

Uso de *Design Rationale* para gestão de conhecimento em acessibilidade na Web

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

¹Departamento de Ciência da Computação – Universidade Federal de Goiás
CEP: 75704-020 – Catalão – GO – Brasil

²ICMC – Universidade de São Paulo
Caixa Postal: 668 – São Carlos – SP – Brasil

ferrbontempo@gmail.com, jabur@icmc.usp.br

Abstract. *This paper reports accessibility issues and proposes a technique for using Design Rationale to minimize these problems and make the Web a resource accessible to a greater number of users. Thus it is proposed a case study which there is the positive and negative applications, showing how the Design Rationale can assist in making decisions to achieve a greater Web accessibility.*

Resumo. *Esse artigo relata os problemas de acessibilidade e propõe uma técnica para o uso do Design Rationale para minimizar esses problemas e tornar a Web um lugar acessível para um maior número de usuários. Para tanto é proposto um estudo de caso no qual se verifica os pontos positivos e negativos dos aplicativos, mostrando como o Design Rationale pode auxiliar na tomada de decisões para se chegar a uma maior acessibilidade Web.*

1. Introdução

Atualmente muitos problemas de acessibilidade são encontrados em *sites* [Maia *et al.*, 2010], [Maia, 2010] e [Lima, 2007]. Tal cenário surge, principalmente, da falta de considerações desse tema no momento do desenvolvimento. É verificado em vários projetos que há documentação insuficiente e decisões que não levam em consideração boas práticas de acessibilidade o que, se realizado com sucesso, pode ser útil para reuso em outros desenvolvimentos e treinamento de equipe [Trewin *et al.*, 2010].

Nesse domínio recursos computacionais vêm sendo cada vez mais utilizados e a Web vem sendo o maior intermediador entre as pessoas e a informação. Segundo Tim Berners-Lee, criador da Web, seu poder está em sua universalidade e segundo Ron Mace, o princípio da universalidade é conceber produtos, meios de comunicação e ambientes para serem utilizados por todas as pessoas o maior tempo possível [Tangarife e Alvão, 2005].

Para facilitar o uso e registro de técnicas e decisões de acessibilidade na Web pode ser usada a metodologia de *Design Rationale* (DR), que visa ser uma base útil de dados para reuso e consulta pela equipe. Para tanto é importante conhecer os conceitos de acessibilidade e DR, para tornar a Web um espaço democrático e acessível a um maior número de pessoas que não tenham conhecimentos específicos da mesma.

Uma das formas de amenizar os obstáculos de acesso ao conteúdo Web é a utilização de recomendações que direcionam a promoção de acessibilidade [Maia, 2010]. Para alcançar a acessibilidade é necessário o uso dessas recomendações, que são apresentadas na seção 3, e para validá-la, usa-se uma ferramenta para verificar se um

aplicativo Web é ou não acessível, ou seja, se segue as normas de acessibilidade segundo a legislação vigente.

Para ampliar a acessibilidade em aplicativos e auxiliar na tomada de decisões, propõe-se neste trabalho verificar se o DR é efetivamente útil e como pode ser utilizado no âmbito da avaliação e aplicação de recursos de acessibilidade em sistemas.

De acordo com [Falessi *et al.*, 2006], DR inclui não só as decisões por trás do design, mas também a justificativa para isso, as alternativas consideradas, as compensações avaliadas e os argumentos que levaram à decisão.

Em se tratando de desenvolvimento Web, acessibilidade e DR devem ser considerados para um aplicativo acessível, visto que é necessário realizar uma série de testes verificando se as ferramentas foram adequadamente utilizadas e o resultado do uso das mesmas.

A divisão do artigo descreve primeiramente a metodologia de pesquisa utilizada para realização do estudo, na seção 2. A seção 3 conceitua acessibilidade e mostra o que é WCAG, apresentando os principais erros de acessibilidade cometidos. Na seção 4, é feita uma breve explicação sobre o que é o DR, propondo um método para uso do DR, dividido em três momentos, e verifica-se também onde o DR é útil. Na seção 5 são apresentados os resultados e na seção 6 os trabalhos relacionados. Por fim, tem-se uma conclusão do estudo feito.

2. Metodologia

A metodologia para esta pesquisa envolveu a análise de estudos científicos realizados e objetivando-se o estudo de caso, pautados na inspeção de diversos *sites* desenvolvidos, para verificação de seus principais pontos positivos e negativos de acessibilidade, buscando apontar as falhas e possíveis melhorias.

Em um primeiro momento foi feita uma investigação do problema, verificando onde existem falhas e como fazer para corrigi-las, analisando processos de desenvolvimento e vendo onde podem ser incluídas as inserções de decisões. Houve também um estudo envolvendo engenharia de *software*, verificando onde pode ser empregado o DR pra contribuir para o projeto de desenvolvimento, sem prejudicar o trabalho feito pela equipe.

Foram escolhidas para esse artigo as diretrizes do WCAG, por serem oficiais e terem evoluído bastante acompanhando a evolução da Web e suas necessidades. E o uso do *Design Rationale* nas tomadas de decisões. Estas, por sua vez, são importantes, pois é quando se decide o que vai ser implementado e como será desenvolvido. Ela influencia no projeto final, pelo fato de ser um roteiro a ser seguido pelos desenvolvedores.

O DR é útil segundo [Dutoit *et al.*, 2006], por permitir e promover a colaboração de conhecimento entre membros de equipe, facilitar a manutenção e o reuso, melhorar a qualidade dos artefatos e embasar novas decisões de projeto.

Depois foi feito um estudo sobre acessibilidade e DR, e verificaram-se como esses conceitos poderiam ser integrados ao desenvolvimento E, por fim, foi desenvolvida uma proposta para utilização dos mesmos na melhoria dos problemas detectados.

3. Acessibilidade e diretrizes do W3C

Acessibilidade é um conceito que possui várias interpretações, mas para desenvolvimento Web, é algo que deve permitir que pessoas com necessidades especiais sejam capazes de usá-la. Mais concretamente, significa uma Web projetada de modo a

que essas pessoas possam perceber, entender, navegar e interagir de uma maneira efetiva com os recursos disponibilizados, bem como criar e contribuir com conteúdos [Lima, 2007].

A acessibilidade Web é importante tanto para deficientes, quanto para pessoas que não têm nenhuma deficiência, porém têm alguma limitação momentânea [Maia, 2010]. Entretanto existem poucas práticas aplicadas no desenvolvimento Web, e muitas decisões de projeto poderiam estar baseadas nas práticas documentadas por órgãos oficiais de regulamentação da Web, como o *Web Content Accessibility Guidelines* (WCAG).

O WCAG é o conjunto de recomendações, que foi lançado em 1999¹, que define várias *guidelines* e *checkpoints* para guiar o desenvolvimento de conteúdo Web visando a acessibilidade [Lara, 2005], [W3C, 1999].

As *guidelines* da WCAG 1.0, são as seguintes: fornecer alternativas equivalentes ao conteúdo sonoro e visual; não recorrer apenas à cor para expressar significado de informações; utilizar corretamente marcações e folhas de estilo; indicar claramente qual o idioma utilizado.

Inclui também criar tabelas passíveis de transformação harmoniosa; assegurar que as páginas dotadas de novas tecnologias sejam transformadas harmoniosamente; assegurar o controle do usuário sobre as alterações temporais do conteúdo; assegurar a acessibilidade direta de interfaces do usuário integradas; projetar páginas considerando a independência de dispositivos.

A WCAG 1.0 engloba também utilizar soluções de transição; utilizar tecnologias e recomendações da W3C - grupo responsável por alavancar iniciativas para promover acessibilidade na Web; fornecer informações de contexto e orientações; fornecer mecanismos de navegação claros; e assegura a clareza e a simplicidade dos documentos [W3C, 1999].

Já a WCAG 2.0 possui o mesmo objetivo de sua antecessora, mas propõe algumas metas adicionais com o objetivo de complementar a WCAG 1.0, que são: assegurar que os requisitos possam ser aplicados a tecnologias; assegurar que os requisitos de conformidade sejam claros; entregar o projeto com a facilidade de uso em mente; escrever para um público mais diversificado; identificar claramente quem se beneficia com conteúdo acessível; e garantir que a revisão seja "compatível para trás e para frente" [W3C, 2008].

As *guidelines* da WCAG são importantes para guiar os desenvolvedores, auxiliando também em critérios que devem ser utilizados no *design rationale*, visando corrigir os erros de acessibilidade e deixando a tecnologia inclusiva às pessoas com deficiências ou com dificuldades.

3.1. Os Principais Erros de Acessibilidade Cometidos

Nem sempre os aplicativos desenvolvidos têm recursos acessíveis. E isso é responsabilidade dos desenvolvedores de *sites* e aplicativos. Um dos grandes problemas de acessibilidade se dá pela falta de informação dos desenvolvedores, que às vezes nem procuram pesquisar sobre o assunto. Conhecimento de acessibilidade não correlaciona com experiência em desenvolvimento Web nessa amostra [Trewin *et al.*, 2010].

Outros fatores limitantes para acessibilidade são: falta de tempo, limitações de tecnologia, falta de conhecimento, orçamento e conflitos com outros requisitos [Trewin

¹ www.w3.org/

et al., 2010]. A maioria dos desenvolvedores não tem tempo para testar se o aplicativo está dentro das especificações de acessibilidade, portanto não a testam adequadamente. Outros alegam que a tecnologia é limitada, pelo fato de não terem acesso às ferramentas importantes de teste de acessibilidade, o que pode ocorrer devido à falta de conhecimento, pois existem várias ferramentas livres para verificação e avaliação de acessibilidade.

Existem também alegações de que o orçamento é baixo para a realização de testes rigorosos e que a acessibilidade conflita com outros requisitos, tais como tempo de resposta do aplicativo.

Os principais erros de acessibilidade giram, então, em torno do desenvolvimento, no qual, devido à falta de planejamento, desenvolvedores não se preocupam em adicionar acessibilidade em seus sistemas prejudicando uma gama de usuários.

Outro ponto que pode se tornar agravante, é que as ferramentas não são 100% confiáveis, o que gera desconfiança dos desenvolvedores que a utilizam [Trewin *et al.*, 2010]. Percebe-se, entretanto, que não há reclamações por parte de vários usuários, por isso os desenvolvedores não se preocupam tanto em pesquisar sobre acessibilidade para colocar seus aplicativos dentro das normas.

4. Design Rationale

[Gruber e Russel, 1991] definem *Design Rationale* (DR) como referência ao raciocínio que justifica um projeto e às descrições que justificam escolhas de estruturas sobre demais alternativas. [MacLean, 1989] e [Lee, 1997] consideram que o DR não apenas inclui a descrição do artefato em potencial, mas também as justificativas das decisões, as experiências, alternativas e argumentações que levaram à decisão.

[Lara, 2005] define *Design Rationale* como possuidor das informações adicionais e relevantes à documentação básica do processo de tomada de decisões durante a elaboração de um artefato do projeto, tais como o raciocínio do projeto, as alternativas consideradas, as discussões e razões que conduziram à decisão final.

Já [Wang e Burge, 2010] mostram que o DR pode ser usado para captura e gerenciamento de conhecimento arquitetural, sendo de suma importância em projetos de *software* e isso inclui informações de ambiente e razões que motivaram o *design* e o processo de negociação que definem o resultado final de um produto.

O conhecimento tem se demonstrado fator decisivo no mundo atual, estando no foco das instituições que precisam utilizar o mesmo, efetuando decisões constantemente. De encontro a esse panorama, no DR são valorizados a captura e o registro de decisões realizadas em projetos e seus impactos, bons ou ruins. Isso porque tais decisões podem causar o sucesso ou o fracasso de um projeto. Documentar experiências é importante para formar uma base de conhecimento para futuros projetos evitando retrabalhos e possibilitando a tomada de decisões melhores.

O DR colabora também para identificação de premissas inadequadas e diminuição da tendência dos projetistas em não perceber alternativas possíveis sobre decisões importantes.

Embora o DR apresente muitas vantagens, ele tem sido pouco utilizado na prática. Dentre alguns problemas detectados pode-se destacar: o tempo adicional gasto para armazenamento do DR, a diferença de formato entre a informação fornecida pelos desenvolvedores e sua representação no DR e a dificuldade de encontrar um sistema de DR que atenda a todas as necessidades de organização e ofereça um mecanismo de recuperação eficiente [Lara, 2005].

4.1. Onde o *Design Rationale* é útil

O DR é útil, pois auxilia muito nas tomadas de decisões, separando as informações de acordo com a importância ou outros aspectos para sua utilização, tornando mais fácil a documentação.

Design Rationale traz benefícios no processo de *software*. Contribui para a redução da arbitrariedade, tornando o processo democrático na tomada de decisões, visto que as justificativas e os argumentos são arquivados e poderão ser recuperados. Contribui para a compreensão do sistema, diminuindo a complexidade de manutenção e ajudando nas avaliações das conseqüências que podem surgir devido a alterações do sistema [Lara, 2005].

Ele pode contribuir para melhor comunicação entre os membros da equipe do projeto, entre os projetistas e os usuários do sistema, auxiliando também novos integrantes do grupo. Auxilia no acompanhamento e descobertas de erros durante o desenvolvimento do projeto e possibilita que razões, discussões e decisões de um projeto sirvam de base para o desenvolvimento de outros projetos [Lara, 2005].

Para acessibilidade os benefícios do DR, auxiliam bastante, visto que as tomadas de decisões são arquivadas, os desenvolvedores que desejam incluir acessibilidade em seus aplicativos, não perdem tanto tempo. Na documentação já pode ser visto quais diretrizes foram usadas, onde devem ser usadas e como devem ser realizados os testes de acessibilidade.

5. Resultados

O *Design Rationale* deve ser usado em vários momentos de desenvolvimento de sistemas, pois é importante para formar uma base histórica de decisões de desenvolvimento, agora agregando características de acessibilidade com reuso de experiências listadas nesse banco de dados. Ele auxilia também para que novos desenvolvedores verifiquem decisões anteriores e se integrem na equipe mais facilmente, e em caso de mudança de tecnologia poderão ser vistas as principais decisões.

Nesse contexto são destacados os três momentos em que o DR deve ser utilizado:

5.1. Antes do desenvolvimento

Antes de um projeto ser executado, devem ser estabelecidos os objetivos e escopos do produto, soluções alternativas devem ser consideradas e as restrições técnicas e gerenciais devem ser identificadas. Com essas informações pode ser feito um planejamento, que é importante para definir estimativas de custos razoáveis e precisas, avaliações efetivas de risco, relações realísticas de tarefas de projeto ou cronograma de projeto gerenciável [Pressman, 2006].

Nesse contexto, em relação à acessibilidade, é preciso planejar a arquitetura e estrutura do sistema para suportar as modificações que devem ser feitas e os testes necessários. Além disso, é necessário treinamento adequado aos desenvolvedores e, feito o planejamento, a aquisição das ferramentas necessárias para o desenvolvimento.

O registro das informações relativas a decisões é muito importante, pois soluções discutidas e adotadas em um projeto podem ser relevantes também em outros projetos, uma vez que erros já cometidos poderiam ser evitados e alternativas já investigadas poderiam ser mais bem aproveitadas [Lara 2005].

A ideia é associar as decisões de projeto com as diretrizes existentes. Fazendo um registro completo, que possa ser reutilizado para outras tomadas de decisão. Na figura 1 é mostrado um exemplo de funcionamento do DR na tomada de decisões.

Todas as decisões, sendo positivas ou negativas, precisam ser registradas para se ter um controle do que já foi pensado e do que precisa ser mudado, para evitar perda de tempo e de recursos.

A interface de usuário para o registro de decisões de acessibilidade. No topo, há o título "Atividade 2.2" e um link "Clique Aqui para cadastrar" rotulado com a letra **A**. Abaixo, há um campo "Principais Diretrizes:" com uma lista suspensa rotulado com a letra **B**. A data "Data: 16/08/2011 às 18:32" é rotulada com a letra **C**. O conteúdo principal da decisão é rotulado com a letra **D**. Abaixo, há um campo "Comentários:" rotulado com a letra **E** e um botão "Salvar". O campo para redigir o comentário é rotulado com a letra **F** e possui um botão "Enviar comentário" na base.

Figura 1. Exemplo do registro e uso do DR na tomada de decisões

Na figura 1 é mostrado como o DR é útil na tomada de decisões. Em **A** se tem o comando onde o usuário pode se cadastrar e ter acesso às informações do documento. Em **B** é feita a seleção das principais diretrizes a serem discutidas para determinado projeto, onde está **C** tem-se o dia e o horário em que foi proposta a decisão de acessibilidade.

Em **D** é descrita a decisão de acessibilidade a ser tomada, **E** mostra onde são visíveis os comentários e **F** onde se devem redigir os comentários das pessoas cadastradas. No botão "salvar" o redator da decisão salva a descrição feita e em "enviar comentários" são salvos os comentários obtidos.

5.2. Durante o desenvolvimento

Durante o desenvolvimento é interessante ter uma equipe para verificar e avaliar a acessibilidade diariamente, e registrando tudo que está acontecendo. Visto que empresas pequenas não possuem esses recursos, os próprios inspetores do sistema devem se atentar a esses fatos, pois na competição do mercado a acessibilidade vem sendo um grande diferencial em sistemas de vários portes.

É importante também nessa fase, o registro de tudo que está sendo verificado, se tudo está ocorrendo como o planejado, se as metas estão sendo alcançadas e a acessibilidade está sendo bem empregada. Com o registro, desenvolvedores e toda a equipe ficam cientes das decisões tomadas, facilitando o trabalho e possibilitando o reuso do documento para outros sistemas a serem desenvolvidos.

5.3. Após o desenvolvimento

Após o desenvolvimento é necessário realizar testes. Uma decisão importante que deve ser registrada é qual tipo a ser realizado. Com os testes é possível verificar a acessibilidade de um *site*, devido às avaliações realizadas. Dependendo da natureza dos requisitos de acessibilidade, diferentes técnicas de inspeções e testes de acessibilidade Web podem ser combinados, tais como: avaliação com ferramentas automatizadas, julgamento de especialistas em acessibilidade e testes com usuário [Maia *et al.*, 2010].

As ferramentas automatizadas de avaliação de acessibilidade Web determinam a conformidade do *site* à recomendação a que ele foi submetido e ajudam os revisores nos ajustes necessários quando o mesmo não está de acordo com tal recomendação [Maia *et al.*, 2010].

Se erros forem encontrados, a equipe precisa discutir como corrigir os mesmos sem maiores custos para correção. Essa discussão precisa ser bem rigorosa devido ao fato de que correção de erros se torna muito dispendiosa depois do *software* pronto, e os desenvolvedores tem um orçamento a cumprir. Todas as decisões também precisam ser registradas para que o cliente entenda o que está sendo feito no sistema, e porque tal orçamento.

6. Trabalhos relacionados

Nesta seção são apresentados alguns trabalhos relacionados ao gerenciamento de acessibilidade em sistemas Web. São mostrados também os projetos que estão sendo desenvolvidos para a melhoria das tomadas de decisões.

Em [Falessi *et al.*, 2005], é mostrada a necessidade de melhoria na colaboração entre as tomadas de decisão do projeto, refinando a rastreabilidade de requisitos e otimizando a utilização de novas tecnologias. Assim é desenvolvida uma técnica de documentação específica para impulsionar os objetivos de decisão e projetos alternativos disponíveis.

Em [Gunderson *et al.*, 2006], os autores mostram algumas ferramentas de avaliação de acessibilidade e como devem ser utilizadas para se conseguir resultado esperado. A mesma abordagem é usada por [Maia, 2010], que propõe um processo para o desenvolvimento de aplicações Web acessíveis, onde são mostrados a importância da acessibilidade e também de planejamento, bem como a validação dos mesmos.

Em [Lara, 2005], é mostrado a utilização do *Design Rationale*, relatando a importância do registro das tomadas de decisão, focando em uma ferramenta de registro de decisões. Mas não é mostrado especificamente em relação à acessibilidade.

Os trabalhos da literatura pesquisada que mais se aproximaram em relação ao sugerido aqui foi em avaliação [Lara, 2005], [Falessi *et al.*, 2005] e [Gunderson *et al.*, 2006], conceituação, mas não integrando ao processo de desenvolvimento nem usando DR, que é o objetivo deste trabalho.

7. Conclusão

Como visto neste artigo, a acessibilidade tem grande importância no produto final dos projetos, para pessoas com deficiências físicas ou cognitivas. Para tanto é importante realizar um planejamento “bem feito” inserindo considerações relacionadas a essas diretrizes.

Com uma maior acessibilidade as pessoas até então “excluídas” terão liberdade para acessar as informações que desejarem, quando e onde quiserem. Essa técnica é importante, pois é uma maneira de se inserir o tema acessibilidade próximo aos

desenvolvedores, junto ao seu ambiente de trabalho e registrando decisões de projeto relacionado com as diretrizes existentes para o domínio, separando informações por ordem de importância ou outros aspectos. Isto facilita bastante em sua reutilização para outros projetos, sendo assim mais ágil o desenvolvimento gastando-se menos tempo.

Sendo assim o uso do *Design Rationale* pode se tornar um grande aliado dos desenvolvedores, que por sua vez se tornaram pessoas aptas a entender as dificuldades alheias, diminuindo as adversidades existentes entre Web e pessoas com deficiências ou necessidades especiais. O importante é a inclusão digital para os usuários e ferramentas inovadoras de trabalho para os desenvolvedores que sempre precisam se atualizar.

8. Referências

- W3C, (2008), “Web Content Accessibility Guidelines (WCAG) 2.0”, disponível em <http://www.w3.org/TR/WCAG20/>, acessado em 14 de agosto de 2011.
- Dutoit, A. H., R. McCall, I. Mistrík, e B. Paech. “Rationale Management in Software Engineering: Concepts and Techniques”. A. H. Dutoit and R. McCall and I. Mistrík and B. Paech (eds.), *Rationale Management in Software Engineering*, Springer, p. 1-48, 2006.
- Falessi, D., Cantone, G., Becker, M., “Documenting Design Decision Rationale to Improve Individual and Team Design Decision Making: An Experimental Evaluation”, publicado em *ISESE '06*, setembro 2006, páginas 134 - 143.
- Grubber, T. R. e D. M. Russel. “Design Knowledge and Design Rationale: A Framework for Representation, Capture, and Use”. Technical Report KSL 90-45, Knowledge Systems Laboratory, Standford, California, 40p., 1991.
- Gunderson, J., Rangan, H., Hoyt, N., “Functional Web Accessibility Techniques and Tools from the University of Illinois”, publicado em *Assets '06*, outubro 2006, páginas 269 - 270.
- Lara, S., (2005) “Um suporte à captura informal do Design Rationale”, Dissertação – Instituto de Ciências Matemáticas e de Computação (ICMC – USP), São Carlos - SP.
- Lee, J. *Design Rationale Systems: Understanding the Issues*. IEE expert/Intelligent Systems and Their Applications, v. 12, n. 3, p. 78-85, 1997.
- Lima, S. T. (2007), “Avaliação de acessibilidade de sítios Web por meio de métricas de software”, Dissertação – Universidade Católica de Brasília, Brasília – DF.
- Maia, L., (2010) “Um processo para o desenvolvimento de aplicações Web acessíveis”, Dissertação – Universidade Federal do Mato Grosso do Sul.
- Maia, L., Turine, M., Sandim, H., Paiva, D., “Um modelo para o Desenvolvimento de Aplicações Web Acessíveis”, publicado em *WebMedia*, 2010, páginas 235 – 242.
- Maclean, A.; R M. Young e T. Moran. "Design rationale: the argument behind the artifact", 1989. SIGCHI Bull. 20, pp. 247-252 114-125
- Pressman, R. (2006), “Engenharia de Software”, McGraw-Hill, 6ª edição, página 483.
- Tangarife, T., Alvão, C., “Estudo Comparativo Utilizando uma Ferramenta de “Avaliação de Acessibilidade para Web”, publicado em *CLIHC '05*, outubro de 2005, páginas 313 - 318.
- Trewin, S., Cragun, B., Swart, C., Brezin, J., Richards, J., “Accessibility Challenges and Tool Features: An IBM Web Developer Perspective”, publicado em *W4A '10*, abril de 2010.
- Wan, W., Burge, J. E., “Using Rationale to Support Pattern-Based Architectural Design”, publicado em *SHARK '10*, maio de 2010.