'Reilly Media, 2020. Citado na página 52.

RYHUS, A. *Differences Between UI Kits, Design Systems, and Guidelines*. Medium, 2024. Acesso em 1 jul. 2025. Disponível em: <<https://medium.com/@artur.ryhus/differences-between-ui-kits-design-systems-and-guidelines-ff5ec5db57b0>>. Citado na página 45.

SHAH, H. Advancing web accessibility: A guide to transitioning design systems from wcag 2.0 to wcag 2.1. In: WYLD, D. C. et al. (Ed.). *IoTE, CNDC, ACITY, DPPR, AIAA, NLPTA, WEST, ICSS - 2023*. [S.l.]: Computer Science & Information Technology (CS & IT), 2023. (Computer Science & Information Technology (CS & IT), CSCP 2023), p. 233–245. Citado na página 32.

VIERA, D. F. et al. Inclusive design toolkit for the creation of intergenerational gamified experiences. In: *Proceedings of the 8th International Conference on Technological Ecosystems for Enhancing Multiculturality (TEEM '20)*. New York, NY, USA: Association for Computing Machinery, 2020. p. 707–712. ISBN 9781450388504. Disponível em: <<https://doi.org/10.1145/3434780.3436697>>. Citado na página 51.

W3C. *Web Content Accessibility Guidelines (WCAG) 2.2*. 2023. <<https://www.w3.org/TR/WCAG22/>>. Acessado em 13/02/2025. Citado 3 vezes nas páginas 30, 31 e 33.

W3C. *WAI-ARIA Overview*. 2025. Acesso em: 14 jul. 2025. Disponível em: <<https://www.w3.org/WAI/standards-guidelines/aria/>>. Citado na página 34.

W3C. *Web Content Accessibility Guidelines (WCAG) 2.1*. 2025. <Https://www.w3.org/TR/WCAG21/>. Acessado em 5 jul. 2025. Citado na página 32.

W3C Web Accessibility Initiative. *Accessibility Authoring Tools Guidelines (ATAG)*. 2024. Accessed: 2025-07-14. Disponível em: <<https://www.w3.org/WAI/standards-guidelines/atag/>>. Citado 2 vezes nas páginas 36 e 37.

WAI-ARIA Authoring Practices Task Force. *WAI-ARIA Authoring Practices Guide*. 2023. <<https://www.w3.org/WAI/ARIA/apg/>>. Consultado em julho de 2025. Citado 3 vezes nas páginas 34, 35 e 42.

Web Accessibility Initiative (WAI). *Designing for Web Accessibility*. 2017. <<https://www.w3.org/WAI/gettingstarted/tips/designing>>. Acesso em: 23 jun. 2025. Citado na página 40.

WEBAIM. *WebAIM: Screen Reader User Survey 9 Results*. 2023. Acesso em: 10 jun. 2025. Disponível em: <<https://webaim.org/projects/screenreadersurvey9/>>. Citado 2 vezes nas páginas 27 e 28.

ZDRAVKOVA, K.; DALIPI, F.; KRASNIQI, V. Remote education trajectories for learners with special needs during the covid-19 outbreak: An accessibility analysis of the learning platforms. *International Journal of Emerging Technologies in Learning (iJET)*, v. 17, n. 21, 2022. Disponível em: <<https://doi.org/10.3991/ijet.v17i21.32401>>. Citado na página 32.

# ANEXO A – VISÃO POR COMPONENTE WILIA

## A.1 REGIÕES

Figura 14 – Componente: Regiões/Landmarks

### Diretrizes para Design

O papel do design é atuar como “arquiteto da informação”, dividindo o layout da página em suas seções lógicas e funcionais.

#### Etiquetas

**Header** **Main** **Nav** **Search** **Footer**

**Etiquetas Específicas (Header, Main, Nav, Footer, Search):** Devem ser usadas para demarcar as regiões primárias e inequívocas da página.

- Header: Para o cabeçalho global do site.
- Main: Para o conteúdo principal é único da página. Regra crítica: deve haver apenas um Main por página.
- Nav: Para blocos de navegação primários (menu do site, navegação de seção, paginação).
- Search: Para o formulário de busca principal do site.
- Footer: Para o rodapé global do site.

**Section** **Article**

**Etiquetas Article, Section:** Essas são regiões mais granulares. A etiqueta deve ser usada para agrupar conteúdo tematicamente relacionado, como um post de blog (Article) ou uma seção temática em uma página (Section).

- **Regiões Condicionais:** As regiões <section> e <article> só são consideradas landmarks se tiverem um nome acessível. O designer deve usar a etiqueta e especificar o nome na nota (ex: “Tipo: Seção, Rótulo: Nossos Parceiros”).

**B123**

**Etiqueta Abstrata de Região + Nota:** Utilizada para casos mais complexos:

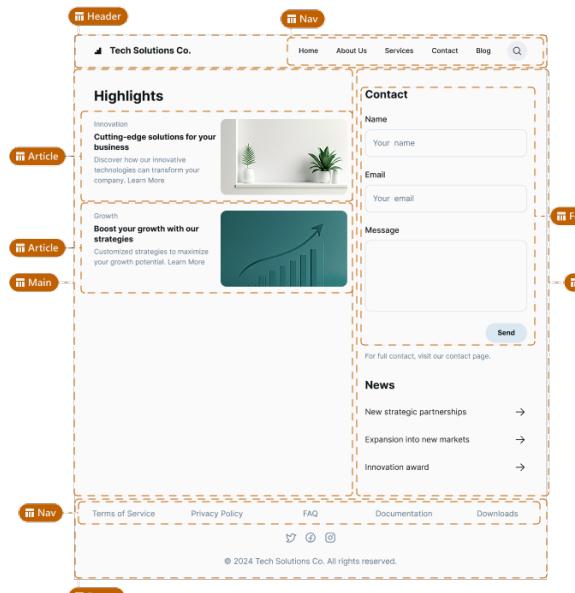
#### Anotações

Para instruções adicionais ou exceções, a etiqueta de cabeçalho deve ser conectada a um componente de “Nota” com os detalhes técnicos.



**Múltiplas Regiões do Mesmo Tipo:** Se a página tiver duas seções de navegação, o designer deve usar a etiqueta de região abstrata para cada uma e, em uma “Nota”, especificar o tipo e um rótulo acessível único. Ex: “Tipo: Navegação, Rótulo: Menu principal” e “Tipo: Navegação, Rótulo: Navegação de ajuda”.

**Diretriz Essencial:** Todo o conteúdo visível de uma página deve estar contido dentro de alguma região de landmark. Isso garante que nenhum conteúdo fique “órfão” ou “perdido” para usuários que navegam por este mapa estrutural.



Fonte: Elaborado pelo autor, 2025.

## A.2 CABEÇALHOS

Figura 15 – Componente: Cabeçalhos

### Diretrizes para Design

A especificação de design deve definir a hierarquia semântica do conteúdo, uma camada de informação que existe independentemente da apresentação visual.

#### Etiqueta



Aplicar uma única vez por página para designar o título principal que descreve o propósito geral do documento.



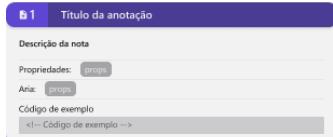
Aplicar para designar subtítulos, criando seções e subseções.

#### Hierarquia Lógica

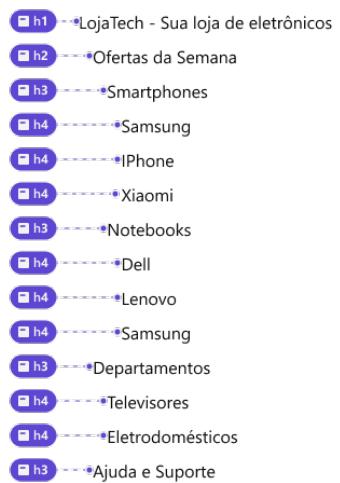
É fundamental que a hierarquia de níveis seja consistente e não seja interrompida. A quebra na sequência invalida a estrutura lógica que as tecnologias assistivas utilizam para gerar o mapa de navegação da página.

#### Anotações

Para instruções adicionais ou exceções, a etiqueta de cabeçalho deve ser conectada a um componente de "Nota" com os detalhes técnicos.



**Diretriz Essencial:** O nível de um cabeçalho (h1, h2, etc.) denota sua função estrutural, não seu estilo visual. A equipe de design deve especificar a hierarquia semântica correta, enquanto as propriedades visuais são definidas de forma independente durante o desenvolvimento.



Fonte: Elaborado pelo autor, 2025.

## A.3 BOTÕES

Figura 16 – Componente: Botões

### Diretrizes para Design

A especificação de design deve garantir que todo elemento que executa uma ação seja identificado como um botão e que sua função seja clara para todos os usuários.

#### Etiqueta



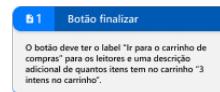
Aplicar para identificar todos os elementos interativos que realizam ações. Caso o protótipo já defina que o elemento será um botão nos componentes, seu uso não é necessário.

- Para botões com texto visível, o próprio texto serve como nome acessível (ex: "Salvar").
- Para botões que contêm apenas um ícone (ex: um 'X' para fechar), o nome acessível deve ser explicitamente especificado em um componente de "Nota" conectado (ex: "Fechar janela").
- Para botões com texto visível sem contexto aprimorado sobre a ação, utilize também um componente de "Nota" para definir um nome acessível contextual.



#### Estados Visuais

Devem ser definidos todos os estados do botão: padrão, :hover (ponteiro sobre), :focus (foco do teclado), :active (pressionado) e desabilitado. O estado :focus é obrigatório para acessibilidade.



#### Anotações

Para instruções adicionais ou exceções, a etiqueta de botão deve ser conectada a um componente de "Nota" com os detalhes técnicos.



**Diretriz Essencial:** Um botão executa uma ação. Um link realiza uma navegação. Esta distinção deve ser respeitada rigorosamente no design. Se um elemento leva o usuário para uma nova página ou recurso, deve ser especificado como um Hiperlink, não como um Botão.

Fonte: Elaborado pelo autor, 2025.

## A.4 CAMPOS DE ENTRADA

Figura 17 – Componente: Campos de Entrada

### Diretrizes para Design

O design deve especificar o tipo, o rótulo e o comportamento esperado para cada elemento do formulário.

#### Etiqueta



O componente de input está devidamente documentado para apoiar os desenvolvedores na implementação, quando sua estrutura e finalidade já estiverem bem definidas no protótipo. Se o campo estiver corretamente rotulado visual e semanticamente (com rótulo visível, tipo de dado claro e função compreensível), nenhuma especificação auxiliar é necessária. Opcionalmente, recomenda-se alterar o nome da etiqueta para refletir o tipo de campo representado — por exemplo: Texto, Data, Select, CPF, etc.

#### Princípios Gerais (Aplicáveis a Todos os Campos)

**Rótulo Visível:** Todo campo, sem exceção, deve ter um rótulo textual visível e persistente.

**Tratamento de Erros:** Deve existir um design claro para o estado de erro, incluindo um indicador visual que não dependa apenas da cor, uma mensagem de erro textual e, para formulários longos, um sumário de erros no topo.

#### Campos de Texto (input[type=text], textarea)

- Além do rótulo, especificar se o campo é de linha única (input) ou múltipla (textarea).
- Utilizar uma "Nota" para definir máscaras de formato (ex: CPF, telefone) e textos de exemplo (placeholder).

#### Caixas de Seleção (checkbox)

- Cada checkbox individual deve ter seu próprio rótulo (ex: "Aceito os termos").
- Se várias caixas de seleção formam um grupo respondendo a uma pergunta (ex: "Quais tópicos de interesse?"), o grupo inteiro deve receber uma legenda principal.
- Os estados (marcado, desmarcado, indeterminado) devem ser visualmente distintos.

#### Botões de Rádio (radio)

- É mandatório que todo grupo de botões de rádio tenha uma legenda que represente a pergunta (ex: "Qual o seu nível de escolaridade?").
- Cada opção de rádio deve ter seu próprio rótulo.
- Apenas uma opção no grupo pode ser selecionada por vez

#### Menus de Opções (select)

- O menu deve ter um rótulo claro que descreva o que está sendo escolhido (ex: "Selecionar o seu estado").
- Se a lista de opções for muito longa e puder ser categorizada (ex: estados por região), a especificação deve incluir o uso de grupos de opções (optgroup) para melhorar a usabilidade.

#### Campos de Data (input[type=date] e customizados)

**Seletor Nativo:** Especificar o uso do campo de data nativo do navegador. Esta é a opção preferencial por ser acessível por padrão, embora seu visual varie entre navegadores.

**Seletor Customizado:** Se um seletor customizado for absolutamente necessário, a especificação de design deve ser extremamente detalhada, incluindo: a navegação completa via teclado (setas para dias, PageUp/Down para meses, Esc para fechar), os nomes acessíveis para botões de navegação ("Próximo mês", "Mês anterior") e todos os estados visuais.

**Diretriz Essencial:** A complexidade da especificação aumenta com a complexidade do componente. Para componentes não-nativos como um seletor de data customizado, a responsabilidade do design é definir todo o comportamento interativo, não apenas a aparência estática.

#### Anotações

Para instruções adicionais ou exceções, a etiqueta do input deve ser conectada a um componente de "Nota" com os detalhes técnicos para descrever melhor o componente e solicitar atributos complementares como textos descritivos, ajuda, autocompletamento, etc.



B 1	Título da anotação
Descrição da nota	
Propriedades: <input type="text" value="anotação"/>	
Ano: <input type="text" value="2025"/>	
Código de exemplo	<input type="button" value="..."/>
<small>x!—Código de exemplo --&gt;</small>	

**Diretriz Essencial:** Um formulário é uma conversa. Cada pergunta (rótulo/legenda) deve ser clara, e o sistema deve responder de forma útil quando o usuário comete um erro, guiando-o para a correção em vez de apenas bloquear seu progresso.

Nome Completo	<input type="text" value="Insira seu nome completo"/>	<input type="button" value="Texto"/>												
Email *	<input type="text" value="Insira seu email"/>	<input type="button" value="Email"/>												
<table border="1"> <tr> <td>B 1</td> <td>Campo de email</td> </tr> <tr> <td colspan="2">Deve aceitar apenas e-mails válidos, disponibilizar o autocompleto do navegador e fazer com que o campo é obrigatório para leitores. O asterisco tem que ser ouvido para leitores.</td> </tr> <tr> <td colspan="2">ou</td> </tr> <tr> <td>B 1</td> <td>Campo de email</td> </tr> <tr> <td colspan="2">Propriedades: <input type="text" value="autocomplete='email' required"/></td> </tr> <tr> <td colspan="2">Aria: <input type="text" value="aria-describedby='email'"/></td> </tr> </table>			B 1	Campo de email	Deve aceitar apenas e-mails válidos, disponibilizar o autocompleto do navegador e fazer com que o campo é obrigatório para leitores. O asterisco tem que ser ouvido para leitores.		ou		B 1	Campo de email	Propriedades: <input type="text" value="autocomplete='email' required"/>		Aria: <input type="text" value="aria-describedby='email'"/>	
B 1	Campo de email													
Deve aceitar apenas e-mails válidos, disponibilizar o autocompleto do navegador e fazer com que o campo é obrigatório para leitores. O asterisco tem que ser ouvido para leitores.														
ou														
B 1	Campo de email													
Propriedades: <input type="text" value="autocomplete='email' required"/>														
Aria: <input type="text" value="aria-describedby='email'"/>														

Fonte: Elaborado pelo autor, 2025.

## A.5 IMAGENS

Figura 18 – Componente: Imagens

### Diretrizes para Design

A responsabilidade do design é determinar o propósito de cada imagem no contexto do layout, pois esta definição é a base para a implementação acessível.

#### Etiquetas



Aplicar para identificar os elementos de imagem.

#### Utilização

É mandatório conectar uma "Nota" a cada etiqueta de imagem para especificar sua categoria e, se aplicável, seu texto alternativo. A nota deve conter uma das seguintes diretrizes:

**Informativa:** Fornecer o texto alternativo exato que descreve a imagem ou ícone. O texto deve ser conciso e objetivo.

**Decorativa:** Indicar "Ignore". Nenhum texto adicional é necessário.

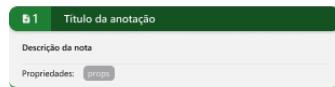
**Complexa:** Indicar "Imagen" e fornecer um texto alternativo curto. A especificação também deve indicar onde a descrição longa e detalhada da imagem estará localizada no conteúdo adjacente.

**Imagen de Texto:** Indicar "Imagen" e fornecer na nota o texto exato contido na imagem.

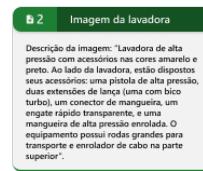
**Ícone:** Avalie se o ícone traz valor de contexto. Se sim: Indique "Ícone" e inclua um descriptivo na nota. Se não: Indique "Ignorar".

#### Anotações

Para instruções adicionais ou exceções, a etiqueta de imagem ou ícone devem ser conectadas a um componente de "Nota" com os detalhes técnicos.



**Diretriz Essencial:** A equipe de desenvolvimento não tem como prever o propósito ou o conteúdo informacional de uma imagem. A ausência de uma especificação clara no design resulta em uma implementação inacessível por padrão. O designer é a fonte de verdade para o conteúdo e o contexto da imagem.



Fonte: Elaborado pelo autor, 2025.

## A.6 LINKS

Figura 19 – Componente: Links

labelfig:componente-links

### Diretrizes para Design

O design deve garantir que links sejam visualmente identificáveis e que seu texto descreva claramente o destino da navegação.

#### Etiqueta



Aplicar para identificar todos os elementos que navegam o usuário.

#### Texto Descriptivo

A especificação deve conter o texto exato do link. Este texto deve ser autoexplicativo. Por exemplo, em vez de um card de notícias com um link "Leia mais", o título da notícia em si deve ser o link.

#### Aparência e Estados

O design deve diferenciar links de texto comum (ex: usando cor e sublinhado). Todos os estados devem ser definidos: padrão (link), :visited (visitado), :hover (ponteiro sobre), :focus (foco do teclado) e :active (pressionado).

#### Hierarquia Lógica

Texto Descriptivo:

Aparência e Estados:

Componente de "Nota": Utilizar uma nota para especificar comportamentos especiais:

Abertura em Nova Aba: Indicar se o link deve abrir em uma nova aba (target="\_blank"). Este comportamento deve ser usado com moderação, pois pode desorientar os usuários.

Links para Documentos: Indicar se o link aponta para um tipo de arquivo específico (ex: "[PDF]", "[DOCX]").

**Diretriz Essencial:** Um hyperlink realiza uma NAVEGAÇÃO. O seu texto deve descrever o DESTINO da forma mais clara possível. Se um elemento executa uma AÇÃO na página atual (como "Salvar" ou "Fechar"), ele deve ser um Botão.

#### Anotações

Para instruções adicionais ou exceções, a etiqueta de Hiperlink deve ser conectada a um componente de "Nota" com os detalhes técnicos.



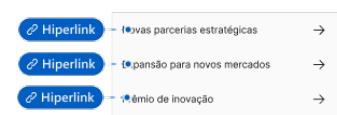
B1	Link da anotação
Descrição da nota	
Propriedades: props	
Aba: props	

Abertura em Nova Aba: Indicar se o link deve abrir em uma nova aba (target="\_blank"). Este comportamento deve ser usado com moderação, pois pode desorientar os usuários.

Links para Documentos: Indicar se o link aponta para um tipo de arquivo específico (ex: "[PDF]", "[DOCX]").

**Diretriz Essencial:** Um hyperlink realiza uma NAVEGAÇÃO. O seu texto deve descrever o DESTINO da forma mais clara possível. Se um elemento executa uma AÇÃO na página atual (como "Salvar" ou "Fechar"), ele deve ser um Botão.

Fonte: Elaborado pelo autor, 2025.



Crescimento  
Impulsiona seu crescimento com nossas estratégias  
Estratégias personalizadas para maximizar seu potencial de crescimento. Saiba Mais



B1 Link facebook  
O link deve ter o label para os leitores "Perfil do facebook" e o contexto "Abre em nova guia".

ou  
B1 Link instagram  
Propriedades: target=\_blank  
Aba: aria-label="Perfil do instagram" aria-describedby="Abre em nova guia"

## A.7 ORDEM DE LEITURA

Figura 20 – Componente: Ordem de Leitura

### Diretrizes para Design

O papel do design é criar layouts que possam ser implementados com uma ordem de código lógica. A anotação da ordem de leitura só é necessária quando um layout complexo puder gerar ambiguidade para a equipe de desenvolvimento.

#### Etiqueta

lk 1

Diferente da ordem de foco, que se aplica a poucos elementos, numerar cada parágrafo seria impraticável. Em vez disso, para layouts complexos (ex: painéis, dashboards, layouts de revista), deve-se usar setas ou uma numeração simplificada para indicar o fluxo de leitura desejado entre os grandes blocos de conteúdo.

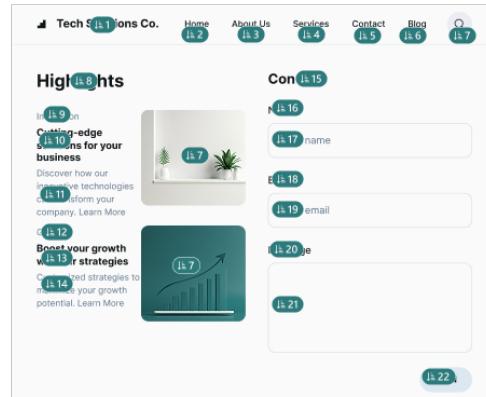
#### Validação do Design

Validação do Design: Uma pergunta fundamental que a equipe de design deve fazer é: "Se todo o CSS fosse removido, o conteúdo desta página ainda faria sentido lido de cima para baixo?". Se a resposta for não, o layout tem um problema inerente de ordem de leitura que precisa ser resolvido ou, no mínimo, claramente anotado.

Janelas Modais: É mandatório especificar o comportamento do foco para modais.

- Ao abrir a modal, o foco deve ser movido para o primeiro elemento interativo dentro dela.
- A ordem de foco deve ficar "presa" (trap) dentro da modal, circulando apenas entre seus elementos.
- Ao fechar a modal, o foco deve retornar programaticamente para o elemento que a ação abriu.

**Diretriz Essencial:** A ordem de leitura deve seguir a narrativa lógica do conteúdo. Um bom design visual não deve depender de uma estrutura de código ilógica. A apresentação visual (CSS) deve servir à estrutura do conteúdo (HTML), e não o contrário.



Fonte: Elaborado pelo autor, 2025.

## A.8 ORDEM DE FOCO

Figura 21 – Componente: Ordem de Foco

### Diretrizes para Design

O papel do design é especificar a ordem de foco apenas quando ela não for óbvia ou quando um fluxo de interação específico for crucial. Na maioria dos layouts padrão (cabecalho, conteúdo, rodapé), a ordem natural do código será suficiente e não requer especificação.

#### Etiqueta



Para especificar uma ordem de foco customizada, devem ser utilizadas as etiquetas numeradas do kit. Cada etiqueta é posicionada ao lado, sobre ou direcionada um elemento interativo no design. A sequência dos números (1, 2, 3, ...) define o caminho exato que a tecla Tab deve seguir.

#### Quando Especificar

**Layouts Complexos:** Em dashboards ou interfaces com múltiplos painéis, a ordem de foco pode não seguir um fluxo linear de cima para baixo.

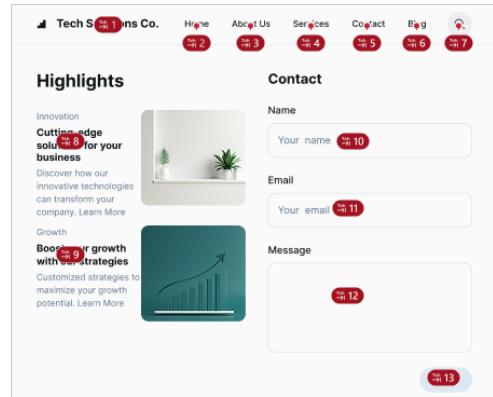
**Janelas Modais:** É mandatório especificar o comportamento do foco para modais.

- Ao abrir a modal, o foco deve ser movido para o primeiro elemento interativo dentro dela.
- A ordem de foco deve ficar "presa" (trap) dentro da modal, circulando apenas entre seus elementos.
- Ao fechar a modal, o foco deve retornar programaticamente para o elemento que a acionou.

**Não é necessário anotar tudo:** Em um formulário simples, por exemplo, não é preciso numerar cada campo. A ordem de cima para baixo já é esperada e será o padrão.

**Diretriz Essencial:** A ordem de foco deve, por padrão, seguir a ordem de leitura visual da página (de cima para baixo, da esquerda para a direita em culturas ocidentais).

Qualquer desvio deste padrão é uma exceção que deve ser explicitamente desenhada e justificada para melhorar a experiência do usuário, nunca para complicá-la.



Fonte: Elaborado pelo autor, 2025.