

Aplicação de ferramentas automatizadas para geração de formulários Web acessíveis

Fernanda B. Faria¹, Thiago J. Bittar^{1,2}, Luanna L. Lobato¹

¹Departamento de Ciências da Computação – Universidade Federal de Goiás (UFG)
CEP: 75704-020 – Catalão – GO – Brasil

² Instituto de Ciências Matemáticas e de Computação – Universidade de São Paulo (USP)
Caixa Postal 668 – 13560-970 – São Carlos – SP – Brasil

ferrbontempo@gmail.com, jabur@icmc.usp.br, lll@catalao.ufg.br

Abstract. *This paper shows how the use of automated tools may help in the generation of accessible forms. Therefore, we used two tools that assist the developer in this task, a form to generate itself, called Quick Form Builder and another to validate the code generated by this tool, the W3C code validator. Thus it is apparent that the tools help developers to develop affordable forms, since it does not need to meet accessibility guidelines, simply use the tools that are intuitive and easy to access.*

Resumo. *Neste trabalho é apresentado como a utilização de ferramentas automatizadas podem ajudar na geração de formulários acessíveis. Para tanto, foram utilizadas duas ferramentas que auxiliam ao desenvolvedor nesta tarefa, uma para gerar formulário em si, chamada Quick Form Builder e outra para validar o código html de formulários Web, o validador de código do W3C. Com isso, é perceptível que as ferramentas auxiliam o desenvolvedor a desenvolver formulários acessíveis, visto que o mesmo não precisa conhecer as diretrizes de acessibilidade, é necessário apenas usar as ferramentas que são intuitivas e de fácil acesso.*

1. Introdução

Para que haja um desenvolvimento ágil e com mais qualidade é necessário que se tenha ferramentas que agregam mais facilidades e recursos aos desenvolvedores e que evoluam de acordo com suas necessidades. Uma preocupação maior que se tem, além do desenvolvimento rápido, é que sistemas Web possuam acessibilidade, para que a inclusão digital seja de fato efetivada. Assim, é necessário produzir artefatos Web sem barreiras, ou seja, eliminar as dificuldades de acesso às pessoas com deficiência e para isso, podem ser utilizadas ferramentas para auxiliar o desenvolvedor e também para que este tenha conhecimento das diretrizes de acessibilidade e como aplicá-las [Sloan et al. 2006], [Maia 2010].

Acessibilidade Web está relacionada com a possibilidade de que diferentes perfis de usuários e, principalmente, pessoas com deficiência sejam capazes de usá-la [Lima 2007]. Mais precisamente, uma Web com conteúdo interativo, fácil de entender, compreender e manipular por pessoas com os mais diversos tipos de deficiência [W3C 2008].

Para interagir em sistemas e sites Web, na maioria das vezes é necessário o preenchimento de formulários, seja para cadastro ou simplesmente para receber uma informação de retorno. Entretanto, como visto em um estudo de caso recente [Bittar et al. 2012a], os formulários, sobretudo das universidades públicas do Brasil, não atendem às diretrizes de acessibilidade, o que dificulta o acesso pelos usuários, principalmente aqueles que possuem algum tipo de deficiência.

Os problemas identificados nos formulários Web pelos autores supracitados, ressaltam a importância de ferramentas automatizadas para auxiliar o desenvolvedor no desenvolvimento de tais recursos. Assim, é visto no presente trabalho como ferramentas automatizadas auxiliam na geração de formulários acessíveis. Para tanto, é mostrado o uso de um gerador de formulários acessíveis, *Quick Form Builder*¹, desenvolvido pela empresa *Accessify*², que foi escolhido devido a fácil utilização e popularização da ferramenta, juntamente com um validador de código, do *World Wide Web Consortium (W3C)*³, para que se comprove a eficácia do gerador de formulários.

O trabalho é apresentado da seguinte forma: Na Seção 2 é mostrada a metodologia utilizada para a realização da pesquisa. Na Seção 3 tem-se os trabalhos relacionados. Na Seção 4 é visto o funcionamento do gerador de formulários acessíveis, em que seu código é testado pelo validador na Seção 5, na qual mostra como o mesmo funciona. Na Seção 6 são mostradas as contribuições que são obtidas utilizando essas ferramentas. E, por fim, na Seção 7, tem-se a conclusão do que foi feito.

2. Metodologia

Inicialmente, foi feito um estudo envolvendo as diretrizes de acessibilidade, tanto para conteúdo *Web Content Accessibility Guidelines 2.0 (WCAG 2.0)* [W3C 2008], quanto para ferramentas de autoria *Authoring Tool Accessibility Guidelines (ATAG)* [W3C 2000].

Posteriormente, verificou-se a acessibilidade em formulários de contato em sites de 109 universidades públicas brasileiras, procurando abranger o máximo possível deste cenário, utilizando diretrizes do WCAG 2.0, na qual constatou-se falta de acessibilidade nos critérios analisados: formulários feitos utilizando tabelas, o que não é recomendando; formulários feitos com *labels*; formulários que utilizam CAPTCHA (que é uma ferramenta contra acessos de robôs ou mecanismos que não sejam pessoas); formulários que validam campos; e formulários que funcionam corretamente [Bittar et al. 2012a]. Após essa verificação, foi feita a análise em ferramentas de autoria, utilizando as diretrizes da ATAG, e foi constatado que nenhuma delas oferece o suporte adequado para desenvolvimento de conteúdo acessível [Bittar et al. 2012b].

Visto este cenário e o estudo realizado por Trewin *et al.* (2010), que mostrou que a maioria dos desenvolvedores não utilizam recursos e diretrizes de acessibilidade no desenvolvimento por falta de conhecimento dos mesmos, faz-se necessário o uso de ferramentas que auxiliem aos desenvolvedores a suprir essa deficiência.

Logo, é feito neste trabalho uma análise de um gerador de formulários acessíveis automatizado, *Quick Form Builder*, mostrando cada etapa para o desenvolvimento dos

¹<http://accessify.com/tools-and-wizards/accessibility-tools/quick-form-builder/>

²<http://accessify.com/about/>

³<http://validator.w3.org/>

formulários, de maneira intuitiva e testando o formulário gerado no validador do W3C. Com isso, pode-se analisar como essas ferramentas podem contribuir no processo de desenvolvimento acessível.

3. Trabalhos relacionados

Bittar *et al.* (2012a) mostram a verificação de formulários de contato em universidades públicas brasileiras e constata a defasagem do uso das diretrizes do WCAG 2.0, visto que haviam formulários feitos com tabelas, outros que utilizam CAPTCHA sem recurso auditivo, o que prejudica o acesso de pessoas com deficiência à informação. Visto que, a leitura do CAPTCHA é feita independentemente da leitura do formulário, pois leitores de tela leem apenas o conteúdo do formulário e é um robô realizando esta leitura, o CAPTCHA precisa prover recursos para ser ouvido.

Bittar *et al.* (2012b) apresentam um estudo sobre as principais ferramentas de autoria utilizadas por desenvolvedores no Brasil tais como: *Dreamweaver*®), Eclipse (Helios)®), *Netbeans 7.1*®), *NVU 1.0*®), e *Microsoft Expression Web*®), foi constatado que nenhuma dessas ferramentas segue corretamente todos os critérios de acessibilidade analisados, o que dificulta o desenvolvimento de sistemas Web acessíveis.

Além de acessibilidade em formulários, existem outros trabalhos, como o apresentado por Santos (2012), que trata da acessibilidade em menus na Web para pessoas de meia idade, verificando as dificuldades enfrentadas por aqueles que possuem problemas de visão. O que também poderia ser evitado com o conhecimento das diretrizes de acessibilidade ou com ferramentas que auxiliassem aos desenvolvedores a seguir esses critérios.

Já em Sousa (2011) é feita uma avaliação em portal de instituições de ensino superior tendo em foco pessoas com deficiência visual, analisando portais da Universidade de Pernambuco (UPE), Universidade Federal do Rio de Janeiro (UFRJ) e Universidade Federal do Estado do Rio de Janeiro (UNIRIO), nos quais também são encontrados problemas de acessibilidade, que poderiam ser evitados.

4. Ferramenta de geração de formulários acessíveis

Para auxiliar no desenvolvimento de formulários acessíveis existem algumas ferramentas disponíveis ao desenvolvedor de rápida e intuitiva manipulação. Ou seja, ferramentas que geram formulários sem utilização de tabelas, utilizando *labels* de maneira adequada, de acordo com o critério 3.3.2 do WCAG 2.0 consistente de que quando o conteúdo exigir a entrada de dados pelo usuário é obrigatório o uso de rótulos, para evitar erros [W3C 2008]. Verificam se o fechamento as *tags* está sendo feito corretamente, o que condiz com o critério 4.1.1 do WCAG 2.0, na qual consiste que os elementos que dispõem de marcas de início e de fim devem ser completos, ou seja, devem ser abertos e fechados corretamente, caso contrário são considerados atributos incorretos ou incompletos [W3C 2008].

Como exemplo, tem-se o *Quick Form Builder* que tem por objetivo gerar conteúdo acessível e também ser de fácil manuseio ao desenvolvedor, o que é extremamente necessário visto que o operador desta ferramenta é o desenvolvedor de sistemas para Web que utilizam formulários de contato.

Na Figura 1, é ilustrada a página principal da ferramenta, em que devem ser definidos campos que o formulário precisa ter. Caso o usuário não saiba o que inserir, existem

alguns exemplos de dados que podem ser inseridos e, também, logo abaixo ao local de inserção, tem-se as instruções que se devem ser seguidas para usar a ferramenta adequadamente.

Figura 1. Página inicial da ferramenta, primeiro passo

Após a conclusão da etapa inicial, ilustrada na Figura 1, pressiona-se o botão *Next Step*, para que seja executado o próximo passo do desenvolvimento do formulário no qual se definirá os tipos destes campos, como pode ser visto na Figura 2.

Figura 2. Segundo passo, opções de marcação

Na Figura 2 é apresentado como se define cada tipo dos campos do formulário bem como os campos obrigatórios a serem preenchidos pelo usuário. Esta etapa é significativamente importante, pois com os campos obrigatórios torna-se mais fácil o preenchimento do formulário.

Na Figura 3 são apresentados os tipos de marcação, como de sintaxe, se será XHTML ou HTML. A forma de envio de dados podem ser por meio dos métodos post ou get. Além disso, o usuário pode definir o nome do formulário, a mensagem de erro caso os dados não sejam inseridos corretamente pelo usuário final e a mensagem do botão de submissão.

Figura 3. Terceiro passo, opções de marcação

Concluindo o passo 3, ocorre a geração do código fonte do formulário, que é ilustrado na Figura 4. Com a geração do código, está finalizado o desenvolvimento do formulário.



Figura 4. Quarto passo, geração de código fonte

Após visualizar o código, o desenvolvedor tem a opção de voltar ao passo anterior, ou começar um novo formulário, retornando ao passo 1.

O código gerado possui os *labels* especificados e validação dos campos, mas ao executar o código é perceptível que não há a marcação dos campos obrigatórios na interface, o que pode prejudicar na prevenção de erros pelo usuário, que é uma falha de acessibilidade. A ferramenta também não gera o recurso de CAPTCHA, o que torna o sistema menos seguro, porém mais acessível.

O formulário da Figura 5 possui os campos de nome, email, assunto e mensagem, que são os campos básicos para um formulário de contato, é uma tela simples e de fácil entendimento para o usuário.

Formulário de Contato

Nome:

Email:

Assunto:

Mensagem:

Enviar

Figura 5. Formulário gerado com a ferramenta Quick Form Builder

Na próxima seção é apresentado o validador de códigos do W3C, que faz a verificação do código gerado pela ferramenta *Quick Form Builder* aqui apresentada, analisando se o mesmo é usável ou não e apontando os erros cometidos.

5. Validador de código do W3C

O validador de código do W3C é um *software* livre utilizado para verificar o código fonte de páginas, apontando erros de grafia, uso impróprio de CSS e também alerta o desenvolvedor sobre potenciais erros de acessibilidade [W3C 2007].

A inserção do código em HTML pode ser feita por três vias, inserindo-se a URL da página que se deseja validar, fazendo *upload* de arquivos locais ou por entrada direta, na qual se coloca o código fonte diretamente no validador, como pode ser observado na Figura 6. Cada aba do validador indica uma forma de entrada do código a ser analisado, é

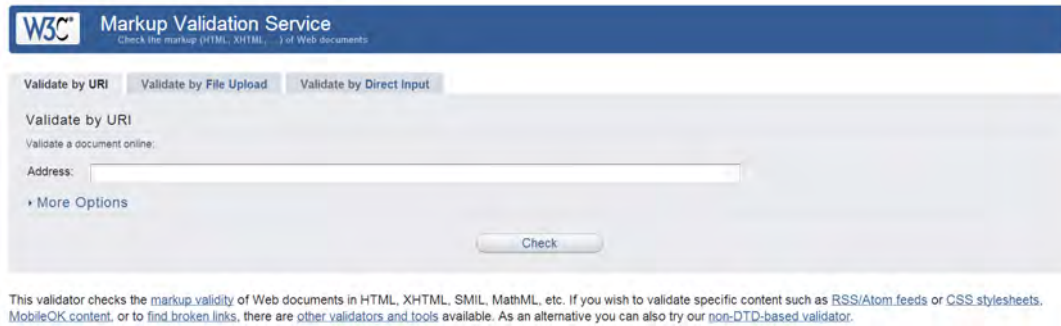


Figura 6. Interface do validador de código do W3C

um sistema simples e intuitivo para os desenvolvedores, que desejam validar seus códigos em HTML.

Na Figura 7 é ilustrada a inserção de código (que foi realizada neste trabalho) e faz-se a validação do mesmo. Neste caso, o mesmo foi verificado com sucesso, ou seja, não possui erros de formação, o que é importante para prover acessibilidade.

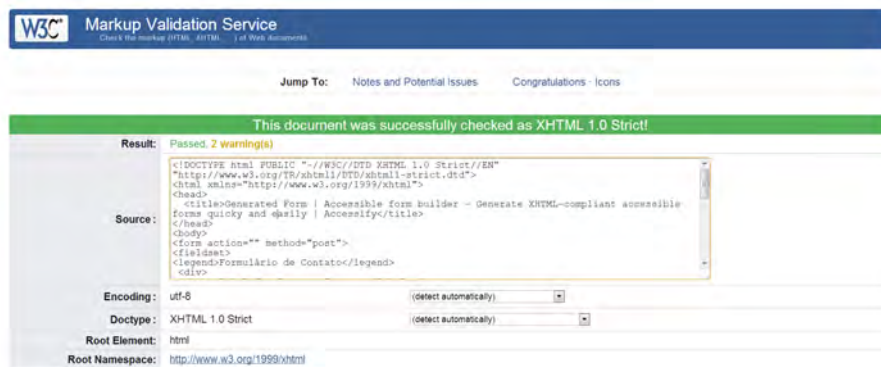


Figura 7. Validando o código

Caso o arquivo possua erros ou *warnings*, o sistema apresenta uma mensagem ao desenvolvedor informando-o sobre os erros e suas respectivas localizações no código, como mostrado nas Figuras 8 e 9, nas quais foram utilizados o mesmo código da Seção 4 removendo-se os cabeçalhos do mesmo.

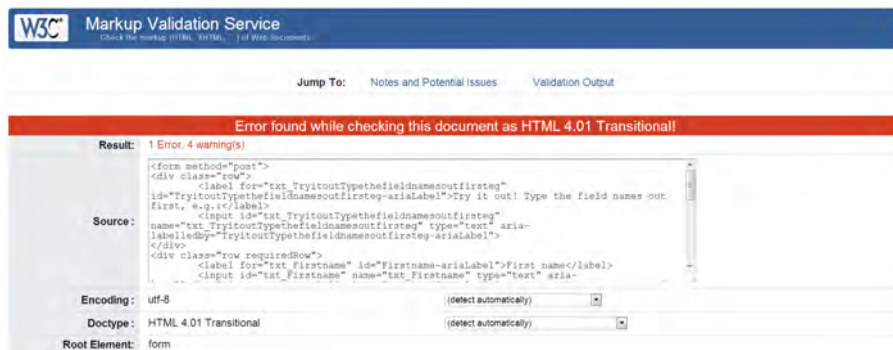


Figura 8. Código com erro e *warning*

Na Figura 8 é visto que 1 erro foi encontrado no documento e 4 *warnings*, logo, o código não passou na avaliação validador. Quando isso ocorre, o desenvolvedor pode adequar seu código no próprio editor da ferramenta e revalidá-lo, ou resubmetê-lo sempre que necessário.

Já na Figura 9 é mostrado como o validador indica ao usuário os erros encontrados e sugestões de correção, o que torna mais fácil a adequação do código, visto que o desenvolvedor pode não ter conhecimento do que fez de errado e nem onde cometeu tal erro.

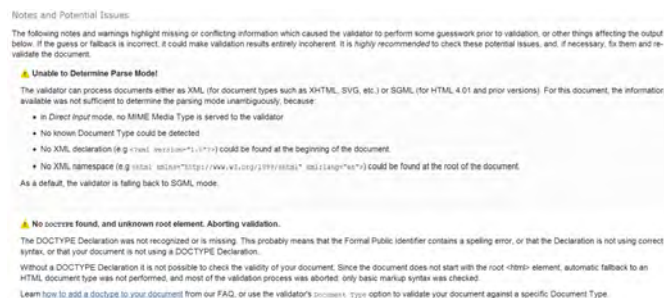


Figura 9. Indicando quais erros encontrados

6. Contribuições da utilização das ferramentas

Com a utilização destas ferramentas torna-se mais fácil ao desenvolvedor prover acessibilidade em formulários, pois pode-se validar o código verificando se está bem formado. Além disso, caso hajam erros, o validador avisa qual tipo e em qual parte do código está, o que auxilia na identificação dos mesmos e na sua correção efetiva.

Assim, com esse auxílio, torna-se mais fácil o desenvolvimento de formulários acessíveis, visto que não é necessário conhecer as diretrizes de acessibilidade pois o validador auxilia nesta parte.

7. Conclusão

Foi mostrado neste trabalho uma ferramenta que auxilia o desenvolvimento de um formulário acessível, sendo a mesma intuitiva e de fácil manuseio. Para tanto, foi realizado teste em um validador de código para inspecionar se o formulário é realmente acessível. Percebeu-se então que, o código gerado pelo gerador de formulário é acessível e usável. Contudo, verifica-se que a ferramenta não indica quais são os campos obrigatórios o que também não foi condenado pelo validador de código. Com isso, pode-se sugerir melhorias para a ferramenta de autoria tais como, mostrar os campos obrigatórios do formulário.

Logo, percebe-se que os desenvolvedores possuem uma alternativa para desenvolver formulários acessíveis, visto que ambas as ferramentas são *software* livre, ou seja, não gera custo adicional e que são de fácil acesso.

É proposto como trabalho futuro a realização de um estudo de caso para melhorar a ferramenta, atrelando-a a um validador de acessibilidade, para que em uma única ferramenta se tenha as correções efetivadas de maneira imediata.

Referências

- Bittar, T., Faria, F., do Amaral, L., and de Mattos Fortes, R. (2012a). An assessment of accessibility in contact forms of brazilian public universities. In *Information Systems and Technologies (CISTI), 2012 7th Iberian Conference on*, pages 1 – 6, Madrid, Espanha. IEEE Xplorer.
- Bittar, T. J., do Amaral, L. A., Faria, F. B., and de Mattos Fortes, R. P. (2012b). Supporting the developer in an accessible edition of web communications: a study of five desktop tools. In *Proceedings of the Workshop on Information Systems and Design of Communication*, ISDOC '12, pages 3–9, New York, NY, USA. ACM.
- Lima, S. T. (2007). Avaliação da acessibilidade de sítios web por meio de m'etricas de software. Mestrado em gestão de conhecimento e da tecnologia da informação, Universidade Católica de Brasília, Brasília - DF.
- Maia, L. S. (2010). Um processo para o desenvolvimento de aplicações web acessíveis. Mestrado em ciências da computação, Universidade Federal de Mato Grosso do Sul.
- Santos, E. P. B. (2012). Acessibilidade em menus de navegação horizontais na web para pessoas de meia-idade. Mestrado em ciências de computação e matem'atica computacional, Universade São Paulo, São Carlos - SP.
- Sloan, D., Heath, A., Hamilton, F., Kelly, B., Petrie, H., and Phipps, L. (2006). Contextual web accessibility - maximizing the benefit of accessibility guidelines. In *Proceedings of the 2006 international cross-disciplinary workshop on Web accessibility (W4A): Building the mobile web: rediscovering accessibility?*, W4A '06, pages 121–131, New York, NY, USA. ACM.
- Sousa, M. F. C. (2011). Acessibilidade web: Uma avaliação em portal de instituições de ensino superior visando pessoas com deficiência visual. Monografia, Universidade de Pernambuco, Pernambuco. p. 89.
- Trewin, S., Cragun, B., Swart, C., Brezin, J., and Richards, J. (2010). Accessibility challenges and tool features: an ibm web developer perspective. In *Proceedings of the 2010 International Cross Disciplinary Conference on Web Accessibility (W4A)*, W4A '10, pages 32:1–32:10, New York, NY, USA. ACM.
- W3C (2000). Authoring tool accessibility guidelines 2.0. <http://www.w3.org/WAI/intro/atag.php>. Acessado em 10 de fevereiro de 2012.
- W3C (2007). Serviço de validação de css. <http://jigsaw.w3.org/css-validator/about.html.pt-BR>. Acesso em 25 de janeiro de 2013.
- W3C (2008). Web content accessibility guidelines (wcag) 2.0. <http://www.w3.org/TR/WCAG20/>. Acessado em 10 de fevereiro de 2012.