

Provendo Acessibilidade em Sites com WAI-ARIA

Andressa Vergutz, Bruno Batista Boniati

Universidade Federal de Santa Maria (UFSM)
Caixa Postal 54 – 98.400-000 – Frederico Westphalen – RS – Brasil
{andressavergutz, brunoboniati}@gmail.com

Resumo. *Observa-se que nos últimos anos houve um grande crescimento na utilização da Web como plataforma de trabalho e comunicação. Em função disso, os sites têm utilizado muitos recursos para prover interfaces ricas. Porém, quanto mais rica é a interface, mais difícil ela pode se tornar para usuários com necessidades especiais. A partir do HTML5 existe uma especificação para prover acessibilidade na Web, trata-se da WAI-ARIA (Web Accessibility Initiative - Accessible Rich Internet Applications). Por meio deste trabalho, pretende-se estudar tal especificação e desenvolver um site que apresente requisitos de acessibilidade.*

Abstract. *It is observed that in last years, there was a large growth in the use of the Web as a working and communication platform. For this reason, the websites have used many resources for provide rich interfaces. However, the richer the interface is, the more difficult it can become for users with special needs. From the HTML5 exists a specification to provide Web accessibility, it is the WAI-ARIA (Web Accessibility Initiative - Accessible Rich Internet Applications). Through this work, it is intended to study such specification and to develop a website that presents accessibility requirements.*

1. Introdução

A WEB como se apresenta hoje, é uma das ferramentas mais revolucionárias de nossa sociedade. Além de ser um espaço livre e democrático, tem permitido uma revolução em nossas relações, tanto pessoais, como profissionais e até mesmo econômicas. Não é desejável que toda essa revolução seja acessível apenas a uma parcela da população, desse modo, é essencial a acessibilidade estar presente na Web a fim de proporcionar igualdade de acesso e oportunidades para todas as pessoas [W3C 2014].

Considerando tal realidade, desenvolvedores e organizações procuram achar soluções para atender o maior número possível de usuários que possuam algum tipo de deficiência. Pode-se citar a W3C que durante anos cria padrões acessíveis a serem seguidos pelos desenvolvedores web. Dessa forma, o objetivo deste trabalho é estudar a especificação WAI-ARIA da W3C, a fim de desenvolver um *site web* acessível. Este *site*, que no momento se encontra em fase de desenvolvimento, abordará o conteúdo do projeto de extensão cujo nome é "Capacitação para o Uso de *Tablets* Educacionais em Espaços Escolares" e objetiva compartilhar materiais que venham auxiliar docentes na utilização do *tablet*. Como o conteúdo da aplicação web será acessível, pretende-se utilizar a ARIA e outras recomendações da W3C conforme necessidades.

2. Web e Acessibilidade

Quando a Web foi concebida por Tim Berners-Lee sua intenção principal propunha uma forma de navegação em documentos de texto. Com o passar do tempo, percebeu-se que

a *Web* era uma ótima forma de distribuição de informações e sua evolução foi exponencial. Hoje, além de textos, é comum encontrar grande quantidade de imagens, sons e vídeos. A forma de interação também evoluiu bastante, não são mais apenas *links* para navegar entre as páginas de textos, hoje é possível realizar operações como: arrastar e soltar, duplo clique, parar sobre algum componente, e, além disso, a forma como os dados são apresentados também pode ser diferente [Watanabe 2014].

Atualmente, as aplicações *Web 2.0* passaram a apresentar maior interatividade, permitindo que o usuário influencie na forma como o conteúdo é apresentado [Watanabe 2014]. Não há dúvidas de que tais aplicações, conhecidas como *Rich Internet Applications* (RIA), podem melhorar a experiência dos usuários na *web*. No entanto, o aumento da interatividade apresenta uma série de barreiras de acessibilidade para usuários que interagem com o navegador utilizando Tecnologias Assistivas, como leitores de tela [Merayo 2011]. Para criar essa interatividade e obter um visual mais sofisticado as aplicações da *Web 2.0* utilizam linguagem de *script* e outras tecnologias avançadas. Porém, grande parte dessas características exige percepção visual do usuário e requerem interações do *mouse* para que possam ser operadas, resultando num impacto negativo na acessibilidade da aplicação [Watanabe 2014].

RIA e componentes de interface Ajax alteram e atualizam dinamicamente a estrutura DOM (*Document Object Model*) de uma página *web* sem a necessidade de gerar uma atualização completa da página [Watanabe 2014]. As Tecnologias Assistivas, em grande parte, não estão preparadas para que novos conteúdos apareçam em lugares arbitrários da estrutura da página. Para tais tecnologias não é possível ler o código fonte e descrever seu comportamento antes que a interação seja realizada.

3. WAI-ARIA

A W3C por meio da WAI (*Web Acessibility Initiative*) trabalha em diversos padrões e recomendações que tem por objetivo melhorar a acessibilidade dos *sites* [Grillo 2014]. A especificação WAI-ARIA, *Accessible Rich Internet Applications Suite*, é umas dessas recomendações que trata da acessibilidade em interfaces ricas. A ARIA ajuda especialmente em conteúdos dinâmicos e controles avançados de interface de usuários, focando principalmente em usuários dependentes de leitores de tela e pessoas que não podem utilizar o *mouse* [W3C 2014].

ARIA define um conjunto de atributos para serem incluídos nos elementos HTML. Esses atributos compreendem as seguintes extensões: estado (*states*), propriedade (*property*) e papel (*role*). Estados e propriedades são usadas para declarar atributos importantes de um elemento e informam a descrição e interação do mesmo. Os papéis indicam o tipo de elemento, ou seja, informará como o elemento se comportará, por exemplo, no elemento LI do MENU o *role* (papel) informará que esta lista se comportará como um item de menu (`<li role="menuitem">`). A Figura 1 permite observar que o atributo *role* informa que essa DIV se comporta como um *slider*, e os outros atributos estão informando propriedades importantes para o elemento que são os valores máximo, mínimo e atual do elemento *slider*. Desse modo, a tecnologia assistiva receberá esses atributos e o usuário com deficiência conseguirá manipular o elemento pelo teclado [Merayo 2011].

```
<div id="slider" role="slider" aria-valuemin="0"  
aria-valuemax="100" aria-valuenow="0"></div>
```

Figura 8. Exemplo de utilização da especificação WAI-ARIA.

4. Trabalhos Relacionados

Analisando trabalhos na área de acessibilidade *Web* com estudos de casos sobre ARIA, podemos identificar algumas iniciativas de pesquisa parecidas com o presente trabalho.

Watanabe (2014) propõe melhoria no processo de avaliação automática de aplicações *web* ricas (RIA). Assim, elabora diferentes estratégias para avaliar automaticamente a acessibilidade considerando a utilização de leitores de tela para a navegação. Após estudos e testes realizados, conclui que para avaliar a acessibilidade dessas aplicações *Web* é necessário não avaliar apenas o conteúdo HTML, mas também o seu comportamento implementado utilizando JavaScript e CSS. Conclui também, que as ferramentas de avaliação automáticas se sobressaem das ferramentas de avaliação de conteúdo estático HTML [Watanabe 2014].

Ghelardi (2012) apresenta impactos da especificação ARIA na construção de aplicações *web* ricas, baseando-se no desenvolvimento de um *player* de mídia acessível. Concluiu que a partir da utilização de especificações *Web* como o HTML5 e ARIA é perfeitamente possível a construção de soluções acessíveis, sem sacrificar a aparência e a interatividade das mesmas [Ghelardi 2012].

5. Resultados Parciais

Salienta-se que o presente trabalho se encontra em fase de desenvolvimento. Entretanto, já foram determinados alguns requisitos da ARIA que serão utilizados no *site web*, três deles serão descritos abaixo.

Um destes requisitos trata da forma de como avisar o usuário de algum erro ocorrido durante o preenchimento de formulários. Geralmente, quando algum campo é preenchido de forma incorreta o usuário é notificado disso com a alteração da apresentação dos campos (por exemplo, muda a cor). Porém, estas práticas não são acessíveis. A fim de resolver estes problemas, ARIA introduz as propriedades `aria-required` e `aria-invalid` para marcar os campos que são obrigatórios e caso houver algum erro de preenchimento deverá avisar o usuário, respectivamente, onde esse aviso será através da tecnologia assistiva. Um exemplo de tais propriedades é ilustrado na figura 2.

```
<label for="email"> * E-mail:</label>  
<input type="text" id="email" aria-required="true"  
aria-invalid="true" aria-label="O campo email é obrigatório">
```

Figura 2. Exemplo de propriedades da especificação WAI-ARIA.

Outro requisito que será utilizado são as *Roles Landmarks* que marcam as principais estruturas da aplicação *web* de forma a facilitar a navegação do usuário com o teclado. As *landmarks* descrevem e marcam estruturas comuns da *web* como menus (*navigation*), conteúdo principal (*main*), rodapés (*contentinfo*), caixa de pesquisa (*search*), entre outros. Com isso, o usuário não precisa ficar pressionando a tecla TAB muitas vezes. A figura 3 ilustra a utilização das *roles* citadas.

```
<div role="contentinfo"></div>  
<div role="banner"></div>  
<div role="main"></div>  
<div role="navigation"></div>  
<div role="search"></div>
```

Figura 3. Exemplo do atributo *Roles Landmarks* da ARIA.

A propriedade `aria-describedby` também será utilizada na implementação da aplicação *web*. Esta propriedade tem como função passar alguma informação importante sobre o elemento, essa informação pode estar relacionada à função do elemento por exemplo. Através dela o usuário dependente de leitores de tela conseguirá saber o que o elemento fará e para que ele serve. A figura 4 abaixo possui um exemplo da utilização desta propriedade.

```
<button aria-describedby="infobutton" aria-label="enviar">Enviar</button>  
<div id="infobutton">Clicando neste botão os dados digitados nos campos acima serão enviados!</div>
```

Figura 4. Propriedade `aria-describedby` da WAI-ARIA.

6. Considerações Finais

A partir do estudo e testes sobre WAI-ARIA, a especificação se mostrou útil e eficiente, através dela é possível resolver vários problemas de acessibilidade tanto em interfaces ricas quanto em aplicações *web* não tão sofisticadas. A utilização da ARIA não afeta muito a implementação, o tempo gasto utilizando ou não a especificação é praticamente o mesmo. Outro ponto relevante é que sua documentação no *site* da W3C é muito rica, possui exemplos e descrições de cada propriedade e atributo da especificação. Desse modo, conclui-se que é de extrema importância abordar essas recomendações e utilizá-las para tornar a *Web* um mundo mais acessível para que todos possam contribuir de alguma forma.

Referências

- Ghelardi, A., Otsuka, J. L. and Kawakami, C. (2012). Acessibilidade na Educação a distância: desenvolvimento de um player de mídia acessível utilizando html5 e wai-aria e sua integração com o moodle. In *Anais do Simpósio Brasileiro de Informática na Educação*, volume 23.
- Grillo, F. D. N. (2014). *Uma ferramenta acessível de apoio à modelagem de software na web*. PhD thesis, Universidade de São Paulo.
- Merayo, R. V. N. (2011). Rich internet applications (RIA) and web accessibility. <http://www.upf.edu/hipertextnet/en/numero-9/ria-and-web-accessibility.html>. Set/2014.
- W3C (2014). Wai-aria overview. <http://www.w3.org/WAI/intro/aria>. Set/2014.
- Watanabe, W. M. (2014). *Avaliação automática de acessibilidade em RIA*. PhD thesis, Universidade de São Paulo.