

Plugin de acessibilidade para sites institucionais da UFJF

João Victor Pereira dos Anjos

Jan 2025

Sumário

1	Resumo	1
2	Resumo	1
3	Introdução	2
3.1	Metodologia	3
3.1.1	Análise de Requisitos	4
3.1.2	Desenvolvimento do Plugin	6
3.1.3	Avaliação da Ferramenta	8
4	Conclusão	9
5	Agradecimentos	10

1 Resumo

2 Resumo

This is the first section.

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Etiam lobortis facilisis sem. Nullam nec mi et neque pharetra sollicitudin. Praesent imperdiet mi nec ante. Donec ullamcorper, felis non sodales..

3 Introdução

A palavra acessibilidade, sua origem etimológica é derivada do latim *accessibilitas* e significa “condição para utilização, com segurança e autonomia, total ou assistida, dos espaços, mobiliários e equipamentos urbanos, das edificações, dos serviços de transporte e dos dispositivos, sistemas e meios de comunicação e informação, por pessoa com deficiência ou mobilidade reduzida” (CÂMARA DOS DEPUTADOS, 2025).

No Brasil, a acessibilidade é um direito garantido pela Constituição Federal de 1988, pela Lei Brasileira de Inclusão (LBI) de 2015 (BRASIL, 2015) e por normas técnicas específicas, como a NBR 9050/2015 da Associação Brasileira de Normas Técnicas (ABNT, 2015). Essas legislações estabelecem parâmetros para a promoção da acessibilidade em espaços públicos e privados, visando a inclusão de pessoas com deficiência física, visual, auditiva, intelectual e múltipla.

No âmbito digital a acessibilidade web é um pilar fundamental para a inclusão, o WCAG 2.1/2.2 (W3C, 2024) sendo um conjunto de diretrizes internacionais para a acessibilidade de conteúdo web, tem por objetivo, tornar os sites mais acessíveis para as pessoas com deficiência visual, auditiva, motora e cognitiva, garantindo a igualdade de acesso à informação e aos serviços online. O eMag é um modelo nacional de acessibilidade em governo eletrônico que estabelece diretrizes para a promoção da acessibilidade em sites governamentais, com o objetivo de garantir a inclusão digital e o acesso à informação para todos os cidadãos Brasileiros.

A Universidade Federal de Juiz de Fora (UFJF), como uma instituição pública, gerência uma grande quantidade de sites e portais, que são regularmente atualizados por diversas pessoas, como professores, pesquisadores, bolsistas e servidores, da qual chamamos de conteudistas. A diversidade de conteúdos e responsáveis torna o processo de garantia de acessibilidade primordial para atender a legislação e promover a inclusão digital.

A fim de divulgar as informações referentes aos seus setores e atividades a Universidade Federal de Juiz de Fora, UFJF, através do Centro de Gestão do Conhecimento Organizacional (CGCO), é responsável por controlar a disponibilização de sites, a padronização dos layouts e o suporte técnico. Para a sustentação desse serviço, é utilizado o CMS WordPress (FOUNDATION, W., 2003), uma plataforma de gerenciamento de conteúdo que permite a criação e manutenção de sites de forma simplificada e intuitiva.

Neste cenário o temos uma complexidade ao realizar auditorias manuais,

que consomem tempo e recursos, e principalmente a falta de ferramentas centralizadas dentro da UFJF para aplicar o Modelo de Acessibilidade em Governo Eletrônico (GOVERNO FEDERAL BRASILEIRO, 2004), conhecido como eMag, e as Diretrizes de Acessibilidade para Conteúdo Web (W3C, 2024), conhecidas como WCAG, visto que esse documentos estabelecem parâmetros técnicos para essa inclusão.

Em uma análise preliminar avaliando a presença de elementos de acessibilidade nos sites da UFJF, foi identificado que boa parte apresentavam falhas de acessibilidade, como imagens sem texto alternativo e baixo contraste de cores, limitando o acesso de usuários com deficiência visual. Além disso, a falta de padronização e de um processo de auditoria contínuo dificulta a identificação e correção dessas falhas, comprometendo a qualidade e a usabilidade dos sites.

Diante deste desafio, este artigo propõe uma solução técnica inovadora para o contexto da UFJF, baseada em um plugin WordPress de auditoria de acessibilidade que integra tecnologias modernas de automação, análise técnica e processamento de dados. O sistema opera como um serviço independente, com suporte a regras WCAG 2.1/2.2 e eMAG, permitindo avaliações em tempo real e personalização de regras de acessibilidade.

3.1 Metodologia

Diante do contexto apresentado a seguinte metodologia proposta neste trabalho para a implementação do sistema de auditoria de acessibilidade, é dividida em três etapas principais: análise de requisitos, onde são identificadas as necessidades da UFJF e feitos testes com ferramentas de acessibilidade já existentes; desenvolvimento do plugin, onde são definidas as tecnologias à serem utilizadas, a arquitetura do sistema, a implementação das funcionalidades e a integração com o WordPress Multi-Sites (FOUNDATION, W., 2025b); e por fim, a avaliação da ferramenta, onde são feitos testes de usabilidade, acessibilidade, desempenho e escalabilidade do sistema.

Fragmentando a metodologia em etapas, é possível garantir um desenvolvimento mais organizado e eficiente, com foco na qualidade e na usabilidade do sistema. Além disso, a divisão em etapas permite a identificação de possíveis problemas e a realização de ajustes ao longo do processo, garantindo a entrega de um produto final que atenda às expectativas e necessidades da UFJF.

3.1.1 Análise de Requisitos

A primeira etapa consiste na análise dos requisitos do sistema, com base nas necessidades da UFJF e nas diretrizes de acessibilidade. Para isso, a partir de reuniões com a equipe de TI e conteudistas, chegou-se a um conjunto de funcionalidades essenciais para o plugin, como: integração com o WordPress Multi-Sites, avaliação em tempo real, suporte as diretrizes de acessibilidade, relatórios claros e simples, personalização de regras a serem avaliadas e a possibilidade de visualização de erros e sugestões de correção.

Como já citado, ter o suporte as regras WCAG 2.1/2.2 e eMAG é fundamental para garantir a conformidade dos sites da UFJF com as normas de acessibilidade. Além disso, a utilização de um sistema como o CMS WordPress dentro da UFJF é uma realidade, e portanto, a integração do plugin de acessibilidade com essa plataforma não só é desejável, como também é essencial para garantir que os relatórios possam ser facilmente compreendidos, visto que os conteudistas da UFJF responsáveis pela atualização dos sites, muitas vezes não possuem conhecimento técnico em relação à elementos HTML, CSS e JavaScript, o que torna-se relevante a disponibilização dos relatórios de forma clara e simples, e facilmente seguidos para a correção dos erros.

É importante notar que o WordPress permite a padronização de layouts, e certos elementos das páginas são gerenciados pelos temas desenvolvidos pelos desenvolvedores da UFJF, portanto, tais elementos não são passíveis de alteração pelos conteudistas, o que torna a avaliação de acessibilidade desses elementos não necessária para o conteudista, sendo assim, o plugin deve permitir a personalização tanto das regras a serem avaliadas, quanto dos elementos.

Uma API REST (RED HAT, 2023), é uma interface de programação de aplicações, que permite a comunicação entre sistemas, e é amplamente utilizada para integração de sistemas e serviços. A utilização de uma API REST somente para a geração dos relatórios em tempo real, sem a necessidade de armazenamento, é uma solução mais eficiente e escalável, visto que a UFJF possui uma quantidade bastante significativa de sites (Inserir Valor), e armazenar todos os relatórios de acessibilidade em um banco de dados, é tarefa que consumiria muitos recursos e espaço em disco, tornando inviável tal abordagem.

Em fase preliminar ao desenvolvimento do projeto, realizamos testes com o AccessMonitor (MODERNIZAÇÃO ADMINISTRATIVA, 2021), uma fer-

ramenta de auditoria de acessibilidade online, que permite a avaliação de sites em tempo real. Embora a solução ofereça uma interface intuitiva para avaliação pontual de páginas web, identificamos limitações significativas para o contexto operacional da UFJF. As principais restrições estavam na ausência de mecanismos de personalização de regras, impedindo a adaptação a contextos específicos, e a escalabilidade restrita, com um limite operacional páginas a serem avaliadas por um endereço IP. Adicionalmente, a necessidade de inserção manual de URLs por avaliação tornava inviável a análise de grandes portfólios de sites.

Testamos também a utilização do QualWeb (QUALWEB, 2020), que é a ferramenta de avaliação por trás do AccessMonitor, porém, esbarramos em limitações técnicas, semelhantes às encontradas no AccessMonitor, e principalmente, na falta de suporte na tradução de relatórios para o português, o que dificultaria a compreensão dos conteudistas da UFJF.

Diante dessas limitações, optamos por desenvolver um plugin de acessibilidade personalizado, que atendesse às necessidades específicas da UFJF e que fosse compatível com a infraestrutura tecnológica existente. Para auxiliar no desenvolvimento da API REST que gera os relatórios de acessibilidade, buscamos por ferramentas e tecnologias que avaliasse a acessibilidade do conteúdo DOM (MDN WEB DOCS, 2025) de uma página web.

Encontramos o IBM Equal Access Toolkit (IBMA, 2020), uma ferramenta de código aberto desenvolvida pela IBM, que permite a avaliação de acessibilidade de páginas web em tempo real, com suporte às diretrizes de acessibilidade. A ferramenta oferece o accessibility-checker (IBM, 2020) um pacote Node (FOUNDATION, N., 2009) que permite a testes automatizados de acessibilidade. o pacote permite também a personalização de regras a serem avaliadas, porém, a dificuldade de integração com a nossa API, e a falta de suporte para o português, impediu de seguirmos com essa ferramenta.

A ferramenta da qual decidimos utilizar foi o axe-core (SYSTEMS, 2015), a quarta ferramenta de avaliação de acessibilidade que testamos. O axe-core é uma biblioteca de código aberto, desenvolvida pela Deque Systems, que permite a avaliação de acessibilidade de páginas web. O ponto chave que nos fez optar por essa ferramenta, foi a facilidade de integração com a nossa API REST, a possibilidade de personalização de regras a serem avaliadas, e o suporte para o português, o que facilitaria a compreensão dos conteudistas da UFJF.

3.1.2 Desenvolvimento do Plugin

A segunda etapa consiste no desenvolvimento do plugin de acessibilidade, que integra tecnologias modernas de automação, análise técnica e processamento de dados. Para isso, optamos por utilizar a linguagem de programação JavaScript, em conjunto com o framework Express.js (FOUNDATION, O., 2010), que é um framework web para Node.js, que permite a criação de aplicações web de forma rápida e eficiente.

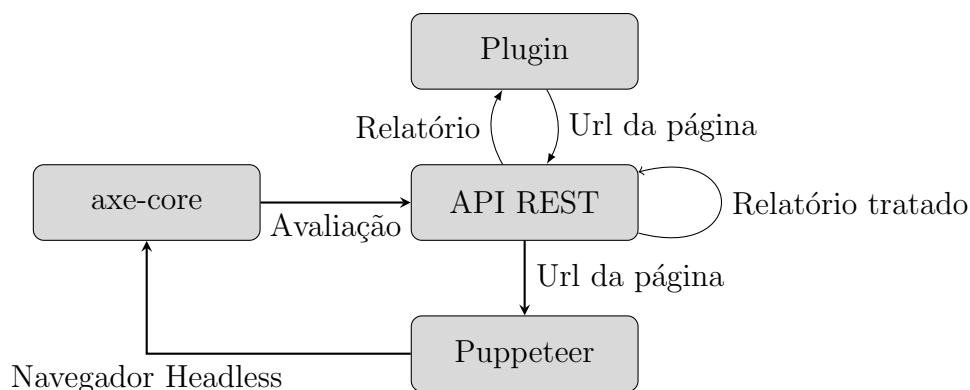
Para entendermos a integração com o WordPress Multi-Sites, precisamos entender o conceito de Endpoints, Rotas e o que significa Multi-Sites no contexto do WordPress. Um endpoint (IBM, 2025) é uma URL específica que é usada para acessar recursos em uma API, como por exemplo, uma lista de posts ou páginas de um site. Uma rota (WIKIPEDIA, 2025e) é um caminho específico que é usado para acessar um endpoint, e é definida no arquivo de rotas da aplicação. O WordPress Multi-Sites é uma funcionalidade do WordPress que permite a criação de vários sites a partir de uma única instalação, e cada site possui um ID único e um domínio específico, além de possuir seu próprio conjunto de posts e páginas.

Partindo desse entendimento, para o desenvolvimento do sistema, o dividimos em duas partes, o plugin no gerenciador de conteúdo WordPress, e a API REST hospedada em um servidor externo. O plugin é responsável por consultar a API REST, e exibir os relatórios de acessibilidade na interface do WordPress, enquanto a API REST é responsável por receber as requisições do plugin, avaliar a acessibilidade das páginas web e retornar os relatórios para o plugin.

Das rotas disponíveis na API, as mais relevantes são aquelas dedicadas à avaliação de acessibilidade: uma destinada a analisar uma página ou post específico e outra voltada para a verificação integral de todos os posts de um site. Ambas compartilham um elemento central em sua estrutura: o ID do site, que atua como parâmetro obrigatório e chave para identificar o contexto da análise. Além disso, seguem um núcleo operacional comum, no qual são acionados os processos de geração de relatórios detalhados de acessibilidade, garantindo consistência e padronização nos resultados.

A distinção entre as duas rotas reside em um detalhe fundamental: a rota de avaliação de um post específico possui um escopo mais restrito, focando apenas naquele post em particular, e principalmente, é por essa rota que o plugin é acionado, permitindo que o conteudista realize a avaliação de acessibilidade de um post específico, e visualize o relatório na interface do

WordPress. Por outro lado, a rota de avaliação de todos os posts de um site opera de forma mais abrangente, percorrendo todas as páginas disponíveis e gerando relatórios. Essa diferença de escopo é essencial para atender às demandas específicas dos conteudistas e dos desenvolvedores da UFJF.



Fluxo de execução da api

Seguindo como base o fluxo de execução da API apresentado na figura acima, a requisição do plugin é recebida pela API REST, que se utiliza do Puppeteer (LLC, 2017), uma biblioteca Node.js que fornece uma API de alto nível para controlar o Chrome (WIKIPEDIA, 2025a) através do protocolo DevTools (LLC, 2008). O Puppeteer é utilizado para abrir uma nova aba no navegador de forma headless (LLC, 2024), ou seja, sem interface gráfica, e carregar a URL da página especificada na requisição. Após o carregamento da página, a API executa a ferramenta axe-core, usando como parametro a página carregada pelo Puppeteer e a personalização definida pelo CGCO, e gera uma avaliação de acessibilidade. O resultado da avaliação é retornado para a API, que processa o relatório removendo informações irrelevantes e organizando os dados para a integração com o plugin. O relatório final então é enviado de volta para o plugin, que exibe as informações na interface do WordPress.

As principais personalizações implementadas no API serão: tradução dos relatórios para o português; Quais regras de acessibilidade não serão avaliadas como: "blink", que avalia se o texto pisca, "html-has-lang", que avalia se o HTML possui o atributo "lang", "meta-refresh", que avalia se a página possui um meta refresh, e "accesskeys", que avalia se a página possui o atributo "accesskey"; Quais diretrizes serão avaliadas, como: WCAG 2.1/2.2, ACT e Best Practices; e qual Tag HTML será avaliada, que para o contexto da

UFJF, é o elemento com ID "conteudo-main", que é o elemento que contém o conteúdo de responsabilidade do conteudista.

Entendendo o funcionamento da API, e sua importancia para o plugin, podemos seguir para a implementação do plugin em si. O plugin é desenvolvido em PHP (GROUP, 1995), uma linguagem de programação amplamente utilizada para o desenvolvimento web, e é a linguagem utilizada pelo WordPress.

O WordPress disponibiliza a possibilidade de podermos adicionar elementos novos na interface do seu gerenciador de conteúdo, através de Hooks (FOUNDATION, W., 2025a), que são pontos de extensão que permitem a modificação do comportamento padrão do WordPress. Em seu gerenciador de conteúdo, o WordPress possui uma área de administração das páginas, onde os conteudistas podem visualizar todos os posts e páginas do site, e editar o conteúdo de cada um deles.

Tendo isso em mente, o plugin é desenvolvido para adicionar um novo botão na interface de administração das páginas, que ao ser clicado, aciona a API REST, passando como parâmetro a URL da página que será avaliada. O plugin então aguarda a resposta da API, que retorna o relatório de acessibilidade. O plugin então exibe o relatório na interface do WordPress, organizando os dados por categoria e criticalidade. Em sua interface disponibilizamos informações como o número de erros e avisos encontrados, o título da regra de acessibilidade, a descrição do erro, uma sugestão de correção, e um link que abre a página em uma nova aba, com o elemento que apresenta o erro de forma destacada.

Dessa forma, o plugin permite que os conteudistas da UFJF possam avaliar a acessibilidade de suas páginas de forma simples e rápida, sem a necessidade de conhecimentos técnicos em relação à elementos HTML, CSS e JavaScript. Além disso, o plugin permite que os conteudistas possam visualizar o que deverá ser corrigido, e como fazer isso.

3.1.3 Avaliação da Ferramenta

A terceira etapa consiste na avaliação da ferramenta, que será realizada através de testes de usabilidade, acessibilidade, desempenho e escalabilidade do sistema. Para isso, serão realizados testes com outras ferramentas de acessibilidade, como o AccessMonitor, para avaliarmos a confiabilidade dos relatórios gerados pelo plugin. Durante todo o processo de desenvolvimento, foram realizados testes de desempenho com o Insomnia (INSOMNIA, 2016),

uma ferramenta de teste de APIs, para avaliarmos o tempo de resposta da API, e obtivemos um tempo médio de resposta de 47ms, o que é um tempo considerado bom para uma API REST. Esse tempo de resposta foi obtido em um ambiente de desenvolvimento, e portanto, não reflete o tempo de resposta em um ambiente de produção, onde o tempo de resposta pode variar dependendo da carga do servidor e da quantidade de requisições simultâneas.

Além disso, serão realizados testes de usabilidade com os conteudistas da UFJF, para avaliarmos a facilidade de uso do plugin e a clareza dos relatórios gerados, esses testes serão cruciais para entendermos quais ajustes deverão ser feitos na interface do plugin, e como melhorar a experiência do usuário. Por fim, se utilizando de demandas internas da UFJF, foram realizados testes de escalabilidade, para avaliarmos a possibilidade de utilizarmos a API em outros sites da UFJF, que não são gerenciados pelo WordPress, e a possibilidade de utilizarmos a API em outros sistemas, como a nova versão do SIGAx (UFJF, 2024), que é o sistema de gestão acadêmica da UFJF. Com o testes de escalabilidade, obtivemos resultados positivos, o que nos permitiu criar um novo endpoint na API, que permite a avaliação de acessibilidade de qualquer página web, requisitando apenas o DOM e CSS da página, o que foi recebido com entusiasmo pela equipe de TI da UFJF. Esse novo endpoint permite que a API possa ser utilizada em outros sistemas ainda em ambientes de desenvolvimento, e que não estão disponíveis para o público.

4 Conclusão

A proposta deste artigo é apresentar uma solução técnica inovadora para o contexto da UFJF, baseada em um plugin WordPress de auditoria de acessibilidade que integra tecnologias modernas de automação, análise técnica e processamento de dados. O sistema opera como um serviço independente, com suporte a regras WCAG 2.1/2.2 e eMAG, permitindo avaliações em tempo real e personalização de regras de acessibilidade.

A ferramenta não apenas otimiza processos técnicos, mas democratiza a fiscalização de acessibilidade, empoderando conteudistas não especialistas com dados claros e de fácil identificação. Este trabalho visa, portanto, contribuir para o debate sobre automação e inclusão digital, sugerindo um modelo que buscam alinhar-se às exigências legais e éticas da acessibilidade web, dentro do escopo do eMag, WCAG 2.1/2.2 e o contexto de cada organização.

5 Agradecimentos

Agradeço ao Centro de Gestão do Conhecimento Organizacional (CGCO) da Universidade Federal de Juiz de Fora (UFJF) pela oportunidade de desenvolver este trabalho e pela confiança depositada em mim. Agradeço também ao meu orientador, Gildo de Almeida Leonel, pela orientação e apoio durante todo o processo de desenvolvimento do plugin de acessibilidade. Agradeço também a toda a equipe do CGCO, que sempre esteve disposta a ajudar e a contribuir com o desenvolvimento do plugin.

Referências

- ABNT. NBR 9050: Acessibilidade a edificações, mobiliário, espaços e equipamentos urbanos. In: ASSOCIAÇÃO Brasileira de Normas Técnicas. [S.l.: s.n.], 2015. Disponível em: https://acessibilidade.unb.br/images/PDF/NORMA_NBR-9050.pdf.
- BRASIL. Lei nº 13.146, de 6 de julho de 2015, Lei Brasileira de Inclusão da Pessoa com Deficiência. **Diário Oficial da República Federativa do Brasil**, 2015. Disponível em: http://www.planalto.gov.br/ccivil_03/_ato2015-2018/2015/lei/113146.htm. Acesso em: 20 fev. 2025.
- CÂMARA DOS DEPUTADOS. **Glossário de Acessibilidade**. Disponível em: <https://www2.camara.leg.br/a-camara/estruturaadm/gestao-na-camara-dos-deputados/responsabilidade-social-e-ambiental/acessibilidade/glossarios/glossario.html>. Acesso em: 13 fev. 2025.
- CARVALHO, Vinícios; CAGNIN, Maria; PAIVA, Débora. Avaliação de Acessibilidade de Web Sites de Governos Estaduais do Brasil. In: ANAIS do XIII Simpósio Brasileiro de Sistemas de Informação. Lavras: SBC, 2017. P. 116–123. DOI: 10.5753/sbsi.2017.6033. Disponível em: <https://sol.sbc.org.br/index.php/sbsi/article/view/6033>.
- FOUNDATION, Node.js. **Node.js**. [S.l.: s.n.], 2009. Disponível em: <https://nodejs.org/>.
- FOUNDATION, OpenJS. **Express**. [S.l.: s.n.], 2010. Disponível em: <https://expressjs.com/>.

FOUNDATION, WordPress. **Hooks**. Disponível em:
<https://developer.wordpress.org/plugins/hooks/>. Acesso em: 25
mar. 2025.

_____. **WordPress**. [S.l.: s.n.], 2003. Disponível em:
<https://wordpress.org/>.

_____. **WordPress Multisite / Network**. Disponível em: <https://developer.wordpress.org/advanced-administration/multisite/>.
Acesso em: 20 fev. 2025.

GOVERNO FEDERAL BRASILEIRO. **eMAG - Modelo de
Acessibilidade em Governo Eletrônico**. [S.l.], 2004. Disponível em:
<https://emag.governoeletronico.gov.br/>.

GROUP, The PHP. **PHP**. [S.l.: s.n.], 1995. Disponível em:
<https://www.php.net/>.

IBM. **Accessibility-Checker**. [S.l.: s.n.], 2020. Disponível em:
<https://www.npmjs.com/package/accessibility-checker>.

_____. **O que é um endpoint de API?** Disponível em:
<https://www.ibm.com/br-pt/topics/api-endpoint>. Acesso em: 20 fev.
2025.

IBMA. **IBM Equal Access Accessibility Checker**. [S.l.: s.n.], 2020.
Disponível em: <https://github.com/IBMa/equal-access>.

IETF. **JSON Web Token (JWT)**. [S.l.], 2015. Disponível em:
<https://tools.ietf.org/html/rfc7519>.

INSOMNIA. **Insomnia**. [S.l.: s.n.], 2016. Disponível em:
<https://insomnia.rest/>.

LLC, Google. **DevTools**. [S.l.: s.n.], 2008. Disponível em:
<https://developer.chrome.com/docs/devtools/>.

_____. **Modo headless do Chrome**. 2024. Disponível em:
<https://developer.chrome.com/docs/chromium/headless?hl=pt-br>.
Acesso em: 25 mar. 2025.

_____. **Puppeteer**. [S.l.: s.n.], 2017. Disponível em:
<https://pptr.dev/>.

MDN WEB DOCS. **O que é o DOM?** Disponível em:
|https://developer.mozilla.org/pt-BR/docs/Web/API/Document_Object_Model/Introduction#o_que_%C3%A9_o_dom|. Acesso em: 25 mar. 2025.

MODERNIZAÇÃO ADMINISTRATIVA, AMA - Agência para a.
accessmonitor. [S.l.: s.n.], 2021. Disponível em:
|<https://accessmonitor.acessibilidade.gov.pt/>|.

QUALWEB. **QualWeb**. [S.l.: s.n.], 2020. Disponível em:
|<https://qualweb.di.fc.ul.pt/evaluator/about>|.

RED HAT. **O que é uma API REST?** [S.l.], 2023. Disponível em:
|<https://www.redhat.com/pt-br/topics/api/what-is-a-rest-api>|.

SYSTEMS, Deque. **axe-core**. [S.l.: s.n.], 2015. Disponível em:
|<https://github.com/dequelabs/axe-core>|.

UFJF. **UFJF lança nova versão do Siga X com mais recursos**. 2024. Disponível em: |<https://www2.ufjf.br/noticias/2024/12/10/ufjf-lanca-nova-versao-do-siga-x-com-mais-recursos/>|. Acesso em: 25 mar. 2025.

W3C. **Web Content Accessibility Guidelines (WCAG) 2.2**. [S.l.], 2024. Disponível em: |<https://www.w3.org/TR/WCAG22/>|.

WIKIPEDIA. **Google Chrome**. Disponível em:
|https://pt.wikipedia.org/wiki/Google_Chrome|. Acesso em: 25 mar. 2025.

_____. **Hypertext Transfer Protocol**. Disponível em:
|https://pt.wikipedia.org/wiki/Hypertext_Transfer_Protocol|. Acesso em: 25 mar. 2025.

_____. **JSON Web Token**. Disponível em:
|https://pt.wikipedia.org/wiki/JSON_Web-Token|. Acesso em: 25 mar. 2025.

_____. **Middleware**. Disponível em:
|<https://pt.wikipedia.org/wiki/Middleware>|. Acesso em: 25 mar. 2025.

_____. **Routing**. 2025. Disponível em:
|<https://en.wikipedia.org/wiki/Routing>|. Acesso em: 20 fev. 2025.