

**UNIVERSIDADE DE SÃO PAULO**

Instituto de Ciências Matemáticas e de Computação

**SRS - um simulador visual de leitor de tela para desenvolvedores web**

**Felipe Tassario Gomes**

Dissertação de Mestrado do Programa de Pós-Graduação em Ciências de Computação e Matemática Computacional (PPG-C<sup>2</sup>MC)



SERVIÇO DE PÓS-GRADUAÇÃO DO ICMC-USP

Data de Depósito:

Assinatura: \_\_\_\_\_

**Felipe Tassario Gomes**

## SRS - um simulador visual de leitor de tela para desenvolvedores web

Dissertação apresentada ao Instituto de Ciências Matemáticas e de Computação – ICMC-USP, como parte dos requisitos para obtenção do título de Mestre em Ciências – Ciências de Computação e Matemática Computacional. *EXEMPLAR DE DEFESA*

Área de Concentração: Ciências de Computação e Matemática Computacional

Orientadora: Profa. Dra. Renata Pontin de Mattos Fortes

**USP – São Carlos**  
**Março de 2018**

Ficha catalográfica elaborada pela Biblioteca Prof. Achille Bassi  
e Seção Técnica de Informática, ICMC/USP,  
com os dados inseridos pelo(a) autor(a)

G633s      Gomes, Felipe Tassario  
              SRS - um simulador de leitor de tela para  
desenvolvedores web / Felipe Tassario Gomes;  
orientadora Renata Pontin de Mattos Fortes. -- São  
Carlos, 2018.  
              83 p.

Dissertação (Mestrado - Programa de Pós-Graduação  
em Ciências de Computação e Matemática  
Computacional) -- Instituto de Ciências Matemáticas  
e de Computação, Universidade de São Paulo, 2018.

1. Acessibilidade na Web. 2. Usabilidade. 3.  
Tecnologia Assistiva. 4. Leitor de Tela. 5.  
Elementos de Interação. I. Fortes, Renata Pontin de  
Mattos, orient. II. Título.

**Felipe Tassario Gomes**

**SRS - a visual simulator of a screen reader for web  
developers**

Master dissertation submitted to the Institute of  
Mathematics and Computer Sciences – ICMC-USP,  
in partial fulfillment of the requirements for the  
degree of the Master Program in Computer Science  
and Computational Mathematics. *EXAMINATION  
BOARD PRESENTATION COPY*

Concentration Area: Computer Science and  
Computational Mathematics

Advisor: Profa. Dra. Renata Pontin de Mattos Fortes

**USP – São Carlos**  
**March 2018**



*Este trabalho é dedicado a minha família.  
Em especial, à minha mãe, meus avós, tios e tias,  
por me mostrarem o que é o amor incondicional.*



# AGRADECIMENTOS

---

---

Agradeço a todos que colabaram, diretamente ou indiretamente, no meio caminho acadêmico e profissional para chegar até aqui.

Em especial:

À Renata Pontin, minha orientadora, por sempre ser tão bondosa, paciente e atenciosa durante o desenvolvimento deste trabalho. E ao Thiago Pardo, orientador de iniciação científica na graduação, por ter me apresentado o caminho da pesquisa científica.

À Lianna Duarte, André Salgado e Flávia Santos, pelo suporte e ajuda crucial na reta final.

Ao Sergio Campos, o “culpado” por dar a motivação inicial para este trabalho de mestrado, e a Marcio Galli e Fabrício Zuardi, por iniciarem a minha carreira profissional.

À minha mãe e a minha família, por sempre acreditarem em mim e me apoiarem.

À Mariela Mayumi, por aguentar as minhas crises e dilemas.

Às amigas que também enfrentam ou já enfrentaram este trabalho árduo: Livia Bueno, Letícia Portella, Ariane Sasso, Mariana Bunoro; pela empatia com esse sofrimento e por sempre compartilharem conselhos e motivação.

Ao Matt, por provar por  $A + B$  que os canadenses são tão legais quanto todos falam; e ao Frank e Jared, por mostrar que os americanos também são.

E, principalmente, aos meus amigos do coração, que estão presentes nos melhores momentos da minha vida, e sempre me trazem alegria e fazem tudo valer a pena: Theodoro, Matheus, Kadu, Diogo, Luís Fernando, Pedro, Ana Carolina, Vinicius, Leonardo, Caio, Filípi, Kermentz...



*“Somos poeira das estrelas.  
Nós somos uma maneira de o cosmos se autoconhecer.”  
(Carl Sagan)*



# RESUMO

GOMES, F. T. **SRS - um simulador visual de leitor de tela para desenvolvedores web**. 2018. 83 p. Dissertação (Mestrado em Ciências – Ciências de Computação e Matemática Computacional) – Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo, São Carlos – SP, 2018.

A acessibilidade, como um conceito geral, se refere à capacidade de produtos, meios, serviços e dispositivos em alcançar o uso por pessoas com deficiência. A acessibilidade na Web diz respeito à capacidade destas pessoas em utilizá-la, acessar os conteúdos e poder participar ativamente dela, interagindo e gerando novos conteúdos. A Web de hoje em dia oferece ricas experiências aos usuários, porém muitas barreiras são enfrentadas por usuários com deficiências visuais, devido à falta de atenção dada pelos desenvolvedores às questões de acessibilidade das páginas. A construção de uma página que atenda adequadamente aos requisitos de acessibilidade é uma tarefa técnica complicada. Diversos padrões e recomendações foram criadas para tal, mas o domínio desses padrões e a compreensão de como eles ajudarão na experiência de usuários com deficiências não é simples. Ferramentas existentes validam detalhes técnicos da implementação das páginas, geralmente seguindo uma lista de itens ou recomendações a serem verificadas, mas essas ferramentas não expõem ao desenvolvedor o efeito final na experiência que um usuário cego terá ao visitá-la, ou evidenciam a importância relativa de cada problema encontrado. Neste trabalho, estudou-se a viabilidade e o potencial de um simulador de leitor de tela, cujo público alvo são os próprios desenvolvedores, de forma a auxiliar a análise de acessibilidade de uma página para usuários com deficiência visual. Dois protótipos foram desenvolvidos e avaliados, buscando verificar as lacunas existentes em outras abordagens que ele possa alcançar.

**Palavras-chave:** Acessibilidade na Web, Usabilidade, Tecnologia Assistiva, Leitor de Tela, Elementos de Interação.



# ABSTRACT

GOMES, F. T. **SRS - a visual simulator of a screen reader for web developers**. 2018. 83 p. Dissertação (Mestrado em Ciências – Ciências de Computação e Matemática Computacional) – Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo, São Carlos – SP, 2018.

Accessibility, as a general concept, refers to the capacity of products, services and environments to be used by people with disabilities. Web accessibility, therefore, constitutes the ability of the web to be any person, including people with disabilities. The web nowadays offers rich experiences to its users, but there are many barriers that a person with visual disabilities faces while browsing the web. Building a web page that is well adapted to accessibility requirements is a complicated technical task. Several standards and guidelines have been created to achieve this, but mastering these standards and understanding how they will help these users' experiences isn't simple. There are existing tools to validate technical details of a web page's implementation, usually referring back to the standards or a list of items to be verified, but these tools don't tell the web developer the effects that each problem will have in the final experience, or the relative importance of each existing problem. In this research, we studied the viability and potential of a screen reader simulator, targeting web developers as its main user group, in order to support the accessibility analysis of a webpage for blind users. Two prototype versions were developed and evaluated, in order to verify the gaps from other types of tools that this approach can reach.

**Keywords:** Web Accessibility, Usability, Assistive Technology, Screen Reader, Interaction Elements.



# LISTA DE ILUSTRAÇÕES

---

---

Figura 1 – Relação entre as recomendações do W3C e seus componentes de atuação . . .	30
Figura 2 – Conjunto de problemas encontrados por usuários com deficiência visual . . .	33
Figura 3 – Ferramenta Vischeck . . . . .	44
Figura 4 – Ferramenta Fangs Screen Reader Emulator . . . . .	45
Figura 5 – Ferramenta WebAIM LowVision Simulator . . . . .	46
Figura 6 – Ferramenta NoCoffee - Vision Simulation . . . . .	47
Figura 7 – Ferramenta Accessibility Designer (aDesigner) . . . . .	48
Figura 8 – SRS Versão 1: antes de ser ativado o simulador, ilustração da página em seu estado original. . . . .	53
Figura 9 – SRS Versão 1: após o simulador ser ativado, ilustração da página com todo o seu conteúdo escurecido, exceto pelo bloco de texto que contém foco. . . . .	53
Figura 10 – Uso da SRS - Versão 1: ilustração da página ao ser utilizada com o simulador ativado, em que o foco de texto é percorrido com a tecla TAB até atingir outro conteúdo. . . . .	54
Figura 11 – SRS - Versão 2 - Simulador ativado e seu painel de controle . . . . .	57
Figura 12 – SRS - Versão 2 - Simulador no modo de semi-transparência . . . . .	57
Figura 13 – SRS - Versão 2 - Destaque para o painel de controle com funções de navegação	58
Figura 14 – Taxa de Sucesso em tarefas no site São Carlos Agora . . . . .	63
Figura 15 – Taxa de Sucesso em tarefas no site da Polícia Federal . . . . .	63
Figura 16 – Distribuição de respostas sobre características positivas . . . . .	66
Figura 17 – Distribuição de respostas sobre características negativas . . . . .	66



# LISTA DE TABELAS

---

---

Tabela 1 – Tarefas para o site SCA . . . . .	61
Tabela 2 – Tarefas para o site PF . . . . .	62
Tabela 3 – Distribuição de Tarefas aos Grupos . . . . .	62
Tabela 4 – Afirmções positivas do questionário de carga mental de trabalho . . . . .	65
Tabela 5 – Afirmções negativas do questionário de carga mental de trabalho . . . . .	65
Tabela 6 – Avaliação das páginas por meio de validadores . . . . .	68
Tabela 7 – Erros relatados pelas ferramentas de validação . . . . .	68
Tabela 8 – Erros relatados pela ferramenta de simulação . . . . .	69



# LISTA DE ABREVIATURAS E SIGLAS

---

---

API	<i>Application Programming Interface</i>
ATAG	<i>Authoring Tool Accessibility Guidelines</i>
CMYK	<i>Cyan, Magenta, Yellow, Key</i>
CRT	<i>Cathode Ray Tube</i>
eMAG	Modelo de Acessibilidade em Governo Eletrônico
HTML	HyperText Markup Language
HTML	HyperText Markup Language
JAWS	<i>Job Access With Speech</i>
LCD	<i>Liquid-Crystal Display</i>
NVDA	<i>NonVisual Desktop Access</i>
PF	Polícia Federal
RIA	<i>Rich Internet Application</i>
SCA	São Carlos Agora
UAAG	<i>User Agent Accessibility Guidelines</i>
URL	<i>Uniform Resource Location</i>
WAI	<i>Web Accessibility Initiative</i>
WAVE	<i>Web Accessibility Evaluation Tool</i>
WCAG	<i>Web Content Accessibility Guidelines</i>



# SUMÁRIO

---

---

1	<b>INTRODUÇÃO</b>	23
1.1	Motivação	24
1.2	Objetivos	26
1.3	Organização	26
2	<b>TECNOLOGIA ASSISTIVA E ACESSIBILIDADE</b>	27
2.1	Tecnologia Assistiva	27
2.2	Leitores de Tela	28
2.3	Acessibilidade na Web	29
2.4	Acessibilidade na Web para pessoas com deficiência visual	31
2.5	Problemas de acessibilidade para pessoas cegas	34
2.6	Considerações Finais	39
3	<b>TRABALHOS RELACIONADOS</b>	41
3.1	Validadores	43
3.2	Simuladores	43
3.2.1	<i>Vischeck</i>	44
3.2.2	<i>Fangs Screen Reader Emulator</i>	45
3.2.3	<i>WebAIM Simulation</i>	45
3.2.4	<i>Accessibility Designer - aDesigner</i>	47
3.3	Considerações Finais	48
4	<b>SRS - UM SIMULADOR DE LEITOR DE TELA PARA DESENVOLVEDORES WEB</b>	49
4.1	Contexto do Simulador SRS	49
4.2	O simulador SRS e seu funcionamento	51
4.3	SRS - Versão 1	52
4.4	SRS - Versão 2	55
4.5	Considerações Finais	58
5	<b>AVALIAÇÃO</b>	59
5.1	Testes com Usuários executando Tarefas	59
5.1.1	<i>Metodologia</i>	60
5.1.2	<i>Resultados</i>	61

5.2	<b>Avaliação de Carga Mental de Trabalho</b> . . . . .	64
5.2.1	<i>Metodologia</i> . . . . .	64
5.2.2	<i>Resultados</i> . . . . .	65
5.3	<b>Inspeção por especialistas</b> . . . . .	67
5.3.1	<i>Metodologia</i> . . . . .	67
5.3.2	<i>Resultados</i> . . . . .	67
5.4	<b>Considerações Finais</b> . . . . .	69
6	<b>CONCLUSÕES</b> . . . . .	71
	<b>REFERÊNCIAS</b> . . . . .	75
	<b>GLOSSÁRIO</b> . . . . .	83

---

## INTRODUÇÃO

---

A proposta original da criação da Web, no início da década de 90, foi de que ela consistisse numa rede descentralizada de conteúdos interligados por meio de referências, os chamados *hiperlinks*. Essas ligações permitem que um documento referencie inúmeros outros de uma maneira direta e absoluta (por meio de *Uniform Resource Location* (URL)). Dessa forma, permite ao usuário navegar facilmente entre os documentos com o auxílio de um navegador, que também é usualmente chamado de “Agente do Usuário” (*User Agent*) nos meios acadêmicos e de desenvolvimento dos navegadores e padrões Web (AGHAEI; NEMATBAKSH; FARSANI, 2012).

Essa Web, como originalmente concebida (BERNERS-LEE, 1989), foi amplamente bem sucedida, e tornou-se uma rede com conteúdo de diversas origens, em diversos idiomas, hospedados de maneira distribuída ao redor do mundo. Um dos fatores significantes para seu sucesso, além das propriedades previamente mencionadas, foi a relativa facilidade para criação e publicação de novos conteúdos, sem a necessidade de obter permissão para interligá-los aos conteúdos já existentes.

Além disso, a grande capacidade de formatação e apresentação dos conteúdos de diversas maneiras, tamanhos e cores, incentivou a participação mais ampla das pessoas. Assim, não apenas os criadores de conteúdo para Web, mas também as pessoas interessadas nos aspectos gráficos, de design, e artísticos da composição dos documentos (também comumente conhecidos por “páginas”), passaram a contribuir com a diversidade de conteúdos disponibilizados na Web.

Inicialmente, a grande maioria dos conteúdos disponíveis na Web consistiam em documentos estáticos, completamente pré-definidos e que se apresentavam de uma forma única a todos usuários que o acessavam. Entretanto, com o crescimento da Web, e a diversificação de conteúdos e das pessoas que os desenvolviam, a demanda por capacidades dinâmicas nos navegadores começou a surgir, tanto para motivos puramente estéticos (como a disponibilização de diversos *layouts* para uma mesma página, a expansão de seções do conteúdo que se apresentavam

inicialmente escondidas, a alteração de cores e fontes), quanto para motivos de personalização da experiência (como a interação do usuário com a página e a apresentação diversificada e personalizada de conteúdos específicos para cada usuário que visita a página).

Assim, todas essas características de disponibilização de conteúdos, aos mais diversos tipos de usuários, passaram a exigir que cada vez mais a Web se tornasse o mais acessível possível (POWER, 2012).

## 1.1 Motivação

Com todos os avanços da Web, no início da década de 2000, os desenvolvedores web da época começaram a perceber que eles tinham em mãos, com o uso do navegador, um ambiente completo para desenvolver experiências que iam além de um simples documento de coluna simples com textos e imagens.

Na Web inicial, posteriormente apelidada de Web 1.0, o conteúdo estático dos documentos tornava o seu tratamento simples. Os *user agents* conseguiam aplicar transformações necessárias a estas páginas para servir diversas necessidades dos seus usuários, como por exemplo: aumentar tamanho do texto, remover imagens de fundo, alterar esquemas de cores, etc. Tais modificações eram geralmente suficientes para atender algumas necessidades especiais e tornar o conteúdo acessível a pessoas com deficiências visuais, por exemplo. Assim, por meio do uso de ferramentas de Tecnologia Assistiva (a serem apresentadas no [Capítulo 2](#), um usuário com deficiência visual é capaz de perceber e ter acesso ao conteúdo da página e navegar com facilidade com a utilização de seus links.

No entanto, com a Web 2.0, segundo (COOPER, 2007), esse acesso se tornou uma tarefa mais difícil, devido à dinamicidade do conteúdo das páginas, e da utilização do HyperText Markup Language (HTML) tanto para marcas semânticas (BERNERS-LEE; HENDLER; LASSILA, 2001) quanto para fins estruturais do *layout*. Isto se deve, em parte, à ordem dos acontecimentos: a criação de conteúdos mais complexos na Web surgiu de uma maneira *ad-hoc*, naturalmente, sem um planejamento ou desenvolvimento de tecnologias específicas para tal; e só após algumas práticas da Web 2.0 ficarem bem estabelecidas é que se procurou adaptá-las de modo a torná-las propriamente acessíveis.

Um problema gerado por essa falta de padronização é a diversidade de abordagens possíveis para implementar algo desejado. Por exemplo, no desenvolvimento tradicional de aplicações para *desktop*, o sistema operacional provê funções para definir botões em aplicativos. Ao utilizar essas funções, a partir de uma *Application Programming Interface* (API) explicitamente definida pelo sistema operacional, o sistema é capaz de entender claramente o propósito daquilo (um botão), aplicar um estilo gráfico padrão que é uniformemente adotado por todo o sistema e para todos aplicativos, reconhecer o seu texto, modificar cores ou tamanhos conforme a necessidade dos usuários, etc.

Para a Web, no entanto, apesar de existir uma primitiva HTML para a geração de botões, existem inúmeras outras formas comumente utilizadas para a criação de botões: utilizando-se de um bloco de texto com borda e uma cor de fundo, ou de uma imagem com o texto do botão gravado na imagem, ou a utilização de uma célula de uma tabela, etc. (CASTELEYN; GARRIG'OS; MAZ'ON, 2014). Com isto, um botão numa aplicação web se torna menos evidente para o *user-agent*. Isto dificulta e até mesmo pode impossibilitar a sua interpretação e adaptação para diferentes necessidades (por exemplo, tornar praticamente impossível ler o seu conteúdo textual caso ele tenha sido implementado inteiramente por meio de uma imagem).

Numa tentativa de solucionar uma gama desses problemas, posteriormente foram criadas diversas extensões sobre os padrões Web, como por exemplo a WAI-ARIA (W3C, 2006), que define novos atributos na linguagem HTML com o propósito de “anotar” a aplicação web com o significado e o propósito de cada elemento semanticamente significante na aplicação. Existem diversas versões destes documentos e o suporte a essas extensões vem aumentando nos navegadores e nas ferramentas de auxílio à acessibilidade externas.

No entanto, estas extensões só funcionam quando a aplicação web faz uso explícito delas, e fazer uso delas se torna uma tarefa extra e potencialmente complicada para os desenvolvedores. Observa-se que muito frequentemente os desenvolvedores acabam adiando ou mesmo deixando a utilização dessas extensões para segundo ou terceiro plano, devido aos prazos, à falta de experiência e familiaridade com tais tecnologias, ou mesmo desinteresse sobre o assunto.

Diante de problemas relativos à evolução rápida das tecnologias na Web, especialmente as que aumentam o nível de complexidade para compreensão pelos desenvolvedores web, mesmo interessados em resolver potenciais barreiras aos diversos usuários, os desenvolvedores muitas vezes encontram dificuldades para conseguir analisar as características de acessibilidade de suas páginas. Faltam-lhes conhecimentos e informação tanto sobre como determinar se as características implementadas por eles nos *website* são suficientes para serem utilizáveis por usuários cegos, bem como sobre como encontrar e implementar melhorias a fim de ampliar adequadamente a acessibilidade.

Existem diversos mecanismos que auxiliam na análise e correção de questões de acessibilidade de páginas web. Muitos deles são disponibilizados por meio de ferramentas automatizadas (IVORY; HEARST, 2001). Porém, estas ferramentas muitas vezes não são capazes de demonstrar os reais problemas de acessibilidade, focando nos problemas técnicos de implementação, sem serem capazes de associar a lacuna técnica com o efeito final que um usuário deficiente irá encontrar. Assim, os desenvolvedores têm dificuldades para interpretar e compreender as barreiras e problemas reais enfrentados por usuários com deficiências, ao visitarem sua página.

Nesse contexto, este trabalho científico buscou, portanto, responder à seguinte questão de pesquisa:

**Questão de Pesquisa:** *Como auxiliar desenvolvedores web na busca por barreiras de aces-*

*bilidade, que são encontradas por pessoas com deficiência visual que usam leitores de tela?*

## 1.2 Objetivos

O presente trabalho teve por objetivo propor uma abordagem para auxiliar desenvolvedores Web a identificarem barreiras de acessibilidade enfrentadas por usuários que utilizam leitores de tela. Assim, a abordagem proposta neste trabalho de mestrado foi concebida na forma de um simulador de leitor de tela, para ser usado por desenvolvedores web, evidenciando a experiência de um usuário com deficiência visual. Tal simulação tem o potencial de identificar problemas reais de acessibilidade enfrentados por usuários com deficiência visual. Assim, uma ferramenta foi desenvolvida, para demonstrar a viabilidade da abordagem proposta.

## 1.3 Organização

Esta dissertação está organizada da seguinte forma: no [Capítulo 2](#) são apresentados os principais conceitos estudados na literatura científica relacionados à área de acessibilidade na Web e Tecnologia Assistiva. O [Capítulo 3](#) apresenta uma discussão sobre os principais trabalhos relacionados publicados na literatura que abordam propostas para auxiliar o desenvolvimento de páginas web acessíveis para pessoas com deficiência visual, que utilizam leitores de tela como Tecnologia Assistiva. No [Capítulo 4](#), apresenta-se de forma detalhada os protótipos desenvolvidos durante este trabalho de mestrado. No [Capítulo 5](#) apresenta-se a metodologia de avaliação utilizada, bem como seus resultados e discussões sobre os resultados. O documento se encerra com as conclusões no [Capítulo 6](#).

---

# TECNOLOGIA ASSISTIVA E ACESSIBILIDADE

---

A acessibilidade, como um conceito geral, se refere à capacidade de produtos, meios, serviços e dispositivos em alcançar o uso por pessoas com deficiência. Tal uso pode se dar de forma direta ou indireta, com apoio de recurso de Tecnologia Assistiva. Tecnologia Assistiva é um termo ainda novo, utilizado para identificar todo o arsenal de Recursos e Serviços que contribuem para proporcionar ou ampliar habilidades funcionais de pessoas com deficiência e consequentemente promover Vida Independente e Inclusão<sup>1</sup>.

A acessibilidade não deve ser confundida com usabilidade (PETRIE; KHEIR, 2007). Usabilidade refere-se às características inerentes de um produto a ser usado por um usuário qualquer, englobando critérios de eficiência, facilidade, prática e satisfação (MATERA; RIZZO; CARUGHI, 2006). Para Thatcher, Waddell e Burks (2003), a acessibilidade é um subconjunto de usabilidade, em que os problemas de acessibilidade são casos particulares de problemas de usabilidade.

Acessibilidade está também relacionada com o conceito de “usabilidade universal” (SHNEIDERMAN, 2000; SHNEIDERMAN, 2002), termo criado por Shneiderman (2000) para englobar tanto a acessibilidade quanto a usabilidade. A usabilidade universal diz respeito à criação e ao oferecimento de produtos e serviços para a maior gama de pessoas possível, independente de possuírem ou não necessidades especiais.

## 2.1 Tecnologia Assistiva

Tecnologia Assistiva é um termo relativamente novo, que carrega várias definições ligeiramente diferentes, por ser definida independentemente por diversas portarias governamentais, geralmente de áreas encarregadas em tratar direitos humanos. Bersch (2017) apresenta di-

---

<sup>1</sup> <http://www.assistiva.com.br/tassistiva.html>

versas dessas definições, e propõe uma visão geral em que Tecnologia Assistiva diz respeito a todo o arsenal de recursos e serviços que contribuem para proporcionar ou ampliar habilidades funcionais de pessoas com deficiência.

Em particular, no Brasil, a definição de Tecnologia Assistiva é dada pela Secretaria Especial dos Direitos da Pessoa com Deficiência (SDH/PR, 2009):

“Tecnologia Assistiva é uma área do conhecimento, de característica interdisciplinar, que engloba produtos, recursos, metodologias, estratégias, práticas e serviços que objetivam promover a funcionalidade, relacionada à atividade e participação, de pessoas com deficiência, incapacidades ou mobilidade reduzida, visando sua autonomia, independência, qualidade de vida e inclusão social”.

Existe uma diversidade de produtos de Tecnologia Assistiva destinados a usuários com deficiência visual, parcial ou completa. Alguns exemplos destes são Displays de Braille, Lupas Virtuais, Temas de Alto-Contraste, Leitores de Tela, entre outros. Na seção a seguir, são apresentados os detalhes de Leitores de Tela.

## 2.2 Leitores de Tela

Os leitores de tela são ferramentas que auxiliam usuários com deficiência visual a utilizarem as interfaces gráficas de um *software*. Um leitor de tela incorpora fundamentalmente um sistema de narração de texto, que emite esta narração de modo audível. No entanto, ele vai além deste sistema, sendo capaz de interpretar e interagir com a interface gráfica do programa, descrevendo-a para o usuário e ajudando-o na sua utilização (AFB, 2017).

Um leitor de tela tipicamente consegue descrever os elementos fundamentais de uma interface gráfica de um software, percorrer e colocar foco em cada elemento, e indicar ao usuário cego qual ação ele pode realizar no elemento atual. Por exemplo, um leitor de tela descreve para a usuário a existência de botões como "Abrir", "Fechar", "Avançar", "Voltar"; elementos de interação como caixas de texto, caixas de seleção, menu de opções, etc; bem como os próprios textos não-interativos que fazem parte do conteúdo sendo exibido pelo programa em utilização.

Diversos materiais podem ser encontrados na Internet para o aprendizado de uso destas ferramentas. Por exemplo, o Centro Tecnológico para Acessibilidade <sup>2</sup> do Instituto Federal Rio Grande do Sul criou um manual de utilização do leitor de tela NVDA <sup>3</sup> (CTA, 2014), um leitor gratuito de tela gratuito e de código aberto. A partir dos estudos deste manual, é possível compreender as funcionalidades básicas e também algumas mais avançadas dos leitores de tela em geral, que são:

---

<sup>2</sup> <http://cta.ifrs.edu.br/>

<sup>3</sup> <https://www.nvaccess.org/>

- Navegação na página com base nos tipos de elementos: formulários, botões, cabeçalhos, tabelas, etc.
- Grande quantidade de teclas de atalho disponíveis para suas funções
- Controle dos parâmetros da sintetização de voz: volume, timbre, velocidade de fala, etc.

Apesar das capacidades técnicas dos leitores de tela, a liberdade de forma presente na criação de uma página web gera desafios significativos para a questão de sua acessibilidade. Observa-se uma falta de padronização de conceitos da interface de usuário, como por exemplo um site que se utiliza de menus para a navegação entre áreas, enquanto outro que faz uso de *sidebars* para o mesmo propósito.

Esta falta de padronização faz cada página ser um desafio individual para um usuário com necessidades especiais, que precisa conseguir contextualizar, entender e aprender a navegá-la. Em um trabalho intitulado “*Como é a Web se você não pode vê-la*”, [Asakawa \(2005\)](#) discute essa questão, mostrando as dificuldades reportadas por usuários com deficiência visual, e explicando quais erros (técnicos ou não) de criação daquela página levam a estes problemas.

Muitas vezes, entretanto, o desenvolvedor web não se preocupa ou não possui domínio técnico para realizar uma implementação de qualidade para atender essas demandas, o que prejudica significativamente a experiência para usuários com deficiências.

## 2.3 Acessibilidade na Web

Acessibilidade na Web refere-se a permitir a percepção, o entendimento, a navegação e a interação com a Web, por todas as pessoas, que livres de barreiras (ou com barreiras reduzidas), ainda que contando com uso de recursos de Tecnologia Assistiva, também possam utilizar ou contribuir com a geração de conteúdo (Web 2.0) e desenvolvimento de sistemas web ([RUTTER et al., 2006](#); [GRILLO; FORTES, 2014](#)). Portanto, em termos gerais, a definição de acessibilidade na Web visa garantir o uso da Web, principalmente, por pessoas com deficiência.

De acordo com a *Web Accessibility Initiative* (WAI), organização criada pelo W3C que tem como missão definir princípios e regras de *design* e desenvolvimento de sites que sejam acessíveis a pessoas com deficiências ([LUCCA; FASOLINO; TRAMONTANA, 2005](#)), a acessibilidade na Web engloba as questões relacionadas com o modo como todas as deficiências, seja visual, auditiva, cognitiva, da fala, física ou neurológica, afetam a Web ([W3C, 2005](#)). As ideias e conceitos relacionados ao tema estendem-se também às pessoas idosas, que mesmo não sendo classificadas como pessoas com deficiência, em geral apresentam limitações que dificultam sua interação com a Web. Esses conceitos, portanto, podem beneficiar todas as pessoas.

Assim, considerando a importância da acessibilidade na Web em função da crescente necessidade por inclusão digital, vários países criaram leis para garantir que as informações

disponíveis na Web sejam acessíveis. A “Section 508” por exemplo, em vigor desde 1998 nos Estados Unidos, determina que a tecnologia eletrônica e de informação dos órgãos federais seja acessível às pessoas com deficiência (FREIRE, 2008; TANGARIFE; MONT’ALVÃO, 2005b).

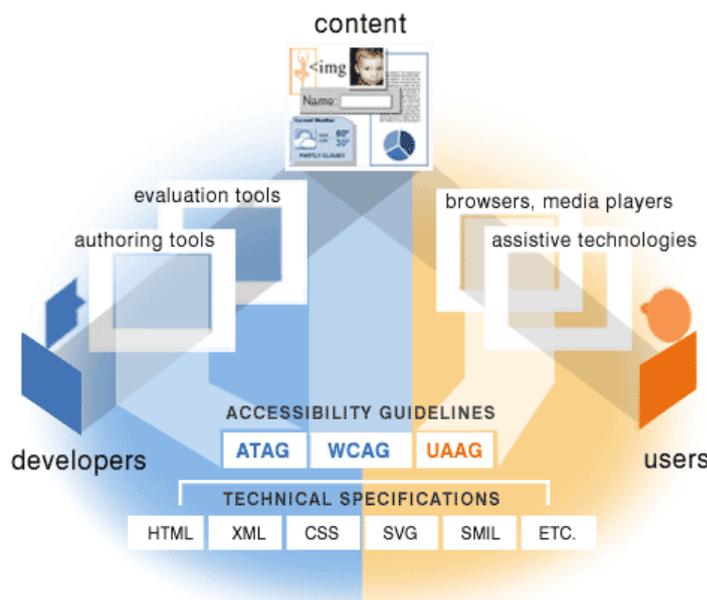
No continente europeu, Portugal foi o primeiro país a regulamentar a adoção de regras de acessibilidade na concepção da informação disponível na Web pela administração pública em 1999. Esta iniciativa foi depois estendida para outros 15 países participantes da União Europeia (TANGARIFE; MONT’ALVÃO, 2005a; GODINHO *et al.*, 2004).

O Brasil também possui normas regulamentadoras que visam garantir a acessibilidade na Web. Uma dessas normas brasileiras está presente no Decreto N° 6.949, de 25 de agosto de 2009, que institucionalizou o Modelo de Acessibilidade em Governo Eletrônico (eMAG) no âmbito do Sistema de Administração dos Recursos de Tecnologia da Informação.

Apesar desses documentos apresentarem particularidades, uma vez que são elaborados para atenderem às necessidades inerentes a cada país, todos estão fundamentados nas *guidelines* propostas pelo W3C (TANGARIFE, 2007; NICÁCIO, 2010), sendo a WCAG<sup>4</sup> uma das referências de maior importância quanto às questões de acessibilidade na Web (KELLY *et al.*, 2005; FREIRE; GOULARTE; FORTES, 2007).

Na Figura 1 são apresentadas as três principais recomendações do W3C, na forma de documentos que contêm as *guidelines* (diretrizes) de acessibilidade e quais áreas ou componentes da Web elas abordam.

Figura 1 – Relação entre as recomendações do W3C e seus componentes de atuação



Fonte: W3C (2005).

<sup>4</sup> Web Content Accessibility Guidelines (WCAG)

O documento *Authoring Tool Accessibility Guidelines* (ATAG) é voltado para fabricantes de ferramentas de criação de conteúdo, pois fornece orientações para implementação de funcionalidades nos produtos desenvolvidos, que ajudem na criação de conteúdo obedecendo as recomendações descritas na WCAG (W3C, 2000).

Por sua vez, o documento *Web Content Accessibility Guidelines* (WCAG) é voltado para a criação de conteúdo web, visto que orienta como tornar o conteúdo de sites mais acessível às pessoas com deficiência. Atualmente, o documento possui duas versões (W3C, 2012). A WCAG 1.0 é composta de recomendações gerais e pontos de verificação específicos que são divididos em 3 níveis de prioridade. Alguns exemplos de exigências são: fornecer equivalentes relativas à imagem, fornecer informações do contexto e fornecer mecanismos de navegação. Ao passo que a WCAG 2.0 é uma atualização da WCAG 1.0 para abranger tecnologias mais avançadas da Web 2.0.

O terceiro documento, *User Agent Accessibility Guidelines* (UAAG), busca definir padrões e recomendações para o desenvolvimento e funcionamento dos softwares pelos quais se acessa a Web, bem como outras tecnologias assistivas. Em suma, explica como desenvolver navegadores e reprodutores multimídia acessíveis para pessoas com deficiência (W3C, 2002).

Esses três documentos de recomendações juntas fornecem soluções complementares, resultando em acessibilidade na Web tanto quanto possível. Entretanto, para efeito direto neste trabalho, as diretrizes da WCAG receberam maior atenção, por serem as diretrizes que norteiam o desenvolvimento de aplicações acessíveis, foco desta pesquisa.

Segundo Tangarife (2007), a adoção das *guidelines* de acessibilidade elaboradas pelo W3C, por meio da WAI, é vantajosa para qualquer usuário. Desta forma, a sua adoção pela comunidade de desenvolvimento, de forma universal, representa um passo importante na busca por tornar a Web mais acessível.

## 2.4 Acessibilidade na Web para pessoas com deficiência visual

As dificuldades encontradas por um usuário durante a utilização de um *website* variam de pessoa para pessoa, sendo que diversos problemas podem contribuir ou não para uma experiência bem sucedida. Segundo Petrie e Kheir (2007), certas características de uma página web podem afetar usuários com uma certa deficiência e serem totalmente transparentes para outros usuários. Um exemplo seria um diagrama de cores que possa afetar usuários daltônicos, porém não usuários com visão normal. Outros problemas, como falhas de usabilidade no *website*, podem afetar todos usuários, porém de maneiras diferentes. Combinadas, essas falhas afetam de forma particularmente significativa os usuários com deficiência, podendo inclusive causar uma inutilização ou utilização significativamente deteriorada da página.

Desta forma, é importante entender as dificuldades encontradas por cada grupo de usuário, a fim de corrigi-las e eliminá-las. Um grupo importante de usuários com necessidades especiais são os usuários com deficiência visual. Este grupo é importante por incorporar um grande número de usuários, que possuem tal deficiência oriunda de diversas fontes: usuários idosos, usuários com cegueira parcial ou completa, de nascença ou adquirida, etc.

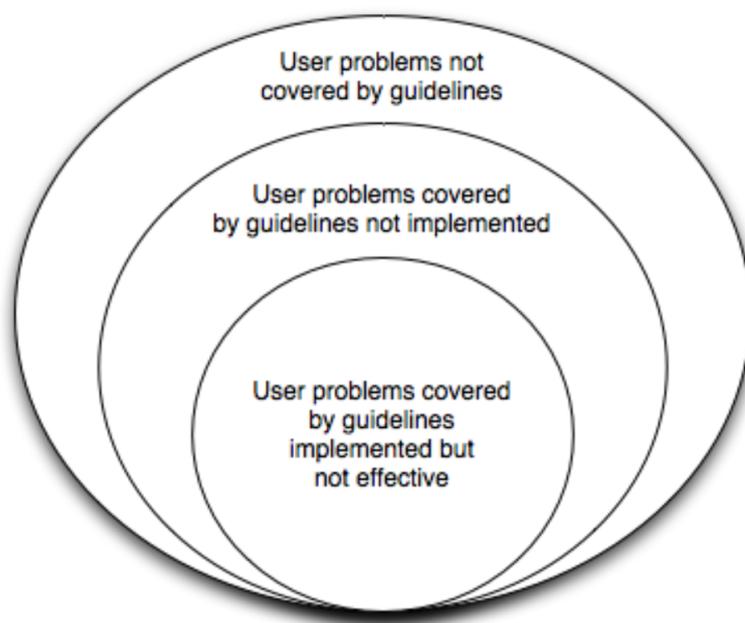
Bigham, Lin e Savage (2017) discute os efeitos sentidos por usuários com deficiência visual ao se deparar com os problemas de acessibilidade na Web. Um dos efeitos mais marcantes é o que os autores chamaram de "não saber o que não sabe", ou, de forma mais clara, não saber o que está presente e o que está faltando em uma página. Por exemplo, ao enfrentar uma situação na qual tal usuário não consegue encontrar uma informação que ele busca, este usuário não consegue deduzir facilmente se o problema é que esta informação realmente não existe, ou ele não foi capaz de achá-la.

O resultado é que ele, muitas vezes, perde mais tempo em busca desta informação, ao mesmo passo que um usuário vidente consegue mais claramente concluir que tal informação não existe. Ao se depararem com este problema, os usuários com deficiência acabam, inclusive, culpando-se a si mesmos (e suas habilidades técnicas) por não terem conseguido.

Entretanto, ao obterem mais experiência, estes usuários aprendem a diagnosticar e a atribuir o erro à uma falha estrutural da página, ou de implementação incorreta de atributos de acessibilidade.

Um outro problema é que as *guidelines* de acessibilidade, mesmo se implementadas, muitas vezes são insuficientes ou inefetivas em garantir um acesso adequado em uma página. Power *et al.* (2012) apresenta um estudo de problemas reportados por usuários com deficiência visual, e verificou-se que apenas 50,4% dos problemas reportados podiam ser abrangidos pelos critérios de sucesso da WCAG 2.0. Neste trabalho, os autores trazem um diagrama (Figura 2) classificando em 3 categorias os tipos de problemas encontrados em seu estudo.

Figura 2 – Conjunto de problemas encontrados por usuários com deficiência visual



Fonte: Power *et al.* (2012).

O primeiro plano da [Figura 2](#) apresenta o círculo mais interno, o qual representa a categoria de problemas cobertos por *guidelines* e implementados na página, mas de forma inefetiva. Um exemplo deste problema são os textos alternativos de imagens que mesmo presentes, não as descrevem adequadamente ([BIGHAM, 2007](#)). Em um segundo plano, englobando a primeira categoria, são problemas cobertos por *guidelines*, mas que simplesmente não foram seguidas pelo desenvolvedor. Por final, englobando essas duas categorias, estão os problemas que nem sequer estão cobertos por *guidelines*.

Uma observação importante é que a forma de uso de uma página, comparada entre um usuário normal e um usuário com deficiência visual, são completamente diferentes ([WILLIAMS \*et al.\*, 2014](#)), e não podem ser comparadas ([SAVIDIS; STEPHANIDIS, 1998](#)). Desde as estratégias de navegação, tanto quanto as diferenças entre ferramentas utilizadas (mouse, teclado, softwares auxiliares como leitores de tela, etc.). Assim, há uma distância muito grande entre experiências, que dificultam a compreensão ou até mesmo o conhecimento de tal distância por parte de usuários reais e dos autores de páginas.

Esta diferença é encontrada não somente no âmbito de experiências virtuais e de tecnologia, mas também no mundo concreto. Estudos mostram que a falta de um sentido (por exemplo, a visão) desenvolve habilidades e percepções amplificadas em outros sentidos, sendo que pessoas cegas desenvolvem estratégias de navegação diferentes de pessoas “videntes”<sup>5</sup>. Estas

<sup>5</sup> Usuários cegos comumente se referem às pessoas que enxergam como **videntes**.

estratégias se mostram até mesmo conflitantes, como por exemplo um cego que caminha pela calçada próximo à parede, por utilizá-la como objeto de referência para o seu mapeamento do espaço (MAIDENBAUM *et al.*, 2014).

Diversos trabalhos na literatura se dedicaram a observar e a analisar as técnicas avançadas empregadas por usuários deficientes ao navegarem na Web. Borodin *et al.* (2010) descreve o uso do leitor de tela, e como esses usuários aprendem a superar problemas ao agilizar suas tarefas ao adquirirem conhecimento prévio do *site* ou da navegação em geral.

Um exemplo apresentado é a necessidade de uso frequente de um botão que é de difícil acesso em uma página de um *webmail*. Para acessá-lo mais facilmente, um usuário encontrou um outro elemento próximo a ela que tinha acesso mais fácil, e memorizou a quantos elementos atrás este botão se encontrava, para então utilizar os comandos de percorrer itens em ordem reversa para atingir esse botão. Há relatos também de usuários que aprenderam a ler o código-fonte da página em busca de informações que se mostravam mais difíceis de serem encontradas com o *layout* visual da página.

## 2.5 Problemas de acessibilidade para pessoas cegas

A navegação na página é um dos obstáculos mais marcantes (GERALDO; FORTES, 2013), mas existem inúmeros outros, de diversas categorias diferentes. Uma categorização proposta por Geraldo e Fortes (2013) consiste em três classes de problemas que se apresentam como potenciais barreiras para pessoas cegas, diante do acesso às páginas web:

- Problemas técnicos
- Problemas semânticos ou cognitivos
- Problemas na navegação

A seguir estes problemas são descritos em mais detalhes.

### ***Problemas técnicos***

Esta categoria de problemas se refere aos aspectos relacionados com questões tecnológicas adotadas na implementação das páginas. São eles:

**Atualização automática da página** - A página web é automaticamente atualizada após um certo tempo decorrido (BRAJNIK, 2009). Como isso não é informado ao usuário cego, ele perde totalmente o controle e foco da interação atual. Além disso, como os usuários do leitor de tela levam mais tempo para percorrer uma página, a atualização automática pode forçar esses usuários a reiniciar a operação de busca do conteúdo desejado.

**Formulários sem rótulos adequados** - A página apresenta um formulário em que seus campos apresentam labels ausentes ou irrelevantes (BRAJNIK, 2009). Isso impossibilita que o usuário cego possa interagir com o formulário. Além disso, o uso de símbolos, como o asterisco, para representar obrigatoriedade do preenchimento de um campo do formulário apresenta uma dificuldade para os

**Novas janelas** - A página contém código que abre novas janelas do navegador quando o usuário utiliza links ou botões ou após a página ter carregado. Isso afeta o foco do usuário cego, dificultado a sua navegação (BRAJNIK, 2009).

**Formulários com redirecionamento** - A página contém um formulário que seu botão de submissão leva a uma nova atualização de página. Isso faz com que o navegador exiba uma nova página e posicione o foco do usuário no início desta página (BRAJNIK, 2009).

**Componentes inacessíveis** - A página apresenta componentes inacessíveis pelo leitor de tela, por exemplo, um objeto Flash ou um menu que não pode ser utilizado pelo teclado (BRAJNIK, 2009).

**Eventos disparados apenas com uso do mouse** - A página possui funções Javascript que são invocadas apenas com a utilização de mouse, como por exemplo, on mouse over e on mouse out (BRAJNIK, 2009).

**Imagens sem descrição equivalente** - A página apresenta imagens como fonte de informação, porém não existe uma descrição textual equivalente para descrever essas imagens (BRAJNIK, 2009).

**Imagens funcionais sem texto alternativo** - A página contém imagens utilizadas como botões ou links clicáveis, porém falta um texto equivalente para descrever essas funções (BRAJNIK, 2009).

**Imagens funcionais incorporadas no plano de fundo** - A página contém imagens funcionais incorporadas ao plano de fundo do site, impossibilitando que sejam associadas descrições textuais equivalentes a essas imagens (BRAJNIK, 2009).

**Uso de frames** - O projeto da página é baseado em *frames*. Isso faz com que ação tomada pelo usuário na página possa afetar outro *frame* que está fora do foco atual do usuário do leitor de tela (BRAJNIK, 2009).

**Vídeos sem descrição** - A página contém um elemento multimídia com um vídeo ou uma animação sem uma descrição textual equivalente para os diálogos e cenas (ENCALLE *et al.*, 2011).

**Páginas sem títulos** A página não possui a *tag* TITLE, ou o conteúdo dessa *tag* não provê informação útil (BRAJNIK, 2009). Isso faz com que o usuário cego não possa induzir corretamente o que será encontrado nessa página.

**Tag de idioma não definido** - A página não define o atributo LANG corretamente na *tag* HyperText Markup Language (HTML). Os leitores de tela podem utilizar essa informação para saber em qual idioma o conteúdo deverá ser lido (FERREIRA; NUNES; SILVEIRA, 2012).

**Página sem uso de headings** - A página não utiliza as *tags* h1, h2, ..., h6 para organizar o seu conteúdo (BRAJNIK, 2009). O uso dessas tags facilita a compreensão da página pelo usuário cego.

**Janela sem controles do navegador** - A página foi iniciada em uma nova janela do navegador, porém os controles usuais do navegador, como a barra de endereço e os botões voltar, avançar e atualizar não estão aparecendo (BRAJNIK, 2009).

**Captcha** - A página utiliza um serviço de *Captcha* que faz o usuário ler um conjunto distorcido de caracteres de uma imagem e digitar estes caracteres em um formulário (FUGLERUD *et al.*, 2012). Alguns serviços de *Captcha* também fornecem áudio, mas problemas graves de usabilidade e acessibilidade também foram encontrados nesses serviços (BIGHAM; CAVENDER, 2009).

### ***Problemas semânticos ou cognitivos***

Esta categoria de problemas se refere aos aspectos relacionados com características reconhecidas pelos usuários cegos, para compreensão de seu conteúdo. São eles:

**Conteúdo alterado dinamicamente** - A página apresenta um mecanismo que atualiza a página com nova informação ou realoca a informação existente sem o reconhecimento dos usuários (BAVANI; JAAFAR; YATIM, 2010). Essa adaptação dinâmica do conteúdo pode causar desorientação nos usuários cegos (CANDAN *et al.*, 2009). Esse problema ocorre devido à falta de *feedback* para informar sobre as ações que modificaram a página (POWER *et al.*, 2012).

**Conteúdo irrelevante** - A página apresenta um conteúdo irrelevante que não pode ser evitado e está posicionado antes do conteúdo desejado, como por exemplo, um menu superior de navegação (POWER *et al.*, 2012). Isso frustra os usuários, que são forçados a percorrer esse conteúdo irrelevante a cada atualização de página (BAVANI; JAAFAR; YATIM, 2010). Como os leitores de tela utilizam uma estratégia linear para processar cada página (de cima para baixo e da esquerda para direita), é muito difícil, tedioso e frustrante ser obrigado a sempre evitar as informações necessárias até encontrar o conteúdo ou link desejado.

**Layout da página muito complexo** - A página apresenta um *layout* complexo, dificultando que a pessoa cega tenha uma visão geral da página web (BAVANI; JAAFAR; YATIM, 2010).

**Layout baseado em tabelas** - O *layout* da página é baseado em tabelas, misturando conteúdo e estrutura. Como as tabelas não podem ser linearizadas, o leitor de tela (que lê essas tabelas linha a linha) anuncia o conteúdo da página fora de ordem, consequentemente, a informação se torna confusa ou enganosa para o usuário cego (BUZZI *et al.*, 2010). As tabelas também não devem ser utilizadas para organizar componentes da página, como formulários por exemplo (BUZZI *et al.*, 2010).

**Dados em tabelas** - A página apresenta uma tabela e o seu conteúdo está organizado em colunas. Uma vez que os leitores de tela interpretam o conteúdo da tabela por linhas, ele irá anunciar fora de ordem o conteúdo desta tabela (BUZZI *et al.*, 2010). Além disso, a necessidade de memorização dos dados da tabela também gera uma barreira aos usuários cegos (FERREIRA; NUNES; SILVEIRA, 2012).

**Conteúdo inconsistente** - A página apresenta conteúdo inconsistente, como por exemplo: links quebrados, conteúdo encontrado não era o esperado pelo usuário, o conteúdo não é encontrado onde era esperado, a funcionalidade não funciona como o esperado, funcionalidade esperada não está presente e a organização do conteúdo se encontra inconsistente com as convenções da Web ou o senso comum (POWER *et al.*, 2012).

**Páginas com carregamento lento (POWER *et al.*, 2012)** - O problema ocorre pois o leitor de tela pode não fornecer o *feedback* informando que todo o conteúdo da página não terminou de ser carregado, causando desorientação ao usuário cego.

**Falta de contexto** - O conteúdo da página web apresenta falta de semântica, fazendo com que as pessoas cegas não são capazes de dar sentido ao conteúdo (POWER *et al.*, 2012). Os usuários do leitor de tela precisam construir uma representação mental da página web e das estruturas de navegação para que eles possam buscar o conteúdo de forma eficiente. Sem um monitor persistente, as pessoas cegas ficam facilmente desorientadas (ZHAO *et al.*, 2008).

**Dados que necessitam de auxílio visual** - A página contém dados que dependem de ajuda visual para o seu entendimento, como por exemplo, dados georreferenciados (ZHAO *et al.*, 2008).

**Alterações no layout da página** - O layout de uma página web existente é alterado, obrigando os usuários cegos a novamente aprender o nome de posição de todos os elementos importantes da página (FERREIRA; NUNES; SILVEIRA, 2012). Além disso, a falta de *feedback* na alteração do layout pode causar desorientação no usuário cego, que estava esperando por outro conteúdo ao voltar a página já previamente conhecida.

**Links sem separação** - A página apresenta uma sequência de links que não são separados por nenhum caractere ou símbolo. Assim, o usuário cego pode enfrentar dificuldade para entender onde termina um link e começa outro (BRAJNIK, 2009).

**Página com muitos links** - A página web apresenta muitos links, causando desorientação na pessoa cega que é obrigada a memorizá-los para eventualmente decidir qual dos links deve ser utilizado. O problema piora consideravelmente caso os links não estejam organizados em categorias significativas (BRAJNIK, 2009). Além disso, O uso do leitor de tela e do teclado para buscar por um link pode ser ineficiente em grandes documentos com muitos links ou hierarquias profundas.

**Texto Espaçado** - A página web apresenta palavras ou termos que contêm espaços extras, como por exemplo o texto “S E J A - B E M - V I N D O”, com o objetivo de conseguir um determinado efeito visual (BRAJNIK, 2009).

**Links indevidamente rotulados** - A página apresenta links inadequadamente rotulados, como por exemplo, “clique aqui”, “acesse aqui” ou “link” (BAVANI; JAAFAR; YATIM, 2010). Como o usuário cego navega pelo site por meio de links, ele não é capaz de compreender o significado destes links mal rotulados sem acessar toda a informação contextual em torno dos mesmos.

**Arte ASCII** - A página web contém texto que representa decorações em vez de palavras, como por exemplo, *smileys* (BRAJNIK, 2009).

**Dependência de cor** - A conteúdo da página web contém material (texto, imagem, fundo, vídeos) em que a cor é usada como o único meio de transporte de informação, como por exemplo, o uso de diferentes cores para links previamente acessados (BRAJNIK, 2009).

**Sobrecarga de informação** - A página apresenta uma sobrecarga de informação, que prejudica a experiência de navegação dos usuários cegos, por exemplo, muitos passos para encontrar o conteúdo (POWER *et al.*, 2012).

### ***Problemas na navegação***

Esta categoria de problemas se refere aos aspectos relacionados com as formas disponibilizadas para localização no conteúdo disponibilizado e questões sobre as alternativas de interação oferecidas. São eles:

**Menus muito complexos** - A página apresenta menus muito complexos, com muitos links, o que dificulta a navegação em um leitor de tela. Isso ocorre pois o usuário cego apresenta dificuldade em encontrar links relevantes em meio a grandes quantidades de links oralizados, resultado em desorientação (BUZZI *et al.*, 2010). Além disso, em geral não há nada explicitamente afirmando que um elemento da página é um menu. O conhecimento sobre qual elemento é um menu está implícita na sua representação visual (LUNN; HARPER; BECHHOFFER, 2011). Usuários cegos não são capazes de adivinhar as relações entre os itens primários e secundários de navegação se elas forem expressas apenas visualmente.

**Aplicações de Internet Rica** - A página apresenta um componente de *Rich Internet Application* (RIA) que foi implementado sem seguir as diretrizes de acessibilidade. RIAs estão repletas de elementos dinâmicos e interativos, tornando-se assim, difíceis de serem utilizadas com um leitor de tela.

**Retorno ao conteúdo previamente acessado** - A página não apresenta uma maneira de auxiliar os usuários de leitor de tela a retomar o conteúdo previamente acessado. Esse problema ocorre quando estes usuários necessitam voltar para buscar por alguma informação específica ou retomar orientação. Como eles necessitam ouvir novamente e reconhecer pelo menos uma parte de cada página intermediária antes de atingir o conteúdo desejado, o retorno ao conteúdo previamente acessado com o uso de leitores de tela é um processo longo e ineficiente (GHAHARI *et al.*, 2012).

**Conteúdo inacessível pelo teclado** - A página requer o uso de mouse para a sua navegação. Pessoas cegas não são capazes de utilizar uma página web que apresenta a navegação pelo mouse como única alternativa, dado que eles não conseguem ver onde clicar (FUGLERUD *et al.*, 2012).

**Falta de *feedback* e ajuda** (FERREIRA; NUNES; SILVEIRA, 2012) - A página apresenta elementos interativos que não fornecem *feedback* adequado durante o seu uso. Além disso, a página não apresenta nenhum auxílio ao usuário em caso de questionamento sobre o sistema (manual de uso, mapa do site, estado atual da interação, etc).

**Falta de recuperação de erro** - A página apresenta elementos não válidos do atual contexto e não auxilia o usuário a se recuperar de um possível erro durante a interação (FERREIRA; NUNES; SILVEIRA, 2012).

**Armadilhas para o teclado** A página contém elementos que travam o foco do usuário assim que ele os atinge com o teclado. Isso força o usuário a recarregar a página para retomar o controle do foco (BRAJNIK, 2009).

## 2.6 Considerações Finais

Neste capítulo, introduziu-se os conceitos Tecnologia Assistiva, acessibilidade e acessibilidade na Web, e listou-se quais são os principais problemas enfrentados por um usuário com deficiência visual ao navegar na Web. Esses conceitos foram estudados de modo a fundamentar o desenvolvimento proposto neste projeto de mestrado.

Existem ferramentas que auxiliam no desenvolvimento e análise de páginas de acordo com os padrões de acessibilidade mencionados. Um estudo sobre os tipos e características destas ferramentas será apresentado no [Capítulo 3](#).



---

## TRABALHOS RELACIONADOS

---

A construção de *websites* acessíveis para usuários com deficiência é uma tarefa muitas vezes difícil e custosa para desenvolvedores web. Mesmo bem intencionado, o desenvolvedor precisa ter um domínio técnico específico sobre o assunto, que frequentemente não é obtido necessariamente junto com o conhecimento sobre a criação de páginas web em geral.

Para iniciar um trabalho de introdução ou ampliação de acessibilidade a um *website*, o desenvolvedor precisa em primeira instância conhecer a existência do problema. A partir daí, ele precisa buscar diversas técnicas, *guidelines*, padrões e recomendações; entender a aplicação de cada; como implementá-las e testá-las; e os resultados que serão obtidos; para analisar o custo-benefício e a dificuldade de cada tarefa.

Assim, um conhecimento prévio do problema não se faz suficiente para uma criação bem sucedida de uma página acessível, e ferramentas que ajudam os desenvolvedores no processo de criação se demonstram fundamentais para a obtenção de resultados satisfatórios.

Um estudo foi feito sobre ferramentas existentes para o auxílio ao desenvolvimento de *websites* acessíveis. [Ivory e Hearst \(2001\)](#) propuseram uma taxonomia para classificar a automação de testes de usabilidade que também podem ser aplicados para acessibilidade, como feito por [Takagi et al. \(2003\)](#). Esta taxonomia estabeleceu 4 critérios para identificar as diversas metodologias, e gerou uma classificação final baseada na classe do método, com diversas sub-categorias em cada. Os quatro critérios são:

1. Classe do método
  - Teste, inspeção, questionamento, modelagem analítica, simulação
2. Tipo de método
  - Análise de logs, *surveys*, revisão de diretrizes, etc.
3. Tipo de automatização

- Nenhuma, fase de captura, fase de análise, fase de crítica
4. Nível de esforço
- Mínimo, desenvolvimento de modelo, uso informal, uso formal

As categorias são sub-divididas nas diversas formas em que elas podem ser executadas. Uma descrição mais detalhada, juntamente com exemplos para cada categoria, é apresentada a seguir:

**Método de teste** - Enquadram-se nesta categoria as metodologias que envolvem a participação do usuário (leigo ou experiente) como objeto principal da metodologia. Exemplos nesta categoria são métodos como a conversa com o usuário enquanto ele usa a ferramenta; o protocolo "*Think aloud*" em que o usuário é encorajado a descrever o que pensa e qual ação pretende tomar; testes guiados; testes remotos; gravações de áudio e vídeo para análise posterior; etc.

**Método de questionamento** - Este método é similar ao método de teste, porém aqui o método de análise se dá indiretamente por meio da participação do usuário, geralmente na forma de questionários, pós-entrevistas, *checklists*, auto-relatos, relatórios de observação por terceira pessoa, grupos de foco, etc.

**Método de inspeção** - Neste método enquadram-se ferramentas que analisam e validam o funcionamento de uma ferramenta por meio de um conjunto de regras ou diretrizes, e produzem como saída uma pontuação ou uma lista de problemas ou potenciais problemas. Ferramentas automáticas de validação são o exemplo típico desta categoria, mas que também podem incluir validações manuais, heurísticas, de consistência, feitas por *experts*, etc.

**Método de modelagem analítica** - Aqui enquadram-se metodologias que fazem, por exemplo, modelagem do uso da ferramenta para a predição de tempo e complexidade que um usuário levaria para utilizá-la; *design reviews* de complexidade de tarefas; análise de carga cognitiva; e também modelos programáveis que estabelecem passos ou heurísticas para um programa automatizado agir como se fosse um usuário real.

**Método de simulação** - São métodos que pretendem imitar as condições de uso ou simular o uso da ferramenta por um usuário típico (ou um usuário específico a ser estudado). Isto inclui simular a experiência que um usuário iniciante terá ao utilizar o objeto de estudo, seus passos de navegação e de descoberta, etc.

Com esta classificação em mente, foi realizada uma pesquisa sobre as ferramentas existentes que tratam especificamente de análise de acessibilidade em páginas da Web. Muitas

das ferramentas encontradas utilizam a abordagem do método de inspeção, devido à grande orientação da área rumo às diretrizes de acessibilidade, discutidas no Capítulo 2.

Além das ferramentas validadoras, dedicou-se particular esforço para buscar na literatura trabalhos sobre ferramentas de simulação, por serem a motivação de estudo deste trabalho. A seguir apresentamos com mais detalhes as características sobre estes dois tipos de ferramentas.

## 3.1 Validadores

Validadores são ferramentas de auxílio de desenvolvimento que se baseiam na análise do código fonte da página, possivelmente com seus arquivos-suporte (por exemplo, arquivos de estilo CSS), e verificam, sobre este código, um conjunto de regras fixos e pré-estabelecidos. Ao término desta verificação, os validadores emitem em sua saída uma lista de problemas identificados, ocasionalmente com sugestões de correção, explicações técnicas ou não-técnicas sobre o problema, graus de importância e ordem de prioridades.

As ferramentas validadoras se diferenciam em sua capacidade de avaliação de regras (de variados níveis de dificuldades) e forma de execução, bem como no próprio conjunto de regras que cada ferramenta cobre. Desta forma, um estudo eficiente sobre a acessibilidade com tais ferramentas geralmente envolvem a utilização conjunta de diversas ferramentas diferentes, de forma à englobar o maior conjunto de regras e detecção de problemas possíveis.

As regras das ferramentas de validação são definidas internamente, e geralmente são derivadas dos diversos padrões e recomendações para a criação de sites acessíveis, como o WCAG 1.0, WCAG 2.0, Section 508, etc. É possível encontrar na web e na literatura inúmeras destas ferramentas. O *website* do W3C sobre ferramentas de validação de acessibilidade<sup>1</sup>, por exemplo, lista atualmente 69 ferramentas disponíveis.

## 3.2 Simuladores

Diferentemente da análise dos problemas técnicos utilizando-se de ferramentas de validação, uma segunda abordagem existente em ferramentas para auxílio à acessibilidade consiste na detecção e interpretação de obstáculos práticos que serão enfrentados por usuários com necessidades especiais ao visitar uma página web.

Segundo Takagi *et al.* (2003), esta abordagem é importante e promissora, pois usuários com deficiência acessam a Web por meio de ambientes muito diferentes do que usuários comuns, e desenvolvem estratégias diferentes de uso e navegação. Dessa forma, ferramentas que simulam estas diferenças podem revelar problemas ou combinações de problemas que podem não ser revelados com a adoção de outros métodos.

<sup>1</sup> <https://www.w3.org/WAI/ER/tools/>

A seguir apresentamos algumas destas ferramentas.

### 3.2.1 *Vischeck*

A ferramenta *Vischeck*<sup>2</sup> (DOUGHERTY; WADE, 2006) tem como principal objetivo simular a visão de uma pessoa daltônica sobre imagens ou páginas na Web. Segundo os autores, uma em cada 20 pessoas possuem algum tipo de daltonismo, e a ferramenta é capaz de simular vários tipos de daltonismo por exemplo, e exibir os resultados lado a lado para comparação.

Esta ferramenta simula um modelo computacional completo de visão, separado em três estágios. O primeiro estágio diz respeito às propriedades físicas da meio (visores CRT<sup>3</sup>, LCD<sup>4</sup> ou uma impressão CMYK<sup>5</sup> em papel), além de propriedades sobre iluminação ambiente e fatores fisiológicos que afetam a luz que é captada pelos olhos, como miopia ou hipermetropia.

O segundo estágio modela a transformação da informação ótica na retina para a representação neural da imagem no nervo ótico. É neste estágio que anomalias como o daltonismo são modelados e introduzidos.

O último estágio remete à renderização da informação ótica pelo córtex visual na percepção visual obtida pelas pessoas, incluindo fatores sobre como padrões espaciais, movimento e cores são interpretados pelo cérebro humano.

Ao final do processo, uma imagem como a exemplificada na Figura 3 é exibida. A ferramenta pode ser utilizada online ou localmente, e pode ser utilizada sobre uma imagem única, um conjunto de imagens, ou uma página Web.

Figura 3 – Ferramenta *Vischeck*



Ilustração de uma imagem original e processada para daltonismo vermelho/verde e amarelo/azul.

<sup>2</sup> <https://www.vischeck.com>

<sup>3</sup> *Cathode Ray Tube* (CRT)

<sup>4</sup> *Liquid-Crystal Display* (LCD)

<sup>5</sup> *Cyan, Magenta, Yellow, Key* (CMYK)

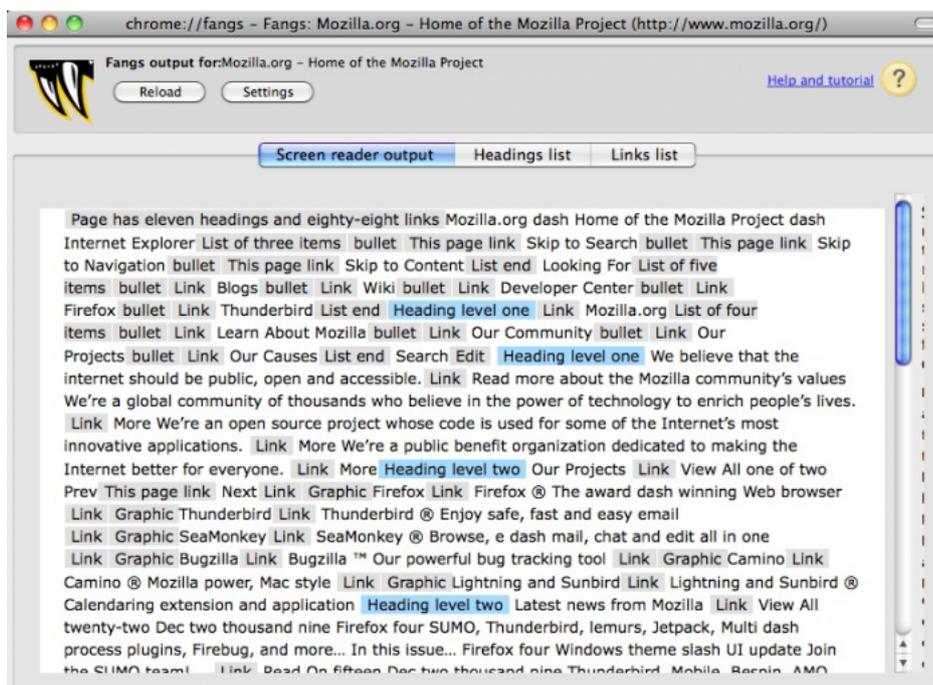
### 3.2.2 Fangs Screen Reader Emulator

O Fangs Screen Reader Emulator<sup>6</sup> (KRANTZ, 2004) é uma extensão para o navegador Firefox<sup>7</sup> que se propõe a apresentar para o usuário a mesma informação que um leitor de tela apresentaria de uma página Web para um usuário cego.

Neste emulador, ilustrado na Figura 4, toda a formatação da página Web é removida, e apenas o seu conteúdo textual é exibido, de forma corrida. Além disso, informações extras como rótulos de imagens e descrições de links são inseridas nos casos em que um leitor de tela apresentaria estas informações extras.

Entretanto, este emulador não permite que o usuário interaja com a página original, uma vez que ele obtém o conteúdo da página original, porém apresenta os resultados num contexto diferente em uma nova janela.

Figura 4 – Ferramenta Fangs Screen Reader Emulator



Fonte – Ferramenta Fangs ao ser utilizada sobre uma página Web, exibindo apenas a informação textual presente na página.

### 3.2.3 WebAIM Simulation

O WebAIM (WEBAIM, 1999) é um projeto comercial que possui diversas ferramentas de auxílio e análise de acessibilidade de páginas Web. WAVE<sup>8</sup>, um dos seus produtos, é uma

<sup>6</sup> <https://addons.mozilla.org/en-US/firefox/addon/fangs-screen-reader-emulator/>

<sup>7</sup> <http://www.mozilla.org/firefox>

<sup>8</sup> *Web Accessibility Evaluation Tool (WAVE)*

ferramenta de análise por inspeção, que checa regras de acessibilidade e compatibilidade com padrões e recursos dos leitores de tela existentes.

Na página do WebAIM é disponibilizada também uma simulação de um leitor de tela<sup>9</sup>, que tem como objetivo introduzir a usuários comuns as barreiras encontradas por usuários de leitores de tela. Esta ferramenta se propõe a simular a experiência de um usuário cego por meio de diversas características comuns a usuários de leitores de tela, como a impossibilidade de utilizar o mouse, e o uso extensivo de teclas de atalho.

A simulação ocorre sobre uma página Web hipotética criada especificamente para o assunto, imitando uma página de uma universidade. A ferramenta propõe, então, que o usuário realize diversas tarefas sobre a página, como por exemplo encontrar um número de telefone desta universidade, ou encontrar informações relevantes à matrícula em uma determinada disciplina oferecida.

Outra simulação existente na página do WebAIM se trata de uma simulação de problemas de visão, como glaucomas e cataratas. Uma imagem estática de um website é apresentada, com filtros que simulam tais problemas, e ao usuário é dada a tarefa de buscar informações nesta página, como demonstrado na Figura 5.

Outras simulações similares foram encontradas na web, como ilustrada na Figura 6.

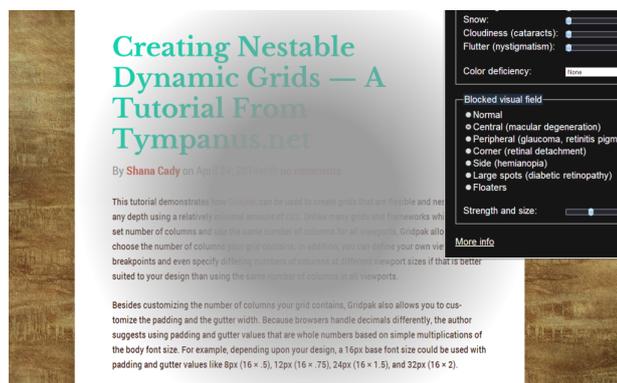
Figura 5 – Ferramenta WebAIM LowVision Simulator



Fonte – A ferramenta de simulação de problemas de visão, como catarata, do projeto WebAIM

<sup>9</sup> <http://webaim.org/simulations/screenreader>

Figura 6 – Ferramenta NoCoffee - Vision Simulation



Fonte – Ilustração da ferramenta NoCoffee (NOCOFFEE, 2013)

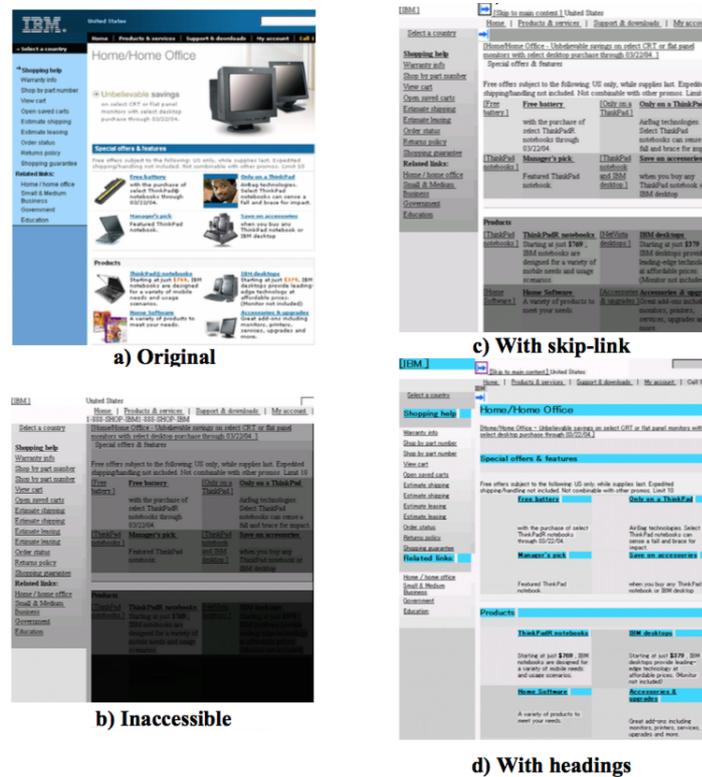
### 3.2.4 Accessibility Designer - aDesigner

A ferramenta aDesigner (TAKAGI *et al.*, 2003) se propõe a possibilitar análises de características de navegabilidade importantes para um website, como o tempo para alcance de um determinado objeto, a complexidade e dificuldades de alcance para o conteúdo principal da página, e as qualidades de navegabilidade dos links presentes na página.

Ao ser acionada, a ferramenta visita um website e extrai informações do seu código HTML e DOM, e posteriormente plota a página com seus elementos destacados de acordo com a dificuldade ou não de serem atingidos. Isto proporciona um efeito para inspeção visual muito rápida para ajudar na identificação de partes corretas e partes possivelmente problemáticas na página em estudo.

Na Figura 7 podemos ver um comparativo de uma página em seu formato original (a) e com seu conteúdo original analisado por aDesigner (b). Além disso, as Figuras (c) e (d) ilustram a página analisada novamente pela ferramenta, após serem feitas algumas modificações com o intuito de melhorar a acessibilidade sobre ela.

Figura 7 – Ferramenta Accessibility Designer (aDesigner)



Fonte – Ilustração da ferramenta aDesigner com a imagem da página original (a) e com a renderização proporcionada pela ferramenta (b), (c), (d).

### 3.3 Considerações Finais

Neste capítulo foi apresentada uma visão geral sobre as ferramentas existentes para análise de usabilidade e, principalmente, acessibilidade na Web. Em especial, foram estudadas as ferramentas que tratam de simulações sobre as experiências de usuários cegos ou com deficiências visuais.

Apresentou-se também uma taxonomia proposta para a classificação de tais ferramentas, exemplificando as diversas categorias propostas desta taxonomia. Foram descritas duas categorias, relevantes para o presente estudo, por serem mais utilizadas na área em questão (ferramentas de validação na área de acessibilidade), e por apresentarem semelhanças com a ferramenta proposta por este trabalho (ferramenta de simulação).

Assim, contextualizamos o estado da arte atual e as diversas abordagens existentes e testadas para se tratar do problema aqui estudado. No próximo capítulo é apresentada a proposta concreta do simulador a ser prototipado, os critérios de implementação, testes e avaliação a serem utilizados, bem como os objetivos a serem atingidos com este trabalho.

---

# SRS - UM SIMULADOR DE LEITOR DE TELA PARA DESENVOLVEDORES WEB

---

Neste capítulo, apresentamos o projeto desenvolvido como prova de conceito para a pesquisa realizada no decorrer deste projeto de mestrado. Foi construído um protótipo de ferramenta com finalidade de simular, para um usuário interessado (em geral, um desenvolvedor *Web* analisando o seu próprio *website*), a experiência vivenciada por um usuário com deficiência visual ao navegar no seu site utilizando-se de um leitor de tela.

A ferramenta foi nomeada como *SRS - Screen Reader Simulator*, que em Português pode ser traduzido como Simulador de Leitor de Tela.

O desenvolvimento deste protótipo teve o objetivo de viabilizar a realização de análises sobre o potencial de uma ferramenta de simulação de leitor de tela e, assim, auxiliar desenvolvedores web não cegos a realizarem a atividade de avaliação de acessibilidade de uma página web. Em especial, a partir da ferramenta SRS implementada, buscamos verificar as lacunas que outras abordagens não alcançam ou que sejam limitadas para identificação das barreiras de acessibilidade, as quais geralmente são vivenciadas por usuários com deficiência visual utilizando leitor de tela.

Nas seções a seguir, descrevemos o contexto de criação da ferramenta SRS ([Seção 4.1](#)), suas principais funcionalidades ([Seção 4.2](#)), bem como suas etapas de desenvolvimento e evolução ([Seção 4.3](#) e [Seção 4.4](#)).

## 4.1 Contexto do Simulador SRS

O processo de criação de páginas web envolve o conhecimento e o domínio técnico de diversas tecnologias e padrões. Embora criar uma página simples, com conteúdo puramente textual, seja uma tarefa relativamente simples, a criação de páginas mais complexas não é de

forma alguma uma tarefa trivial.

Durante o processo de criação, a preocupação com a acessibilidade geralmente fica em segundo plano, devido a limitações de tempo, de orçamento ou mesmo de conhecimento sobre o assunto. Takagi *et al.* (2003) mencionam que a criação bem sucedida de uma página acessível requer um empenho dedicado, desde os passos iniciais, para observar com atenção quais os elementos que requerem aspectos tecnológicos de soluções alternativas, sendo fundamental a estruturação da página de uma forma que facilite sua implementação e sua adequação para atender às diversas diretrizes necessárias para assegurar acessibilidade.

Entretanto, as ferramentas de auxílio à avaliação de acessibilidade existentes, geralmente, produzem resultados muito complexos ou muito exaustivos, sendo difícil verificar a importância relativa de cada problema na experiência do usuário final. Por exemplo, uma ferramenta de validação pode indicar a quantidade de links na página sem uma descrição textual apropriada; porém, tal detecção não é capaz de interpretar ou contextualizar a importância relativa de cada link problemático diante do conteúdo completo da página.

No cenário real de desenvolvimento de *websites*, no qual orçamentos e esforços são recursos limitados para os desenvolvedores, nem sempre é possível identificar quais medidas, se implementadas, terão melhor retorno na melhoria da acessibilidade de uma página.

Desta forma, faz-se necessário um auxílio na forma de uma ferramenta que possa ampliar as informações relevantes durante avaliação de acessibilidade em páginas web. Assim, a proposta nesse auxílio é a de expor não os problemas técnicos por trás da página (erros ou omissões no HTML, por exemplo), mas sim os problemas de experiência e usabilidade vivenciados por parte de usuários com deficiência. Takagi *et al.* (2003) avaliam a simulação de necessidades especiais como uma das mais promissoras maneiras de auxílio na criação de conteúdo acessível, tendo em vista que a Web é acessada e utilizada de formas totalmente diferentes por usuários com deficiências.

Neste trabalho, foram concentrados esforços em construir uma ferramenta que pudesse auxiliar na criação e avaliação de acessibilidade para pessoas com deficiência visual. Como uma das principais ferramentas utilizadas por estes usuários são os leitores de tela, a ferramenta aqui proposta (SRS - *Screen Reader Simulator*) pretende, então, simular a experiência de um leitor de tela para um usuário “vidente”<sup>1</sup>.

Espera-se que a simulação desta experiência possa proporcionar, aos seus usuários desenvolvedores Web, obstáculos e dificuldades semelhantes aos que são encontrados na mesma página ao ser visitada por usuários com deficiência visual, considerando-se que esses usuários utilizam um leitor de tela.

A ferramenta SRS visa simular a experiência do leitor de tela em diversos patamares.

---

<sup>1</sup> O termo vidente é usualmente adotado para qualificar as pessoas que não possuem deficiência visual (FRANÇA-FREITAS; GIL, 2012)

Assim, a interação com SRS deve proporcionar a sensação dos aspectos característicos do cenário que usuários cegos percebem, quando usando seu leitor de tela. Desse modo, os seguintes requisitos mais especializados desse cenário de uso foram especificados para a SRS:

**R1 - somente teclado:** a interação com o site prioritariamente por meio do teclado;

**R2 - sem imagens:** impossibilidade de visualização de imagens da página;

**R3 - um bloco de texto:** visualização de apenas um bloco de texto por vez, simulando a “leitura” de cada bloco (no caso de um leitor de tela real, esse bloco seria reproduzido em voz alta);

**R4 - foco pontual:** ausência de auxílio contextual por meio do layout visual da página, ou seja, simulando a percepção pontual somente do elemento em foco pelo leitor de tela;

**R5 - navegação sequencial:** necessidade de navegação na página percorrendo todos os seus elementos, na ordem em que estão dispostos no conteúdo da página Web, até encontrar o elemento desejado.

Nesse contexto, a seguir é descrito um conjunto de características funcionais que compõem a ferramenta desenvolvida.

## 4.2 O simulador SRS e seu funcionamento

A ferramenta SRS foi desenvolvida para proporcionar a experiência de uso de um leitor de tela, por desenvolvedores Web, para auxiliá-los durante a avaliação de acessibilidade em uma página da Web. Com base nesse objetivo, a ferramenta foi proposta para ser um simulador de leitor de tela a ser utilizada pelos desenvolvedores web. SRS consiste de um conjunto de transformações e ações sobre uma página da Web visando aproximar a sua “visualização” e utilização, daquela percepção vivenciada pelos usuários que necessitam do leitor de tela como Tecnologia Assistiva.

Ao ser adicionada à página da Web sendo avaliada, a ferramenta SRS executa diversos procedimentos de inicialização com o propósito de ativar e desativar funcionalidades da página, bem como ser capaz de detectar a navegação do usuário por ela.

Em um primeiro momento, o simulador desativa a interação com a página por meio de acionamento do *mouse*, permitindo apenas que a navegação por teclado ocorra (obs.: o mouse não será desativado do computador, apenas não terá efeito algum ao clicar ou interagir com a página). Desse modo, o requisito “R1 - somente teclado” é atendido. Apenas com essa medida, dentre os problemas de experiência que já podem ser expostos, tem-se, por exemplo:

- Botões ou links não acessíveis por meio do teclado;

- Rolagem da página customizada ou difícil de ser controlada sem o auxílio de um mouse.

Atendendo ao requisito “R4 - foco pontual”, o simulador esconde todos os elementos da página que não estejam sendo percebidos pelo leitor de tela, ou seja, exibe apenas o bloco de texto que contém o foco do cursor atual. Dessa forma, o contexto global da página é escondido, restando apenas a informação pontual que pode ser lida para o usuário. Com isso, o usuário necessita percorrer o conteúdo completo da página, em ordem, até atingir o bloco de texto com a informação desejada. Esse percurso sequencial para acesso ao conteúdo de cada bloco de texto na página atende aos requisitos “R3 - um bloco de texto” e “R5 - navegação sequencial.”

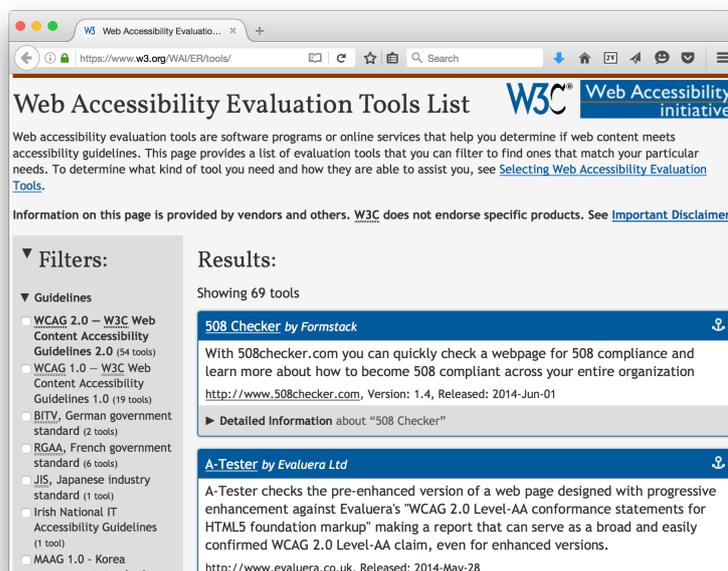
Outros obstáculos para usuários cegos, identificados na literatura como relacionados à acessibilidade na Web, também podem ser expostos por esse método elaborado para atender aos requisitos R3, R4 e R5. Um exemplo disso é o caso identificado das atualizações dinâmicas de página, que podem ser imperceptíveis para o usuário e fazer com que ele se perca quanto às informações importantes ou não seja capaz de contextualizar corretamente a informação sendo exibida.

A seguir, são descritas as duas versões construídas para a ferramenta SRS. A primeira versão da ferramenta foi construída para fundamentar os principais recursos relacionados com técnicas de implementação de um simulador de leitor de tela conforme a literatura mencionava, e os requisitos foram especificados. Em seguida, melhorias foram feitas na primeira versão, a partir de avaliação com usuários, e foi desenvolvida a segunda versão desenvolvida.

### 4.3 SRS - Versão 1

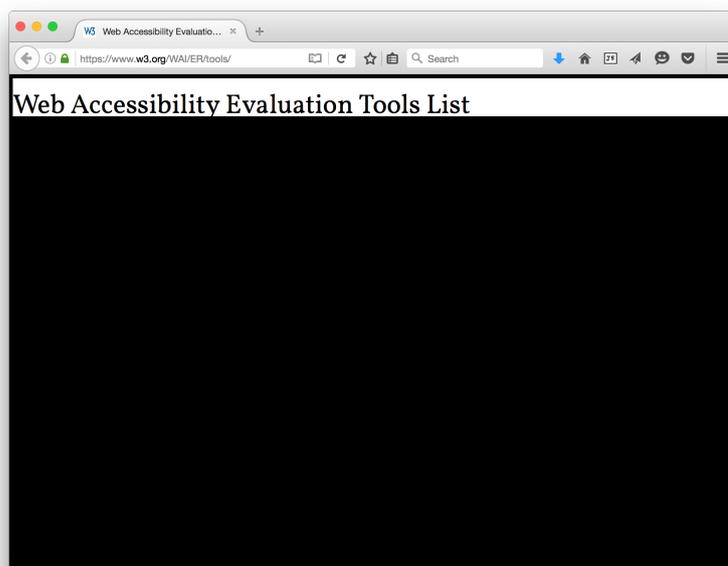
Em sua versão 1, a ferramenta SRS apresenta as funcionalidades básicas desejadas, conforme descritas na [Seção 4.2](#). Ao ser ativada, a SRS desativa o uso do mouse na página e a obscurece por completo, exceto para o elemento em foco. As Figuras 8 e 9 ilustram esse seu funcionamento.

Figura 8 – SRS Versão 1: antes de ser ativado o simulador, ilustração da página em seu estado original.



Fonte: Elaborada pelo autor.

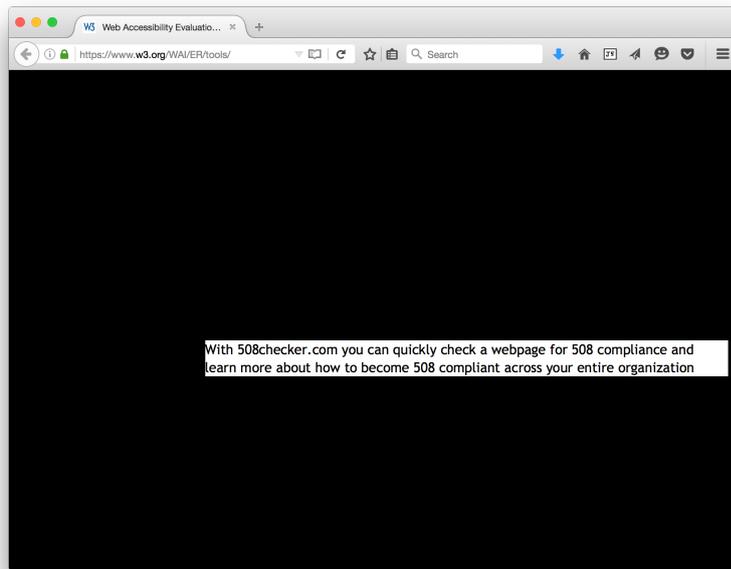
Figura 9 – SRS Versão 1: após o simulador ser ativado, ilustração da página com todo o seu conteúdo escurecido, exceto pelo bloco de texto que contém foco.



Fonte: Elaborada pelo autor.

Para a navegação na página, o desenvolvedor usando a SRS versão 1m deve utilizar a tecla TAB do teclado. Ao pressioná-la, o foco da página (e do simulador) é então movido para o próximo elemento textual da página. A Figura 10 ilustra a navegação na página enquanto o simulador está ativado. Para ler o conteúdo presente na página, o usuário percorre os blocos de texto utilizando a tecla TAB.

Figura 10 – Uso da SRS - Versão 1: ilustração da página ao ser utilizada com o simulador ativado, em que o foco de texto é percorrido com a tecla TAB até atingir outro conteúdo.



Fonte: Elaborada pelo autor.

## 4.4 SRS - Versão 2

Com base na primeira versão da SRS, foram realizados testes iniciais com usuários (descritos na [Seção 5.1](#)), tanto para validar a ideia proposta, quanto para analisar a viabilidade de criação desta ferramenta. Também foi realizada uma avaliação qualitativa com usuários, com base numa análise da execução pelos usuários de tarefas pré-definidas, conforme descrita no [Capítulo 5](#).

Além dos conhecimentos adquiridos durante a implementação da SRS versão 1, prosseguimos estudando sobre o funcionamento dos leitores de tela presentes no mercado, bem como a forma que eles são tipicamente usados pelos usuários com deficiência visual.

De imediato, percebe-se que um leitor de tela moderno não se limita a simplesmente ler os textos visíveis em uma página: existem informações invisíveis a um usuário comum que são expostas quando usa-se um leitor de tela, como descrições alternativas para links, legendas para imagens, links especiais destinados à utilização do *website* por teclado, entre outros.

Os padrões de comportamento durante a utilização dos leitores de tela, pelos seus usuários, são bastante diversos. [Borodin et al. \(2010\)](#) relatam que existem inúmeras formas de estratégias para uso das teclas e para navegação em uma página, quando um leitor de tela é utilizado, e cada usuário desenvolve seus próprios costumes e estratégias. Essas diferenças se dão por inúmeros motivos, dentre eles:

- Diferentes níveis de experiência dos usuários;
- Ferramentas específicas que apresentam funcionalidades distintas, sendo utilizadas;
- Preferências pessoais diversas.

Alguns exemplos de estratégias de navegação descritos por [Borodin et al. \(2010\)](#) são a navegação com a utilização dos elementos *headings* na página; a busca por palavras-chave utilizando o mecanismo de busca do navegador; a busca por proximidade, aonde o usuário tenta encontrar um link ou algum texto que acredita estar próximo, para então continuar uma busca sequencial a partir dali; entre outros.

Assim, conhecer o comportamento e as funcionalidades específicas de cada ferramenta se faz fundamental. Dois leitores de tela se destacam como os mais conhecidos para a utilização junto aos navegadores web:

- O *NonVisual Desktop Access* (NVDA) - um leitor de tela gratuito e de código aberto, e
- O *Job Access With Speech* (JAWS) - uma opção comercial bem estabelecida no mercado.

Vale destacar que existe uma vasta disponibilidade de conteúdo na Web cujo propósito é ensinar sobre a utilização dessas ferramentas, tais como: documentação oficial <sup>2 3</sup>, guias de utilização <sup>4</sup>, e-books (NAPNE, 2009), vídeo-tutoriais (SVRC, 2013), (CNIB, 2015), etc.

Com base nos trabalhos relatados na literatura sobre o uso dos leitores de tela, juntamente com o estudo desses materiais disponibilizados na Web, as seguintes funcionalidades foram adicionadas à segunda versão do simulador SRS:

- R6** - Preservação da tecla TAB com seu comportamento padrão no navegador (percorre apenas elementos de interação, como links, botões e campos de texto)
- R7** - Parada em elementos de imagem para a visualização de sua legenda
- R8** - Parada em elementos especiais somente visíveis para leitores de tela
- R9** - Parada em elementos não-visíveis, porém acessíveis, como elementos de um menu que se encontra fechado
- R10** - Painel de controle com botões de comando e exibição de textos alternativos, quando disponíveis
- R11** - Navegação capaz de percorrer para frente ou para trás, por todos elementos
- R12** - Navegação capaz de percorrer para frente ou para trás, exclusivamente pelos elementos de *headers*
- R13** - Funcionalidade de semi-transparência para facilitar análise e compreensão por parte do desenvolvedor

Além desses 8 novos requisitos especificados, o contraste anterior do fundo de tela preto com a janela de visualização branca causava um certo desconforto ao utilizar o simulador. Para melhorar esta questão, um fundo mais brando, de cor cinza claro, foi utilizado.

A Figura 11, a Figura 12 e a Figura 13 mostram *screendumps* da segunda versão da ferramenta SRS em funcionamento.

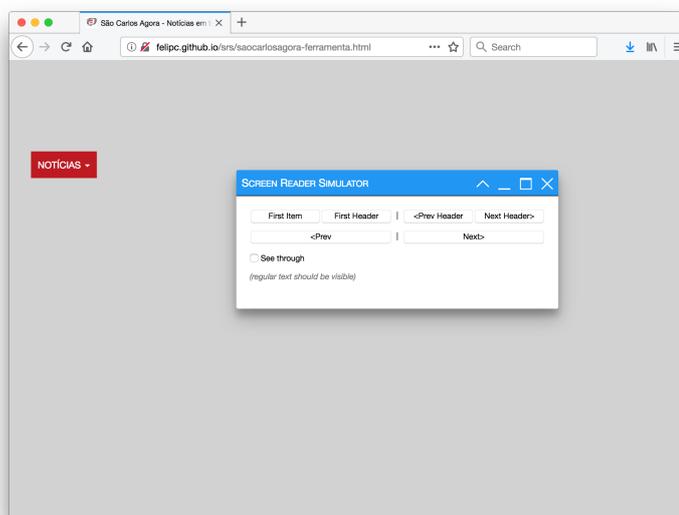
---

<sup>2</sup> <http://www.freedomscientific.com/Downloads/Documentation>

<sup>3</sup> <https://www.nvaccess.org/about/nvda-features/>

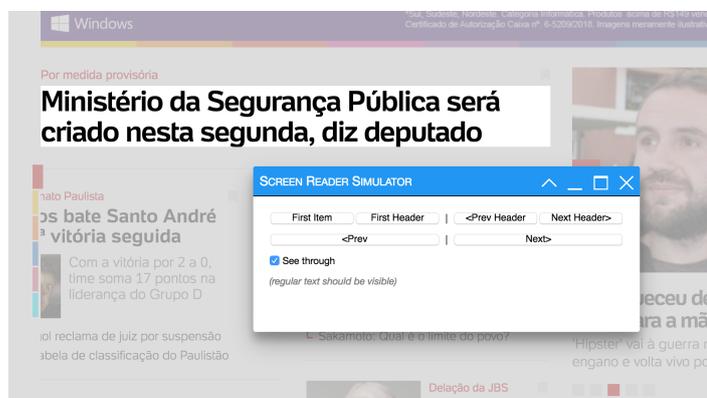
<sup>4</sup> <https://www.nvaccess.org/files/nvda/documentation/userGuide.html>

Figura 11 – SRS - Versão 2 - Simulador ativado e seu painel de controle



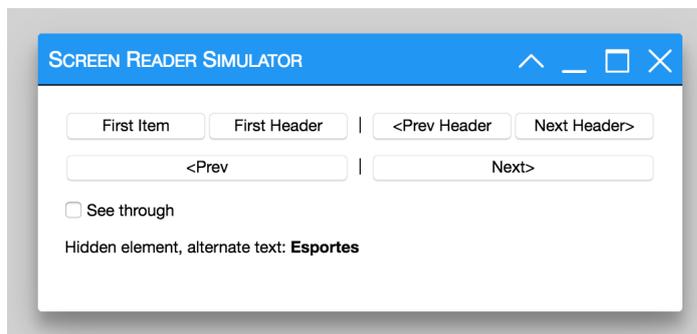
Fonte: Elaborada pelo autor.

Figura 12 – SRS - Versão 2 - Simulador no modo de semi-transparência



Fonte: Elaborada pelo autor.

Figura 13 – SRS - Versão 2 - Destaque para o painel de controle com funções de navegação



Fonte: Elaborada pelo autor.

## 4.5 Considerações Finais

Neste capítulo, foi descrito o simulador SRS, uma ferramenta para auxiliar desenvolvedores videntes a realizarem avaliação de acessibilidade em página Web. Contextualizou-se o objetivo do projeto desenvolvido tendo a ferramenta SRS como prova de conceito, frente aos temas estudados e abordados nos capítulos anteriores. Foi apresentado inicialmente o levantamento de requisitos. Em seguida, foram apresentadas as versões de SRS desenvolvidas, bem como as funcionalidades presentes em cada uma delas.

No [Capítulo 5](#), a seguir, serão apresentadas as metodologias de análise empregadas para avaliar o trabalho aqui exposto, bem como os resultados obtidos.

---

## AVALIAÇÃO

---

Para avaliar o potencial do simulador SRS em auxiliar desenvolvedores Web testarem suas páginas da Web, com o intuito de verificar suas características de acessibilidade, foram empregadas três mecanismos de avaliação. Neste capítulo são descritas as seguintes avaliações realizadas: testes com usuários (tarefas típicas para avaliação) na [Seção 5.1](#), avaliação da carga mental de trabalho do usuário na [Seção 5.2](#), e inspeção por especialistas na [Seção 5.3](#).

### 5.1 Testes com Usuários executando Tarefas

O método de avaliação primeiramente adotado foi o de Testes com Usuários. Na literatura especializada, esse tipo de avaliação possibilita a definição de tarefas a serem realizadas pelos usuários (SHNEIDERMAN, 2000). [Pereira, Ferreira e Archambault \(2015\)](#) se utilizaram de uma abordagem de tarefas para avaliar elementos de acessibilidade em páginas web, com base nos 2 seguintes passos:

1. Identificação de *websites* relevantes no contexto social da pesquisa realizada, por exemplo:
  - Portais de notícias e entretenimento
  - Sistemas de e-mail
  - *Website* de empresa prestadora de serviço (água, luz, energia)
  - Redes sociais
  - Páginas de programas sociais ou ferramentas governamentais, etc.
2. Identificação de tarefas típicas a serem realizadas nestes sites, por exemplo:
  - Visualização das notícias em destaques em um portal
  - Busca de notícias sobre um tema específico (por exemplo, Esportes)

- Busca de meios de contato com uma empresa
- Preenchimento de formulários
- Escrita de e-mails
- Busca de amigos em redes sociais.

De modo semelhante à abordagem utilizada por [Pereira, Ferreira e Archambault \(2015\)](#), para este estudo, buscou-se utilizar *websites* que fazem parte do contexto social da universidade e da cidade. Os dois *websites* escolhidos foram:

1. São Carlos Agora (SCA) - Portal de notícias da cidade de São Carlos<sup>1</sup>;
2. Polícia Federal (PF) - Site da Polícia Federal, com acesso ao serviço de passaportes<sup>2</sup>.

Esses sites representam os de maior interesse social da comunidade de usuários (universitários da cidade de São Carlos), à época de realização dessa avaliação. Com o intuito de identificar o conjunto de tarefas que podiam ser realizadas nos dois *websites* escolhidos, procedemos uma exploração empírica de seus conteúdos. Foram elencadas tarefas de variados níveis de dificuldade.

As tarefas definidas para avaliação do site São Carlos Agora (SCA) foram as 19 elencadas na [Tabela 1](#); as tarefas definidas para avaliação do site Polícia Federal (PF) foram as 16 elencadas na [Tabela 2](#).

### 5.1.1 Metodologia

Para a execução da avaliação, uma turma de 26 alunos de graduação, cursando a disciplina de Interação Usuário-Computador, foi convidada a participar, desempenhando o papel de desenvolvedor Web. Esses alunos foram divididos igualmente em dois grupos, Grupo A e Grupo B. A cada um dos participantes foi dada a liberdade de não participar dos testes, bem como garantia de que poderia desistir da atividade a qualquer momento, sem qualquer prejuízo. Os alunos assinaram o Termo de Consentimento Livre e Esclarecido (TCLE), de modo a garantir também a privacidade dos dados a serem coletados.

Para cada um desses grupos foi solicitado que realizassem dois Testes (com e sem o SRS). Além disso, cada Sessão de Teste era composta pelo conjunto de tarefas a serem executadas nos dois sites, SCA e PF. A organização dessa distribuição de tarefas nos Testes, os quais seriam realizados nos dois sites pode ser vista na [Tabela 3](#).

Um formulário interativo com a lista de tarefas e instruções de execução do teste foi fornecido aos participantes, com as seguintes instruções:

<sup>1</sup> <http://www.saocarlosagora.com.br>

<sup>2</sup> <http://www.pf.gov.br>

Tabela 1 – Tarefas para o site SCA

T1	Encontrar a data no site
T2	Encontrar a notícia principal no site
T3	Ler a notícia principal e determinar a data/horário da última atualização
T4	Verificar a quantidade de notícias em destaque
T5	Encontrar a seção "Últimas notícias"
T6	Encontrar os botões de imprimir / compartilhar notícia na página de uma notícia
T7	Ir para página com notícias de esportes
T8	Verificar pódio (3 primeiros colocados) da corrida de F1 noticiada
T9	Verificar quantos pilotos não completaram a corrida
T10	Encontrar seção de colunistas
T11	Ir para pagina 2 dos colunistas
T12	Ir para a ultima pagina dos colunistas
T13	Encontrar a pagina de contato
T14	Verificar com quantos departamentos é possível entrar em contato usando a página de contato
T15	Encontrar numero do <i>WhatsApp</i> do site para entrar em contato
T16	Encontrar páginas com notícias da cidade
T17	Encontrar matéria sobre olimpíada de robótica
T18	Verificar a autora da matéria sobre a olimpíada de robótica
T19	Verificar quantos itens existem no menu "notícias"

Fonte: Elaborada pelo autor.

1. Percorrer a lista de tarefas em ordem
2. Passar para a próxima tarefa ao completar a atual ou ao julgá-la impossível
3. Ao realizar cada tarefa, marcar numa *checkbox* correspondente, caso ela tenha sido completada com sucesso
4. Limite máximo de tempo: 30 minutos.

Além disso, ao final do teste, outras questões de usabilidade do *site* foram apresentadas, com o objetivo de realizar outra avaliação, qualitativa, conforme apresentada na [Seção 5.2](#).

Uma versão estática da página principal e de todas as suas sub-páginas (até 2 níveis) foi armazenada e utilizada para os testes, para garantir que todos usuários pudessem visualizar exatamente as mesmas páginas, com conteúdos idênticos. Esta avaliação foi realizada utilizando-se a versão 1 da ferramenta SRS, nos dias 3 e 4 de maio de 2016.

### 5.1.2 Resultados

Em [Bigham, Lin e Savage \(2017\)](#), é relatada uma avaliação baseada em testes que foi realizada, comparando os resultados entre dois grupos: usuários videntes e usuários com deficiência visual. Cada tarefa estipulada foi classificada em fácil, difícil, inacessível ou impossível, e comparada entre os grupos e também com uma resposta-base aguardada pelos autores.

Tabela 2 – Tarefas para o site PF

T1	Encontrar página referente a informações sobre passaporte
T2	Encontrar lista documentos requeridos para solicitação de passaporte
T3	Encontrar lista de documentos requeridos para passaporte para menores de 18 anos
T4	Encontrar tabela com validades dos passaportes
T5	Encontrar link externo para o serviço de agendamento de passaporte
T6	Achar informações de acessibilidade e teclas de atalho
T7	Navegar pelo site utilizando os <i>breadcrumbs</i>
T8	Encontrar mapa do site
T9	Achar a página do estado de São Paulo no mapa do site
T10	Verificar o último item do mapa do site ("Agência de notícias")
T11	Encontrar endereço do posto de emissão de passaportes em São Carlos (difícil)
T12	Encontrar caixa de busca do site
T13	Achar como limitar a busca com a opção "só nessa seção"
T14	Achar página de serviços para estrangeiros
T15	Verificar quantos itens estão presentes na lista de serviços para estrangeiros
T16	Encontrar link para "Termos de solicitação de refúgios" na página de serviços para estrangeiros

Fonte: Elaborada pelo autor.

Tabela 3 – Distribuição de Tarefas aos Grupos

<b>Grupo A</b>	
Teste 1	Tarefas do site São Carlos Agora, sem o simulador (página original)
Teste 2	Tarefas do site da Polícia Federal, com o simulador
<b>Grupo B</b>	
Teste 1	Tarefas do site da São Carlos Agora, com o simulador
Teste 2	Tarefas do site da Polícia Federal, sem o simulador (página original)

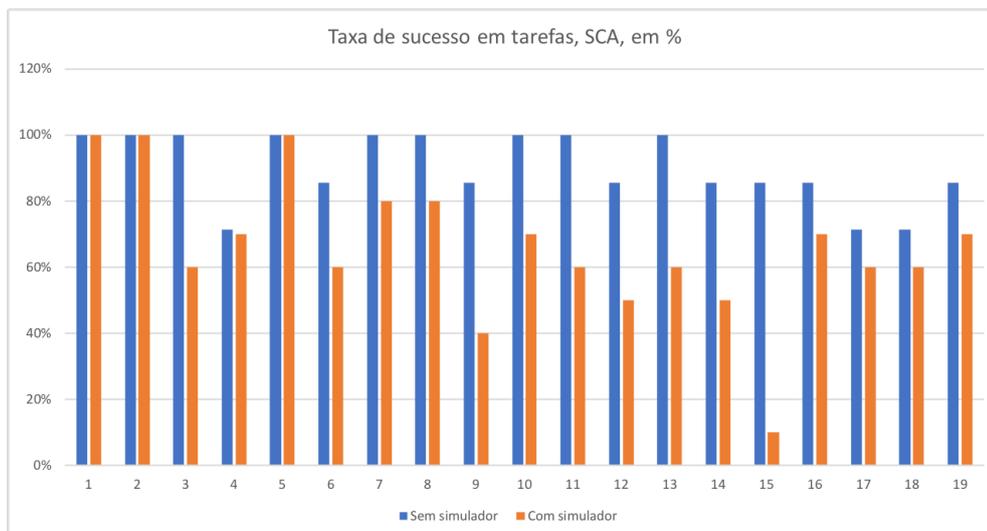
Fonte: Elaborada pelo autor.

Tomando como base a comparação de [Bigham, Lin e Savage \(2017\)](#), de modo similar, os testes com usuários executados neste mestrado foram analisados comparando-se as respostas obtidas, para cada site, entre as duas classes de teste:

- Usuários videntes, com a página original (sem simulador);
- Usuários videntes, com a página utilizada no SRS.

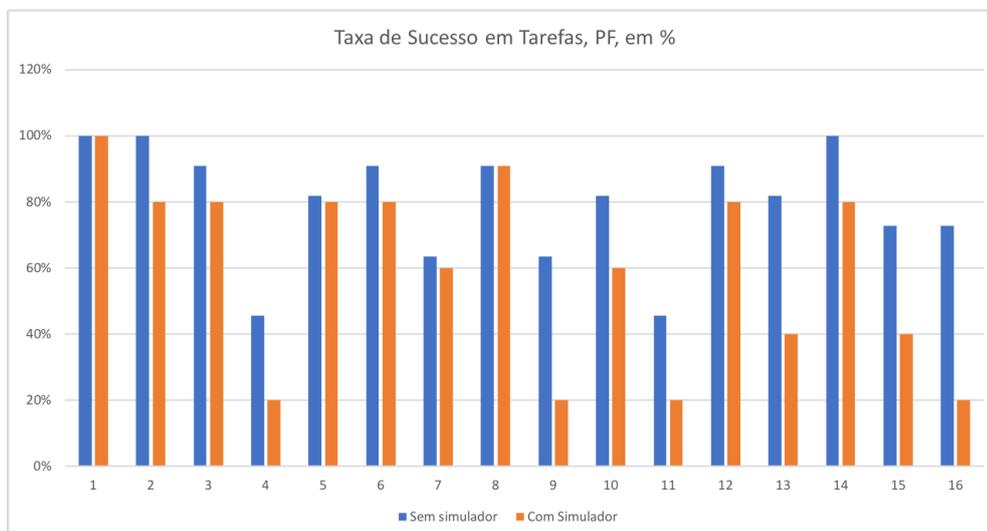
As respostas foram classificadas em bem-sucedidas ou não, de acordo com que o usuário relatou no formulário. A [Figura 14](#) mostra os resultados obtidos no site São Carlos Agora, e a [Figura 15](#), no site da Polícia Federal.

Figura 14 – Taxa de Sucesso em tarefas no site São Carlos Agora



Fonte: Dados da pesquisa.

Figura 15 – Taxa de Sucesso em tarefas no site da Polícia Federal



Fonte: Dados da pesquisa.

Analisando os gráficos dos resultados obtidos, é possível notar que o simulador de tela SRS consistentemente diminuiu a taxa de sucesso na realização das tarefas propostas. Desta forma, acredita-se que a ferramenta é capaz de simular, para o desenvolvedor web interessado, alguns obstáculos nos *websites*, os quais não foram antes encontrados por usuários videntes.

## 5.2 Avaliação de Carga Mental de Trabalho

Além da avaliação por meio de testes com usuários executando tarefas, o simulador SRS foi também aliado sob a perspectiva de carga mental de trabalho exigida dos usuários. A avaliação de carga mental de trabalho se fundamenta em um conjunto de métricas chamado NASA-TLX (HART, 1986)<sup>3</sup>. Essas métricas possuem âmbito multi-dimensional, cujo propósito é cobrir os seguintes aspectos sobre a execução de um trabalho (HART; STAVENLAND, 1988):

- Demanda física, mental e emocional
- Performance
- Esforço
- Frustração

### 5.2.1 Metodologia

Com base no conjunto de métricas NASA-TLX, proposto na década de 80, e amplamente utilizado e revisitado (HART, 2006) na literatura desde então, Kirakowski e Claridge (2012) apresentam uma série de questões direcionadas para análise de carga mental de trabalho em *websites*. Dentre essas questões, as mais relevantes para o contexto deste trabalho foram escolhidas e apresentadas para os dois grupos de usuários que participaram do teste executando tarefas (Seção 5.1), ao final das sessões.

Cada item do questionário é apresentado na forma de uma afirmação a ser julgada pelo participante (usuário), resultando em uma apreciação que deve ser manifestada por ele(a) com a escolha de uma opção de valor numa escala de 1 a 5, onde 1 representa “Discordo plenamente”, e 5 representa “Concordo plenamente”.

Para analisar os resultados deste questionário, é preciso se atentar às características subjetivas de cada item, dado que alguns itens possuem conotações positivas (por exemplo, “*Este site me agrada*”), e alguns possuem conotações negativas (por exemplo, “*Este site é muito difícil de navegar*”).

Os itens incluídos no questionário são listados na Tabela 4 e na Tabela 5.

<sup>3</sup> <https://en.wikipedia.org/wiki/NASA-TLX> e <https://humansystems.arc.nasa.gov/groups/tlx/>

Tabela 4 – Afirmações positivas do questionário de carga mental de trabalho

(a) Este <i>website</i> é interessante para mim
(b) Eu consigo encontrar rapidamente o que eu quero neste site
(c) O site tem lógica para mim
(d) Eu me sinto no controle ao usar este <i>website</i>
(e) O site me ajuda a encontrar o que eu procuro
(f) Eu consigo entrar em contato facilmente com este <i>website</i>
(g) Eu me sinto eficiente ao usar este <i>website</i>
(h) Usar este site pela primeira vez é fácil
(i) Eu obtenho o esperado ao clicar neste site
(j) Tudo neste site é fácil de entender

Fonte: Elaborada pelo autor.

Tabela 5 – Afirmções negativas do questionário de carga mental de trabalho

(a) É difícil navegar neste site
(b) Aprender a me localizar neste site é problemático
(c) O site precisa de mais explicações introdutórias
(d) O site é muito lento
(e) Eu não gosto de usar este site
(f) É difícil determinar se este site tem o que eu quero
(g) Este site tem elementos irritantes
(h) Me lembrar de onde estou neste site é difícil
(i) Este site é uma perda de tempo

Fonte: Elaborada pelo autor.

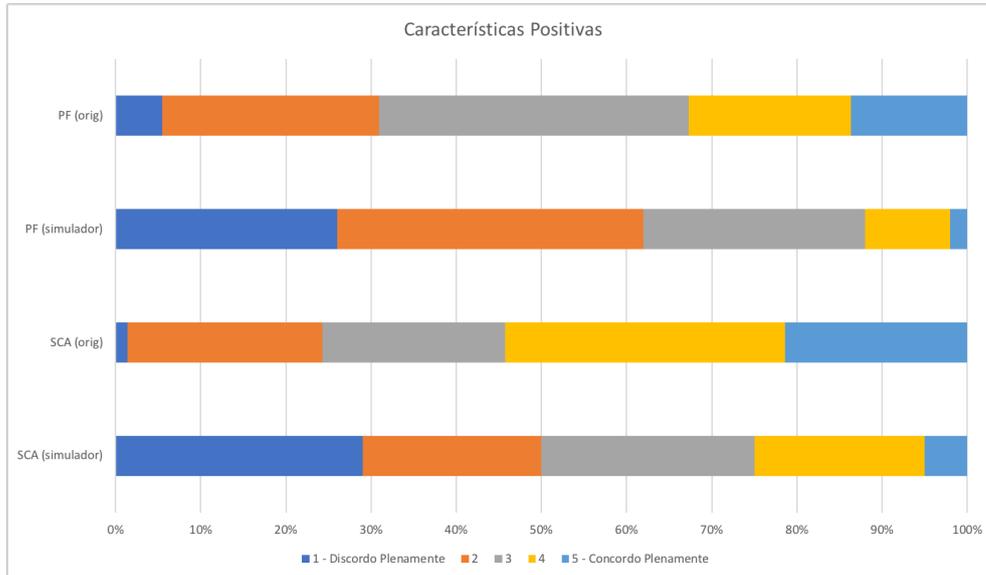
## 5.2.2 Resultados

A [Figura 16](#) e a [Figura 17](#) mostram a distribuição das respostas obtidas dos usuários, para as afirmações que lhes foram apresentadas. Considerando-se a distinção entre as respostas sobre as características positivas das negativas, pode-se identificar impressões subjetivas que os usuários manifestaram quando realizaram a atividade de avaliação de acessibilidade nos sites propostos, com auxílio e sem auxílio da ferramenta SRS.

A [Figura 16](#), referente às respostas de características positivas, traz as respostas dos usuários sobre a experiência em ambos os sites (SCA e PF). Nos dois casos, fazendo uma comparação das respostas sobre o site original e as respostas sobre o site com o simulador, pode-se ver um aumento na distribuição das respostas mais baixas ("1 - discordo plenamente" e "2") para o caso em que o simulador foi utilizado.

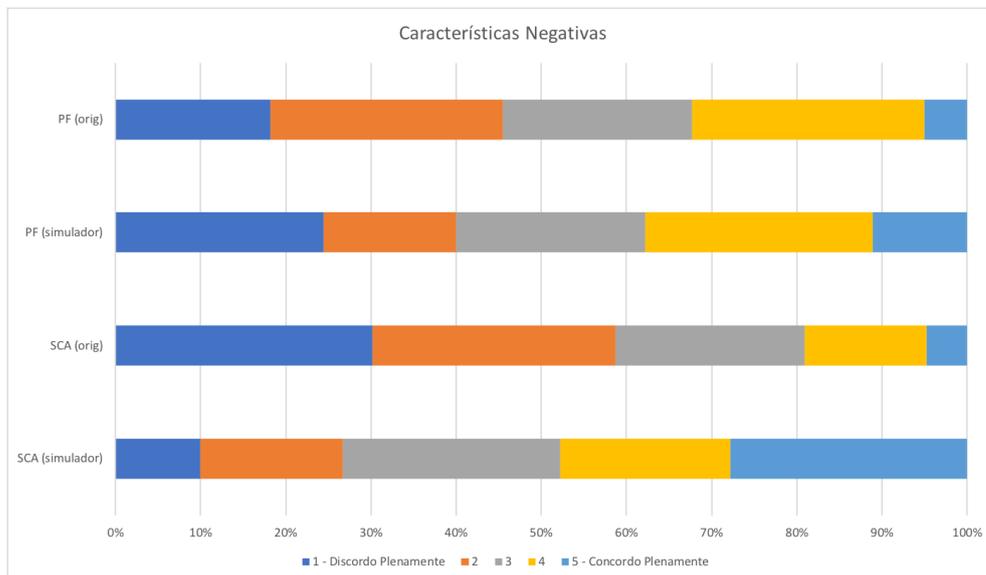
Similarmente, na [Figura 17](#), referente às respostas de características negativas, nota-se um aumento na distribuição das respostas mais altas ("4" e "5 - concordo plenamente"), para o caso de uso do simulador.

Figura 16 – Distribuição de respostas sobre características positivas



Fonte: Dados da pesquisa.

Figura 17 – Distribuição de respostas sobre características negativas



Fonte: Dados da pesquisa.

Estes resultados indicam que houve uma maior carga mental de trabalho relatada pelos usuários que executaram as tarefas com apoio do simulador, quando comparados com os usuários que executaram as mesmas tarefas sobre o site original, sem contar com o SRS. Essa carga mental sugere que com o SRS, algumas barreiras são apresentadas aos usuários desenvolvedores, que estão com o principal objetivo de encontrar esses problemas de acessibilidade, que são

percebidos como barreiras para os usuários com deficiência visual.

## 5.3 Inspeção por especialistas

Com a segunda versão da ferramenta, empregou-se uma nova metodologia de avaliação, visando realizar uma análise empírica sobre as capacidades do simulador SRS, e comparativa entre ferramentas de validação (tipicamente utilizadas para análise de acessibilidade) e o SRS. O intuito desta abordagem é verificar em quais aspectos o simulador é capaz de ir além das técnicas atuais e oferecer novos benefícios para as atividades de verificação de acessibilidade de um *website*.

Para esta abordagem, contou-se com a colaboração de dois especialistas em acessibilidade na Web para pessoas com deficiência visual. Esses especialistas foram convidados por já estarem bastante habituados a realizar avaliações especializadas, como planejadas neste trabalho.

### 5.3.1 Metodologia

Para esta avaliação, uma nova seleção de *sites* foi realizada. Selecionou-se novamente o *site* São Carlos Agora <sup>4</sup>, e escolheu-se também o portal UOL <sup>5</sup>, para fazer uso de duas páginas com a temática de portal de notícias, que possuem uma variedade de conteúdos diferentes, proporcionando uma experiência mais exploratória. Para ambos, a avaliação de acessibilidade foi feita com sua página inicial, que foi salva estaticamente para garantir que as análises ocorressem sobre o mesmo conteúdo.

O primeiro avaliador foi encarregado de realizar uma avaliação de acessibilidade típica, utilizando ferramentas de validação. Para isso, duas ferramentas foram utilizadas: *TAW* (CTIC, 2008) e *Cynthia Says* (ICDRI, 2003). Ambas trabalham sobre um endereço de página fornecido para realizar uma validação de conformidade com o padrão WCAG 2.0.

O segundo avaliador foi encarregado de realizar a avaliação sobre as mesmas duas páginas, usando exclusivamente o simulador de leitor de tela.

O simulador, por sua característica exploratória e interativa, não gera resultados quantitativos como as ferramentas de validação. Assim, a comparação entre os resultados se fez de forma qualitativa, observando as capacidades de cada ferramenta em detectar e relatar erros presentes nestas páginas.

### 5.3.2 Resultados

Na *Tabela 6* é mostrado um resumo dos resultados apresentados pelas duas ferramentas de validação sobre os dois sites analisados.

<sup>4</sup> <http://www.saocarlosagora.com.br>

<sup>5</sup> <http://www.uol.com.br>

Tabela 6 – Avaliação das páginas por meio de validadores

Normas WCAG 2.0	SCA		UOL	
	TAW	CynthiaSays	TAW	CynthiaSays
Aprovadas	2	1	1	1
Reprovadas	16	9	17	11
Análise Humana	26	39	26	38
Não se Aplica	17	12	17	11

Fonte: Elaborada pelo autor.

A [Tabela 7](#) lista exemplos de alguns erros evidenciados pelas duas ferramentas de validação.

Tabela 7 – Erros relatados pelas ferramentas de validação

(a) Elementos <form> sem <label> correspondentes
(b) Texto e imagem consecutivas apresentam link para o mesmo lugar
(c) Link com texto muito longo
(d) Nenhum elemento <h1>
(e) Links com o mesmo texto mas com destinos diferentes
<i>etc.</i>

Fonte: Elaborada pelo autor.

Um fator interessante que pode ser observado nos resultados é que os tipos de erros apontados podem ou não representar problemas reais de acessibilidade nestas páginas, dependendo de qual elemento e conteúdo específico a que se referem. Outra característica, ainda, é a grande quantidade de pontos que estas ferramentas são incapazes de analisar, delegando a maioria dos seus critérios à uma análise humana.

Para a ferramenta de simulação SRS, os erros apontados foram descritos manualmente pelo avaliador, após utilizar a página por meio do simulador de leitor de tela. A [Tabela 8](#) lista exemplos de erros relatados por este avaliador.

Com a utilização de SRS, o avaliador foi capaz de detectar diversos problemas de acessibilidade e usabilidade nessas páginas. É possível separar os erros encontrados com o uso do simulador em dois grupos:

**Erros em comum com validadores** - Alguns dos erros encontrados também foram apontados na análise com os validadores. Entretanto, mesmo neste caso, a simulação é capaz de apontar com maior objetividade os erros que representam obstáculos ou frustrações reais do uso daquele *website*, ao invés de fazer parte de uma lista extensa de erros agrupados juntos pela ferramenta de validação.

**Erros inéditos** - Outros erros só foram detectados com o uso do simulador de leitor de tela, como por exemplo o erro em que a descrição textual de uma imagem está implementada,

Tabela 8 – Erros relatados pela ferramenta de simulação

(a) Primeiro elemento de cabeçalho da página está após muito conteúdo
(b) Textos alternativos são pouco explicativos
(c) Menus apresentam opções em excesso
(d) Existem elementos sem texto alternativo
(e) Primeiro elemento da página não é útil
(f) Uso de jargão e siglas em conteúdo alternativo (exemplo, "SCA"no São Carlos Agora)
(g) Descrição alternativa da imagem não a descreve, apenas detalha <i>copyright</i>
(h) Conteúdo principal da página não está descrito por um elemento <i>heading</i>
(i) Existe muito conteúdo secundário antes de se chegar ao principal
<i>etc.</i>

Fonte: Elaborada pelo autor.

porém o texto não é útil para sua compreensão, ou o erro que aponta que o conteúdo principal do site não corresponde ao conteúdo em que o elemento `<h1>` foi utilizado.

Desta forma, a ferramenta se mostrou capaz de detectar alguns tipos de erros que não estavam abrangidos pelas ferramentas de validação tradicionais, bem como adicionar valor aos erros detectados pelo validador, como forma de apoio para o caso de uma avaliação de acessibilidade feita com múltiplas abordagens.

## 5.4 Considerações Finais

Neste capítulo, foram apresentadas as diversas metodologias utilizadas para validar, iterar e avaliar o trabalho realizado neste projeto de mestrado.

Com a utilização de uma abordagem de tarefas e de um questionário, verificou-se que o simulador de leitor de tela é capaz de aumentar a carga mental de trabalho e a dificuldade na realização de algumas tarefas em uma página alvo, evidenciando dificuldades enfrentadas por usuários de um leitor de tela real.

A partir de uma análise com auxílio de especialistas em acessibilidade, verificou-se que o simulador é capaz de detectar erros de implementação e de conteúdo da página que se traduzem diretamente em problemas enfrentados por usuários com deficiência visual.



---

## CONCLUSÕES

---

Este trabalho de mestrado foi motivado pela necessidade de manter a Web um meio de comunicação e de conhecimento acessível a todos, independente de suas necessidades especiais.

Com a rápida evolução das tecnologias na Web, os desenvolvedores web muitas vezes encontram dificuldades para conseguir analisar as características de acessibilidade de suas páginas devido a várias razões, dentre elas, destacam-se: o pouco conhecimento sobre como verificar se suas soluções implementadas são utilizáveis por usuários cegos e pouco apoio para identificar e implementar melhorias a fim de ampliar adequadamente a acessibilidade.

Diferentes mecanismos têm sido propostos para auxiliar a análise e correção de questões de acessibilidade de páginas web. Muitos deles são disponibilizados por meio de ferramentas automatizadas (IVORY; HEARST, 2001). Porém, muitas vezes, essas ferramentas ainda não dão suporte para apresentar as reais barreiras de acessibilidade, por não serem capazes de associar a lacuna técnica com o efeito final que um usuário com deficiência visual poderá encontrar.

Nesse contexto, foi proposta uma abordagem para auxiliar desenvolvedores web a identificarem barreiras de acessibilidade enfrentadas por usuários que utilizam leitores de tela. No presente trabalho, a abordagem proposta foi concebida na forma de um simulador de leitor de tela, para ser usado por desenvolvedores web, evidenciando a experiência de um usuário com deficiência visual.

Para a realização do trabalho, primeiro estudou-se na literatura o contexto, a fundamentação conceitual e a situação atual da Web em si. Os tópicos estudados foram relacionados diretamente à acessibilidade, e em especial, a acessibilidade na Web para pessoas com deficiência visual. Buscou-se compreender as dificuldades enfrentadas no dia-a-dia desses usuários ao navegar na Web, bem como adquirir conhecimentos sobre as técnicas e padrões existentes de desenvolvimento, os quais se propõem a facilitar e garantir acesso aos conteúdos da Web.

A partir do estudo sobre os tópicos relacionados com acessibilidade, foi possível expandi-

lo, em mais profundidade, sobre as ferramentas de auxílio à utilização do computador por usuários com deficiência visual (Tecnologia Assistiva), bem como sobre as ferramentas que apoiam o desenvolvimento de *sites* acessíveis. Desse modo, foi possível identificar o que existe atualmente no mercado, bem como as ideias propostas em trabalhos publicados na literatura, para propor uma nova abordagem. Assim, uma nova ferramenta foi projetada, com o intuito de se averiguar a sua viabilidade e seus benefícios diante da demanda por um auxílio para desenvolvedores web, durante a atividade de avaliação de acessibilidade.

Um protótipo de simulador de leitor de tela (SRS - descrito no [Capítulo 4](#)) foi desenvolvido, com o intuito de oferecer aos desenvolvedores uma nova forma de avaliar o grau de acessibilidade de suas páginas para usuários de leitores de tela real. Com base nos resultados obtidos a partir de testes de uso desse protótipo, por usuários reais, uma segunda versão de SRS foi desenvolvida, contando com melhorias e novas funcionalidades.

A abordagem proposta visou simplificar a compreensão do uso de leitor de telas em páginas Web para avaliadores humanos videntes. Neste sentido, a SRS traduz os resultados auditivos de um leitor de telas em resultados visuais. Assim, o trabalho científico realizado buscou responder à seguinte questão de pesquisa:

**Questão de Pesquisa:** *Como auxiliar desenvolvedores web na busca por barreiras de acessibilidade, que são encontradas por pessoas com deficiência visual que usam leitores de tela?*

Entendemos que a criação de ferramentas de simulação, como o SRS, ajude os desenvolvedores web na interpretação de resultados diagnósticos de problemas de acessibilidade gerados por ferramentas automáticas de avaliação. A prova de conceito conduzida neste estudo mostrou que o SRS foi capaz de ajudar tais avaliadores na compreensão de questões de acessibilidade para usuários cegos, ou com outra deficiência visual, envolvidas na interação dos mesmos com páginas da Web. Entendemos que a ferramenta auxiliou o avaliador vidente a compreender melhor a perspectiva de usuários do respectivo perfil a partir de dicas visuais, traduzindo os aspectos lidos por leitores de tela em elementos visuais gerados pela ferramenta.

Os resultados obtidos com as avaliações realizadas nesse trabalho de mestrado ([Capítulo 5](#)) indicam que o SRS possui potencial de reduzir as dificuldades para encontrar barreiras de acessibilidade na interpretação de resultados de avaliação de acessibilidade em páginas da Web.

Com a avaliação realizada, buscou-se validar a ideia da abordagem proposta, e evidenciar que a fundamentação teórica por trás dela é viável. Este trabalho apresentou ideias para ferramentas que possam ir além das atuais, detectando classes de erros que não são expostos pelas abordagens tradicionais. Além da detecção de novos erros, acredita-se que esta abordagem também possa auxiliar na compreensão de problemas existentes e a melhor forma de resolvê-los.

Espera-se também que o estudo da acessibilidade na Web continue progredindo e evoluindo, para que o conteúdo nela presente possa um dia ser acessível igualmente por todos os seus usuários.



## REFERÊNCIAS

---

AGHAEI, S.; NEMATBAKHSI, M. A.; FARSANI, H. K. Evolution of the world wide web: From web 1.0 to web 4.0. **International Journal of Web & Semantic Technology (IJWesT)**, Academy & Industry Research Collaboration Center(AIRCC), v. 3, n. 1, p. 1–10, Jan. 2012. ISSN 0975 - 9026. Citado na página 23.

AMERICAN FOUNDATION FOR THE BLIND. **Screen Reading Technology and Refreshable Braille Displays**. 2017. Disponível em: <<http://www.afb.org/info/living-with-vision-loss/using-technology/assistive-technology-videos/screen-reading-technology/1235>>. Acesso em: 15 fev. 2018. Citado na página 28.

ASAKAWA, C. What's the Web Like if You Can't See It? In: **Proceedings of the 2005 International Cross-Disciplinary Workshop on Web Accessibility (W4A 05)**. Chiba, Japan: ACM, 2005. p. 1–8. ISBN 1-59593-219-4. Disponível em: <<http://doi.acm.org/10.1145/1061811.1061813>>. Citado na página 29.

BAVANI, R.; JAAFAR, A.; YATIM, N. F. M. A study on web experience among visually impaired users in malaysia. In: **User Science and Engineering (i-USER), 2010 International Conference on**. [S.l.: s.n.], 2010. p. 11–15. Citado nas páginas 36 e 38.

BERNERS-LEE, T. **Information Management: A Proposal**. 1989. Disponível em: <<http://www.w3.org/History/1989/proposal.html>>. Citado na página 23.

BERNERS-LEE, T.; HENDLER, J.; LASSILA, O. The Semantic Web. **Scientific American**, v. 284, n. 5, p. 34–43, 2001. Citado na página 24.

BERSCH, R. **Introdução à Tecnologia Assistiva**. 2017. Disponível em: <[http://www.assistiva.com.br/Introducao\\_Tecnologia\\_Assistiva.pdf](http://www.assistiva.com.br/Introducao_Tecnologia_Assistiva.pdf)>. Acesso em: 10 jan. 2018. Citado na página 27.

BIGHAM, J. P. Increasing web accessibility by automatically judging alternative text quality. In: **Proceedings of the 12th International Conference on Intelligent User Interfaces**. New York, NY, USA: ACM, 2007. (IUI '07), p. 349–352. ISBN 1-59593-481-2. Disponível em: <<http://doi.acm.org/10.1145/1216295.1216364>>. Citado na página 33.

BIGHAM, J. P.; CAVENDER, A. C. Evaluating existing audio captchas and an interface optimized for non-visual use. In: **Proceedings of the SIGCHI Conference on Human Factors in Computing Systems**. New York, NY, USA: ACM, 2009. (CHI '09), p. 1829–1838. ISBN 978-1-60558-246-7. Disponível em: <<http://doi.acm.org/10.1145/1518701.1518983>>. Citado na página 36.

BIGHAM, J. P.; LIN, I.; SAVAGE, S. The effects of "not knowing what you don't know" on web accessibility for blind web users. In: **Proceedings of the 19th International ACM SIGACCESS Conference on Computers and Accessibility**. New York, NY, USA: ACM, 2017. (ASSETS '17), p. 101–109. ISBN 978-1-4503-4926-0. Disponível em: <<http://doi.acm.org/10.1145/3132525.3132533>>. Citado nas páginas 32, 61 e 62.

- BORODIN, Y.; BIGHAM, J. P.; DAUSCH, G.; RAMAKRISHNAN, I. V. More than meets the eye: A survey of screen-reader browsing strategies. In: **Proceedings of the 2010 International Cross Disciplinary Conference on Web Accessibility (W4A)**. New York, NY, USA: ACM, 2010. (W4A '10), p. 13:1–13:10. ISBN 978-1-4503-0045-2. Disponível em: <<http://doi.acm.org/10.1145/1805986.1806005>>. Citado nas páginas 34 e 55.
- BRAJNIK, G. **Barrier Walkthrough: Heuristic Evaluation Guided by Accessibility Barriers**. 2009. Disponível em: <<https://users.dimi.uniud.it/~giorgio.brajnik/projects/bw/bw.html>>. Acesso em: 1 mar. 2016. Citado nas páginas 34, 35, 36, 37, 38 e 39.
- BUZZI, M. C.; BUZZI, M.; LEPORINI, B.; MORI, G.; PENICHER, V. M. R. Accessing google docs via screen reader. **Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)**, v. 6179 LNCS, n. PART 1, p. 92–99, 2010. ISSN 03029743. Citado nas páginas 37 e 38.
- CANADIAN NATIONAL INSTITUTE FOR THE BLIND. **NVDA video tutorials**. 2015. Disponível em: <<http://www.cnib.ca/en/living/how-to-videos/tools-and-tech/Pages/The-free-NVDA-screen-reader.aspx>>. Acesso em: 10 jan. 2018. Citado na página 56.
- CANDAN, K. S.; DÖNDERLER, M. E.; HEDGPETH, T.; KIM, J. W.; LI, Q.; SAPINO, M. L. Sea: Segment-enrich-annotate paradigm for adapting dialog-based content for improved accessibility. **ACM Trans. Inf. Syst.**, ACM, New York, NY, USA, v. 27, n. 3, p. 15:1–15:45, maio 2009. ISSN 1046-8188. Disponível em: <<http://doi.acm.org/10.1145/1508850.1508853>>. Citado na página 36.
- CASTELEYN, S.; GARRIG'OS, I.; MAZ'ON, J.-N. Ten Years of Rich Internet Applications. **ACM Transactions on the Web**, v. 8, n. 3, p. 1–46, 2014. ISSN 15591131. Disponível em: <<http://dl.acm.org/citation.cfm?id=2639948.2626369>>. Citado na página 25.
- CENTRO TECNOLÓGICO DE ACESSIBILIDADE DO INSTITUTO FEDERAL RIO GRANDE DO SUL. **Manual Rápido do NVDA**. 2014. Disponível em: <[http://www.acessibilidadelegal.com/nvda/manual\\_nvda\\_ptbr.zip](http://www.acessibilidadelegal.com/nvda/manual_nvda_ptbr.zip)>. Acesso em: 15 fev. 2018. Citado na página 28.
- COOPER, M. Accessibility of emerging rich web technologies: Web 2.0 and the semantic web. In: **Proceedings of the 2007 International Cross-disciplinary Conference on Web Accessibility (W4A)**. New York, NY, USA: ACM, 2007. (W4A '07), p. 93–98. ISBN 1-59593-590-8. Disponível em: <<http://doi.acm.org/10.1145/1243441.1243463>>. Citado na página 24.
- CTIC FOUNDATION TECHNOLOGY CENTRE. **TAW - Web Accessibility and W3C Standardization Services**. 2008. Disponível em: <<https://www.tawdis.net>>. Acesso em: 27 fev. 2018. Citado na página 67.
- DOUGHERTY, B.; WADE, A. **Vischeck**. 2006. Disponível em: <<http://vischeck.com/>>. Acesso em: 1 mar. 2016. Citado na página 44.
- ENCELLE, B.; OLLAGNIER-BELDAME, M.; POUCHOT, S.; YANNICK, P. Annotation-based video enrichment for blind people : A pilot study on the use of earcons and speech synthesis. **The proceedings of the 13th international ACM SIGACCESS conference on Computers and accessibility, ASSETS '11**, n. OCTOBER, 2011. Citado na página 35.

FERREIRA, S. B. L.; NUNES, R. R.; SILVEIRA, D. S. da. Aligning Usability Requirements with the Accessibility Guidelines Focusing on the Visually-Impaired. **Procedia Computer Science**, v. 14, p. 263–273, 2012. ISSN 18770509. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S1877050912007922>>. Citado nas páginas 36, 37 e 39.

FRANÇA-FREITAS, M. L. P.; GIL, M. S. C. A. O desenvolvimento de crianças cegas e de crianças videntes. **Revista Brasileira de Educação Especial**, Scielo, v. 18, p. 507 – 526, 09 2012. ISSN 1413-6538. Disponível em: <[http://www.scielo.br/scielo.php?script=sci\\_arttext&pid=S1413-65382012000300010&nrm=iso](http://www.scielo.br/scielo.php?script=sci_arttext&pid=S1413-65382012000300010&nrm=iso)>. Citado na página 50.

FREIRE, A. P. **Acessibilidade no desenvolvimento de sistemas web: um estudo sobre o cenário brasileiro**. 136 f. Dissertação (Mestrado em Ciências de Computação e Matemática Computacional) — Instituto de Ciências Matemáticas e de Computação de São Carlos (ICMC/USP), Universidade de São Paulo, São Carlos - SP, 2008. Disponível em: <<http://www.teses.usp.br/teses/disponiveis/55/55134/tde-06052008-101644/>>. Acesso em: 26 set. 2012. Citado na página 30.

FREIRE, A. P.; GOULARTE, R.; FORTES, R. P. de M. Techniques for developing more accessible web applications: A survey towards a process classification. In: **Proceedings of the 25th Annual ACM International Conference on Design of Communication**. New York, NY, USA: ACM, 2007. (SIGDOC '07), p. 162–169. ISBN 978-1-59593-588-5. Disponível em: <<http://doi.acm.org/10.1145/1297144.1297177>>. Citado na página 30.

FUGLERUD, K. S.; TJØSTHEIM, I.; GUNNARSSON, B. R.; TOLLEFSEN, M. Use of social media by people with visual impairments: Usage levels, attitudes and barriers. In: **Proceedings of the 13th International Conference on Computers Helping People with Special Needs - Volume Part I**. Berlin, Heidelberg: Springer-Verlag, 2012. (ICCHP'12), p. 565–572. ISBN 978-3-642-31521-3. Disponível em: <[http://dx.doi.org/10.1007/978-3-642-31522-0\\_85](http://dx.doi.org/10.1007/978-3-642-31522-0_85)>. Citado nas páginas 36 e 39.

GERALDO, R. J.; FORTES, R. P. M. Dificuldades de usuários cegos na interação com a Web: uma análise sobre as pesquisas. **Revista de Sistemas de Computação - RSC**, v. 3, n. 2, p. 146–160, 2013. Citado na página 34.

GHAHARI, R. R.; FERATI, M.; YANG, T.; BOLCHINI, D. Back navigation shortcuts for screen reader users. In: **Proceedings of the 14th International ACM SIGACCESS Conference on Computers and Accessibility**. New York, NY, USA: ACM, 2012. (ASSETS '12), p. 1–8. ISBN 978-1-4503-1321-6. Disponível em: <<http://doi.acm.org/10.1145/2384916.2384918>>. Citado na página 39.

GODINHO, F.; SANTOS, C.; COUTINHO, A. F.; TRIGUEIROS, P. **Tecnologias de Informação sem Barreiras no Local de Trabalho**. 1. ed. Vila Real, 2004. Citado na página 30.

GRILLO, F. D. N.; FORTES, R. P. M. Tests with Blind Programmers Using AWMo: An Accessible Web Modeling Tool. In: **Universal Access in Human-Computer Interaction. Design and Development Methods for Universal Access**. Heraklion, Crete, Greece: Springer, 2014. p. 104–113. Citado na página 29.

HART, S. G. Nasa task load index (tlx). volume 1.0; paper and pencil package. 1986. Citado na página 64.

- \_\_\_\_\_. Nasa-task load index (nasa-tlx); 20 years later. **Proceedings of the Human Factors and Ergonomics Society Annual Meeting**, v. 50, n. 9, p. 904–908, 2006. Disponível em: <<https://doi.org/10.1177/154193120605000909>>. Citado na página 64.
- HART, S. G.; STAVENLAND, L. E. Development of nasa-tlx (task load index): Results of empirical and theoretical research. In: **Human Mental Workload**. Elsevier, 1988. cap. 7, p. 139–183. Disponível em: <[http://ntrs.nasa.gov/archive/nasa/casi.ntrs.nasa.gov/20000004342/\\_1999205624.pdf](http://ntrs.nasa.gov/archive/nasa/casi.ntrs.nasa.gov/20000004342/_1999205624.pdf)>. Citado na página 64.
- INTERNATIONAL CENTER FOR DISABILITY RESOURCES ON THE INTERNET. **CynthiaSays**. 2003. Disponível em: <<http://www.cynthiasays.com>>. Acesso em: 27 fev. 2018. Citado na página 67.
- IVORY, M.; HEARST, M. The state of the art in automating usability evaluation of user interfaces. **ACM Computing Surveys**, v. 33, n. 4, p. 470–516, 2001. ISSN 03600300. Disponível em: <<http://portal.acm.org/citation.cfm?doid=503112.503114>>. Citado nas páginas 25, 41 e 71.
- KELLY, B.; SLOAN, D.; PHIPPS, L.; PETRIE, H.; HAMILTON, F. Forcing standardization or accommodating diversity?: A framework for applying the wcag in the real world. In: **Proceedings of the 2005 International Cross-Disciplinary Workshop on Web Accessibility (W4A)**. New York, NY, USA: ACM, 2005. (W4A '05), p. 46–54. ISBN 1-59593-219-4. Disponível em: <<http://doi.acm.org/10.1145/1061811.1061820>>. Citado na página 30.
- KIRAKOWSKI, J.; CLARIDGE, N. **UsabilityNet**. 2012. Disponível em: <<http://www.wammi.com/usabilitynet/>>. Acesso em: 1 mar. 2016. Citado na página 64.
- KRANTZ, P. **Fangs - the screen reader emulator**. 2004. Disponível em: <<http://www.standards-schmandards.com/projects/fangs/>>. Acesso em: 1 mar. 2016. Citado na página 45.
- LUCCA, G. D.; FASOLINO, A.; TRAMONTANA, P. Web site accessibility: Identifying and fixing accessibility problems in client page code. In: **Seventh IEEE International Symposium on Web Site Evolution**. IEEE, 2005. p. 71–78. ISBN 0-7695-2470-2. ISSN 1550-4441. Disponível em: <<http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=1517983>>. Citado na página 29.
- LUNN, D.; HARPER, S.; BECHHOFER, S. Identifying behavioral strategies of visually impaired users to improve access to web content. **ACM Trans. Access. Comput.**, ACM, New York, NY, USA, v. 3, n. 4, p. 13:1–13:35, abr. 2011. ISSN 1936-7228. Disponível em: <<http://doi.acm.org/10.1145/1952388.1952390>>. Citado na página 38.
- MAIDENBAUM, S.; CHEBAT, D.-R.; LEVY-TZEDEK, S.; AMEDI, A. Vision-deprived virtual navigation patterns using depth cues & the effect of extended sensory range. **Proceedings of the extended abstracts of the 32nd annual ACM conference on Human factors in computing systems - CHI EA '14**, p. 1231–1236, 2014. Disponível em: <<http://dl.acm.org/citation.cfm?doid=2559206.2581343>>. Citado na página 34.
- MATERA, M.; RIZZO, F.; CARUGHI, G. Web usability: Principles and evaluation methods. In: **Web Engineering**. Springer Berlin Heidelberg, 2006. p. 143–180. ISBN 978-3-540-28196-2. Disponível em: <[http://dx.doi.org/10.1007/3-540-28218-1\\_5](http://dx.doi.org/10.1007/3-540-28218-1_5)>. Citado na página 27.
- NICÁCIO, J. M. **Técnicas de acessibilidade: criando uma Web para todos**. Maceió - AL: Edufal - Editora da Universidade Federal de Alagoas, 2010. 100 p. Citado na página 30.

NOCOFFEE - VISION SIMULATION FOR CHROME. **NoCoffee - Vision Simulation**. 2013. Disponível em: <<https://accessgarage.wordpress.com/2013/02/09/458/>>. Acesso em: 1 mar. 2016. Citado na página 47.

NÚCLEO DE ATENDIMENTO ÀS PESSOAS COM NECESSIDADES ESPECIAIS. **Manual - Leitor de Tela NVDA**. 2009. Disponível em: <[http://www.londrina.pr.gov.br/dados/images/stories/Storage/cmdpd/NVDA/manual\\_NVDA.pdf](http://www.londrina.pr.gov.br/dados/images/stories/Storage/cmdpd/NVDA/manual_NVDA.pdf)>. Acesso em: 10 jan. 2018. Citado na página 56.

PEREIRA, L. S.; FERREIRA, S. B. L.; ARCHAMBAULT, D. Preliminary Web Accessibility Evaluation Method through the Identification of Critical Items with the Participation of Visually Impaired Users. **Procedia Computer Science**, Elsevier Masson SAS, v. 67, n. Dsai, p. 77–86, 2015. ISSN 18770509. Disponível em: <<http://linkinghub.elsevier.com/retrieve/pii/S1877050915030975>>. Citado nas páginas 59 e 60.

PETRIE, H.; KHEIR, O. The relationship between accessibility and usability of websites. In: **Proceedings of the SIGCHI Conference on Human Factors in Computing Systems**. New York, NY, USA: ACM, 2007. (CHI '07), p. 397–406. ISBN 978-1-59593-593-9. Disponível em: <<http://doi.acm.org/10.1145/1240624.1240688>>. Citado nas páginas 27 e 31.

POWER, C.; FREIRE, A.; PETRIE, H.; SWALLOW, D. Guidelines are only half of the story: Accessibility problems encountered by blind users on the web. In: **Proceedings of the SIGCHI Conference on Human Factors in Computing Systems**. New York, NY, USA: ACM, 2012. (CHI '12), p. 433–442. ISBN 978-1-4503-1015-4. Disponível em: <<http://doi.acm.org/10.1145/2207676.2207736>>. Citado nas páginas 32, 33, 36, 37 e 38.

POWER, M. **Delivering Web to Mobile**. Bath, Somerset, UK, 2012. Disponível em: <<http://opus.bath.ac.uk/34846/>>. Citado na página 24.

RUTTER, R.; LAUKE, P.; WADDELL, C.; THATCHER, J.; HENRY, S.; LAWSON, B.; KIRKPATRICK, A.; HEILMANN, C.; BURKS, M.; REGAN, B. *et al.* **Web Accessibility: Web Standards and Regulatory Compliance**. Apress, 2006. (Apresspod Series). ISBN 9781590596388. Disponível em: <<http://books.google.com.br/books?id=dIJ94KZqwqcC>>. Citado na página 29.

SAVIDIS, A.; STEPHANIDIS, C. The HOMER UIMS for dual user interface development: Fusing visual and non-visual interactions. v. 11, p. 173–209, 1998. Citado na página 33.

SDH/PR, S. de Direitos Humanos da Presidência da R. **Tecnologia Assistiva**. 1. ed. [S.l.]: Comitê de Ajudas Técnicas, 2009. 138 p. Citado na página 28.

SHNEIDERMAN, B. Universal usability. **Commun. ACM**, ACM, New York, NY, USA, v. 43, n. 5, p. 84–91, maio 2000. ISSN 0001-0782. Disponível em: <<http://doi.acm.org/10.1145/332833.332843>>. Citado nas páginas 27 e 59.

\_\_\_\_\_. Promoting universal usability with multi-layer interface design. **SIGCAPH Comput. Phys. Handicap.**, ACM, New York, NY, USA, n. 73-74, p. 1–8, jun. 2002. ISSN 0163-5727. Disponível em: <<http://doi.acm.org/10.1145/960201.957206>>. Citado na página 27.

STATEWIDE VISION RESOURCE CENTRE. **JAWS video tutorials**. 2013. Disponível em: <<http://svrc.vic.edu.au/tutorials/windows/videos-jaws-basics/>>. Acesso em: 10 jan. 2018. Citado na página 56.

TAKAGI, H.; ASAKAWA, C.; FUKUDA, K.; MAEDA, J. Accessibility designer: visualizing usability for the blind. **ACM SIGACCESS Accessibility and Computing**, n. 77-78, p. 177, 2003. ISSN 15582337. Disponível em: <<http://dl.acm.org/citation.cfm?id=1029014.1028662>>. Citado nas páginas 41, 43, 47 e 50.

TANGARIFE, T. M. **A acessibilidade nos websites governamentais: um estudo de caso do site da ELETROBRÁS**. 394 f. Dissertação (Mestrado em Artes e Design) — Pontifícia Universidade Católica do Rio de Janeiro (PUC-RIO), Rio de Janeiro - RJ, 2007. Disponível em: <<http://migre.me/cqVHA>>. Acesso em: 26 set. 2012. Citado nas páginas 30 e 31.

TANGARIFE, T. M.; MONT'ALVÃO, C. Acessibilidade web: Um estudo exploratório do conhecimento do desenvolvedor web brasileiro. **III Seminário, II Oficina e II Mostra "Acessibilidade, TI e Inclusão Digital"**, São Paulo, SP, set. 2005. Disponível em: <[http://www.prodiam.sp.gov.br/multimedia/midia/cd\\_atiid/conteudo/ATIID2005/MR2/01/AcessibilidadeWeb.pdf](http://www.prodiam.sp.gov.br/multimedia/midia/cd_atiid/conteudo/ATIID2005/MR2/01/AcessibilidadeWeb.pdf)>. Citado na página 30.

\_\_\_\_\_. Estudo comparativo utilizando uma ferramenta de avaliação de acessibilidade para web. In: **Proceedings of the 2005 Latin American conference on Human-computer interaction**. New York, NY, USA: ACM, 2005. (CLIHC '05), p. 313–318. ISBN 1-59593-224-0. Disponível em: <<http://doi.acm.org/10.1145/1111360.1111394>>. Citado na página 30.

THATCHER, J.; WADDELL, C.; BURKS, M. **Constructing accessible web sites**. 1. ed. San Francisco, CA: Glasshaus Birmingham, 2003. ISBN 1590591488. Citado na página 27.

W3C WEB ACCESSIBILITY INITIATIVE. **Accessible Rich Internet Applications**. 2006. Disponível em: <<https://www.w3.org/WAI/intro/aria>>. Acesso em: 1 mar. 2016. Citado na página 25.

WEBAIM - WEB ACCESSIBILITY IN MIND. **WebAIM - Web Accessibility In Mind**. 1999. Disponível em: <<http://webaim.org/>>. Acesso em: 1 mar. 2016. Citado na página 45.

WILLIAMS, M. a.; GALBRAITH, C.; KANE, S. K.; HURST, A. "Just Let the Cane Hit It": How the Blind and Sighted See Navigation Differently. **Proceedings of the 16th international ACM SIGACCESS conference on Computers & accessibility - ASSETS '14**, p. 217–224, 2014. Disponível em: <<http://dl.acm.org/citation.cfm?id=2661334.2661380>>. Citado na página 33.

WORLD WIDE WEB CONSORTIUM. **Authoring Tool Accessibility Guidelines 1.0**. 2000. Disponível em: <<http://www.w3.org/TR/ATAG10/>>. Acesso em: 14 dez. 2012. Citado na página 31.

\_\_\_\_\_. **User Agent Accessibility Guidelines 1.0**. 2002. Disponível em: <<http://www.w3.org/TR/UAAG10/>>. Acesso em: 14 dez. 2012. Citado na página 31.

\_\_\_\_\_. **Essential Components of Web Accessibility**. 2005. Disponível em: <<http://www.w3.org/WAI/intro/components.php>>. Acesso em: 14 dez. 2013. Citado nas páginas 29 e 30.

\_\_\_\_\_. **A vocabulary and associated APIs for HTML and XHTML**. 2012. Disponível em: <<http://www.w3.org/TR/2012/CR-html5-20121217/>>. Acesso em: 30 jan. 2013. Citado na página 31.

---

ZHAO, H.; PLAISANT, C.; SHNEIDERMAN, B.; LAZAR, J. Data Sonification for Users with Visual Impairment. **ACM Transactions on Computer-Human Interaction**, v. 15, n. 1, p. 1–28, 2008. ISSN 10730516. Disponível em: <<http://dl.acm.org/citation.cfm?id=1352782.1352786>>. Citado na página 37.



---

## GLOSSÁRIO

---

---

**API:** *Application Programming Interface*, termo utilizado para representar o “protocolo” definido em um programa que permite que outros programas interajam com ele. Por exemplo, o sistema operacional de um computador expõe várias APIs para os programas nele instalados executarem diversas funcionalidades em comum, como apresentar janelas, botões, mensagens ao usuário, mecanismos de entrada (mouse, teclado), etc.

**HTML:** *HyperText Markup Language*, a linguagem de marcação utilizada para a criação de páginas *web*. Esta linguagem provê marcas estruturais e semânticas, a qual são colocadas ao redor do conteúdo, com o objetivo tanto de dar uma aparência visual a este conteúdo (estruturando o seu *layout*), quanto um significado semântico (marcando parágrafos, ênfases, tabelas, formulários, etc.).

**Web:** Sinônimo mais conhecido de *World Wide Web* (WWW). É a interface gráfica da Internet que torna os serviços disponíveis totalmente transparentes para o usuário e ainda possibilita a manipulação multimídia da informação.

**WYSIWYG:** “What You See Is What You Get” ou “O que você vê é o que você obtém”. Recurso tem por objetivo permitir que um documento, enquanto manipulado na tela, tenha a mesma aparência de sua utilização, usualmente sendo considerada final. Isso facilita para o desenvolvedor que pode trabalhar visualizando a aparência do documento sem precisar salvar em vários momentos e abrir em um *software* separado de visualização.

