

AVALIAÇÃO AUTOMÁTICA DE ACESSIBILIDADE DO PORTAL DA UFRN

Automatic Accessibility Evaluation of the UFRN Portal

Jefferson Ferreira Ribeiro*

Bruno Santana da Silva**

Resumo: A inclusão de pessoas com deficiência no ensino superior tem destacado a necessidade da acessibilidade nos sistemas computacionais interativos utilizados nesse ambiente. A nova versão do Portal da Universidade Federal do Rio Grande do Norte acompanhou esse movimento, considerando seu uso com leitores de tela e VLibras. Entretanto, sua acessibilidade ainda não havia sido avaliada. Este trabalho tem por objetivo identificar problemas de acessibilidade nesse Portal para subsidiar melhorias futuras na experiência de uso de pessoas com deficiência. Em junho de 2019, foi realizada uma avaliação por inspeção com os avaliadores automáticos de acessibilidade *AInspector*, *HTML Code Sniffer* e *Siteimprove Accessibility Checker*. Foram analisadas quantidade total, classificação como erros e avisos, princípios e critérios violados e o local de ocorrência dos problemas encontrados. Foi possível identificar oportunidades de melhoria do Portal nos aspectos de percepção, operação, compreensão e robustez de acessibilidade nas quatro páginas analisadas: “home”, “acadêmico”, “todas as notícias” e na página de uma notícia detalhada.

Palavras-chave: acessibilidade. avaliação de interface. avaliador automático. wcag

Abstract: The inclusion of people with disabilities in higher education has highlighted the need of accessibility of interactive computational systems used in this context. The new version of the Portal of Rio Grande do Norte Federal University (UFRN) followed this movement, considering its use through screen readers VLibras. However, its accessibility was not evaluated yet. This work aims to identify accessibility problems in this Portal and from that to support future improvements on user experience of people with disabilities. In June 2019, an evaluation by inspection was conducted with the automatic evaluators of accessibility *AInspector*, *HTML Code Sniffer* e *Siteimprove Accessibility Checker*. It analyzed the total amount, error and warning classification, principles and criteria violation and occurrence location of found problems. We have identified improvement opportunities in Portal on aspects of perception, operation, understanding and robustness of accessibility in the four analyzed pages: home, academic, all the news and a detailed news page.

Keywords: accessibility. user interface evaluation. automatic evaluation. wcag.

* Graduando em Tecnologia da Informação pela Universidade Federal do Rio Grande do Norte. E-mail: jfrsnr@gmail.com.

** Professor Doutor do Instituto Metrópole Digital da Universidade Federal do Rio Grande do Norte. <https://orcid.org/0000-0002-7689-8000>. E-mail: bruno@imd.ufrn.br

1 Introdução

Apesar das dificuldades em lidar com o diferente, a sociedade passa por movimentos de reconhecimento, aceitação e valorização das diferenças humanas. Pessoas que possuem limitações eram deixadas à margem das atividades sociais e civis. Aos poucos, as pessoas com deficiência têm recebido a atenção que merecem e sido favorecidas por esforços de inclusão na sociedade para participação autônoma e plena nas atividades e convívio social. Leis que garantem os direitos de pessoas com deficiência têm sido marcos importantes para promover a inclusão. O Brasil ratificou o acordo da Convenção sobre os Direitos das Pessoas com Deficiência das Nações Unidas por meio do Decreto Legislativo nº 186, de 9 de julho (BRASIL, 2008). A legislação brasileira tem sido atualizada, a exemplo da Lei 13.146 (Lei Brasileira da Inclusão), para contemplar a inclusão de pessoas com deficiência na educação, cultura, lazer, saúde, trabalho e em outros aspectos (BRASIL, 2015).

Os instrumentos legais e políticos na educação brasileira promovem a inclusão de pessoas com deficiência desde o ensino fundamental até o superior. Esses esforços de inclusão na educação evoluíram ao ponto de (1) existirem cotas para pessoas com deficiência nas seleções de ingresso em instituições federais de ensino superior amparadas pela Lei nº 13.409, de 28 de dezembro (BRASIL, 2016) e de (2) o instrumento de avaliação do INEP (BRASIL, 2017) para reconhecimento de cursos de graduação possuir critérios explícitos de acessibilidade.

Há mais de uma década, a Universidade Federal do Rio Grande do Norte (UFRN) tem desenvolvido iniciativas para promover a inclusão de pessoas com deficiência nos seus cursos e atividades. Melo, Silva e Andrade (2019) apresentam um panorama recente dos avanços e desafios da UFRN em cada dimensão de acessibilidade: programática, arquitetônica, instrumental, comunicacional, digital, metodológica e atitudinal. No âmbito digital, a universidade desenvolve internamente muitos dos sistemas computacionais que utiliza, incluindo os sites que divulgam ações da instituição. Gradualmente, os processos de design e desenvolvimento de sistemas computacionais na instituição estão empregando práticas do design inclusivo para promover a acessibilidade digital (W3C BRASIL, 2013). Por exemplo, o portal da UFRN (UFRN, 2020) foi atualizado recentemente (GRILO et al., 2017) tendo em vista, dentre outros fatores, seu uso com leitores de tela e com o VLibras. Entretanto, a acessibilidade desse portal ainda não havia sido avaliada.

Este trabalho visa contribuir com esforços de inclusão de pessoas com deficiência na UFRN através da avaliação da acessibilidade do portal *web* da instituição. Essa análise é importante para identificar barreiras que pessoas com deficiência podem encontrar ao interagir com o portal. No futuro, os problemas identificados podem subsidiar melhorias no portal.

2 Acessibilidade Web

A acessibilidade torna possível que qualquer pessoa alcance, perceba, entenda e interaja com sistemas web para utilização, participação e contribuição, com autonomia e segurança, independentemente de sua capacidade motora, visual, auditiva, intelectual e social (W3C BRASIL, 2013). Um sistema web acessível possui uma interface com usuário que não impõe obstáculos (ou barreiras) a usuários com diferentes capacidades motoras, de sentido e cognitivas (BARBOSA; SILVA, 2010).

A organização internacional *World Wide Web Consortium* (W3C, 1994) define padrões para programação (e.g. HTML, CSS, JavaScript) e diretrizes de acessibilidade web. Essas diretrizes são conhecidas pelo acrônimo WCAG, de *Web Content Accessibility Guidelines*.

Apesar de a versão 2.1 da WCAG ter sido publicada em 2018 (W3C, 2018), a versão 2.0 da WCAG publicada em 2008 ainda deve ser a mais conhecida e utilizada (W3C, 2008). Essa tem sido a principal orientação para projeto, desenvolvimento e avaliação de sistemas web acessíveis.

A WCAG (W3C, 2008) organiza suas diretrizes em termos de quatro princípios na perspectiva de quem usa sistemas web: "perceptível", "operável", "compreensível" e "robusto". Cada princípio deu origem a recomendações que, por sua vez, são detalhadas por critérios e especificadas por técnicas. Por exemplo, o Princípio 1, "perceptível", está associado à Recomendação 1.1: "fornecer alternativas textuais para qualquer conteúdo não textual" e à Recomendação 1.2: "fornecer alternativas para mídias baseadas em tempo", dentre outras. A Recomendação 1.1 é detalhada pelo Critério 1.1.1: "Todo o conteúdo não textual que é exibido ao usuário tem uma alternativa textual", e assim por diante. A numeração dos princípios, as recomendações e os critérios são definidos de forma sistemática e utilizados como índice nos documentos da W3C. Por exemplo, a Diretriz 1.2.3 da WCAG representa o Princípio 1, a Recomendação 2 e o Critério 3. Essa numeração também é utilizada ao longo deste trabalho.

Parte das diretrizes da WCAG pode ser verificada diretamente no código-fonte em HTML e CSS resultante da programação dos sistemas web. Outra parte precisa de interpretações e análises humanas dos envolvidos (usuários com e sem deficiência, designer, desenvolvedor, etc.) sobre o que ocorre durante o uso do sistema computacional. Ainda que a análise do código-fonte seja limitada, ela representa um avanço significativo na promoção da acessibilidade digital na web. Essa verificação do código-fonte pode ser realizada manualmente por inspeção de um avaliador humano, como também pode ser realizada de forma automática por inspeção de um software (BARBOSA; SILVA, 2010). Os softwares avaliadores automáticos de código-fonte da interface tendem a reduzir o esforço e o tempo necessários para execução da avaliação de acessibilidade, bem como tendem a evitar ruídos na avaliação causados por falta de atenção, esquecimento ou equívocos. Desse modo, softwares avaliadores automáticos de acessibilidade podem ser responsáveis pelo trabalho básico, pesado e até certo ponto mecânico da avaliação de acessibilidade; enquanto as pessoas podem se concentrar em um conjunto menor de atividades de avaliação mais sofisticadas intelectualmente.

Trabalhos anteriores também utilizaram softwares avaliadores automáticos baseados na WCAG para avaliar a acessibilidade de sistemas web. Por exemplo, Tangarife e Mont'Alvão (2005) avaliaram o portal da Eletrobrás com o avaliador automático de acessibilidade "da Silva". Pivetta e colegas (2013) analisaram parte do Moodle com avaliador automático WAVE. Mezzaroba e colegas (2016) avaliaram a acessibilidade do portal do Supremo Tribunal Federal com o software ASES.

3 Avaliação do Portal da UFRN

O objetivo deste trabalho é avaliar a acessibilidade do portal da UFRN (UFRN, 2020) através da inspeção de softwares avaliadores automáticos de código-fonte. A W3C (1994) apresenta uma lista de avaliadores automáticos classificados em sete tipos (W3C, 2016). Os tipos com maior quantidade de exemplares são ferramentas online, extensões de navegadores web e ferramentas desktop, respectivamente. Optou-se por utilizar extensões de navegadores web pela facilidade de instalação comparado a ferramentas desktop (basta utilizar uma loja de extensões do navegador ou um script salvo nos favoritos do navegador) e pelo maior controle sobre os dados em comparação a ferramentas online (os dados podem ser processados localmente). Ribeiro e Silva (2019) fizeram um estudo comparativo dos avaliadores automáticos que são extensão de navegadores web. Os três avaliadores mais promissores nesse



estudo anterior foram escolhidos para avaliar o portal da UFRN: *AInspector*, *HTML Code Sniffer* e *Siteimprove Accessibility Checker*. Todos eles funcionam com base na WCAG 2.0 (W3C, 2008). A Tabela 1 detalha o que foi utilizado nesta avaliação do portal: nome completo e versão dos avaliadores automáticos, navegadores web e um nome resumido para referenciar o avaliador nas figuras e tabelas com os resultados da avaliação.

Tabela 1 - Avaliadores automáticos de acessibilidade utilizados na avaliação do Portal.

Ferramenta	Versão	Navegador	Nome resumido
AInspector	0.95.0	Firefox	ainspector
Html_CodeSniffer	2.1.1	Chrome	code sniffer
Siteimprove Accessibility Checker	117	Chrome	accessibility checker

Fonte: Os autores.

O portal da UFRN possui muitas páginas. Apesar de a *home* ser uma página importante e possuir elementos diversificados, optou-se por avaliar um total de quatro páginas do portal para ampliar a abrangência e a diversidade dos elementos de interface considerados. A Tabela 2 indica as páginas do portal da UFRN analisadas neste estudo.

Tabela 2 - Páginas avaliadas no portal da UFRN.

Página	Funcionalidade
<i>home</i>	Apresenta uma visão geral da universidade, com espaço destacado para últimas notícias, eventos, matérias especiais e vídeos de divulgação.
acadêmico	Apresenta as principais áreas de atuação na universidade. Possui elementos textuais, fotos e <i>links</i> para mais informações.
notícias	Lista as notícias da universidade com paginação e permite busca por palavra-chave, data, <i>tags</i> . Permite o cadastro de email para receber as novidades.
notícia	Detalha uma notícia da universidade, com títulos, <i>tags</i> , textos e vídeo.

Fonte: Os autores.

Em junho de 2019, cada um dos três avaliadores automáticos selecionados (Tabela 1) foi executado para cada uma das quatro páginas analisadas do portal da UFRN (Tabela 2). Os avaliadores apresentaram uma lista de resultados detalhados. Por exemplo, a Figura 1 ilustra um dos problemas listados pelo *Html_CodeSniffer* na página de uma notícia.

Os problemas de acessibilidade encontrados foram tabulados em termos de avaliador, página, local de ocorrência dentro da página, diretriz da WCAG violada e tipo do problema encontrado (erro ou aviso). Considerou-se também como avisos os *minor checkers* do *AInspector* e os *reviews* do *SiteImprove Accessibility Checker*. Conforme os problemas foram sendo tabulados, o local de ocorrência de cada problema foi identificado manualmente por números sobre uma imagem da página avaliada. A mesma numeração dos locais em cada página foi utilizada para indicar problemas encontrados por todos os avaliadores. Os dados tabulados foram inseridos em um banco de dados para facilitar análises quantitativas por meio de consultas SQL. A análise dos dados foi fundamentalmente quantitativa com a contabilização de dados estratificados e de porcentagens. Isso permitiu a construção de gráficos e tabelas apresentados e discutidos a seguir.



Figura 1 – Exemplo de problema de acessibilidade encontrado pelo Html_CodeSniffer na página de notícia.

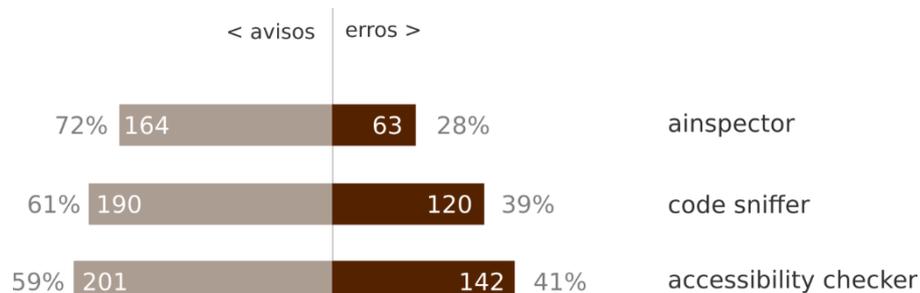


Fonte: Os autores.

4 Resultados

A Figura 2 apresenta a quantidade de problemas encontrados pelos avaliadores automáticos em todas as quatro páginas analisadas em junho de 2019. A quantidade de avisos encontra-se no lado esquerdo do gráfico (parte mais clara), enquanto a quantidade de erros foi indicada no lado direito (parte mais escura). O *AInspector* foi o avaliador que encontrou a menor quantidade de problemas no portal, 63 erros, seguido pelo *Code Sniffer*, com 120 erros, e depois pelo *Accessibility Checker*, com 142 erros. O somatório foi de 325 erros encontrados, provavelmente contendo repetições entre avaliadores. Os avisos relatados tiveram comportamentos similares, apesar de valores absolutos maiores. O *AInspector* apresentou 164 avisos, seguido pelo *Code Sniffer*, com 190 problemas e do *Accessibility Checker*, com 201 problemas. O somatório de avisos reportados foi de 555, também provavelmente contendo repetições entre os avaliadores.

Figura 2 - Quantidade total de avisos e erros encontrados por cada avaliador automático de acessibilidade nas quatro páginas analisadas.

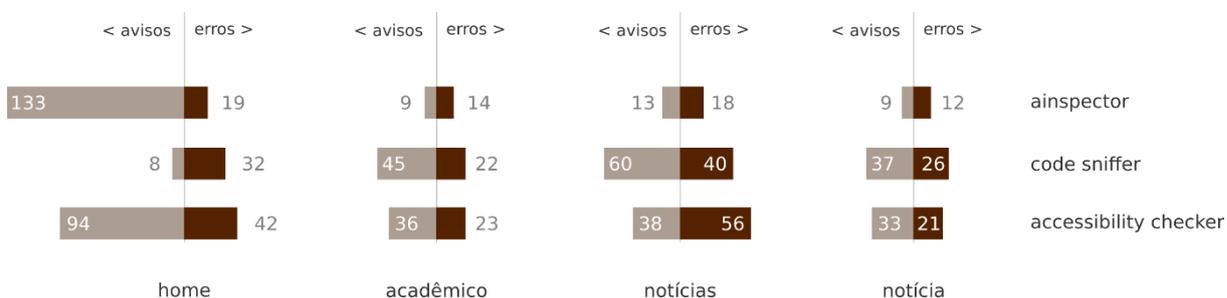


Fonte: Os autores.



A Figura 3 apresenta a quantidade estratificada dos avisos e erros encontrados em cada página por cada avaliador automático. Em geral, a quantidade dos erros não apresentou uma variação muito grande entre as páginas de um mesmo avaliador, exceto alguns destaques pontuais na página de notícias e na página *home*. Já os avisos apresentaram variação mais significativa nos resultados de um mesmo avaliador, principalmente na página *home*. As páginas que apresentaram maior quantidade de erros no total, provavelmente com repetições entre avaliadores, foram notícias, *home*, notícia e acadêmico, respectivamente. A ordem muda quando se somam os avisos: *home*, notícias, acadêmico e notícia.

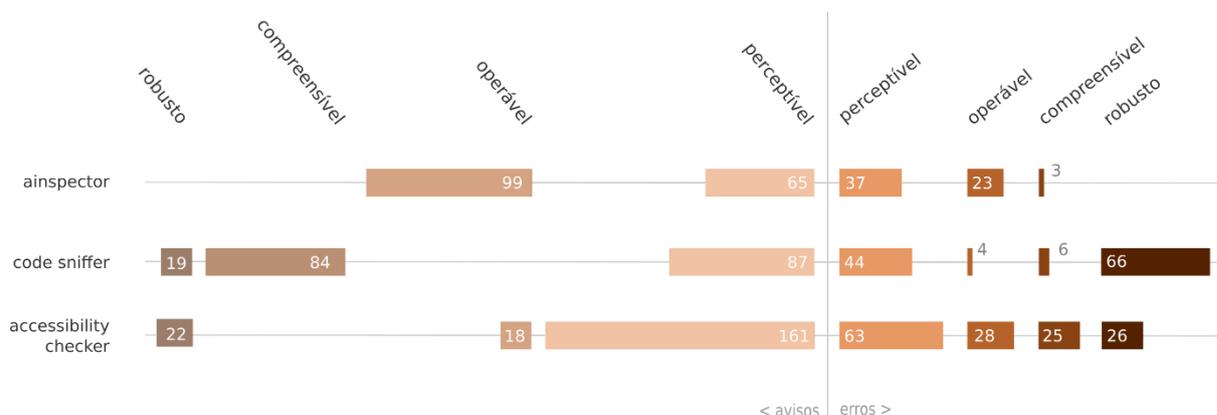
Figura 3 - Quantidade de avisos e erros encontrados pelos avaliadores automáticos de acessibilidade estratificada pelas quatro páginas analisadas.



Fonte: Os autores.

Depois de analisar a quantidade de erros e avisos por página, foram identificados quais princípios da WCAG foram violados. Para cada avaliador, a Figura 4 apresenta, à esquerda, o total de avisos estratificada pelo princípio violado e, à direita, o total de erros de modo análogo. Em geral, a maioria dos erros identificados apontam para barreiras à percepção do usuário. A seguir, aparecem os erros de robustez, de operação e de compreensão, nessa ordem. Os avisos indicam uma possibilidade de barreiras à percepção do usuário na operação, na compreensão e na robustez do sistema, respectivamente.

Figura 4 - Quantidade de avisos e erros encontrados por princípio em cada avaliador.



Fonte: Os autores.



Esses resultados foram estratificados na Figura 5 para indicar a quantidade de problemas que violaram cada critério da WCAG. No princípio "perceptível", os erros concentraram-se principalmente nos critérios 1.1.1, 1.3.1 e 1.4.6, e os avisos concentram-se em particular nos critérios 1.3.1, 1.4.6 e 1.4.5. No princípio "operável", a principal concentração dos erros ocorreu nos critérios 2.4.4 e 2.4.9 e 2.4.1 e dos avisos ocorreu nos critérios 2.4.4, 2.4.6 e 2.4.3. Apenas no princípio "compreensível" os erros ocorreram em critérios completamente distintos dos avisos. Os erros ocorreram nos critérios 3.2.2 e 3.3.2 e os avisos apenas no critério 3.2.5. No princípio "robusto", erros e avisos se concentraram no critério 4.1.2, apesar de existirem erros também que violaram o critério 4.1.1. Além desses destaques, é interessante observar que também foram identificados avisos em vários outros critérios dos princípios "perceptível" e "operável".

Figura 5 - Quantidade de avisos e erros encontrados por critério em cada avaliador.

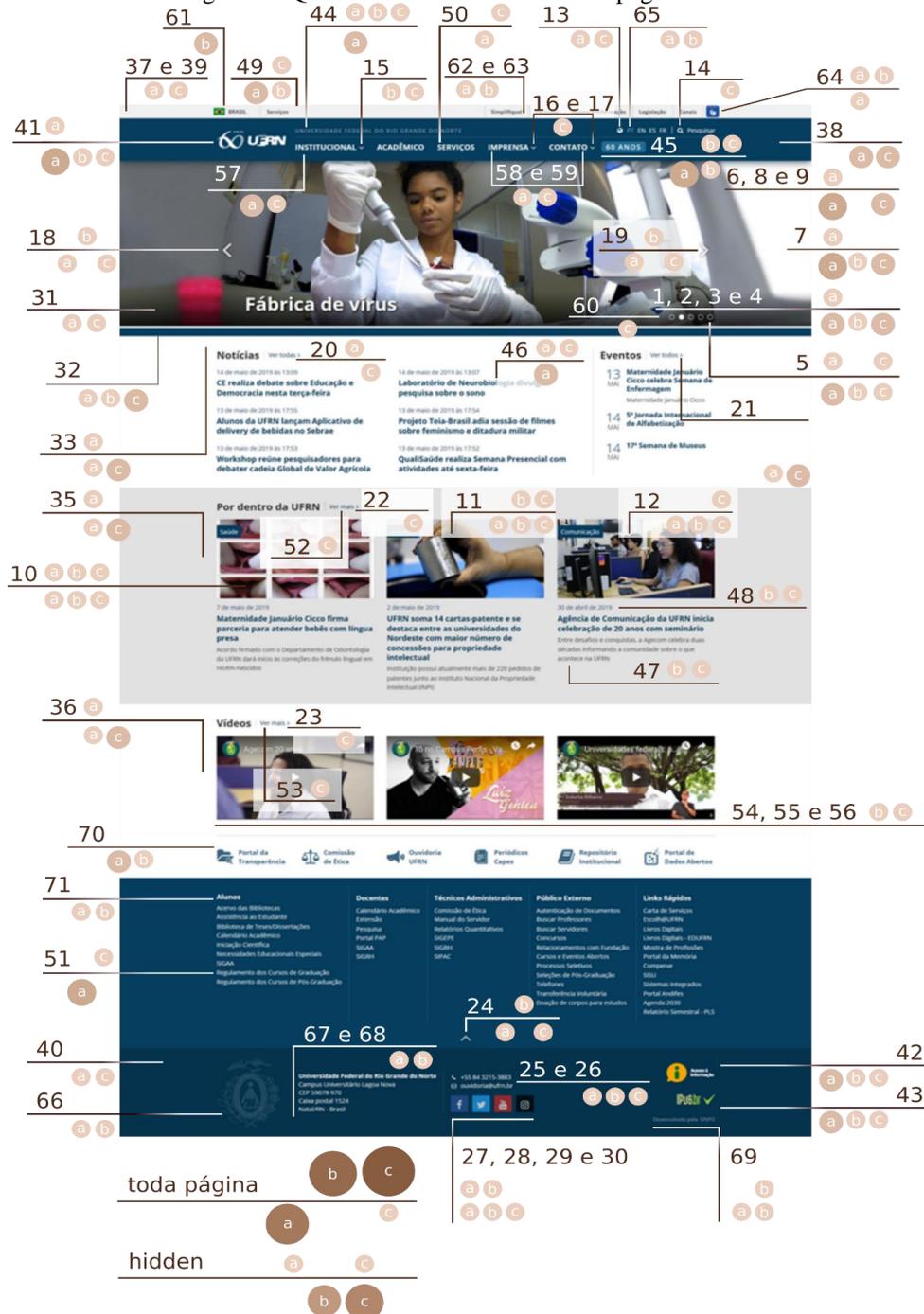
	avisos			erros		
		code sniffer	accessibility checker	code sniffer	accessibility checker	ainspector
perceptível	1.1.1		6 9	11 12		
	1.3.1	21	129 24	13 27 35		
	1.3.2		18			
	1.3.3		1			
	1.4.1		1	5		
	1.4.2		1			
	1.4.3	1	10	4 2		
	1.4.4		1			
	1.4.5		26			
	1.4.6	65		20 15		
operável	2.1.1		13			
	2.2.1	4	1			
	2.2.2		6			
	2.3.1		5			
	2.4.1	4	1	4 1 5		
	2.4.2		1			
	2.4.3	10	10			
	2.4.4		31	13 18		
	2.4.5		1			
	2.4.6		20	2		
2.4.7		10				
2.4.9			12			
compreensível	3.2.2			6 6		
	3.2.5	84				
	3.3.2			19 3		
robusto	4.1.1			2 4		
	4.1.2	19 22		64 22		

Fonte: Os autores.



As Figuras 5, 6, 7 e 8 indicam o local de ocorrência dos erros e avisos encontrados em cada página analisada. O local é indicado por um número sequencial com um compartimento superior para os erros e inferior para os avisos. A quantidade de erros e avisos é representada pelo diâmetro e cor do círculo contendo uma letra. Quanto maior e mais escuro for o círculo, maior será a quantidade de ocorrências de barreiras de acessibilidade naquele local. Os avaliadores automáticos foram indicados por uma letra dentro do círculo: A para o AInspector, B para Code Sniffer e C para o Site Improve. Os círculos estão organizados por colunas (um espaço vertical) para cada avaliador. Assim, fica mais fácil identificar e comparar a ocorrência de erros e avisos do mesmo avaliador.

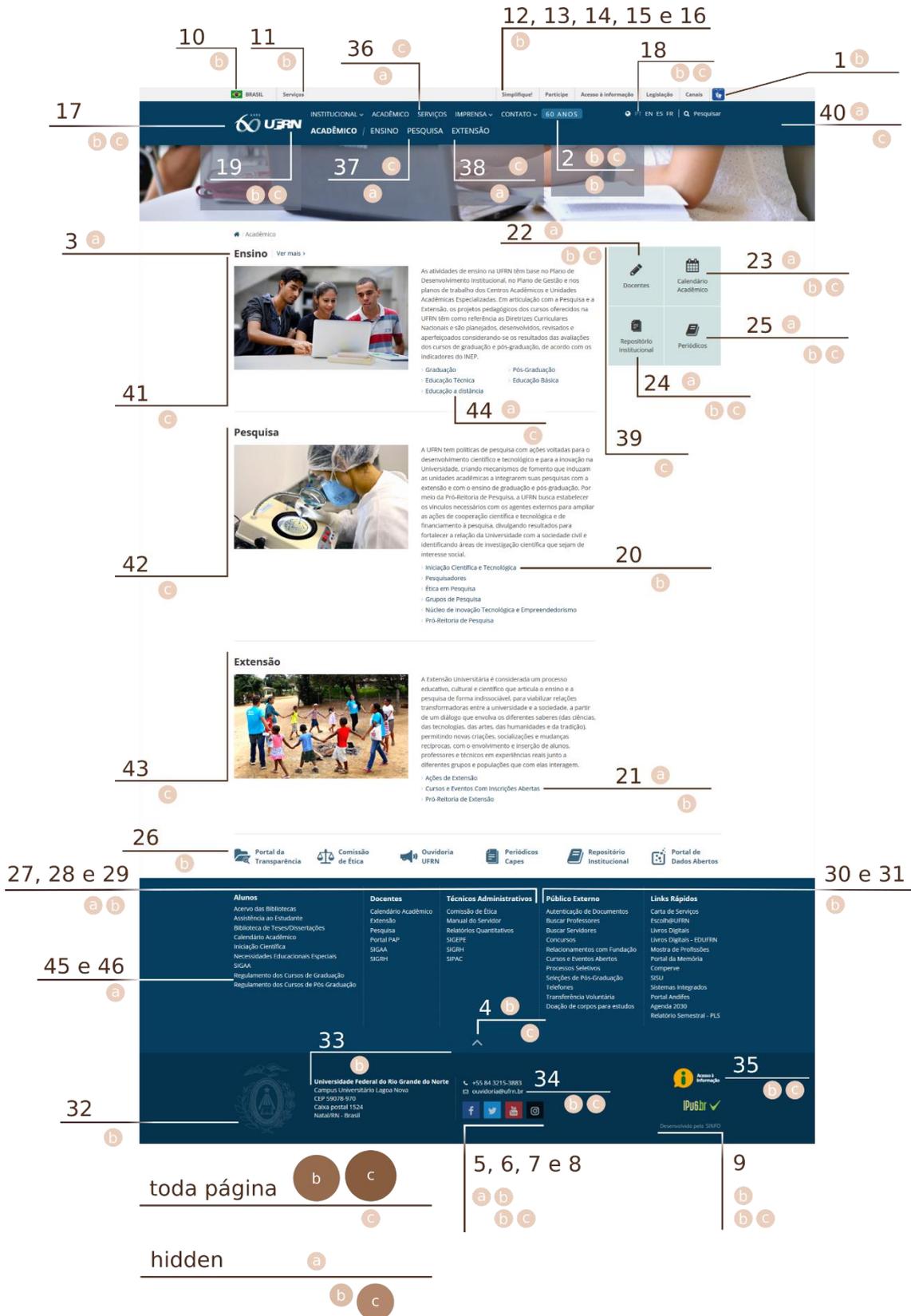
Figura 6 - Quantidade de avisos e erros na página *home*.



Fonte: Os autores.



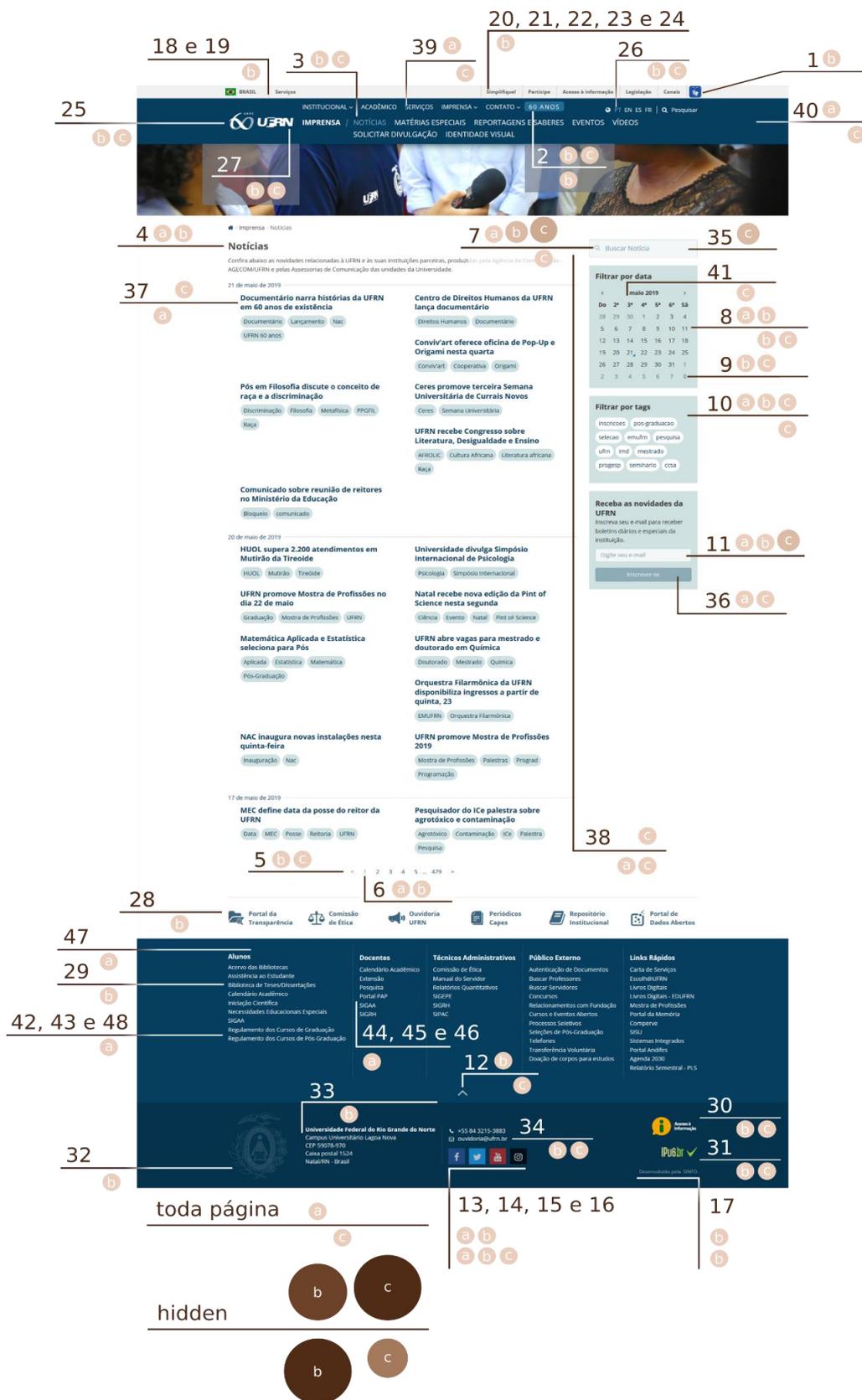
Figura 7 - Quantidade de avisos e erros na página acadêmico.



Fonte: Os autores.



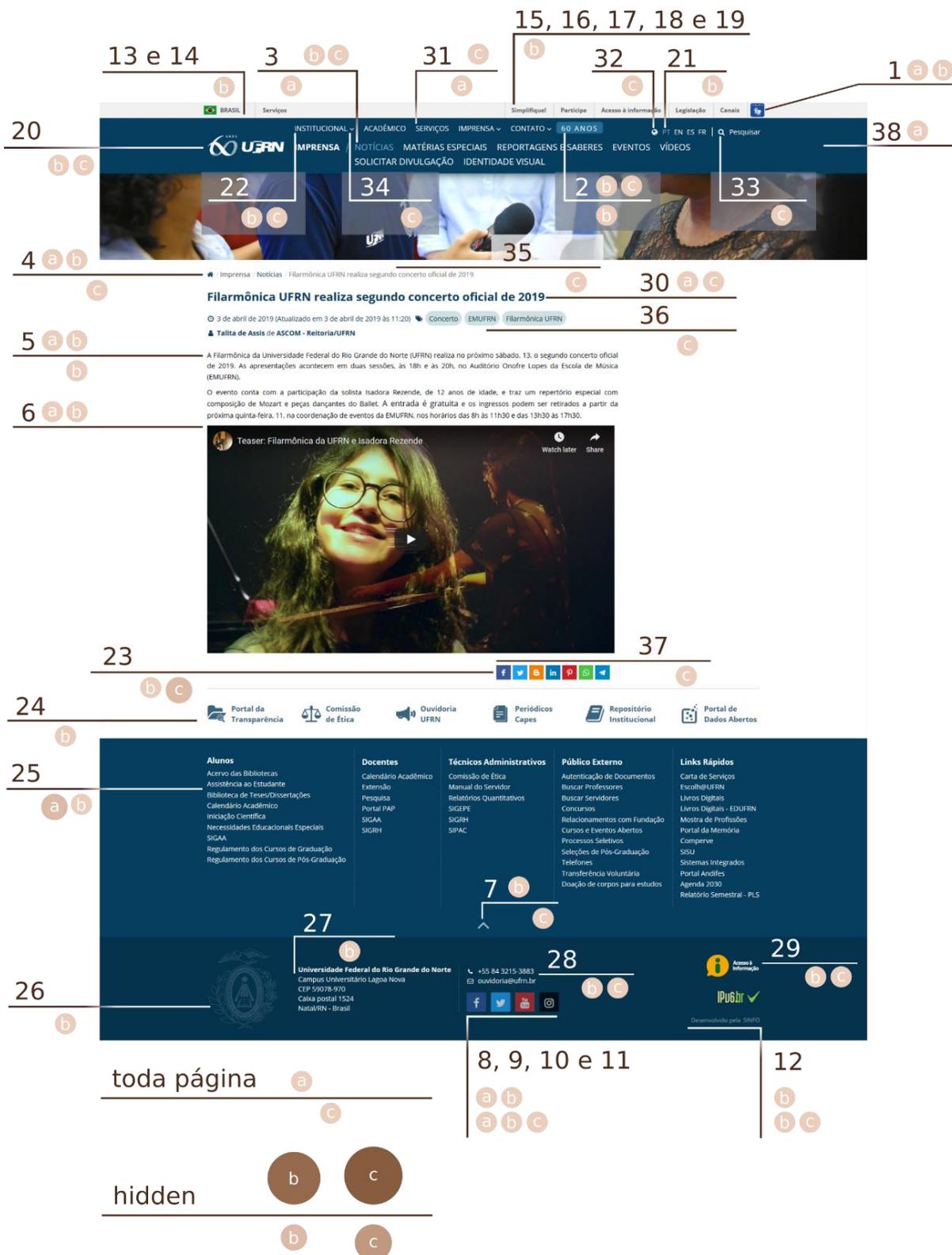
Figura 8 - Quantidade de avisos e erros na página notícias.



Fonte: Os autores.



Figura 9 - Quantidade de avisos e erros na página notícia.



Fonte: Os autores.

A página *home* (Figura 6) apresentou uma quantidade pequena de erros de acessibilidade em vários locais, desde o cabeçalho até o rodapé. Muitas partes da interface e diferentes elementos HTML foram afetados. Os avisos (potenciais problemas de acessibilidade) tiveram comportamento semelhante, apesar de serem em quantidade maior do que os erros. Não houve concentração de erros e avisos em local específico, exceto nos elementos HTML que caracterizam toda a página *web* e nos elementos ocultos (*hidden*) na página. É interessante observar que uma página que foi construída para ser utilizada com leitores de tela ainda precisa ser verificada e melhorada em tantos pontos dispersos para ser mais acessível.

A ocorrência e a distribuição de erros e avisos encontrados foram semelhantes nas outras páginas avaliadas: acadêmico (Figura 7), notícias (Figura 8) e notícia (Figura 9). Ainda que o cabeçalho e o rodapé dessas páginas sejam iguais aos da página *home*, os conteúdos principais (“parte do meio”) dessas páginas também apresentaram erros e avisos de acessibilidade. Isso é um indício de que prováveis equívocos ou oportunidades de melhoria ainda não exploradas no desenvolvimento configuram-se como desafios à promoção da acessibilidade no portal da UFRN. Os desenvolvedores e designers precisam prestar maior atenção aos detalhes das páginas relacionados com a acessibilidade.

A Tabela 3 indica os locais das páginas onde provavelmente existem barreiras de percepção, que precisam ser verificados e corrigidos pelos desenvolvedores e designers do portal da UFRN. Algumas imagens na página *home* provavelmente deveriam ter alguma descrição textual (critério 1.1.1), ainda que seja apenas para ilustrar (e.g. notícias) ou indicar a função de um elemento de interface (e.g. elemento de controle do carrossel). Seria interessante considerar as orientações da Nota Técnica nº 21 de 2012 do MEC/SECADI (BRASIL, 2012) para descrição dessas imagens.

Deve-se verificar se informações estão representadas adequadamente no código-fonte de todas páginas avaliadas para, assim, serem apresentadas de modo acessível aos usuários (critério 1.3.1). Também é importante que a sequência de leitura na *home* esteja explícita no código-fonte (critério 1.3.2). As orientações na interface sobre o seu uso devem estar explícitas no código-fonte da *home* (critério 1.3.3). Essas prováveis barreiras de percepção apontam para necessidade de verificar se o código-fonte reflete a intenção do projeto da interface.

A cor não pode ser a única forma de representar informações em elementos de interface ocultos (*hidden*) em todas as páginas (critério 1.4.1). O áudio da *home* deve poder ser controlado pelo usuário (critério 1.4.2). O contraste de alguns elementos de interface em todas páginas deve ser verificado (critérios 1.4.3 e 1.4.6). O tamanho do texto na página *home* poderia ser redimensionável pelos usuários (critério 1.4.4). As tecnologias assistivas podem fazer melhor seu trabalho quando o conteúdo está definido em um arquivo HTML separado do arquivo CSS que define suas características de apresentação (1.4.5). É preciso revisar esse ponto em algumas partes de todas as páginas avaliadas.

Tabela 3 - Indicação dos locais das prováveis barreiras de percepção.

barreiras de percepção	locais dos erros	locais dos avisos
1.1.1 Conteúdo não textual		
Todo conteúdo não textual deve possuir uma descrição textual alternativa para que as tecnologias assistivas apresentem-no adequadamente aos usuários.	<i>home</i> : 6, 7, 8, 9, 10, 11, 12, 64 e <i>hidden</i> acadêmico: <i>hidden</i> notícias: <i>hidden</i> notícia: <i>hidden</i>	<i>home</i> : 1, 2, 3, 4, 41, 42 e 43 notícias: 41 notícia: 23



1.3.1 Informações e relações

O conteúdo, a estrutura e o relacionamento das informações presentes no portal devem ser determinados no código-fonte das páginas para que as tecnologias assistivas possam apresentá-los adequadamente aos usuários.

home: 1, 2, 3, 4, 5, 10, 20, 33, 35, 36, 41, 46 e *hidden*

acadêmico: 3, 21, 22, 23, 24, 25, 40, 44 e *hidden*

notícias: 4, 7, 8, 10, 11, 35, 36, 39, 40 e *hidden*

notícia: 4, 5, 30, 38 e *hidden*

home: 1, 2, 3, 4, 5, 6, 7, 8, 9, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 44, 45, 51, 57, 65, 71 e *hidden*

acadêmico: 4, 5, 6, 7, 8, 18, 22, 23, 24, 25, 27, 34, 39, 40, 41, 42, 43 e *hidden*

notícias: 7, 8, 10, 12, 13, 14, 15, 16, 26, 34, 38, 39, 40, 41, 47 e *hidden*

notícia: 4, 7, 8, 9, 10, 11, 23, 25, 28, 32, 33, 34, 35, 36, 37 e *hidden*

1.3.2 Sequência com significado

A sequência de leitura deve estar explícita no código-fonte para que as tecnologias assistivas possam considerá-las.

home: 1, 2, 3, 4, 5, 13, 18, 19, 31, 33, 37, 38, 41, 44, 45, 57, 64 e 65

1.3.3 Características sensoriais

As explicações que auxiliam a compreensão e a utilização do portal devem ser determinadas no código-fonte para que as tecnologias assistivas apresentem-nas adequadamente aos usuários.

home: toda página

1.4.1 Utilização de cores

Cor não deve ser utilizada como único meio de representar informação. A informação deve estar explícita no código-fonte para que as tecnologias assistivas apresentem-nas adequadamente aos usuários.

home: *hidden*

acadêmico: *hidden*

notícias: *hidden*

notícia: *hidden*

home: toda página

1.4.2 Controle de áudio

Poder controlar a reprodução de áudio.

home: toda página

1.4.3 Contraste mínimo

O contraste da figura ou texto em relação ao fundo deve ser de no mínimo 4.5:1.

home: 44 e *hidden*

notícias: 5, 6, 9 e 36

home: 6, 7, 8, 9, 25, 26, 49, 66 e 69
notícia: 5

1.4.4 Redimensionar texto

Poder alterar tamanho do texto.

home: toda página

1.4.5 Imagens de Texto

Separar estrutura do documento em HTML das suas características de representação em CSS

home: 6, 7, 8, 9, 10, 11, 12, 41, 42, 43 e *hidden*

acadêmico: 9, 17, 35 e *hidden*

notícias: 25, 30, 31 e *hidden*

notícia: 12, 20, 29 e *hidden*



1.4.6 Contraste melhorado

O contraste da figura ou texto em relação ao fundo deve ser no mínimo de 7:1

home: 5, 24, 44, 45, 46, 47, 48 e 69

acadêmico: 2, 4 e 9

notícias: 2, 3, 5, 6, 9, 12, 17, 36, 37

notícia: 2, 3, 7 e 12

home: 1, 2, 3, 4, 5, 7, 10, 11, 12, 15, 25, 26, 32, 42, 49, 61, 62, 63, 65, 66, 67 e *hidden*

acadêmico: 10, 11, 12, 13, 14, 15, 16, 18, 19, 32, 33, 34 e 35

notícias: 18, 19, 20, 21, 22, 23, 24, 26, 27, 30, 32, 33 e 34

notícia: 13, 14, 15, 16, 17, 18, 19, 21, 22, 23, 26, 27, 28 e 29

Fonte: Os autores.

A Tabela 4 apresenta os locais das páginas onde provavelmente existem barreiras de operação, que precisam ser verificados e corrigidos pelos desenvolvedores e designers do portal da UFRN. É possível navegar pelo menu principal e pelo carrossel na *home* apenas com o teclado, mas ainda assim é preciso verificar se existe alguma outra funcionalidade não operável pelo teclado (critérios 2.1.1 e 2.1.3). O usuário pode pausar a passagem automática dos itens do carrossel enquanto posicionar o cursor do mouse sobre ele. Todavia, também é preciso verificar se outros comportamentos da interface determinados pela passagem do tempo são configuráveis pelo usuário em todas as páginas avaliadas (critério 2.2.1).

É preciso permitir que o usuário possa ignorar blocos durante a navegação por todo o site para tornar a leitura mais eficiente e menos entediante (critério 2.4.1). É muito provável que isso não seja possível em algumas partes das páginas *home* e notícia. O título das páginas indica que o usuário se encontra no portal da UFRN. Entretanto, vale refletir sobre quais são as vantagens e desvantagens de o título mudar quando se navega pelas páginas do portal para refletir uma parte mais específica dele (critério 2.4.2). Deve-se verificar se a ordem de navegação de alguns elementos das páginas *home* e notícia está coerente com as intenções de design (critério 2.4.3). As finalidades de vários *links* em todas as páginas precisam ser explicitadas no código-fonte do próprio *link* ou de elementos relacionados (critérios 2.4.4 e 2.4.9). É preciso verificar se a *home* oferece mecanismos flexíveis para navegação (critério 2.4.5). Os cabeçalhos e rótulos parecem informativos e descritos de forma adequada, mas vale verificar se correspondem com a intenção do projeto (critério 2.4.6). Alguns elementos na *home* deveriam estar destacados quando recebem o foco (critério 2.4.7).

Tabela 4 - Indicação dos locais das prováveis barreiras de operação.

barreiras de operação	locais dos erros	locais dos avisos
2.1.1 Teclado (com exceção) As funcionalidades devem ser operáveis pelo teclado		<i>home</i> : 1, 2, 3, 4, 5, 6, 7, 8, 9, 57, 58, 59 e toda página
2.2.1 Ajustável por Temporização Permitir controlar os limites de tempo do comportamento automático da interface		<i>home</i> : toda página acadêmico: toda página notícias: toda página notícia: toda página
2.1.3 Teclado (sem exceção) Todas as funcionalidades devem ser operáveis pelo teclado		<i>home</i> : 10, 11, 12, 41 e toda página



2.4.1 Ignorar Blocos

Existem mecanismos para ignorar a leitura de blocos de conteúdo repetidos

home: 54, 55, 56 e toda página
acadêmico: toda página
notícias: toda página
notícia: 6, 30 e toda página

home: toda página
acadêmico: toda página
notícias: toda página
notícia: toda página

2.4.2 Página com Título

Os títulos das páginas descrevem o tópico ou a finalidade.

home: toda página

2.4.3 Ordem do Foco

A ordem de navegação pelos elementos da página é coerente com seus significados e operações.

home: 1, 2, 3, 4, 5, 6, 7, 8, 9, 24, 45, 46, 49, 51 e 70
notícia: 23

2.4.4 Finalidade do link em contexto

A finalidade do link deve estar explícita em algum lugar da página para que as tecnologias assistivas possam apresentá-la adequadamente ao usuário.

home: 27, 28, 29, 30, 49, 50, 51, 52, 53, 64 e *hidden*
acadêmico: 5, 6, 7, 8, 36, 37 e 38
notícias: 13, 14, 15, 16, 38 e *hidden*
notícia: 1, 8, 9, 10, 11, 31 e *hidden*

home: 1, 8, 9, 10, 11 e 31
acadêmico: 36, 37, 38, 45 e 46
notícias: 13, 14, 15, 16, 38 e 48
notícia: 3, 8, 9, 10, 11, 25 e 31

2.4.5 Várias formas de navegar pelas páginas

Existem várias formas de navegar pelas páginas do portal.

home: toda página

2.4.6 Cabeçalhos e Rótulos

Os cabeçalhos e os rótulos descrevem o tópico ou a finalidade.

notícias: 35 e *hidden*

home: 1, 2, 3, 4, 5, 38, 51, 57, 58 e 59
acadêmico: 27, 28 e 29
notícias: 37, 42, 43, 44, 45 e 46
notícia: 25

2.4.7 Foco Visível

O foco do local de operação deve estar visível.

home: 6, 7, 8, 9, 24, 45, 46, 49, 51 e 70

2.4.9 Finalidade do link nele próprio

A finalidade do *link* deve estar explícita nele próprio para que as tecnologias assistivas possam apresentá-la adequadamente ao usuário.

home: 49, 50, 51, 52, 53 e *hidden*
acadêmico: 36, 37 e 38
notícias: 38
notícia: 31 e *hidden*

Fonte: Os autores.



Os locais das prováveis barreiras de compreensão no portal da UFRN são apresentados na Tabela 5. A confirmação da busca de notícias e sua respectiva alteração de contexto pode ser melhor comunicada para o usuário. Além disso, é preciso verificar se outras mudanças de contexto associadas a elementos de interface ocultos (*hidden*) foram comunicadas adequadamente em todas as páginas (critério 3.2.2). Como o portal avaliado tem por objetivo geral apresentar a UFRN, é aceitável que ele encaminhe o usuário para vários outros sistemas existentes que abordem aspectos específicos da instituição (critério 3.2.5). Essa questão muito provavelmente foi analisada com atenção no projeto desta versão do portal (GRILO et al., 2017). Alguns elementos da interface de notícias não comunicam adequadamente ao usuário as entradas de dados esperadas (critério 3.3.2). Os elementos de interface ocultos (*hidden*) identificados neste caso provavelmente não precisam desta comunicação ao usuário.

Tabela 5 - Indicação dos locais das prováveis barreiras de compreensão.

barreiras de compreensão	locais dos erros	locais dos avisos
3.2.2 Comportamento previsível na entrada de dados		
Comunicar ao usuário as alterações de contexto associadas a ações que ele pode realizar na interface.	<i>home: hidden</i> <i>acadêmico: hidden</i> <i>notícias: 7 e hidden</i> <i>notícia: hidden</i>	
3.2.5 Alteração de contexto mediante solicitação		
O contexto deve ser alterado mediante solicitação do usuário		<i>home: 26, 27, 28, 29, 30, 42, 43, 45, 49, 61, 62, 63, 69, 70, 71 e hidden</i> <i>acadêmico: 2, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 34, 35 e hidden</i> <i>notícias: 2, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 28, 29, 30, 31, 34 e hidden</i> <i>notícia: 2, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 24, 25, 29 e hidden</i>
3.3.2 Rótulos ou Instruções		
Rótulos e instruções devem comunicar ao usuário a entrada de dados esperada em cada campo da interface.	<i>home: hidden</i> <i>acadêmico: hidden</i> <i>notícias: 7, 11 e hidden</i> <i>notícia: hidden</i>	

Fonte: Os autores.

Os locais das barreiras de robustez entrados no portal estão indicados na Tabela 6. A programação de alguns elementos da página de notícias não seguiu todas as especificações dos padrões *web* (critério 4.1.1). Os elementos de interface criados ou modificados nas páginas do



portal devem ser definidos adequadamente para poderem ser apresentados pelas tecnologias assistivas aos usuários (critério 4.1.2).

Tabela 6 - Indicação dos locais das prováveis barreiras de robustez.

barreiras de robustez	locais dos erros	locais dos avisos
4.1.1 Análise do código-fonte das páginas web		
O código-fonte das páginas deve estar construído conforme a especificação do HTML e CSS	notícias: 7, 10 e <i>hidden</i>	
4.1.2 Nome, função, valor dos componentes de interface		
Todos os componentes de interface criados pelo programador ou aqueles componentes padrão <i>web</i> cujos comportamentos foram modificados pelo programador devem ter nome, função e valor definidos adequadamente para que as tecnologias assistivas possam apresentá-los aos usuários.	<i>home</i> : 18, 19, 27, 28, 29, 30, 54, 55, 56, 64 e <i>hidden</i> <i>acadêmico</i> : 1, 3, 5, 6, 7, 8 e <i>hidden</i> notícias: 1, 4, 7, 11, 13, 14, 15, 16, 35 e <i>hidden</i> notícia: 1, 4, 7, 11, 13, 14, 15, 16, 35 e <i>hidden</i>	<i>home</i> : 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 41, 57, 58, 59, 60 e <i>hidden</i> <i>acadêmico</i> : 17, 19, 44 e <i>hidden</i> notícias: 25, 27 e <i>hidden</i> notícia: 20, 22, 23 e 37

Fonte: Os autores.

5 Considerações Finais

A acessibilidade de sistemas computacionais interativos tem assumido papel importante para a inclusão de pessoas com deficiência no ensino superior. A nova versão do Portal da UFRN foi um dos sistemas da instituição concebido considerando seu uso com leitores de tela e VLibras (GRILO et al., 2017). Contudo, sua acessibilidade ainda não havia sido avaliada.

Este trabalho relatou uma avaliação de acessibilidade do Portal da UFRN por inspeção com três avaliadores automáticos baseados na WCAG 2.0 (W3C, 2008): *AInspector*, *HTML Code Sniffer* e *Siteimprove Accessibility Checker*. Foram encontrados 325 erros e 555 avisos totais em quatro páginas do portal: *home*, *acadêmico*, *notícias* e de uma notícia. Ainda que existam repetições entre avaliadores, esses erros e avisos encontrados evidenciam locais na interface com usuário que provavelmente apresentam barreiras de acessibilidade na percepção, operação, compreensão e robustez do sistema. São pontos que deveriam ser considerados pelos designers e desenvolvedores para a melhoria da acessibilidade do Portal da UFRN.

O cuidado com a acessibilidade durante o design do Portal foi importante, mas não foi suficiente para evitar problemas de acessibilidade. Esses problemas podem ter surgido por equívocos, falta de atenção ou mesmo por desconhecimento sobre acessibilidade durante a programação do sistema. Esses resultados destacam a importância dos detalhes de programação da interface para promover a acessibilidade em sistemas computacionais interativos.

A avaliação de acessibilidade demonstrou ter papel complementar fundamental aos cuidados com a acessibilidade durante o projeto e a programação deste sistema web. Isso é coerente com o recomendado em geral para cuidar da qualidade de uso (BARBOSA; SILVA, 2010). As avaliações da interface com usuário deveriam conjugar vários métodos, inclusive de tipos diferentes, para obter uma compreensão mais ampla das experiências de uso e eventuais necessidades de melhoria (SANTOS; SILVA, 2018). Assim, uma avaliação por inspeção com avaliadores automáticos como a relatada neste trabalho pode ser uma das avaliações de acessibilidade realizadas durante o desenvolvimento de sistemas computacionais para as instituições de ensino superior. Ela tem a vantagem de ser de execução rápida, barata, detalhada

