

---

Um processo para sistemas web com foco em  
acessibilidade e usabilidade

*Ana Luiza Dias*

---



SERVIÇO DE PÓS-GRADUAÇÃO DO ICMC-USP

Data de Depósito:

Assinatura: \_\_\_\_\_

# Um processo para sistemas web com foco em acessibilidade e usabilidade

**Ana Luiza Dias**

***Orientadora: Profa. Dra. Renata Pontin de Mattos Fortes***

***Co-orientador: Prof. Dr. Paulo César Masiero***

Tese apresentada ao Instituto de Ciências Matemáticas e de Computação - ICMC-USP, como parte dos requisitos para obtenção do título de Doutor em Ciências - Ciências de Computação e Matemática Computacional. *VERSÃO REVISADA*

**USP – São Carlos**  
**Outubro de 2014**

Ficha catalográfica elaborada pela Biblioteca Prof. Achille Bassi  
e Seção Técnica de Informática, ICMC/USP,  
com os dados fornecidos pelo(a) autor(a)

D541p      Dias, Ana Luiza  
Um processo para sistemas web com foco em  
acessibilidade e usabilidade / Ana Luiza Dias;  
orientadora Renata Pontin de Mattos Fortes; co-  
orientador Paulo César Masiero. -- São Carlos, 2014.  
275 p.

Tese (Doutorado - Programa de Pós-Graduação em  
Ciências de Computação e Matemática Computacional) --  
Instituto de Ciências Matemáticas e de Computação,  
Universidade de São Paulo, 2014.

1. Engenharia Web. 2. Acessibilidade. 3.  
Usabilidade. 4. Interação Humano-Computador. I.  
Fortes, Renata Pontin de Mattos, orient. II.  
Masiero, Paulo César, co-orient. III. Título.

# Agradecimentos

---

---

Agradeço primeiramente a Deus pela vida, por iluminar meu caminho, me acompanhar em todos os momentos e por me dar a oportunidade de ter os melhores pais do mundo. Mãe Zeneida e Pai Albano, nunca haverá palavras suficientes para que eu possa agradecer todo o amor, amizade, dedicação, incentivo, parceria... enfim, tudo o que vocês representam em minha vida. Mil vezes obrigada. Deixo claro que sem vocês, eu jamais teria chegado até aqui!

Jailson que se tornou meu marido no início do Doutorado, mas que há mais de doze anos sempre foi o meu grande companheiro, muito obrigada! Agradeço por ser tão compreensivo comigo, por aguentar todos esses anos de muito trabalho sem pedir nem exigir a atenção que eu deveria ter lhe dado. Que eu consiga retribuir à altura e que essa cumplicidade permaneça por muitos e muitos anos.

Agradeço a meu querido irmão, Albaninho. Sem dúvidas você é um grande amigo e incentivador, que está sempre ao meu lado participando de todos os momentos. Obrigada por motivar-me e orientar-me sempre da melhor maneira. Obrigada Lu, Ike e agora Bia, cunhada e sobrinhos respectivamente, que alegram e adoçam a minha vida.

Obrigada Profa. Renata, que foi muito mais que minha orientadora, mas uma mãe amiga em todos os momentos. Você me ensinou muitas coisas durante esses anos tranquilos que convivemos juntas, entre elas a sabedoria para superar os obstáculos da vida com serenidade e firmeza. Você é uma mulher guerreira, a qual eu me orgulho muito.

Obrigada Prof. Masiero, o título de co-orientador é pouco pra você, que sempre esteve disponível para todas as questões que eu precisei. Para mim, você é um exemplo de professor, aquele que inspira e estimula a buscar o melhor caminho. A parceria Rê, Masiero e Ana foi fantástica assim como vocês são e não há como agradecer por todos esses anos de aprendizado.

Agradeço aos colegas de pesquisa, Willian, Filipe, Guilherme, Silvana, Thiago, Johana, Eduardo, Rafael, Elias, bem como os alunos de Iniciação Científica Lucas, Matheus, Fábio, Vinício e Maiser. Obrigada por todas as ajudas e sugestões para a concretização deste trabalho.

Minha família e meus amigos também merecem um agradecimento especial. Eles alegram minha vida e fazem com que as dificuldades sejam superadas com mais facilidade e as alegrias sejam multiplicadas quando nos encontramos.

Agradeço à CAPES pelo apoio financeiro durante grande parte deste Doutorado, bem como à USP pelo financiamento nas viagens internacionais para eventos científicos.

Agradeço a todos do ICMC, professores e funcionários que contribuíram para que este trabalho fosse realizado.

Agradeço à EMBRAPA, por oferecer a oportunidade de me dedicar à finalização deste Doutorado por alguns meses.

Por fim, agradeço a todos aqueles que sempre se dispuseram e atenderam as minhas solicitações de colaboração com esta pesquisa.

Muito obrigada!



“Nunca deixe que lhe digam que não vale a pena  
Acreditar no sonho que se tem  
Ou que seus planos nunca vão dar certo  
Ou que você nunca vai ser alguém  
Quem acredita sempre alcança...”

(Mais uma Vez – Legião Urbana)





# Resumo

---

Com o aumento do uso e da complexidade de sistemas web, o desenvolvimento de tais sistemas com qualidade exige a adoção de uma abordagem sistemática e bem definida. Assim, a engenharia web é uma disciplina essencial que considera, além de características da engenharia de software, fatores inerentes aos sistemas web, como a multiplicidade de perfis de usuários. A engenharia web é apoiada por processos, métodos, técnicas e ferramentas que são elementos fundamentais para o desenvolvimento de sistemas web, os quais devem ser adequados para fornecer suporte às ações inerentes ao projeto e à implementação. Esses elementos devem ser selecionados, combinados e tecnicamente implementados de modo a produzir um sistema web acessível e usável. Nesta tese é proposto um processo de desenvolvimento que possui fases gerais bem definidas para a inserção de requisitos de acessibilidade e usabilidade no desenvolvimento de sistemas web, garantindo o seu uso pela maioria das pessoas e facilitando seus meios de acesso. Um estudo de caso foi realizado para verificar a efetividade da aplicação do processo formalizado, o qual possibilitou o desenvolvimento de um sistema acadêmico de agendamento de bancas. Considerando-se a dificuldade prática de avaliar diretamente o processo, foram realizados um experimento controlado e um estudo de viabilidade comparando o sistema acadêmico desenvolvido com outro sistema de mesmo propósito e funcionalidades, mas desenvolvido de maneira *ad-hoc*. Por meio das avaliações realizadas nos dois sistemas de agendamento de bancas, indiretamente avaliou-se o processo formalizado e foram encontrados fortes indícios sobre a efetividade do processo proposto. Adicionalmente, foi criado um instrumento de medição objetivo e quantitativo das características de acessibilidade e usabilidade de um sistema web. Foi também criado um método para avaliar, comparar e melhorar a acessibilidade e a usabilidade de sistemas web existentes. Tanto o instrumento de medição quanto o método de avaliação podem ser aplicados, independentemente, a qualquer sistema web.



# Abstract

---

With the increasing use and complexity of web systems, the development of such systems with quality demands the adoption of a systematic and well-defined approach. Thus, web engineering is an essential discipline that considers, in addition to characteristics of software engineering, factors inherent to web systems, such as the multiplicity of the users' profiles. Web engineering is supported by engineering processes, methods, techniques and tools that are fundamental elements to the development of web systems, which must be adequate to support the activities of design and implementation. These elements should be selected, combined and technically implemented to produce an accessible and usable web system. This thesis proposes a development process with well-defined phases for including requirements of accessibility and usability in the development of web systems, enabling their use by most people, facilitating their means of access and foremost. A case study was conducted to verify the effectiveness of applying the process proposed and with this objective, a system to schedule thesis and dissertation presentations was developed. Considering the practical difficulty of directly measuring the proposed process, a controlled experiment and a feasibility study was conducted to compare the academic system developed with a legacy system with the same purpose and functionality but developed using an ad-hoc process. Considering the evaluation of both systems, the development process was indirectly evaluated and evidences related to its effectiveness have been identified. Additionally, an objective and quantitative method for measuring accessibility and usability of web systems was created. Finally, it was also created a method to evaluate, compare and improve the accessibility and usability of existing web systems, which was used to evaluate the system developed using the proposed process. Both the measuring instrument and the evaluation method can be applied, independently, to any web system.



# Sumário

---

---

<b>1 INTRODUÇÃO.....</b>	<b>1</b>
1.1. CONTEXTUALIZAÇÃO .....	1
1.2. MOTIVAÇÃO .....	2
1.3. QUESTÃO DE PESQUISA E DEFINIÇÃO DO ESCOPO .....	3
1.4. OBJETIVO.....	6
1.5. PROCEDIMENTOS METODOLÓGICOS .....	6
1.6. ORGANIZAÇÃO.....	9
<b>2 ACESSIBILIDADE E USABILIDADE.....</b>	<b>11</b>
2.1 CONSIDERAÇÕES INICIAIS.....	11
2.2 ACESSIBILIDADE .....	11
2.2.1 Diretrizes de acessibilidade.....	12
2.2.2 Desafios encontrados de acordo com os tipos de deficiências.....	15
2.3 USABILIDADE .....	17
2.3.1 Heurísticas de usabilidade .....	17
2.3.2 Padrões web .....	19
2.4 ACESSIBILIDADE VERSUS USABILIDADE .....	21
2.5 CONSIDERAÇÕES FINAIS.....	23
<b>3 ENGENHARIA WEB: PROCESSOS, ABORDAGENS DE AVALIAÇÃO E MÉTRICAS .....</b>	<b>25</b>
3.1 CONSIDERAÇÕES INICIAIS.....	25
3.2 PROCESSOS DE DESENVOLVIMENTO .....	26
3.2.1 Requisitos de software e a web 2.0.....	28
3.2.2 Desenvolvimento de sistemas web.....	30
3.3 AVALIAÇÃO DE SISTEMAS WEB .....	40
3.3.1 Avaliação de acessibilidade e usabilidade em sistemas web.....	41
3.4 MEDIÇÕES E MÉTRICAS DE SISTEMAS .....	47
3.5 CONSIDERAÇÕES FINAIS.....	50
<b>4 PROCESSO DE DESENVOLVIMENTO DE SISTEMAS WEB COM FOCO EM</b>	
<b>ACESSIBILIDADE E USABILIDADE .....</b>	<b>53</b>
4.1 CONSIDERAÇÕES INICIAIS.....	53
4.2 FUNDAMENTAÇÃO DO PROCESSO .....	54
4.3 O PROCESSO PDWAU .....	57
4.3.1 Comunicação.....	59

4.3.2	<i>Planejamento</i> .....	62
4.3.3	<i>Modelagem</i> .....	65
4.3.4	<i>Construção</i> .....	69
4.3.5	<i>Implantação</i> .....	72
4.4	<b>ESTUDO DE CASO: APLICAÇÃO DO PROCESSO PDWAU</b> .....	74
4.4.1	<i>Comunicação</i> .....	75
4.4.2	<i>Planejamento</i> .....	78
4.4.3	<i>Modelagem</i> .....	82
4.4.4	<i>Construção</i> .....	89
4.4.5	<i>Implantação</i> .....	99
4.5	<b>O SISTEMA AGENDÁLOCA</b> .....	101
4.6	<b>REÚSO DA INTERFACE DO AGENDÁLOCA</b> .....	104
4.7	<b>TRABALHOS RELACIONADOS</b> .....	108
4.8	<b>CONSIDERAÇÕES FINAIS</b> .....	113
<b>5 AVALIAÇÃO EXPERIMENTAL DO PDWAU</b> .....		<b>115</b>
5.1	<b>CONSIDERAÇÕES INICIAIS</b> .....	115
5.2	<b>PLANEJAMENTO DA AVALIAÇÃO EXPERIMENTAL</b> .....	116
5.2.1	<i>Definição do experimento</i> .....	116
5.2.2	<i>Definição das hipóteses</i> .....	118
5.2.3	<i>Definição do projeto do experimento</i> .....	119
5.2.4	<i>Definição da operação do experimento</i> .....	123
5.3	<b>EXECUÇÃO DA AVALIAÇÃO EXPERIMENTAL</b> .....	125
5.3.1	<i>Realização da avaliação sob o ponto de vista de inspeção</i> .....	125
5.3.2	<i>Realização da avaliação sob o ponto de vista de ferramenta</i> .....	126
5.3.3	<i>Realização da avaliação sob o ponto de vista de usuário</i> .....	127
5.3.4	<i>Realização da avaliação sob o ponto de vista de especialista</i> .....	134
5.4	<b>UM MÉTODO DE AVALIAÇÃO</b> .....	137
5.4.1	<i>Perspectiva de inspeção</i> .....	139
5.4.2	<i>Perspectiva de ferramentas</i> .....	139
5.4.3	<i>Perspectiva de usuário</i> .....	140
5.4.4	<i>Perspectiva de especialista</i> .....	141
5.5	<b>CONSIDERAÇÕES FINAIS</b> .....	142
<b>6 ESTUDO DE VIABILIDADE DO PDWAU USANDO UMA MÉTRICA DE ACESSIBILIDADE E USABILIDADE</b> .....		<b>143</b>
6.1	<b>CONSIDERAÇÕES INICIAIS</b> .....	143
6.2	<b>MEDIÇÕES EXISTENTES POR MEIO DE QUESTIONÁRIOS</b> .....	144
6.3	<b>FORMALIZAÇÃO DA MÉTRICA DE ACESSIBILIDADE E USABILIDADE</b> .....	146

6.4	<i>ESTUDO DE VIABILIDADE: APLICAÇÃO DA MÉTRICA <math>M_{HEUA}</math></i> .....	151
6.5	<i>CONSIDERAÇÕES FINAIS</i> .....	163
<b>7</b>	<b>CONCLUSÕES</b> .....	<b>165</b>
6.6	<i>CONSIDERAÇÕES INICIAIS</i> .....	165
6.7	<i>PRINCIPAIS CONTRIBUIÇÕES</i> .....	166
6.8	<i>LIMITAÇÕES E TRABALHOS FUTUROS</i> .....	168
<b>REFERÊNCIAS BIBLIOGRÁFICAS</b> .....		<b>171</b>
<b><u>APÊNDICES:</u></b>		
<b>TIPOS DE DEFICIÊNCIAS</b> .....		<b>185</b>
<b>REQUISITOS DE USABILIDADE E ACESSIBILIDADE</b> .....		<b>191</b>
<b>TCLE - TERMO DE CONSENTIMENTO LIVRE E ESCLARECIDO</b> .....		<b>197</b>
<b>QUESTIONÁRIO PRÉ-SESSÃO</b> .....		<b>199</b>
<b>ROTEIRO DE TAREFAS</b> .....		<b>203</b>
<b>QUESTIONÁRIO PÓS-SESSÃO</b> .....		<b>207</b>
<b>QUESTIONÁRIO HEUA</b> .....		<b>211</b>
<b><u>ANEXOS:</u></b>		
<b>CRITÉRIOS DE SUCESSO A PARTIR DAS DEMANDAS DE ADULTOS MAIS VELHOS</b> .....		<b>227</b>
<b>PADRÕES WEB DE MONTERO: DESCRIÇÃO (PROBLEMA E SOLUÇÃO)</b> .....		<b>231</b>
<b>DOCUMENTO DE REQUISITOS DO SISTEMA AGENDALOCA</b> .....		<b>235</b>
<b>DESCRIÇÃO DOS CASOS DE USO DO SISTEMA AGENDALOCA</b> .....		<b>243</b>
<b>PARECER SUBSTANCIADO E APROVADO PELO COMITÊ DE ÉTICA EACH-USP</b> .....		<b>255</b>
<b>APLICAÇÃO DE HEUA NO SISTEMA AGENDAMENTO</b> .....		<b>261</b>
<b>APLICAÇÃO DE HEUA NO SISTEMA AGENDALOCA</b> .....		<b>269</b>





# Lista de Figuras

---

Figura 1: Publicações sobre acessibilidade classificadas nas atividades da ISO/IEC 12207. ....	5
Figura 2: Procedimentos metodológicos executados neste trabalho. ....	9
Figura 3: Linguagem de Padrões de Montero <i>et al.</i> (2002). ....	20
Figura 4: <i>Framework</i> genérico de desenvolvimento de sistemas web (Pressman e Lowe, 2009). ....	27
Figura 5: Ciclo de desenvolvimento de um sistema web 2.0. ....	30
Figura 6: <i>Storyboards</i> (esquerda, centro) e o protótipo do design da interface (direita) (Thew <i>et al.</i> , 2009). ....	31
Figura 7: Resumo da abordagem de análise de requisitos ADVISES (Thew <i>et al.</i> , 2009). ....	31
Figura 8: Grafo de requisitos de acessibilidade (Baguma <i>et al.</i> , 2009). ....	33
Figura 9: Caso de uso do módulo de revisão com requisitos de acessibilidade (Baguma <i>et al.</i> , 2009). ....	34
Figura 10: Interface da ferramenta aDesigner (Takagi <i>et al.</i> , 2004). ....	35
Figura 11: Avaliação de um sistema web usando a ferramenta aDesigner (Takagi <i>et al.</i> , 2004). ....	35
Figura 12: <i>Framework</i> AURA (Rubegni <i>et al.</i> , 2008). ....	37
Figura 13: Sistema visual à esquerda e sistema oral à direita (Rubegni <i>et al.</i> , 2008). ...	38
Figura 14: Estruturas das camadas no MLD (Christiernin, 2010). ....	38
Figura 15: Nível de experiência dos usuários por meio do RDPM (Christiernin, 2010). ....	39
Figura 16: Aplicação desenvolvida por meio de design em camadas (Christiernin, 2010). ....	39
Figura 17: Modelo GQM. ....	50
Figura 18: Exemplo de aplicação do GQM. ....	50
Figura 19: Atividades e ações do PDWAU. ....	58
Figura 20: Tarefas da atividade de comunicação do PDWAU. ....	59
Figura 21: Tarefas da atividade de planejamento do PDWAU. ....	62
Figura 22: Tarefas da atividade de modelagem do PDWAU. ....	66

Figura 23: Tarefas da atividade de construção do PDWAU. ....	69
Figura 24: Tarefas da atividade de implantação do PDWAU. ....	72
Figura 25: <i>Workflow</i> de desenvolvimento do sistema AgendAloca. ....	82
Figura 26: Hierarquia de usuários com as tarefas/opções gerais disponíveis no sistema. .....	83
Figura 27: Hierarquia de usuários com as tarefas de usabilidade e acessibilidade disponíveis no sistema. ....	84
Figura 28: Protótipo da página inicial do AgendAloca. ....	85
Figura 29: Página inicial do AgendAloca após <i>login</i> do usuário. ....	85
Figura 30: Projeto de interface do AgendAloca após <i>login</i> do aluno. ....	86
Figura 31: Projeto de interface do AgendAloca após <i>login</i> do professor. ....	86
Figura 32: Projeto de interface dos menus do AgendAloca. ....	87
Figura 33: Projeto de informação do AgendAloca. ....	87
Figura 34: Projeto funcional do AgendAloca. ....	88
Figura 35: Projeto técnico do AgendAloca. ....	88
Figura 36: Exemplo de menu <i>Superfish</i> . ....	90
Figura 37: Teste do AgendAloca com o WAVE. ....	91
Figura 38: Teste do AgendAloca com o WAVE – Função <i>Errors, Features, and Alerts</i> . .....	92
Figura 39: Teste do AgendAloca com o WAVE – Função <i>Structure/Order</i> . ....	93
Figura 40: Teste do AgendAloca com o WAVE – Função <i>Text-only</i> . ....	93
Figura 41: Teste do AgendAloca com o WAVE – Função <i>Outline</i> . ....	94
Figura 42: Teste do AgendAloca com o WAVE – Opção de contraste. ....	94
Figura 43: Teste do AgendAloca com o aDesigner – Tela principal. ....	95
Figura 44: Teste do AgendAloca com o aDesigner – Simulação para cegos I. ....	95
Figura 45: Teste do AgendAloca com o aDesigner – Simulação para cegos II. ....	96
Figura 46: Teste do AgendAloca com o aDesigner – Simulação para cegos III. ....	96
Figura 47: Teste do AgendAloca com o aDesigner – Simulação para cegos IV. ....	97
Figura 48: Teste do AgendAloca com o aDesigner – Simulação para cegos V. ....	98
Figura 49: Teste do AgendAloca com o aDesigner – Simulação para cegos VI. ....	98
Figura 50: Teste do AgendAloca com a tecla TAB. ....	99
Figura 51: Teste das funcionalidades do AgendAloca. ....	100
Figura 52: Alteração do posicionamento dos menus do AgendAloca. ....	100

Figura 53: Visão do AgendAloca por parte do aluno para agendar o trabalho. ....	101
Figura 54: Visão do AgendAloca por parte do aluno após agendar seu trabalho. ....	102
Figura 55: Visão do AgendAloca por parte do aluno. ....	102
Figura 56: Visão do AgendAloca por parte do professor para agendar sua participação em uma ou mais bancas. ....	103
Figura 57: Visão do AgendAloca por parte do professor após agendar sua presença em uma ou mais bancas. ....	103
Figura 58: Padrão de interface baseado no AgendAloca. ....	104
Figura 59: Padrão de interface aplicado no sistema AWMo (Grillo, Fortes e Lucrédio, 2012). ....	105
Figura 60: Perspectivas de acessibilidade no uso do padrão de interface aplicado no sistema AWMo (Fortes <i>et al.</i> , 2012). ....	106
Figura 61: Comparação entre os resultados obtidos nos sistemas Agendamento e AgendAloca após a avaliação sob o ponto de vista de inspeção. ....	126
Figura 62: HTA da tarefa “cadastrar aluno” no Agendamento. ....	129
Figura 63: HTA da tarefa “cadastrar aluno” no AgendAloca. ....	129
Figura 64: Tempo médio dos usuários para executar as tarefas. ....	130
Figura 65: Total de erros dos participantes por tarefa. ....	130
Figura 66: Porcentagem de sucesso por tarefa – Agendamento. ....	131
Figura 67: Porcentagem de sucesso por tarefa – AgendAloca. ....	131
Figura 68: Quantidade media de cliques dos usuários por tarefa. ....	132
Figura 69: Interação dos usuários com os sistemas Agendamento e AgendAloca. ....	132
Figura 70: Visão geral do método ACCESSA. ....	138
Figura 71: Porcentagem de requisitos de usabilidade e acessibilidade que se concatenam. ....	147
Figura 72: Planejamento do desenvolvimento de HEUA. ....	148
Figura 73: Resultados da aplicação de HEUA - Agendamento. ....	154
Figura 74: Resultados da aplicação de HEUA - AgendAloca. ....	155
Figura 75: Quantidade de requisitos atendidos de HEUA – Agendamento e AgendAloca. ....	156
Figura 76: Porcentagem de requisitos atendidos de usabilidade e acessibilidade – Agendamento e AgendAloca. ....	158



# Lista de Tabelas

---

Tabela 1: Fundamentação de princípios teóricos. ....	55
Tabela 2: Quantidade de tarefas e subtarefas adicionadas ao processo genérico.....	74
Tabela 3: Esforço demandado para cada caso de uso de requisitos funcionais.....	77
Tabela 4: Esforço demandado para cada caso de uso de requisitos não funcionais.....	77
Tabela 5: Cronograma de implantação preliminar macroscópico. ....	79
Tabela 6: Cronograma de implantação preliminar microscópico.....	79
Tabela 7: Possíveis riscos no desenvolvimento do AgendAloca. ....	81
Tabela 8: Projeto do experimento.....	119
Tabela 9: Resultado da avaliação sob o ponto de vista de inspeção nos sistemas Agendamento e AgendAloca.....	125
Tabela 10: Resultado da avaliação dos sistemas Agendamento e AgendAloca sob o ponto de vista de ferramenta.....	127
Tabela 11: Dados do teste com usuários. ....	128
Tabela 12: Listas de tarefas executadas em cada sistema. ....	128
Tabela 13: Dados coletados no questionário pós-sessão. ....	133
Tabela 14: Problemas de acessibilidade e usabilidade no Agendamento melhorados no AgendAloca. ....	135
Tabela 15: Resumo de HEUA. ....	146
Tabela 16: Exemplos de descrição de requisitos de HEUA. ....	149
Tabela 17: Exemplo de avaliação pelo especialista.....	151
Tabela 18: Justificativa das opções escolhidas pelo especialista durante a avaliação..	152
Tabela 19: Aplicação de HEUA no Agendamento.....	152
Tabela 20: $Q_{r's}$ encontrados no Agendamento.....	153
Tabela 21: Aplicação de HEUA no AgendAloca.....	155
Tabela 22: Requisitos de HEUA separados em usabilidade e acessibilidade. ....	157
Tabela 23: Resultado da aplicação de HEUA separados em usabilidade e acessibilidade – Agendamento e AgendAloca.....	158
Tabela 24: Requisitos de acessibilidade e usabilidade não atendidos pelo sistema Agendamento e atendidos pelo sistema AgendAloca. ....	161
Tabela 24: Mapeamento do Processo PDWAU. ....	162



# Lista de Acrônimos

---

ACCESSA	<i>Approach to improve the aCCESsibility and uSAbility of existing web system</i>
API	<i>Application Programming Interface</i>
AWA	<i>Acessibility for Web Application</i>
AWMo	<i>Accessible Web Modeler</i>
CMMI	<i>Capability Maturity Model Integration</i>
CMS	<i>Content Management System</i>
CRUD	<i>Create, Read, Update and Delete</i>
CSS	<i>Cascading Style Sheets</i>
CSV	<i>Comma-Separated Values</i>
eMAG	Modelo de Acessibilidade de Governo Eletrônico Brasileiro
ES	Engenharia de Software
EW	Engenharia Web
GCS	<i>Group Calendar Systems</i>
GQM	<i>Goal-Question-Metric</i>
HEUA	<i>Heuristic Evaluation with Usability and Accessibility requirements to assess web systems</i>
HTA	<i>Hierarchical Task Analysis</i>
HTML	<i>HyperText Markup Language</i>
HTTP	<i>HyperText Transfer Protocol</i>
IBM	<i>International Business Machines Corporation</i>
IC	Iniciação Científica
ICMC	Instituto de Ciências Matemáticas e de Computação
ID	Identificador/Identidade
IEC	<i>International Electrotechnical Commission</i>
IEEE	<i>Institute of Electrical and Electronics Engineers</i>
IHC	Interação Humano Computador
ISO	<i>International Standard Organization</i>
JSF	<i>Java Server Faces</i>
MLD	<i>Multi-Layered Design</i>

MTA	Modelo de Tarefas de Acessibilidade
NVDA	<i>NonVisual Desktop Access</i>
OMG	<i>Object Management Group</i>
OOHDM	<i>Object-Oriented Hypermedia Design Model</i>
OOWS	<i>Object-Oriented Web Solutions</i>
PDA	<i>Personal Digital Assistant</i>
PDWAU	<i>Process for Developing Web system with Accessibility and Usability</i>
QUIS	<i>Questionnaire for User Interface Satisfaction</i>
RDPM	<i>Radar Diagram Process for Multiple layers</i>
SBC	Sociedade Brasileira de Computação
SLR	<i>Systematic Literature Review</i>
SUMI	<i>Software Usability Measurement Inventory</i>
TCC	Trabalho de Conclusão de Curso
TCLE	Termo de Consentimento Livre e Esclarecido
TI	Tecnologia da Informação
TICs	Tecnologia da Informação e Comunicação
UCD	<i>User Centered Design</i>
URLs	<i>Uniform Resource Locator</i>
W3C	<i>World Wide Web Consortium</i>
WAI	<i>Web Accessibility Initiative</i>
WAMMI	<i>Website Analysis and MeasureMent Inventory</i>
WCAG	<i>Web Content Accessibility Guidelines</i>
webML	<i>Web Modeling Language</i>
WWW	<i>World Wide Web</i>
XML	<i>Extensible Markup Language</i>
XSL	<i>Extensible Stylesheet Language</i>



---

# *Introdução*

---

## *1.1. Contextualização*

A Tecnologia da Informação (TI), assim como a *World Wide Web* (WWW), tem se incorporado no cotidiano de trabalho, negócios e atividades sociais, tornando-se progressivamente indispensável para empresas e para a vida pessoal (Natukunda, 2008). Embora os usuários da web, incluindo as pessoas com deficiência, possam realizar convenientemente certas tarefas que de outra maneira seriam difíceis ou impossíveis, muitos tipos de sistemas web como *e-learning*, *e-commerce* e *e-government*, ainda não são acessíveis a todos os usuários (Baguma e Lubega, 2008).

No contexto desta tese, um sistema web é um sistema de software executado no ambiente web, envolvendo aspectos de hipermídia (hipertextos e multimídia, por exemplo) que são combinados com a lógica de sistemas convencionais (entrada de dados e sistemas de informação com transações de negócios).

De acordo com o *World Wide Web Consortium* (W3C, 1999a), “acessibilidade na web” corresponde à possibilidade de todas as pessoas acessarem a web em diferentes situações. Essas situações envolvem tanto requisitos tecnológicos necessários para a interação, quanto características do usuário, tais como suas habilidades, necessidades e limitações motoras e cognitivas distintas.

A WWW foi concebida com o principal intuito de prover uma tecnologia para disponibilização de conteúdo em um formato padronizado simples e poderoso, por meio de informações no formato de hipertexto utilizando HTML (*Hypertext Markup Language*) (W3C, 1999b). Desde a concepção inicial da web, Tim Berners Lee (inventor da web e diretor do W3C) declarou que:

“O poder da web está em sua universalidade. Ser acessada por todos, independentemente de deficiências, é um fator essencial”.

Leis que regulamentam a disponibilização de conteúdo acessível por parte de órgãos governamentais em diversos países têm aumentado (W3C, 1999a, 2008a; eMAG, 2005). Tais leis devem ser obedecidas não só pelos desenvolvedores de software, mas também por todos os profissionais da área de TI, reforçando a conscientização da importância da acessibilidade nos sistemas web.

Corroborando com a importância deste tema, a Sociedade Brasileira de Computação (SBC) propôs cinco grandes desafios para a computação brasileira (2006-2016), dos quais dois exploram a questão da diversidade e da acessibilidade, que são: I) modelagem computacional de sistemas complexos artificiais, naturais e socioculturais e da interação homem-natureza; e II) acesso participativo e universal do cidadão brasileiro ao conhecimento.

A acessibilidade a sistemas web é uma necessidade crescente e a observância dos aspectos necessários para implementá-la é vital para o desenvolvimento de sistemas web, tanto do ponto de vista tecnológico, quanto dos pontos de vista social, político e até mesmo legal.

## **1.2. Motivação**

Atualmente, existem várias plataformas de desenvolvimento que procuram apoiar os desenvolvedores durante o desenvolvimento de sistemas web, facilitando a criação de bancos de dados e apoiando a separação entre controle e apresentação dos dados. No entanto, essas plataformas não têm explorado o uso efetivo da acessibilidade. Dessa maneira, um grande desafio de pesquisa é identificar características de acessibilidade e usabilidade em soluções de interface e de interação para prover mecanismos, processos e estratégias de apoio que facilitem o desenvolvimento de sistemas web acessíveis e

usáveis. Além disso, a dificuldade em integrar os termos acessibilidade e usabilidade precisa ser considerada nesse contexto.

Os sistemas web possuem características de um sistema convencional e, portanto, requerem o uso de processos e métodos sólidos e confiáveis para seu desenvolvimento. Esses sistemas também possuem características que os tornam diferentes de outros sistemas, tais como requisitos de navegação, uso de imagens, textos, sons e animações, usabilidade elaborada e acesso disponível a vários tipos de usuários, e exigem que os métodos disponíveis para o seu desenvolvimento atendam a tais características (Bianchini, 2008).

Considerando as diretrizes de acessibilidade disponíveis atualmente, os desenvolvedores podem implementar sistemas web que são acessíveis à maioria dos usuários. No entanto, eles muitas vezes não conseguem implementar essas diretrizes efetivamente (Asakawa, 2005). Uma das razões para este problema é que a maioria das diretrizes de acessibilidade tem difícil integração com os fluxos de trabalho de desenvolvimento existentes e raramente oferecem sugestões específicas que são orientadas ao desenvolvedor. Bigham e Ladner (2007) afirmam também que os desenvolvedores web estariam mais aptos a desenvolver sistemas web acessíveis se lhes fossem dadas sugestões específicas e suporte efetivo sobre como fazer isso.

Além da acessibilidade, a usabilidade é um fator crucial no desenvolvimento de sistemas web, pois visa aumentar a satisfação das pessoas durante a interação com os computadores, gerando eficiência e conforto para o usuário. Adicionalmente, aumentar a vantagem competitiva e o valor percebido do sistema, bem como minimizar custos posteriores são razões que justificam o investimento em interfaces de sistemas web (Shneiderman, 1993).

O desenvolvimento de pesquisas com foco em acessibilidade e usabilidade, como a apresentada nesta tese, é importante porque falta evidência de que os atuais processos de Engenharia Web (EW) apoiam a incorporação desses conceitos nos sistemas. São conceitos complexos que envolvem várias competências, mas que são complementares e precisam ser abordados no desenvolvimento de sistemas.

### ***1.3. Questão de pesquisa e definição do escopo***

Para nortear os rumos da pesquisa apresentada nesta tese foi elaborada uma pergunta que serviu como ponto de partida:

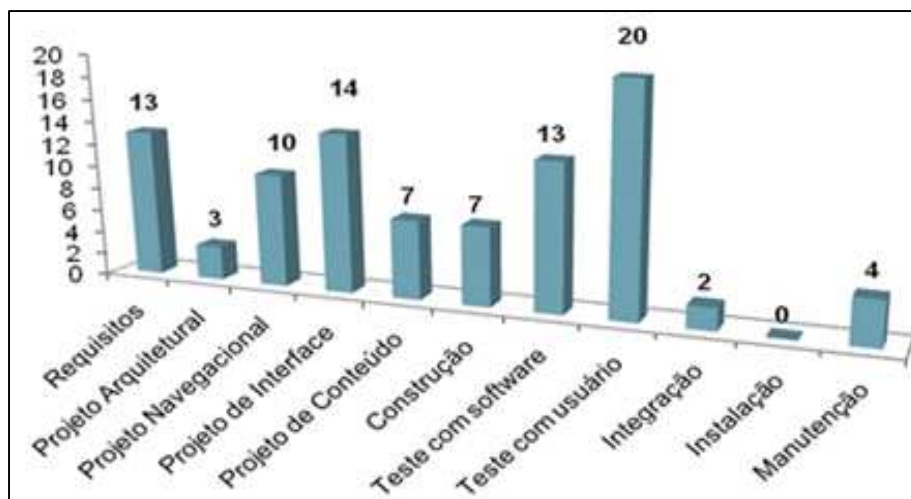
– Existem elementos apropriados para fornecer suporte às ações inerentes ao projeto e à implementação de sistemas web, que podem ser selecionados, combinados e tecnicamente implementados para produzir um sistema web acessível?

Nessa pergunta, usou-se apenas o termo “acessível”, pois era uma questão herdada da pesquisa realizada por Freire, Goularte e Fortes (2007), que foi continuada devido à sua importância e aos resultados encontrados. Após a leitura dos artigos selecionados e encontrados na literatura, percebeu-se a importância de incluir a preocupação quanto à usabilidade no desenvolvimento desta pesquisa. Usabilidade e acessibilidade são conceitos fortemente relacionados que não implicam em limitações entre si (Seção 2.4).

Para responder à pergunta mencionada, foi realizada inicialmente, uma revisão sistemática (SLR – *Systematic Literature Review*), seguindo o roteiro proposto por Kitchenham (2004), para extrair informações relacionadas à acessibilidade no desenvolvimento de sistemas web. Atualmente, o procedimento de SLR tem sido chamado de mapeamento sistemático, após uma revisão dos conceitos realizada por Kitchenham e Charters (2007).

Todos os passos envolvidos durante a realização da SLR foram documentados, com o objetivo de garantir a validade dos resultados obtidos a partir dos dados analisados. O foco dessa SLR foi pesquisar por evidências de uso de métodos, técnicas e abordagens para o tratamento da questão da acessibilidade na web, sob o ponto de vista da engenharia web (ver Capítulo 3).

Além das referências obtidas da SLR realizada, outras referências foram utilizadas nessa pesquisa. Essas novas referências foram obtidas por indicações de especialistas das áreas de engenharia web e acessibilidade e também de novas buscas nas bases de dados utilizadas anteriormente durante a realização da SLR. Os artigos foram classificados como técnica de apoio a uma ou mais atividades da norma ISO/IEC 12207 (1998), conforme apresentado na Figura 1.



**Figura 1: Publicações sobre acessibilidade classificadas nas atividades da ISO/IEC 12207.**

Com a realização dessa pesquisa, obteve-se um panorama sobre as técnicas existentes, classificadas de acordo com as atividades do processo de engenharia web. De fato, existem muitas diretrizes de acessibilidade propostas na literatura, tornando difícil encontrar, selecionar e lidar com todas elas durante a atividade de desenvolvimento.

Uma preocupação em apoiar usuários que são criadores de conteúdo na web é evidente, já que os usuários, que por sua vez passam a ser também desenvolvedores, precisam de instruções para deixar seus conteúdos disponíveis a todos (Arrue, Vigo e Abascal, 2007).

Em um artigo recente, Lazar e Olalere (2011) definiram sugestões gerais encontradas na literatura sobre acessibilidade e usabilidade na web. São elas:

- 1) É necessário inserir a acessibilidade no início do processo de desenvolvimento;
- 2) Desenvolvimento de conteúdo deve ser guiado pelo conhecimento do comportamento dos usuários da web, ou seja, guiado pela inteligência coletiva;
- 3) É necessária uma abordagem de ciclo de vida que integre processos de engenharia de usabilidade e acessibilidade com engenharia de software;
- 4) São necessárias mais pesquisas sobre formas de adaptar interfaces para que usuários com deficiências possam usá-las com sucesso;
- 5) Requisitos para projetar e testar a acessibilidade devem ser claramente indicados nos artefatos do sistema, os quais devem ser avaliados em termos de sua compreensão e disposição para alcançar acessibilidade; e

- 6) Órgãos governamentais deveriam exigir testes de usuários com participantes que têm deficiências visuais, cognitivas, auditivas e motoras, bem como grupos de deficiências.

Considerando-se as sugestões gerais encontradas na literatura sobre acessibilidade e usabilidade na web, nota-se que ainda existe uma lacuna de pesquisas e propostas, discussões, resultados e validações sobre como compreender, projetar e avaliar sistemas web que sejam acessíveis e usáveis, ou seja, que considerem as necessidades dos usuários. Dessa maneira, o objetivo desta tese foi definido.

#### **1.4. Objetivo**

O objetivo desta pesquisa é a definição de um processo para o desenvolvimento de sistemas web com foco em acessibilidade e usabilidade. Para a formalização deste processo, elementos existentes na literatura foram selecionados e combinados, de maneira que pudessem ser repetíveis durante as iterações do processo de desenvolvimento de software, a fim de oferecer ao desenvolvedor um apoio efetivo para a criação de sistemas web com alto grau de acessibilidade e usabilidade.

Processos abrangentes e integrados com esse objetivo não foram evidenciados na revisão sistemática realizada. A efetividade desse processo definido deve ser avaliada de diferentes formas e pontos de vista seguindo processos metodológicos rigorosos.

Entendem-se como sistemas web acessíveis e usáveis aqueles sistemas que podem ser usados por seus usuários típicos, incluindo pessoas com deficiência que utilizam recursos de tecnologia assistiva, tais como leitores de tela, navegadores de voz e monitores Braille (Lazar, Dudley-Sponaugle e Greenidge, 2004).

Além disso, o processo formalizado deve apoiar os desenvolvedores, refletindo consequentemente na satisfação e possibilidade de uso efetivo de sistemas web por todos os usuários, com ou sem deficiências, contribuindo assim para a inclusão digital desses usuários.

#### **1.5. Procedimentos metodológicos**

Para que as ações que fazem parte da engenharia web possam ser cumpridas, elas devem ser executadas de acordo com um processo ou método de desenvolvimento. Dessa maneira, estudos sobre processos de desenvolvimento de sistemas web, tal como o

processo de Pressman e Lowe (2009) foram realizados, além de estudos sobre propostas, discussões, resultados e validações sobre como compreender, projetar e avaliar sistemas web que sejam acessíveis e usáveis.

Assim, iniciou-se o desenvolvimento de um processo com a definição de tarefas e subtarefas de acessibilidade e usabilidade, desde a atividade inicial de comunicação, até a atividade de implantação (Dias, Fortes e Masiero, 2012). Para isso, utilizou-se o processo genérico de engenharia web (Pressman e Lowe, 2009) porque além de conter atividades básicas, fundamentais e bem definidas, ele herda características da engenharia de software e engloba a maior parte das atividades que outros processos e métodos consideram como fundamentais. Trata-se ainda de um dos poucos, senão o único, processo abrangente de desenvolvimento de sistemas web bem documentado e público.

Após a formalização do processo com base principalmente em um referencial teórico, um estudo de caso foi realizado para verificar se o processo era exequível e efetivo. Para isso, desenvolveu-se um sistema de agendamento de bancas, com o objetivo principal de auxiliar o agendamento de bancas referentes aos trabalhos de conclusão de curso desenvolvidos no ICMC-USP (Instituto de Ciências Matemáticas e de Computação da Universidade de São Paulo). Os usuários deste sistema são alunos (podem acessar o sistema para agendar sua data e horário de apresentação) e professores (podem acessar o sistema para escolher em quais bancas participarão como avaliadores).

Vale ressaltar que antes do desenvolvimento do estudo de caso, já havia um sistema de agendamento de bancas com o mesmo propósito e as mesmas funcionalidades (Santos e Fortes, 2010). No entanto, ele havia sido desenvolvido com um processo *ad-hoc* e sem preocupações quanto aos requisitos de acessibilidade e usabilidade.

Após o desenvolvimento do novo sistema de agendamento de bancas, foi realizado um experimento (ver Capítulo 5) para avaliar os dois sistemas existentes: o sistema desenvolvido com o processo formalizado e o sistema já existente desenvolvido de maneira *ad-hoc*. O resultado obtido pela avaliação serviria para indicar indícios de que além de exequível, o processo pode ser efetivo para apoiar o desenvolvimento de sistemas web com acessibilidade e usabilidade.

Portanto um experimento controlado, desenvolvido em ambiente acadêmico, foi realizado. Com os resultados obtidos após a realização desse experimento, foi possível confirmar a suposição dos autores sobre a efetividade do processo desenvolvido e também aperfeiçoar a primeira versão do processo. Além disso, definiu-se um método que pode

ser útil para avaliar, comparar e melhorar a acessibilidade e a usabilidade de sistemas web existentes, agindo principalmente no projeto de interface, sem mudar os requisitos funcionais dos sistemas (Dias *et al.*, 2013).

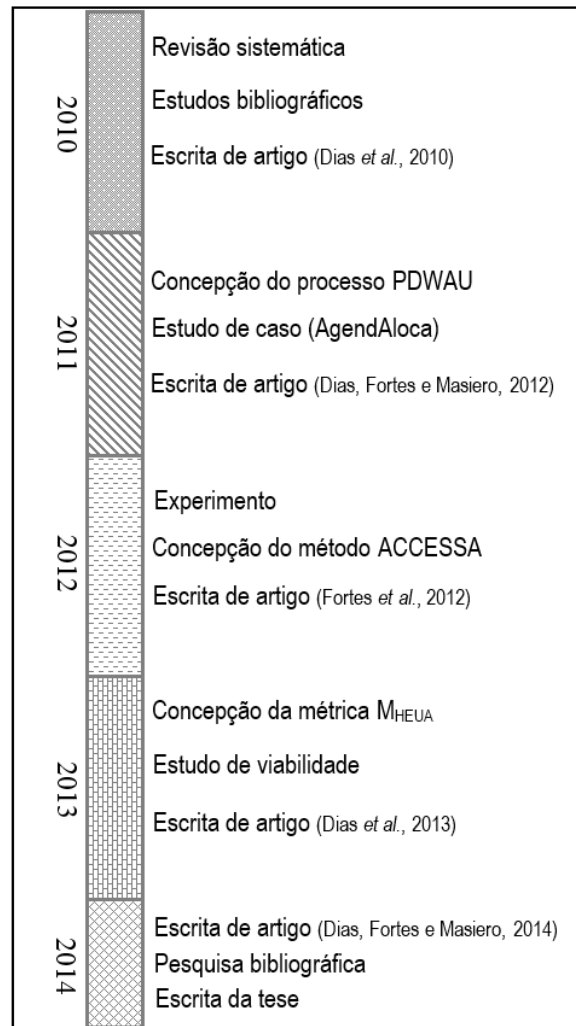
Além do experimento realizado, percebeu-se a ausência de um instrumento que permitisse uma avaliação mais objetiva e quantificável dos produtos de software resultantes do processo proposto, que evidenciasse as características de acessibilidade e usabilidade tratadas. A ideia era identificar alguma forma de se obter valores para quantificar respostas aos requisitos de usabilidade e acessibilidade verificados nos sistemas web existentes a fim de compará-los. Assim, foi definida uma métrica que pudesse mostrar essa medida e então, por meio de um estudo de viabilidade, reforçar os indícios sobre a efetividade do processo formalizado, apresentados nos resultados do experimento realizado (Dias, Fortes e Masiero, 2014).

Com as avaliações realizadas nos dois sistemas de agendamento de bancas, por meio do experimento e do estudo de viabilidade, indiretamente avaliou-se o processo formalizado para sistemas web com foco em acessibilidade e usabilidade. Como a métrica foi aplicada ao sistema (que é um produto do processo proposto), ela foi um mecanismo de avaliação indireta do processo.

Além disso, a métrica foi relevante por identificar as características específicas que foram destacadas no experimento, bem como mostrar uma medida objetiva e direta das características de acessibilidade e usabilidade de modo geral. Percebeu-se então, a contribuição do método para avaliar quantitativamente, comparar e melhorar a acessibilidade e a usabilidade de sistemas web existentes. Assim, o método e a métrica podem ser utilizados independentemente, tanto para melhorar um sistema web como para comparar sistemas existentes.

Na Figura 2 é apresentada a sequência de realização dos procedimentos metodológicos de pesquisa mencionados nesta tese, em ordem cronológica.





**Figura 2: Procedimentos metodológicos executados neste trabalho.**

Ressalta-se que os procedimentos metodológicos foram desenvolvidos e são explicados em mais detalhes, na medida em que os capítulos são apresentados.

## 1.6. Organização

Esta tese está organizada da seguinte maneira: no **Capítulo 2**, uma síntese dos conceitos de usabilidade e acessibilidade é apresentada. Uma introdução aos conceitos de engenharia de sistemas web, além de outros conceitos, é apresentada no **Capítulo 3**, com o objetivo de fundamentar as pesquisas desenvolvidas e apresentadas nos próximos capítulos, bem como apresentar uma revisão da literatura relativa aos trabalhos mais relevantes para a tese.

O processo formalizado a partir da combinação de elementos existentes na literatura, bem como sua aplicação em um estudo de caso no desenvolvimento de um

sistema web é apresentado no **Capítulo 4**. Além disso, apresenta-se um padrão de interface que foi percebido e utilizado a partir do sistema desenvolvido no estudo de caso.

No **Capítulo 5**, o experimento realizado para avaliar o processo formalizado, comparando-se o sistema desenvolvido com o apoio do processo com o sistema já existente desenvolvido de maneira *ad-hoc* é apresentado. Adicionalmente, apresenta-se o método percebido com a realização do experimento, que pode ser útil para avaliar, comparar e melhorar a acessibilidade e a usabilidade de sistemas web existentes.

A fim de possibilitar uma avaliação objetiva por meio de quantificadores, formalizou-se uma métrica, que avalia requisitos de usabilidade e acessibilidade em um sistema web. Essa métrica, apresentada no **Capítulo 6**, foi aplicada nos sistemas de agendamento de bancas a fim de obter uma comparação quantificável de acessibilidade e usabilidade que cada sistema web apresenta.

Por fim, no **Capítulo 7**, é possível encontrar as conclusões desta tese. As principais contribuições são relacionadas, bem como as limitações e os trabalhos futuros.

---

# *Acessibilidade e usabilidade*

---

## *2.1 Considerações iniciais*

Neste capítulo são apresentados de maneira sucinta os conceitos de usabilidade e acessibilidade. O capítulo está dividido da seguinte maneira: Diretrizes de acessibilidade e tipos de deficiências são comentadas na Seção 2.2. Na Seção 2.3, conceitos iniciais sobre usabilidade são definidos e discutidos, bem como heurísticas e padrões web. Uma comparação entre acessibilidade e usabilidade é realizada na Seção 2.4. E por fim, na Seção 2.5 são apresentadas as considerações finais do capítulo.

## *2.2 Acessibilidade*

De acordo com a ISO 9241-171 (2008), a acessibilidade pode ser definida como "a usabilidade de um produto, serviço, ambiente ou facilidade de uso por meio de pessoas com a mais ampla gama de capacidades". De acordo com Freire (2012), esta definição parece ser uma extensão da definição de usabilidade, definida pela ISO 9241-11 (1998) como a "medida na qual um produto pode ser usado por usuários específicos para alcançar objetivos específicos com eficácia, eficiência e satisfação em um contexto específico de uso".

Segundo Petrie e Kheir (2007), a definição de acessibilidade deve ter como base o usuário. Para esta finalidade foi realizada uma adaptação na definição da ISO 9241-171

(2008), resultando na seguinte definição de acessibilidade: “O grau em que um produto/site pode ser usado por usuários específicos com deficiências específicas para alcançar objetivos específicos com efetividade, eficiência e satisfação em um contexto específico de uso”.

Acessibilidade é o termo geral usado para indicar a possibilidade de qualquer pessoa usufruir todos os benefícios da vida em sociedade, como também o uso da Internet. Quando se consideram as questões de acessibilidade na web, uma das referências de maior importância é o WCAG (*Web Content Accessibility Guidelines*), elaborada pela WAI (*Web Accessibility Initiative*) (Freire, Goularte e Fortes, 2007). A WAI é uma organização criada pelo W3C que tem como missão definir princípios e regras de interface e desenvolvimento de sistemas que sejam acessíveis a pessoas com necessidades especiais (Lucca *et al.*, 2005).

Além disso, em dezembro de 2004 foi aprovado no Brasil o Decreto n. 5.296, regulamentando leis anteriores e estabelecendo prazo para tornar acessíveis todos os portais da administração pública, de interesse público ou financiado pelo governo. Como resultado, desenvolveu-se o Modelo de Acessibilidade Brasileiro (eMAG), elaborado pelo Departamento de Governo Eletrônico, fundamentado na análise das normas de acessibilidade de vários países e nas diretrizes propostas pelo W3C, com o propósito de facilitar e padronizar a acessibilidade dos sistemas web (eMAG, 2005).

### **2.2.1 Diretrizes de acessibilidade**

No documento do WCAG 1.0 (W3C, 1999a) é definido um conjunto de quatorze diretrizes que tratam diversos problemas relacionados à acessibilidade em sistemas web. Cada diretriz possui pontos de verificação, que tem um nível de prioridade baseado no seu impacto na acessibilidade (exemplo: o nível de prioridade 1 está relacionado a pontos que o desenvolvedor deve obrigatoriamente atender. Caso contrário, um ou mais grupos de usuários ficarão impossibilitados de acessar informações contidas nos documentos. Atender a esses pontos é requisito básico para que certos grupos acessem documentos disponíveis na web).

Já o conceito de nível de conformidade varia de acordo com a satisfação das diretrizes de acessibilidade (exemplo: o nível de conformidade ‘A’ indica que todas as recomendações de prioridade 1 foram atendidas). No documento de recomendações do WCAG versão 2.0 (W3C, 2008b), as diretrizes são organizadas em torno de quatro

princípios, de forma que a maioria das recomendações presentes no WCAG 1.0 são também atendidas e acrescentadas novas recomendações.

Cada diretriz apresenta um determinado número de critérios de sucesso, que descrevem especificamente o que deve ser alcançado. Cada critério é classificado em níveis de conformidade entre A, AA e AAA, em que o nível A representa os critérios de maior importância. Essa classificação de níveis foi determinada de acordo com os seguintes fatores:

- 1) Possibilidade da tecnologia assistiva tornar o conteúdo acessível;
- 2) Aplicabilidade a todos os sistemas e tipos de conteúdo;
- 3) Facilidade de compreensão por autores de conteúdo;
- 4) Limitações na apresentação, funcionalidade, liberdade de expressão e estética; e
- 5) Existência de outros meios para solucionar o problema.

No WCAG 2.0, como citado anteriormente, as diretrizes são divididas em quatro princípios que servem como características necessárias ao conteúdo web, sendo eles:

**Princípio 1.** O conteúdo deve ser perceptível: usuários devem ser capazes de perceber a informação sendo apresentada, ou seja, o conteúdo não pode ser invisível a todos os seus sentidos.

- 1.1 Alternativas textuais: disponibilizar alternativas textuais para conteúdos não textuais;
- 1.2 Mídias temporais: disponibilizar alternativas para mídias temporais;
- 1.3 Adaptabilidade: criar conteúdo que possa ser disponibilizado de diferentes maneiras sem perder a informação ou estrutura;
- 1.4 Distinguíveis: facilitar aos usuários ver e ouvir o conteúdo, apresentando foco ao conteúdo principal sendo disponibilizado.

**Princípio 2.** Os componentes de interface com o usuário devem ser operáveis: a interface não pode exigir interações com as quais um usuário não possa realizar.

- 2.1 Acessível pelo teclado: tornar todas as funcionalidades acessíveis pelo teclado;
- 2.2 Tempo suficiente: disponibilizar tempo suficiente de leitura e utilização do conteúdo;

**2.3** Apreensibilidade: não estruturar o sistema com conteúdos que possam causar apreensão nos usuários, como por exemplo: flash de frequência superior a três vezes por segundo;

**2.4** Navegabilidade: disponibilizar meios que auxiliem a navegação do usuário, busca por conteúdos e localização.

**Princípio 3.** O conteúdo e os controles do usuário devem ser fáceis de entender: o conteúdo e as operações não podem ir além do conhecimento do usuário.

**3.1** Legível e compreensível: disponibilizar o conteúdo de forma legível e compreensível aos usuários;

**3.2** Previsibilidade: os sistemas devem aparecer e operar por meios previsíveis;

**3.3** Assistência de entrada: auxiliar os usuários a evitar e corrigir erros.

**Princípio 4.** O conteúdo deve ser suficientemente robusto: os usuários devem ser capazes de acessar o conteúdo com as tecnologias atuais e futuras.

**4.1** Compatibilidade: maximizar a compatibilidade com agentes de usuário e tecnologias disponibilizados atualmente ou no futuro.

Ressalta-se que para atender às diretrizes do WCAG 2.0, é necessário cumprir todos os critérios de sucesso pertencentes a cada uma delas. De acordo com Mbipom e Harper (2009), a evolução do WCAG para a versão 2.0 resultou na necessidade de alteração significativa nas ferramentas de avaliação e de apoio na reparação de problemas de acessibilidade. Os autores afirmam, ainda, que existem diretrizes cuja avaliação automática é difícil ou inviável, como por exemplo:

- Oferecimento de símbolos gráficos na língua de sinais para a totalidade do áudio pré-gravado existente (critério de sucesso 1.2.6); e
- Presença de jargões e palavras não usuais (critério de sucesso 3.1.3).

Um recente reconhecimento do WCAG 2.0, ocorrido em outubro de 2012, foi a sua normatização como sendo um padrão ISO/IEC de identificador 40500:2012 (ISO 40500, 2012) pelo Comitê Técnico Conjunto JTC 1 (Tecnologia da Informação).

Além dos critérios de sucesso definidos no WCAG 2.0, Lara (2012) propôs a inserção de alguns critérios de sucesso a partir das demandas dos adultos mais velhos, que também devem ser levadas em consideração. Eles estão organizados de acordo com os princípios do WCAG 2.0 (ver ANEXO I).

Além dos critérios de sucesso definidos pelo WCAG 2.0, os critérios de sucesso definidos por Lara (2012) também são utilizados no trabalho prático desta tese, pois tratam de questões importantes relativas aos idosos e que também devem ser aplicados a outros públicos-alvo.

Ressalta-se que os sistemas web devem ser implementados para que possam ser apresentados e visualizados, independentemente de dispositivos, sistema operacional e navegadores utilizados. Por outro lado, nas situações relacionadas com as características do usuário é priorizada a adaptação, redundância e substituição do conteúdo de acordo com as deficiências físicas ou cognitivas que os usuários possam apresentar.

### 2.2.2 Desafios encontrados de acordo com os tipos de deficiências

Apesar de haver uma grande variedade de usuários, cada qual com suas peculiaridades; eles podem ser divididos em quatro grupos a partir de suas deficiências (Sun e Zhang, 2009):

- 1) **Deficiência visual:** alguns dos desafios deste tipo de usuário podem ser citados: (I) os usuários não podem ver/perceber as informações visuais (ex: imagens, leiaute ou informações coloridas); (II) eles normalmente precisam de um teclado para navegar no conteúdo disponível no sistema web; (III) eles não podem compreender o conteúdo que é apresentado em ordem lógica linear ou que contenha textos dispensáveis (ex: longos endereços da web, palavra por palavra, etc.); e (IV) eles são ajudados por técnicas que nem sempre são capazes de acessar um vasto leque de tecnologias, especialmente se essas tecnologias são novas.
- 2) **Deficiência auditiva:** os usuários de sistemas computacionais deste perfil sentem dificuldades em acessar (perceber/ouvir) determinadas informações que se encontram disponíveis por meio de dispositivos de áudio. Para fazer conteúdo de áudio da web que seja acessível aos surdos, alguns desenvolvedores pensaram que o melhor meio seria fazer uma versão da linguagem gestual para o conteúdo de áudio. Mas existem alguns problemas com esta opção: (I) nem todas as pessoas surdas utilizam as línguas de sinais; (II) a imagem do vídeo na web muitas vezes não é suficientemente grande ou com imagens claras para tornar a linguagem escrita compreensível; (III) nem todo mundo fala a mesma língua de sinais. Dessa maneira, a forma de tornar acessíveis os conteúdos de áudio para os usuários com deficiência auditiva é

fornecer legendas ou transcrições, de acordo com a língua que esses usuários utilizam para se comunicar.

- 3) **Deficiência física:** os usuários deste perfil sentem várias dificuldades na utilização do mouse e também do teclado comum. Além disso, podem ter dificuldades em posicionar o mouse em pequenos elementos ou *links* devido à falta de coordenação. A maioria dos recursos de tecnologia assistiva para pessoas com deficiência motora usa o teclado para emular as funcionalidades do teclado comum. Desta maneira, é necessário garantir que os conteúdos estejam acessíveis por meio do teclado e assegurar que o sistema seja navegável com o uso de poucas teclas.
- 4) **Deficiência mental:** alguns conteúdos podem ser muito complexos para determinados públicos-alvo, isso é inevitável. No entanto, ainda existem algumas soluções que podem ser feitas pelos desenvolvedores para aumentar a acessibilidade ao conteúdo da web às pessoas com deficiências cognitivas menos graves. Os usuários não podem compreender o conteúdo que é apresentado da mesma maneira para os usuários sem deficiências. Eles precisam de ajuda para reter o aprendizado, para que não usem o sistema erroneamente etc. Em muitos casos, a maneira de tornar o conteúdo web mais acessível às pessoas com deficiências cognitivas é por meio de técnicas para uma comunicação eficaz.
- 5) Adicionalmente, conceitua-se como **deficiência múltipla** a associação de duas ou mais deficiências (Bailey e Pearson, 2011). Neste caso, a interação do usuário com o sistema poderá ser feita de várias maneiras, como as citadas anteriormente em cada tipo de deficiência. Ressalta-se que as dificuldades específicas de aprendizado como a dislexia, por exemplo, não serão tratadas nesta tese.

Para o desenvolvimento prático desta tese, os critérios de sucesso de acessibilidade são utilizados. Além disso, as heurísticas e os padrões web, apresentados na próxima seção, também devem ser considerados durante o desenvolvimento de sistemas web.



## 2.3 Usabilidade

A ISO (*Internacional Standard Organization*), dispõe de duas definições de usabilidade:

- 1) “A usabilidade refere-se à capacidade de um software de ser compreendido, aprendido, utilizado e ser atrativo para o usuário, em condições específicas de utilização” (ISO/IEC 9126, 2001). Esta definição faz ênfase aos atributos internos e externos do produto, os quais contribuem à sua usabilidade, funcionalidade e eficiência.
- 2) “Usabilidade é a capacidade de um produto ser usado por usuários específicos para atingir objetivos específicos com eficácia, eficiência e satisfação em um contexto específico de uso” (ISO 9241-11, 2010). Essa definição é centrada no conceito de qualidade de utilização, isto é, refere-se em como o usuário realiza tarefas específicas em cenários específicos com efetividade.

Segundo Guedes (2008), o termo usabilidade se origina na ciência cognitiva e significa ser fácil de usar. Emprega-se o termo usabilidade para descrever a qualidade da interação de uma interface com os usuários. Em outras palavras, a usabilidade é a capacidade de um produto ser usado por usuários para atingir objetivos específicos com eficácia, eficiência e satisfação (ISO 9241-210, 2010).

De acordo com Shneiderman (1993), uma interface só é bem sucedida se ela der o suporte adequado aos objetivos e ao comportamento do usuário real. Portanto, conhecer o usuário é fundamental, buscando saber o que o usuário pensa, o que quer e como age, aplicando técnicas de pesquisa como grupos de foco ou testes de usabilidade. As recomendações de usabilidade são apresentadas por meio de princípios e pra que elas sejam cumpridas, as heurísticas são encontradas na literatura em geral como diretrizes práticas.

### 2.3.1 Heurísticas de usabilidade

Para guiar o desenvolvimento de sistemas visando usabilidade, Nielsen (1993) definiu dez heurísticas que tratam os principais pontos que devem ser verificados durante o desenvolvimento de um sistema de software:

- 1) **Visibilidade do status do sistema:** o sistema precisa manter os usuários informados sobre o que está acontecendo, fornecendo *feedback* adequado em um tempo razoável;

- 2) **Compatibilidade do sistema com o mundo real:** o sistema precisa ter palavras, frases e conceitos que sejam familiares ao usuário, ao invés de termos orientados ao sistema. O sistema deve seguir convenções do mundo real, fazendo com que a informação apareça numa ordem natural e lógica;
- 3) **Controle do usuário e liberdade:** usuários frequentemente escolhem funções do sistema por engano e precisam ter claras as saídas de emergência, para que possa sair do estado indesejado sem ter que percorrer um extenso percurso. Para tanto é necessário, por exemplo, fornecer funções de *undo* (desfaz) e *redo* (faz de novo);
- 4) **Consistência e padrões:** usuários não precisam adivinhar que diferentes palavras, situações ou ações significam a mesma coisa. Portanto, é necessário seguir convenções da plataforma computacional;
- 5) **Prevenção de erros:** melhor do que uma boa mensagem de erro é um design cuidadoso, que previne o erro antes dele acontecer;
- 6) **Reconhecimento ao invés de lembrança:** tornar objetos, ações e opções visíveis. O usuário não deve ter que lembrar informações de uma parte para outra de um diálogo; ao contrário, as instruções para uso do sistema devem estar visíveis e facilmente recuperáveis quando necessário;
- 7) **Flexibilidade e eficiência de uso:** usuários novatos se tornam experientes com o uso. Prover aceleradores de forma a aumentar a velocidade de interação, assim como permitir que usuários experientes possam “cortar caminho” em ações frequentes é um aspecto importante para a usabilidade;
- 8) **Estética e Design Minimalista:** diálogos não devem conter informação irrelevante ou raramente necessária. Qualquer unidade de informação extra no diálogo irá competir com unidades relevantes de informação e diminuir sua visibilidade relativa;
- 9) **Ajudar os usuários a reconhecer, diagnosticar e corrigir erros:** mensagens de erro devem ser expressas em linguagem clara (sem códigos) indicando precisamente o problema e sugerindo uma solução construtivamente; e
- 10) **Help e Documentação:** embora o ideal seja um sistema que possa ser usado sem auxílio nenhum, é necessário ter a opção de *help* e documentação. Essas informações devem ser de fácil localização, focadas nas tarefas dos usuários e não muito extensas.

Segundo a norma ISO 9241-210 (2010), são cinco os princípios de usabilidade que contribuem para uma boa interface:

- 1) Facilidade de aprendizagem;
- 2) Eficiência de uso;
- 3) Facilidade de memorização;
- 4) Baixas taxas de erro; e
- 5) Satisfação subjetiva.

Esses princípios são mais concisos e embora estejam presentes na norma que é mais recente do que as heurísticas de Nielsen (1993), eles consolidam práticas de trabalhos com usuários reais desenvolvidos por muitos pesquisadores e profissionais da área, que geralmente se guiaram por heurísticas.

No contexto desta tese, são adotadas especialmente as heurísticas de Nielsen para o desenvolvimento do trabalho prático, pois são frequentemente utilizadas por especialistas para avaliações analíticas e são amplamente divulgadas. Além das heurísticas, os padrões web são grandes aliados para o desenvolvimento de sistemas web, considerando que se são padrões, já foram testados quanto à sua utilidade e real relevância.

### **2.3.2 Padrões web**

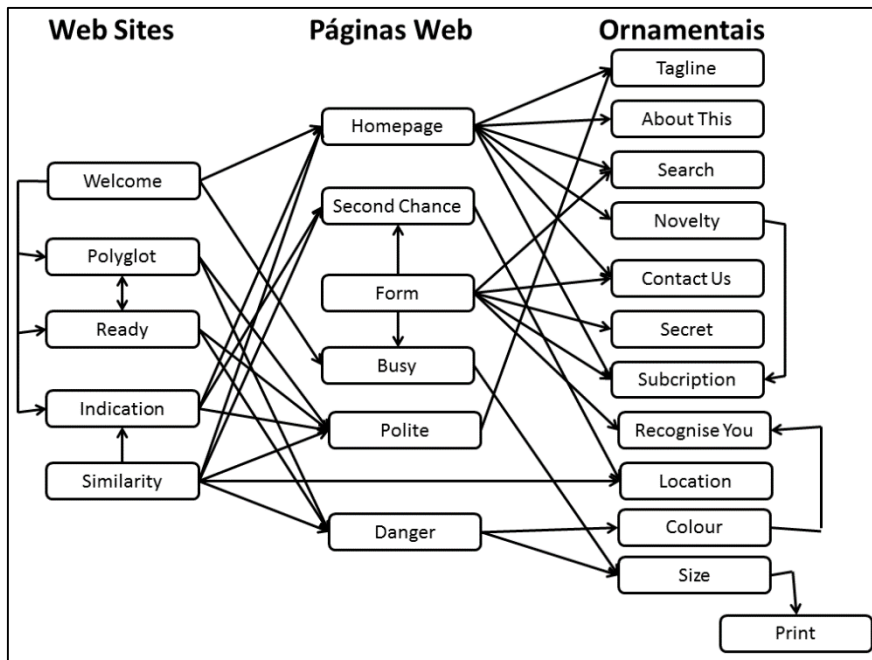
Segundo Borchers (2001), os primeiros padrões foram escritos durante a época do Renascimento, por Francesco do Giorgio (1439-1501) que sistematicamente coletou, estruturou e documentou os conhecimentos arquiteturais de seus projetos na cidade de Siena. Porém, o conceito original de padrões utilizado atualmente surgiu no final da década de 1970, concebido pelo arquiteto e urbanista Christopher Alexander, que escreveu sobre a aplicação de padrões ao design e à construção de cidades e edifícios.

Para Alexander *et al.* (1977), padrão é a essência de uma solução de um problema recorrente em certo contexto. Solução refere-se a cada padrão identificado, que descreve uma solução para o problema especificado. Essência de uma solução são os elementos principais do padrão que são descritos. Problema recorrente significa que o problema deve ter sido repetido diversas vezes de modo a viabilizar o esforço despendido para a descrição da solução. Finalmente, valida e reconhece a solução em um contexto particular (Dias, 2009).

Na área da Computação, o conceito de padrões foi definido inicialmente com sucesso pela comunidade de Engenharia de Software (ES) (Gamma *et al.*, 1995), em que registraram técnicas de programação repetitivas documentando problemas recorrentes e suas soluções (Bittar, 2013). Já no domínio de Interação Humano Computador (IHC), de acordo com Fincher *et al.* (2003), um dos desafios recorrentes está relacionado ao desenvolvimento de técnicas eficientes para documentar e compartilhar o conhecimento em relação à interação com o usuário, que os projetistas adquiriram, e aplicá-las em determinadas circunstâncias.

De acordo com Garzotto *et al.* (1999) e Rossi, Schwabe e Lyardet (1999) é possível obter benefícios usando padrões web, principalmente no auxílio à arquitetura de navegação, especificação de requisitos e modelagem conceitual. Dessa maneira, uma linguagem de padrões é muito importante para desenvolvedores inexperientes, pois existem muitos princípios, diretrizes, regras de ouro etc. cuja utilização não é trivial (Montero *et al.*, 2002).

Sendo assim, Montero *et al.* (2002) formalizaram uma linguagem de padrões<sup>1</sup> para web, classificada em três níveis de escala: websites, páginas web e detalhes de ornamentação, conforme apresentada na Figura 3.



**Figura 3: Linguagem de Padrões de Montero *et al.* (2002).**

<sup>1</sup> Uma linguagem de padrões é definida quando os padrões que fazem parte dela se complementam e são interligados entre si, representando a influência na adoção de um em relação aos que se ligam a ele.

Os 23 padrões que fazem parte da linguagem de padrões, formalizados por Montero *et al.* (2002), visam facilitar a construção de ferramentas e softwares para web, bem como ressaltar as melhores práticas para o desenvolvimento web. Eles podem ser encontrados no ANEXO II com uma breve descrição (problema e solução).

De acordo com Norman (1986, 1999), ao longo de todo o processo de desenvolvimento de sistemas, o projetista deve centrar seu trabalho nas necessidades e escolhas do usuário, ou seja, construir interfaces orientadas à usabilidade e acessibilidade que tenham como propósito atender às necessidades dos usuários. Usabilidade e acessibilidade são conceitos fortemente relacionados, pois ambos buscam melhorar a satisfação e a eficiência de uso da interface (Milne *et al.*, 2005).

## **2.4 Acessibilidade versus usabilidade**

A acessibilidade diz respeito a locais, produtos, serviços ou informações efetivamente disponíveis ao maior número e variedade possível de pessoas independentemente das suas capacidades físico-motoras, perceptivas, culturais e sociais, ou seja, aquilo que se pode atingir, alcançar ou obter facilmente (eMAG, 2005). A usabilidade normalmente se refere à simplicidade e facilidade com que uma interface, um sistema ou equipamento computacional pode ser utilizado (Akhter *et al.*, 2009).

Um conceito mal interpretado em relação à acessibilidade é o de que o desenvolvimento visando acessibilidade implica em limitações na usabilidade. Muitos desenvolvedores argumentam que as recomendações de acessibilidade limitam o uso de tecnologias como *JavaScript*, animações, *applets* e outras tecnologias que podem ser utilizadas para melhorar a usabilidade de um sistema. As recomendações não impedem o uso dessas tecnologias, apenas indicam que elas devem ser utilizadas seguindo certos princípios e diretrizes, e que não devem impedir o acesso de usuários que não disponham delas, para que não se transformem em barreiras (Alexander, 2006).

Petrie e Kheir (2007) destacam a ideia de que, embora haja pontos de conflito quando se prioriza a acessibilidade em relação à usabilidade e vice-versa, ambos os conceitos devem ser considerados de forma integrada para o alcance da usabilidade universal. A usabilidade universal surge como uma forma de aliar e diminuir o espaço incompleto existente entre os princípios de usabilidade e as diretrizes de acessibilidade.

Ressalta-se que um sistema orientado à usabilidade não é necessariamente orientado à acessibilidade, e vice-versa. Ele pode mostrar-se fácil para usuários sem

deficiência, mas inacessível para quem tem necessidades especiais; assim também é possível ter a situação de um sistema web acessível, cuja usabilidade fica prejudicada se o seu projeto de interação for restrito somente a usuários com deficiência.

Para criar um sistema com usabilidade universal é fundamental identificar as dificuldades e habilidades dos usuários, pois elas norteiam o modelo mental de suas interações. Com essa modelagem, é possível obter princípios de usabilidade que, se observados juntamente com as diretrizes de acessibilidade, podem tornar harmoniosa a interação do deficiente e, ao mesmo tempo, oferecer sistemas com conteúdo compreensível e navegável (Ferreira e Nunes, 2008).

De acordo com Petrie e Kheir (2007), os problemas de usabilidade e acessibilidade podem ser vistos como conjuntos que se sobrepõem, o que incluiria três categorias:

- Problemas que afetam **somente pessoas com deficiências** podem ser denominados problemas puramente de acessibilidade;
- Problemas que afetam **somente pessoas sem deficiências** podem ser denominados problemas puramente de usabilidade; e
- Problemas que afetam **pessoas com e sem deficiências** podem ser chamados de problemas de usabilidade universal.

De acordo com Ferreira e Nunes (2008), as sequências da interação do usuário com o computador que são direcionadas à usabilidade, podem ser agrupadas em duas categorias: visualização da informação e entrada de dados. Essas sequências direcionadas à usabilidade somada com acessibilidade podem ser agrupadas em três categorias: percepção da informação, navegação e entrada de dados.

Durante o desenvolvimento de um sistema web, é importante fundamentar-se nos princípios de usabilidade alinhados às diretrizes de acessibilidade desenvolvidas pelo W3C e outros órgãos, para tornar harmoniosa e acessível a interação para o maior número possível de usuários, independentemente de suas características físicas, cognitivas, emocionais etc. (Petrie e Kheir, 2007).

Além da preocupação com as características e diversidade dos usuários, a maneira como eles interagem com a informação e com os serviços disponibilizados na web, também deve ser considerada. De acordo com Ferreira e Nunes (2008), a acessibilidade na web contribui para o design universal, uma vez que a criação de sistemas acessíveis beneficia uma grande variedade de pessoas, não só aquelas que apresentam algum tipo de deficiência, como também as que se encontram em situações precárias ou adversas, como

por exemplo: pessoas com dispositivos de tela pequena (celular, palmtop, PDA - *Personal Digital Assistant* etc.), Internet discada e conexão lenta, equipamentos sem som, impressoras monocromáticas, navegadores web não muito comuns, locais muito ruidosos e ambientes com pouca iluminação.

Design universal é "o design de produtos e ambientes para serem usáveis por todas as pessoas, na maior extensão possível, sem a necessidade de adaptação ou design especializado". Isso não significa que é necessário desenvolver um único produto para todos os usuários, mas é importante que sejam atendidos o maior número possível de princípios do design universal, que são: uso equitativo, flexibilidade de uso, simples e intuitivo, informação perceptível, tolerância ao erro, baixo esforço físico, tamanho e espaço pra aproximação e uso (Connell *et al.*, 1997).

Bailey e Pearson (2011) acreditam que dentre os fatores que contribuem para o problema contínuo da falta de acessibilidade em sistemas web estão a ausência de conhecimento e entendimento dos desenvolvedores e técnicas de implementação insuficientes para apoiar a acessibilidade. Além disso, há baixo envolvimento dos usuários com deficiência para melhorar a acessibilidade na web e há dificuldades de integração das diretrizes de acessibilidade nos processos de desenvolvimentos existentes para apoiar os desenvolvedores (Bigham e Ladner, 2007).

## **2.5 Considerações finais**

O desenvolvimento de sistemas web acessíveis permite uma web mais inclusiva e acessível a todos, apesar das necessidades especiais que possam ser apresentadas pelos usuários (Freire *et al.*, 2008).

Na literatura, existem relatos e propostas de abordagens de desenvolvimento web acessível que guiam o processo de desenvolvimento durante as ações de elicitação de requisitos, design de interface, design de navegação e design da arquitetura, entre outras. Apesar desses esforços, muitos estudos reportam que a acessibilidade ainda é um requisito não funcional pouco abordado em muitos sistemas web (Freire, Bittar e Fortes, 2008; Goette, Collier e White, 2006; Hanson e Richards, 2013).

No próximo capítulo é apresentada a importância do desenvolvimento de sistemas web com o apoio de processos bem definidos, bem como possíveis métricas para a medição desses sistemas.





---

# *Engenharia web: processos, abordagens de avaliação e métricas*

---

## *3.1 Considerações iniciais*

A engenharia web surgiu como uma iniciativa disciplinada que visa apoiar e otimizar os esforços frequentemente exigidos nos processos de desenvolvimento, gerenciamento, manutenção e controle de qualidade de sistemas web. Ela se refere ao estabelecimento e uso de sólidos conceitos científicos e de engenharia, de princípios de gerenciamento e de formas disciplinadas e sistemáticas para desenvolver, tornar disponível e efetuar manutenção de sistemas web de alta qualidade (Deshpande *et al.*, 2002).

Ressalta-se que um dos objetivos da engenharia web, assim como na engenharia de software tradicional, é desenvolver sistemas nos quais suas estruturas, funcionalidades, aspectos navegacionais e de interação com o usuário estejam representados de maneira apropriada. Assim, processos de desenvolvimento para sistemas web devem produzir representações de projetos com características de sistemas tradicionais, como estrutura e

funcionalidades; e também para características orientadas para a web, como navegação e apresentação, que podem ser descritas nos projetos das interfaces, com design de páginas e mapas de navegação.

Neste capítulo são apresentados e discutidos conceitos de engenharia web, especificamente processos, abordagens de avaliação e métricas existentes na literatura da área (livros e artigos em periódicos e conferências) e que são relacionados com a proposta desta tese. O capítulo está dividido da seguinte maneira: processos de desenvolvimento de sistemas web e requisitos são discutidos na Seção 3.2. Na Seção 3.3, abordagens de avaliação de requisitos de acessibilidade e usabilidade em sistemas web são apresentadas resumidamente e discutidas. A importância da medição e das métricas em sistemas web é discutida na Seção 3.4. Por fim, na Seção 3.5 são apresentadas as considerações finais deste capítulo.

### **3.2 Processos de desenvolvimento**

Um processo, segundo o IEEE (*Institute of Electrical and Electronics Engineers*), é uma sequência de passos executados com um determinado objetivo; segundo o CMMI (*Capability Maturity Model Integration*), é um conjunto de ações e atividades inter-relacionadas realizadas para obter um conjunto específico de produtos, resultados ou serviços (Paula Filho, 2009).

Um processo é definido quando há uma descrição específica e bem estabelecida, ou seja: o que é feito (produto/sistema), quando (ações/atividades), por quem (papéis), os recursos que usa (insumos) e os resultados que produz. Os processos podem ser definidos com mais ou menos detalhes, como acontece com qualquer prescrição. As atividades de um processo podem ter ordenação apenas parcial, o que pode permitir paralelismo entre algumas delas (Paula Filho, 2009; Sommerville, 2011).

Conte, Mendes e Travassos (2005) realizaram uma revisão sistemática, em que descreveram sucintamente os processos de desenvolvimento de sistemas web. É possível perceber que vários métodos voltados para o desenvolvimento de sistemas web têm sido propostos, tais como: webML – *web Modeling Language* (Ceri *et al.*, 2000), OOHDM – *Object-Oriented Hypermedia Design Model* (Schwabe, Almeida Pontes e Moura, 1999) e OOWS – *Object-Oriented Web Solutions* (Fons *et al.*, 2003).

A construção de modelos conceituais específicos para a web é uma tarefa relevante nesses métodos (Conte, Mendes e Travassos, 2005). No entanto, não há técnicas

de desenvolvimento padronizadas para projetos de desenvolvimento web (Mendes, Mosley e Counsell, 2004). Segundo Standing (2002), os processos aplicados na prática são em sua maioria *ad-hoc*. Os processos *ad-hoc* não são formalmente definidos e muitas vezes as melhores práticas de engenharia de software, que focam em garantir a qualidade de processo e de produto, não estão explícitas ou mesmo não são atendidas.

Para que as ações que fazem parte da engenharia web possam ser cumpridas, elas devem ser executadas de acordo com um processo ou método de desenvolvimento. Pressman e Lowe (2009) definiram um conjunto de atividades como um *framework* genérico que se aplica à maioria dos projetos de sistemas web, conforme apresentado na Figura 4.

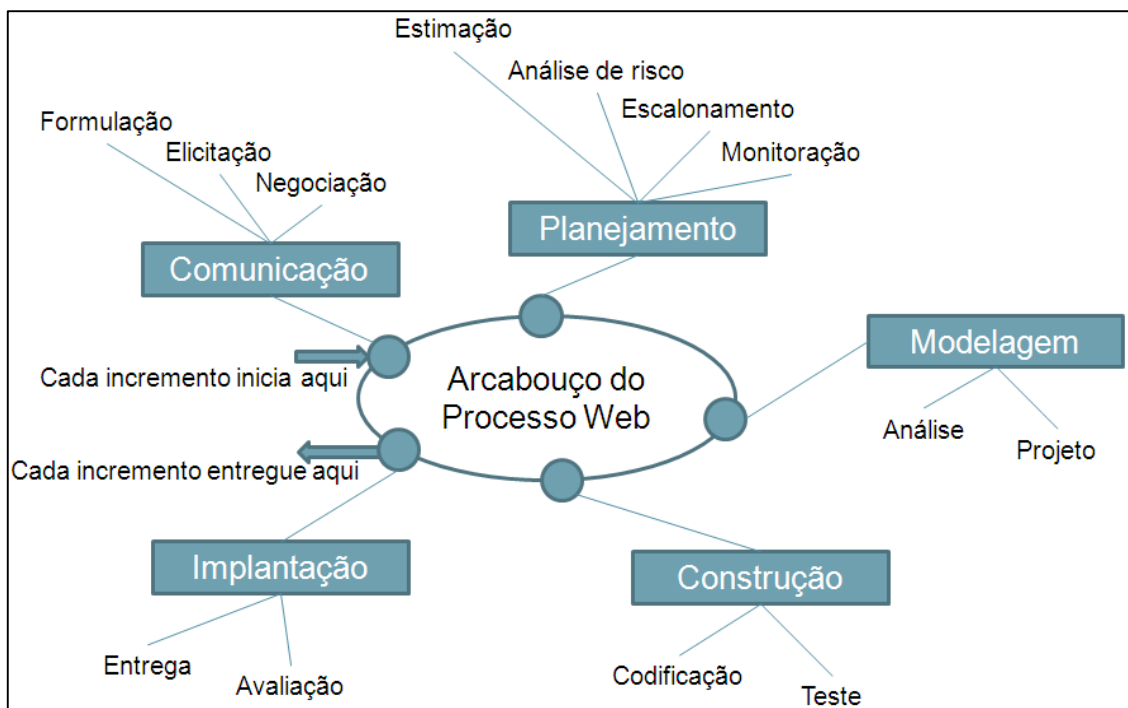


Figura 4: *Framework* genérico de desenvolvimento de sistemas web (Pressman e Lowe, 2009).

As ações do *framework* genérico apresentadas na Figura 4 iniciam-se na formulação do sistema, o que minimiza o possível impacto de novas alterações. Além disso, elas são evolutivas, adequando-se à característica evolutiva de sistemas web. As atividades do *framework* genérico (Pressman e Lowe, 2009) são:

- Comunicação: as necessidades do negócio e a descrição dos objetivos são identificadas, criando um escopo inicial, com o objetivo de realizar o levantamento de requisitos;

- Planejamento: o custo total do projeto é estimado, os riscos associados são avaliados e cronogramas para as tarefas iniciais e subsequentes são definidos;
- Modelagem: elaboração de modelos que auxiliam a comunicação entre o desenvolvedor e o usuário, para que possam entender melhor os requisitos do sistema;
- Construção: codificação das soluções tecnológicas do sistema web sendo desenvolvido; projetos de interface, de arquitetura e de navegação produzindo sistemas executáveis que possam ser testados a fim de revelar erros no código são definidos; e
- Implantação: o incremento do sistema web é entregue ao usuário e ele pode solicitar modificações, que serão alteradas ou adicionadas no escopo do projeto e que são integradas em uma futura iteração.

Pressman e Lowe (2009) também destacam atributos distintos de sistemas web, quando comparados a softwares convencionais (como aplicações *desktop*): uso intensivo de rede, acessos simultâneos, carga imprevisível, disponibilidade, orientada a dados, sensível ao conteúdo, evolução contínua, urgência, segurança e estética.

O desenvolvimento de sistemas web tem aumentado exponencialmente na última década (Molina e Toval, 2009). Uma vez que o tamanho e a complexidade desses sistemas são tipicamente elevados, observa-se a necessidade de elaboração de processos e métodos de desenvolvimento que incorporem requisitos não funcionais como desempenho, segurança, escalabilidade, disponibilidade, manutenibilidade, usabilidade e qualidade visual em uma arquitetura de software para esse tipo de sistemas (Navarro *et al.*, 2008; Acerbis *et al.*, 2008). No escopo de estudos desta tese, o entendimento que se tem é que usabilidade e acessibilidade são requisitos não funcionais essenciais para uma boa experiência dos usuários com os sistemas web (Fortes *et al.*, 2012).

### 3.2.1 Requisitos de software e a web 2.0

Uma característica de software é definida no glossário do IEEE (IEEE *Standard Glossary of Software Engineering Terminology*) como um traço, qualidade ou propriedade de um produto de software (Paula Filho, 2009). Essas características são frequentemente classificadas como (Sommerville, 2011):

- Requisitos funcionais: são declarações de serviços que o sistema deve fornecer, de como o sistema deve reagir a entradas específicas e de como o sistema deve

se comportar em determinadas situações. Em alguns casos, os requisitos funcionais também podem explicitar o que o sistema não deve fazer.

- **Requisitos não funcionais:** são restrições aos serviços ou funções oferecidas pelo sistema. Incluem restrições de *timing*, restrições no processo de desenvolvimento e restrições impostas pelas normas. Ao contrário das características individuais ou serviços do sistema, os requisitos não funcionais, muitas vezes, aplicam-se ao sistema como um todo.

Por exemplo, em um terminal de caixa automático, os tipos de transações bancárias oferecidas são características funcionais. A facilidade de uso, o tempo de resposta e o tempo médio entre falhas são características não funcionais (Paula Filho, 2009).

Com o estabelecimento da web 2.0, novos padrões e características passaram a ser apresentados como requisitos para os sistemas web sendo desenvolvidos. A web 2.0 também introduziu características que impactam na forma como os sistemas deveriam ser implementados e entregues (*released*) (Wang e Zahadat, 2009). Os sistemas deixaram de ser um meio de acesso a informações e passaram a ser utilizadas como uma plataforma que: I) agrega informações de múltiplas fontes e dispositivos; II) permite que indivíduos contribuam com suas ideias e conteúdo; e III) disponibiliza o software na forma de um serviço atualizado continuamente.

Uma consequência da web ser utilizada como uma plataforma é que os sistemas e serviços permanecem em um estágio de “beta perpétuo”, no qual seus recursos e características são constantemente refinados e melhorados (Umbach, 2006).

O baixo custo associado à disponibilização de novos requisitos e funcionalidades dada à característica de atualizações serem realizadas em um servidor único, sem a necessidade de instalação nos múltiplos clientes, torna possível a liberação contínua de novas e melhoradas versões de serviços aos usuários (Munson e Pimentel, 2008). Os usuários podem responder às alterações e oferecer sugestões, criando um ciclo de *feedback* e desenvolvimento (Matheson, 2006). Assim, todas as alterações realizadas no servidor são instantaneamente disponibilizadas a todos os usuários e os componentes do software podem ser revisados iterativa e continuamente.

Nesse contexto, um novo tipo de ciclo de desenvolvimento de software foi estabelecido e deve ser considerado nos processos de desenvolvimento de sistemas web

(Wang e Zahadat, 2009). Na Figura 5 é ilustrado o processo de desenvolvimento de sistemas web 2.0.

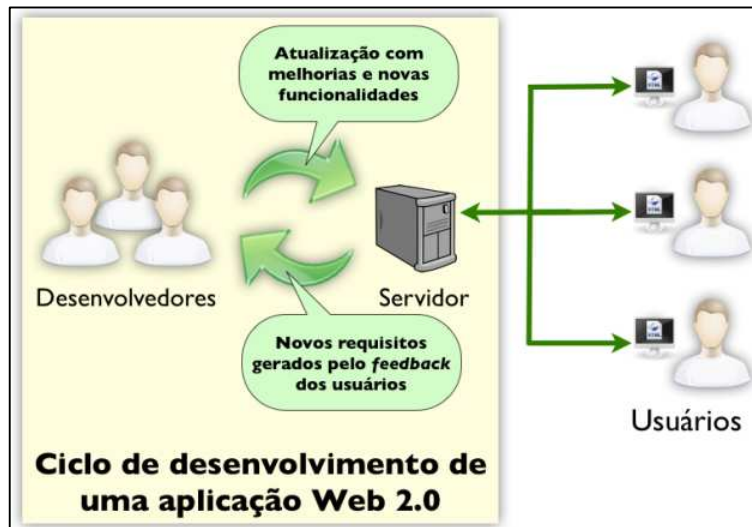


Figura 5: Ciclo de desenvolvimento de um sistema web 2.0.

Na Figura 5, o ciclo de desenvolvimento no qual os desenvolvedores atualizam e recebem os novos requisitos não é interrompido pelo uso do sistema, e os usuários permanecem continuamente obtendo em seus *browsers* as novas versões dos sistemas disponibilizadas pelos desenvolvedores, e gerando novas demandas para que o processo de evolução no desenvolvimento continue, sem interrupção.

Empresas que trabalham com esse tipo de ciclo de desenvolvimento, com entregas constantes de novas funcionalidades, reportam que uma iteração desse ciclo pode levar menos que uma hora, conforme reportado na implementação do sistema *Flickr* (Matheson, 2006). Assim, torna-se necessária a utilização de um processo de desenvolvimento que suporte um ciclo de desenvolvimento iterativo e de rápidas mudanças de requisitos, considerando a iteração de *feedback* dos usuários e melhorias do sistema.

### 3.2.2 Desenvolvimento de sistemas web

Alguns pesquisadores propõem desenvolver sistemas web por meio de uma combinação de técnicas (Gregg, 2009). Thew *et al.* (2009) apresentam uma ferramenta de análise geográfica para epidemiologistas que foi desenvolvida com o uso de cenários, *storyboards*, observação e *workshop* em ciclos iterativos de elicitação de requisitos, exploração de projeto e *feedback* dos utilizadores, conforme apresentado na Figura 6.



Figura 6: *Storyboards* (esquerda, centro) e o protótipo do design da interface (direita) (Thew *et al.*, 2009).

Além de apresentar a ferramenta desenvolvida utilizando *storyboards* e protótipos, os autores mostram uma abordagem de análise de requisitos utilizada no projeto, chamada de ADVISES (Figura 7). Essa abordagem é fundamentada no projeto que faz uso de cenários e da engenharia de requisitos centrado no usuário; ambos defendem o uso de cenários, *storyboards* e protótipos em ciclos iterativos de requisitos de projeto para a elicitación de requisitos e *feedback* dos utilizadores.

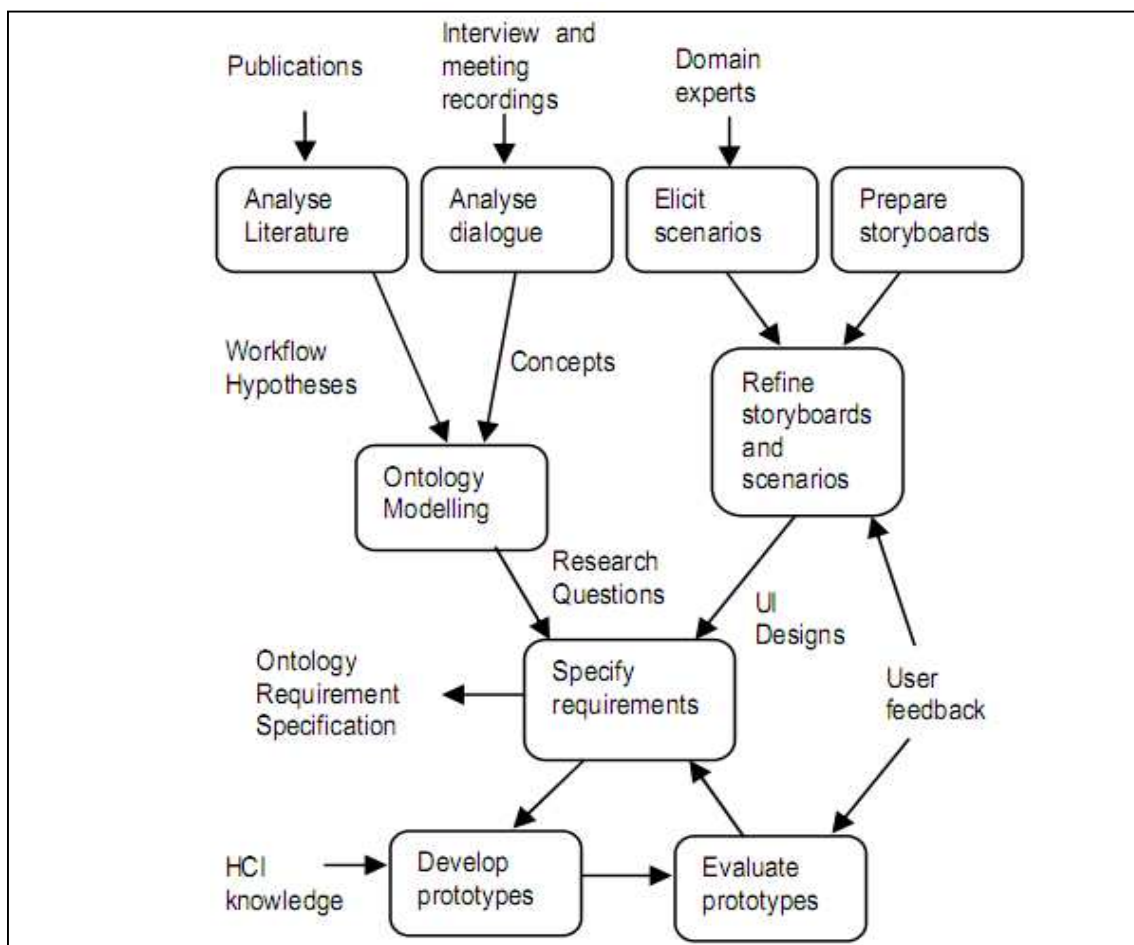


Figura 7: Resumo da abordagem de análise de requisitos ADVISES (Thew *et al.*, 2009).

A abordagem ADVISES pode ser adotada para facilitar a elicitación de requisitos de sistemas complexos, pois muitas vezes os requisitos podem ser alterados constantemente, principalmente em áreas da saúde (Thew *et al.*, 2009). Devido à sua iteratividade, essa abordagem pode facilitar a comunicação entre desenvolvedores e usuários finais.

Há ainda outras pesquisas que se preocupam em descobrir novos requisitos de acessibilidade para os desenvolvedores no desenvolvimento de sistemas web para usuários com deficiências. Akhter *et al.* (2009) propõem algumas diretrizes para promover a confiança em transações comerciais para as pessoas com deficiência, tais como: informar ao usuário que ele está acessando um sistema protegido usando o protocolo (*HTTPs - HyperText Transfer Protocol*) e inserir informações de segurança e política de privacidade no início do sistema ao invés de estarem dispostos no lado inferior direito do sistema, como geralmente ocorre. Para solucionar esse problema, os autores sugerem duplicar essa informação no código do sistema, pois assim ele fica invisível aos olhos humanos, mas perceptível logo no início a pessoas com deficiência visual.

Assim como Masuwa-Morgan (2008), Baguma *et al.* (2009) também acreditam que os requisitos de acessibilidade deveriam ser considerados como requisitos não funcionais, assim como segurança por exemplo. Dessa forma, os desenvolvedores teriam que garantir também a especificação e satisfação de tais requisitos. Baguma *et al.* (2009) propõem uma abordagem para integrar a acessibilidade em análise de requisitos e especificação de atividades. O objetivo desta abordagem é tratar da acessibilidade no contexto do que será feito e quais pessoas estarão participando. Os autores apresentam como os desenvolvedores de sistemas web podem obter requisitos de acessibilidade juntamente com os requisitos funcionais e integrá-los, a fim de obter modelos conceituais com traços explícitos de requisitos de acessibilidade, conforme apresentado na Figura 8.



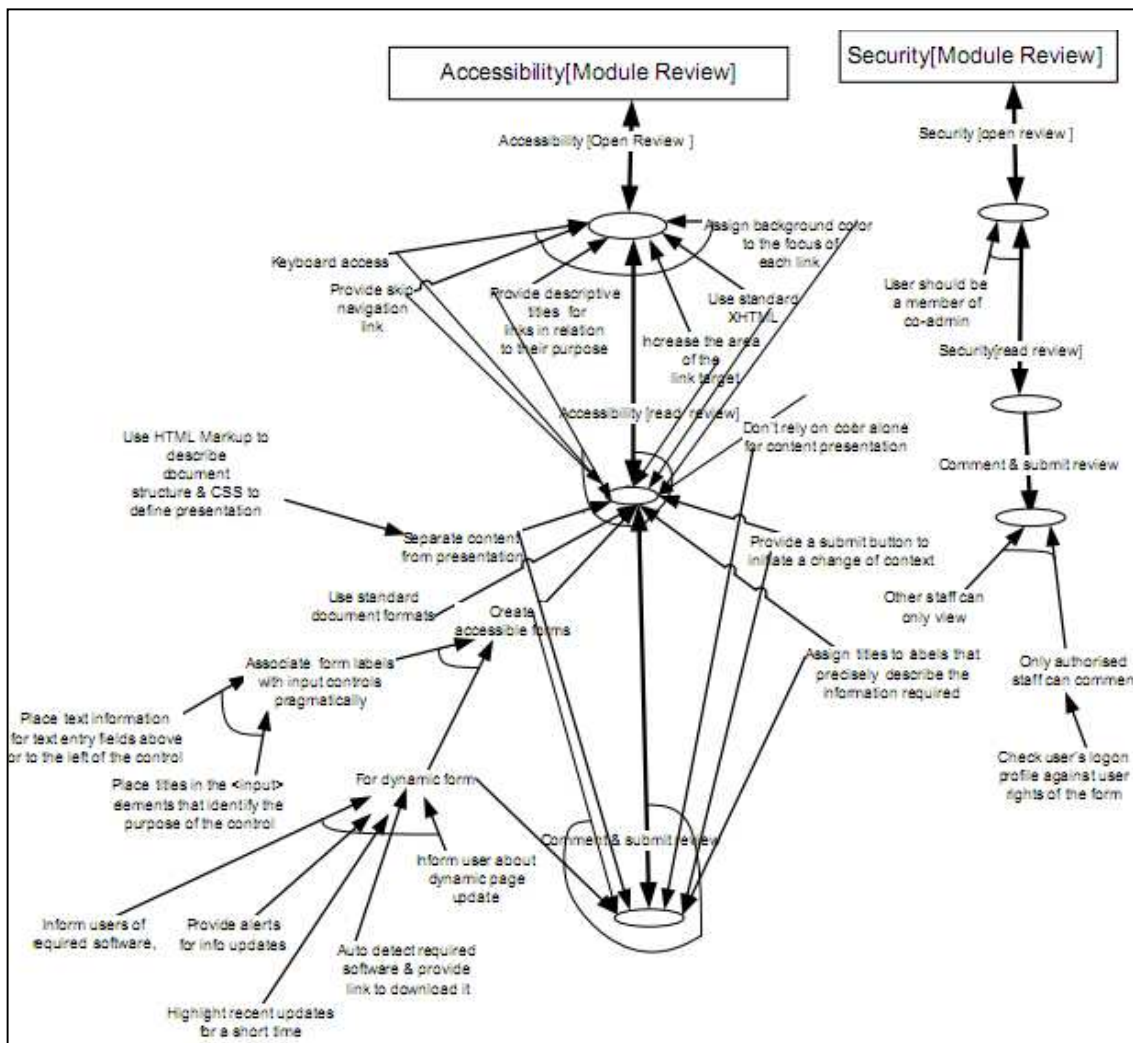


Figura 8: Grafo de requisitos de acessibilidade (Baguma *et al.*, 2009).

Na Figura 8 é apresentado um grafo de requisitos de acessibilidade em que cada tarefa contém um nó com todas as considerações de acessibilidade necessárias. Para chegar nesta união de outros requisitos com os requisitos de acessibilidade, os autores utilizaram técnicas e orientações práticas sobre como abordar acessibilidade no projeto centrado no usuário (*UCD – User Centered Design*), tais como (Henry *apud* Baguma *et al.*, 2009): as metas de negócio e de usabilidade devem incluir a união de requisitos de acessibilidade; compreender as características dos usuários devem incluir usuários com diferentes tipos de deficiências; aspectos ambientais devem incluir situações de limites, como o uso de recursos de tecnologia assistiva por exemplo; cenários devem incluir o uso de uma tecnologia assistiva; os testes de usabilidade devem incluir participantes com deficiências.

Após inserir requisitos de acessibilidade como requisitos não funcionais, Baguma *et al.* (2009) integraram esses requisitos também ao diagrama de caso de uso. De acordo

com os autores, esses requisitos de acessibilidade são necessários para tornar acessível um sistema web. O diagrama de caso de uso com requisitos de acessibilidade é apresentado na Figura 9.

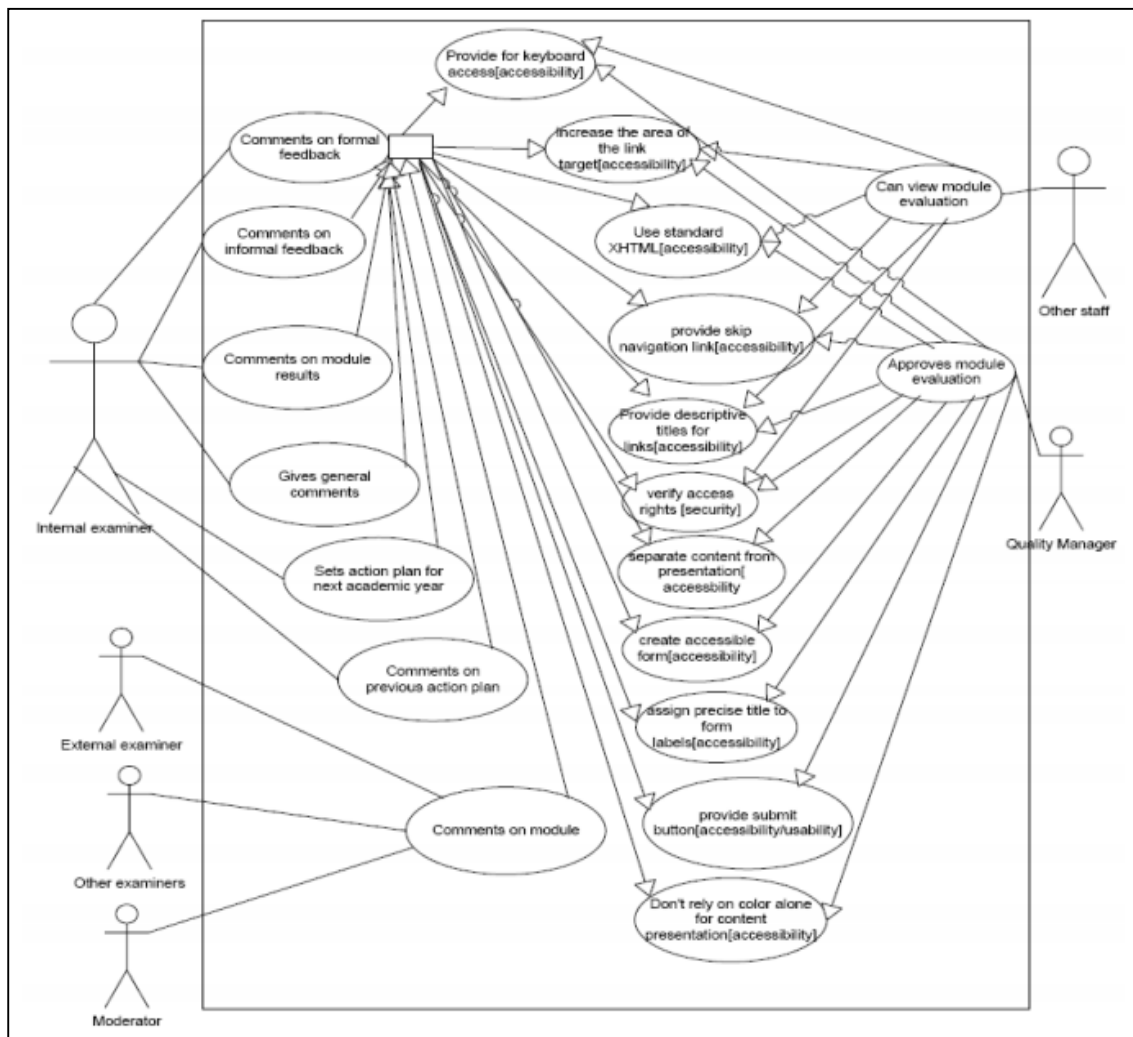


Figura 9: Caso de uso do módulo de revisão com requisitos de acessibilidade (Baguma *et al.*, 2009).

Percebe-se que no caso de uso apresentado na Figura 9 há um retângulo no caso de uso “comentários de *feedback* formal” (*comments on formal feedback*), que é utilizado para simplificar a apresentação. Ao incluir *links* para o retângulo, além de estar ligando ao caso de uso “comentários de *feedback* formal”, estará também fazendo referências aos casos de uso que apresentam considerações de acessibilidade.

García, González e García (2009) propõem técnicas de acessibilidade que podem apoiar os usuários que atualmente são desenvolvedores de seus próprios conteúdos, chamados de *prosumers* (que vem da junção da palavra *producers* – produtores com a palavra *users* - usuários). Os autores classificam essas técnicas de acessibilidade em três categorias: I) técnicas dependentes de plataforma, como por exemplo, fazer com que o

framework utilizado gere código acessível, limitando o que os usuários podem criar; II) técnicas dependentes do criador (desenvolvedor ou usuário), fornecer substituições quando a mídia alternativa não estiver disponível, por exemplo; e III) técnicas dependentes de comunidade como por exemplo criar ferramentas de reparação de acessibilidade ou adicionar acessibilidade por meio de moderação.

Outra frente de trabalho é a de Takagi *et al.* (2004), que verificam a acessibilidade de sistemas web para usuários cegos usando cores e gradações por meio da ferramenta aDesigner<sup>2</sup> (*Accessibility Designer*) conforme apresentado na Figura 10.

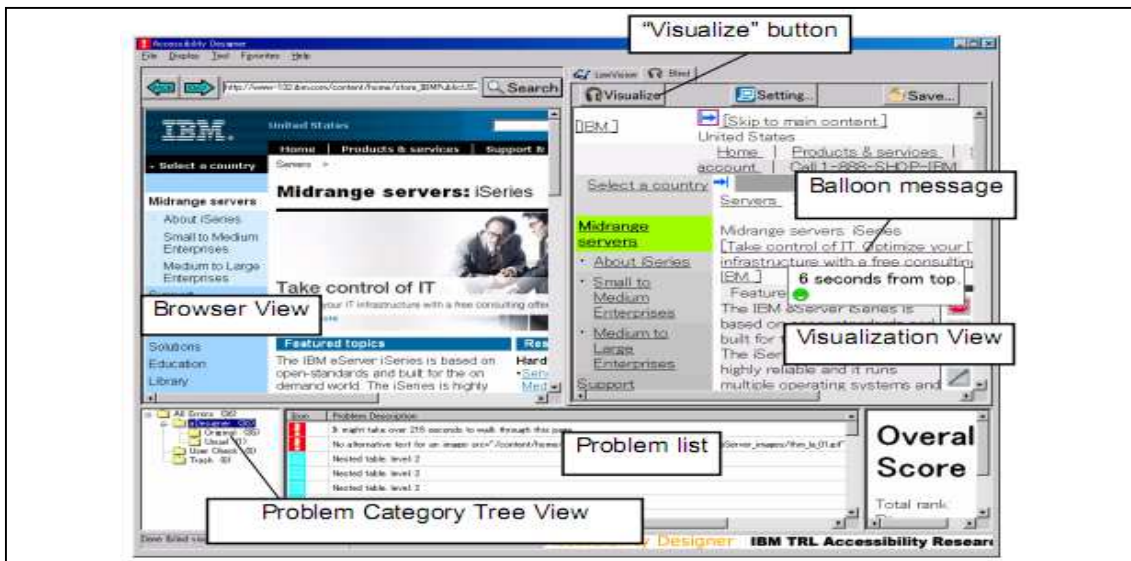


Figura 10: Interface da ferramenta aDesigner (Takagi *et al.*, 2004).

A função de visualização permite que os desenvolvedores compreendam os pontos fracos em seus sistemas, e reconheçam partes que são acessíveis ou inacessíveis. Como resultado da aplicação desta ferramenta, os autores apresentam a interface do sistema IBM<sup>3</sup> após a avaliação, conforme apresentado na Figura 11.

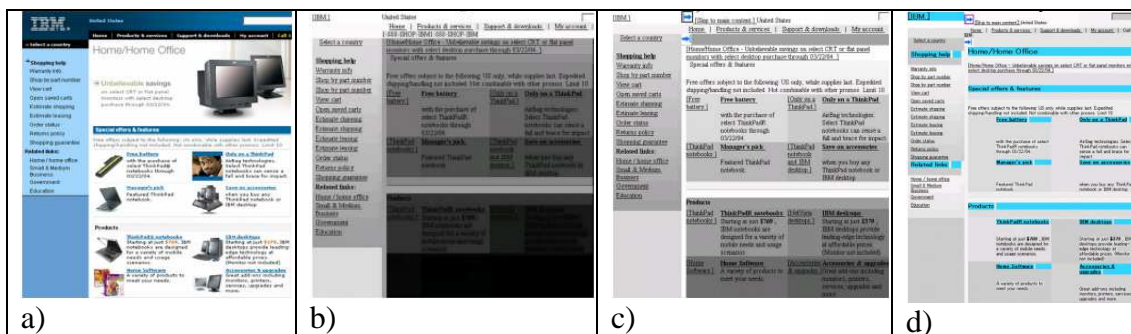


Figura 11: Avaliação de um sistema web usando a ferramenta aDesigner (Takagi *et al.*, 2004).

<sup>2</sup> Disponível em: <http://www.alphaworks.ibm.com/tech/adesigner>

<sup>3</sup> Disponível em: <http://www.ibm.com>

A ferramenta aDesigner calcula os tempos de acesso ao conteúdo por meio de um leitor de tela a partir do topo de uma página para cada elemento que o usuário possa navegar, em seguida, visualiza esses tempos usando gradações de cor. Na Figura 11 (a), um sistema web é apresentado em sua versão visual; na Figura 11 (b) é apresentado o sistema inacessível, já que não há marcas de título ou quaisquer *links* que permitam ir para o conteúdo sem precisar navegar por todo o sistema. As áreas de conteúdos sombreadas em preto mostram que o usuário cego vai demorar muito tempo para chegar a elas. Na Figura 11(c) é mostrado o mesmo sistema melhorado com um *link* que salta para o conteúdo diretamente. A área de conteúdos é sombreada com uma cor cinza claro, mostrando como os usuários cegos podem agora acessar o conteúdo principal facilmente. Já na Figura 11 (d) existem vários destaques em azul, seguido por áreas de cor mais clara. Nesse caso, isso acontece porque há funções que permitem ir diretamente para o conteúdo de cada título do sistema, facilitando a navegação por parte dos usuários cegos, ajudando-os a compreender a visão geral do sistema.

Essa ferramenta pode ser muito útil aos desenvolvedores, já que eles podem reconhecer como os usuários podem chegar ao conteúdo principal, rapidamente ou lentamente, referindo-se às diferenças das cores. A cor mais escura indica que o usuário gastará um longo tempo, enquanto uma cor mais clara indica um tempo mais curto para acessar um conteúdo específico.

Seguindo a mesma linha de estudos de Takagi *et al.*, (2004), Rubegni *et al.* (2008) também se preocupam em projetar para usuários cegos. Os autores apresentam AURA, um *framework* para garantir acessibilidade para usuários cegos. Entre seus principais componentes, alguns se destacam:

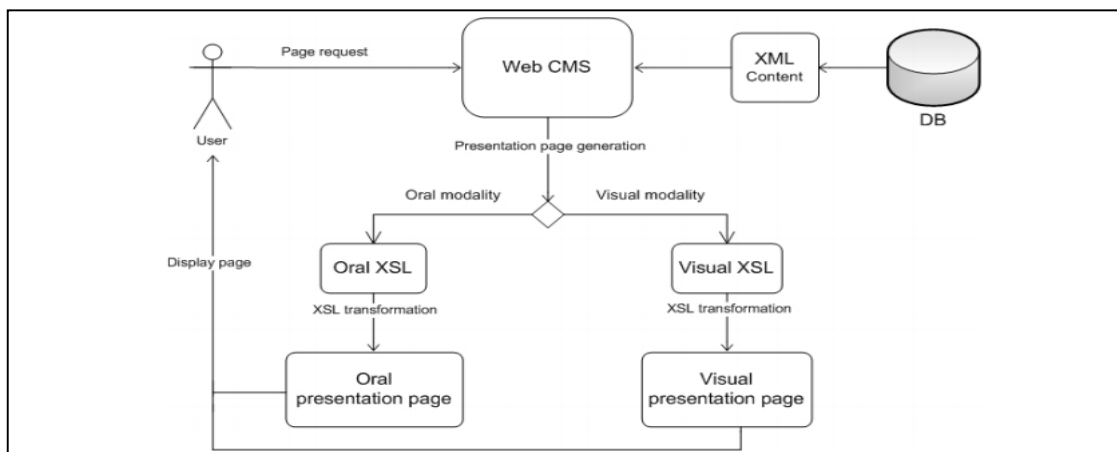
- **Design para acessibilidade:** uma regra deste componente seria quebrar uma longa lista de itens em sublistas (no máximo 6-8 itens de comprimento). Apesar de parecer trivial, uma grande quantidade de sistemas que são compatíveis com as diretrizes do W3C não aplica essa regra;
- **Leitura da estratégia:** a implementação deste componente pode seguir duas regras:
  - I) TAGS: escolhidas entre um conjunto de possibilidades pré-definidas como marcas, elas podem ser embutidas no HTML. Se leitores de tela forem utilizados, essas marcas os forçam a um comportamento

interativo adequado, como o de listagem das seções, permitindo saltar de uma seção para outra; e

- II) Duas apresentações: duas páginas de apresentação diferentes podem ser geradas a partir do mesmo conjunto de conteúdos, uma apresentação para a versão visual da página e outra versão para "ouvir" o mesmo conjunto de conteúdos.

Rubegni *et al.* (2008) apresentam também um sistema desenvolvido seguindo os passos do *framework*. Nesse sistema, os desenvolvedores optaram por adotar a regra de duas apresentações do componente leitura da estratégia, ou seja, gerar duas apresentações para o mesmo conteúdo: uma versão visual do sistema e uma versão oral. O usuário pode mudar de uma modalidade para a outra.

De acordo com a Figura 12, quando o usuário solicita uma página, o CMS (*Content Management System*) gera a página desejada. O conteúdo é obtido a partir de uma base de dados como um arquivo XML (*Extensible Markup Language*); a página é então gerada por meio de uma transformação XSL (*Extensible Stylesheet Language*) do arquivo XML. Essa transformação é diferente de acordo com a modalidade selecionada pelo usuário.



**Figura 12: Framework AURA (Rubegni *et al.*, 2008).**

O mesmo conteúdo pode ser apresentado por meio de duas estratégias diferentes: visual (Figura 13, à esquerda) e oral (Figura 13, à direita). Isto significa que o mesmo sistema pode ser gerado nas duas modalidades. As diferentes seções do design estão destacadas na versão visual e sublinhadas na versão oral.

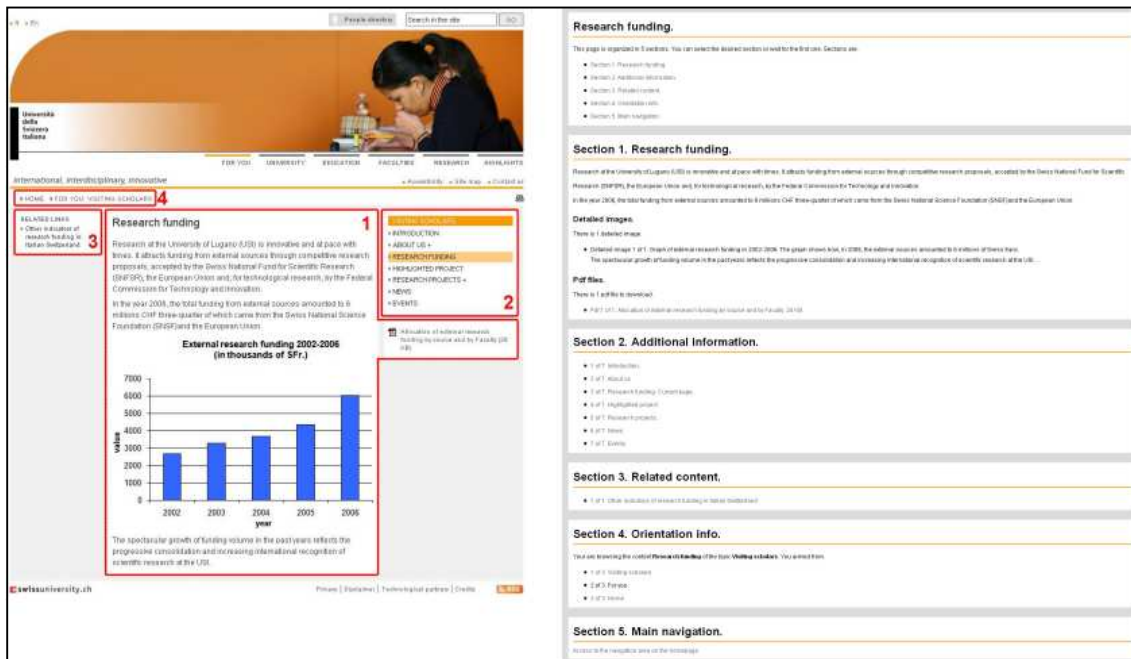


Figura 13: Sistema visual à esquerda e sistema oral à direita (Rubegni *et al.*, 2008).

Para desenvolvedores que não têm muita experiência em definir os locais de cada componente na interface, Lai *et al.* (2010) descrevem modelos computacionais fundamentados em pesos visuais para calcular a simetria e equilíbrio de imagens e textos sobrepostos. O trabalho de Christiernin (2010) segue a mesma ideia de Lai *et al.* (2010), pois ambos preocupam-se com a disposição e quantidade de objetos na tela.

Christiernin (2010) separa a interface gráfica do usuário em várias camadas (MLD - *Multi-Layered Design*), fundamentados nas habilidades dos usuários, nas competências e nos níveis de experiência. As camadas são ordenadas em uma sequência lógica com a primeira camada contendo um conjunto mínimo de funções e a última camada contendo todas as funcionalidades disponíveis. As camadas no MLD podem ser ordenadas em diferentes tipos de estruturas, conforme apresentado na Figura 14.

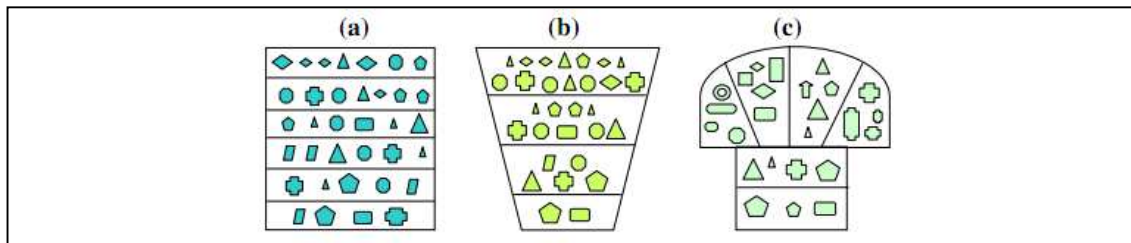
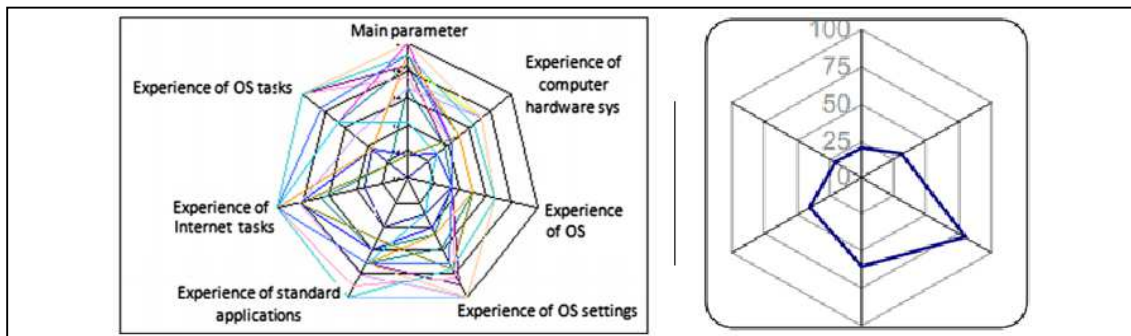


Figura 14: Estruturas das camadas no MLD (Christiernin, 2010).

Com as funcionalidades separadas por camadas, os usuários experientes que lidam com tarefas avançadas, podem começar a trabalhar diretamente nos níveis mais elevados das camadas que proporcionam funções mais complexas. O resultado é uma solução de

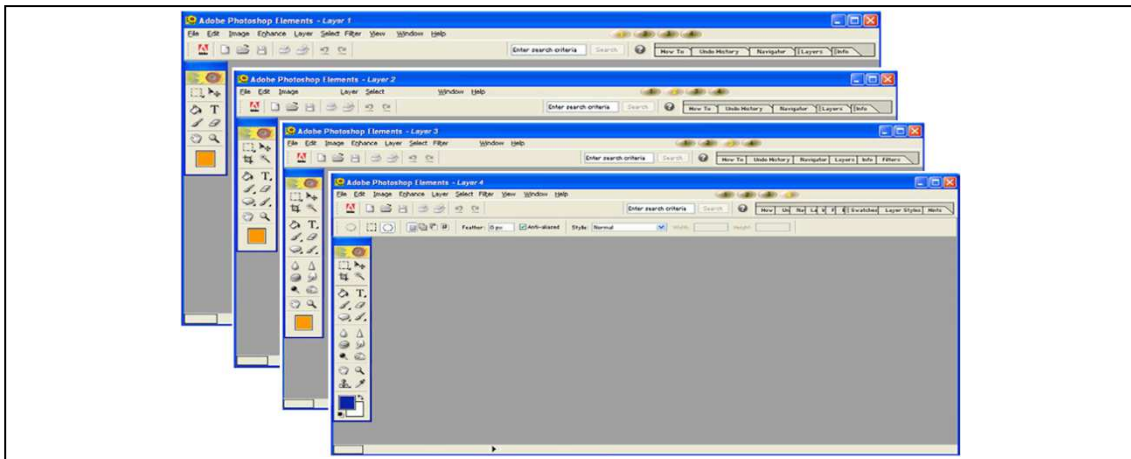


projeto individual. Após dividir as funcionalidades, a tarefa é identificar quais camadas cada usuário poderá utilizar. Para isso, os autores enfocam o processo de diagrama radar para camadas múltiplas (RDPM - *Radar Diagram Process for Multiple layers*). A Figura 15 apresenta o processo de diagrama que ajuda a identificar o nível de experiência do usuário (figura à esquerda) e apresenta um modelo de experiência de um usuário (figura à direita).



**Figura 15:** Nível de experiência dos usuários por meio do RDPM (Christiernin, 2010).

Soluções de design em camadas permitem escolhas flexíveis de interfaces, em que os usuários podem escolher suas próprias camadas para trabalhar. A Figura 16 apresenta um sistema desenvolvido por meio de design de camadas (menu de opções à esquerda), comumente encontradas em jogos computacionais em que o jogador deve completar o primeiro nível antes de tentar o segundo e assim por diante.



**Figura 16:** Aplicação desenvolvida por meio de design em camadas (Christiernin, 2010).

É essencial estimar as habilidades dos usuários em usar diferentes tipos de funcionalidade em um mesmo sistema. Deve-se, de uma forma adequada, medir ou identificar os fatores que afetam o uso da camada e quão bem os usuários serão capazes de usar a estrutura do MLD. Além de fazer o design de um sistema por meio de camadas de experiência do usuário, as camadas podem ser feitas pensando nos perfis ou

deficiências dos usuários, com soluções individuais, como a solução apresentada por Rubegni *et al.* (2008) para deficientes visuais.

Já o trabalho de Chen, Yesilada e Harper (2010) investiga os possíveis erros que os usuários cometem ao interagir com aparelhos móveis, geralmente relacionados à digitação e apontamento. Há pesquisas que se preocupam em avaliar a acessibilidade de sistemas web por meio de ferramentas que percorrem o código HTML para verificar se está ou não de acordo com o conjunto de regras estabelecidas; no final, geram relatórios com uma lista dos problemas encontrados, que devem ser corrigidos para que o sistema possa ser considerado acessível.

Um dos problemas dessas ferramentas é que elas dependem de técnicas de verificação sintática dos sistemas e só avaliam se a sintaxe está de acordo com as diretrizes do WCAG (W3C, 1999a, 2008b) ou eMAG (2005), ou seja, elas detectam erros que se limitam à camada de descrição de *tags*. Além disso, a avaliação automatizada produz um resultado limitado sobre a acessibilidade do sistema, pois somente 50% das diretrizes da WAI podem ser avaliadas de forma totalmente automática (Cooper e Rejmer, 2001).

### **3.3 Avaliação de sistemas web**

Como são necessários testes de funcionalidade para verificar a robustez da implementação, é necessária uma avaliação de interface para avaliar a qualidade do uso de software. Quanto mais cedo os problemas forem encontrados na interface, menor o custo para repará-los.

Um desenvolvedor não deve presumir que basta seguir métodos e princípios de design de interface para garantir uma elevada qualidade do seu sistema web. Além disso, ele também não deve presumir que os usuários são como ele, e que, portanto, a sua avaliação individual seria suficiente para atestar esta qualidade (Preece, Rogers e Sharp, 2002).

Alguns dos principais objetivos para avaliar sistemas web são (Hailpern *et al.*, 2009; Preece, Rogers e Sharp, 2002):

- Identificar as necessidades dos usuários e/ou verificar a compreensão dos desenvolvedores sobre estas necessidades;
- Identificar os problemas de interação ou de interface;
- Investigar como uma interface afeta a maneira como os usuários interagem;



- Comparar alternativas de projeto de interface;
- Attingir metas mensuráveis em medidas de usabilidade; e
- Verificar a conformidade com um padrão ou conjunto de heurísticas.

Infelizmente, é comum encontrar gerentes de projetos que pensam apenas nos custos envolvidos na realização de seus sistemas. Isso geralmente acontece devido à falta de benefícios associados a essas avaliações. Dependendo do momento em que a avaliação é conduzida, os benefícios podem ter efeito imediato sobre grandes reparos no início do desenvolvimento ou mesmo em longo prazo, apoiando o planejamento das próximas versões de software (Prates e Barbosa, 2003).

Realizar avaliações em sistemas web, para apresentar os aspectos que podem ser melhorados, é um fator crítico. Se avaliações fossem realizadas, evitariam vários problemas que as interfaces com baixa qualidade podem ter, tais como:

- Exigir treinamento excessivo;
- Desencorajar a navegação;
- Confundir os usuários;
- Induzir os usuários ao erro;
- Gerar insatisfação; e
- Reduzir a produtividade, não trazendo o retorno do investimento esperado.

Estes problemas podem ser detectados por vários métodos de avaliação, realizados em todo o processo de desenvolvimento. De acordo com Mack e Nielsen (1994), é muito importante que um método de avaliação ajude os desenvolvedores na identificação de problemas de usabilidade (identificar, classificar e contar o número de problemas de usabilidade encontrados durante a inspeção) e na seleção dos problemas que devem ser corrigidos (após a identificação dos problemas, a equipe do projeto deve redesenhar a interface para corrigir o maior número de problemas. Os problemas a serem corrigidos são priorizados de acordo com a gravidade do problema e aos custos associados com a correção).

### **3.3.1 Avaliação de acessibilidade e usabilidade em sistemas web**

A avaliação de acessibilidade na web é uma ação que pode ser realizada durante as etapas de desenvolvimento de sistemas web ou para determinar o nível de acessibilidade atingido por determinados sistemas. Em ambos os casos, o objetivo

principal é a identificação de barreiras de acessibilidade a fim de reportar esses problemas para correção (Fortes *et al.*, 2005).

Uma das formas mais simples de se realizar a avaliação de acessibilidade de um sistema web é por meio de ferramentas de avaliação automática de acessibilidade. Essas ferramentas têm por finalidade a avaliação de acordo com um conjunto de padrões de acessibilidade, como por exemplo, as diretrizes de acessibilidade do WCAG elaboradas pela WAI. Como ferramentas de avaliação automática de acessibilidade podem ser citadas<sup>4</sup>:

- DaSilva<sup>5</sup>: avalia a conformidade com o WCAG 1.0, WCAG 2.0 e e-MAG.
- Hera<sup>6</sup>: avalia a conformidade com o WCAG 1.0.
- A-Checker<sup>7</sup>: avalia a conformidade com o WCAG 1.0 e o WCAG 2.0.

Processos de avaliação de acessibilidade de interfaces web, restritos a revisões com ferramentas de avaliação de acessibilidade automáticas, não são suficientes (Tanaka e Rocha, 2011; López *et al.*, 2012; Power *et al.*, 2012). Um dos problemas com as avaliações baseadas apenas na verificação automática de acessibilidade é que nem todos os pontos de verificação podem ser verificados automaticamente (Fernandes *et al.*, 2013). Outro problema, citado por Brajnik (2008a), é que as diretrizes, como as do WCAG, não permitem que o avaliador distinga problemas sérios de trivial, independentemente da existência de níveis de prioridade bem definidos.

Por exemplo, na diretriz 1.1, do WCAG 1.0, recomenda-se que todas as imagens que são apresentadas em um sistema web devem ter um texto alternativo. No entanto, de acordo com Brajnik (2008a), a maioria das imagens da web tem um propósito "emocional", e nem sempre o texto agregado à imagem repassa esse fator. Essa percepção somente é encontrada com a inspeção de um especialista. Assim, a avaliação de acessibilidade em sistemas web deve incluir, além de ferramentas de verificação automáticas, verificações de tarefas de usuário focadas nas dificuldades encontradas na realização de trabalhos diários.

---

<sup>4</sup> Além dos exemplos citados, é possível encontrar uma lista extensa de ferramentas na página da W3C, disponível em: <http://www.w3.org/WAI/ER/tools/complete>

<sup>5</sup> Disponível em: <http://www.dasilva.org.br/>

<sup>6</sup> Disponível em: <http://www.sidar.org/hera/index.php.en>

<sup>7</sup> Disponível em: <http://achecker.ca/checker/index.php>

Considerando essa limitação das ferramentas de avaliação automática de acessibilidade, a WAI apresenta formas de avaliar a acessibilidade e conformidade com suas diretrizes de sistemas web. Dentre elas pode-se destacar:

- Avaliação preliminar: revisão de sistemas, combinando a utilização de ferramentas automáticas de avaliação e a análise manual de determinados critérios de acessibilidade. Essa técnica não é suficiente para determinar se uma página é acessível ou não, no entanto pode reportar barreiras de acessibilidade mais óbvias por meio de uma rápida avaliação inicial (W3C, 2013). Os passos necessários para sua aplicação são: seleção de amostras representativas da página a ser avaliada, avaliação manual das amostras em diferentes navegadores; avaliação manual em navegadores especializados (navegadores textuais, com síntese de voz, entre outros); utilização de ferramentas de avaliação automática de acessibilidade; e sumarização dos resultados encontrados.
- Avaliação de conformidade: avaliação de conformidade de uma página com as diretrizes da WAI (WCAG). Combina o uso de ferramentas de avaliação automática e testes manuais de acessibilidade (W3C, 2008a). As atividades que compõem a avaliação de conformidade são: determinar o escopo da avaliação: identificar o nível de conformidade do WCAG a ser atingido (A, AA ou AAA) e selecionar amostras representativas de sistemas que serão avaliados; utilizar ferramentas de avaliação automática; avaliação manual das amostras; e sumarização dos resultados encontrados.
- Avaliação com usuários: a participação de usuários nos processos de avaliação é tida como uma forma de auxílio à compreensão, por parte dos desenvolvedores, sobre como usuários com deficiência interagem com o sistema por meio de recursos de tecnologia assistiva (W3C, 2010). Os problemas levantados nessa avaliação são classificados, de acordo com suas respectivas causas, em: problemas de marcação do conteúdo, sem disponibilidade de funcionalidades na tecnologia assistiva sendo utilizada, não familiaridade do usuário com recursos de tecnologia assistiva e problemas de usabilidade comuns a todos os usuários. Os passos que envolvem o desenvolvimento desse tipo de avaliação são: realização de uma avaliação preliminar de acessibilidade no sistema; e uma avaliação informal com o

usuário, analisando o seu uso com o sistema web e considerações a respeito da interface apresentada.

Há ainda trabalhos que destacam a importância da avaliação com o usuário, por meio de uma abordagem UCD (Romen e Svanaes, 2008; Brajnik, Yesilada e Harper, 2010; Watanabe, 2007). São propostas técnicas que consideram a interação com o usuário como forma de garantia de acessibilidade. São utilizados, como variáveis de avaliação contextos de uso (categorias de usuário, cenários de utilização e objetivos dos usuários) (Brajnik, 2008a, 2006) e testes com os usuários (Watanabe, 2007; Romen e Svanaes, 2008). Dentre essas técnicas são destacadas:

- *Think aloud*: essa técnica consiste de avaliações qualitativas de uma determinada interface, por meio da análise de comentários feitos em voz alta por usuários (Brajnik, 2008a; Romen e Svanaes, 2008). Os usuários são solicitados a verbalizar seus pensamentos, sentimentos e opiniões enquanto realizam uma ou mais tarefas no sistema em avaliação. A verbalização permite que o avaliador identifique os problemas de usabilidade enfrentados pelos usuários em situações reais, por meio da observação da interação e da interpretação das considerações expressas verbalmente pelos usuários. A verbalização pode ser: simultânea (ocorrer no momento em que o usuário interage com o sistema), consecutiva (ocorrem em entrevista logo após a realização do teste) ou estimulada (o avaliador, ao invés de aguardar a verbalização livre do usuário, estimula-o com perguntas diretas).
- *Barrier Walkthrough*: técnica de inspeção de acessibilidade em que o contexto de utilização de um sistema é explicitamente considerado. Um avaliador deve identificar um determinado número de barreiras de acessibilidade, de acordo com o contexto sendo avaliado, que são interpretações e extensões de princípios de acessibilidade bem definidos (Brajnik, 2008b). Essa técnica foi desenvolvida com base no percurso cognitivo (técnica de inspeção de usabilidade). Devem ser efetuadas as seguintes atividades durante a execução dessa técnica (Brajnik, 2006): levantamento de contextos de uso do sistema (características dos usuários, cenário de utilização, atividades a serem realizadas e os objetivos da interação com o sistema); avaliação de presença de barreiras de acordo com um determinado contexto sendo avaliado (essa etapa deve ser realizada para cada

um dos contextos considerados na primeira etapa); e elaboração de uma lista de problemas associados a determinados contextos, seus respectivos níveis de severidade e atributos de *performance* que sejam afetados (eficiência, produtividade, satisfação e segurança).

- Teste com usuários: técnica de avaliação quantitativa relativa à *performance* de usuários no uso de um sistema. Como uma forma de evidência empírica da efetividade da diretriz 3.5, Watanabe (2007) utilizou esse método de avaliação analisando os tempos necessários para completar determinadas tarefas em sistemas que utilizam marcações de elementos “*heading*” adequadamente e sistemas que não os utilizam.

Auditorias de acessibilidade, realizadas por meio de avaliação de conformidade, consistem em verificar as características de um sistema web para saber se ele satisfaz os critérios de conformidade que são especificados no WCAG (ou e-Mag, no caso de sistemas nacionais). Para os engenheiros web que estão familiarizados com acessibilidade, este é o tipo mais comum de avaliação feita, o que se deve à influência que o WCAG tem sobre a visão política e jurídica (Freire, 2012).

Já a avaliação de usabilidade foca em perceber o quão bem os usuários podem aprender e usar um produto para atingir seus objetivos. Também se refere ao quanto os usuários estão satisfeitos com o produto. Para obter essas informações, os especialistas usam uma variedade de métodos que coletam *feedback* dos usuários sobre o produto, bem como possíveis planos para um novo produto (Usability.gov, 2014).

De acordo com Karat (1993), antes de declarar se um sistema web está pronto para ser usado, é importante saber se ele apoia adequadamente os usuários na execução de suas tarefas e no ambiente em que ele será usado.

O envolvimento de usuários com deficiência torna os processos de implementação de acessibilidade mais efetivos e eficientes. No entanto, mesmo estudos de grande escala não seriam capazes de cobrir toda a diversidade de deficiências, estratégias de adaptação e recursos de tecnologia assistiva existentes. Assim, a avaliação por meio de abordagens centradas no usuário e em conformidade com o WCAG é tida como uma das formas mais adequadas de se avaliar a acessibilidade de sistemas web atualmente (W3C, 2010).

As abordagens propostas na literatura, para avaliar o grau de usabilidade de um sistema, são diferentes. No entanto, todas elas visam o mesmo objetivo: "a avaliação de usabilidade de um sistema interativo deve verificar características de qualidade de

software e obter indícios do nível de satisfação do usuário, identificando problemas de usabilidade durante tarefas específicas em seu contexto de uso. Um problema de usabilidade pode ser definido como de qualquer característica observada em uma dada situação, que pode atrasar, dificultar ou inviabilizar a conclusão de uma tarefa, ser irritante, embaraçosa ou traumatizante ao usuário" (Dias, 2003).

Os métodos de avaliação de usabilidade podem ser classificados em três grupos, de acordo com sua natureza: os métodos de inspeção, métodos de testes com usuários e métodos baseados em modelo. A avaliação heurística é considerada uma avaliação por inspeção. Métodos de inspeção foram formalizados primeiramente na década de 90 (Nielsen e Molich, 1990; Nielsen, 1993), baseados na inspeção realizada por especialistas em usabilidade. Os avaliadores, de acordo com Nielsen, podem ser especialistas em usabilidade, consultores de desenvolvimento de software, usuários finais com conhecimento especializado ou outros profissionais.

Rosa e Veras (2013) realizaram uma avaliação heurística, nos sites das revistas "O Estado de São Paulo" e "Folha de São Paulo". O resultado da avaliação mostra uma concentração de violações de usabilidade em heurísticas relacionadas à visibilidade do status do sistema, consistência e normas e prevenção de erros. No entanto, a avaliação não detectou problemas de acessibilidade, como mudar o tamanho da fonte dos textos ou prover navegação via teclado, por exemplo.

Santos *et al.* (2011) investigaram o efeito de oito diferentes tipos de menus quando usados por pessoas com idade superior a 40 anos que tinham alguma experiência na utilização da Internet. O experimento revelou que em menus que apresentaram propriedades comumente encontradas em sistemas web, o tempo de conclusão da tarefa e o número de erros foi menor. No entanto, faltava recursos para distinguir opções de menu que já haviam sido visitados e um tempo maior para mostrar menus selecionados por usuários que têm pouca coordenação.

O trabalho realizado por Freire, Goularte e Fortes (2007) apresentam uma abordagem baseada em métricas para avaliação de sistemas da web utilizando ferramentas de avaliação de acessibilidade automáticas. Os resultados mostraram que o uso da métrica de acessibilidade pode ser uma ferramenta para ajudar na medição e verificação da acessibilidade nos sistemas web governamentais. No entanto, uma avaliação de usabilidade não foi contemplada e nem todos os postos de controle de acessibilidade puderam ser automaticamente verificados.

Freire, Bittar e Fortes (2008) descrevem uma abordagem com base em métricas para medir a acessibilidade web em sistemas web de municípios brasileiros. Monitorar os níveis de acessibilidade de sistemas web é uma atividade de grande importância para melhorar a acessibilidade no produto sendo desenvolvido. Dada a natureza de atualizações frequentes que caracteriza os sistemas web, monitorar a evolução de acessibilidade requer métricas eficazes para impedir que um sistema web seja colocado em produção contendo barreiras de acessibilidade (Vigo *et al.*, 2007; Freire *et al.*, 2009).

De fato, há muitos meios para avaliar acessibilidade e usabilidade separadamente. No entanto, as características de usabilidade e acessibilidade são difíceis de serem avaliadas em um sistema web por profissionais que não têm conhecimento de como seus princípios e suas diretrizes poderiam ser unidas. Como em geral, os desenvolvedores web não são usuários dos sistemas web que desenvolvem, eles têm que priorizar o tempo de desenvolvimento. Portanto, instrumentos de avaliação de itens subjetivos relacionados com as questões de usabilidade e acessibilidade são necessários. Pode-se dizer que um problema que afeta um perfil específico de usuários é tipicamente um problema de acessibilidade; já um problema que afeta uma variedade ampla de pessoas, é um problema de usabilidade.

### **3.4 *Medições e métricas de sistemas***

Segundo o IEEE, uma métrica é “uma medida quantitativa do grau com que um sistema, componente ou processo possui um dado atributo” (Paula Filho, 2009). A medição de software preocupa-se com a derivação de um valor numérico ou o perfil para um atributo de um componente de software, sistema ou processo. Comparando esses valores entre si e com os padrões que podem ser aplicáveis, é possível tirar conclusões sobre a qualidade do software ou avaliar a eficácia dos métodos, das ferramentas e dos processos de software.

Nesse contexto, estudos considerando a utilização das diretrizes da WAI, na implementação de sistemas acessíveis, destacam a importância de métricas de acessibilidade no monitoramento e gerenciamento de acessibilidade no processo de engenharia web (Freire *et al.*, 2008). Quando conduzido adequadamente, o monitoramento contínuo de métricas relacionadas a atributos de qualidade, como a acessibilidade, promove importantes contribuições a atividades de melhoria desses atributos.

Destaca-se também que o modelo de avaliação proposto pelo WCAG (que resulta nas classificações A, AA ou AAA) não é suficientemente preciso para monitorar a acessibilidade (Vigo *et al.*, 2007), uma vez que uma página em conformidade com o WCAG em todas as diretrizes de prioridade 1 e quase todas de prioridade 2, teria a mesma classificação que uma página em conformidade com todas as diretrizes de prioridade 1 e nenhum de prioridade 2.

Métricas de acessibilidade, muitas vezes, utilizam diretrizes, como o WCAG, para calcular um valor quantitativo que represente o nível de acessibilidade de um sistema web (Vigo *et al.*, 2007). Como por exemplo, Sullivan e Matson (2000) propõem o cálculo da proporção entre o número de problemas de acessibilidade identificados em um determinado sistema web (considerando como problema todas as falhas de verificação dos pontos de verificação do WCAG 1.0) e o número de barreiras de acessibilidade potenciais para calcular a métrica de acessibilidade de uma página web.

Outros estudos que abordam a elaboração de métricas de acessibilidade descrevem a inclusão de pesos diferentes de acordo com pontos de verificação que possuam prioridades diferentes (Parmanto e Zeng, 2005); definição de valores de peso definidos com base em experimentos realizados com usuários com deficiência (Cluster, 2006); utilização da proporção do número de falhas de pontos de verificação semelhantes e o número total de falhas de verificação (Bühler *et al.*, 2006); entre outras abordagens.

A tarefa de realizar o monitoramento dos níveis de acessibilidade manualmente de sistemas web possui um custo elevado, principalmente quando se considera ambientes de desenvolvimento que realizam alterações e atualizações frequentemente (Watanabe, Fortes e Dias, 2012). Nesse contexto, Freire, Bittar e Fortes (2008), adaptaram a ferramenta de avaliação automática Hera para identificar as barreiras de acessibilidade em sistemas web e calcular as suas métricas de acessibilidade. Apesar dos métodos de avaliação automática não considerarem muitos aspectos que devem ser avaliados manualmente, os resultados desse trabalho mostram que, na média, os sistemas web apresentam um nível de barreiras médio a alto não desejável.

Em outro trabalho, Freire, Russo e Fortes (2008) utilizaram um questionário para investigar o conhecimento dos conceitos relacionados à acessibilidade na web por desenvolvedores web do Brasil. Identificar como os desenvolvedores percebem a acessibilidade, quão acessíveis são os sistemas web desenvolvidos por eles, e porque eles implementam sistemas web que são acessíveis ou não é um fator crítico para propor novas



abordagens que afetem significativamente a acessibilidade da web. Os resultados desse questionário apontaram que poucas pessoas que atuam no desenvolvimento web no Brasil conheciam os conceitos de acessibilidade.

É importante observar que em ambos os trabalhos (Freire, Bittar e Fortes, 2008; Freire, Russo e Fortes, 2008) são apresentados resultados de investigações sobre como a acessibilidade é considerada em projetos de desenvolvimento web e têm como foco as diretrizes da especificação WCAG 1.0 (W3C, 1999a). Observa-se que esses estudos não consideraram diretrizes da especificação WCAG 2.0 (W3C, 2008b), bem como princípios de usabilidade.

A gestão quantitativa dos processos é tratada nas áreas de processo do nível 4 do CMMI. Esse nível de gestão é conseguido quando as métricas não são apenas coletadas e usadas para prever esforços, prazos e riscos de novos projetos, mas para analisar os processos de forma quantitativa, com o objetivo de colocar sob controle estatístico o nível de qualidade dos produtos e o desempenho dos próprios processos (Paula Filho, 2009).

De acordo com Basili e Weiss (1984), as medições devem ser definidas de acordo com alguns objetivos específicos para melhorar a efetividade. Para tal, um modelo conhecido para a análise quantitativa que é bastante usado para definir, medir, analisar e melhorar os processos, é o GQM (*Goal-Question-Metric*).

GQM é uma abordagem de cima para baixo (*top-down*) para estabelecer um sistema de medição direcionado a metas para o desenvolvimento de sistema, em que a equipe começa com metas organizacionais, define a medição das metas, levanta questões a abordar os objetivos e identifica as métricas que proporcionem respostas às perguntas. Após a definição *top-down* do processo de medição, analisa-se *bottom-up* os dados resultantes (Basili e Weiss, 1984).

Algumas premissas para a medição são: prover resultados consistentes; permitir sua obtenção por não especialistas em informática; ser de fácil aprendizado; ser compreensível ao usuário final; servir para estimativas; permitir automatização; e possibilitar obter séries históricas. GQM define um modelo de medição, em três níveis, como ilustrado na Figura 17.

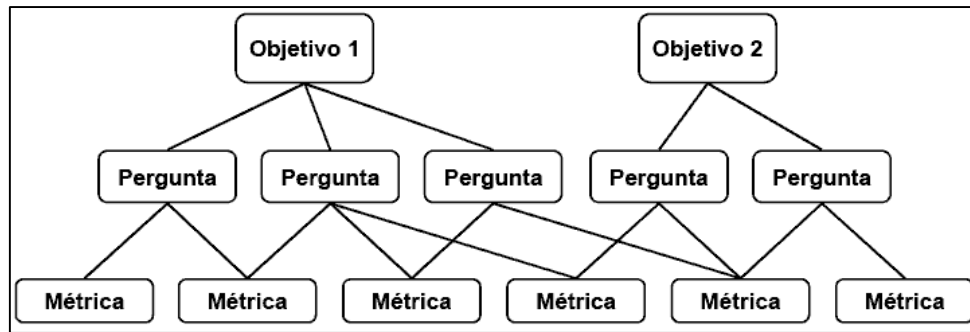


Figura 17: Modelo GQM.

De acordo com a Figura 17, é necessário listar os principais objetivos do processo de medição, derivar de cada objetivo as perguntas que devem ser respondidas para determinar se os objetivos foram alcançados e decidir o que precisa ser medido para ser capaz de responder as perguntas adequadamente. Um exemplo da aplicação do GQM pode ser encontrado na Figura 18.

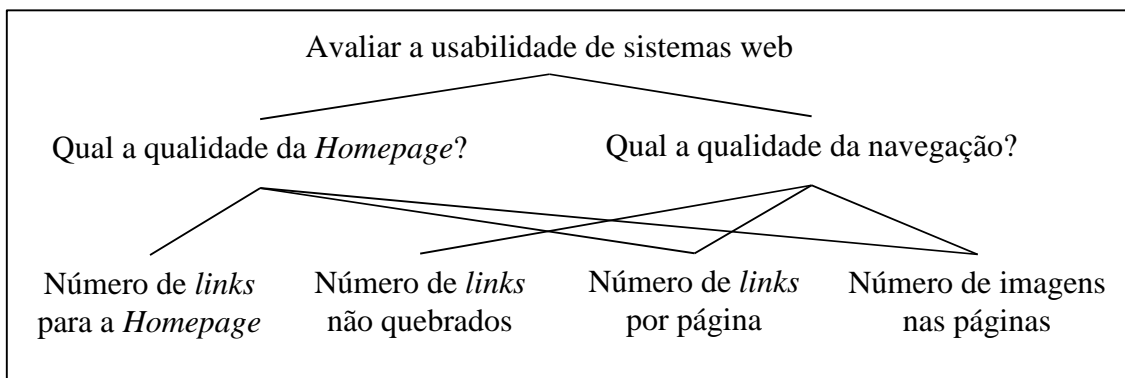


Figura 18: Exemplo de aplicação do GQM.

A utilização do GQM é importante para que as métricas sejam úteis, simples e diretas. Entretanto, as métricas não são definidas no nível de detalhamento necessário para garantir confiabilidade.

### 3.5 Considerações finais

Os conceitos que estão envolvidos na fundamentação teórica dos estudos desenvolvidos nesta tese foram apresentados neste capítulo. Os usos da engenharia web e de processos de desenvolvimento de software, entre outros estudos, foram utilizados como apoio para a formalização do processo de desenvolvimento de sistemas web com foco em acessibilidade e usabilidade.

Os próximos capítulos são apresentados seguindo a ordem em que a pesquisa foi desenvolvida. No Capítulo 4, o processo formalizado nesta tese é apresentado, bem como um estudo de caso em que um sistema web foi desenvolvido com o uso desse processo. Um experimento e um estudo de viabilidade, desenvolvidos para avaliar o processo proposto, são apresentados nos Capítulos 5 e 6, respectivamente.



---

# *Processo de desenvolvimento de sistemas web com foco em acessibilidade e usabilidade*

---

## *4.1 Considerações iniciais*

Como mencionado nos capítulos anteriores, uma questão crítica no desenvolvimento de sistemas web está relacionada à utilização das diretrizes de acessibilidade e dos princípios de usabilidade nos processos de desenvolvimento de sistemas web. Neste sentido, em um projeto web, é importante identificar quais, como e quando esses princípios e diretrizes devem ser considerados.

O problema de pesquisa que norteou esta tese é a necessidade de incluir acessibilidade e usabilidade no desenvolvimento de sistemas web. Apesar de serem muitas as opções existentes para apoiar essa inclusão nos sistemas web, falta apoio efetivo ao desenvolvedor para que em todo o processo de desenvolvimento de sistemas web a usabilidade e a acessibilidade sejam consideradas (Lazar e Olalere, 2011).

O objetivo é mostrar que existem técnicas apropriadas para fornecer suporte às ações inerentes ao projeto e à implementação, que podem ser selecionados, combinados

e tecnicamente implementados de modo a produzir um sistema web com acessibilidade e usabilidade. Sendo assim, nesta pesquisa propõe-se um processo, que é apresentado nesse capítulo, considerado aqui como um conjunto estruturado de práticas que pode ser repetível durante um processo, neste caso, durante o processo de desenvolvimento de software.

Este capítulo está organizado da seguinte maneira: inicialmente, na Seção 4.2, a fundamentação do processo é apresentada. Posteriormente, na Seção 4.3 é apresentado o processo. Na Seção 4.4, tem-se um estudo de caso com a aplicação do processo formalizado, bem como os resultados obtidos. Na Seção 4.5 o sistema desenvolvido no estudo de caso é apresentado e o reuso da interface desse sistema é apresentado na Seção 4.6. Os trabalhos relacionados ao processo formalizado podem ser encontrados na Seção 4.7. Por fim, na Seção 4.8, tem-se as contribuições finais.

## **4.2 Fundamentação do processo**

É importante retomar que a preocupação inicial desta pesquisa foi inserir acessibilidade durante todas as fases de desenvolvimento de sistemas web (ver Seção 1.3). Como percebeu-se que acessibilidade e usabilidade são conceitos fortemente relacionados, o termo usabilidade passou a ser integrado na pesquisa. As diretrizes de acessibilidade e os princípios de usabilidade foram tratados como requisitos não funcionais, transformados em tarefas e subtarefas no processo formalizado (ver Seção 4.3).

A concepção do processo envolveu a investigação da problemática (Dias *et al.*, 2010) e fundamentação de princípios teóricos (conceitos envolvidos), conceitos e trabalhos relacionados por meio de estudos sobre acessibilidade, usabilidade e engenharia web disponíveis na literatura (Capítulos 2 e 3). Toda essa fundamentação foi essencial para que sugestões fossem selecionadas, classificadas e inseridas no processo de desenvolvimento de sistemas web com foco em acessibilidade e usabilidade.

As sugestões selecionadas, bem como as referências de onde elas foram retiradas com uma possível classificação de atividades que fazem parte de todo processo de desenvolvimento de sistemas web, podem ser encontradas na Tabela 1.

Tabela 1: Fundamentação de princípios teóricos.

Atividades	Sugestões selecionadas	Referências
Comunicação	Considerar requisitos de acessibilidade e usabilidade no desenvolvimento de sistemas web.	Masuwa-Morgan, 2008. Baguma <i>et al.</i> , 2009. López, 2010. Maia, 2010.
	Categorizar usuários, cenários de utilização e objetivos dos usuários.	Brajnik, 2006. Brajnik, 2008a.
	Considerar os tipos de deficiências existentes nas categorias de usuário do sistema web a ser desenvolvido.	Sun e Zhang, 2009. Lazar e Olalere, 2011.
Planejamento	Considerar os riscos de acessibilidade e usabilidade (ex: os desenvolvedores muitas vezes não conseguem implementar essas diretrizes efetivamente).	Asakawa, 2005.
	Acompanhar o desenvolvimento dos requisitos do sistema por meio de uma ferramenta de monitoramento é essencial. Isso possibilita oferecer sugestões específicas e suporte efetivo ao desenvolvedor.	Bigham e Ladner, 2007.
Modelagem	Usar cenários, <i>storyboards</i> , observação e <i>workshop</i> em ciclos iterativos de elicitación de requisitos é importante.	Thew <i>et al.</i> , 2009.
	Separar a interface do usuário em várias camadas, podendo ser fundamentadas nas habilidades dos usuários, nas competências e nos níveis de experiência.	Ferreira e Nunes, 2008. Christiernin, 2010.
	Integrar a acessibilidade em análise de requisitos e especificação de atividades (ex: diagrama de caso de uso).	Baguma <i>et al.</i> , 2009.
	Quebrar uma longa lista de itens em sublistas para que as opções do sistema fiquem mais claras e fáceis de navegar (ex: menus).	Rubegni <i>et al.</i> , 2008.
Construção	Realizar testes com especialistas (inspeção).	Nielsen e Molich, 1990. Nielsen, 1993. Kelly <i>et al.</i> , 2005. Brajnik, 2006.

		Abou-zahra, 2008. Brajnik, 2008b. W3C, 2008a. Nietzio <i>et al.</i> , 2010.
	Realizar testes com ferramentas automáticas.	Takagi <i>et al.</i> , 2004. Abou-zahra, 2008.
	Realizar testes de aceitação.	Watanabe, Fortes e Dias, 2012.
Implantação	Realizar testes com usuários.	Melo e Baranaukas, 2006. Watanabe, 2007. Abou-zahra, 2008. Romen e Svanaes, 2008. Brajnik, Yesilada e Harper, 2010.
	Gerar um relatório com as modificações a serem realizadas na próxima iteração do processo, baseado nas indicações de usuários e especialistas.	Karat, 1993. Abou-zahra, 2008. O'Reilly, 2005 <i>apud</i> Gregg, 2010. Pressman e Lowe, 2009.

É importante ressaltar que o processo foi desenvolvido seguindo uma metodologia *top-down*, sendo criado e refinado evolutivamente, a partir dos resultados coletados por meio de revisões bibliográficas, de pesquisas empíricas, ações práticas (estudo de caso – desenvolvimento de um sistema web) e do *feedback* recebido na apresentação de trabalhos em conferências e reuniões em grupo.

O processo foi desenvolvido para ser utilizado de forma sistemática, iterativa e interativa, como se segue:

- Sistemático: segue uma sequência de ações que devem ser executadas, tendo entradas que devem ser consideradas e apresentando um conjunto de artefatos que devem ser produzidos como saída.
- Iterativo: o processo deve ser contínuo durante o desenvolvimento do sistema, na forma de atividades que devem ser executadas, de modo a separar e fundamentar práticas de desenvolvimento com acessibilidade e usabilidade para produzir um bom projeto.
- Interativo: as saídas de uma ação podem ser utilizadas como entradas das ações subsequentes, ou seja, as ações não são isoladas umas das outras.



Apesar de o processo ter passado por uma evolução contínua, melhorado à medida que novos achados eram identificados e refinamentos realizados, não há garantias de que o processo ainda não deva ser evoluído. No entanto, permite ter uma base sólida em relação à sua construção sobre os aspectos de acessibilidade e usabilidade, visto que foi desenvolvido sobre resultados de estudos que tinham esse tema como foco principal.

### **4.3 O processo PDWAU**

O processo de desenvolvimento de sistemas web com acessibilidade e usabilidade foi denominado PDWAU<sup>8</sup> - *Process for Developing Web system with Accessibility and Usability*. Para o desenvolvimento do PDWAU, utilizou-se o processo de desenvolvimento de sistemas web proposto por Pressman e Lowe (2009), somado ao novo paradigma de sistemas web (O'Reilly, 2005 *apud* Gregg, 2010), o beta perpétuo (conta-se com a intervenção do usuário para apontar possíveis falhas), princípios de usabilidade e diretrizes de acessibilidade, bem como outros processos, métodos, técnicas e ferramentas encontrados na literatura, apresentados na Tabela 1.

O *framework*/processo genérico de engenharia web (Pressman e Lowe, 2009) foi adotado, pois contém atividades básicas, fundamentais e bem definidas que podem guiar facilmente os desenvolvedores em suas tarefas para qualquer processo de desenvolvimento de sistemas web. Além disso, esse processo é relativamente novo e atual na área de engenharia web, herda características da engenharia de software e engloba todas as atividades que outros processos e métodos consideram como fundamentais. Esse processo também é independente de linguagem, facilitando seu uso pelos desenvolvedores independentemente de plataforma.

Como apresentado anteriormente, na Figura 4 (Seção 3.2), as cinco atividades desse processo genérico são: comunicação, planejamento, modelagem, construção e implantação. Como ações, tem-se: formulação, elicitação, negociação, estimação, análise de risco, escalonamento, monitoração, análise, projeto, codificação, testes, entrega e avaliação. No processo proposto nesta tese, foram inseridas novas tarefas e subtarefas nas ações existentes. A versão atual do processo PDWAU formalizado pode ser visualizada na Figura 19.

---

<sup>8</sup> Lê-se Pedal.

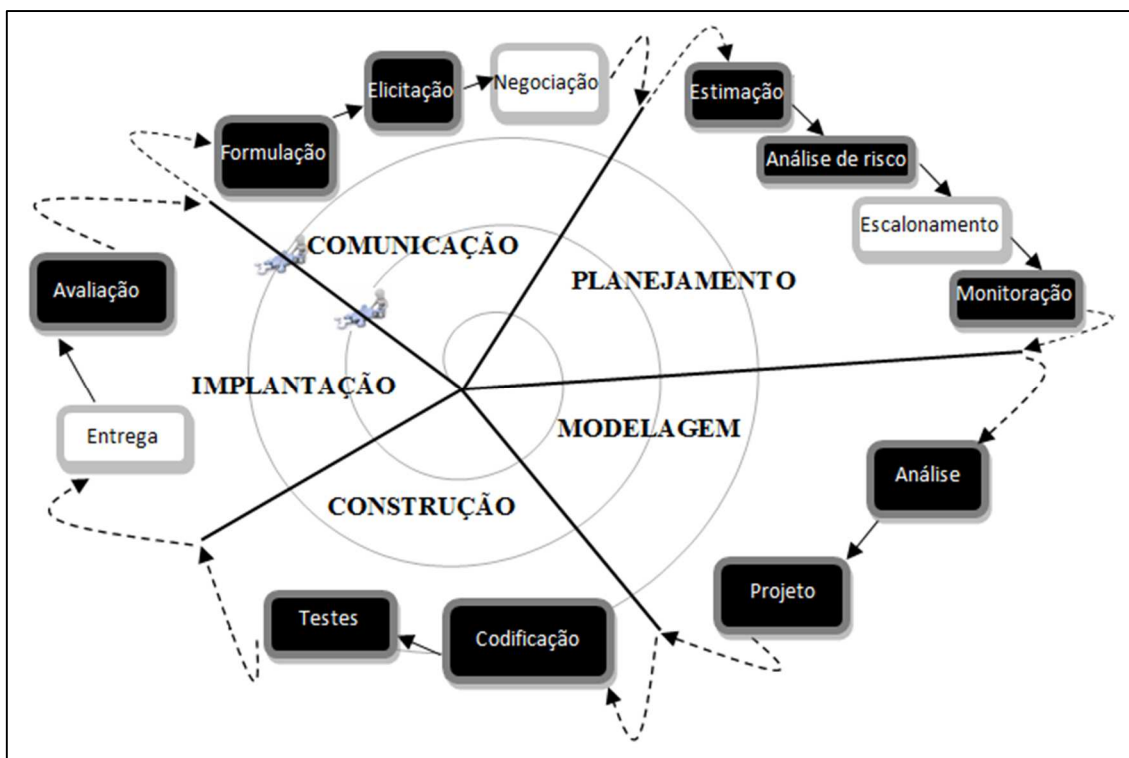


Figura 19: Atividades e ações do PDWAU.

Na Figura 19 estão hachuradas as ações que tiveram suas tarefas alteradas no processo genérico de engenharia web formalizado por Pressman e Lowe (2009). As ações negociação, escalonamento e entrega, não foram modificadas e, portanto, permaneceram em branco com as bordas em cinza e não serão detalhadas a seguir. É importante ressaltar que para que os especialistas consigam usar o PDWAU, é necessário seguir também o livro de Pressman e Lowe (2009) para acompanhar as ações genéricas com mais detalhes.

Para avaliar o processo PDWAU foi realizado um estudo de caso (ver Seção 4.4) em que desenvolveu-se o sistema AgendAloca, um sistema web de agendamento de bancas que será detalhado na Seção 4.5. Antes disso, nas subseções a seguir, serão apresentados os detalhes de todas as tarefas e subtarefas sobre acessibilidade e usabilidade que foram inseridas no processo genérico.

As novas tarefas de usabilidade e acessibilidade estão destacadas com **dois asteriscos**. Elas foram consideradas novas tarefas por gerarem novos artefatos, antes não considerados. Já os destaques com **um asterisco** são modificações em tarefas existentes (subtarefas) que ao invés de gerarem algum artefato novo, simplesmente alteram os artefatos já gerados por tarefas originais do processo genérico.

### 4.3.1 Comunicação

Conforme apresentado na Figura 19, o processo da engenharia web começa com a atividade de comunicação, na qual os engenheiros web e os interessados se engajam em uma série de ações que:

- Perguntam e respondem a um conjunto de questões fundamentais sobre o sistema web<sup>9</sup> e seu contexto de negócios;
- Elicitam requisitos que servirão de base para todas as ações seguintes; e
- Negociam as necessidades contra as realidades de tempo, recursos e tecnologia.

As ações dessa atividade são chamadas de formulação, elicitação (também conhecida como levantamento de requisitos) e negociação, respectivamente. As tarefas e subtarefas dessas tarefas são apresentadas a seguir, na Figura 20.

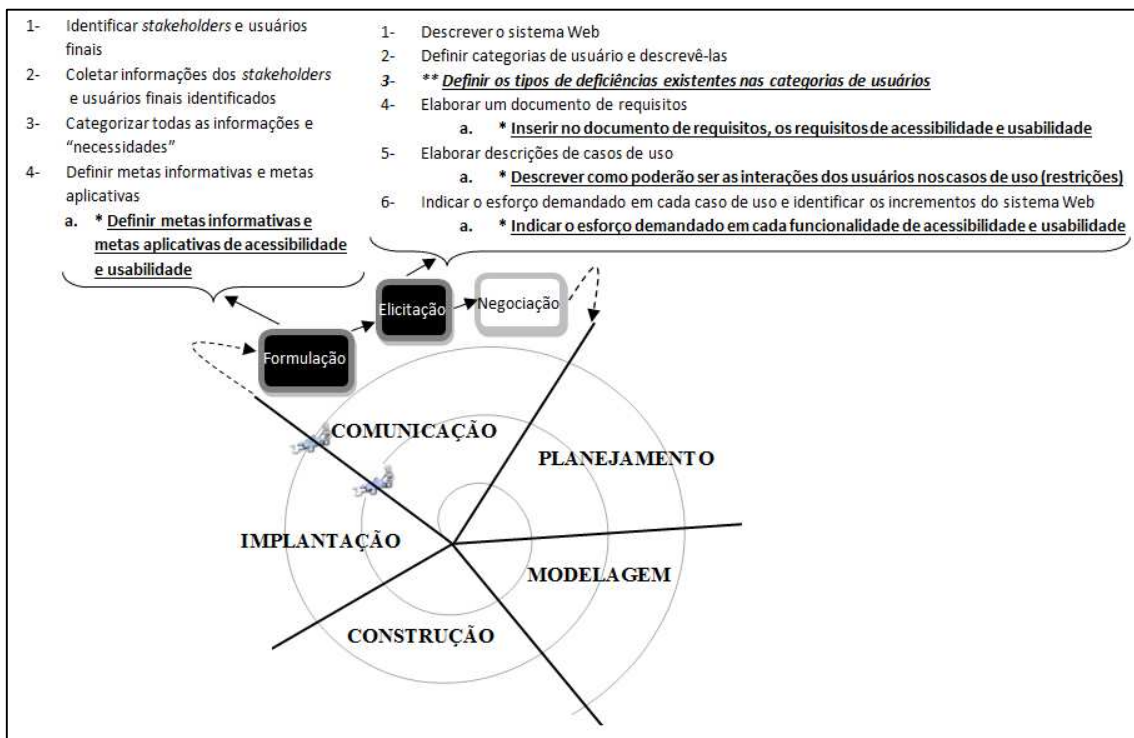


Figura 20: Tarefas da atividade de comunicação do PDWAU.

Na ação de formulação, uma subtarefa foi adicionada. Na ação elicitação, uma tarefa foi adicionada e três novas tarefas foram adicionadas. Essas ações que foram modificadas e serão apresentadas e detalhadas a seguir.

<sup>9</sup> Para o processo PDWAU, foi adotado o termo "sistema web" para o termo "webApp" indicado no livro de Pressman e Lowe (2009).

#### 4.3.1.1 Formulação

A ação de formulação começa com a identificação de uma necessidade de negócio, passa por uma descrição dos objetivos do sistema web, as principais características e estabelece uma ação para a elicitación de requisitos. Esta ação é composta por cinco tarefas, que são:

- 1) Identificar interessados (*stakeholders*) e usuários finais (qualquer um que se beneficia de uma maneira direta ou indireta do sistema que será desenvolvido).
- 2) Coletar informações dos interessados identificados, utilizando técnicas tais como:
  - a) Grupo de foco tradicional (discutir o sistema web);
  - b) Grupo de foco eletrônico (discussão eletrônica moderada);
  - c) Pesquisas iterativas (pesquisas breves sobre questões específicas, por meio de sistemas web ou correio eletrônico);
  - d) Pesquisa exploratória (pesquisa vinculada a sistemas web que possuem usuários semelhantes aos que usarão o sistema web a ser desenvolvido); e
  - e) Criação do cenário (interessados criam cenários de uso que descrevam interações específicas com o sistema web).
- 3) Categorizar todas as informações e “necessidades” dos interessados (incluindo ideias inconsistentes e em conflito).
- 4) Definir metas informativas (indicam a intenção de fornecer conteúdo específico e/ou informação para o usuário final) e metas aplicativas (indicam a capacidade de realizar alguma tarefa dentro do sistema web).
  - a) \* **Definir metas informativas e metas aplicativas de usabilidade e acessibilidade**<sup>10</sup>.

---

<sup>10</sup> Essa foi uma sub tarefa inserida no processo genérico de Pressman e Lowe (2009) e que agora faz parte do processo PDWAU, conforme indicado na Figura 20.

Ressalta-se que as novas tarefas e sub tarefas sugeridas nesta pesquisa para a formalização do PDWAU serão apresentadas em negrito e asteriscos, assim como as Figuras (20, 21, 22, 23 e 24) indicativas do processo, a fim de facilitar o entendimento do leitor.

#### 4.3.1.2 Elicitação

O objetivo dessa ação é identificar um conjunto de requisitos que focalize o conteúdo do sistema web, a função, a interação do usuário e a interoperabilidade, tais como os sistemas de negócios e banco de dados. Esta ação é composta pelas seguintes tarefas:

- 1) Descrever o sistema web (usando as informações derivadas da ação formulação – ver Subseção 4.3.1.1).
- 2) Definir categorias de usuário e desenvolver descrições para cada categoria.
- 3) **\*\* Definir os tipos de deficiências existentes nas categorias de usuários já definidas (utilizar o APÊNDICE A).**
- 4) Elaborar um documento de requisitos contendo requisitos funcionais e não funcionais do sistema web.
  - a) **\* Inserir no documento de requisitos os requisitos de usabilidade e acessibilidade (consultar o APÊNDICE B).**
    - I) **Adicionar os requisitos que devem ser atendidos pelo sistema de acordo com os tipos de deficiências existentes definidas na tarefa 3;**
    - II) **Justificar a exclusão dos requisitos não selecionados.**

**Obs: O Sistema deve considerar as atualizações das recomendações do W3C e também do eMAG, evitando assim, a utilização de tecnologias obsoletas. Dessa maneira, o desenvolvedor não deve se pautar somente no APÊNDICE B, mas tê-lo como um apoio inicial.**
- 5) Elaborar descrições de casos de uso para cada categoria de usuário.
  - a) **\* Descrever como poderão ser as interações dos usuários nos casos de uso desenvolvidos, de acordo com as restrições dos usuários (no campo canal para ator).**
- 6) Indicar o esforço demandado por cada caso de uso definido e identificar os incrementos do sistema web, indicando os casos de uso que serão desenvolvidos.
  - a) **\* Indicar o esforço demandado para cada funcionalidade de acessibilidade e usabilidade, indicando as funcionalidades que serão desenvolvidas.**

### 4.3.1.3 Negociação

Nesta ação, as funcionalidades e requisitos não funcionais são equilibrados com custo e tempo para entrega. A intenção dessa negociação é atender às necessidades do cliente enquanto, ao mesmo tempo, procuram atender às restrições do mundo real (tempo, pessoal e orçamento, por exemplo) que foram impostas.

**Obs: Nesta ação, a alteração poderia ser na verificação de apoios (frameworks, tecnologias, padrões, pessoal especializado etc.) para inserir usabilidade e acessibilidade, que possam interferir no custo e tempo de entrega do sistema.**

### 4.3.2 Planejamento

Esta atividade só pode ocorrer depois que os requisitos básicos do sistema web tiverem sido identificados, ou seja, após a realização da atividade de comunicação. A atividade de planejamento leva a um conhecimento do contexto de negócios, objetivos informativos, funcionalidades do sistema web, restrições do sistema e questões de desempenho, focalizando as tarefas e produtos de trabalho a serem produzidos para um incremento específico, bem como os mecanismos de gerenciamento de risco, qualidade e mudança que serão aplicados à medida que cada incremento é produzido.

Nesta atividade deve ser estabelecido um plano incremental para o trabalho do sistema web. Ele descreve as ações do sistema web que ocorrerão, as tarefas técnicas a serem realizadas, os riscos prováveis, os recursos que serão exigidos, os produtos de trabalho a serem produzidos e um cronograma de trabalho. Para isso, são necessárias as ações: estimação, análise de risco, escalonamento e monitoração.

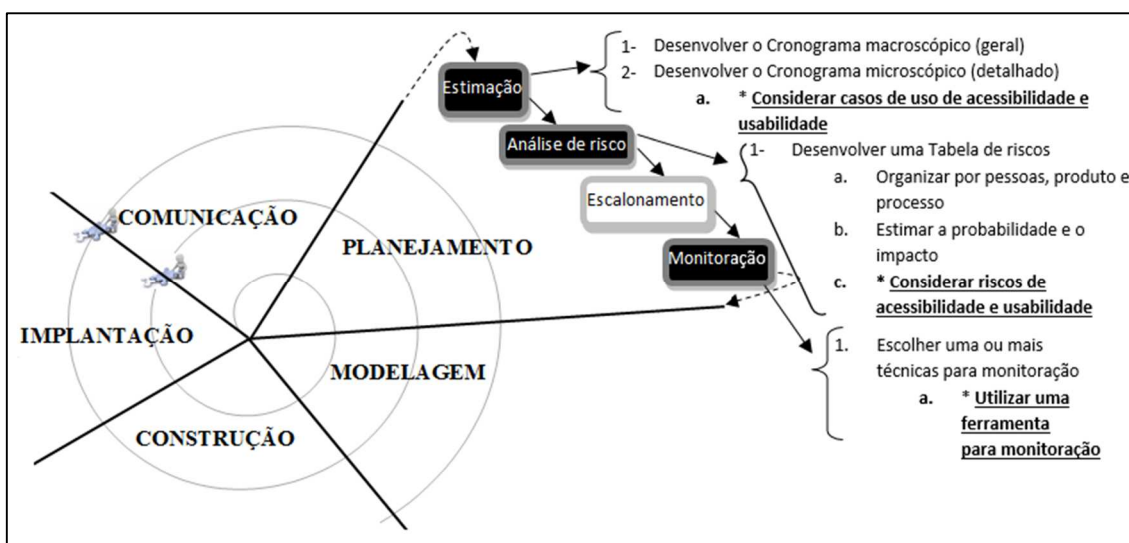


Figura 21: Tarefas da atividade de planejamento do PDWAU.

Como apresentado na Figura 21, a ação escalonamento não foi modificada. Já as ações estimação, análise de risco e monitoração foram modificadas e serão detalhadas a seguir.

#### 4.3.2.1 Estimação

Nesta ação deve ser desenvolvido um cronograma de projeto do sistema web, identificando todos os incrementos, projetando datas em que cada um será implantado. Esta ação tem as seguintes tarefas:

- 1) Cronograma de implantação preliminar macroscópico (geral) para todos os incrementos identificados na atividade anterior (comunicação). Ressalta-se que alguns dos incrementos podem ser desenvolvidos em paralelo.
- 2) Cronograma de incremento microscópico (detalhado), medindo o esforço (em homens-dias) e o tempo de desenvolvimento (em dias - esforço). Ele é desenvolvido baseando-se nos casos de uso definidos para os incrementos a serem construídos. Da mesma maneira que os incrementos, os casos de uso também podem ser desenvolvidos em paralelo.
  - a) \* **Considerar os casos de uso de usabilidade e acessibilidade (definidos na tarefa 5a da ação de elicitação da atividade de comunicação).**

#### 4.3.2.2 Análise de risco

Um risco é um problema em potencial, que pode ou não acontecer. É importante identificá-lo, avaliar sua probabilidade de ocorrência, estimar seu impacto e estabelecer um plano de contingência caso o problema realmente ocorra. O gerenciamento de risco compreende várias tarefas que ajudam uma equipe a entender e gerenciar os muitos problemas que podem acontecer em um projeto de sistema web. Esta ação tem as seguintes tarefas:

- 1) Criar uma tabela de riscos, contendo:
  - a) Lista de riscos, organizada em:
    - I) Riscos de pessoas: são problemas em potencial que podem ser rastreados diretamente em alguma ação ou falha humana;
    - II) Riscos de produto: são problemas em potencial associados ao conteúdo, funções, restrições ou desempenho do sistema web;
    - III) Riscos de processo: são problemas que estão ligados às ações e tarefas de arcabouço que foram escolhidas pela equipe.

- b) Avaliação dos riscos encontrados a fim de estimar a probabilidade e o impacto potencial, de duas maneiras:
- I) A probabilidade de que o risco se torne uma realidade;
  - II) As consequências dos problemas associados ao risco, caso ele ocorra (variando de 1 a 4).
- c) \* **Na lista de riscos, considerar riscos de usabilidade e acessibilidade por parte de pessoas, produtos e até mesmo de processo.**

#### 4.3.2.3 Escalonamento

A equipe de desenvolvimento seleciona o conjunto de ações e tarefas que são mais apropriadas para incremento do sistema web a ser desenvolvido.

**Obs: Esta ação não foi modificada, portanto não será detalhada.**

#### 4.3.2.4 Monitoração

Para um projeto de engenharia web, a implantação de um incremento de um sistema web normalmente é a principal medida do progresso geral. Mas, antes que o incremento esteja disponível, é necessário saber “onde se está” no desenvolvimento do sistema. Para isso, algumas técnicas podem ser utilizadas nesta ação:

- 1) Escolher uma ou mais técnicas para monitorar o desenvolvimento do sistema web.
  - Realizar uma enquete com a equipe de engenharia web para determinar quais atividades do processo já foram completadas.
  - Determinar quantos cenários de uso foram implementados e quantos cenários do usuário (para determinado incremento) restam para ser implementados.
  - Se a equipe tiver construído um cronograma de trabalho detalhado para cada incremento, o progresso pode ser acompanhado determinando-se quantas tarefas de trabalho foram completadas, quantos produtos de trabalho foram produzidos e revisados, e quão confiantes os membros da equipe estão na entrega na data de término do incremento.



- a) **\* Definir e utilizar uma ferramenta<sup>11</sup> em que os próprios envolvidos no desenvolvimento do sistema web possam inserir os casos de uso/requisitos de usabilidade e acessibilidade na monitoração do desenvolvimento para que não seja necessário gastar tempo e recursos no final de cada incremento e/ou projeto para inseri-las. Adicionalmente, o gerente de projeto e todos os outros envolvidos poderão acompanhar o desenvolvimento do sistema web.**

### 4.3.3 Modelagem

A atividade de modelagem ajuda a resolver problemas de ambiguidade da linguagem natural usando uma combinação de textos, gráficos e formas diagramáticas para representar conteúdo e função, arquitetura e detalhe do componente, interfaces, navegação e estética de maneira que são relativamente fáceis de entender e simples de revisar a exatidão, completude e consistência. Adicionalmente, um modelo eficaz permite que desenvolvedores e analistas de negócios vejam como um sistema web, sua arquitetura ou sua tecnologia de suporte poderiam impor restrições sobre uma organização de negócios.

É importante ressaltar que há várias maneiras de modelar a análise e o projeto de um sistema web, que se relacionam a diferentes pontos de vista. É necessário optar por modelagens que serão úteis para o processo em desenvolvimento. Duas ações são necessárias nessa atividade: análise e projeto.

Na ação de análise, é necessário detalhar, estruturar e validar os requisitos em termos de um modelo conceitual do problema. Os analistas devem fornecer informações suficientemente detalhadas para que o projetista elabore um projeto tecnologicamente bom, evitando a ambiguidade e reduzindo a manutenção corretiva das especificações do sistema.

Na ação de projeto, é necessário formular um modelo estrutural do sistema, que sirva de base para a implementação. O projetista deve fornecer informações suficientes para que o analista possa dizer se os requisitos dos usuários podem ser completamente atendidos ou devem ser modificados.

---

<sup>11</sup> Um exemplo de ferramenta que pode ser utilizada para esse propósito seria o Redmine (<http://www.redmine.org/>)

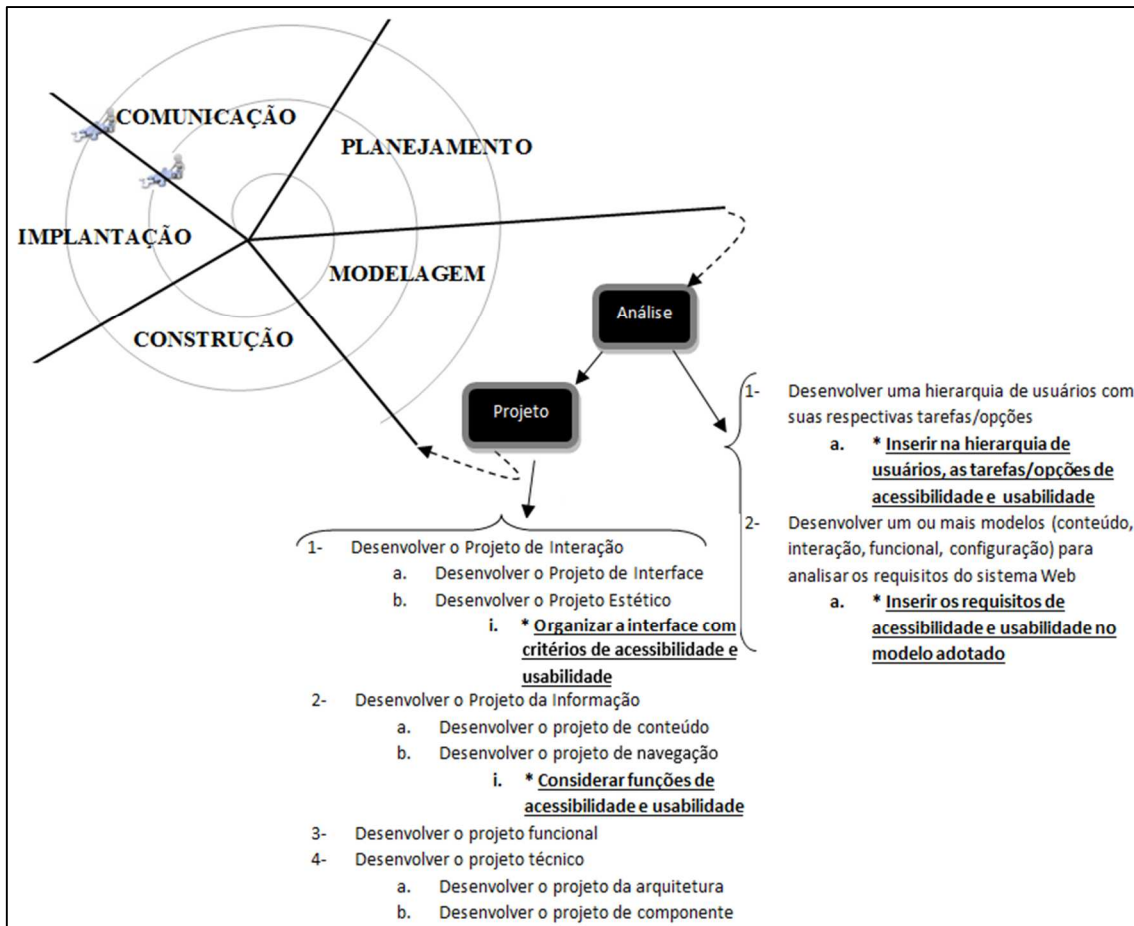


Figura 22: Tarefas da atividade de modelagem do PDWAU.

Conforme indicado na Figura 22, as ações de análise e projeto foram modificadas e serão detalhadas a seguir.

#### 4.3.3.1 Análise

Nesta ação, os requisitos dos interessados são examinados em um esforço para entender o que o sistema web deve fazer.

- 1) Desenvolver uma hierarquia de usuários, com suas respectivas tarefas/opções. Para isso, revisar a categoria de usuários identificada na tarefa 1 da ação formulação da atividade de comunicação. Essa tarefa pode ajudar a equipe de desenvolvimento a compreender melhor os relacionamentos entre as diferentes categorias de usuários e o próprio sistema web. Uma hierarquia de usuários pode também ser importante quando se define conjuntos de funcionalidade coerentes e o conteúdo que poderia ser visível a diferentes usuários.

- a) \* **Inserir na hierarquia de usuários, as tarefas/opções de usabilidade e acessibilidade.**
- 2) Desenvolver um ou mais modelos para analisar os requisitos do sistema web de uma maneira estruturada:
- Modelo de conteúdo: identifica o espectro completo do conteúdo a ser fornecido pelo sistema web. O conteúdo inclui dados de texto, gráficos e imagens, áudio/vídeo e pode utilizar diagramas de troca de informações da web.
  - Modelo de interação: descreve a maneira como os usuários interagem com o sistema web. Pode ser composto de um ou mais dos seguintes elementos: casos de uso, diagramas de sequência, diagramas de estado e/ou protótipos da interface do usuário.
  - Modelo funcional: define as operações que serão aplicadas ao conteúdo do sistema web e descreve outras funções de processamento que são independentes do conteúdo, mas necessárias para o usuário final. Pode ser composto pelo diagrama de atividade, quando a funcionalidade a ser desenvolvida for relativamente complexa.
  - Modelo de configuração: descreve o ambiente e a infraestrutura em que o sistema web reside. O diagrama de implantação pode ser utilizado em situações em que arquiteturas de configuração complexas devem ser consideradas.
- a) \* **Inserir os requisitos de usabilidade e acessibilidade do sistema web de uma maneira estruturada no modelo adotado.**

#### 4.3.3.2 Projeto

A modelagem do projeto trata de compreender a estrutura interna do sistema web sendo desenvolvido e com isso cria a forma do sistema web que foi identificada pela modelagem da ação de análise.

- 1) Desenvolver o projeto de interação que forma a base da interação do usuário.
- a) Desenvolver o projeto de interface: descreve a estrutura e a organização da interface com o usuário, incluindo uma representação do layout da tela, uma definição dos modos de interação e uma descrição dos mecanismos de navegação.

- b) Desenvolver o projeto estético: também chamado de projeto gráfico, descreve a aparência e o estilo do sistema web. Incluem esquemas de cor, leiaute geométrico, tamanho de texto, fonte e posicionamento, o uso de figuras gráficas e decisões estéticas relacionadas.
  - I) \* **Organizar a interface das categorias de usuários de acordo com requisitos de usabilidade e acessibilidade, priorizando as deficiências existentes. (Para isso, podem-se consultar as dificuldades de cada deficiência no APÊNDICE A).**
- 1) Desenvolver o projeto da informação que forma a base do conteúdo que sustenta o sistema web, o modo como esse conteúdo é organizado e a maneira como o usuário trafega pelo sistema para acessar o conteúdo ou uma funcionalidade.
  - a) Desenvolver o projeto de conteúdo: define o leiaute, a estrutura e o esboço para todo o conteúdo que é apresentado como parte do sistema web. Ele estabelece os relacionamentos entre os objetos de conteúdo.
  - b) Desenvolver o projeto de navegação: representa o fluxo de navegação entre os objetos de conteúdo e para todas as funções do sistema web. Ele descreve os modos como o usuário localiza e interage com o conteúdo.
    - I) \* **Considerar funções de usabilidade e acessibilidade que o sistema web terá. Essas funções são baseadas nos requisitos de usabilidade e acessibilidade (selecionados na ação de elicitação da atividade de comunicação, com o apoio do APÊNDICE B).**
- 2) Desenvolver o projeto funcional, que identifica o comportamento geral e a funcionalidade que é provida pelo sistema web, incluindo aspectos como apoio ao fluxo de trabalho, adaptação e/ou personalização do conteúdo e da interface, entrada de pedido, processos de banco de dados, funções computacionais e outros.
- 3) Desenvolver o projeto técnico que trata tanto do projeto da arquitetura quanto do projeto dos componentes do sistema web. Além disso, oferece uma base para a atividade de construção.
  - a) Desenvolver o projeto da arquitetura: identifica a estrutura geral para o sistema web.

- b) Desenvolver o projeto de componente: desenvolve a lógica de processamento detalhada exigida para implementar componentes funcionais que dão apoio a um conjunto completo de funções do sistema web.

#### 4.3.4 Construção

Ferramentas e tecnologias que são aplicadas para desenvolver o sistema web. É nesta atividade que os componentes definidos pelo incremento são construídos (codificados e testados). Após os incrementos, testes rápidos são realizados para garantir que os erros no projeto (conteúdo, arquitetura, interface, navegação, entre outros) sejam descobertos.

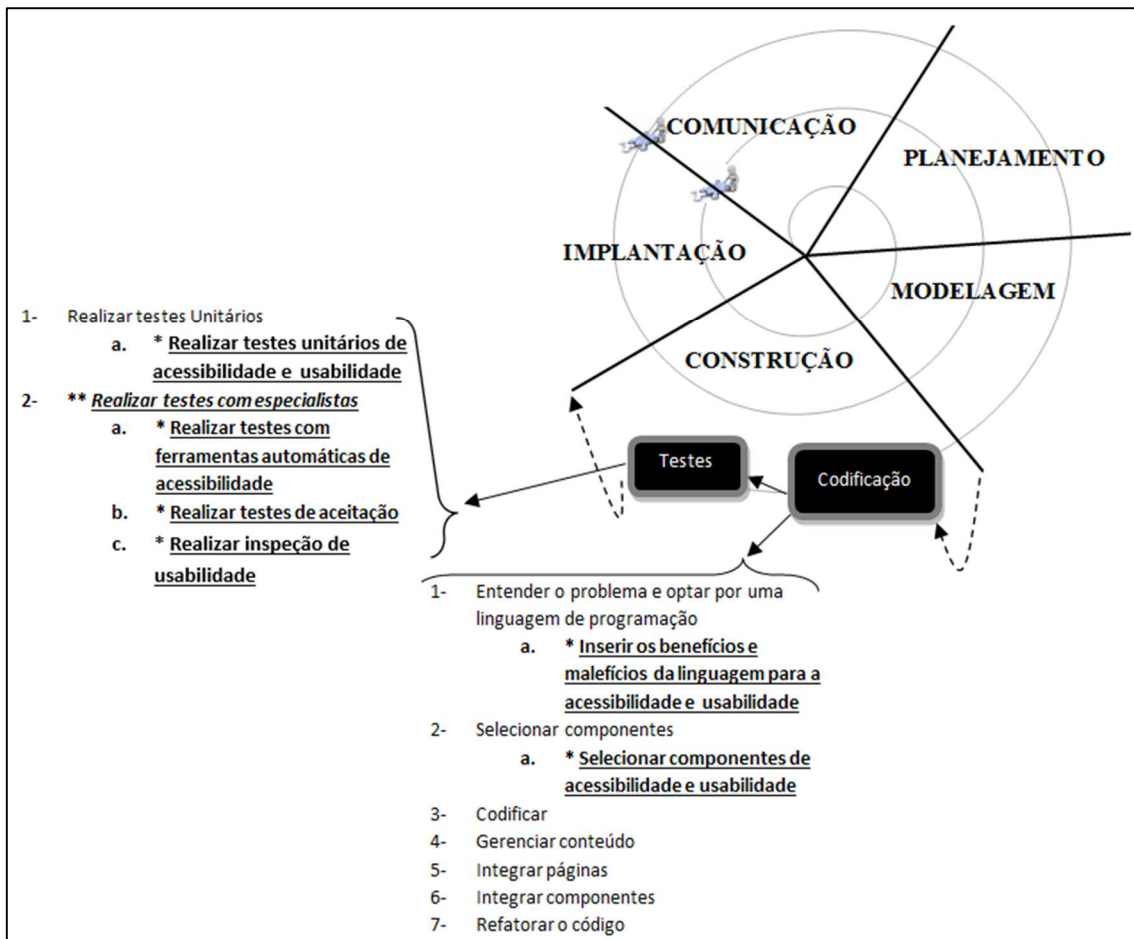


Figura 23: Tarefas da atividade de construção do PDWAU.

Conforme apresentado na Figura 23, as ações para a atividade de construção são: codificação e testes. Ambas foram modificadas e serão detalhadas a seguir.

#### 4.3.4.1 Codificação

Na ação de codificação, é o momento de traduzir a representação do projeto detalhado em termos de uma linguagem de programação. Para isso, algumas tarefas e subtarefas são necessárias:

- 1) Entender o problema e optar por uma linguagem: nesta tarefa é necessário entender bem o problema que se está tentando solucionar, juntamente com os princípios e os conceitos básicos do projeto do sistema web. É preciso optar por uma linguagem que atenda às necessidades do sistema web e que ofereça opções que tornarão o trabalho dos desenvolvedores mais fácil. Pode-se também criar um conjunto de testes unitários que serão aplicados quando os componentes do sistema web estiverem completos.
  - a) \* **Inserir quais são os benefícios e malefícios que as linguagens selecionadas trarão para a usabilidade e acessibilidade.**
- 2) Selecionar componentes: envolve a identificação de componentes relevantes preexistentes que possam ser reutilizados dentro do projeto proposto. É importante também considerar as habilidades e o conhecimento dos desenvolvedores e prováveis mantenedores.
  - a) \* **Selecionar também componentes de usabilidade e acessibilidade que já tenham sido desenvolvidas e que tenham obtido bons resultados.**
- 3) Codificar: cobre a adaptação de componentes existentes ou a criação de novos componentes, e pode envolver a criação direta de HTML ou código fonte em linguagem de script ou a geração automática de código usando uma representação de projeto intermediário do componente a ser construído.
- 4) Gerenciar conteúdo: quando o conteúdo for gerenciado, é necessário certificar-se de selecionar estruturas de dados que atendam às necessidades do projeto, garantindo a coerência nos formatos e estruturas de dados.
- 5) Integrar páginas: envolve a integração do conteúdo bruto com o projeto gráfico (leiaute) e o mapeamento do conteúdo nas telas e páginas. Neste momento devem-se considerar continuamente questões de usabilidade e acessibilidade.
- 6) Integrar componentes: compreende o vínculo do código, conteúdo e apresentação nos componentes finais a serem liberados. Quando houver integração dos componentes, é necessário certificar-se de manter cópias de

segurança (*backups*) – preferencialmente em alguma forma de controle de versão; procurar divergências ou inconsistências na interface do componente e; identificar componentes que precisam de refatoração.

- 7) Refatorar o código: é uma ação iterativa para “polir” os componentes implantados a fim de melhorar sua estrutura e remover o código redundante.

#### 4.3.4.2 Testes

O objetivo da ação de testes é encontrar defeitos no sistema, para que eles possam ser corrigidos pela equipe de desenvolvedores, antes da entrega para o usuário, ou seja, verificar por meio de uma execução controlada se o comportamento ocorre de acordo com o especificado. Para isso, algumas tarefas são necessárias:

- 1) Teste unitário: envolve a verificação de que os diversos componentes e objetos estão corretos. É necessário realizar testes unitários e corrigir os erros encontrados, a partir de testes que mais provavelmente localizarão erros, em vez de ocultá-los.
  - a) \* Teste unitário de acessibilidade e usabilidade: envolve a verificação do uso de TAGS HTML, que influenciam a navegação dos usuários.
- 2) \*\* Testes com especialistas: neste momento é necessário não só realizar testes de código, mas a partir do momento em que requisitos de acessibilidade e usabilidade foram inseridos no documento de requisitos, é necessário testá-los por meio de vários testes.
  - a) \* Testes com ferramentas automáticas de acessibilidade<sup>12</sup>: Os avaliadores ou validadores de acessibilidade são ferramentas automáticas que fazem uma pesquisa no código de uma página emitindo relatórios nos quais indicam os erros de acessibilidade segundo as prioridades sugeridas nas diretrizes para a acessibilidade dos conteúdos da web.
  - b) \* Testes de aceitação: Um teste formal que determina se um sistema satisfaz o critério de aceitação por meio de asserções e verificações realizadas na interface externa.

---

<sup>12</sup> Como ferramentas automáticas de avaliação de acessibilidade, pode-se citar: DaSilva e Hera. Uma lista completa dos avaliadores cadastrados na W3C pode ser encontrada em:

<http://www.w3.org/WAI/ER/tools/complete>

- c) \* Inspeção de usabilidade: São realizados para detectar problemas de navegação e de interface. Algumas técnicas podem ser: avaliação heurística; inspeção baseada em padrões, inspeção por *checklists*, etc.

### 4.3.5 Implantação

O sistema web é entregue aos usuários finais e um período de avaliação é iniciado. O *feedback* da avaliação é apresentado à equipe que participou do desenvolvimento do sistema web e o incremento é modificado conforme a necessidade. Como o desenvolvimento de sistemas web é incremental por natureza, a implantação acontece não uma vez, mas diversas vezes enquanto o sistema web se encaminha para o término de seu desenvolvimento.

Cada ciclo de liberação de pacote oferece aos usuários finais um incremento operacional do sistema web que oferece funções e recursos úteis. Cada ciclo de avaliação oferece à equipe do sistema web orientações importante que resultam em modificações no conteúdo, funções, características e enfoque usados para o próximo incremento.

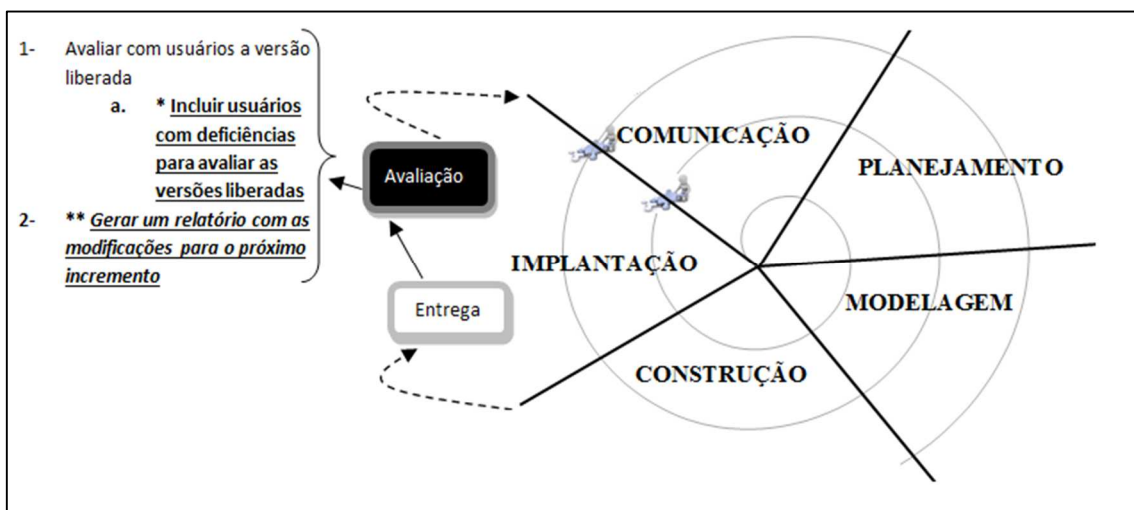


Figura 24: Tarefas da atividade de implantação do PDWAU.

Conforme apresentado na Figura 24, a atividade de implantação compreende das ações de entrega e avaliação. Somente a ação de avaliação foi modificada e será detalhada a seguir.

#### 4.3.5.1 Entrega

É o momento de empacotar os componentes (inclui identificar o conjunto de componentes a liberar para o servidor de produção e as orientações sobre quando e como essa liberação poderá ocorrer) e liberar as versões do sistema (deve-se considerar algumas



questões, são elas: como as mudanças serão comunicadas ou anunciadas? Como os usuários serão orientados pelo uso de conteúdo ou funcionalidade complexa?).

**Obs: Esta ação não foi modificada, portanto não será detalhada.**

#### 4.3.5.2 Avaliação

Na ação de avaliação, o sistema já se encontra disponível aos usuários. Nesse momento, é preciso identificar as qualidades relacionadas com a interação entre o usuário e o sistema. Para isso, é necessário:

- 1) Avaliar com usuários a versão liberada: à medida que um incremento é colocado em uso, os usuários finais devem ser encorajados a comentar sobre os recursos e funções, facilidade de uso, confiabilidade e quaisquer outras características que sejam apropriadas. O *feedback* deverá ser coletado e registrado pela equipe de desenvolvimento e usado para:
  - Fazer modificações imediatas no incremento entregue (se for necessário);
  - Definir mudanças a serem incorporadas no próximo incremento planejado;
  - Fazer modificações de projeto necessárias para acomodar as mudanças;
  - Revisar o plano (incluindo o cronograma de entrega) para o próximo incremento, para refletir as mudanças.
- a) \* **Incluir usuários com deficiências para avaliar as versões liberadas.**
- 2) \*\* **Gerar um relatório com as modificações necessárias que foram percebidas pelos usuários durante a interação com o sistema web.**

Com a formalização do processo PDWAU, é possível perceber que o processo genérico de Pressman e Lowe (2009) foi modificado com o acréscimo de tarefas e subtarefas de acessibilidade e usabilidade, conforme indicado na Figura 19 no início dessa seção. O número de atividades e ações permaneceu o mesmo, mas a quantidade de tarefas e subtarefas foram modificadas.

Assim, conforme indicado na Tabela 2, com o processo PDWAU foram adicionadas três tarefas e dezoito subtarefas ao processo genérico, relacionadas especialmente a questões de acessibilidade e usabilidade. Ressalta-se que as ações negociação, escalonamento e entrega (mantidas na cor cinza – Figura 19) não tiveram suas tarefas e subtarefas modificadas do processo genérico de Pressman e Lowe (2009), portanto não estão contabilizadas na Tabela 2.

**Tabela 2: Quantidade de tarefas e subtarefas adicionadas ao processo genérico.**

	<b>Processo genérico (Pressman e Lowe, 2009)</b>	<b>Processo PDWAU</b>
<b>Tarefas</b>	28	+ 3
<b>Subtarefas</b>	8	+ 18

Adicionalmente, como mencionado anteriormente, durante a formalização do PDWAU, um sistema de agendamento de bancas foi desenvolvido e quatro iterações do processo foram executadas. A cada iteração, o processo foi sendo consolidado e aperfeiçoado, enquanto o sistema era desenvolvido.

#### **4.4 Estudo de caso: aplicação do processo PDWAU**

De acordo com Brink (1998), agendas em grupo (GCS - *Group Calendar Systems*) permitem marcar, gerenciar projetos e coordenar ações e compromissos entre muitas pessoas por meio da visualização do seu calendário de atividades pessoais. Um dos benefícios de utilização desses sistemas é o fato de que, como diversos usuários acessam os GCS's, é possível que encontrem horários disponíveis para marcar um compromisso, além de possibilitar que a divulgação facilite o conhecimento de quais atividades estão previstas (Santos e Fortes, 2010).

Uma atividade importante no âmbito universitário consiste na apresentação de Trabalhos de Conclusão de Curso (TCC), em que os alunos em geral apresentam para uma banca de professores seu trabalho final. O auxílio dos GCS's é importante para conciliar os diversos horários para a apresentação desses trabalhos. Antes do desenvolvimento do sistema de agendamento de bancas com o apoio do processo PDWAU, estava em operação o sistema Agendamento<sup>13</sup>, que foi desenvolvido de maneira *ad-hoc* para gerenciar os horários e foi utilizado durante vários anos. No entanto, essa versão inicial do sistema ficou em desuso por não atender mais os novos requisitos, tais como a participação dos alunos e dos professores via sistema, bem como requisitos de acessibilidade e usabilidade.

Com base nos requisitos disponíveis no sistema Agendamento, foi desenvolvido o sistema AgendAloca<sup>14</sup> no ICMC-USP com o apoio do processo PDWAU. Para isso, um

<sup>13</sup> Disponível em: <http://agua.intermidia.icmc.usp.br/agendamento/>

<sup>14</sup> Disponível em: <http://garapa.intermidia.icmc.usp.br:3000/agendaloca/>

estudo de caso foi realizado e será detalhado nas próximas Subseções. Os resultados apresentados a seguir, seguindo a ordem das atividades e ações do processo, são recortes das quatro iterações do processo PDWAU, durante o desenvolvimento do sistema AgendAloca. Esses recortes foram selecionados por evidenciarem as tarefas e subtarefas especialmente adicionadas ao processo genérico, dedicadas às questões de acessibilidade e usabilidade.

#### **4.4.1 Comunicação**

Como o sistema Agendamento foi desenvolvido de maneira *ad-hoc*, não havia nenhuma documentação que pudesse ser aproveitada. Portanto, todas as atividades do processo foram realizadas para o desenvolvimento do AgendAloca, começando pela ação de formulação.

##### **4.4.1.1 Formulação**

Na ação de formulação, os *stakeholders* foram definidos: administrador, alunos e professores. A técnica utilizada para a elicitação de requisitos foi o “grupo de foco tradicional”, que permitiu a reunião com os interessados no processo para discutir sobre o sistema de agendamento de bancas e a partir da discussão, entender melhor os requisitos do sistema.

Como informações básicas do sistema, tem-se que o administrador registrará os alunos e os professores; os alunos terão domínio sobre o registro de bancas e os professores poderão candidatar-se a uma ou mais bancas para participar.

Algumas metas informativas puderam ser definidas, tais como: oferecer aos alunos os horários disponíveis para que registrem seu trabalho e oferecer ao professor uma lista dos trabalhos ainda disponíveis, incluindo título do trabalho, horário da banca e nome do orientador. Como meta informativa de acessibilidade, os usuários poderão obter informações sobre o nível de acessibilidade do sistema, baseado nos critérios de sucesso do WCAG (por meio do selo de acessibilidade emitido por ferramentas de avaliação automática, por exemplo).

Para metas aplicativas tem-se: permitir que o desenvolvedor registre alunos e professores e permitir que o aluno registre seu trabalho. A meta adicionada pensando nos requisitos de acessibilidade foi oferecer ferramentas que permitirão aos usuários navegar de maneira mais acessível (aumento de fonte e descrição de imagens, por exemplo).

#### 4.4.1.2 Elicitação

O Sistema AgendAloca tem como objetivo principal auxiliar o agendamento de bancas referentes aos trabalhos de conclusão de curso desenvolvidos no ICMC-USP. Os usuários deste sistema são alunos e professores. Os alunos poderão acessar o sistema para agendar sua data e horário de apresentação e os professores poderão acessar o sistema para escolher em quais bancas participarão como avaliadores.

Como categoria de usuários tem-se: administrador, alunos e professores. Para tanto, o sistema de agendamento de bancas deverá apresentar três áreas para interação com os usuários:

- 1) Área do administrador, responsável por gerenciar as bancas e os usuários do sistema de reservas, atribuindo-lhes permissões para o uso do sistema.
- 2) Área para alunos cadastrados no sistema, onde eles podem fazer buscas e registrar banca em data específica.
- 3) Área para professores cadastrados no sistema, onde eles podem fazer buscas, indicar, modificar e excluir participação em bancas em datas específicas.

A princípio, o desenvolvimento do sistema AgendAloca não teve como foco alguma deficiência específica, ou seja, categoria de usuários separados por grupos de usuários. Portanto, a meta é alcançar a acessibilidade básica (atender ao maior número possível de requisitos de acessibilidade e usabilidade, sem considerar uma deficiência especificamente). De acordo com o APÊNDICE A, pode-se dizer nesse caso, que a preocupação é em atender múltiplas deficiências ao invés de uma deficiência específica.

O documento de requisitos do sistema AgendAloca é encontrado no ANEXO III. Os requisitos de usabilidade e acessibilidade foram inseridos como requisitos não funcionais, com o apoio do APÊNDICE B. Já no ANEXO IV, encontra-se as descrições de casos de uso para cada categoria de usuário, bem como as interações dos usuários de acordo com as suas restrições (no campo canal para ator).

O esforço demandado para executar cada caso de uso descrito no ANEXO IV é encontrado na Tabela 3.

Tabela 3: Esforço demandado para cada caso de uso de requisitos funcionais.

Número do caso de uso	Descrição	Indicador de esforço (1 a 4)
1	Fazer <i>login</i>	2
2	Esqueceu nome de usuário e/ou senha	1
3	Modificar dados pessoais	2
4	Exibir informações do sistema	1
5	Obter ajuda	1
6	Sair do sistema	1
7	Criar usuário	3
8	Cadastrar datas e horários disponíveis	4
9	Emitir relatórios	2
10	Buscar horários vagos	3
11	Agendar data e hora da apresentação	4
12	Visualizar apresentação agendada	1
13	Buscar trabalhos	4
14	Marcar participação na apresentação de trabalhos	4
15	Visualizar agenda de compromissos	2

Conforme indicado na Tabela 3, há o indicador de esforço necessário para cada caso de uso. Esses casos de uso são de requisitos funcionais do sistema. Os requisitos não funcionais do sistema (usabilidade e acessibilidade) são encontrados na Tabela 4. Os casos de uso de usabilidade (U) e acessibilidade (A) são chamados de transversais (T1 a T17 – Tabela 4), pois precisam estar presentes em todas as páginas do sistema web.

Tabela 4: Esforço demandado para cada caso de uso de requisitos não funcionais.

Número do caso de uso transversal	Descrição	Indicador de esforço (1 a 4)
U-T1	Prover interface baseada em padrões de interface e navegação	2
U-T2	Prover <i>breadcrumb</i>	2
U-T3	Assegurar termos familiares aos usuários	2
U-T4	Oferecer atalhos	3
U-T5	Prover mensagens de erro precisas	2
U-T6	Oferecer uma área de busca	3
A-T7	Prover acesso via teclado	4
A-T8	Fornecer alternativas textuais para conteúdos sonoro e visual	3
A-T9	Garantir a percepção do conteúdo mesmo quando visto sem cores	2
A-T10	Separar conteúdo da apresentação	1
A-T11	Fornecer uma área clicável maior para <i>links</i>	2
A-T12	Criar tabelas acessíveis	2

A-T13	Prover opção de aumentar e diminuir a fonte do conteúdo	3
A-T14	Oferecer opção de ir direto para o conteúdo	1
A-T15	Possibilitar alteração do contraste do conteúdo	3
A-T16	Prover títulos descritivos para <i>links</i>	2
A-T17	Criar formulários acessíveis	4

Uma lista de incrementos foi definida:

- Incremento 1: Casos de Uso 1, 2 e 3
- Incremento 2: Casos de Uso 4, 5 e 6
- Incremento 3: Casos de Uso 7 e 8
- Incremento 4: Casos de Uso 9, 10, 11, 12, 13, 14 e 15
- **Incremento 1: Casos de Uso U-T1, U-T3, U-T5, U-T6 e A-T10**
- **Incremento 2: Casos de Uso A-T7, AT8, AT11 e A-T14**
- **Incremento 3: Casos de Uso A-T2, A-T13, A-T15 e A-T16**
- **Incremento 4: Casos de Uso A-T4, A-T9, A-T12 e A-T17**

Ao finalizar a ação de elicitação, os requisitos do AgendAloca foram definidos e esclarecidos entre os *stakeholders* e desenvolvedores envolvidos.

#### 4.4.1.3 Negociação

Como o sistema AgendAloca foi desenvolvido para ser usado no âmbito universitário, a ação de negociação não foi realizada. Além disso, não foram adicionadas mudanças de usabilidade e acessibilidade para essa ação, apenas algumas possíveis sugestões foram oferecidas.

#### 4.4.2 Planejamento

Após a identificação dos requisitos básicos do sistema, a atividade de planejamento foi iniciada. Algumas tarefas foram realizadas, tais como: um plano incremental, as tarefas técnicas a serem realizadas, os riscos prováveis, os recursos que serão exigidos, os produtos de trabalho a serem produzidos e um cronograma de trabalho, que serão explicados nas ações de estimação, análise de risco e monitoração, a seguir.

##### 4.4.2.1 Estimação

Baseado nas descrições dos casos de uso, bem como no esforço indicado em cada um deles, o cronograma macroscópico foi desenvolvido. O cronograma indicado na Tabela 5 apresenta uma visão geral dos incrementos principais e da quantidade de semanas envolvidas.

Tabela 5: Cronograma de implantação preliminar macroscópico.

Incrementos	Semanas										
	1	2	3	4	5	6	7	8	9	10	
1. Tela inicial do sistema com informações e funcionalidades de acessibilidade básicas	■										
2. Tela inicial do sistema com informações detalhadas e funcionalidades de acessibilidade específicas				■							
3. Interações básicas do sistema divididas em visões de usuários (admin, professor e aluno)					■						
4. Controle de todo o sistema e disponibilização de todos os serviços								■			

A partir do cronograma macroscópico indicado na Tabela 5, foi definido um cronograma microscópico, ou seja, um detalhamento do esforço médio ( $E_{méd}$ ) exigido para implementar cada caso de uso (em homens-dias). Esse esforço médio e os casos de uso a serem realizados em cada incremento podem ser encontrados na Tabela 6.

Tabela 6: Cronograma de implantação preliminar microscópico.

Casos de Uso	$E_{méd}$
1. Incremento	
Fazer <i>login</i>	1
Esqueceu nome de usuário e/ou senha	1
Modificar dados pessoais	1
<u>Prover interface baseada em padrões de interface e navegação</u>	1
<u>Assegurar termos familiares aos usuários</u>	1
<u>Prover mensagens de erro precisas</u>	1
<u>Oferecer uma área de busca</u>	1
<u>Separar conteúdo da apresentação</u>	1
2. Incremento	
Exibir informações do sistema	1
Obter ajuda	1
Sair do sistema	1
<u>Prover acesso via teclado</u>	2
<u>Fornecer alternativas textuais para conteúdos sonoro e visual</u>	1
<u>Fornecer uma área clicável maior para links</u>	1
<u>Oferecer opção de ir direto para o conteúdo</u>	1

3. Incremento	
Criar usuário	2
Cadastrar datas e horários disponíveis	2
<u>Prover breadcrumb</u>	1
<u>Prover opção de aumentar e diminuir a fonte do conteúdo</u>	1
<u>Possibilitar alteração do contraste do conteúdo</u>	1
<u>Prover títulos descritivos para links</u>	1
4. Incremento	
Emitir relatórios	1
Buscar horários vagos	2
Agendar data e hora da apresentação	2
Visualizar apresentação agendada	1
Buscar trabalhos	1
Marcar participação na apresentação de trabalhos	2
Visualizar agenda de compromissos	1
<u>Oferecer atalhos</u>	2
<u>Garantir a percepção do conteúdo mesmo quando visto sem cores</u>	1
<u>Criar tabelas acessíveis</u>	2
<u>Criar formulários acessíveis</u>	2

É interessante notar que tanto os casos de uso de requisitos funcionais quanto os casos de uso dos requisitos não funcionais foram adicionados no cronograma microscópico apresentado na Tabela 6.

#### 4.4.2.2 Análise de risco

Para identificar os possíveis riscos no desenvolvimento do sistema AgendAloca, bem como de qualquer outro sistema, é necessário classificar os riscos oriundos de pessoas, produto e processo, conforme apresentado na Tabela 7.



Tabela 7: Possíveis riscos no desenvolvimento do AgendAloca.

Riscos	Probabilidade	Impacto
Pessoas		
Pouca ou nenhuma experiência no <i>framework</i> adotado	80%	2
Interessados não cooperativos	70%	2
<b><u>Pouco ou nenhum conhecimento em requisitos de acessibilidade</u></b>	90%	3
<b><u>Pouco ou nenhum conhecimento no desenvolvimento de funcionalidades de acessibilidade</u></b>	90%	3
Produto		
Alteração de leiaute mais difícil do que o esperado	40%	1
<b><u>Dificuldade em inserir funcionalidades de acessibilidade e usabilidade</u></b>	60%	2
Processo		
Documentação incompleta nas ações e/ou tarefas a serem desenvolvidas	60%	2
Muitas tarefas em uma única ação	50%	1
<b><u>Não há ênfase suficiente na inserção de acessibilidade e usabilidade</u></b>	90%	3

Na Tabela 7 é possível perceber que além da classificação dos riscos, é importante indicar qual a probabilidade do risco acontecer, bem como o impacto que isso pode refletir no desenvolvimento do sistema.

#### 4.4.2.3 Escalonamento

A ação escalonamento também não foi realizada, pois o sistema AgendAloca foi desenvolvido por alunos de Iniciação Científica (IC), em que cada um desenvolvia uma determinada parte do sistema ou quando era necessário mais de um aluno, eles trabalhavam em conjunto, não sendo necessário realizar escalonamento.

#### 4.4.2.4 Monitoração

Para facilitar o acompanhamento do desenvolvimento, tanto por parte do desenvolvedor, quanto por parte do gestor do projeto, a ferramenta online Redmine (Figura 25) foi escolhida para a geração do *workflow* de desenvolvimento.

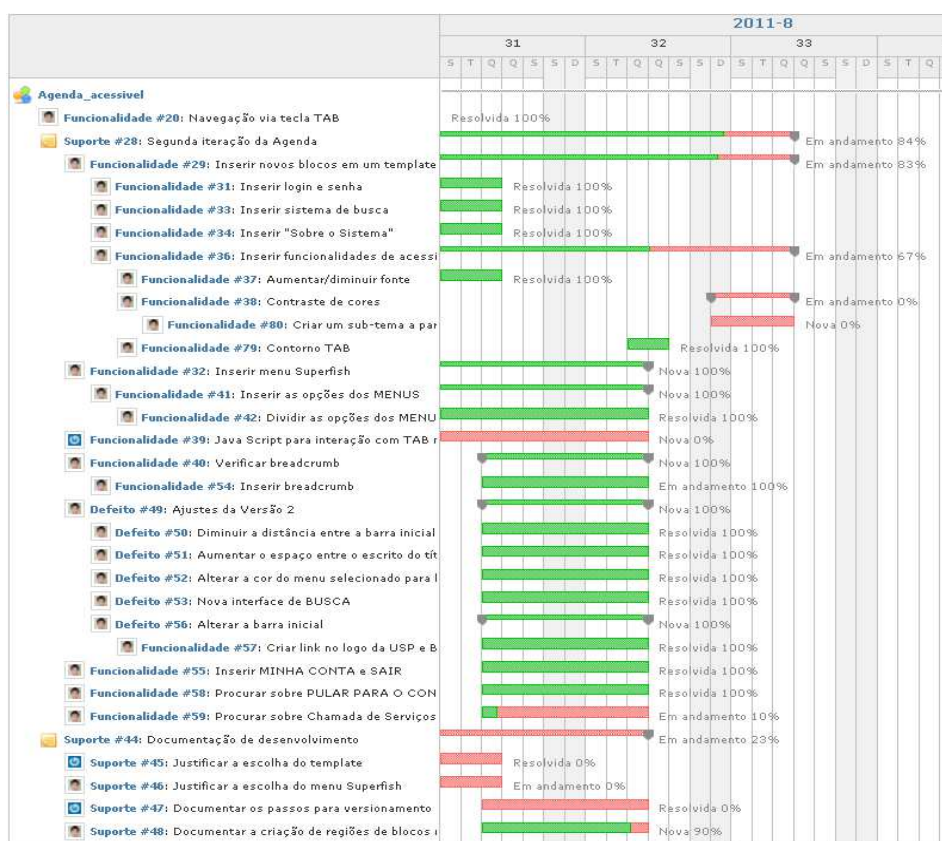


Figura 25: Workflow de desenvolvimento do sistema AgendAloca.

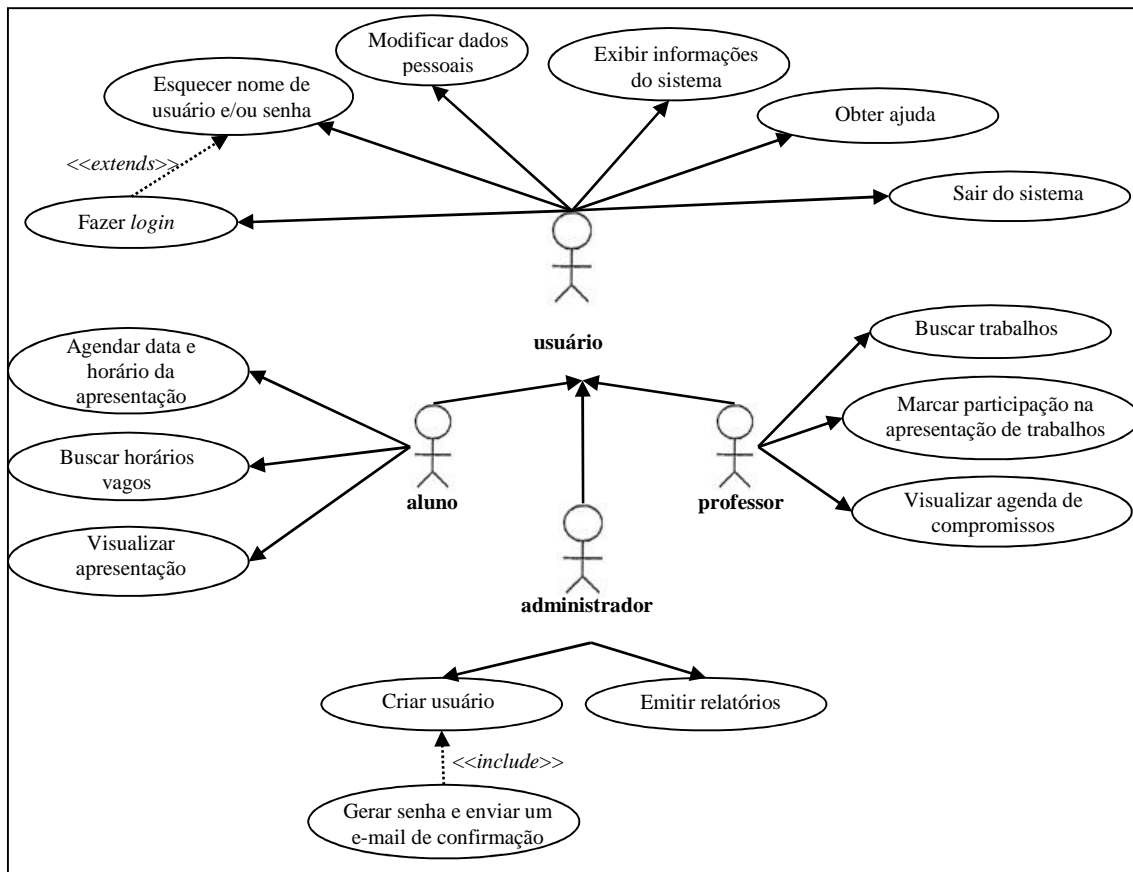
O registro do *workflow* de desenvolvimento do sistema é muito útil para o acompanhamento dos passos que já foram realizados e dos próximos passos a serem realizados, tanto por parte do gerente quanto do desenvolvedor.

#### 4.4.3 Modelagem

A atividade de modelagem ajuda a resolver problemas de ambiguidade encontrados na linguagem natural e precisa ser utilizada de modo favorável. Portanto, para o desenvolvimento do AgendAloca, foram realizadas modelagens conforme a necessidade percebida pelos envolvidos no desenvolvimento do sistema.

##### 4.4.3.1 Análise

Nesta ação os requisitos dos interessados são examinados em um esforço para entender o que o sistema web deve fazer. A primeira análise realizada foi uma hierarquia de usuários com as tarefas/opções disponíveis no sistema, conforme indicado na Figura 26.



**Figura 26: Hierarquia de usuários com as tarefas/opções gerais disponíveis no sistema.**

Como podem ser percebidos na Figura 26, os *stakeholders* são aluno, professor e administrador. Cada um tem suas tarefas específicas e também tarefas em comum que podem ser encontradas no ator usuário. Essas tarefas/opções se limitam nos requisitos funcionais. As tarefas/opções de requisitos não funcionais de usabilidade e acessibilidade foram adicionadas na Figura 27.



Figura 27: Hierarquia de usuários com as tarefas de usabilidade e acessibilidade disponíveis no sistema.

Na Figura 27 é possível perceber que os requisitos não funcionais de usabilidade e acessibilidade devem ser aplicados a todos os usuários e, portanto eles têm ligações com todas as tarefas/opções gerais apresentadas na Figura 26.

Além da hierarquia de usuários com seus principais requisitos, uma análise dos requisitos por meio do modelo de interação, mais especificamente utilizando protótipo, foi realizada. Os protótipos ofereceram um entendimento do leiaute e da estética do sistema, dos mecanismos de navegação e, em casos mais limitados, da funcionalidade e do conteúdo de informação, como indicado na Figura 28.

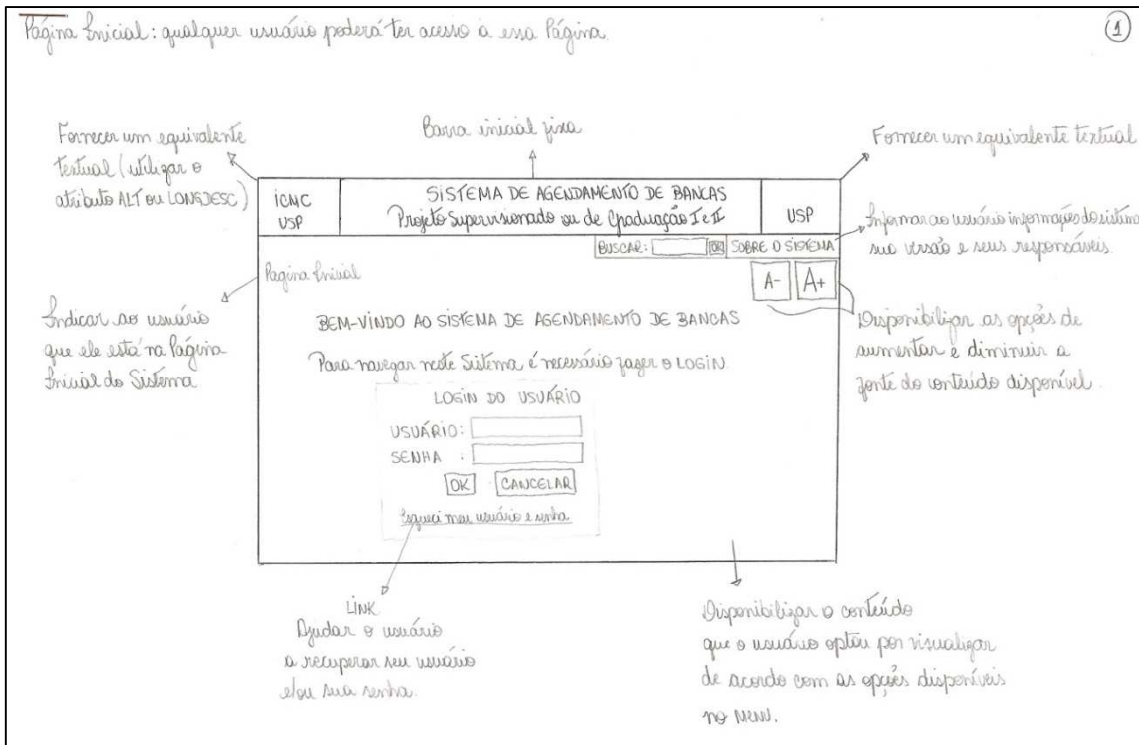


Figura 28: Protótipo da página inicial do AgendAloca.

Percebe-se, na Figura 28, uma interface básica inicial do sistema que deve ser a primeira visão que todos os usuários poderão encontrar, independente de categoria. Já na Figura 29, os requisitos de usabilidade e acessibilidade foram também indicados no protótipo de uma maneira estruturada.

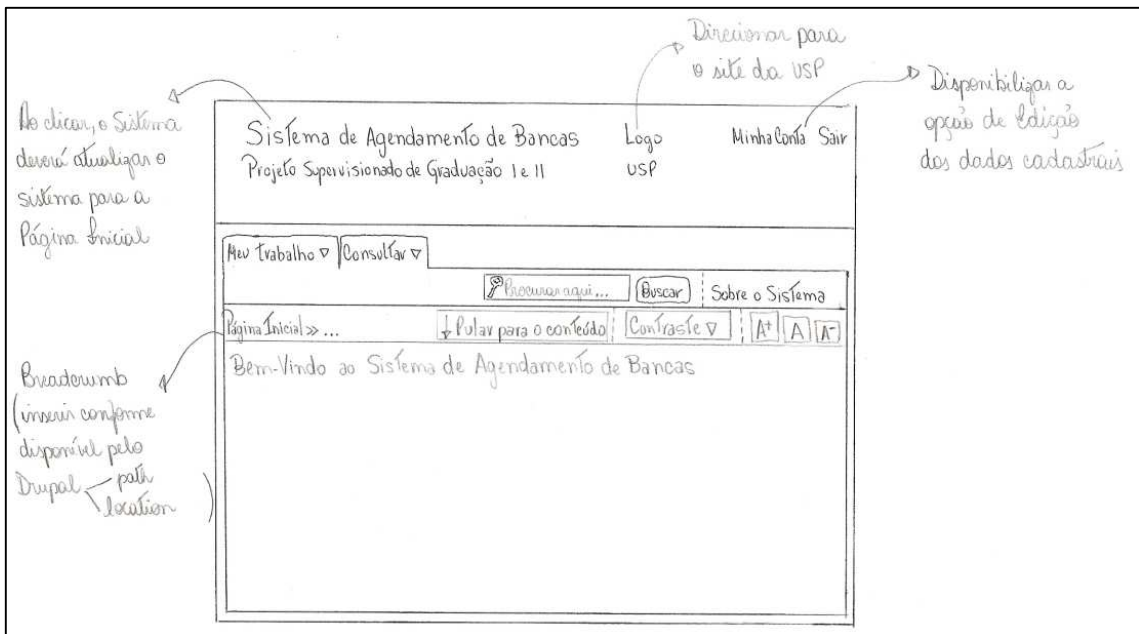


Figura 29: Página inicial do AgendAloca após login do usuário.

Após a análise dos requisitos, a ação de projeto foi iniciada.

### 4.4.3.2 Projeto

A modelagem na atividade do projeto foi utilizada para compreender a estrutura interna do sistema web em desenvolvimento para cada categoria de usuário, conforme apresentada nas Figuras 30 e 31.

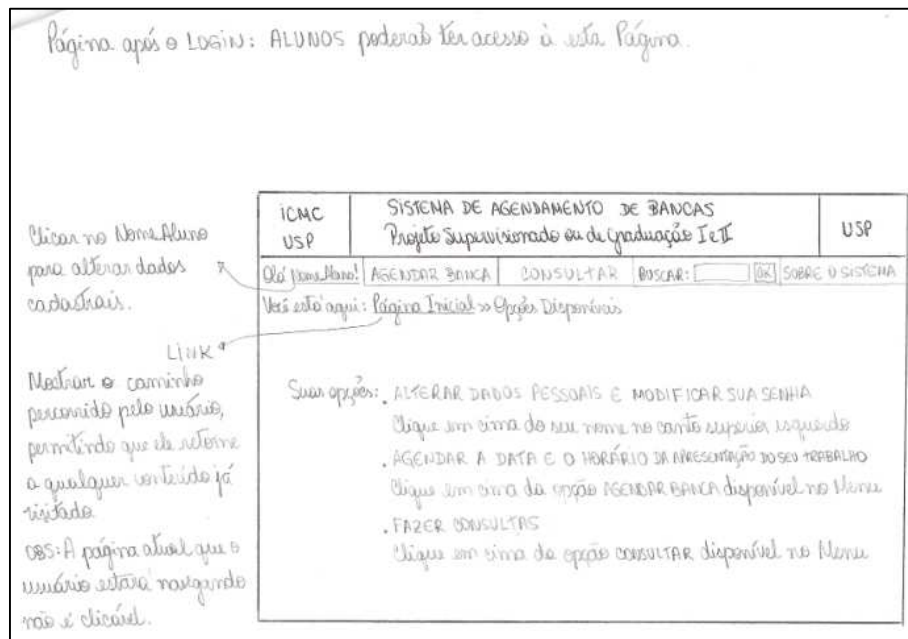


Figura 30: Projeto de interface do AgendAloca após login do aluno.

Na Figura 30 o projeto de interface do aluno é apresentado, enquanto na Figura 31 tem-se o projeto de interface do professor, após o login.

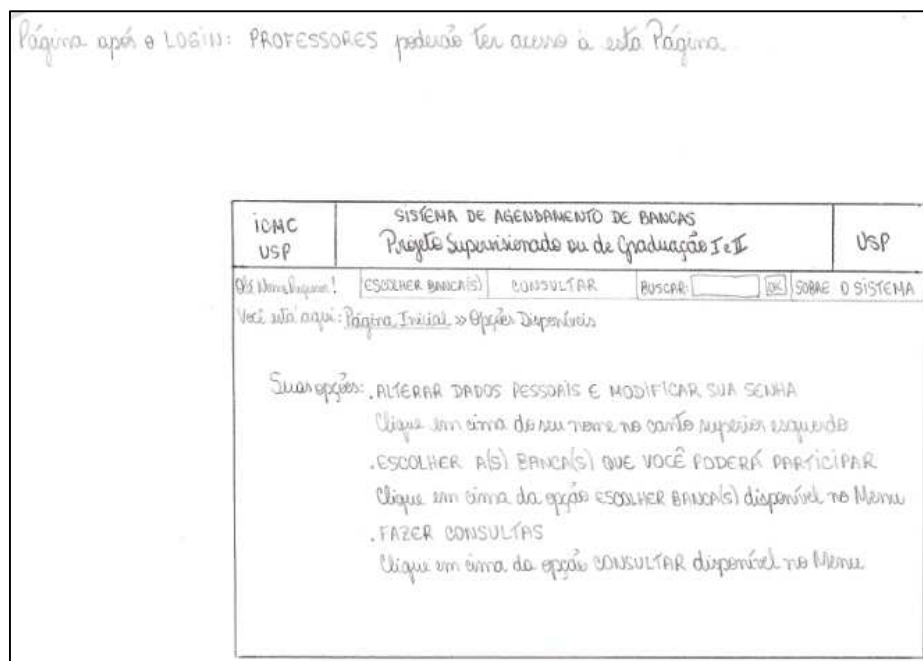


Figura 31: Projeto de interface do AgendAloca após login do professor.





Um esboço do projeto funcional do sistema também foi desenvolvido, conforme apresentado na Figura 34.

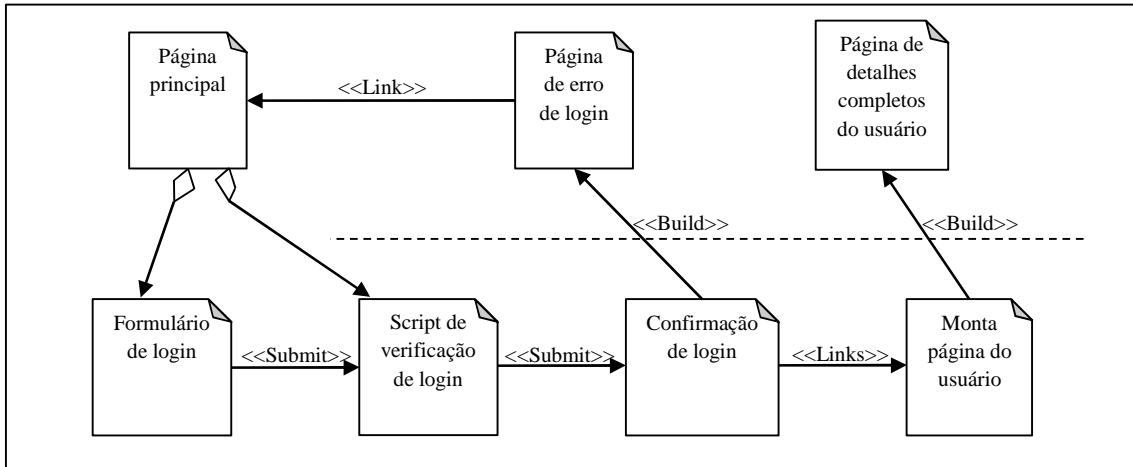


Figura 34: Projeto funcional do AgendAloca.

No projeto técnico, apresentado na Figura 35, foi criado um modelo de classes que identifica a estrutura geral, apoiando a lógica de processamento detalhada exigida para implementar componentes funcionais.

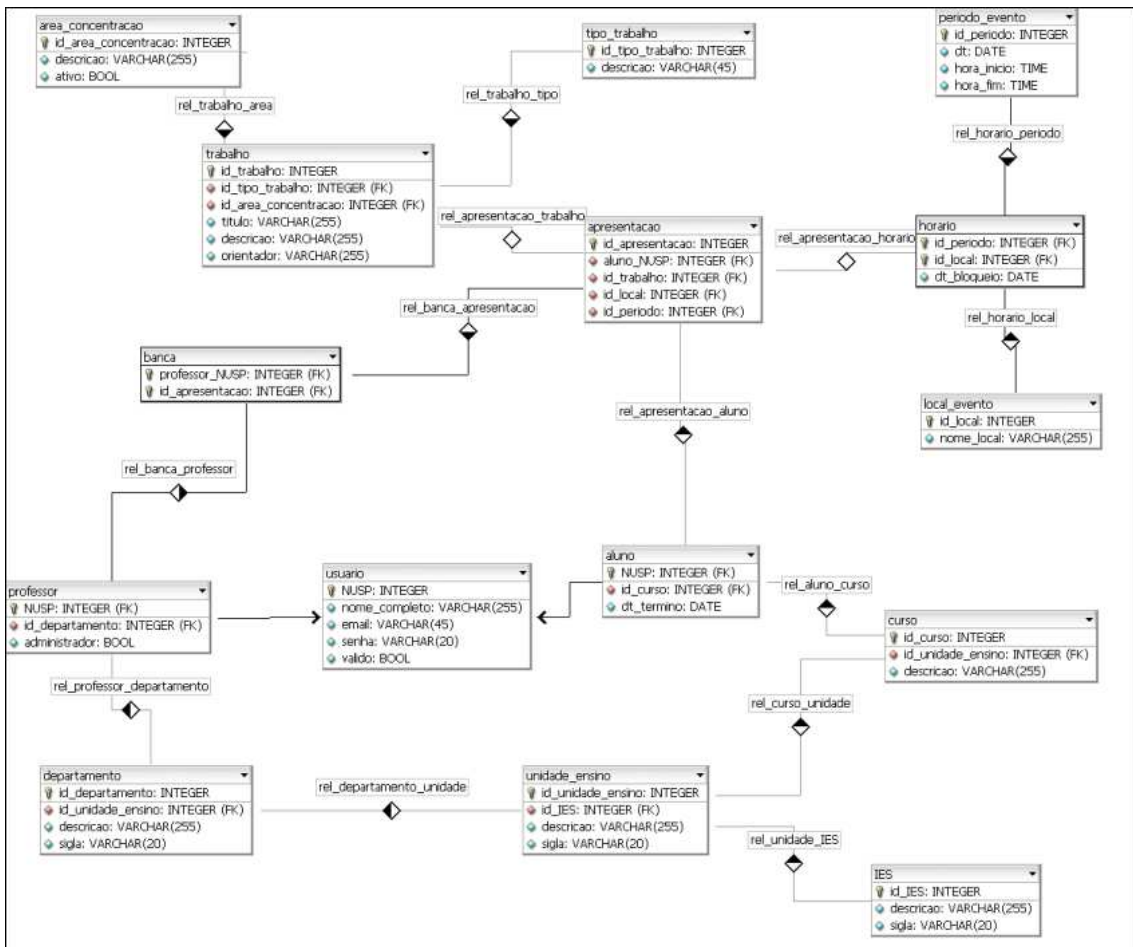


Figura 35: Projeto técnico do AgendAloca.



Toda a modelagem realizada nas ações de análise e projeto foi fundamental para ajudar os desenvolvedores no entendimento dos requisitos, antes de iniciar a atividade de construção do sistema.

#### **4.4.4 Construção**

Nessa atividade algumas tecnologias foram utilizadas para desenvolver o sistema AgendAloca. Elas serão descritas nas ações de codificação e testes.

##### **4.4.4.1 Codificação**

Primeiramente, optou-se por desenvolver o AgendAloca utilizando o CMS (*Content Management System*) Drupal versão 7, que tem o núcleo seguindo as diretrizes WCAG 2.0 e ATAG 2.0, a fim de visar usabilidade e acessibilidade em todo processo de desenvolvimento. Apesar das características de acessibilidade percebidas no núcleo do Drupal, não há garantia que elas sejam seguidas fora do núcleo principal, ou seja, não é possível certificar que os módulos, disponibilizados pela comunidade científica e pelos desenvolvedores do Drupal, seguem algumas das diretrizes de acessibilidade.

Percebendo a necessidade de alterar o código fonte de vários módulos e do núcleo do CMS para que fosse alcançada a funcionalidade desejada, notou-se que o escopo do CMS, que visa à criação e alteração de conteúdo em nível visual, estava sendo violado. Com a falta de documentação dos módulos e a dificuldade de modificar o código fonte, o desenvolvimento de um sistema web acessível em Drupal 7 foi considerado insatisfatório, e foi descontinuado.

Pouco antes da descontinuação do sistema, foi iniciado em paralelo o desenvolvimento do mesmo sistema de agendamento de bancas, no *framework* JSF (*Java Server Faces*), que se mostrou promissora por dar suporte à comunicação com serviços web (*web services*) desde o início. Como outro pesquisador do grupo estava desenvolvendo seu trabalho de mestrado com *web services* e seria interessante aproveitar esses conceitos no desenvolvimento do AgendAloca, optou-se por continuar o desenvolvimento do sistema com o *framework* JSF.

Portanto, o desenvolvimento do AgendAloca foi feito por meio de serviços web *RESTful*. Isso possibilitou separar facilmente a camada de leiaute com a camada de funcionalidade, apoiando o desenvolvimento de requisitos de usabilidade e acessibilidade durante todo o processo de desenvolvimento.

Os serviços web *RESTful* disponibilizados foram realizados usando duas API's (*Application Programming Interface*) Java: JAX-RS e JAXB. A API JAX-RS permitiu a criação de clientes para a chamada aos recursos disponibilizados pelos serviços web. Já a API JAXB permitiu realizar o mapeamento de objetos Java para documentos XML e vice-versa. Dessa forma, se tornou simples a utilização de representações dos dados no formato XML, que é uma linguagem independente de plataforma, garantindo assim a portabilidade e o reuso desses serviços (Sanchez, Oliveira e Fortes, 2014). Portanto, como o desenvolvimento do sistema AgendAloca contou com o apoio de serviços web, foi possível um código bem enxuto, não sendo necessário passar pela ação de refatoração

Como o AgendAloca foi desenvolvido desde o início para inserir usabilidade e acessibilidade, não pôde-se aproveitar nenhuma parte do código existente no sistema Agendamento. No entanto, pôde-se aproveitar uma referência que já havia sido desenvolvida por outros colegas do grupo de desenvolvimento, o menu *Superfish*. Em uma pesquisa realizada por Santos *et al.* (2011), esse menu recebeu uma das melhores avaliações do estudo, sendo o segundo preferido pelos usuários, tanto no quesito facilidade de uso, quanto na avaliação por nota. Houve poucos erros, e de todos os menus, foi o que demonstrou ter o aprendizado de utilização mais rápido.

*Superfish* é um menu amplamente utilizado em sistemas web. Apresenta sinalização de submenus (através de uma seta indicando a direção da aparição do submenu) e sua área de ativação corresponde a toda a caixa do menu, sendo que cada divisão entre as opções é marcada por uma linha vertical. Os submenus são apresentados abaixo da opção principal com o efeito *fade-in* e apresenta um leve sombreamento de contorno, conforme apresentado na Figura 36.



Figura 36: Exemplo de menu *Superfish*.

O menu apresenta navegação pela tecla <tab> do teclado, sendo possível acompanhar a navegação de forma visual, além do destacamento de cores que indica o

caminho dentro dos subníveis. Após a codificação, alguns testes foram realizados no sistema AgendAloca e serão apresentados a seguir.

#### 4.4.4.2 Testes

O primeiro teste foi realizado com o software WAVE<sup>15</sup> em outubro de 2011, conforme indicado na Figura 37.



Figura 37: Teste do AgendAloca com o WAVE.

Para realizar o teste verificando a acessibilidade do sistema AgendAloca por meio do software WAVE, foi utilizado o *addon* para Firefox que está disponível no site da ferramenta. Este *addon* permite avaliar páginas dinâmicas, que não estão online ou protegidas por senha, entre outros fatores que o site não é capaz de avaliar.

A barra da ferramenta contém quatro funções principais, que foram abordadas nesse teste: I) *Errors, Features, and Alerts*; II) *Structure/Order*; III) *Text-only*; e IV) *Outline*.

A primeira função “*Errors, Features, and Alerts*” exhibe os possíveis erros, características e alertas da página. Com esta ferramenta é possível fazer uma verificação do código da página de forma visual, fazendo com que seja mais fácil a visualização dos erros existentes, conforme apresentado na Figura 38.

<sup>15</sup> Disponível em: <http://wave.webaim.org/>



Figura 38: Teste do AgendAloca com o WAVE – Função *Errors, Features, and Alerts*.

De acordo com a Figura 38, é possível perceber que não há nenhum erro. No entanto, a ferramenta informa que é necessária uma verificação por parte de um ser humano, pois a ferramenta não passa de uma simulação.

Como característica presente nesse sistema, tem-se a existência de títulos alternativos nas imagens e nos botões. Além disso, são exibidos informações sobre os *links* ou listas (em verde ou branco) e alertas relativos a eventos *JavaScript* (em amarelo). É recomendável utilizar o mínimo de *JavaScript* possível, para manter a compatibilidade com os diversos navegadores.

Já na Figura 39, a função executada é a *Structure/Order*, que separa a página em suas estruturas principais e exhibe em que ordem os elementos serão acessados por meio do teclado. Esta visualização é muito útil para encontrar problemas de acessibilidade para um usuário cego que irá navegar essencialmente com o teclado.

Um problema desta ferramenta é que ela não conta o acesso aos itens do menu, mas a posição de cada elemento principal. Isso pode ser muito útil para verificar qual elemento deve ser posicionado antes ou depois de outros.



Figura 39: Teste do AgendAloca com o WAVE – Função *Structure/Order*.

Na Figura 40 é possível verificar a função *Text-only*, em que todos os elementos visuais da página são removidos. Com esta visualização, fica claro como um usuário cego utilizaria a página por meio de uma tecnologia assistiva e também permite a possibilidade de avaliar como seria a exibição em um *browser* somente texto.

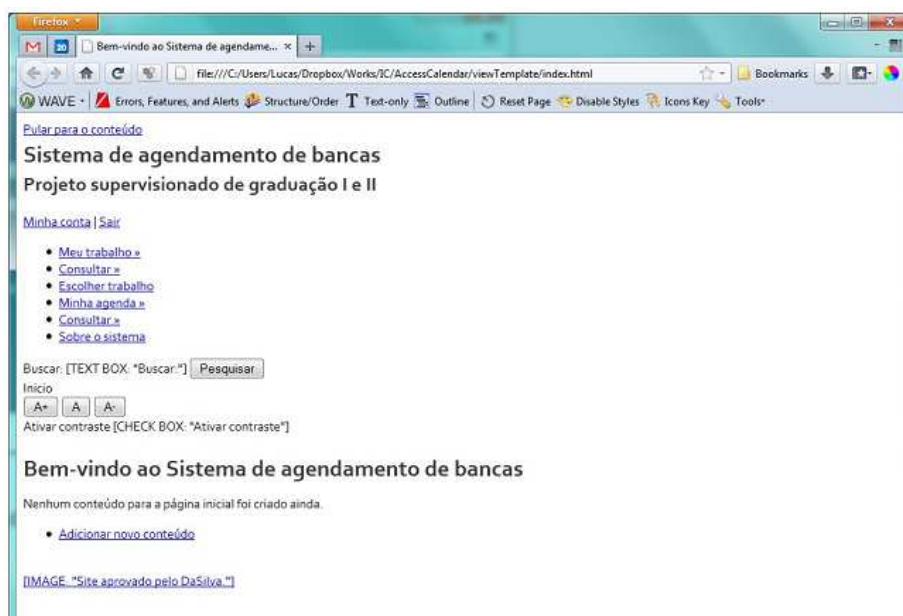


Figura 40: Teste do AgendAloca com o WAVE – Função *Text-only*.

Já na função *Outline* (Figura 41), são exibidos apenas os títulos, para que o desenvolvedor possa avaliar se está fazendo o uso apropriado das *tags* H1, H2, etc.





Figura 41: Teste do AgendAloca com o WAVE – Função Outline.

Como comentado no início do teste com a ferramenta WAVE, a vantagem deste *addon* sobre o site da ferramenta é a possibilidade de visualizar páginas dinâmicas. Na Figura 42, a opção de contraste do sistema foi ativada e, em seguida, a opção “*Errors, Features, and Alerts*” da ferramenta foi acionada. Desta forma é possível avaliar páginas que são criadas em tempo de execução ou páginas que precisam de *login* e senha para ser acessadas, por exemplo. Diversas ferramentas para avaliação de acessibilidade não são capazes de realizar uma avaliação tão profunda como este realizado pelo software WAVE.

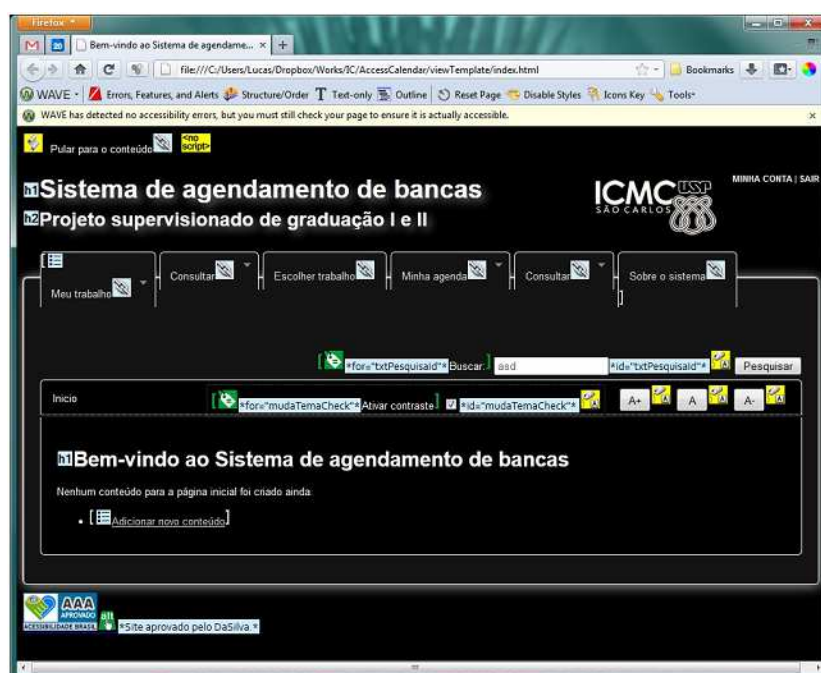


Figura 42: Teste do AgendAloca com o WAVE – Opção de contraste.

Além do software WAVE, utilizou-se também o software aDesigner em outubro de 2011. Com o aDesigner é possível simular possíveis problemas existentes no sistema web que possam impedir o acesso a deficientes visuais.

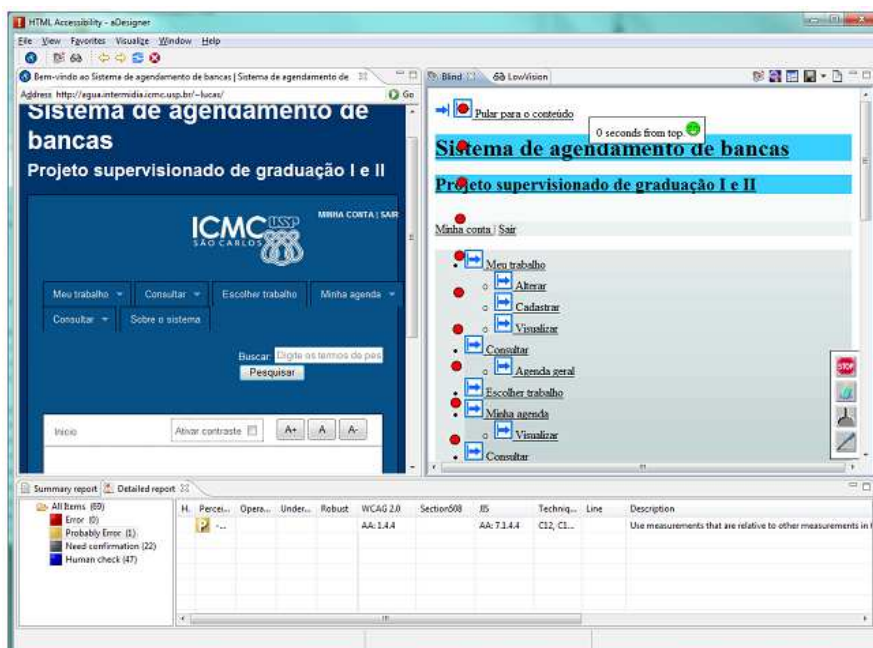


Figura 43: Teste do AgendAloca com o aDesigner – Tela principal.

Conforme indicado na Figura 43, à esquerda é exibida a página que está sendo analisada em seu formato original. À direita é possível ver uma simulação de como um usuário cego navegaria na página, inclusive contando o tempo que ele provavelmente gastaria para alcançar determinados conteúdos.

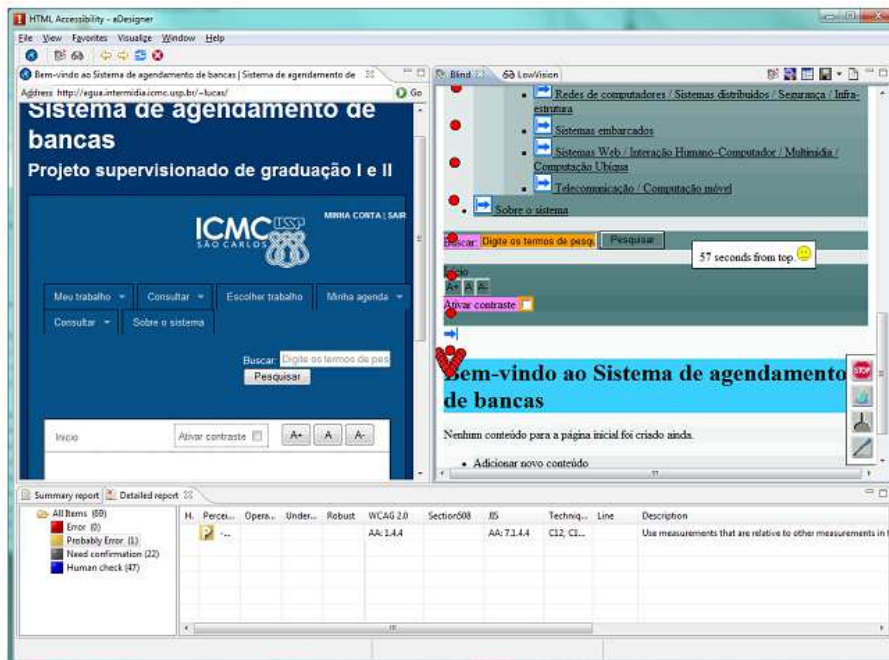


Figura 44: Teste do AgendAloca com o aDesigner – Simulação para cegos I.

Na simulação para cegos, quanto mais escuro for o componente, mais tempo o usuário irá demorar para acessar esse conteúdo. Por exemplo, na Figura 44, é possível perceber que o tempo necessário para se chegar ao campo de busca é de 57 segundos.

Esse tempo é aceitável, porém, poderia ser melhorado colocando-se um *link* direto para este campo, ou colocando-o antes do menu. No caso do menu, as opções menos procuradas pelos usuários deverão ser colocadas por último, enquanto as mais frequentes deverão ser colocadas primeiro.

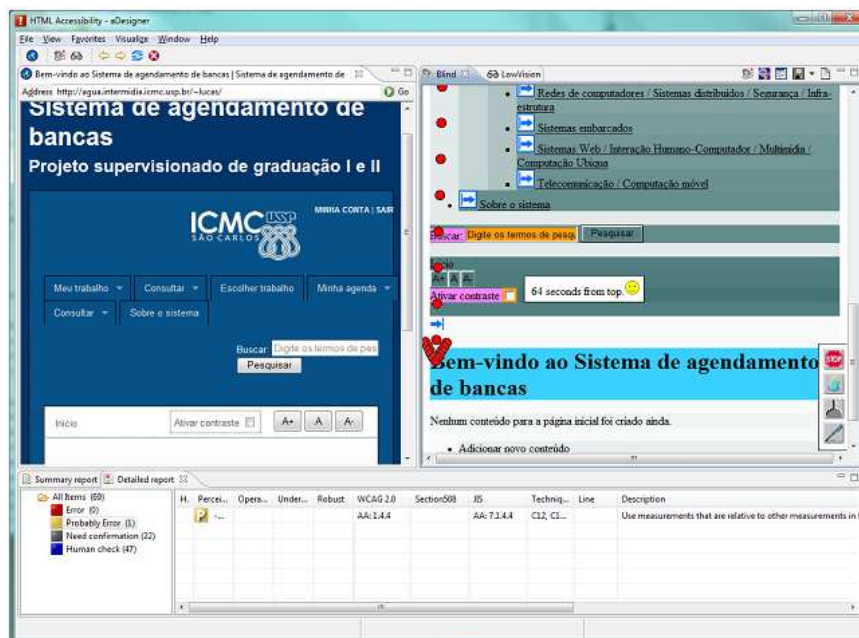


Figura 45: Teste do AgendAloca com o aDesigner – Simulação para cegos II.

Na Figura 45, é possível perceber que os botões para aumentar a fonte ou mudar o contraste do sistema são colocados por último, uma vez que um usuário com deficiência visual que navega apenas com a ajuda de leitores de tela não irá precisar destas opções.

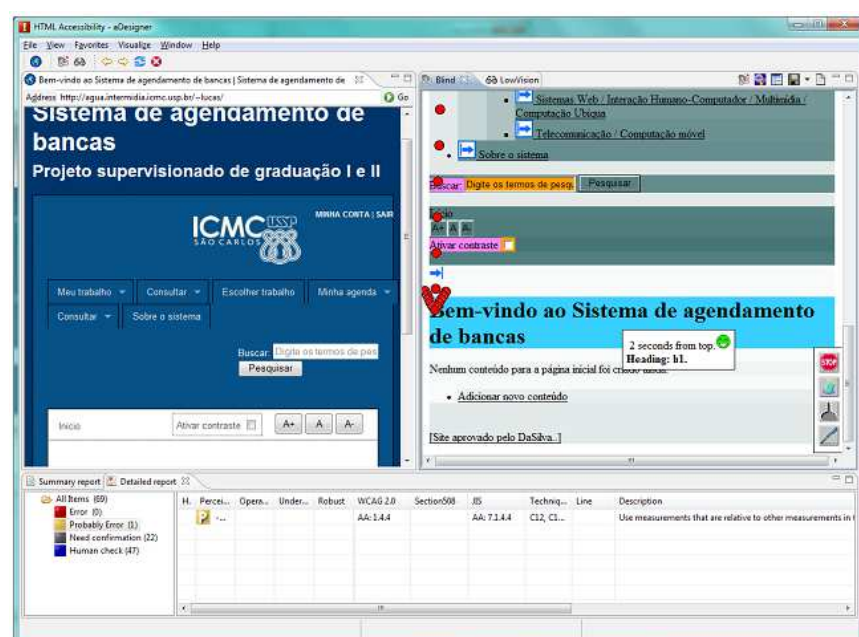


Figura 46: Teste do AgendAloca com o aDesigner – Simulação para cegos III.



Na Figura 46 é possível perceber o efeito do *link* no início da página, que pula direto para o conteúdo da agenda. O aDesigner é capaz de contar o tempo necessário para acessar o conteúdo principal através da utilização deste *link*. Há também a opção de simular como um usuário com uma baixa capacidade de visão iria enxergar o sistema. Analisando esta simulação (Figura 47), é detectado um problema no menu, pois as fontes são pequenas e o contraste é muito baixo, sendo difícil a leitura do texto dos itens disponíveis.

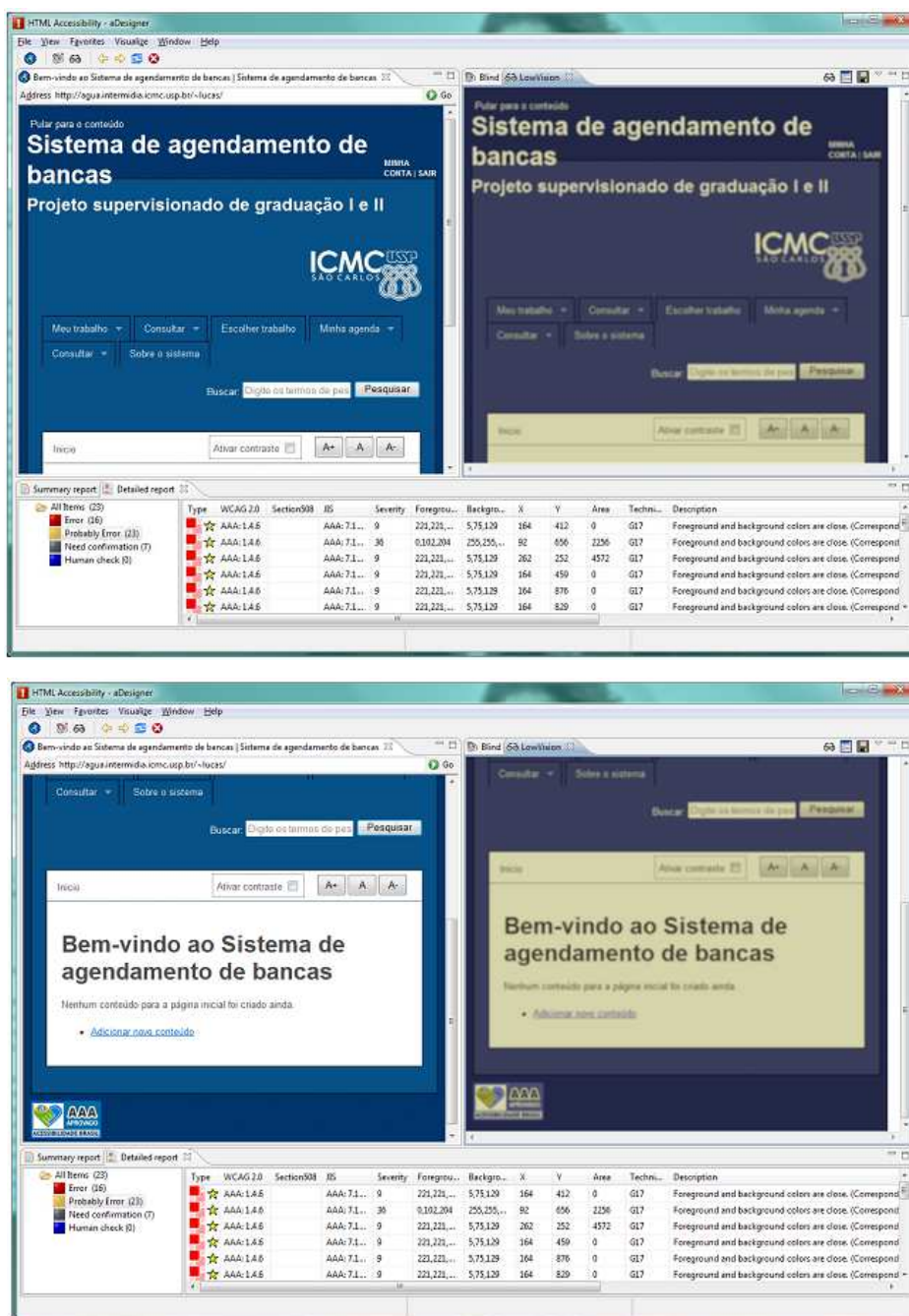


Figura 47: Teste do AgendAloca com o aDesigner – Simulação para cegos IV.

O problema com o contraste de cores, ilustrado na Figura 47, pode ser minimizado quando utiliza-se a opção de aumentar a fonte, conforme Figura 48.

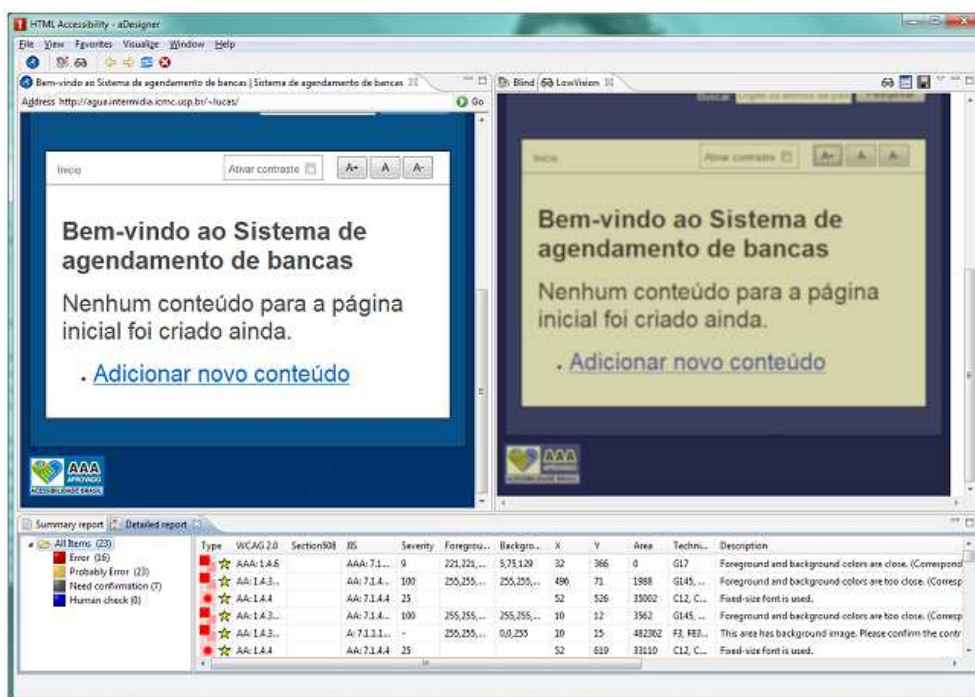


Figura 48: Teste do AgendAloca com o aDesigner – Simulação para cegos V.

É interessante notar na Figura 49, ao utilizar o botão para ativar o contraste, o problema demonstrado na Figura 47 e melhorado na Figura 48 é praticamente eliminado.

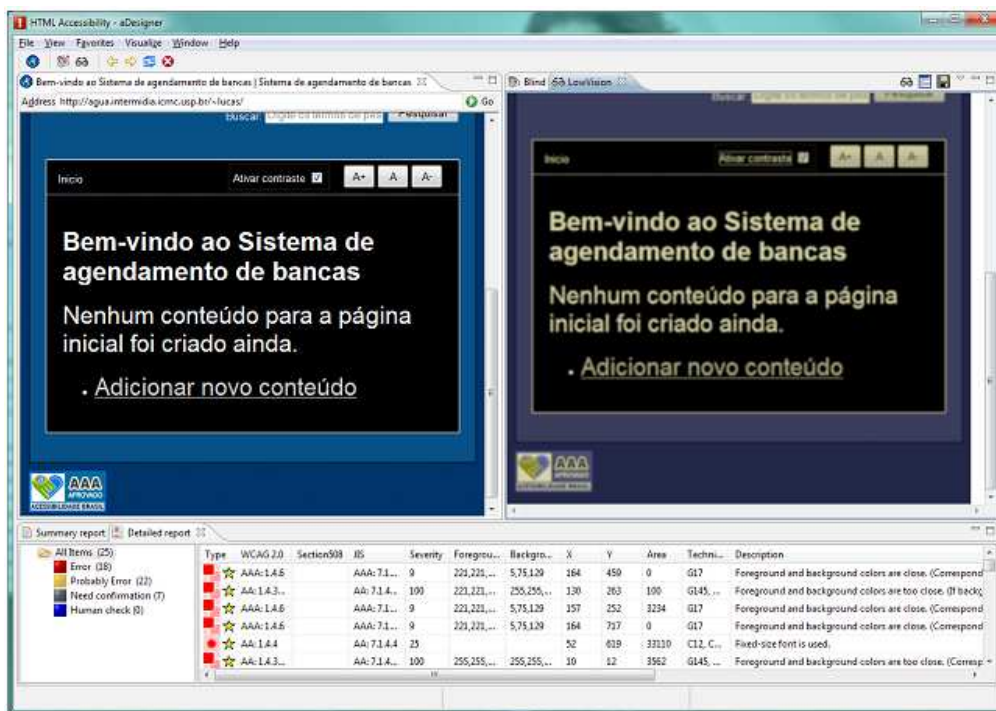


Figura 49: Teste do AgendAloca com o aDesigner – Simulação para cegos VI.

Após a realização de alguns testes, as versões do sistema puderam ser entregues. Para isso, passa-se para a última atividade do processo, a implantação.

#### 4.4.5 Implantação

Na atividade de implantação, encerra-se a iteração do processo. É o momento de disponibilizar as versões do sistema desenvolvidas até aqui. Para o AgendAloca, foram realizadas quatro iterações no processo PDWAU. Conforme comentado anteriormente, nesse estudo de caso mostra-se um recorte de todo o desenvolvimento realizado.

##### 4.4.5.1 Entrega

Todos os componentes do AgendAloca foram desenvolvidos com versionamento, o que muito facilitou na hora de empacotar o que realmente iria para o servidor e o que estaria disponível ao usuário. As versões do AgendAloca foram disponibilizadas a cada iteração concluída, de acordo com as alterações solicitadas pelos usuários e com as falhas percebidas pelos desenvolvedores.

##### 4.4.5.2 Avaliação

A avaliação do AgendAloca ocorreu de naturalmente, com o apoio dos usuários finais (alunos e professores) que acessavam o sistema e forneciam seus *feedbacks*, sem formalidades. O apoio do grupo de pesquisa também foi fundamental, pois várias questões eram discutidas e melhoradas posteriormente.

Adicionalmente, algumas avaliações também foram realizadas com especialistas simulando usuários. Um dos primeiros testes foi a realização da navegação por meio da tecla TAB.



Figura 50: Teste do AgendAloca com a tecla TAB.

Conforme apresentado na Figura 50, o menu disponível no sistema permite que todas as opções estejam acessíveis usando a tecla TAB. Esse mecanismo é um grande facilitador para usuários que utilizam leitores de tela para navegar em sistemas web.

Outros testes com as opções de usabilidade e acessibilidade disponíveis no AgendAloca foram realizados, tais como (Figura 51): a) aumentar e diminuir fonte; b) contraste de cores; c) alteração de idioma; e d) apoio e navegação no *Help*.

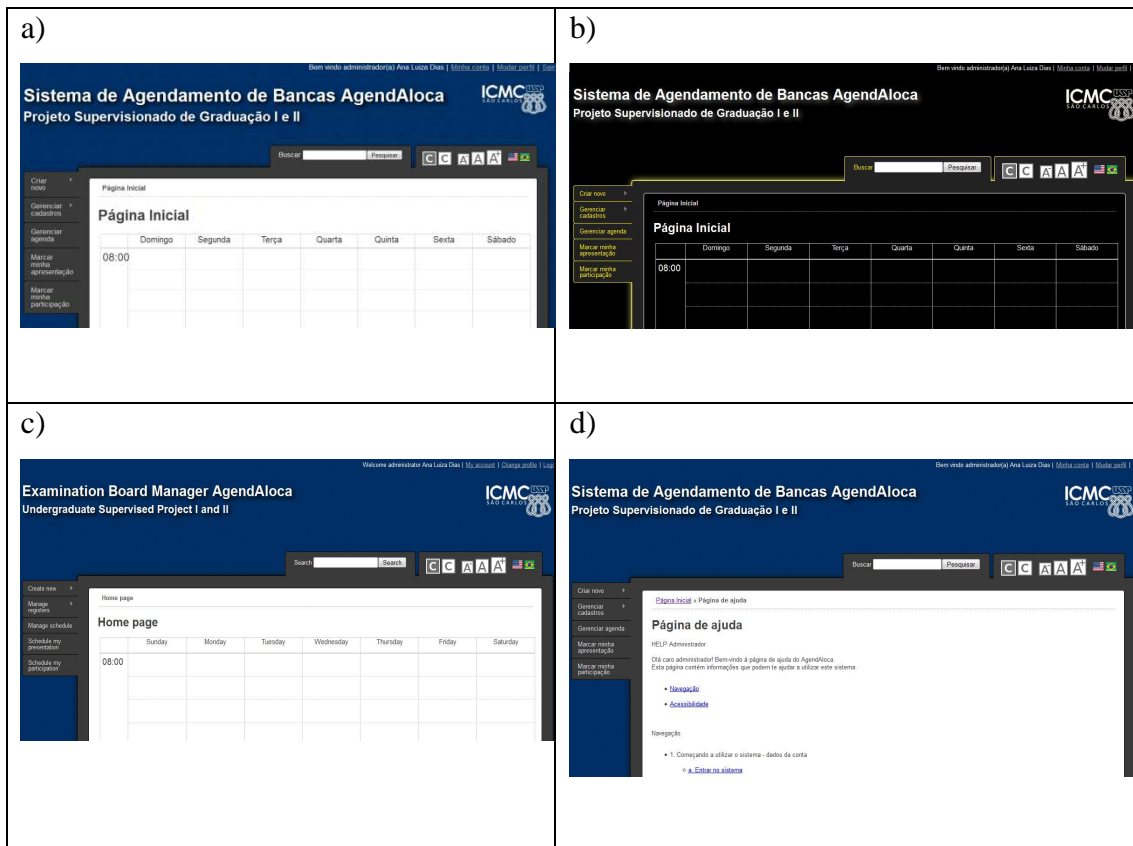


Figura 51: Teste das funcionalidades do AgendAloca.

Alguns problemas percebidos durante as iterações e melhorados posteriormente podem ser citados. Na Figura 52, por exemplo, é possível perceber que o posicionamento do menu anteriormente era horizontal.



Figura 52: Alteração do posicionamento dos menus do AgendAloca.



O menu estava localizado na parte superior do sistema (Figura 52). Porém, ao testar o sistema na ferramenta aDesigner, um problema foi constatado: o usuário que utilizasse o teclado para navegar no sistema iria demorar muito para conseguir chegar ao campo de pesquisa. Para resolver este problema, o menu foi colocado no lado esquerdo do sistema (Figuras 51). Dessa maneira, quando o usuário usa o teclado para navegar no sistema, o campo de pesquisa será acessado primeiro.

Após as quatro iterações no processo PDWAU, tornou-se disponível a versão atual do AgendAloca, que será apresentado na próxima Seção.

## 4.5 O Sistema AgendAloca

Nesta seção a versão final do AgendAloca é apresentada com algumas características e funcionamento básico. Como explicado nas seções anteriores, há duas visões principais: visão do aluno que vai fazer o cadastro de seu trabalho de conclusão de curso, visão do professor que vai escolher uma ou mais bancas cadastradas pelos alunos para marcar participação na avaliação e visão do administrador que tem acesso a todas as opções do sistema.

Nas Figuras 53, 54 e 55 é possível perceber a visão do aluno. Há a agenda que contém todos os horários vagos, previamente cadastrados pelo administrador, e o aluno escolhe apenas um desses horários vagos para agendar sua apresentação. É possível perceber também que de acordo com critérios de usabilidade, fez-se uso de cores para facilitar a visualização por parte dos usuários (Dias, 2009). Portanto, a cor verde são os horários disponíveis, conforme apresentado na Figura 53.



Figura 53: Visão do AgendAloca por parte do aluno para agendar o trabalho.

Após o aluno agendar sua banca, o horário reservado aparecerá em azul para ele, conforme apresentado na Figura 54.



**Figura 54:** Visão do AgendAloca por parte do aluno após agendar seu trabalho.

Já as cores vermelhas são os horários já agendados por outros alunos, os quais não estão mais disponíveis. Ou seja, se o aluno X agendar sua apresentação para ocorrer das 9:00 às 11:00 horas, o aluno Y quando estiver logado no sistema para selecionar uma data para que ele também possa apresentar seu trabalho, esse horário que foi agendado por X estará em vermelho. Assim, somente os horários disponíveis estarão em verde, conforme apresentado na Figura 55.



**Figura 55:** Visão do AgendAloca por parte do aluno.

De acordo com critérios de acessibilidade, a utilização apenas de cores não é suficiente para garantir a interação de usuários com deficiência visual. Além disso, sabe-

se que alguns daltônicos não diferenciam a cor verde da cor vermelha. Pensando nisso, foi adicionada uma *tag* que pode conter um dos termos “Aberto” (significa que o horário ainda está disponível), “Escolhido” (significa que o horário foi escolhido por outra pessoa) e “Escolhi” (significa que o horário foi escolhido pelo próprio aluno). Seguindo este mesmo raciocínio, é possível perceber as interfaces que os professores poderão visualizar, apresentadas nas Figuras 56 e 57.



Figura 56: Visão do AgendAloca por parte do professor para agendar sua participação em uma ou mais bancas.

Na Figura 56 é possível perceber que o professor pode visualizar os horários que foram escolhidos pelos alunos. Quando o professor optar por participar de uma dessas bancas, ele poderá clicar na opção “Participar da banca avaliadora”. Ao clicar nesta opção, abrirá uma tela com informações detalhadas sobre a banca e o professor irá clicar em “Participar”.

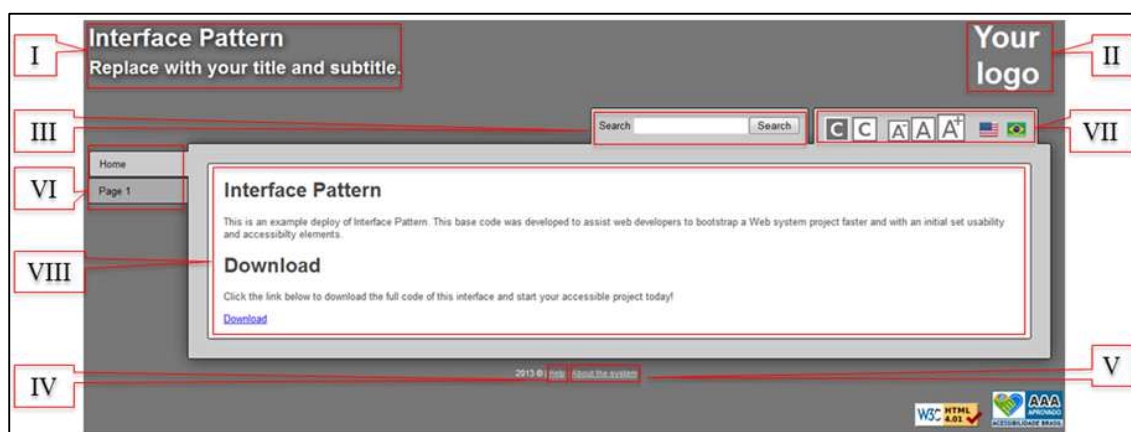


Figura 57: Visão do AgendAloca por parte do professor após agendar sua presença em uma ou mais bancas.

Conforme Figura 57, após selecionar a banca que o professor irá participar, o status passará a ser “Escolhi”, assim como no caso dos alunos. Além disso, a cor passará a ser azul. Como opção posterior, o professor poderá editar as informações dessa agenda, caso haja algum imprevisto e ele precise cancelar a participação na banca.

## 4.6 Reúso da interface do AgendAloca

Como resultado do desenvolvimento do sistema AgendAloca, gerou-se um padrão de interface web para um projeto com usabilidade e acessibilidade. De acordo com Tidwell (2009), um padrão de interface ajuda projetistas no seu trabalho do dia-a-dia, ajudando também toda a indústria a desenvolver melhores ferramentas e paradigmas. O padrão de interface é baseado nas diretrizes de usabilidade e acessibilidade utilizadas no sistema AgendAloca e pode ser processado em um navegador, como apresentado na Figura 58.



**Figura 58: Padrão de interface baseado no AgendAloca.**

Os elementos de interface inseridos considerando-se requisitos de usabilidade e acessibilidade, de acordo com a experiência adquirida no desenvolvimento do sistema AgendAloca, foram:

- I) Barra inicial apresentando o nome do sistema para que os usuários possam identificar rapidamente a intenção do sistema;
- II) Logotipo da universidade e/ou empresa para que o usuário se familiarize rapidamente;
- III) Campo de busca do sistema, em que o usuário pode navegar e pesquisar o que ele deseja;
- IV) *Help* para orientar o usuário em caso de qualquer dúvida;
- V) Informações sobre o sistema, que devem estar sempre disponíveis;



- VI) O menu *dropdown superfish* foi escolhido como um menu acessível, uma vez que tem aspectos que o torna mais acessível e utilizável;
- VII) Características básicas que podem ajudar as pessoas em geral devem ser implementadas, tais como: botões que podem mudar a cor do conteúdo, alterar o tamanho da fonte de conteúdo e mudar o idioma da interface do sistema (Inglês ou Português); e
- VIII) O conteúdo principal, onde o usuário usará para navegar no sistema.

É importante observar que o padrão de interface não é apenas a definição e organização das diretrizes em uma interface, mas também um padrão de código aberto disponível para ser usado por qualquer desenvolvedor<sup>16</sup>. Isso permite que os desenvolvedores se concentrem na lógica de negócios e os principais recursos do seu problema específico, garantindo ainda mais a acessibilidade e usabilidade.

Para validar a utilidade do padrão de interface, o sistema AWMo (*Accessible Web Modeler*), que visa alavancar a colaboração entre os desenvolvedores de software com e sem deficiência visual nas atividades de modelagem, foi desenvolvido. O sistema AWMo pode ser visualizado na Figura 59.

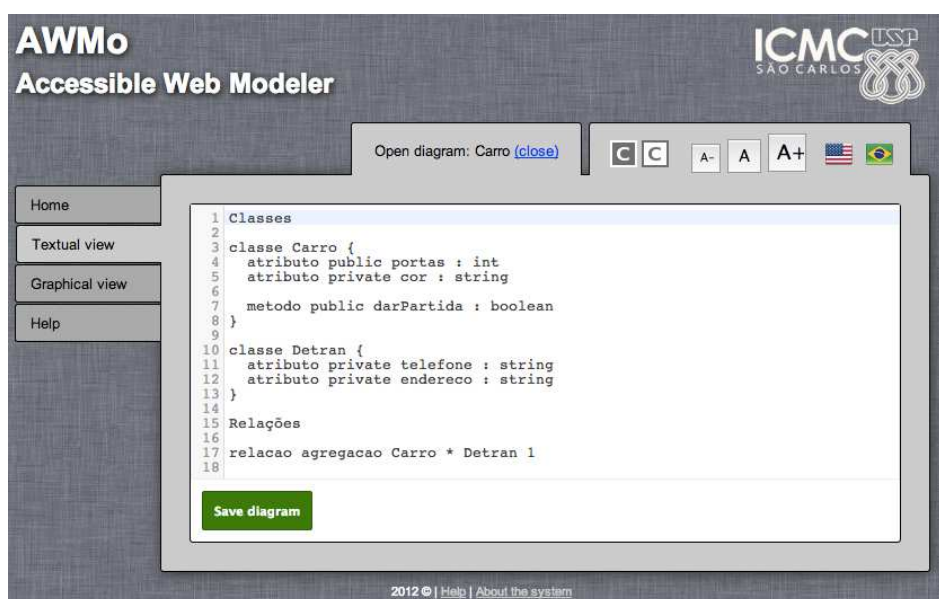


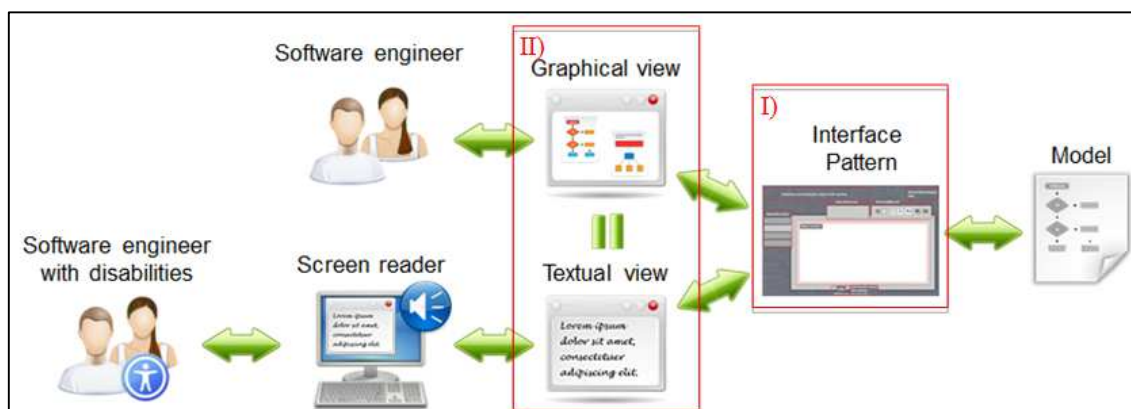
Figura 59: Padrão de interface aplicado no sistema AWMo (Grillo, Fortes e Lucrédio, 2012).

É importante observar que a acessibilidade pode ser vista através de diferentes perspectivas. No caso de AWMo tem-se duas perspectivas:

<sup>16</sup> Disponível em: <http://garapa.intermidia.icmc.usp.br:3000/interface>

- I) A primeira é no uso do padrão de interface que garante a inclusão de características de acessibilidade e usabilidade e;
- II) Posteriormente, no esforço em adicionar uma interface que permite que o usuário com deficiência visual alcance os mesmos resultados que um usuário sem deficiência.

Essas perspectivas podem ser vistas na Figura 60.



**Figura 60: Perspectivas de acessibilidade no uso do padrão de interface aplicado no sistema AWMo (Fortes *et al.*, 2012).**

Ao invés de desenvolver uma interface completamente nova para inserir a acessibilidade, os desenvolvedores optaram por utilizar o padrão de interface. Essa reutilização foi muito útil, pois os desenvolvedores puderam focar os esforços no desenvolvimento das partes principais do sistema.

Ressalta-se ainda que um estudo de viabilidade foi realizado com o sistema AWMo. Dois alunos com deficiências (X e Y) foram convidados a completar uma série de tarefas:

- X é um homem de 41 anos de idade, aluno do terceiro ano do Bacharelado em Sistemas de Informação, um curso de graduação em uma Universidade Pública. Ele também é formado em Matemática. X tem apenas 10% de visão e usa computadores desde 1984 e tem algumas experiências com linguagens de programação como Basic, COBOL, C e Java. Para auxiliar o uso do computador, ele usa o aumento do tamanho da fonte de uma forma que se torne fácil sua leitura.
- Y é um homem de 30 anos, matriculado no terceiro ano do Bacharelado em Sistemas de Informação, em um curso de ensino à distância oferecido por uma Universidade Federal. Y é totalmente cego em decorrência de uma

doença craniana e perdeu a visão quando tinha 15 anos de idade. Y utiliza computadores desde os 7 anos e aprendeu programação há 5 anos. Ele está familiarizado com Java , C # e linguagens de programação *JavaScript* , além de outras tecnologias para o desenvolvimento web, como HTML e CSS. Para auxiliar o uso do computador, ele usa o leitor de tela NVDA (*NonVisual Desktop Access*), que lê cada informação na tela a partir de teclas de atalho para auxiliar a navegação e a interação com o conteúdo.

Esses alunos (X e Y), utilizaram o sistema AWMo e alguns de seus comentários foram:

- Sobre as funcionalidades disponíveis: *"Eu achei o contraste muito legal. Por exemplo, Dev C++ é uma tela branca muito ruim. Normalmente eu preciso alterá-la. Esta cor que você escolheu é muito melhor, porque cansa menos a vista. Há pessoas que gostam do branco, mas no meu caso por exemplo, eu preciso ver o fundo preto. Outra coisa que eu gostei foi a fonte que tem tamanho variável, juntamente com o resto do sistema. Isso é muito legal"*.
- Sobre o sistema: *"O projeto é interessante, porque algumas pessoas que não seriam capazes de fazer um diagrama, agora pode fazê-lo. Ter a opção de forma textual e forma gráfica é muito importante. No caso de alguém que vê pouco, pode usar as duas formas, e no caso de alguém não enxerga nada, pode usar o formulário textual. Isso é fantástico"*.

Além dos usuários terem completado as tarefas passadas a eles, eles deram testemunhos que traz evidências de que a interface AWMo foi bem aceita. Isso é possível por duas razões: primeiro, porque o sistema foi feito usando o padrão de interface que apresenta características de usabilidade e acessibilidade e, segundo, porque os desenvolvedores tiveram mais tempo para se dedicar ao desenvolvimento da funcionalidade do sistema, que também tinha uma proposta de proporcionar acessibilidade a usuários com deficiência.

Ressalta-se então, que com o padrão de interface, é possível desenvolver um sistema web com um baixo número de barreiras de usabilidade e acessibilidade. No entanto, este é o passo inicial para um projeto acessível e usável. Além disso, é necessário que os padrões de usabilidade e acessibilidade sejam considerados ao longo de todo o

desenvolvimento, ou seja, no conteúdo específico do domínio de interesse do aplicativo da web. O resultado parcial obtido neste estudo foi publicado em Fortes *et al.* (2012).

#### 4.7 *Trabalhos relacionados*

Nesta seção são apresentados os trabalhos relacionados ao processo proposto nesta tese. Esses trabalhos foram incluídos aqui, pois de alguma forma, eles propõem o desenvolvimento de sistemas web com usabilidade e/ou acessibilidade, incluindo abordagens, métodos e *frameworks*. Os trabalhos relacionados são apresentados cronologicamente, bem como as diferenças conceituais desses trabalhos com o processo proposto nesta tese.

Kelly *et al.* (2005) propõem um *framework* para auxiliar no uso de diretrizes de acessibilidade, baseado no oferecimento de recomendações de mais alto nível, denominadas camadas (*layers*), as quais são apresentadas a seguir:

- **Contexto:** deve ser verificado qual o contexto de utilização da aplicação web, não existindo uma única solução que atenda a todos os projetos, chamada pelos autores de “*one size-fits all*”;
- **Políticas:** deve abranger áreas como normas técnicas, usabilidade, acessibilidade e as finanças do gerenciamento de projetos;
- **Seleção:** deve haver uma metodologia clara para escolher padrões e as melhores práticas aplicáveis ao contexto particular do projeto;
- **Conformidade:** deve ser garantido que os padrões selecionados sejam bons e que as melhores práticas estejam sendo implementadas corretamente; e
- **Fatores externos:** é necessário considerar uma variedade de fatores externos que influenciam no desenvolvimento. Regras organizacionais e legislação vigente são exemplos comuns de fatores externos que devem ser considerados.

Durante o desenvolvimento desse estudo em 2005, as diretrizes WCAG ainda estavam na versão 1.0 e, apesar dos autores reconhecerem a importância dessas diretrizes, eles apresentam um resumo das principais discussões e problemas em relação ao seu uso, tais como a natureza teórica e a ambiguidade e complexidade das diretrizes, necessitando um nível mínimo de entendimento de acessibilidade.

Kelly *et al.* (2005) apresentam uma percepção de que a acessibilidade não pode ser tratada de forma isolada, mas com a inclusão da usabilidade e fatores locais. Além disso, eles valorizam a importância da avaliação manual de acessibilidade por parte do especialista para um bom resultado do sistema desenvolvido. No processo PDWAU formalizado nesta tese, a inspeção por parte do especialista foi considerada, assim como sugerido por Kelly *et al.* (2005), além de apresentar as tarefas de usabilidade e acessibilidade a serem realizadas dentro de um processo genérico. Adicionalmente, trabalha-se com o WCAG 2.0 bem como heurísticas e padrões web, assuntos não tratados no *framework* formalizado por Kelly *et al.* (2005).

Melo e Baranauskas (2006) apresentam um modelo de processo para design inclusivo de sistemas de informação na web, focando na participação do usuário em espaços colaborativos de desenvolvimento para a concepção, proposição e avaliação de sistemas e suas interfaces. Esse modelo de processo utiliza como referência o metamodelo de processo sugerido em OMG (*Object Management Group*) (2005), composto por atividades em que são inseridas considerações de acessibilidade. O processo proposto foi baseado nas seguintes premissas:

- Promoção da qualidade no uso de sistemas de informação na web a partir de um entendimento da inclusão e das diferenças dos usuários;
- Entendimento dos sistemas de informação de forma abrangente;
- Consideração da diversidade de contextos e situações de uso da tecnologia;
- Reconhecimento e valorização das diferenças entre os usuários em suas capacidades perceptuais, cognitivas e motoras;
- Apoio no desenvolvimento sistemático e iterativo de sistemas de informação acessíveis na web, com a utilização de métodos e técnicas adequados;
- Participação dos usuários em espaços colaborativos de desenvolvimento: na concepção, na proposição e na avaliação de sistemas web inclusivos;
- Promoção da participação das partes interessadas com base na igualdade de direitos e respeito mútuos; e
- Consideração de recomendações do W3C no desenvolvimento web.

Em relação à organização de suas atividades, o modelo de processo apresenta uma divisão das atividades em nove disciplinas: clarificação do problema; requisitos de sistema de informação; design de interface do usuário; design de sistema de software;

implementação; verificação e validação; avaliação de interface de usuário; implantação; e gerência de projeto.

Melo e Baranauskas (2006) apresentam as vantagens do design inclusivo de sistemas de informação na web. As atividades relativas à acessibilidade sugeridas pelas autoras, tais como avaliação de acessibilidade contínua e elicitação de requisitos foram consideradas no processo PDWAU. No entanto, há uma distinção conceitual entre esse modelo de processo e o processo proposto nesta tese, não havendo conflitos, mas sim um nível de detalhamento e escopo diferente, visto que o processo desta tese tem um foco complementar no processo genérico de Pressman e Lowe (2009). Adicionalmente, assim como a proposta de Kelly *et al.* (2005), critérios de sucesso do WCAG 2.0 bem como heurísticas e padrões web são assuntos não tratados na abordagem de Melo e Baranauskas (2006).

Abou-zahra (2008) apresenta um modelo de desenvolvimento focado na avaliação contínua de acessibilidade. Para isso, correções são feitas por etapas a partir de verificações nos desenvolvimentos em diferentes fases do processo de desenvolvimento web. Desse modo, as avaliações devem percorrer todo o período de existência das aplicações, a fim de garantir procedência nos requisitos. Para essas avaliações o autor sugere ainda que haja a combinação de diferentes técnicas de inspeções e testes de acessibilidade web, para que se tenham resultados mais efetivos no apontamento de barreiras para o acesso de pessoas com deficiência. As técnicas sugeridas pelo autor são: avaliação com ferramentas automáticas, inspeção realizada por especialista e teste com usuário.

A importância da avaliação de acessibilidade em diferentes fases do processo de desenvolvimento web é ressaltada por Abou-zahra (2008). No processo PDWAU formalizado nesta tese, há um monitoramento de acessibilidade com diferentes modalidades de verificações para garantia de qualidade no provimento de artefatos acessíveis. O modelo proposto por Abou-zahra (2008) difere do foco do processo proposto, não apresentando, por exemplo, em quais momentos deve haver preocupações com a inserção de requisitos de usabilidade e acessibilidade no desenvolvimento de sistemas web.

Maia (2010) apresenta o Modelo de Tarefas de Acessibilidade (MTA), um processo para o desenvolvimento de sistemas web acessíveis, o qual é composto por uma coleção de tarefas de acessibilidade inseridas nos subprocessos da norma ISO/IEC 12207

(1998). O MTA considera a acessibilidade em todas as etapas do desenvolvimento de sistemas web. Adicionalmente, são especificadas também, nas tarefas desse modelo, associações com as diretrizes WCAG 2.0 (nível A).

Como exemplificação do que é tratado no MTA, são definidas duas importantes tarefas consecutivas. A primeira é o projeto de interfaces acessíveis, sendo subdividida em três subtarefas: i) definir os elementos que irão compor a interface; ii) estabelecer a disposição dos elementos na interface; e iii) definir as cores dos elementos. A segunda tarefa é o projeto navegacional acessível, que tem como objetivo fazer com que os elementos que compõem a interface possam ser localizados, navegados e compreendidos corretamente.

O trabalho proposto por Maia (2010) apresenta uma rica coleção de tarefas de acessibilidade, com apoio ferramental. No entanto, ele está relacionado diretamente ao uso da Norma ISO/IEC 12207 e a não utilização da ideia do compartilhamento de conhecimento e reúso de código. No processo PDWAU formalizado nesta tese, um padrão de interface foi gerado, podendo ser reusado para o desenvolvimento de outros sistemas web. Adicionalmente, diretrizes WCAG 2.0 de todos os níveis e princípios de usabilidade fazem parte das tarefas e subtarefas do processo para o desenvolvimento de sistemas web.

Nietzio *et al.*(2010) apresentam uma abordagem colaborativa para projetos web de governo eletrônico, chamada *eGovMon*, que realiza aferição de alto nível com testes e melhorias de acessibilidade, que são implementadas como correções em um sistema. Essa abordagem conta com os seguintes pilares:

- Um verificador on-line, chamado *eAccessibility Checker*, que fornece avaliações detalhadas de acessibilidade de páginas web;
- Um fórum, chamado *eGovMon Forum*, para mantenedores de sites de municípios, especialistas em acessibilidade web e desenvolvedores; e
- A publicação de relatórios de *benchmarking* de sites governamentais, que permitem a avaliação dos progressos (ou regressos) ao longo do tempo.

Nietzio *et al.*(2010) enfatiza a necessidade da participação de especialistas em acessibilidade em discussões sobre experiências e dúvidas da equipe de desenvolvimento. No entanto, o foco do trabalho é em E-Gov, enquanto o processo PDWAU apresentado nesta tese pode ser aplicado a qualquer domínio de sistema.

López (2010) ressalta que para tratar acessibilidade, não basta apenas seguir estritamente as diretrizes WCAG em relação a aspectos técnicos de implementação. É necessário considerar a arquitetura das aplicações e integrar usabilidade e acessibilidade no desenvolvimento de software web. Para tanto, a autora propõe um *framework* chamado AWA (*Accessibility for Web Application*), o qual oferece um suporte metodológico flexível para o desenvolvimento de aplicações web acessíveis. Suas regras prezam pela incorporação do design centrado no usuário e design inclusivo em todo o ciclo de vida do desenvolvimento de sistemas web.

O AWA é estruturado em modelos e regras, com base nas diretrizes WCAG, sendo composto por mecanismos (chamados de *AWA\_MecanismosWCAG*) e requisitos (chamados de *AWA\_RequisitosWCAG*). Estes são considerados como atuantes no desenvolvimento desde o início das atividades de análise e projeto, incluindo, intrinsecamente, requisitos na modelagem de aplicações web acessíveis.

López (2010) considera o tratamento da arquitetura das aplicações como uma atividade importante, integrando usabilidade e acessibilidade no desenvolvimento de sistemas web. No entanto, os modelos e regras propostos são muito complexos e requerem conhecimento mais profundo para a sua utilização. Além disso, a autora não considera heurísticas nem padrões web como princípios de usabilidade a serem unificados com as diretrizes de acessibilidade, como ocorre no processo PDWAU.

Bittar (2013) propõe uma abordagem que viabilize o apoio a boas práticas para desenvolvimento de aplicações web acessíveis. A abordagem apresentada tem como objetivo fornecer uma maneira sistemática para aplicar as diretrizes de acessibilidade e gerenciar os resultados alcançados durante o desenvolvimento web. Tal abordagem apresenta atividades, papéis, entradas e saídas que devem ser seguidas para que a acessibilidade seja considerada. Essa abordagem, diferentemente do processo PDWAU, não considera princípios de usabilidade nem se baseia em um processo genérico de desenvolvimento.

Assim, com a verificação desses trabalhos relacionados, bem como de outros trabalhos encontrados na literatura e tratados ao longo desta tese, fica nítida a percepção de que há diferentes objetivos e focos em cada estratégia proposta para o desenvolvimento de sistemas web com acessibilidade e usabilidade, tratando as várias questões (frequentemente complexas) para melhoria do cenário atual. É importante ressaltar que foi observado também que há uma evolução contínua nas propostas, sendo que em todas



existem contribuições oportunas e promissoras, em diferentes escopos, sempre visando aprimorar a sistemática do desenvolvimento web, considerando questões de acessibilidade e/ou usabilidade.

#### **4.8 Considerações finais**

Com o apoio de princípios teóricos, estudos sobre usabilidade, acessibilidade e engenharia web, foi possível formalizar o processo PDWAU para o desenvolvimento de sistemas web com foco em acessibilidade e usabilidade. Além disso, um estudo de caso foi realizado a fim de perceber indícios de que o processo é exequível. Esse estudo gerou um sistema web que teve sua interface disponibilizada para reuso, chamada de padrão de interface (Figura 58), posteriormente utilizada e testada com pessoas com deficiência visual em um sistema chamado AWMo (Grillo, Fortes e Lucrédio, 2012).

Considerando-se a dificuldade prática de avaliar diretamente o processo PDWAU formalizado nesta tese, optou-se por avaliar o produto gerado, no caso, o sistema AgendAloca. Para isso, foi desenvolvido um experimento e realizado um estudo de viabilidade, os quais serão detalhados nos Capítulos 5 e 6, respectivamente.



---

# *Avaliação experimental do PDWAU*

---

## *5.1 Considerações iniciais*

A mudança contínua da perspectiva de uso dos sistemas web é um fato irreversível. Conteúdos que eram simples de navegar estão se tornando mais e mais complexos devido às atualizações de muitos componentes dinâmicos incluídos nesses sistemas (Lunn e Harper, 2011). Uma das razões percebidas para estas mudanças é a preocupação da adequação dos conteúdos às características, diversidades e necessidades dos usuários, bem como a forma como eles interagem com a informação e serviços disponíveis na web (Luna *et al.*, 2010).

Percebe-se que, mesmo com a tecnologia avançada para promover várias formas de interatividade na web, pesquisas sobre inclusão digital para pessoas com deficiência têm demonstrado não só a falta de soluções acessíveis em universidades, ambiente de negócios e de governo, mas também a falta de usabilidade em sistemas web. Esses conceitos deveriam receber atenção especial por parte dos desenvolvedores, uma vez que a interface é a parte visível do sistema para o usuário, através da qual ele se comunica para realizar suas tarefas (Freire, Goularte e Fortes, 2007; Lazar e Olalere, 2011).

Nesta tese foi proposto um processo de desenvolvimento de sistemas web com foco em acessibilidade e usabilidade, apresentado no Capítulo 4. Para que o processo PDWAU pudesse ser avaliado, optou-se por comparar o sistema AgendAloca (desenvolvido com o apoio do processo PDWAU) com o sistema Agendamento (sistema com as mesmas funcionalidades do sistema AgendAloca, mas desenvolvido com um processo *ad-hoc* e sem preocupações quanto a requisitos de usabilidade e acessibilidade). Para isso, uma avaliação experimental foi realizada, com o objetivo de buscar indícios de que o processo PDWAU pode ser útil e eficaz no apoio ao desenvolvimento de sistemas web com acessibilidade e usabilidade.

Este capítulo está organizado da seguinte maneira: na Seção 5.2 apresenta-se o planejamento da avaliação experimental do PDWAU, enquanto sua execução é apresentada na Seção 5.3. Na Seção 5.4 apresenta-se um método concebido a partir da realização do experimento, que pode ser útil para avaliar, comparar e melhorar a acessibilidade e a usabilidade de sistemas web existentes (legados). Por fim, as considerações finais são apresentadas na Seção 5.5.

## ***5.2 Planejamento da avaliação experimental***

De acordo com Shull, Carver e Travassos (2001), o primeiro estudo que deve ser feito para avaliar uma nova abordagem é um estudo de viabilidade, chamado aqui de experimento<sup>17</sup>, que visa verificar se esta nova tecnologia é viável e se o tempo gasto é bem utilizado. Estes estudos são normalmente realizados na academia, uma vez que permitem que as novas tecnologias sejam testadas antes de ser transferidas para o setor que irá usá-las no dia-a-dia. Para a realização deste experimento, algumas definições foram necessárias e são apresentadas a seguir.

### **5.2.1 Definição do experimento**

Para o planejamento do experimento, foram necessárias as seguintes definições:

- Objeto de estudo: PDWAU - Processo de desenvolvimento de sistemas web com foco em acessibilidade e usabilidade.

---

<sup>17</sup> Devido ao número de usuários que participaram do experimento, todo o detalhamento do Capítulo 5 pode ser considerado como um Estudo empírico.

- Propósito: avaliar o resultado da utilização do PDWAU para testar sua eficácia no desenvolvimento de sistemas com acessibilidade e usabilidade, comparando o sistema AgendAloca (desenvolvido utilizando o processo PDWAU) com outro sistema com o mesmo propósito e as mesmas funcionalidades, o Agendamento (que foi desenvolvido com um processo *ad-hoc*).
- Foco quantitativo: o foco primário foi o tempo de execução das tarefas propostas e a quantidade de erros cometidos durante a interação dos sujeitos do experimento com ambos os sistemas. O foco secundário foi a quantidade de tarefas concluídas com sucesso e o número de cliques realizados durante cada tarefa.
- Foco qualitativo: o foco qualitativo primário foi identificar as impressões e o sentimento dos sujeitos do experimento sobre o uso dos sistemas. O foco secundário foi avaliar a qualidade da interação do sujeito do experimento com os sistemas, observando-se a utilização do apoio de características de acessibilidade e usabilidade.
- Perspectiva: sob o ponto de vista de especialistas e de usuários a partir das tarefas executadas. Os sujeitos do experimento tiveram acesso aos sistemas web e puderam interagir da forma que lhes foi conveniente, realizando as tarefas propostas neste experimento.
- Contexto: o experimento foi realizado com alunos de graduação do ICMC-USP, em um ambiente controlado (Laboratório Intermídia - ICMC-USP) e com tempo controlado.

Além do foco quantitativo, houve um foco qualitativo obtido por meio da análise de uma pesquisa de opinião realizada tanto para os sujeitos que utilizaram o sistema Agendamento quanto para os que utilizaram o sistema AgendAloca. Os dados foram obtidos por meio de questionários pré-sessão e pós-sessão, aplicação do método *think aloud* e entrevista.

Adicionalmente, foi também realizada uma avaliação sob o ponto de vista de especialista, que realizou uma análise das dificuldades dos sujeitos na execução do experimento para posterior refinamento de seu planejamento e operação. Para isso, o

vídeo da interação, da face do sujeito e o áudio foi analisado. Isso foi possível, pois toda a interação dos sujeitos com os sistemas web foi gravada pelo software *Morae*<sup>18</sup>.

Houve também uma avaliação sob o ponto de vista de inspeção (em que o especialista verificou se o sistema atendia a uma lista de requisitos), bem como sob o ponto de vista de ferramenta (em que o especialista avaliou o sistema utilizando ferramentas de avaliação online).

A realização da análise qualitativa ajudou na identificação de evidências que comprovassem a utilidade das características de usabilidade e acessibilidade adicionadas ao sistema AgendAloca.

### 5.2.2 Definição das hipóteses

Para o planejamento do experimento, foram definidas as hipóteses nulas e as hipóteses alternativas.

- Hipóteses nulas (HN):

**HN1** – O tempo de execução das tarefas executadas do sistema AgendAloca é maior ao tempo de execução das tarefas executadas pelo sistema SAB. A seguinte equação representa esta hipótese:  $TE_{AG} > TE_{SAB}$ .

**HN2** – A quantidade de erros na execução de tarefas do sistema AgendAloca é maior à quantidade de erros na execução de tarefas do sistema SAB. A seguinte equação representa esta hipótese:  $QE_{AG} > QE_{SAB}$ .

**HN3** – A quantidade de tarefas concluídas com sucesso do sistema AgendAloca é menor à quantidade de tarefas concluídas com sucesso do sistema SAB. A seguinte equação representa esta hipótese:  $QT_{AG} < QT_{SAB}$ .

**HN4** – O número de cliques para uma tarefa do sistema AgendAloca é maior ao número de cliques para uma tarefa do sistema SAB. A seguinte equação representa esta hipótese:  $NQ_{AG} > NQ_{SAB}$ .

- Hipóteses alternativas (HA):

**HA1** – O tempo de execução das tarefas executadas do sistema AgendAloca é menor ou igual ao tempo de execução das tarefas executadas pelo sistema

---

<sup>18</sup> Disponível em: <http://www.techsmith.com/download/licenses/morae.asp>

Agendamento - SAB (para as hipóteses, chamaremos de SAB o sistema de Agendamento). A seguinte equação representa esta hipótese:  $TE_{AG} \leq TE_{SAB}$ .

**HA2** – A quantidade de erros na execução de tarefas do sistema AgendAloca é menor ou igual à quantidade de erros na execução de tarefas do sistema SAB. A seguinte equação representa esta hipótese:  $QE_{AG} \leq QE_{SAB}$ .

**HA3** – A quantidade de tarefas concluídas com sucesso do sistema AgendAloca é maior ou igual à quantidade de tarefas concluídas com sucesso do sistema SAB. A seguinte equação representa esta hipótese:  $QT_{AG} \geq QT_{SAB}$ .

**HA4** – O número de cliques para uma tarefa do sistema AgendAloca é menor ou igual ao número de cliques para uma tarefa do sistema SAB. A seguinte equação representa esta hipótese:  $NQ_{AG} \leq NQ_{SAB}$ .

### 5.2.3 Definição do projeto do experimento

Tratou-se de um experimento com um fator (o processo de desenvolvimento) e dois tratamentos (processo de desenvolvimento de sistemas web com acessibilidade e usabilidade e processo *ad-hoc*).

Os sujeitos do experimento utilizaram o sistema AgendAloca e o sistema Agendamento para realizar as tarefas, tendo acesso às tarefas mais comuns/básicas e mais importantes do sistema. Para isso, eles receberam treinamento adequado para cada uma das tarefas utilizando ambos os sistemas.

**Tabela 8: Projeto do experimento.**

<b>Etapas\Sujeitos</b>	<b>AgendAloca</b>	<b>Agendamento</b>
Sujeito 1 (aluno)	1º	2º
Sujeito 2 (aluno)	2º	1º

Na Tabela 8 é apresentado o plano de atribuição dos sistemas na ordem em que foram utilizados pelos sujeitos do experimento. Eles realizaram as tarefas uma a uma, em horário específico agendado previamente e foram alocados nas tarefas aleatoriamente.

### 5.2.3.1 Seleção dos sujeitos

A seleção dos sujeitos do experimento foi feita por conveniência entre alunos por meio de um convite que foi enviado por e-mail. Participaram desta pesquisa, alunos da Graduação, sendo essa população do ICMC/USP, localizado na cidade de São Carlos/SP, público alvo escolhido devido à motivação principal dos sistemas web desenvolvido (apoiá-los durante o final de curso para agendar e organizar as monografias de final de curso e apresentações).

Se mais de doze sujeitos se candidatassem, a equipe de pesquisa daria preferência aos sujeitos com algum tipo de deficiência, os mais velhos, os menos escolarizados e os de classe social mais baixa e que tivessem menos contato com o computador. Essa prioridade seria seguida, pois acredita-se que esses perfis citados podem utilizar em maior escala as características de acessibilidade e usabilidade presentes no sistema. Para a exclusão, os sujeitos que fossem mais jovens e também os experientes na área de IHC estariam na lista de espera.

### 5.2.3.2 Seleção das variáveis

- **Variáveis independentes:**

- ✓ Processo PDWAU (tratamento)
- ✓ Processo *ad-hoc* (tratamento)
- ✓ AgendAloca (objeto experimental)
- ✓ Agendamento - SAB (objeto experimental)
- ✓ Experiência dos sujeitos no uso de sistemas de agendamento de bancas (objeto experimental)
- ✓ Tarefas realizadas pelos sujeitos (contexto)

- **Variáveis dependentes:**

- ✓ Tempo de execução das tarefas executadas
- ✓ Quantidade de erros na execução de tarefas
- ✓ Quantidade de tarefas concluídas com sucesso
- ✓ Número de cliques para uma tarefa



### 5.2.3.3 Ameaças à validade dos resultados do experimento

As ameaças à validade dos resultados do experimento resumem-se em: validade interna, validade externa e validade de conclusão que são detalhadas a seguir.

- **Validade interna:**
  - ✓ Efeitos de treinamento: o treinamento de um dos sistemas utilizados pode ser melhor do que do outro. Para mitigar este possível erro, foi preparado um mesmo treinamento sobre as tarefas realizadas para os sistemas em questão.
  - ✓ Experiência do aluno: a experiência dos alunos com a utilização de sistemas web principalmente para agendamento de bancas e com a área de acessibilidade e usabilidade são fatores que podem influenciar os resultados do experimento. Neste experimento este fator foi mitigado por meio de treinamento e por meio da alocação aleatória dos sujeitos na ordem dos sistemas a serem utilizados.
  - ✓ Serviços utilizados: a complexidade dos sistemas utilizados pode influenciar o resultado. Para mitigar este fator, os sistemas utilizados nesse experimento não foram desenvolvidos pelos condutores do experimento para não haver o risco de que eles favoreçam os resultados do processo avaliado.
  - ✓ Produtividade sob avaliação: pessoas tendem a ser mais produtivas quando estão sob avaliação. Esse fator pode influenciar os resultados porque em geral sujeitos que realizam experimentos tendem a pensar que estão sendo avaliados pelos resultados e não pela participação. Neste caso, o experimento não foi realizado no contexto de uma disciplina, mas os sujeitos foram voluntários que desejaram colaborar com a pesquisa realizada em sua Instituição.
  
- **Validade externa**: a população dos participantes não é representativa estatisticamente, pois se trata de um grupo homogêneo de alunos. A aplicação do experimento foi feita em ambiente controlado. Por esses motivos, a validade externa deste experimento pode ficar comprometida e os resultados e conclusões não podem ser generalizados. Embora acredita-

se que os indícios que foram encontrados neste estudo, sejam de grande validade para a área e para a pesquisa em si.

- **Validade de conclusão:** a quantidade de amostras coletadas e a homogeneidade da amostra são problemas para a validade deste experimento, pois a quantidade de sujeitos não é o ideal do ponto de vista estatístico e por serem todos alunos da mesma Instituição.

#### 5.2.3.4 Riscos e benefícios

Sabe-se que todas as pesquisas envolvendo seres humanos possuem algum tipo de risco aos participantes. Para este experimento, acredita-se que houve dois riscos principais: a exposição à radiação e a violação da privacidade. Para amenizar o risco a exposição à radiação, os participantes não ficaram mais do que 4 (quatro) horas em frente ao computador e esse tempo pode ser considerado um tempo desprezível. Já para amenizar o risco de violação da privacidade dos participantes, os dados coletados durante o experimento não foram expostos na web e somente a equipe de pesquisa teve acesso aos dados para análise posterior. Além disso, os dados publicados não identificarão os participantes devido ao uso de siglas e números para se referir a eles.

Os participantes podem se sentir beneficiados por participarem da pesquisa, uma vez que eles foram convidados a participar de uma atividade especial em que podem interagir com um sistema que facilitará seu trabalho no final do semestre. Também estarão utilizando um sistema em que a Instituição tem interesse de utilizar não apenas para esta pesquisa, mas durante o ano letivo. Além disso, podem se sentir beneficiados por colaborar com o avanço das pesquisas, levando-os conseqüentemente à satisfação interna e também por saber que pessoas com deficiências serão beneficiadas com os resultados desta pesquisa.

Ressalta-se que foi informado a todos os participantes que o experimento se deu ao fato de perceber quão usável, fácil e contextualizado é o sistema pelo ponto de vista dos participantes, portanto, em nenhum momento a avaliação foi baseada no comportamento dos sujeitos.

O grupo de pesquisadores esteve apto a suspender a pesquisa a qualquer momento se percebesse que algum participante da pesquisa estivesse constrangido ou desconfortável, por participar do experimento, bem como se fosse observado algum risco

ou danos à saúde do sujeito participante da pesquisa. Adicionalmente, o grupo de pesquisadores comprometeu-se com o Comitê de Ética, informá-lo de todos os efeitos adversos ou fatos relevantes que alterassem o curso normal do estudo, além de cumprir todas as consequências que viessem a ter, assumindo a responsabilidade de dar assistência integral às complicações e danos decorrentes dos riscos previstos.

Os sujeitos da pesquisa que viessem a sofrer qualquer tipo de dano resultante de sua participação, além do direito à assistência integral, teriam direito à indenização (cobertura material, em reparação a dano imediato ou tardio, causado pela pesquisa ao ser humano a ela submetido). Não foi exigida do participante, sob qualquer argumento, renúncia ao direito à indenização por dano. Como também, pode-se afirmar que o formulário do consentimento livre e esclarecido não continha nenhuma ressalva que afastasse essa responsabilidade ou que implicasse ao sujeito da pesquisa abrir mão de seus direitos legais, incluindo o direito de procurar obter indenização por danos eventuais.

Os responsáveis pela pesquisa asseguraram a confidencialidade e a privacidade, a proteção da imagem e a não estigmatização, garantindo a não utilização das informações em prejuízo das pessoas e/ou das comunidades, inclusive em termos de autoestima, de prestígio e/ou econômico – financeiro.

#### **5.2.4 Definição da operação do experimento**

Para a operação do experimento, foi necessário preparar a programação para que tudo ocorresse conforme o planejado.

##### **5.2.4.1 Descrição da instrumentação**

Os materiais fornecidos aos sujeitos para a realização do experimento foram:

- Termo de consentimento livre e esclarecido (APÊNDICE C).
- Questionário pré-sessão: formulário de pesquisa sobre o perfil dos sujeitos, a experiência deles em relação ao uso de sistemas web e o conhecimento sobre usabilidade e acessibilidade (APÊNDICE D).
- Descrição das tarefas para a realização do experimento (APÊNDICE E).
- Questionário pós-sessão: formulário para coleta de dados para cada sistema utilizado – AgendAloca e Agendamento (APÊNDICE F).

#### **5.2.4.2 Preparação**

A equipe de pesquisa fez contato com alunos do ICMC para convidá-los a participar voluntariamente do experimento. Cada voluntário assinou o termo de consentimento livre e esclarecido.

Antes da execução do experimento os sujeitos responderam ao questionário pré-sessão informando alguns dados e características pessoais e logo após receberam treinamento adequado para utilizar os sistemas AgendAloca e Agendamento. O treinamento foi realizado individualmente e imediatamente antes da execução do experimento.

Após o treinamento, os sujeitos receberam as tarefas que foram executadas nos sistemas web. Eles tiveram no máximo duas horas para realizar as atividades designadas em cada sistema web, totalizando quatro horas no total e responderam ao questionário pós-sessão ao final da interação de ambos os sistemas para coletar suas opiniões durante a execução das tarefas.

#### **5.2.4.3 Execução**

Os sujeitos executaram o experimento individualmente sob a supervisão dos condutores. Optou-se pela execução individual para aumentar o controle sobre o processo e para que o risco de falhas na execução fosse menor. Se alguma falha fosse encontrada no início do experimento seria possível corrigi-la para que os dados coletados dos outros participantes não fossem invalidados.

#### **5.2.4.4 Validação dos dados**

Os formulários preenchidos pelos sujeitos participantes foram conferidos para verificar se foram preenchidos corretamente. Se houvesse alguma característica durante a execução do experimento em que os condutores julgassem crítica, os dados coletados do participante em questão seriam excluídos da análise/estatística final.

Antes de iniciar a próxima seção, é importante destacar que esse experimento, como teve a participação de usuários, foi submetido e autorizado pelo Comitê de Ética em Pesquisa envolvendo Seres Humanos - Escola de Artes, Ciências e Humanidades EACH-USP, com o título “Um método para garantir a acessibilidade no desenvolvimento de sistemas web” e **CAAE: 05700912.5.0000.5390**. O parecer substanciado e aprovado pode ser encontrado no ANEXO V.

### 5.3 Execução da avaliação experimental

O experimento foi executado de acordo com o planejamento definido na Seção 5.2. O detalhamento do experimento realizado com foco qualitativo e quantitativo será apresentado a seguir, na seguinte ordem: realização da avaliação experimental sob o ponto de vista de inspeção, de ferramenta, de usuário e de especialista.

#### 5.3.1 Realização da avaliação sob o ponto de vista de inspeção

A avaliação sob o ponto de vista de inspeção consiste em uma avaliação de conformidade com as diretrizes do WCAG 2.0, sendo necessário que um especialista realize essa avaliação. Nesse experimento, o especialista foi seguindo os critérios de sucesso de cada diretriz, atribuindo uma nota de 0 a 10; obtendo ao final uma média para cada diretriz.

O resultado foi relatado, em uma escala percentual de quantos critérios de sucesso foram atendidos no sistema para cada uma das diretrizes do WCAG 2.0, conforme apresentado na Tabela 9.

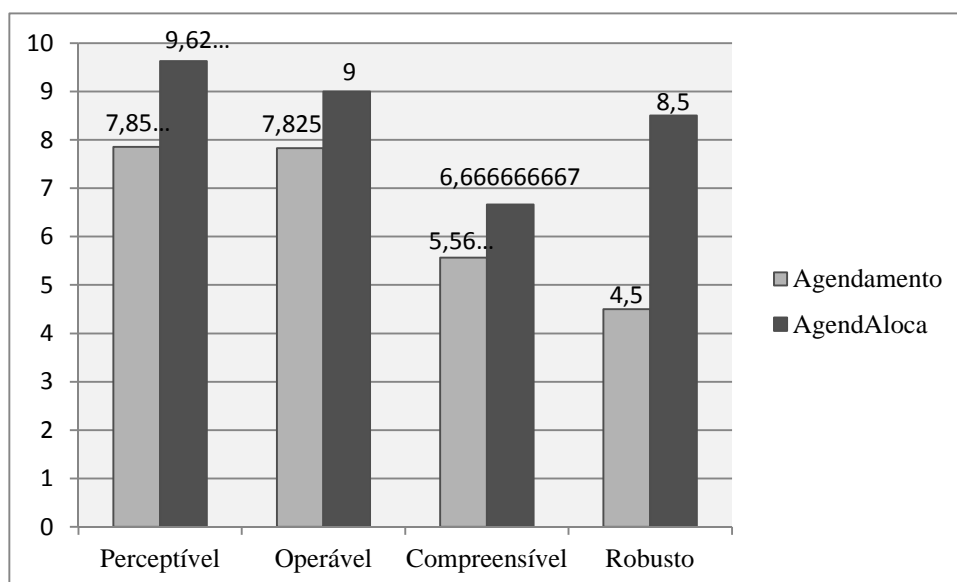
**Tabela 9: Resultado da avaliação sob o ponto de vista de inspeção nos sistemas Agendamento e AgendAloca.**

<b>Princípios</b>	<b>Diretrizes</b>	<b>Agendamento</b>	<b>AgendAloca</b>
Perceptível	1.1 Alternativas textuais	100%	100%
	1.2 Mídias temporais <sup>19</sup>	0	0
	1.3 Adaptabilidade	63,3%	100%
	1.4 Distinguíveis	72,2%	88,8%
<b>Média final - Perceptível:</b>		<b>78,5%</b>	<b>96,2%</b>
Operável	2.1 Acessível pelo teclado	100%	100%
	2.2 Tempo suficiente	75%	75%
	2.3 Apreensibilidade	100%	100%
	2.4 Navegabilidade	38%	85%
<b>Média final - Operável:</b>		<b>75,7%</b>	<b>90%</b>
Compreensível	3.1 Legível e compreensível	21%	33,3%
	3.2 Previsibilidade	92%	100%
	3.3 Assistência de entrada	53%	66,6%
<b>Média final - Compreensível:</b>		<b>55,6%</b>	<b>66,6%</b>
Robusto	4.1 Compatibilidade	45%	85%
<b>Média final - Robusto:</b>		<b>45%</b>	<b>85%</b>
<b><u>Média total de acessibilidade</u></b>		<b><u>63,7%</u></b>	<b><u>84,45%</u></b>

<sup>19</sup> Guideline 1.2 recebeu nota 0 porque o sistema Agendamento não contém elementos em áudio/vídeo.

Com a avaliação realizada pelo especialista, pôde-se observar que o sistema Agendamento não satisfaz a 36,3% de critérios de sucesso de acessibilidade. Isto é justificável, já que o objetivo do desenvolvimento era entregar novas funcionalidades sem entendimento ou conhecimento sobre as diretrizes de acessibilidade. Os poucos recursos de acessibilidade que podem ser percebidos são resultados de módulos nativos do CMS. Por exemplo, o uso obrigatório de um texto alternativo nas imagens.

Já o sistema AgendAloca que foi desenvolvido seguindo o processo PDWAU, conseguiu satisfazer a um maior número de critérios de sucesso de acessibilidade, desatendendo apenas 15,55% de critérios de sucesso. Essa diferença pode também ser vista na Figura 61, em que é possível perceber a evolução do sistema Agendamento para o sistema AgendAloca, de acordo com os quatro princípios do WCAG 2.0.



**Figura 61:** Comparação entre os resultados obtidos nos sistemas Agendamento e AgendAloca após a avaliação sob o ponto de vista de inspeção.

Após aplicar a inspeção em ambos os sistemas, o próximo passo é passá-los por uma avaliação das ferramentas online, que será explicada na próxima subseção.

### 5.3.2 Realização da avaliação sob o ponto de vista de ferramenta

Como o resultado da aplicação de ferramentas online é extenso, um breve resumo será apresentado. Foi utilizada a ferramenta DaSilva e os resultados obtidos podem ser encontrados na Tabela 10.

**Tabela 10: Resultado da avaliação dos sistemas  
Agendamento e AgendAloca sob o ponto de vista de ferramenta.**

<b>DaSilva</b>	<b>Prioridade 1</b>	<b>Prioridade 2</b>	<b>Prioridade 3</b>	<b>Total de erros</b>
<b>Agendamento</b>	11	4	1	16
<b>AgendAloca</b>	0	0	0	0

De acordo com a Tabela 10, a ferramenta DaSilva relatou que o sistema Agendamento apresentou 16 erros. Já para o sistema AgendAloca, nenhum erro foi apresentado, possibilitando o reconhecimento do selo de acessibilidade.

Adicionalmente, para o sistema Agendamento foram relatadas 116 mensagens de advertência e para o AgendAloca 36 mensagens. Essas mensagens de advertência indicam pontos de controle que precisam ser avaliados manualmente e que podem prejudicar a navegação do usuário com o sistema.

Após a aplicação da ferramenta DaSilva em ambos os sistemas, a próxima avaliação realizada foi sob o ponto de vista de usuário.

### **5.3.3 Realização da avaliação sob o ponto de vista de usuário**

Para realizar a avaliação sob o ponto de vista de usuário, dois estudantes de engenharia da USP-São Carlos se ofereceram como voluntários. Ambos desconheciam os sistemas, representando um dos possíveis grupos de usuários alvo do sistema. Foi definida uma lista de tarefas a serem executadas por ambos participantes em cada um dos sistemas (APÊNDICE E). Conforme previsto no planejamento do experimento, a execução do experimento foi monitorada através do *Morae*, com um Observador.

Seguindo o planejamento, o experimento foi executado na seguinte ordem: foi feita a leitura do termo de compromisso (APÊNDICE C), esclarecendo aos participantes a natureza e os objetivos do trabalho que seria realizado. Após o consentimento de ambos, foi aplicado o questionário pré-sessão (APÊNDICE D), em que foi possível avaliar o perfil dos participantes.

As seguintes informações foram obtidas dos dois participantes:

- Sexo masculino, 18 e 19 anos.
- Não possuem nenhuma deficiência.
- Utilizam a web em média 24h/semana.
- Não sentem medo ou insegurança ao utilizar a web.
- Gostam de aprender coisas novas.
- Nunca utilizaram sistema web que tivesse características de acessibilidade.

- Consideram como dificuldades durante o uso de sistemas web:
  - Reverter a ação anterior.
  - Navegar nos menus.

Em seguida, foi realizado o experimento com os dois sistemas, um de cada vez. A ordem dos testes (prevista anteriormente na Tabela 8) e o tempo total do experimento para cada participante podem ser verificados na Tabela 11.

**Tabela 11: Dados do teste com usuários.**

	<b>Participante 1 (P1)</b>	<b>Participante 2 (P2)</b>
<b>Ordem dos testes</b>	1. AgendAloca 2. Agendamento	1. Agendamento 2. AgendAloca
<b>Tempo total do experimento</b>	40 minutos	50 minutos

A lista de tarefas incluiu apenas tarefas já implementadas nos dois sistemas e a única diferença entre as listas de tarefas é a primeira tarefa, referente às tarefas de acessibilidade, que só estão implementadas no sistema AgendAloca. Porém, esta tarefa não consta na comparação entre os sistemas, pois foi incluída para testar o funcionamento destas funções. Na Tabela 12 estão listadas as tarefas.

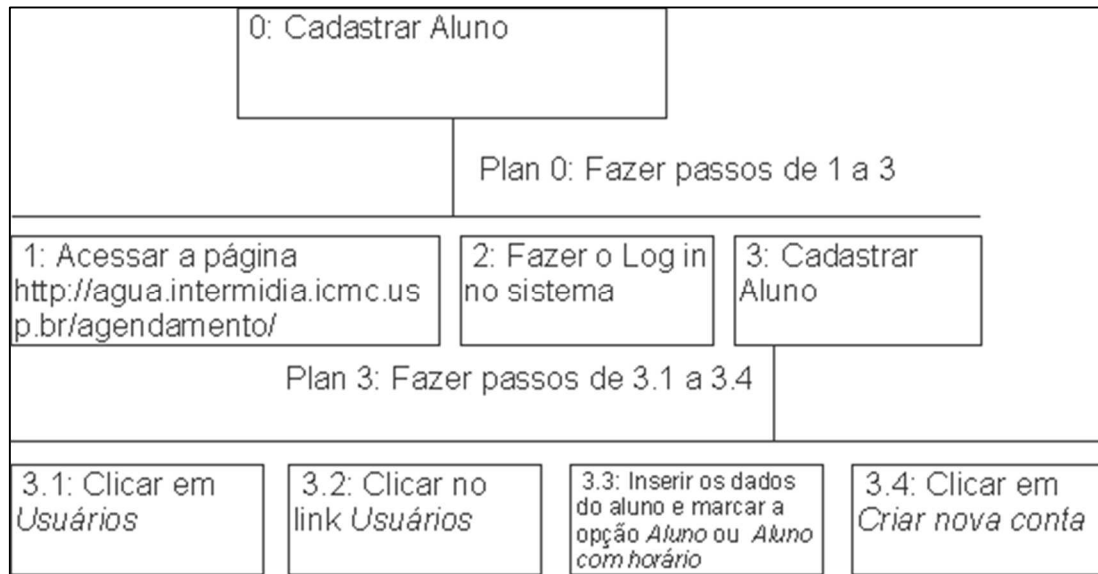
**Tabela 12: Listas de tarefas executadas em cada sistema.**

<b>AgendAloca</b>	<b>Agendamento</b>
0. Ajustar o sistema AgendAloca conforme sua necessidade. Para isso, você poderá utilizar as opções de acessibilidade disponíveis	
1. Fazer <i>login</i> no sistema	1. Fazer <i>login</i> no sistema
2. Visualizar os dados da própria conta	2. Visualizar os dados da própria conta
3. Consultar o calendário de bancas	3. Consultar o calendário de bancas
4. Cadastrar duas contas de aluno	4. Cadastrar duas contas de aluno
5. Fazer <i>logoff</i>	5. Fazer <i>logoff</i>

Verificando os HTAs (*Hierarchical Task Analysis*) das tarefas, percebe-se que a única tarefa que necessita de mais de três ações é a tarefa 4 “Cadastrar duas contas de

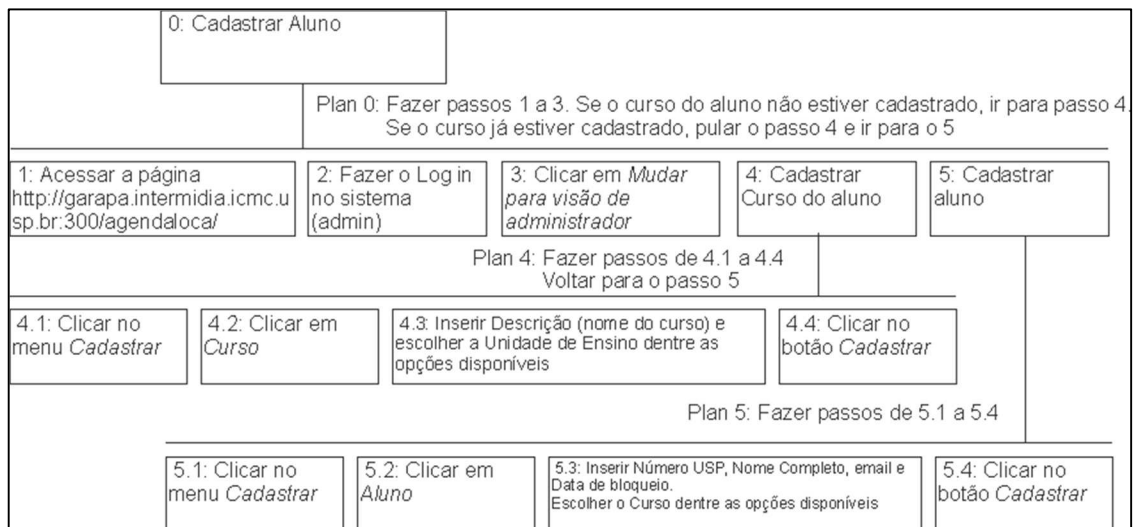


aluno”, sendo as outras quase que imediatas. Os HTAs das tarefas de ambos os sistemas serão apresentados a seguir.



**Figura 62:** HTA da tarefa “cadastrar aluno” no Agendamento.

Na Figura 62 é possível observar o HTA da tarefa “cadastrar aluno” no sistema Agendamento e na Figura 63, observa-se o HTA do sistema AgendAloca.



**Figura 63:** HTA da tarefa “cadastrar aluno” no AgendAloca.

Após analisar os dados coletados no *Morae Manager*, foi possível coletar as variáveis dependentes: tempo de execução das tarefas executadas, quantidade de erros na execução de tarefas, quantidade de tarefas concluídas com sucesso e número de cliques para uma tarefa, expressas na Subseção 5.2.2.

Foi verificada uma grande diferença entre os tempos de execução das tarefas, como demonstrado na Figura 64.

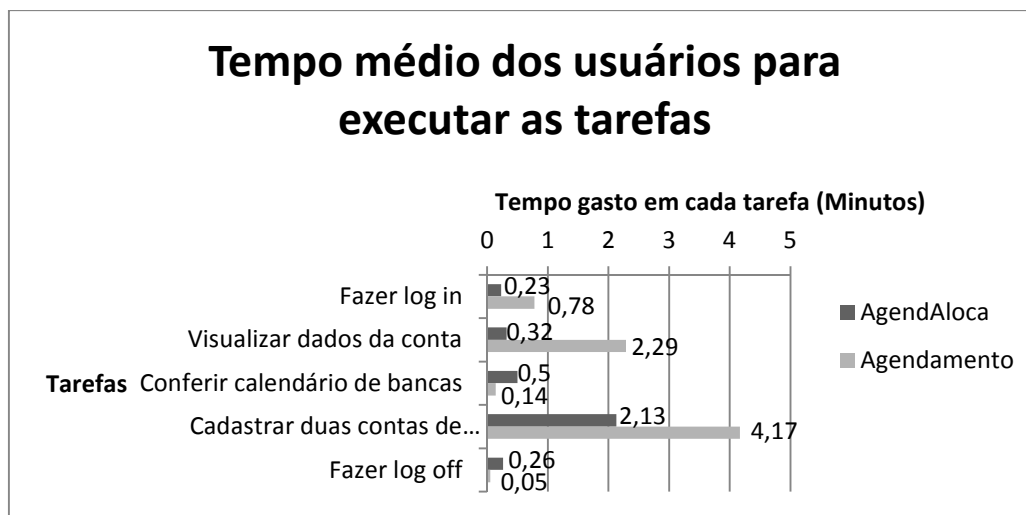


Figura 64: Tempo médio dos usuários para executar as tarefas.

De acordo com a Figura 64, é possível perceber que houve maior dificuldade em tarefas simples, como “Fazer o *login*” e principalmente em “Visualizar dados da conta” (que no HTA leva apenas três ações para ser executado). Isto demonstra que apesar do HTA indicar um caminho rápido, a interface do Agendamento não oferece o mesmo suporte, sendo a execução dessas tarefas confusas para os usuários. Esse resultado apresenta indícios de que a hipótese nula (**HA1** – apresentada na Seção 5.2.2), representada pela equação  $TE_{AG} \leq TE_{SAB}$  é verdadeira.

Com o planejamento dos HTAs, que correspondem ao “caminho perfeito” para a execução de uma tarefa, foi possível medir a quantidade de erros por tarefa realizada pelos participantes, apresentada na Figura 65. Considera-se erro como sendo qualquer interação que fuja do que era esperado pelo HTA. O resultado apresentado na Figura 65 apresenta indícios de que a hipótese nula (**HA2**), representada pela equação  $QE_{AG} \leq QE_{SAB}$ , é verdadeira.

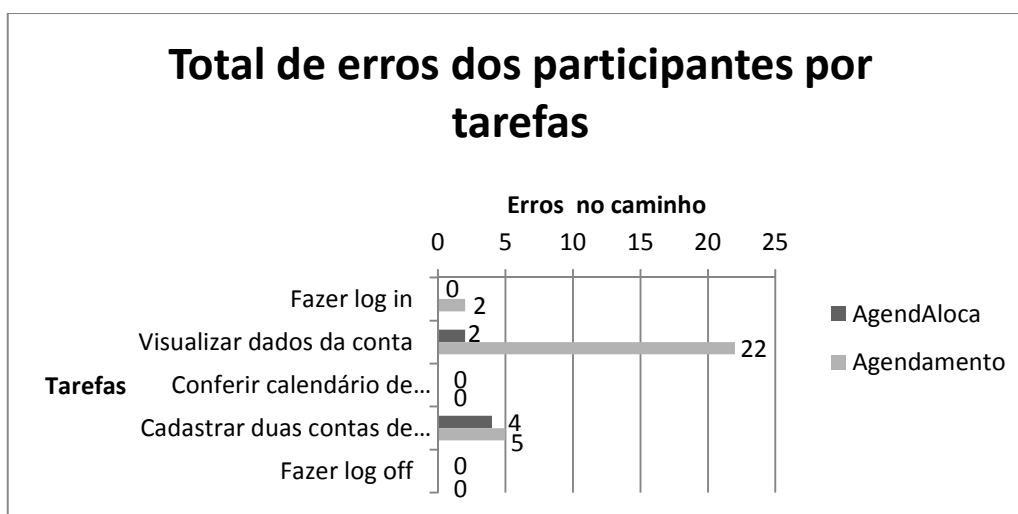
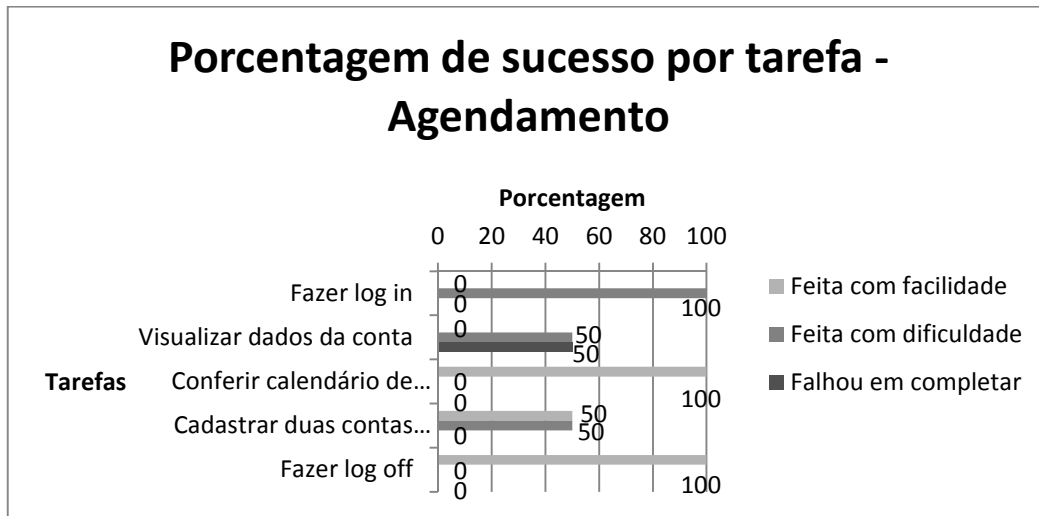


Figura 65: Total de erros dos participantes por tarefa.

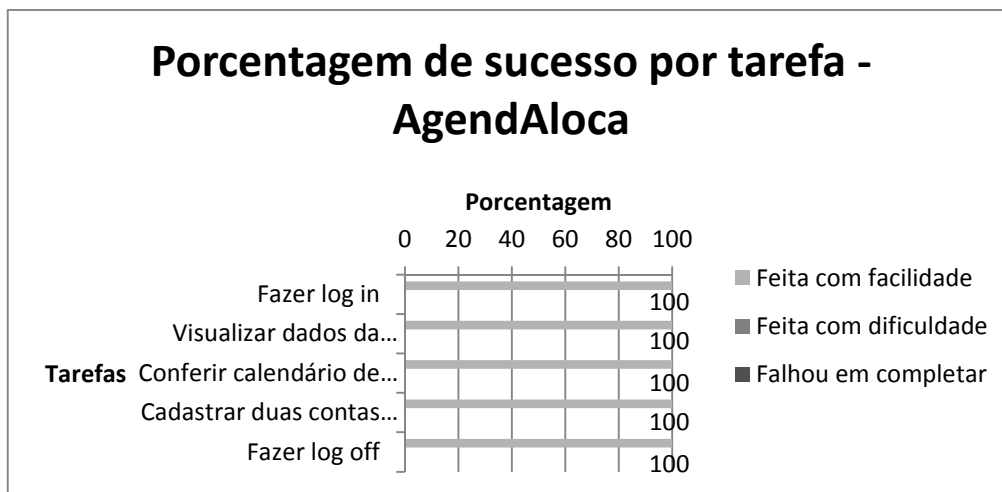
Nas Figuras 66 e 67, é possível verificar a percentagem de tarefas concluídas com facilidade, com dificuldade e as que falharam.



**Figura 66: Porcentagem de sucesso por tarefa – Agendamento.**

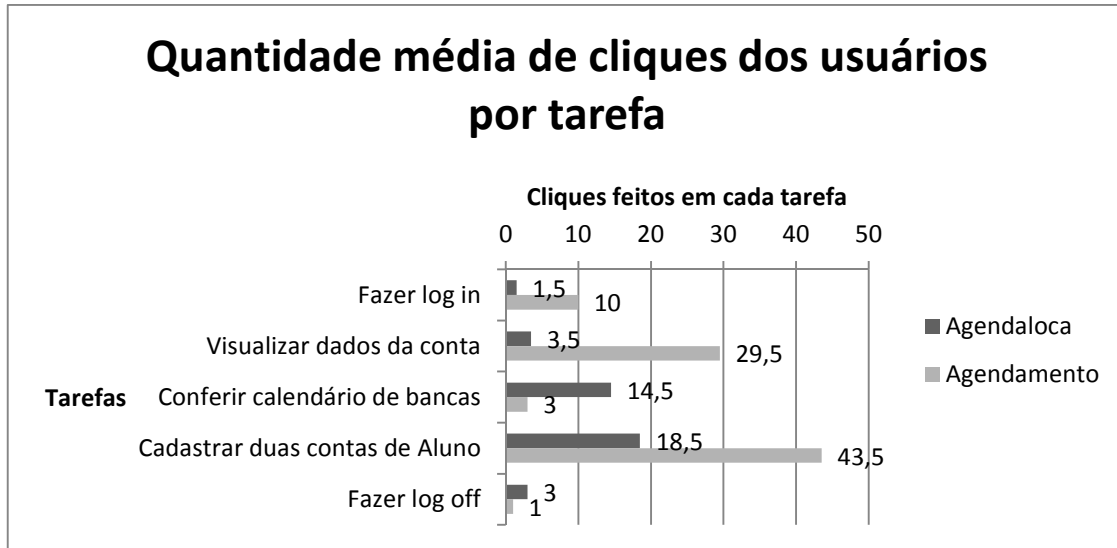
Na Figura 66, é possível perceber que a tarefa 1 “Fazer *login*” foi realizada com dificuldade por ambos os participantes. Já a tarefa 2 “Visualizar dados da conta” foi difícil para um participante, enquanto o outro teve problemas na execução. A tarefa 4 “Cadastrar duas contas de aluno” foi realizada com facilidade por um e com dificuldade por outro e apenas as tarefas 3 “Conferir calendário” e 4 “Fazer *logout*” foram realizadas com facilidade por ambos os participantes.

Contrastando o resultado encontrado na Figura 66 para o sistema Agendamento, os participantes conseguiram realizar todas as tarefas com facilidade no sistema AgendAloca. Os resultados encontrados, apresentados nas Figuras 66 e 67, mostram indícios de que a hipótese nula (**HA3**), representada pela equação  $QT_{AG} \geq QT_{SAB}$ , é verdadeira.



**Figura 67: Porcentagem de sucesso por tarefa – AgendAloca.**

Já a quantidade média de cliques dos usuários por tarefa pode ser encontrada na Figura 68. Novamente, nota-se uma quantidade maior de cliques no Agendamento, comprovando a maior facilidade em executar as tarefas no AgendAloca. Assim como nos casos anteriores, esse resultado apresenta indícios de que a hipótese nula (**HA4**), representada pela equação  $NQ_{AG} \leq NQ_{SAB}$  é verdadeira.



**Figura 68:** Quantidade média de cliques dos usuários por tarefa.

Na Figura 69, é possível ver o momento em que os dois participantes interagem com os sistemas, em que há gravação dos movimentos realizados.



**Figura 69:** Interação dos usuários com os sistemas Agendamento e AgendAloca.

Após a realização do experimento, os participantes foram convidados a responder ao questionário pós-sessão (APÊNDICE F) do sistema testado e tiveram a oportunidade de expressar o que acharam ao utilizar o sistema. Um resumo das principais contribuições dos sujeitos participantes coletadas no questionário pós-sessão pode ser encontrado na Tabela 13.

Tabela 13: Dados coletados no questionário pós-sessão.

	<b>Agendamento</b>	<b>AgendAloca</b>
<b>Dificuldades/ desvantagens</b>	<p><b>P1=</b> O botão de <i>login</i> está escondido em uma aba que, aliás, só existe para confundir quem entra no sistema pela primeira vez. O leiaute do sistema é confuso, sendo mais difícil localizar as informações. As funções do sistema são agrupadas em poucas abas, de forma que são poucas abas para muitas opções/funções.</p> <p><b>P2=</b> A aba de <i>login</i> não estava em um local lógico e alguns campos são confusos.</p>	<p><b>P1=</b> O <i>link</i> para acesso como administrador é um tanto quanto complicado de ver. Não apareceu confirmação de cadastro dos alunos.</p> <p><b>P2=</b> Não considero que o sistema possua dificuldades.</p>
<b>Facilidades/ vantagens</b>	<p><b>P1=</b> O acesso ao cadastramento de bancas parecia bem simplificado.</p> <p><b>P2=</b> As descrições nos <i>links</i> e botões ajudavam bastante na navegação.</p>	<p><b>P1=</b> O processo de cadastro e visualização dos cadastrados é simples e bem explicado.</p> <p><b>P2=</b> O sistema possui facilidades de acesso, com uma tela de <i>login</i> facilmente acessível e todos os campos são bem claros.</p>
<b>Sugestões e/ou comentários</b>	<p><b>P1=</b> Um acesso à região de <i>login</i> mais simples seria bom. Aliás, acho que um simples campo direto na própria página inicial para o <i>login</i> seria melhor.</p> <p><b>P2=</b> O sistema é interessante devido à facilidade em que novos usuários têm ao utilizar o sistema. Porém a aba de <i>login</i> poderia ser mais visível e algumas opções ambíguas poderiam ser consideradas.</p>	<p><b>P1=</b> Inserir a opção de entrar como administrador direto no <i>login</i>. Mostrar confirmação de cadastramento em uma frase simples.</p> <p><b>P2=</b> O sistema é bem simples de ser utilizado, não possui <i>links</i> ambíguos e é facilmente utilizado por leigos e por pessoas com problemas de acessibilidade.</p>
<p><b>P1</b> = Participante 1 <b>P2</b> = Participante 2</p>		

É interessante notar que as análises realizadas na coleta das variáveis dependentes demonstradas nas Figuras 64 a 68, vão de encontro com as percepções dos próprios participantes, coletadas com o questionário pós-sessão, apresentadas na Tabela 13.

A última avaliação realizada como parte do experimento foi a avaliação sob o ponto de vista de especialista, que será explicada na próxima subseção.

### 5.3.4 Realização da avaliação sob o ponto de vista de especialista

Para realizar a avaliação sob o ponto de vista de especialista, o *feedback* indireto coletado pela observação dos usuários durante a interação com os sistemas web foi utilizado. De acordo com Rocha e Baranauskas (2000), o método de observação é a maneira mais valiosa de coletar *feedback* sobre a interação do usuário, pois pode envolver descobertas valiosas sobre os usuários do sistema. A maneira como eles usam as funções do sistema é muito rica e os aspectos do sistema podem ser otimizados, a fim de deixar o uso mais eficaz e eficiente, tornando os usuários mais satisfeitos.

Além da análise da observação indireta da interação dos usuários com os sistemas, foram analisados os vídeos coletados durante a interação dos usuários, a fim de complementar o registro de observação, como sugerido por Nielsen (1993).

Alguns problemas foram detectados durante o experimento. Um deles foi inserir uma tarefa em que o usuário precise verificar se o cadastramento foi realmente realizado com sucesso, pois após alguns cadastramentos eles se perguntavam se de fato haviam realizado os cadastrados com sucesso. Nenhum participante utilizou a barra de acessibilidade com a tarefa *“Ajustar o sistema AgendAloca conforme sua necessidade. Para isso, você poderá utilizar as opções de acessibilidade disponíveis”*, portanto poderia ser melhor a tarefa *“Procurar a barra de acessibilidade e testar suas funcionalidades”*. Dessa forma, mesmo que os participantes não tenham deficiências, poderiam explorar as funcionalidades e contribuir com suas opiniões.

As questões que foram identificadas durante a observação relacionadas à compreensão do sistema Agendamento, já haviam sido resolvidas no desenvolvimento do sistema AgendAloca. Tais como:

- As telas foram refeitas, como consequência ao grande número de informações;
- Os itens dos menus foram refeitos, pois a funcionalidade de alguns itens era incompreensível, porque os seus nomes não eram representativos, e havia uma mistura de palavras em Português e Inglês; e
- Novos menus foram feitos, pois os nomes e a distribuição dos menus não eram intuitivos.

Uma lista de problemas de acessibilidade e usabilidade no Agendamento que foi melhorada com o sistema AgendAloca, pode ser visualizada na Tabela 14.

Tabela 14: Problemas de acessibilidade e usabilidade no Agendamento melhorados no AgendAloca.

Problemas de acessibilidade e usabilidade no Agendamento	Melhorias de acessibilidade e usabilidade realizadas no AgendAloca
Os erros eram apresentados apenas em vermelho, sem explicação do significado do texto.	Todas as informações coloridas tem um equivalente textual. Os erros são apresentados em vermelho, e a palavra "erro" está disponível no início das mensagens de erro, o que pode ser muito útil para usuários que usam leitores de tela.
Ao usar um leitor de tela ou até mesmo a tecla TAB para navegar, os fluxos do calendário eram a partir do início ao fim de uma linha (ou seja, na horizontal). Isso não é instintivo, porque eles navegam por todas as primeiras apresentações de cada dia da semana e, em seguida, salta para a segunda linha, no dia seguinte.	A navegação é iniciada a partir da parte superior da coluna, passando por todas as apresentações desse dia até atingir a extremidade da coluna e, em seguida, passa para o topo da coluna seguinte, no dia seguinte (ou seja, na vertical).
Há apenas um caminho difícil para executar algumas tarefas. Por exemplo, para os administradores adicionarem novos usuários, eles precisam acessar três páginas diferentes, e na última, selecionar uma guia para acessar o formulário de cadastro.	Todas as tarefas básicas do sistema estão disponíveis no menu principal ou usando a barra de pesquisa.
Usa <i>labels</i> para explicar todas as entradas do formulário, no entanto, não usa títulos para explicar a finalidade de cada página/seção.	Usa <i>labels</i> em todas as entradas de formulário e também usa títulos em todas as seções principais/páginas para resumir o conteúdo.
A tradução padrão do CMS não cobre 100% do conteúdo, assim, existem algumas partes do sistema em que o usuário pode encontrar o conteúdo parcialmente em Português e parcialmente em Inglês.	Foram criados botões de internacionalização para mudar todo o conteúdo do sistema para Português ou Inglês.
Alguns botões têm a mesma funcionalidade em páginas diferentes, no entanto, têm diferentes <i>labels</i> . Por exemplo, na página do calendário para filtrar o tipo de projeto a palavra usada é "Aplicar". No entanto, na página de usuários o <i>label</i> do filtro é "Filtrar".	Em todo o sistema, principalmente para as funções criar, ler, atualizar e excluir (CRUD), todos os botões e <i>labels</i> têm a mesma terminologia para não confundir o usuário.
É possível revisar os erros de entrada após a submissão.	É possível revisar as informações inseridas antes da submissão, bem como alterar/atualizar informações se necessário.
Após a submissão do sistema nos avaliadores online, observou-se que a geração automática do código do CMS gera vários elementos com o mesmo ID.	Devido à pequena quantidade de código gerado, a maior parte escrita manualmente, torna-se fácil evitar IDs duplicados e realizar outras especificações deste critério.

Os formulários de <i>login</i> e de pesquisa aparecem em um painel que desliza a partir da parte superior da página quando o usuário clica com o mouse. Essa abordagem diferente não é identificável pelos leitores de tela, então quando o usuário navega usando a tecla TAB, esta parte do sistema é inacessível.	O padrão HTML que funciona bem com leitores de tela foi usado no novo sistema.
---	--

Além de melhorias realizadas com o desenvolvimento do sistema AgendAloca, algumas características que não existiam no Agendamento e foram inseridas no AgendAloca podem ser citadas:

- O texto pode ser redimensionado sem tecnologia assistiva até 200 por cento, sem perda de conteúdo ou funcionalidade (exceto para legendas e imagens de texto);
- Foi adicionado um botão para ativar o alto contraste da página;
- Foi adicionado um *link* "Pular para o conteúdo", que ajuda os usuários de leitores de tela a pular diretamente para a área do conteúdo principal da página;
- O nome do sistema e a finalidade da página são apresentados em cada página;
- Todos os *links* têm um propósito claro de acordo com o texto descritivo. Por exemplo, quando os usuários querem acessar suas informações podem clicar no *link* "Minha Conta" e para registrar uma apresentação, eles precisam clicar no *link* "Registrar minha apresentação"; e
- Todos os elementos mostram uma "borda" quando recebem foco, exceto entradas de formulário que utiliza o cursor como um indicador de foco.

É importante ressaltar que, de acordo com Bonifácio *et al.* (2011) e Conte *et al.* (2007), amostras reduzidas são problemas conhecidos na área de IHC e ES. Portanto, devido a este fator, há limitação nos resultados encontrados com a avaliação experimental do PDWAU, sendo estes resultados considerados indícios e não conclusivos. No entanto, destaca-se que o experimento realizado com os sistemas Agendamento e AgendAloca, sob o ponto de vista de inspeção, de ferramenta, de usuário e de especialista, trouxe resultados significativos.



Com a avaliação sob o ponto de vista de inspeção, apresentada na Subseção 5.3.1, foi realizada uma avaliação de conformidade em que foi possível observar que o sistema Agendamento, ao contrário do sistema AgendAloca, não atende grande parte dos critérios de sucesso do WCAG 2.0.

Na avaliação sob o ponto de vista de ferramenta, apresentada na Subseção 5.3.2, aplicou-se a ferramenta de avaliação online DaSilva, em que foi possível encontrar 16 erros de acessibilidade no sistema Agendamento, enquanto apenas mensagens de advertência (que indicam a necessidade de uma avaliação manual) foram apresentadas para o sistema AgendAloca.

Durante a avaliação sob o ponto de vista de usuário, Subseção 5.3.3, percebeu-se a ocorrência de vários problemas durante a realização das tarefas em ambos os sistemas, principalmente problemas de acessibilidade e usabilidade com o sistema Agendamento. Essa avaliação pôde apresentar indícios de que as hipóteses nulas HA1, HA2, HA3 e HA4 (apresentadas na Subseção 5.2.2), são verdadeiras.

Por fim, na avaliação sob o ponto de vista de especialista, apresentada na Subseção 5.3.4, foi possível realizar uma síntese com as principais diferenças entre os sistemas Agendamento e AgendAloca, bem como uma análise qualitativa dos problemas encontrados, em termos de requisitos de acessibilidade e usabilidade.

Portanto, a partir dos resultados encontrados, foi possível perceber indícios e confirmar a suposição sobre a efetividade do processo PDWAU para o desenvolvimento de sistemas web com foco em acessibilidade e usabilidade. Ressalta-se, ainda, que a chave para o sucesso de qualquer interface de sistemas web é basear-se no maior número de requisitos de acessibilidade e usabilidade quanto possível.

Após a realização do experimento para avaliar o processo PDWAU, um método de avaliação foi concebido como contribuição complementar. Esse método é apresentado na próxima seção.

## **5.4 Um método de avaliação**

Considerando os conhecimentos adquiridos com o desenvolvimento do experimento, a partir dos pontos de vista de inspeção, de ferramenta, de usuário e de especialista (Seção 5.3), percebeu-se que avaliar um sistema web seguindo esses quatro pontos de vista produz um bom resultado, proveniente de sua complementaridade e dos cruzamentos de resultados que se consegue obter.

Dessa maneira, formalizou-se o ACCESSA (*Approach to improve the aCCESSibilidade and usAbility of existing web system*), um método resultante do experimento, apresentado na Figura 70.

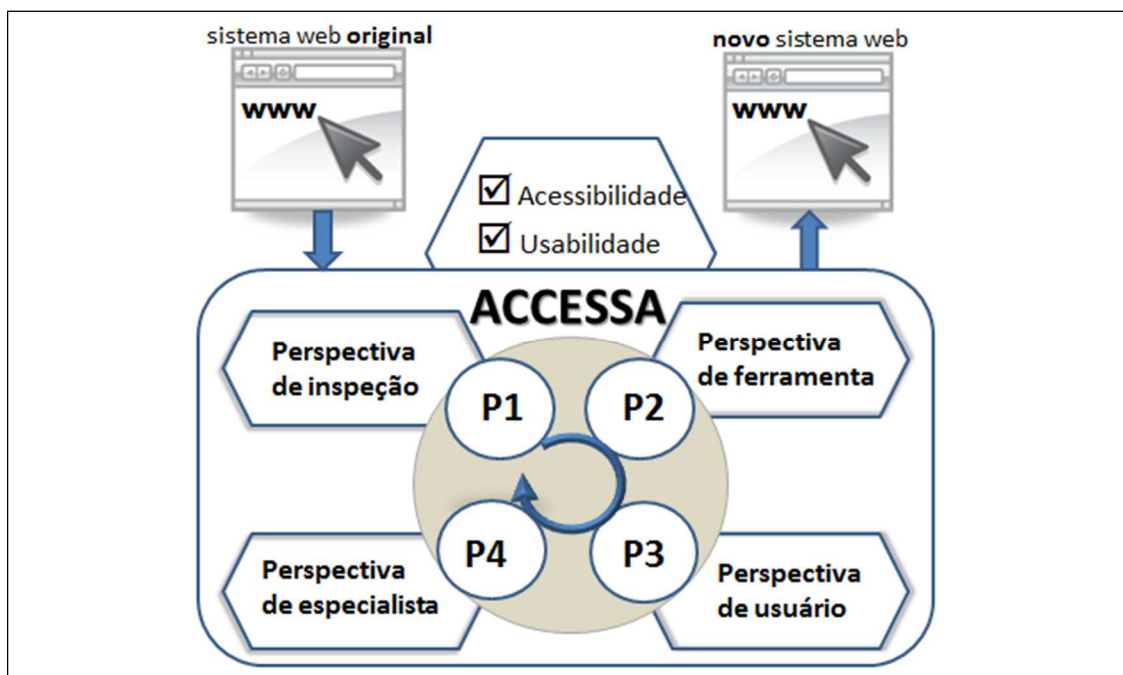


Figura 70: Visão geral do método ACCESSA.

O ACCESSA é apresentado como método de avaliação sob quatro perspectivas (P1, P2, P3 e P4), baseado na avaliação experimental do PDWAU. No entanto, conforme apresentado na Figura 70, o método ACCESSA pode ser utilizado não só para avaliar sistemas web existentes, mas também para comparar e melhorar a acessibilidade e a usabilidade desses sistemas, agindo principalmente no projeto de interface, sem mudar os requisitos funcionais do sistema.

Dessa maneira, com um sistema web já existente (sistema web **original**), após a aplicação do ACCESSA, pode-se produzir uma nova versão do sistema web (**novo sistema web**), com requisitos de acessibilidade e usabilidade. O ACCESSA contribui nesse caso apontando problemas e requisitos com foco em acessibilidade e usabilidade que precisam ser melhorados para uma próxima versão do sistema.

Uma breve apresentação das quatro perspectivas básicas do método ACCESSA (perspectiva de inspeção, perspectiva de ferramenta, perspectiva de usuário e perspectiva de especialista), é apresentada nas próximas subseções.

### 5.4.1 Perspectiva de inspeção

De acordo com a norma ISO/IEC 9126 (2001) com foco na qualidade do produto de software, a conformidade é utilizada para avaliar o quanto o software obedece aos requisitos de legislação e todo o tipo de padronização ou normalização aplicável ao contexto.

A perspectiva de inspeção é uma avaliação de conformidade (Brajnik, 2008a) realizada por um especialista para determinar se um sistema web atende aos critérios de sucesso de acessibilidade, de acordo com as diretrizes do WCAG 2.0, ou seja, ela verifica a correspondência entre a especificação e o software. Como discutido por Kelly *et al.* (2007), como o modelo de acessibilidade do W3C visa acessibilidade universal, pode-se assumir que o sistema web que está em conformidade com as diretrizes do WCAG tem a pré-condição fundamental para isso.

Um dos maiores benefícios de se utilizar inspeções em sistemas web é a detecção de defeitos nas fases iniciais do processo de desenvolvimento de software, facilitando a correção destes defeitos com menor esforço e custo. Esta redução no retrabalho pode implicar em melhorias significativas para a produtividade de software. Apesar das vantagens, inspeções podem não ser adequadas para descobrir defeitos nas interações, demonstrar se o software é útil e verificar requisitos não funcionais, como desempenho, segurança, etc.

Na primeira perspectiva do ACCESSA, perspectiva de inspeção, é necessário verificar a conformidade do sistema com os princípios do WCAG 2.0, que são: perceptível, operável, compreensível e robusto. Essa perspectiva pode ser complementada com uma avaliação de conformidade com outros critérios de sucesso, bem como heurísticas e padrões de usabilidade.

### 5.4.2 Perspectiva de ferramentas

Perspectiva de ferramentas é dada por avaliar a acessibilidade de sistemas web com ferramentas de avaliação automática. Isso se faz necessário, pois as diretrizes do WCAG fornecem critérios objetivos que são testáveis e que podem ser avaliados com uma combinação de testes automáticos. Várias ferramentas podem ser utilizadas, tais como as indicadas na Subseção 3.3.1.

A avaliação de sistemas web por meio de uma ferramenta automatizada agiliza a avaliação de acessibilidade porque permite a imediata identificação de problemas no código-fonte, auxiliando a correção dos mesmos. No entanto, não se pode concluir que

uma página seja mais acessível que outra apenas porque obteve resultado com menos erros de acessibilidade em testes realizados com a mesma ferramenta automática. Apenas pode-se concluir que uma página apresentou menos violações às diretrizes de acessibilidade que a outra. O número de violações não indica o quanto uma página é acessível porque a acessibilidade envolve não apenas a conformidade às diretrizes, mas também a interação de um usuário com essa página.

Além da avaliação de acessibilidade realizada por meio de ferramentas automáticas, deve-se incluir também verificações com usuários (a partir de tarefas), focando em dificuldades encontradas na realização de seus trabalhos diários. Estas tarefas podem orientar modelos mentais do usuário para superar as dificuldades e contribuir para tornar satisfatória a interação de usuários com deficiência com um aplicativo baseado na web.

Por isso, alguns autores têm realizado testes com a participação de usuários reais para verificar a acessibilidade na web (Power *et al.*, 2012; Santos e Fortes, 2010), uma vez que esses testes simulam condições de utilização do ponto de vista do usuário final, priorizando, por exemplo, a análise da facilidade de navegação entre telas da interface do usuário, clareza de seus textos e mensagens. Desta forma, as dificuldades e os problemas enfrentados pelos usuários durante as sessões de uso podem ser identificados, permitindo que os avaliadores formulem uma avaliação diagnóstica mais realista, com medidas corretivas para designers. Dessa maneira, a próxima perspectiva, com usuários, torna-se fundamental.

É importante ressaltar que a perspectiva de inspeção e de ferramenta não foram agrupadas em uma única perspectiva, pois são tarefas que precisam ser realizadas separadamente. No entanto, uma perspectiva não exclui a outra, lembrando que de acordo com o método ACCESSA, é necessário realizar uma triangulação das perspectivas para se alcançar uma avaliação com resultados mais palpáveis.

### **5.4.3 Perspectiva de usuário**

Testes com usuários pode contribuir para identificar as necessidades de usuários ou verificar o entendimento dos projetistas sobre estas necessidades, identificar problemas de interação ou de interface, investigar como uma interface afeta a forma de trabalhar dos usuários, etc. Dependendo do momento em que for realizada a avaliação, estes benefícios podem ter efeito imediato, em consertos importantes no início do desenvolvimento; a médio prazo, no planejamento da estratégia de treinamento e

marketing; ou até mesmo a longo prazo, apoiando o planejamento de versões futuras do software. No entanto, assim como outros testes não automatizados, os testes com usuários podem ser caros de executar.

A perspectiva de usuário é expressa pelo ponto de vista do usuário durante a sua interação com o sistema. Ele é representado pela recuperação de dados de satisfação por meio de três métodos:

- Questionários pré-sessão (para recolher o perfil do usuário) e questionários pós-sessão (para coletar a opinião dos usuários sobre o sistema web em atividade de avaliação);
- Aplicação do método *think aloud* (Brajnik, 2008a; Romen e Svanaes, 2008), que orientam os usuários a dizer em voz alta o que eles estão pensando enquanto navegam no sistema web; e
- Entrevista, a fim de coletar a opinião dos usuários sobre o sistema.

Essa perspectiva é uma das mais importantes, pois o usuário tem contato com o sistema e pode emitir sua opinião, ajudando os desenvolvedores a perceber possíveis problemas que até então eram imperceptíveis.

#### **5.4.4 Perspectiva de especialista**

A última perspectiva do ACCESSA, perspectiva de especialista, provê uma análise da interação que o usuário desempenhou no acesso ao sistema web. Para que isso seja viável, é necessário que a interação do usuário seja gravada e documentada.

Após a avaliação do sistema de acordo com as quatro perspectivas do método ACCESSA, o especialista vai então reunir todos os resultados encontrados em cada uma delas. Como produto final do método, será possível ter um conjunto de requisitos de acessibilidade e usabilidade que podem ser úteis para: I) comparar dois sistemas web e II) indicar requisitos que precisam ser melhorados para uma próxima versão do sistema web.

Ressalta-se ainda que, conseqüentemente, o processo proposto neste trabalho pode ser usado independentemente do método ACCESSA, podendo ser útil para os processos de reengenharia de sistemas web.

A utilização detalhada do método ACCESSA para avaliar sistemas web, conforme o experimento apresentado nas Seções 5.2 e 5.3, foi publicado e pode ser encontrado com maiores detalhes em Dias *et al.* (2013).

## 5.5 *Considerações finais*

O experimento realizado apresentou indícios da efetividade na utilização do PDWAU para o desenvolvimento de sistemas web com foco em acessibilidade e usabilidade. Além disso, foi possível formalizar o método ACCESSA que pode ser útil para gerar um conjunto de requisitos para melhorar a interface de sistemas web existentes (ou legados), bem como comparar sistemas web.

Como foi percebido até aqui, a web representa a principal forma de prestação de serviços de computação para uma enorme variedade de usuários com características diferentes. Assim, características de acessibilidade e usabilidade tornam-se ainda mais críticas para o sucesso de um sistema web e os pesquisadores têm, cada vez mais, preocupações em fornecer mecanismos que sirvam de diretrizes para que os sistemas web sejam melhorados. No caso desta pesquisa, a solução indicada é a utilização de um processo para o desenvolvimento de sistemas web com foco em acessibilidade e usabilidade.

No entanto, após os indícios apresentados com a avaliação experimental do PDWAU, percebeu-se a ausência de um instrumento que fornecesse uma medida direta para comparar sistemas web, bem como de quanto esforço um sistema web precisa para ser melhorado, em termos de acessibilidade e usabilidade. Dessa maneira, fez-se necessário a formalização de uma métrica, que será apresentada no Capítulo 6.

---

# *Estudo de viabilidade do PDWAU usando uma métrica de acessibilidade e usabilidade*

---

## *6.1 Considerações iniciais*

Conforme mencionado em seções anteriores, a avaliação da acessibilidade e usabilidade em sistemas web deve incluir, além de ferramentas de verificação automáticas, avaliação com usuários e de conformidade por especialistas. Testes de conformidade podem ajudar os especialistas em avaliação de sistemas web a simularem condições de utilização do ponto de vista do usuário final, priorizando, por exemplo, a análise da facilidade de navegação entre as interfaces do usuário, bem como a clareza de seus textos e mensagens.

As dificuldades e os problemas enfrentados pelos usuários durante as interações podem ser identificados, permitindo que os avaliadores formulem uma avaliação diagnóstica mais realista, propondo medidas corretivas para os projetistas. Para apoiar essa atividade, neste capítulo é apresentada uma métrica bem definida para medir a acessibilidade e usabilidade de sistemas web.

Este capítulo está organizado da seguinte maneira: medições existentes por meio de questionários são apresentadas na Seção 6.2. A formalização de uma métrica de acessibilidade e usabilidade é apresentada na Seção 6.3. Na Seção 6.4, um estudo de viabilidade realizado como uma segunda avaliação indireta do processo PDWAU, é apresentado. Por fim, na Seção 6.5, encontram-se as considerações finais.

## **6.2 *Medições existentes por meio de questionários***

Os resultados da aplicação de um questionário podem ser conseguidos por meio de uma análise estatística das respostas coletadas, unidas com o propósito de compor uma resposta capaz de espelhar a opinião de um grupo de usuários, ou por meio de análises mais simples, as quais envolvam o cálculo de porcentagens baseadas nos resultados obtidos. Existem algumas medições existentes por meio de questionários que são utilizados para avaliar a satisfação dos usuários, tais como: ISONORM (Prümper, 1993) com base nos princípios da norma ISO 9241-10; o QUIS (Shneiderman, 1988); o SUMI (2012; Kirakowski, 2002); o WAMMI (2005); e o ErgoList (1998).

O ISONORM é um questionário de satisfação que tem o objetivo de direcionar a aplicação da norma ISO 9241-10 aos quesitos apontados como problemáticos pelo usuário, avaliando a conformidade dos produtos de software com as recomendações da norma. Foi criado por uma equipe conduzida por Joachim Prümper, em 1999, e contém 35 perguntas. As respostas são separadas em dois polos, o esquerdo descreve o extremo negativo e o direito descreve o extremo positivo, sendo que os valores possíveis para as respostas variam entre ‘---’ e ‘+++’, codificados em um intervalo de 1 a 7 (Medeiros, 1999). Possui bom grau de aplicabilidade, é limitado a cinco questões para cada princípio da norma, de modo a reduzir o tempo de preenchimento. É dividido em sete seções, conforme os princípios de diálogo da norma.

O QUIS (*Questionnaire for User Interface Satisfaction*) é uma ferramenta desenvolvida por uma equipe multidisciplinar de pesquisadores do Laboratório de Interação Humano-Computador (HCIL) da Universidade de *Maryland* em *College Park*. O QUIS foi projetado para avaliar a satisfação subjetiva dos usuários em relação a aspectos específicos da interface humano-computador, sendo que deve ser adaptado de acordo com as necessidades de cada análise, incluindo somente as seções que são de interesse do usuário (Shneiderman, 1988). Trata-se de um questionário demográfico, apresentada em uma versão curta e uma versão longa, sendo que ambas são divididas em



9 seções (tela, da terminologia e do sistema, aprendendo fatores, capacidades de sistema, manuais técnicos, cursos em linha, multimídias, sistema de teleconferências, e instalação de software) e cada questão mede a satisfação do usuário, em uma escala de 9 graus.

O SUMI (*Software Usability Measurement Inventory*) foi desenvolvido em 1990 por Kirakowski (2002) e é um questionário de medição da qualidade de uso de um produto ou protótipo de software, contendo 50 questionamentos, cada um oferecendo 3 opções de resposta (concordo, indeciso, discordo). Possui cinco aspectos principais: eficiência, empatia, controle, ajuda e aprendizagem. É muito recomendado e com resultados expressivos de desempenho e precisão. É, entretanto, um produto de custo razoavelmente alto (SUMI, 2012).

WAMMI (*Website Analysis and MeasureMent Inventory*) foi desenvolvido por Jurek Kirakowski e Nigel Claridge em 1996. É composto por 20 questões que devem ser respondidas usando uma escala de 5 pontos, indo da opção concordo fortemente à opção discordo fortemente. O WAMMI busca descobrir se a navegação foi facilitada e se a comunicação foi constante (controle); se houve interação agradável (atratividade); se a busca de informações foi satisfatória (eficiência); quais as possibilidades e aspectos de solução de problemas de navegação (suporte ao usuário); e se o uso foi satisfatório na primeira visita com uso exploratório nas visitas subsequentes (facilidade de aprendizado) (WAMMI, 2013).

O ErgoList é um questionário disponibilizado sob a forma de serviço na Internet, composto de uma base de conhecimento em ergonomia, que inspeciona, via *checklist*, interfaces humano computador. Tem uma grande base de questões, cobrindo aspectos amplos de usabilidade e satisfação de uso. Por ser um *checklist*, deve ser utilizado por um especialista com conhecimento de ergonomia e interfaces humano computador (ErgoList, 1998). O questionário leva em consideração dezoito critérios. Para cada um desses critérios, tem-se uma avaliação específica da ergonomia de uma interface humano computador, totalizando 194 questões. As 194 questões do *checklist* vêm acompanhadas de notas explicativas, exemplos e de um glossário a fim de esclarecer possíveis dúvidas associadas às mesmas. O ErgoList é baseado nos critérios ergonômicos propostos por Scapin e Bastien (1993).

Apesar das medições existentes por meio de questionários para avaliar interfaces, nenhuma delas trata de requisitos de usabilidade, acessibilidade e padrões web. Sendo assim, fez-se necessário a formalização de HEUA, apresentado na próxima Subseção.

### 6.3 Formalização da métrica de acessibilidade e usabilidade

HEUA (*Heuristic Evaluation with Usability and Accessibility requirements to assess web systems*) foi concebido como um instrumento de medição que pode ser utilizado para avaliar a acessibilidade e usabilidade de sistemas web existentes. Mais especificamente, é um instrumento direto a ser aplicado, uma vez que contribui para as melhorias feitas nas versões de sistemas web, produzindo um diagnóstico das características tratadas em termos de questões de acessibilidade e usabilidade (Dias, Fortes e Masiero, 2014).

Como os sistemas web têm um processo de desenvolvimento de beta perpétuo, HEUA pode ser visto como um registro que pode acompanhar as versões que são rapidamente construídas. Adicionalmente, HEUA pode ser classificado como um método de inspeção porque é conduzido por especialistas.

HEUA é composto por questões que contém requisitos que foram definidos com base nas referências conhecidas de usabilidade e acessibilidade. Os 23 padrões de Montero *et al.* (2002), apresentados na Subseção 2.3.2, foram usados para medir a usabilidade. Todos os critérios de sucesso do WCAG 2.0 (W3C, 2008b), bem como os critérios para usuários mais velhos formalizados por Lara (2012), apresentados na Subseção 2.2.1, totalizando 107 diretrizes, foram usadas para medir a acessibilidade.

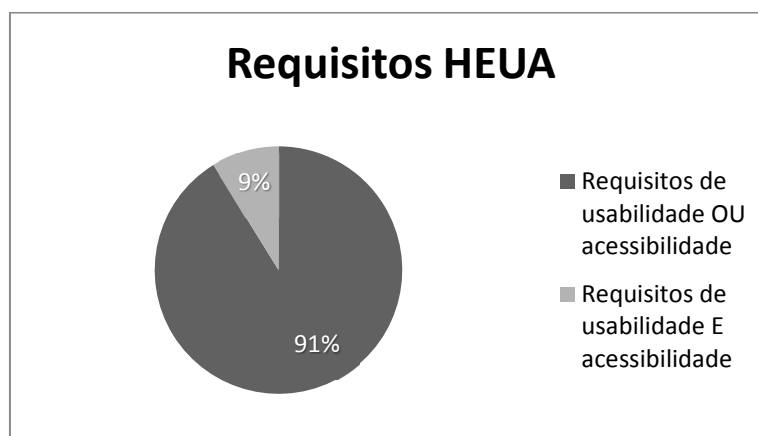
Neste momento, padrões e critérios de sucesso são chamados de requisitos. Todos esses requisitos foram classificados nas dez heurísticas de Nielsen (1993), apresentadas na Subseção 2.3.1, pois de acordo com uma pesquisa realizada por Santos, Gassenferth e Machado (2008), quando os pesquisadores usam o termo usabilidade, a maioria refere-se a Jakob Nielsen. O resumo desta classificação pode ser encontrado na Tabela 15.

**Tabela 15: Resumo de HEUA.**

<b>Questões de HEUA (baseadas nas heurísticas de Nielsen)</b>	<b>Qtde de requisitos</b>
<b>Q1:</b> O sistema mantém o usuário sempre informado sobre o que está ocorrendo por meio de <i>feedbacks</i> em tempo real?	10
<b>Q2:</b> O sistema utiliza a linguagem e o modelo mental do usuário com características na interface que são correspondentes ao mundo real?	8
<b>Q3:</b> O sistema oferece controle e liberdade ao usuário para sair de estados indesejáveis?	13
<b>Q4:</b> O sistema é consistente e segue o mesmo padrão em toda a interface a fim de facilitar o reconhecimento do usuário?	11

<b>Q5:</b> O sistema apresenta um projeto (design) preventivo e cuidadoso que pode ser capaz de evitar algum problema durante a interação do usuário?	19
<b>Q6:</b> O sistema evita a sobrecarga de memória do usuário fornecendo informações contextuais para cada ação?	6
<b>Q7:</b> O sistema oferece flexibilidade e eficiência aos usuários, agilizando o uso para usuários experientes e mantendo a facilidade para os novatos?	14
<b>Q8:</b> O sistema oferece estética e projeto (design) minimalista, mantendo apenas informações úteis, diretas e claras?	5
<b>Q9:</b> O sistema ajuda o usuário a reconhecer, diagnosticar e corrigir erros?	3
<b>Q10:</b> O sistema fornece uma ajuda documentada que pode ser facilmente encontrada em caso de necessidade?	4
<b>Total de requisitos</b>	<b>93</b>

Conforme apresentado na Tabela 15, os 130 requisitos (23 requisitos de usabilidade + 107 requisitos de acessibilidade) foram sintetizados em 93 requisitos. Isso significa que 39% dos requisitos de usabilidade (9 de 23) e 34% dos requisitos de acessibilidade (37 de 107) são comuns entre si. Essa porcentagem de requisitos de usabilidade e acessibilidade que foram agrupadas pode ser visualizada na Figura 71.



**Figura 71: Porcentagem de requisitos de usabilidade e acessibilidade que se concatenam.**

HEUA foi inspirado no método GQM (Basili e Weiss, 1984), pois segue a mesma lógica de desenvolvimento, com objetivo, questões e métricas. No entanto, devido à dificuldade em desenvolver métricas de usabilidade e acessibilidade, conforme apresentado na Figura 18 do Capítulo 3, ao invés de métricas como terceiro nível, tem-se requisitos que precisam ser cumpridos para que a métrica  $M_{HEUA}$  seja calculada. O planejamento do instrumento de medição HEUA, inspirado no método GQM, é apresentado esquematicamente na Figura 72.

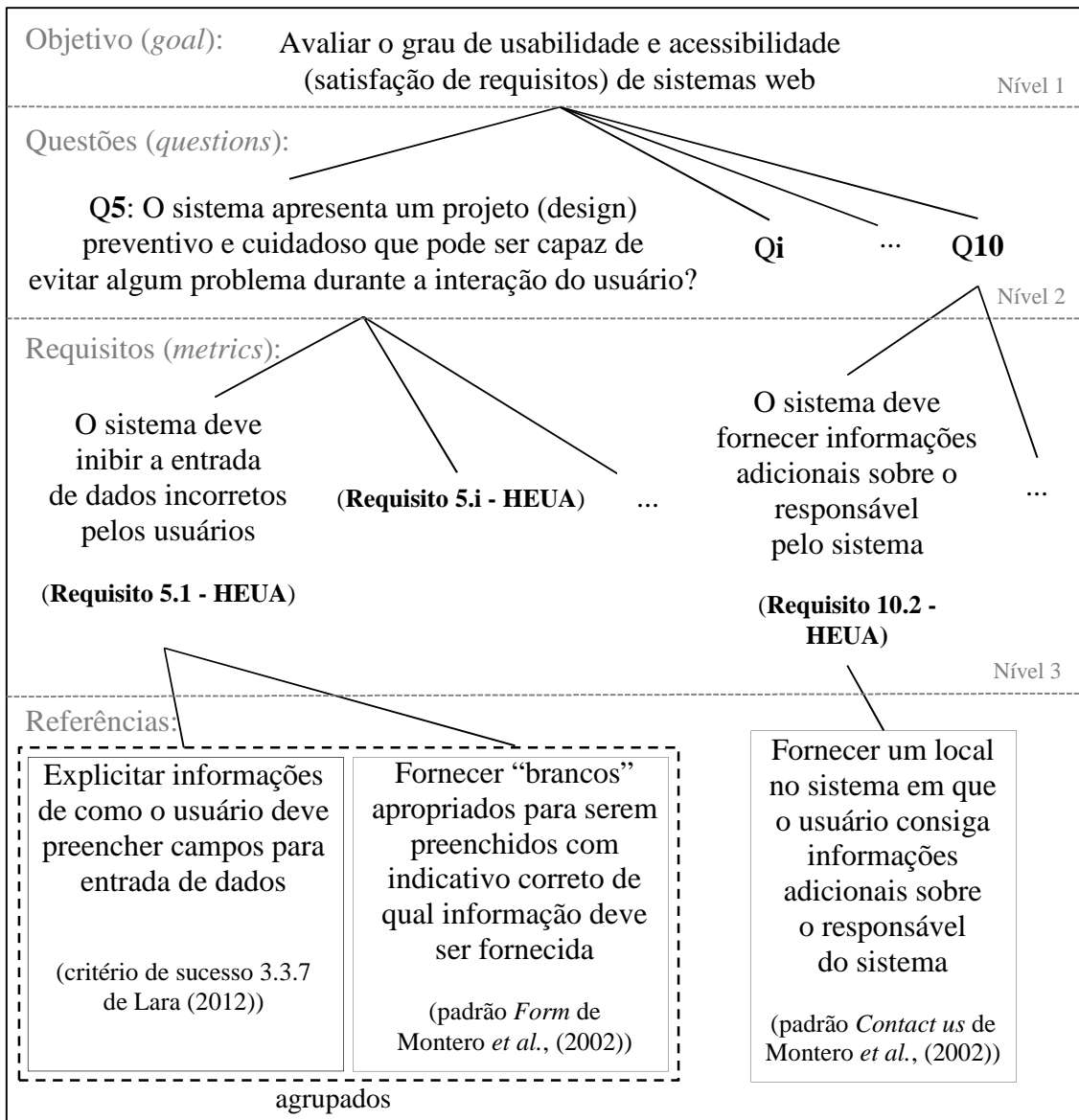


Figura 72: Planejamento do desenvolvimento de HEUA.

A classificação dos requisitos de HEUA foi realizada em conjunto com uma equipe de oito especialistas em usabilidade e acessibilidade, que validaram a classificação dos 130 requisitos (previamente sintetizados em 93 requisitos) nas dez questões. A classificação completa dos requisitos no questionário HEUA pode ser encontrada no APÊNDICE G.

Ressalta-se que para cada requisito, há uma descrição que contém um exemplo que pode apoiar o desenvolvedor na identificação do requisito no sistema, uma motivação que justifica o porquê do requisito estar presente no HEUA e as fontes que indicam a origem de cada requisito, conforme apresentado na Tabela 16.

Tabela 16: Exemplos de descrição de requisitos de HEUA.

Questões e requisitos	Atende	Não atende	Não se aplica
<b>Q1:</b> O sistema mantém o usuário sempre informado sobre o que está ocorrendo por meio de <i>feedbacks</i> em tempo real?			
<b>Requisito 1.1:</b> O sistema deve fornecer o caminho percorrido (clicável) pelo usuário. <u>Exemplo:</u> <i>breadcrumb</i> . <u>Motivação:</u> é importante que os usuários possam rever suas experiências de sucesso como forma de incentivá-los a realizar novas interações, reduzindo a insegurança e a quantidade de erros durante a navegação ou execução de uma tarefa. <u>Fontes:</u> <i>Location</i> – Montero / WCAG 2.4.8 / Lara 2.4.19			
<b>Q7:</b> O sistema oferece flexibilidade e eficiência aos usuários, agilizando o uso para usuários experientes e mantendo a facilidade para os novatos?			
<b>Requisito 7.10:</b> O sistema deve oferecer uma ferramenta de busca para que os usuários localizem as opções disponíveis. <u>Exemplo:</u> campo de busca. <u>Motivação:</u> busca é um dos principais elementos que um sistema deve ter para que os usuários localizem as opções disponíveis de forma fácil e sem muito esforço. Além disso, facilita e agiliza a interação de usuários mais experientes. <u>Fontes:</u> <i>Search</i> – Montero / WCAG 2.4.5			
<b>Q9:</b> O sistema ajuda o usuário a reconhecer, diagnosticar e corrigir erros?			
<b>Requisito 9.2:</b> O sistema deve fornecer mensagens de erro ao usuário que devem ser as mais específicas possíveis. <u>Exemplo:</u> No caso de envio de formulário sem sucesso, reexibir o formulário e indicar os campos de erro. <u>Motivação:</u> mensagens de erro específicas ajudam o usuário a identificar e corrigir o possível erro. <u>Fontes:</u> WCAG 3.3.1			

Para usar HEUA<sup>20</sup>, o especialista precisa verificar todos os requisitos contidos nas dez questões (Q) de HEUA (APÊNDICE G), decidindo se o sistema atende, não atende ou se não se aplica. Depois de aplicar HEUA, um conjunto de medidas deve ser calculado para obter um resultado quantitativo. Esse resultado vai indicar o quanto de requisitos de usabilidade e acessibilidade é contemplado por um sistema web.

<sup>20</sup> Disponível em: <http://migre.me/g7XmV>

A seguinte expressão é uma medida que representa quanto o sistema web satisfaz cada questão ( $Q_t$ ) avaliada pelo especialista:

$$Q_t = \frac{S - NS}{REQ - NA} * 100$$

Onde:

- $Q_t$  são as questões (t variando de 1 a 10); e
- S é o total de requisitos que o sistema "atende" (*satisfied*);
- NS é o total de requisitos que o sistema "não atende" (*not satisfied*);
- NA é o total de requisitos que "não se aplica" ao sistema (*not applicable*);
- REQ é a quantidade de requisitos existentes para cada questão a ser respondida (*requirements*);
- $M_{HEUA}$  é o resultado final.

No final, a soma de todos os resultados de  $Q_t$  fornece uma medida de quanto os requisitos de acessibilidade e usabilidade foram atendidos pelo sistema web, de acordo com HEUA ( $M_{HEUA}$ ). A métrica (soma dos resultados) pode ser calculada como se segue:

$$M_{HEUA} = \frac{\sum_{i=1}^{10} Q_t}{10}$$

O resultado obtido em  $M_{HEUA}$  pode variar de -100 a +100, e, gradualmente, apresenta valores que correspondem a diferentes níveis de adequação em termos de acessibilidade e usabilidade. Assim, quando os valores positivos aumentam, significa que um sistema tem um melhor nível de acessibilidade e usabilidade. Por outro lado, quando os valores negativos diminuem, isso pode indicar que os requisitos de acessibilidade e usabilidade não foram bem empregados no sistema web sob avaliação. Portanto, um sistema web com um valor maior de  $M_{HEUA}$  é mais acessível e usável, em comparação com outro com um valor menor de  $M_{HEUA}$ .

Como uma segunda maneira de avaliar indiretamente a efetividade do processo PDWAU, além do experimento realizado apresentado no Capítulo 5, o instrumento de medição HEUA foi aplicado nos sistemas Agendamento e AgendAloca, a fim de comparar o valor de  $M_{HEUA}$ . Essa avaliação foi realizada por meio de um estudo de viabilidade, que será apresentada na próxima seção.

## 6.4 Estudo de viabilidade: aplicação da métrica $M_{HEUA}$

Um estudo de viabilidade foi conduzido para avaliar indiretamente o processo PDWAU (apresentado no Capítulo 4), a fim de buscar indícios de que ele pode ser útil no desenvolvimento de sistemas web com acessibilidade e usabilidade. Para isso, o instrumento de medição HEUA foi aplicado de forma rigorosa e imparcial por um especialista<sup>21</sup>, para avaliar a acessibilidade e usabilidade dos sistemas Agendamento e AgendAloca, conforme Tabela 17 (mesmos requisitos apresentados na Tabela 16).

**Tabela 17: Exemplo de avaliação pelo especialista.**

Requisitos	Agendamento			AgendAloca		
	Atende	Não atende	Não se aplica	Atende	Não atende	Não se aplica
<b>Requisito 1.1:</b> O sistema deve fornecer o caminho percorrido (clicável) pelo usuário.		x		x		
<b>Requisito 7.10:</b> O sistema deve oferecer uma ferramenta de busca para que os usuários localizem as opções disponíveis.	x			x		
<b>Requisito 9.2:</b> O sistema deve fornecer mensagens de erro ao usuário que devem ser as mais específicas possíveis.		x		x		

Durante a avaliação dos dois sistemas web (Agendamento e AgendAloca), o especialista justificou a opção que foi escolhida para cada requisito. As justificativas para os exemplos da Tabela 17 são apresentadas na Tabela 18. Ressalta-se que a justificativa para a classificação gerada pelo especialista para o sistema Agendamento pode ser encontrada no ANEXO VI e para o sistema AgendAloca pode ser encontrada no ANEXO VII.

<sup>21</sup> O avaliador foi considerado especialista, pois está terminando seu mestrado com ênfase em usabilidade e acessibilidade em sistemas web. Além disso, possui quatro anos de experiência em desenvolvimento web e dois anos de experiência em acessibilidade na web.

Tabela 18: Justificativa das opções escolhidas pelo especialista durante a avaliação.

	<b>Agendamento</b>	<b>AgendAloca</b>
<b>Requisito 1.1</b>	Não atende, pois mesmo com a presença do elemento de <i>breadcrumb</i> no sistema, ele não está corretamente implementado. Isso foi notado ao navegar pelo sistema e não visualizar o exato nome das páginas visitadas no <i>breadcrumb</i> , e em alguns casos ter as páginas diferentes daquelas visitadas pelo usuário expostas no mesmo.	Atende, devido à presença do <i>breadcrumb</i> no sistema.
<b>Requisito 7.10</b>	Atende, pois possui uma ferramenta de busca escondida dentro da opção “Open Panel” que, apesar de não estar sempre visível, possibilita que o usuário a utilize como um atalho para acessar algum conteúdo do sistema.	Atende, pois possui uma ferramenta de busca logo abaixo do cabeçalho da página e o usuário pode utilizar como atalho para acessar algum conteúdo no contexto atual do sistema.
<b>Requisito 9.2</b>	Não atende, pois diversas vezes foi possível verificar mensagens de erro genéricas e inclusive, mensagens não tratadas para o usuário contendo erros de código. Algumas vezes mensagens de erro não eram apresentadas.	Atende, pois nas diversas mensagens de erro testadas, todas retornaram mensagens bem específicas sobre qual foi o problema ocorrido.

Os resultados obtidos com a avaliação do sistema Agendamento são apresentados na Tabela 19.

Tabela 19: Aplicação de HEUA no Agendamento.

<b>Questões</b>	<b>REQ</b>	<b>NA</b>	<b>S</b>	<b>NS</b>
Q <sub>1</sub>	10	6	2	2
Q <sub>2</sub>	8	1	2	5
Q <sub>3</sub>	13	4	4	5
Q <sub>4</sub>	11	3	4	4
Q <sub>5</sub>	19	5	5	9
Q <sub>6</sub>	6	0	2	4
Q <sub>7</sub>	14	3	4	7
Q <sub>8</sub>	5	1	2	2
Q <sub>9</sub>	3	0	1	2
Q <sub>10</sub>	4	0	0	4

**Legenda:**  
**REQ** = quantidade de requisitos existentes para cada questão a ser respondida  
**NA** = total de requisitos que “não se aplica” ao sistema  
**S** = total de requisitos que o sistema “atende”  
**NS** = total de requisitos que o sistema “não atende”



Após aplicar HEUA no sistema Agendamento, aplicou-se a fórmula  $Q_t$  para todas as questões, conforme exemplo ( $Q_1$  e  $Q_2$ ) a seguir:

$$Q_t = \frac{S - NS}{REQ - NA} * 100$$

$$Q_1 = \frac{2 - 2}{10 - 6} * 100 = 0$$

$$Q_2 = \frac{2 - 5}{8 - 1} * 100 = -43$$

Ressalta-se que a fórmula  $Q_t$  considera que as questões tem o mesmo peso, pois independente do número de requisitos em cada questão, todas são fundamentais para alcançar um sistema com boa acessibilidade e usabilidade. Após encontrar o  $Q_t$  para cada questão, tem-se o resultado apresentado na Tabela 20.

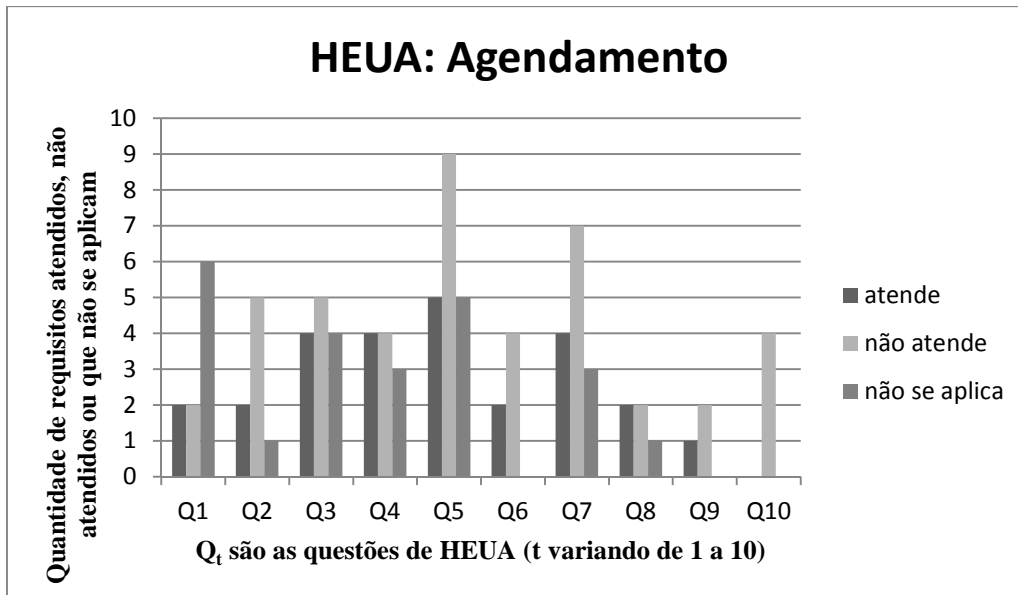
**Tabela 20:  $Q_t$ 's encontrados no Agendamento.**

<b>Q<sub>1</sub></b>	<b>Q<sub>2</sub></b>	<b>Q<sub>3</sub></b>	<b>Q<sub>4</sub></b>	<b>Q<sub>5</sub></b>	<b>Q<sub>6</sub></b>	<b>Q<sub>7</sub></b>	<b>Q<sub>8</sub></b>	<b>Q<sub>9</sub></b>	<b>Q<sub>10</sub></b>
0	-43	-12	0	-28	-33	-27	0	-33	-100

Com os resultados coletados, apresentados na Tabela 20, é possível encontrar o valor  $M_{HEUA}$  para o sistema Agendamento.

$$M_{HEUA} = \frac{\sum_{i=1}^{10} Q_t}{10} = -27,6$$

Se os resultados contidos na Tabela 20 forem transformados em gráfico (Figura 73), é possível observar que a maior parte das questões de HEUA não foi satisfeita no sistema Agendamento. Essa percepção, indicada pela Figura 73, foi confirmada pelo valor de  $M_{HEUA}$ , que foi -27,6.



**Figura 73: Resultados da aplicação de HEUA - Agendamento.**

De acordo com os resultados apresentados na Figura 73, é possível perceber que as heurísticas que merecem mais atenção no sistema Agendamento são: Q2, Q3, Q5, Q6, Q7, Q9 e Q10, que foram menos atendidas pelo sistema.

Depois de aplicar HEUA e encontrar  $M_{HEUA}$  para o sistema Agendamento, aplicou-se HEUA para o sistema AgendAloca. Para isso, o especialista seguiu os mesmos passos, que foram:

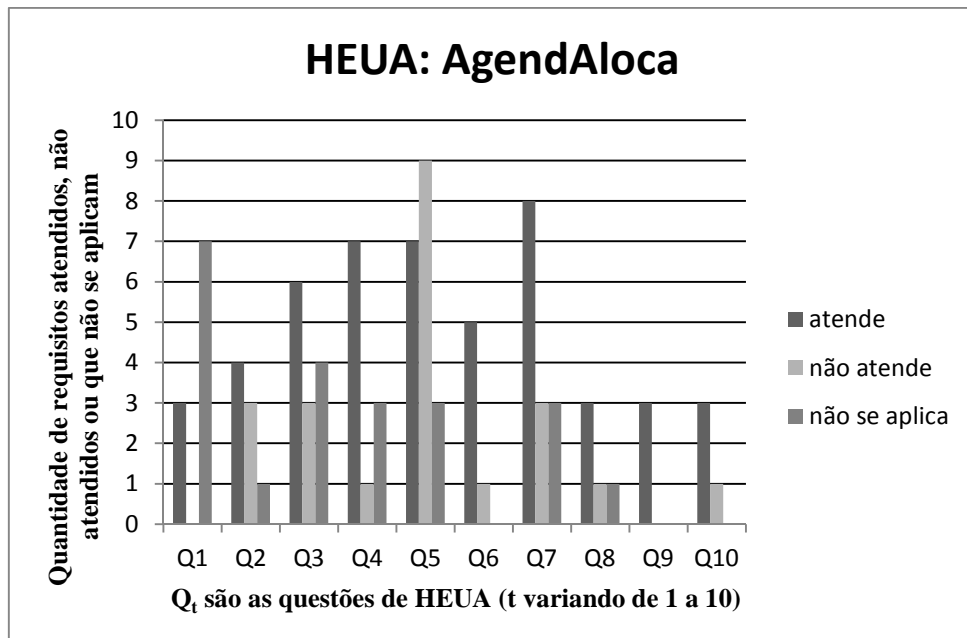
- Aplicar HEUA, verificando se o sistema de "atende", "não atende" ou "não se aplica" a todos os requisitos;
- Aplicar a fórmula  $Q_t$  para todas as questões;
- Encontrar  $M_{HEUA}$ ; e
- Observar os resultados.

Os resultados coletados para o sistema AgendAloca podem ser encontrados na Tabela 21.

**Tabela 21: Aplicação de HEUA no AgendAloca.**

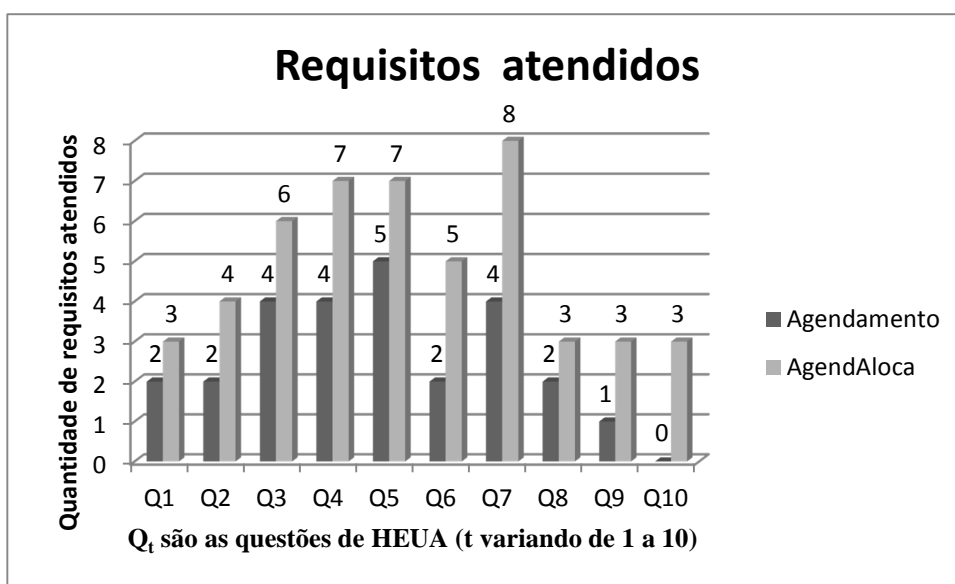
Questões	REQ	NA	S	NS	Q <sub>t</sub>
Q <sub>1</sub>	10	7	3	0	<b>100</b>
Q <sub>2</sub>	8	1	4	3	<b>14</b>
Q <sub>3</sub>	13	4	6	3	<b>33</b>
Q <sub>4</sub>	11	3	7	1	<b>75</b>
Q <sub>5</sub>	19	3	7	9	<b>-12</b>
Q <sub>6</sub>	6	0	5	1	<b>67</b>
Q <sub>7</sub>	14	3	8	3	<b>45</b>
Q <sub>8</sub>	5	1	3	1	<b>50</b>
Q <sub>9</sub>	3	0	3	0	<b>100</b>
Q <sub>10</sub>	4	0	3	1	<b>50</b>
<b>M<sub>HEUA</sub></b>					<b>+52,2</b>

É importante perceber que na Tabela 21, além dos valores de NA, S, NS, encontram-se os resultados das fórmulas Q<sub>t</sub> e M<sub>HEUA</sub>, aplicadas anteriormente para o sistema Agendamento também. É importante ressaltar que os valores de REQ serão sempre os mesmos, até que algum especialista adicione mais requisitos em cada uma das questões de HEUA (APÊNDICE G). Se os resultados apresentados na Tabela 21 forem transformados em gráficos, é possível perceber que a maioria das questões é atendida no sistema AgendAloca (Figura 74).

**Figura 74: Resultados da aplicação de HEUA - AgendAloca.**

É possível notar pelas Figuras 73 e 74, que a maioria das questões de HEUA não é atendida pelo sistema Agendamento, enquanto a maioria das questões é atendida no sistema AgendAloca. Outra maneira de conferir esses resultados é por meio da porcentagem de requisitos atendidos e não atendidos, na proporção dos requisitos que foram aplicados.

Nos resultados apresentados na Figura 73, é possível verificar que 61% dos requisitos não foram atendidos e apenas 39% foram atendidas, indicando que o sistema Agendamento precisa de muito esforço para se tornar usável e acessível. Já na Figura 74, é possível verificar que 31% dos requisitos não são atendidos, enquanto 69% dos requisitos são atendidos, indicando que o sistema AgendAloca precisa de menor esforço para se tornar usável e acessível. Portanto, comparando os resultados dos requisitos atendidos nos sistemas Agendamento e AgendAloca, é possível verificar claramente que o sistema AgendAloca atende um maior número de requisitos do que o sistema Agendamento (Figura 75).



**Figura 75: Quantidade de requisitos atendidos de HEUA – Agendamento e AgendAloca.**

Com os resultados obtidos, apresentados na Figura 75, percebe-se que HEUA pode ser usado para agregar preocupações acerca de usabilidade e acessibilidade em um instrumento único e se difere da avaliação experimental realizada anteriormente (ver Capítulo 5), pois resulta em um valor numérico ( $M_{HEUA}$ ), que permite comparações entre versões de um sistema ou mesmo entre diferentes sistemas web.

Considerando requisitos de usabilidade e acessibilidade juntos, o percentual dos requisitos satisfeitos no sistema Agendamento é inferior aos que foram obtidos no sistema AgendAloca. No entanto, outra avaliação pode ser realizada se os requisitos de usabilidade e acessibilidade forem considerados separadamente. Para isso, pode-se utilizar a Tabela 22.

**Tabela 22: Requisitos de HEUA separados em usabilidade e acessibilidade.**

	<b>Requisitos</b>
<b>Usabilidade</b>	<b>Q1.1, Q1.2, Q1.4, Q1.5</b> <b>Q2.1, Q2.2</b> Q3.1, <b>Q3.8</b> , Q3.9 Q4.1 Q5.1, <b>Q5.6</b> Q6.2 Q7.2, <b>Q7.10</b> , Q7.11 Q8.1, Q8.2, Q8.3, Q8.4 Q9.1 Q10.1, Q10.2
<b>Acessibilidade</b>	<b>Q1.1, Q1.2</b> , Q1.3, <b>Q1.4</b> , Q1.6, Q1.7, Q1.8, Q1.9, Q1.10 <b>Q2.1, Q2.2</b> , Q2.3, Q2.4, Q2.5, Q2.6, Q2.7, Q2.8 Q3.2, Q3.3, Q3.4, Q3.5, Q3.6, Q3.7, <b>Q3.8</b> , Q3.10, Q3.11, Q3.12, Q3.13 Q4.2, Q4.3, Q4.4, Q4.5, Q4.6, Q4.7, Q4.8, Q4.9, Q4.10, Q4.11 Q5.2, Q5.3, Q5.4, Q5.5, <b>Q5.6</b> , Q5.7, Q5.8, Q5.9, Q5.10, Q5.11, Q5.12, Q5.13, Q5.14, Q5.15, Q5.16, Q5.17, Q5.18, Q5.19 Q6.1, Q6.3, Q6.4, Q6.5, Q6.6 Q7.1, Q7.3, Q7.4, Q7.5, Q7.6, Q7.7, Q7.8, Q7.9, <b>Q7.10</b> , Q7.12, Q7.13, Q7.14 Q8.5 Q9.2, Q9.3 Q10.3, Q10.4

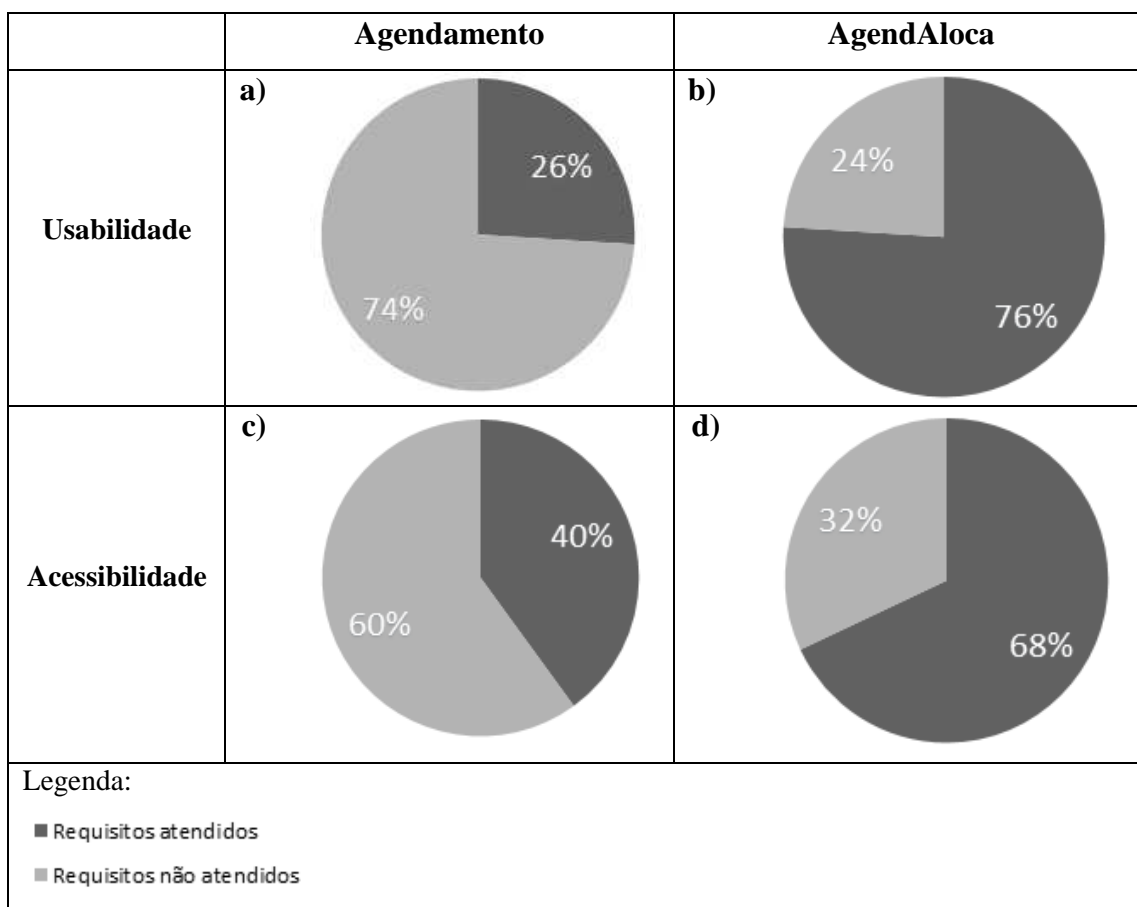
Na Tabela 22, os requisitos de HEUA foram divididos em requisitos de usabilidade e acessibilidade. Quando o requisito tinha apenas referência de usabilidade ele era classificado como requisito de usabilidade e quando o requisito tinha apenas referência de acessibilidade ele era classificado como requisito de acessibilidade. No entanto, quando o requisito tinha referência de usabilidade e acessibilidade (destacados em negrito), eles eram duplicados e classificados como requisitos de usabilidade e acessibilidade.

Os resultados apresentados anteriormente, nas Tabelas 19 e 21, oriundos da avaliação dos sistemas Agendamento e AgendAloca com o uso de HEUA pelo especialista, foram separados em resultados de usabilidade e acessibilidade (de acordo com a Tabela 22). O resultado dessa nova classificação pode ser encontrado na Tabela 23.

**Tabela 23: Resultado da aplicação de HEUA separados em usabilidade e acessibilidade – Agendamento e AgendAloca.**

Requisitos	Agendamento			AgendAloca		
	Atende	Não atende	Não se aplica	Atende	Não atende	Não se aplica
Usabilidade	5	14	4	13	4	6
Acessibilidade	24	35	19	41	19	18

Os resultados classificados na Tabela 23 foram transformados em gráficos e podem ser visualizados na Figura 76.



**Figura 76: Porcentagem de requisitos atendidos de usabilidade e acessibilidade – Agendamento e AgendAloca.**

De acordo com a Figura 76 percebe-se que: a) observando especialmente o sistema Agendamento, os requisitos de usabilidade alcançaram 26%, enquanto os de acessibilidade 40%. Portanto, o desenvolvedor deve corrigir prioritariamente os problemas de usabilidade na interface do Agendamento; b) no sistema AgendAloca, os requisitos de usabilidade e acessibilidade obtiveram percentuais melhores, 76% e 68% respectivamente. Entretanto, observa-se que os desenvolvedores devem priorizar as

questões de acessibilidade em detrimento da usabilidade, uma vez que o valor é inferior em termos de acessibilidade.

Como AgendAloca é um sistema que foi desenvolvido com base no processo PDWAU, já era de se esperar que os requisitos de usabilidade e acessibilidade fossem melhor satisfeitos do que os do sistema Agendamento. Comparando-se a usabilidade do Agendamento com a do AgendAloca, os requisitos atendidos no Agendamento foram de 26%, enquanto no AgendAloca foram de 76%. Assim, a diferença foi de 50% entre um e outro sistema, a qual é relativamente grande.

Quanto aos requisitos de acessibilidade, observadas na Figura 76 (c e d) no sistema Agendamento obteve-se a porcentagem de 40% de requisitos satisfeitos e no AgendAloca obteve-se 68%. Assim, a diferença foi de 28%. Comparando-se finalmente as diferenças em termos de usabilidade e acessibilidade entre um sistema e outro, podemos observar que a usabilidade foi mais atendida do que a acessibilidade. Uma questão em aberto que pode ser evidenciada é que requisitos de acessibilidade podem ser mais difíceis de serem resolvidos do que os requisitos de usabilidade.

Um destaque importante a se fazer é que, mesmo com o uso do PDWAU, 32% de requisitos de acessibilidade e 24% de requisitos de usabilidade não foram satisfeitos pelo sistema AgendAloca. Isso pode ter acontecido por três motivos: I) o sistema atendia um pouco o requisito e entre atender e não atender o especialista decidiu que não atendia; II) o especialista errou na classificação; e III) o sistema não atendia ao requisito, de fato.

Para o motivo I, tem-se o exemplo do **requisito 3.8**: “o sistema deve oferecer ao usuário a opção de alterar o idioma em que o conteúdo está escrito” (ver Apêndice G). Ao aplicar HEUA o especialista justifica: “*Não atende, pois apesar de existir esta opção em todas as telas através de um ícone com a bandeira do país da língua, não é todo o conteúdo do sistema que é transformado para a nova língua, como por exemplo, a página de ajuda*” (ver ANEXO VII). Nesse exemplo, o especialista realizou uma classificação severa, uma vez que todo o sistema é transformado para a nova língua selecionada, ficando somente o texto de ajuda inalterado.

Para o motivo II, pode-se citar o exemplo do **requisito 2.6**: “o sistema deve apresentar ícones que possam ser facilmente associados a objetos do mundo real”. Ao aplicar HEUA no sistema AgendAloca, o especialista justifica o requisito não atendido da seguinte maneira: “*Não atende, devido aos fatos dos ícones de acessibilidade presentes ao longo de todo o sistema, não serem facilmente associados a nenhum objeto*”

do mundo real”. Nesse caso, acredita-se que a classificação do especialista tenha sido equivocada, uma vez que os ícones de acessibilidade utilizados no sistema são ícones comumente usados para tais funções (ver Seções 4.5 e 4.6).

Para o motivo III, os argumentos que podem justificar a presença de problemas de usabilidade e acessibilidade são voltados para a própria diversidade de soluções tecnológicas que implementam os requisitos não funcionais dos sistemas web. Os desenvolvedores, em geral, possuem um foco no projeto das funcionalidades principais do sistema e desconhecem muitas vezes a gama de possibilidades dos recursos disponíveis na web. Por exemplo, quando se tem um menu de opções, são diversas as formas de implementação que podem ser escolhidas, no entanto, para cada uma delas, são requeridos elementos de codificação que habilitam o acesso e o uso amigável por parte dos usuários.

De modo geral, o processo PDWAU visa considerar o maior número de requisitos de acessibilidade e usabilidade, sistematicamente, no desenvolvimento de sistemas web. No entanto, a evolução das tecnologias disponíveis é sempre um desafio para a continuidade de melhorias no processo. Assim, é admissível que novas versões do processo sejam desenvolvidas, considerando tal desafio.



Se HEUA tivesse um grau intermediário de atender parcialmente ao requisito, a porcentagem de requisitos atendidos seria maior, mitigando o problema de classificação severa, como no exemplo do **requisito 3.8**. Já para amenizar o problema de classificação equivocada por parte do especialista, como citado no exemplo do **requisito 2.6**, HEUA poderia ser aplicado sempre por dois ou mais especialistas, fazendo a média da pontuação obtida.

Finalizando, observou-se que o sistema de Agendamento resultou em  $M_{HEUA} - 27,6$  e o sistema AgendAloca resultou em  $M_{HEUA} + 52,2$ , de acordo com a faixa de valores possíveis para  $M_{HEUA}$  (-100 a +100). Assim, é possível ter um roteiro para melhorias com base nos requisitos que tiveram valores negativos mais críticos, bem como comparar objetivamente dois sistemas, com base nos valores obtidos com a aplicação de HEUA.

Verificando-se então os requisitos que tiveram valores negativos, ou seja, que não foram atendidos pelo sistema Agendamento, mas que foram atendidos pelo sistema AgendAloca (Tabela 24), um mapeamento foi realizado.



**Tabela 24: Requisitos de acessibilidade e usabilidade não atendidos pelo sistema Agendamento e atendidos pelo sistema AgendAloca.**

Requisitos de acessibilidade e usabilidade	Agendamento	AgendAloca
O sistema deve fornecer o caminho percorrido (clicável) pelo usuário		
O sistema deve utilizar uma linguagem simples e clara para o usuário		
O sistema deve oferecer ao usuário a possibilidade de compreender a finalidade de cada link a partir do texto do link ou juntamente com o respectivo contexto do link		
O sistema deve oferecer ao usuário a possibilidade de compreender a finalidade de cada link a partir do texto do link		
O sistema deve fornecer cabeçalhos ( <i>headings</i> ) e rótulos ( <i>labels</i> ) que descrevem a sua finalidade		
O sistema deve oferecer ao usuário a possibilidade de navegar sequencialmente utilizando a tabulação do teclado		
O sistema deve oferecer um link (ou caminho direto) para que o usuário consiga voltar ao estado inicial do sistema		
O sistema deve deixar visível o foco nos elementos quando eles são selecionados via teclado		
O sistema deve diferenciar a apresentação de links de outros componentes que não aceitam ativação		
O sistema deve alterar o contexto somente quando um elemento é ativado		
O sistema deve utilizar cabeçalhos para organizar seções de conteúdos		
O sistema deve ser independente do recurso de barra de rolagem		
O sistema deve utilizar a mesma terminologia para referenciar somente as mesmas ações		
O sistema deve utilizar as mesmas cores, fontes, localização de navegação e layouts em toda a interface		
O sistema deve fornecer informações necessárias para a conclusão de tarefas específicas		
O sistema deve oferecer descrição textual em todos os seus elementos não textuais		
O sistema deve oferecer opção de contraste para textos e imagens de texto no mínimo de 4.5:1		
O sistema deve oferecer opção de contraste para textos e imagens de texto no mínimo de 7:1		
O sistema deve oferecer opção de zoom para textos e imagens de texto		

O sistema deve fornecer ferramentas ou informações necessárias para que o usuário acesse no sistema web de modo adequado		
O sistema deve detectar e avisar ao usuário se ele tem todos os plug-ins necessários para que o sistema funcione normalmente		
O sistema deve fornecer mensagens de erro ao usuário que devem ser as mais específicas possíveis		
O sistema deve fornecer informações sobre as tarefas que podem ser realizadas		
O sistema deve fornecer ao usuário informações adicionais sobre o proprietário/desenvolvedor/responsável pelo sistema		
O sistema deve fornecer um canal simples e pragmático para que o usuário possa informar sobre erros verificados		

A partir do mapeamento realizado com a lista de requisitos não atendidos pelo sistema Agendamento, foi possível perceber que alguns requisitos de acessibilidade e usabilidade foram atendidos pelo sistema AgendAloca devido ao uso do Processo PDWAU. Se o sistema Agendamento utilizasse o processo PDWAU, mais especificamente as tarefas/subtarefas listadas na Tabela 25, possivelmente os trinta e um requisitos não atendidos (citados na Tabela 24) seriam atendidos, o que melhoraria em 33.3% o resultado final de  $M_{HEUA}$ .

**Tabela 25: Mapeamento do Processo PDWAU.**

PDWAU		
Atividade	Ação	Tarefas/subtarefas
Comunicação (ver Figura 20)	Elicitação	<b>Tarefa 4.a.</b> - inserir no documento de requisitos, os requisitos de acessibilidade e usabilidade (consultando o APÊNDICE B).
		<b>Tarefa 6.a.</b> – indicar o esforço demandado em cada funcionalidade de acessibilidade e usabilidade.
Planejamento (ver Figura 21)	Estimação	<b>Tarefa 2.a.</b> – desenvolver o cronograma microscópico (detalhado), considerando casos de uso de acessibilidade e usabilidade.
	Monitoração	<b>Tarefa 1.a.</b> – utilizar uma ferramenta para monitoração.

Modelagem (ver Figura 22)	Análise	<b>Tarefa 2.a.</b> – desenvolver um ou mais modelos para analisar os requisitos do sistema, inserindo os requisitos de acessibilidade e usabilidade no modelo adotado.
	Projeto	<b>Tarefa 1.b.i.</b> – organizar a interface com critérios de acessibilidade e usabilidade.
		<b>Tarefa 2.b.i.</b> – no projeto de navegação, considerar funções de acessibilidade e usabilidade.
Construção (ver Figura 23)	Testes	<b>Tarefa 2.c.</b> – realizar testes de usabilidade para verificar se os requisitos foram de fato considerados.
Implantação (ver Figura 24)	Avaliação	<b>Tarefa 1.a.</b> – avaliar com usuários a versão liberada.
		<b>Tarefa 2.</b> – gerar um relatório com modificações para o próximo incremento com os requisitos não atendidos, caso necessário.

É importante ressaltar que a formalização da métrica  $M_{HEUA}$ , bem como sua aplicação nos sistemas web Agendamento e AgendAloca, foram parcialmente publicadas e pode ser encontrada em Dias, Fortes e Masiero (2014).

## 6.5 Considerações finais

Na literatura há algumas pesquisas relacionadas às diretrizes que devem ser implementadas nos sistemas web, em termos de requisitos de usabilidade e acessibilidade. No entanto, falta uma medida que ajude os especialistas a verificar o quanto estes requisitos, especialmente em relação a ambas as questões de usabilidade e acessibilidade, são apresentados em um sistema web.

A falta de instrumentos para ajudar os desenvolvedores e especialistas a monitorar a evolução dos níveis de usabilidade e acessibilidade de acordo com o desenvolvimento de novas versões de sistemas web leva a problemas de barreiras que os usuários finais podem enfrentar na web. Pensando nisso, formalizou-se  $M_{HEUA}$ , uma métrica que apresenta valores sobre a quantidade de acessibilidade e usabilidade de um sistema web, podendo ser utilizada para comparar objetivamente dois sistemas, com base nos valores obtidos com a aplicação de HEUA.

Os resultados da aplicação do instrumento de medição HEUA nos sistemas Agendamento e AgendAloca foram em direção aos resultados encontrados no

experimento apresentado no Capítulo 5. Esse resultado confirma os indícios de que o uso do processo PDWAU pode produzir sistemas web com melhores índices de acessibilidade e usabilidade, se comparado a sistemas web desenvolvidos de maneira *ad-hoc*, sendo, portanto recomendado devido a esse benefício.

---

## *Conclusões*

---

### *7.1 Considerações iniciais*

Prover acessibilidade na web para um público com características heterogêneas não significa apenas desenvolver sistemas compatíveis com as recomendações e diretrizes, mas oferecer recursos para que usuários pouco experientes possam superar as barreiras que enfrentam em sua interação com os sistemas web, e que, muitas vezes, os levam à desistência total desse tipo de experiência. Portanto, garantir acessibilidade e usabilidade no desenvolvimento de sistemas web não auxilia somente desenvolvedores e organizações no desenvolvimento de sistemas com boa qualidade, mas auxilia principalmente na promoção do desenvolvimento de uma web mais inclusiva.

De acordo com Lazar e Olalere (2011), é preferível dedicar mais esforço no desenvolvimento de sistemas web em que os critérios de acessibilidade e usabilidade sejam considerados no início e durante todo o processo de desenvolvimento, ao invés de serem considerados apenas na ação de avaliação, no fim do processo. Essa afirmação foi reforçada com o desenvolvimento desta tese.

É importante ressaltar que, mesmo que a acessibilidade e a usabilidade sejam consideradas desde o início, é necessário que haja verificação constante e contínua desses requisitos no sistema para evitar que pequenas modificações em seu conteúdo ou em sua estrutura comprometam o projeto usável e acessível inicial. Surge, assim, um novo

desafio: desenvolver, administrar e manter sistemas em conformidade com as diretrizes de acessibilidade e princípios de usabilidade para que sejam, ao mesmo tempo, atuais, fáceis de usar, acessíveis e atraentes (Ferreira e Nunes, 2008).

Esta tese compartilha do ideal de que os sistemas computacionais estão cada vez mais presentes na vida das pessoas e é preciso que eles atendam mais usuários, independentemente de suas características físicas, cognitivas, emocionais, ou de necessidades de interação decorrentes de diferentes dispositivos em ambientes de uso diversificados.

A partir dos estudos realizados nesta tese, o foco foi sempre no apoio à inserção de requisitos de acessibilidade e usabilidade em sistemas web. Nesse sentido, reforça-se a necessidade de se considerar, durante o processo de desenvolvimento, os requisitos de usuários que tenham habilidades no uso de TICs, bem como os requisitos daqueles que ainda não estão familiarizados com os mesmos, como os idosos, bem como os requisitos específicos de pessoas com deficiência etc.

Espera-se que o trabalho aqui apresentado apoie os desenvolvedores na identificação das diferentes necessidades dos usuários, desenvolvendo sistemas web com acessibilidade e usabilidade, respeitando e reconhecendo a riqueza da heterogeneidade e fomentando a inclusão social de todos os cidadãos.

## **7.2 Principais contribuições**

A contribuição mais expressiva do trabalho realizado foi a definição do processo PDWAU, que foi apresentado no Capítulo 4; ele foi definido para apoiar o desenvolvimento de sistemas web com foco em acessibilidade e usabilidade. Para que a definição desse processo fosse possível, utilizou-se como base o processo genérico de Pressman e Lowe (2009), bem como trabalhos selecionados e combinados que foram encontrados na extensa revisão da literatura realizada (Capítulos 2 e 3).

O processo PDWAU foi avaliado de forma prática com o desenvolvimento do AgendAloca, um sistema web usado para o agendamento de trabalhos de conclusão de curso no ICMC-USP. A partir da interface desse sistema, foi desenvolvido um padrão de interface, apresentado na Seção 4.6, que por sua vez foi utilizado para o desenvolvimento de outro sistema web, o AWMo (ver Figura 59 – Capítulo 4). O sistema AWMo foi testado por deficientes visuais, obtendo resultados satisfatórios. Esse padrão de interface pode ser útil para desenvolvedores, que ao invés de desenvolver uma interface

---

completamente nova para inserir requisitos básicos de usabilidade e acessibilidade, podem direcionar seus esforços no desenvolvimento nas funcionalidades específicas do sistema (ver Figura 60 – Capítulo 4).

Adicionalmente, a métrica  $M_{HEUA}$  foi formalizada. Nesta tese,  $M_{HEUA}$  foi utilizada para avaliar o processo PDWAU, por meio de um estudo de viabilidade, comparando os sistemas web Agendamento e AgendAloca. Ela permite que um conjunto abrangente de requisitos seja representado por um valor que expressa uma medida de avaliação do grau de acessibilidade e usabilidade de um sistema web, bem como possibilita comparação entre dois ou mais sistemas web.

Outra contribuição derivada desta tese foi a proposição do ACCESSA, um método que pode ser utilizado para avaliar sistemas web, quanto à acessibilidade e usabilidade. O método também pode ser utilizado para reengenharia de sistemas web já existentes ou legados, que buscam a inserção de requisitos de usabilidade e acessibilidade, sem alterar as funcionalidades básicas do sistema.

O método ACCESSA e a métrica  $M_{HEUA}$  podem ser utilizados em sistemas web, independentemente do processo PDWAU, para expressar também o esforço de possíveis melhorias de usabilidade e acessibilidade.

Os resultados coletados com a realização do experimento (Capítulo 5) e do estudo de viabilidade (Capítulo 6) apresentam indícios de que o processo PDWAU pode apoiar no desenvolvimento de sistemas web acessíveis e usáveis. Dessa maneira, é possível responder à pergunta inicial de pesquisa desta tese (ver Seção 1.3); uma vez que o PDWAU foi formalizado com base principalmente em um referencial teórico.

Espera-se que o processo PDWAU possa ser utilizado por desenvolvedores que se apoiam em processos de desenvolvimento. Acredita-se que isso seja viável, pois o PDWAU foi construído com base no processo genérico, definido por Pressman e Lowe (2009). Espera-se ainda, que o uso do PDWAU leve a uma interface com reduzidos problemas de usabilidade e acessibilidade, evitando retrabalho após a implantação do sistema e diminuindo custos.

Adicionalmente, lacunas existentes na literatura, identificadas por Lazar e Olalere (2011), foram atendidas (ver Seção 1.3), conforme segue: a) a primeira lacuna sugerida, em que é necessário inserir a acessibilidade no início do processo de desenvolvimento; b) a sugestão 3, em que é necessária uma abordagem de ciclo de vida que integre processos de engenharia de usabilidade e acessibilidade com engenharia de software; e c) a sugestão

5, em que requisitos para projetar e testar a acessibilidade devem ser claramente indicados nos artefatos do sistema, os quais devem ser avaliados em termos de sua compreensão e disposição para alcançar acessibilidade.

### **7.3 Limitações e trabalhos futuros**

Com o desenvolvimento da presente pesquisa, pôde-se observar que apesar dos indícios positivos obtidos por meio das avaliações realizadas, há muito que se estudar e desenvolver em termos de pesquisas sobre acessibilidade e usabilidade, tratando as várias questões para melhoria do cenário atual. Neste sentido, algumas limitações e trabalhos futuros podem ser apontados.

Uma limitação que pode dificultar o uso do PDWAU está relacionada com a necessidade do desenvolvedor acompanhar paralelamente o arcabouço genérico de Pressman e Lowe (2009). Nesse arcabouço genérico, estão especificadas cada atividade, ação, tarefas e subtarefas que devem ser realizadas durante o desenvolvimento de sistemas web. Esta pesquisa concentrou esforços na descrição do processo PDWAU, somente em termos das características de acessibilidade e usabilidade, inseridas no arcabouço genérico.

No processo proposto nesta tese, a preocupação inicial era tratar questões de acessibilidade. Conforme comentado anteriormente, acessibilidade e usabilidade são conceitos fortemente relacionados, percebeu-se então a necessidade de incluir questões de usabilidade também. Dessa maneira, a usabilidade foi obtida de modo derivado. Um trabalho futuro importante seria incluir no processo PDWAU tarefas e subtarefas relativas aos modelos de tarefas de usuários, conforme previstos em processos com design centrados nos usuários.

A evolução das tecnologias disponíveis é sempre um desafio para a continuidade das melhorias no processo. Assim, é admissível que novas versões do processo PDWAU sejam desenvolvidas, considerando tal desafio. Vale ressaltar que nesta tese, o foco da pesquisa se concentrou em identificar métodos, técnicas e ferramentas para a formalização do PDWAU, a fim de procurar fazer com que um maior número de requisitos de acessibilidade e usabilidade seja considerado em sistemas web. Desse modo, não houve um aprofundamento dos detalhes técnicos de implementação relacionados tanto aos critérios de sucesso trabalhados aqui, quanto aos princípios de usabilidade.



Durante o desenvolvimento desta tese, somente um sistema web foi desenvolvido adotando-se o processo PDWAU. Algumas ações (negociação, escalonamento e entrega) não foram realizadas devido ao fato desse sistema web (AgendAloca) ser um sistema universitário desenvolvido por alunos de iniciação científica. Estudos de caso no âmbito empresarial e em outros domínios de aplicação podem revelar outras perspectivas na adoção do processo. Outro trabalho futuro pode ser a realização de experimentos com maior número de usuários (com e sem deficiências).

Quanto ao método ACCESSA, faz-se necessária uma formalização detalhada, bem como novos experimentos em outros domínios de aplicação, a fim de verificar a sua utilização como um método para a melhoria da usabilidade e da acessibilidade de sistemas web existentes.

Sobre o instrumento de medição HEUA, bem como da métrica  $M_{HEUA}$ , é possível fazer uma extensão ponderada considerando quais requisitos são mais ou menos importantes, baseado nos diferentes níveis de prioridade do WCAG, por exemplo. Essa extensão pode ser útil ao especialista, dependendo do tipo de usuário que ele estiver considerando no desenvolvimento e do domínio da aplicação em questão. Além disso, é necessário desenvolver um apoio ferramental ao desenvolvedor, que facilite sua aplicação e produza formas alternativas de apresentação dos resultados. Outros requisitos de acessibilidade e usabilidade podem também ser inseridos em HEUA para que os valores de  $M_{HEUA}$  sejam pautados em outras referências da área.

Ainda sobre HEUA, ele deve ser aplicado a outros sistemas web, a fim de verificar as facilidades, dificuldades e utilidade ao adotá-lo. Uma nova versão de HEUA com níveis intermediários, incluindo a opção “atende parcialmente ao requisito”, pode também ser desenvolvida e comparada com a versão atual.

Diante das tendências atuais mais relacionadas com pesquisas e trabalhos sobre processos de software, os seguintes trabalhos futuros podem ser vislumbrados: a) no contexto de computação em nuvem, uma possível questão de pesquisa seria investigar quais requisitos os sistemas web acessíveis e usáveis desenvolvidos com o PDWAU poderiam contribuir para soluções de computação em nuvem. Ainda nesse contexto, como o processo pode vir a atender esses requisitos identificados; e b) no contexto de computação social, uma questão de pesquisa poderia ser formulada quanto ao potencial de colaboração entre engenheiros de software e usuários durante a adoção do processo PDWAU.

Finalmente, um trabalho futuro que pode ser bastante útil para os profissionais, que trabalham com o dia-a-dia do desenvolvimento de sistemas web, é o desenvolvimento de um apoio automatizado para o gerenciamento das atividades do processo PDWAU.

Percebe-se que há necessidade ainda de muita pesquisa a ser desenvolvida nas áreas de acessibilidade e de usabilidade. Construir interfaces usáveis e acessíveis para um público tão heterogêneo é um grande desafio. Finalmente, é importante ressaltar que esta tese é uma demonstração de que agregar trabalhos já existentes na literatura, ao invés de apenas contrastá-los, é um apoio fundamental para a formalização de novas ideias.

# Referências Bibliográficas

---

(Abou-zahra, 2008) Abou-zahra, S. Web Accessibility Evaluation. Capítulo de Livro - Human-Computer Interaction Series, Part II. In: HARPER, S.; YESILADA, Y. Web Accessibility. 1. ed. London: Springer, 2008. p. 79-106.

(Acerbis *et al.*, 2008) Acerbis, R.; Bongio, A.; Brambilla, M.; Butti, S.; Ceri, S.; Fraternali, P. *Web applications design and development with webML and webRatio 5.0*. In: Objects, Components, Models and Patterns, Springer Berlin Heidelberg, 2008, p. 392-411 (Lecture Notes in Business Information Processing, v.11)

(Akhter *et al.*, 2009) Akhter, F.; Buzzi, M. C.; Buzzi, M.; Leporini, B. *Conceptual Framework: How to Engineer Online Trust for Disabled Users*. Web Intelligence and Intelligent Agent Technologies. WI-IAT '09. IEEE/WIC/ACM International Joint Conferences on. Vol. 3, pp. 614-617, 2009.

(Alexander *et al.*, 1977) Alexander, C.; Ishikawa, S.; Silverstein, M.; Jacobson, M.; Fiksdahl-King, I.; Angel, S. *A pattern language: towns, buildings, construction*. New York: Oxford University Press, 1977. 1171 p.

(Alexander, 2006) Alexander, D. *Usability and Accessibility: Best friends or Worst enemies?* Disponível em: [http://www.valaconf.org.au/vala2006/papers2006/99\\_Alexander\\_Final.pdf](http://www.valaconf.org.au/vala2006/papers2006/99_Alexander_Final.pdf). Acesso em: janeiro de 2014.

(Arrue, Vigo e Abascal, 2007) Arrue, M.; Vigo, M.; Abascal, J. *Automatic Evaluation of Mobile web Accessibility*. Lecture Notes in Computer Science, Universal Access in Ambient Intelligence Environments, 2007.

(Asakawa, 2005) Asakawa C. *What's the web is Like If You Can't See It?* International Cross Disciplinary Workshop on web Accessibility (W4A), Edinburgh, UK, 23rd-26th May 2005.

(Baguma e Lubega, 2008) Baguma, R.; Lubega, J. T. *A web design framework for improved accessibility for people with disabilities (WDFAD)*. In: Proceedings of the 2008 International Cross-Disciplinary Conference on web Accessibility W4A, pp. 134-140, 2008.

(Baguma *et al.*, 2009) Baguma, R.; Stone, R.G.; Lubega, J.T.; Weide, T.P.van der. *Integrating Accessibility and Functional Requirements*. Lecture Notes in Comp.Science, Universal Access in HCI. Applicat.and Serv., Springer-Verlag Berlin Heidelberg, 2009.

(Bailey e Pearson, 2011) Bailey, C.; Pearson, E. *Development and trial of an educational tool to support the accessibility evaluation process*. In Proceedings of the International Cross-Disciplinary Conference on web Accessibility (W4A '11). ACM, New York, NY, USA, Article 2, 10 pages, 2011.

- (Basili e Weiss, 1984) Basili, V.R.; Weiss, D. M.. 1984. *A Methodology for Collecting Valid Software Engineering Data*. IEEE Trans. Softw. Eng. 10, 6 (November 1984), 728-738.
- (Bianchini, 2008) Bianchini, S. L. Avaliação de Métodos de desenvolvimento de aplicações web. Dissertação de Mestrado do Instituto de Ciências Matemáticas e de Computação (ICMC-USP), 113 p., 2008.
- (Bigham e Ladner, 2007) Bigham, J. P.; Ladner, R. E. *Accessmonkey: A Collaborative Scripting Framework for web Users and Developers*. Banff, Canada. Co-Located with the 16th International World Wide web Conference, W4A 2007 Technical Paper May 07–08, issue 89, pp.3–6, 2007.
- (Bittar, 2013) Bittar, T. J. Uma abordagem de apoio a boas práticas para desenvolvimento de aplicações web acessíveis. 2013. 255p. Tese (Doutorado - Programa de Pós-Graduação em Ciências de Computação e Matemática Computacional). Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo, São Carlos, 2013.
- (Bonifácio *et al.*, 2011) Bonifácio, B.; Fernandes, P.; Oliveira, H. A. B. F.; Conte, T. *UBICUA: A customizable usability inspection approach for web mobile applications*. In Proceedings of the IADIS International Conference on WWW/Internet, 2011.
- (Borchers, 2001) Borchers, J. O. *A pattern approach to Interaction Design*. Chichester: John Wiley & Sons, UK, 2001. 264p.
- (Brajnik, 2006) Brajnik, G. *web accessibility testing: When the method is the culprit*. In: Computers Helping People with Special Needs, Springer Berlin/ Heidelberg, 2006, p. 156–163 (Lecture Notes in Computer Science, v.4061).
- (Brajnik, 2008a) Brajnik, G. *Beyond Conformance: The Role of Accessibility Evaluation Methods*. Proceedings of the 2008 international workshops on web Information systems Engineering. Auckland. New Zealand. 2008.
- (Brajnik, 2008b) Brajnik, G. *A comparative test of web accessibility evaluation methods*. In: Assets '08: Proceedings of the 10th international ACM SIGACCESS conference on Computers and accessibility, New York, NY, USA: ACM, 2008b, p. 113–120.
- (Brajnik, Yesilada e Harper, 2010) Brajnik, G.; Yesilada, Y.; Harper, S. *Testability and validity of WCAG 2.0: the expertise effect*. In: Proceedings of the 12th international ACM SIGACCESS conference on Computers and accessibility (ASSETS'10, v.1), New York, NY, USA: ACM, 2010, p. 43–50.
- (Brink, 1998) Brink, T. *Usability first: Groupware applications*, 1998. Disponível em: <http://www.usabilityfirst.com/groupware/applications.txt>. Acesso em: abril de 2014.

- (Bühler *et al.*, 2006) Bühler, C.; Heck, H.; Perlick, O.; Nietzio, A.; Ulltveit-Moe, N. *Interpreting results from large scale automatic evaluation of web accessibility*. In: Proceedings of the 10th international conference on Computers Helping People with Special Needs (ICCHP'06, v.1), Berlin, Heidelberg: Springer-Verlag, 2006, p. 184–191.
- (Chen, Yesilada e Harper, 2010) Chen, T.; Yesilada, Y.; Harper, S. *What input errors do you experience? Typing and pointing errors of mobile web users*. Journal of HCI Studies, 68(3), pp. 138-157, 2010.
- (Christiernin, 2010) Christiernin, L.G. *Guiding the designer: A radar diagram process for applications with multiple layers*. Interacting with Computers, 22(2), pp. 107-122, 2010.
- (Cluster, 2006) Cluster, W. W. A. B. *Unified web evaluation methodology (UWEM 1.0)*. 2006. Disponível em: <http://www.wabcluster.org/uwem/>. Acesso em: março de 2014.
- (Connell *et al.*, 1997) Connell, B, Jones, M, Mace, R, Mueller, J, Mullick, A, Ostroff, E, Sanford, J, Steinfeld, E, Story, M, & Vanderheiden, G (1997). The principles of universal design: Version 2.0. Raleigh, NC: The Center for Universal Design.
- (Conte *et al.*, 2007) Conte, T.; Massolar, J.; Mendes, E.; Travassos, G. H. *Usability Evaluation based on web Design Perspectives*. In Proceedings of the First International Symposium on Empirical Software Engineering and Measurement, pp. 146 – 155, 2007.
- (Conte, Mendes e Travassos, 2005) Conte, T.; Mendes, E.; Travassos, G. H. *Processos de Desenvolvimento para Aplicações web: Uma Revisão Sistemática*. Proceedings of the 11th Brazilian Symposium on Multimedia and web (webMedia 2005), v. 1; 2005, p. 107-116.
- (Cooper e Rejmer, 2001) Cooper, M.; Rejmer, P. *Case study: localization of an accessibility evaluation*. In Proceedings of SIGCHI Conference on Human Aspects in Computing Systems (CHI'01), ACM Press, New York, NY, 2001.
- (Deshpande *et al.*, 2002) Deshpande, Y.; Murugesan, S.; Ginige, A.; Hansen, S.; Schwabe, D.; Gaedke, M.; White, B. *web Engineering*. Journal of web Engineering, 1(1), pp. 3–17, 2002.
- (Dias *et al.*, 2010) Dias, A. L.; Fortes, R. P. M.; Masiero, P. C.; Goularte, R. *Uma Revisão Sistemática sobre a inserção de acessibilidade nas fases de desenvolvimento da Engenharia de Software em sistemas web*. In: Workshop sobre Aspectos da Interação Humano-Computador na web Social, 2010, Belo Horizonte. Simpósio de Fatores Humanos em Sistemas Computacionais. Porto Alegre: Sociedade Brasileira de Computação, 2010. v. 2. p. 39-48.

(Dias *et al.*, 2013) Dias, A. L., Fortes, R. P. M., Masiero, P. C., Watanabe, W. M., Ramos, M. E. *An approach to improve the accessibility and usability of existing web system*. In: Proceedings of the 31st ACM international conference on Design of communication (SIGDOC '13). ACM, New York, NY, USA, 39-48.

(Dias, 2003) Dias, C. *Usabilidade na web: criando portais mais acessíveis*. Rio de Janeiro: Alta Books, 2003.

(Dias, 2009) Dias, A. L. Cores e seus significados no senso comum para apoiar a colaboração na aprendizagem via web: uma solução em forma de Padrões de Design Motivacionais. 2009. 113f. Dissertação (Mestrado em Ciência da Computação) Departamento de Computação, Universidade Federal de São Carlos, São Carlos, 2009.

(Dias, Fortes e Masiero, 2012) Dias, A. L.; Fortes, R. P. M.; Masiero, P. C. *Increasing the Quality of web systems: By inserting requirements of accessibility and usability*. In: 8th International Conference on the Quality of Information and Communications Technology (QUATIC '12). Thematic Tracks: Quality in ICT Requirements Engineering, September 3-6, 2012, Lisboa, Portugal. p. 224-229.

(Dias, Fortes e Masiero, 2014) Dias, A. L.; Fortes, R. P. M.; Masiero, P. C. *HEUA: a heuristic evaluation with usability and accessibility requirements to assess web systems*. In Proceedings of the 11<sup>th</sup> Web for All Conference (W4A '14). ACM, New York, NY, USA, Article 18, 4 pages.

(eMAG, 2005) Recomendações de acessibilidade para a construção e adaptação de conteúdos do governo brasileiro na Internet. eMAG, Modelo de acessibilidade do Governo Eletrônico, Versão 2.0, 2005. Disponível em: <http://www.consultas.governoeletronico.gov.br/ConsultasPublicas/consultas.do;jsessionid=C13E5697FF43725DB0020A8BD15E77AB?acao=exibir&id=17>. Acesso em: janeiro de 2014.

(ErgoList, 1998) Projeto ErgoList. Home Page do Projeto, 1998. Disponível em: <http://www.labiutil.inf.ufsc.br/ergolist/>. Acesso em: outubro de 2014.

(Fernandes *et al.*, 2013) Fernandes, N.; Batista, A.S.; Costa, D.; Duarte, C.; Carriço, C. 2013. *Three web accessibility evaluation perspectives for RIA*. In Proceedings of the 10<sup>th</sup> International Cross-Disciplinary Conference on web Accessibility (W4A '13). ACM, New York, NY, USA, Article 12, 9 pages.

(Ferreira e Nunes, 2008) Ferreira, S. B. L.; Nunes, R. R. e-usabilidade. Rio de Janeiro: Ed. LTC, 179 p., 2008.

(Fincher *et al.*, 2003) Fincher, S.; Finlay, J.; Greene, S.; Jones, L.; Matchen, P. *Perspectives on HCI patterns: concepts and tools*. Conference on Human Factors in Computing Systems, CHI03. Ft. Lauderdale, Florida, USA: ACM. 2003. p. 1044 - 1045.

(Fons et al., 2003) Fons, J.; Valderas, P.; Ruiz, M.; Rojas, G.; Pastor, O. OOWS: A method to develop web applications from web-oriented conceptual models. In: In: International Workshop on Web Oriented Software Technology (IWWOST). (2003) 65-70, 2003, pp. 65-70.

(Fortes et al., 2005) Fortes, R.P.M.; Pansanato, L.T.E.; Lara, S.M.A.; Pimenta, A.P. Minicursos do webmedia 2005, v. 1. Belo Horizonte-MG: FuMARC Editora, 197-226 p., 2005.

(Fortes et al., 2012) Fortes, R. P. M., Dias, A. L., Grillo, F. D. N., Masiero, P. C. *Creating a project towards universal access: is it possible?* In: INTERACT 2013 Workshop on Rethinking Universal Accessibility: A broader approach considering the digital gap, Cape Town (South Africa), 2013. pp. 1-8.

(Freire et al., 2008) Freire, A.P.; Fortes, R.P.M.; Turine, M.A.S.; Paiva, D. M. B. *An evaluation of web accessibility metrics based on their attributes.* In: SIGDOC '08: Proceedings of the 26th annual ACM international conference on Design of communication, New York, NY, USA: ACM, 2008, pp. 73-80.

(Freire et al., 2009) Freire, A.P.; Power, C.; Petrie, H.; Tanaka, E.H.; Rocha, H.V.; Fortes, R. P. *web accessibility metrics: Effects of different computational approaches.* In: Proceedings of the 5th International Conference on Universal Access in Human-Computer Interaction. Part III: Applications and Services, Berlin, Heidelberg: Springer-Verlag, 2009, p. 664–673 (UAHCI '09, v.1).

(Freire, 2012) Freire, A. P. *Disabled people and the web: User-based measurement of accessibility.* PhD thesis, University of York, 2012.

(Freire, Bittar e Fortes, 2008) Freire, A.P.; Bittar, T. J.; Fortes, R. P. M. *An approach based on metrics for monitoring web accessibility in brazilian municipalities' web sites.* In: SAC'08: Proceedings of the 2008 ACM symposium on Applied computing, New York, NY, USA: ACM, 2008, pp. 2421-2425.

(Freire, Goularte e Fortes, 2007) Freire, A. P.; Goularte, R.; Fortes, R. P. M. *Techniques for developing more accessible web applications: a survey towards a process classification.* In: SIGDOC '07: Proceedings of the 25th annual ACM international conference on Design of communication, New York, NY, USA: ACM, pp. 162-169, 2007.

(Freire, Russo e Fortes, 2008) Freire, A.P.; Russo, C.M.; Fortes, R. P. M. *The perception of accessibility in web development by academy, industry and government: a survey of the brazilian scenario.* New Review of Hypermedia and Multimedia, v. 14, n. 2, pp. 149-175, 2008.

(Gamma et al., 1995) Gamma, E.; Johnson, R.; Helm, R.; Vlissides, J. *Design Patterns: Elements of Reusable Object-Oriented Software.* [S.l.]: Addison Wesley, 1995. 416 p.

- (García, González e García, 2009) García, Y. S. M.; González, B. S. M.; García, J. C. Y. *Prosumers and accessibility: How to Ensure a Productive Interaction*. W4A '09: Proceedings of the 2009 International Cross-Disciplinary Conference on web Accessibility, 2009.
- (Garzotto *et al.*, 1999) Garzotto, F.; Paolini, O.; Bolchini, D.; Valenti, S. *"Modeling-by-patterns" of web applications*. Congrès on Advances in conceptual modeling. Paris: Springer. 1999. pp. 293-306.
- (Goette, Collier e White, 2006) Goette, T.; Collier, C.; White, J. D. *An exploratory study of the accessibility of state government web sites*. *Univers. Access Inf. Soc.*, v. 5, n. 1, pp. 41-50, 2006.
- (Gregg, 2009) Gregg, D. G. *Developing a collective intelligence application for special education*. *Decision Support Systems*, Vol. 47, No. 4, pp. 455-465, 2009.
- (Grillo, Fortes e Lucrédio, 2012) Grillo, F. Del Nero; Fortes, R. P.M.; Lucrédio, D. Towards collaboration between sighted and visually impaired developers in the context of Model-Driven Engineering. In: 1. Workshop GMLD (on Graphical Modeling Language Development), 2012, Lyngby, Dinamarca. Joint Proceedings of co-located Events at the 8th European Conference on Modelling Foundations and Applications (ECMFA 2012). Copenhagen, Denmark: Technical University of Denmark - DTU Informatics, 2012. v. 1. pp. 241-251.
- (Guedes, 2008) Guedes, G. *Interface Humano Computador: prática pedagógica para ambientes virtuais*. Teresina: EDUFPI, 218 p., 2008.
- (Hailpern *et al.*, 2009) Hailpern, J.; Guarino-Reid, L.; Boardman, R.; Annam, S. web 2.0: blind to an accessible new world. In *Proceedings of the 18th international conference on world wide web (WWW '09)*. ACM, New York, NY, USA, pp. 821-830.
- (Hanson e Richards, 2013) Hanson, V.L.; Richards, J. T. *Progress on website accessibility?* *ACM Trans. web*, v. 7, n. 1, p. 2:1–2:30, 2013.
- (ISO 12207, 1998) ISO/IEC 12207 *Standard for Information Technology –Software Lifecycle Processes*. International Standard Organization, NY, 1998.
- (ISO 40500, 2012) ISO/IEC. 40500:2012 - *Information technology. W3C web Content Accessibility Guidelines (WCAG) 2.0*, 2012. Disponível em: [http://www.iso.org/iso/iso\\_catalogue/catalogue\\_tc/catalogue\\_detail.htm?csnumber=58625](http://www.iso.org/iso/iso_catalogue/catalogue_tc/catalogue_detail.htm?csnumber=58625). Acesso em: janeiro de 2014.
- (ISO 9241-11, 1998) ISO 9241-11 - *Ergonomic requirements for office work with visual display terminals (vdts) - part 11: guidance on usability*. [S.l.], International Standards Organization, NY, 1998.
- (ISO 9241-171, 2008) ISO 9241-171 - *Ergonomics of human system interaction*. Part 171: Guidance on software accessibility. [S.l.], International Standards Organization, NY, 2008.



(ISO 9241-210, 2010) ISO 9241-210 *Ergonomics of human-system interaction*. Part 210: Human-centered design for interactive systems. International Standard Organization, NY, 2010.

(ISO/IEC 9126, 2001) *International Organization for Standardization, "ISO Standard 9126: Software Engineering – Product Quality, parts 1, 2 and 3"*, International Organization for Standardization, Geneva, 2001 (part 1), 2003 (parts 2 and 3).

(Karat, 1993) Karat, J. *The cost-benefit and business case analysis of usability engineering*. InterChi '93, Amsterdam, Tutorial Notes 23, 1993.

(Kelly *et al.*, 2005) Kelly, B.; Sloan, D.; Phipps, L.; Petrie, H.; Hamilton, F. Forcing standardization or accommodating diversity?: a framework for applying the WCAG in the real world. Proceedings of the 2005 International Cross-Disciplinary Workshop on Web Accessibility (W4A). Chiba, Japan: ACM. 2005. pp. 46-54.

(Kelly *et al.*, 2007) Kelly, B., Sloan, D., Brown, S., Seale, J., Petrie, H., Lauke, P., Ball, S. *Accessibility 2.0: people, policies and processes*. In: W4A 2007: Proc. of the 2007 international cross-disciplinary conference on web accessibility (W4A), pp. 138–147. ACM, New York, 2007.

(Kirakowski, 2002) Kirakowski, J. 2002. Questionnaires in Usability Engineering: A list of frequently asked questions. Disponível em: <http://www.ucc.ie/hfrg/resources/qfaq1.html>. Acesso em: outubro de 2014.

(Kitchenham e Charters, 2007) Kitchenham, B.; Charters, S. *Guidelines for performing systematic literature reviews in software engineering*. Technical Report EBSE-2007-001, Keele University and Durham University Joint Report, 2007.

(Kitchenham, 2004) Kitchenham, B. *Procedures for performing systematic reviews*. Joint Technical Report Software Engineering Group, Department of Computer Science Keele University, United King and Empirical Software Engineering, National ICT Australia Ltd, 2004.

(Lai *et al.*, 2010) Lai, C.; Chen, P.; Shih, S.; Liu, Y.; Hong, J. *Computational models and experimental investigations of effects of balance and symmetry on the aesthetics of text-overlaid images*. International Journal of Human-Computer Studies, pp. 41-56, 2010.

(Lara, 2012) Lara, S. M. A. de. Mecanismos de apoio para usabilidade e acessibilidade na interação de adultos mais velhos na web. 2012. PhD thesis, ICMC-USP, Brazil, 2012.

(Lazar e Olalere, 2011) Lazar, J.; Olalere, A. *Investigation of best practices for maintaining section 508 Compliance in U.S. federal web sites*. In Proceedings of the 6th international conference on Universal access in human-computer interaction: design for all and eInclusion - Volume Part I (UAHCI'11), Constantine Stephanidis (Ed.). Springer, Berlin, Heidelberg, 498-506.

- (Lazar, Dudley-Sponaugle e Greenidge, 2004) Lazar, J., Dudley-Sponaugle, A., Greenidge, K. *Improving Web Accessibility: A Study of Webmaster Perceptions*. Computers and Human Behavior, 20(2), pp. 269-288, 2004.
- (López *et al.*, 2012) López, J. M.; Pascual, A.; Mendiña, C.; Granollers, T. *Methodology for identifying and solving accessibility related issues in web Content Management system environments*. In: W4A2012 Tech. Paper, April 16-17, 2012. Lyon, France.
- (López, 2010) López, M. *AWA, Marco metodológico específico en el dominio de la accesibilidad para el desarrollo de aplicaciones web*. Tese de Doutorado. Universidad Carlos III de Madrid, Madrid, 395 p., 2010.
- (Lucca *et al.*, 2005) Lucca, G. A. D.; Fasolino, A. R.; Tramontana, P. Web site accessibility: Identifying and fixing accessibility problems in client page code. In: WSE '05: Proceedings of the Seventh IEEE International Symposium on Web Site Evolution, Washington, DC, USA: IEEE Computer Society, pp. 71–78, 2005.
- (Luna *et al.*, 2010) Luna, E. R.; Garrigós, I.; Grigera, J.; Winckler, M. *Capture and evolution of web requirements using webspec*. In Proceedings of the 10th Int. Conference on web engineering (ICWE'10). Springer, Berlin, Heidelberg, pp. 173-188, 2010.
- (Lunn e Harper, 2011) Lunn, D.; Harper, S. *Improving the accessibility of dynamic web content for older users*. In Proceedings of the International Cross-Disciplinary Conference on web Accessibility (W4A '11). ACM, New York, NY, USA, Article 16, 2 pages.
- (Mack e Nielsen, 1994) Mack, R.; Nielsen, J. *Usability Inspection Methods*. New York, NY: John Wiley & Sons, 1994.
- (Maia, 2010) Maia, L. S. Um processo para o desenvolvimento de aplicações web acessíveis. Dissertação de Mestrado. Universidade Federal de Mato Grosso do Sul, Campo Grande, 106 p., 2010.
- (Masuwa-Morgan, 2008) Masuwa-Morgan, K.R. *Introducing AccessOnto: Ontology for Accessibility Requirements Specification*. Ontologies in Interactive Systems. First International Workshop, pp. 33-38, 2008.
- (Matheson, 2006) Matheson, F. *Designing for a moving target*. In: Proceedings of the 4th Nordic conference on Human-computer interaction: changing roles, New York, NY, USA: ACM, 2006, p. 495–496 (NordiCHI '06, v.1).
- (Mbipom e Harper, 2009) Mbipom, G.; Harper, S. *The transition from web content accessibility diretrizs 1.0 to 2.0: what this means for evaluation and repair*. Proceedings of the 27th ACM international conference on Design of communication. Bloomington, Indiana, USA: ACM Press. pp. 37-44, 2009.

(Medeiros, 1999) Medeiros, M. A. ISO 9241: uma proposta de utilização da norma para avaliação do grau de satisfação de usuários de software. Dissertação (Mestrado em Engenharia de Produção) - Programa de Pós-Graduação em Engenharia de Produção - Universidade Federal de Santa Catarina, Florianópolis, 1999.

(Melo e Baranaukas, 2006) Melo, A. M.; Baranaukas, M. C. C. Design para a inclusão: desafios e proposta. Proceedings of VII Brazilian symposium on Human factors in computing systems (IHC '06). Natal: ACM, New York, NY, USA. 2006. pp. 11-20.

(Mendes, Mosley e Counsell, 2004) Mendes, E., Mosley, N., Counsell, S. *Investigating web size metrics for early web cost estimation*. The Journal of System and Software, In Press, Corrected Proof, Available online 22, October 2004.

(Milne *et al.*, 2005) Milne, S.; Dickinson, A.; Carmichael, A.; Sloan, D.; Eisma, R.; Gregor, P. *Are diretrizs enough? An introduction to designing web sites accessible to older people*. In: (ed. A. Birman) IBM Systems Journal Vol. 44, no. 3, pp. 557-572, 2005.

(Molina e Toval, 2009) Molina,F.; Toval, A. *Integrating usability requirements that can be evaluated in design time into model driven engineering of web information systems*. Adv. Eng. Software, v. 40, n. 12, pp. 1306-1317, 2009.

(Montero *et al.*, 2002) Montero, F.; Lozano, M.; Gonzalez, P. e Ramos, I. *A first approach for design web sites by using patterns*. First Nordic Conference on Pattern Languages of Program. Hojstrupgard, Dinamarca: 20 a 22 de setembro de 2002. 2002. pp. 137-158.

(Munson e Pimentel, 2008) Munson, E.V.; Pimentel, M. D. G. *Specialized documents*. In: Web Accessibility, Springer London, 2008, p. 274–285 (Human-Computer Interaction Series, v.4).

(Natukunda, 2008) Natukunda C. *Have PWDs been left out of digital age?*. New Vision Printing and Publishing Corporation, Uganda The New Vision Newspaper, Monday, February 4, 2008.

(Navarro *et al.*, 2008) Navarro, A.; Fernández-Valmayou,A.; Fernández-Manjón, B.; Sierra, J. L. *Characterizing navigation maps for web applications with the NMM approach*. Sci. Comput. Program., v. 71, n. 1, p. 1–16, 2008.

(Nielsen e Molich, 1990) Nielsen, J.; Molich, R. *Heuristic evaluation of user interfaces*. Proc. ACM CHI'90 Conf., Seattle, EUA, 1-5 abril, p. 249-256, 1990.

(Nielsen, 1993) Nielsen, J. *Usability engineering*. Boston, MA: Academic Press, 1993. 362p.

(Nietzio *et al.*, 2010) Nietzio, A.; Olsen, M. G.; Eibegger, M.; Snaprud, M. Accessibility of eGovernment web sites: towards a collaborative retrofitting approach. Proceedings of the 12th international conference on Computers helping people with special needs: Part I (ICCHP'10). Berlin, Heidelberg: Springer-Verlag. 2010. p. 468-475.

- (Norman, 1986) Norman, D. A. *Cognitive engineering*. User centered system design: New perspectives on human-computer interaction, 1986.
- (Norman, 1999) Norman, D. A. *The Invisible Computer: Why Good Products Can Fail*. The Personal Computer Is So Complex, and Information Appliances Are the Solution, 1999.
- (OMG, 2005) OMG. Software Process Engineering Metamodel - Version 1.1. Object Management Group, 2005. Disponível em: <http://www.omg.org/technology/documents/formal/spem.htm>. Acesso em: outubro de 2010.
- (Parmanto e Zeng, 2005) Parmanto, B.; Zeng, X. *Metric for web accessibility evaluation*. J. Am. Soc. Inf. Sci. Technol., v. 56, n. 13, pp. 1394-1404, 2005.
- (Paula Filho, 2009) Paula Filho, W. P. Engenharia de software: fundamentos, Métodos e Padrões. 3ª. Ed. Rio de Janeiro, RJ: LTC, xiii, 1248 p., 2009.
- (Petrie e Kheir, 2007) Petrie, H.; Kheir, O. *The relationship between accessibility and usability of websites*. In Proceedings of the SIGCHI conference on Human factors in computing systems (CHI '07). ACM, New York, NY, USA, pp. 397-406, 2007.
- (Power et al., 2012) Power, C.; Freire, A. P.; Petrie, H. e Swallow, D. *Guidelines are Only Half of the Story: Accessibility Problems Encountered by Blind Users on the web*. In: CHI'12, May 5-10, 2012, Austin, Texas, USA.
- (Prates e Barbosa, 2003) Prates, R.O.; Barbosa, S.D.J. (2003) Avaliação de Interfaces de Usuário - Conceitos e Métodos Anais do XXIII Congresso Nacional da Sociedade Brasileira de Computação. XXII Jornadas de Atualização em Informática (JAI). SBC 2003. Agosto de 2003.
- (Preece, Rogers e Sharp, 2002) Preece, J.; Rogers, Y.; Sharp, E. (2002) *Interaction Design: Beyond Human-computer Interaction*. New York, NY: John Wiley & Sons. 2002.
- (Pressman e Lowe, 2009) Pressman, R. S.; Lowe, D. Engenharia web. São Paulo: LTC, 426 p., 2009.
- (Prümper, 1993) *Software-evaluation based upon ISO 9241 Part 10*. In Grechenig, T., & Tscheligi, M. (Eds.), Human-computer interaction, 255-265. Berlin, Germany: Springer-Verlag.
- (QUIS, 2014) QUIS. *Questionnaire For User Interaction Satisfaction*. Disponível em: <http://lap.umd.edu/quis/>. Acesso em: outubro de 2014.
- (Rocha e Baranauskas, 2000) Rocha, H. V.; Baranauskas, M. C. C. Design e avaliação de interfaces humano-computador. São Paulo: IMW-USP, 2000. 242p.

(Romen e Svanaes, 2008) Romem, D.; Svanaes, D. *Evaluating web site accessibility: validating the WAI diretrizs through usability testing with disabled users*. In: NordiCHI '08: Proceedings of the 5th Nordic conference on Human-computer interaction, New York, NY, USA: ACM, 2008, p. 535-538.

(Rosa e Veras, 2013) Rosa, J. M.; Veras, M. Avaliação heurística de usabilidade em jornais online: Estudo de Caso em dois sites. *Perspectivas em Ciência da Informação*. 2013, vol.18, n.1, pp. 138-157.

(Rossi, Schwabe e Lyardet, 1999) Rossi, G.; Schwabe, D.; Lyardet, F. *Improving web information systems with navigational patterns*. Toronto, Canada: Elsevier North-Holland. 1999. p. 1667-1678.

(Rubegni *et al.*, 2008) Rubegni, E.; Paolini, P.; Terragni, A.; Vaghi, S. *Accessibility for Blind Users: An Innovative Framework*. ICCHP 2008, LNCS 5105, Springer-Verlag Berlin Heidelberg, pp. 250-257, 2008.

(Sanchez, Oliveira e Fortes, 2014) Sanchez, R. V. V.; Oliveira, R. R.; Fortes, R. P. M. RestML: Modeling RESTful Web Services. In: Pautasso, Cesare, Wilde, Erik, Alarcon, Rosa. (Org.). *REST: Advanced Research Topics and Practical Applications*. 1ed. New York, USA: Springer New York, 2014, v. 1, p. 125-143.

(Santos e Fortes, 2010) Santos, E. P. B.; Fortes, R. P. M. *CMS instance of a tool to support the scheduling of undergraduate final year project on the web*. In: Simpósio Brasileiro de Sistemas Multimídia e web - webMedia, 2010, Belo Horizonte MG. Anais do webMedia 2010. Porto Alegre RS: SBC, 2010, 127-129.

(Santos *et al.*, 2011) Santos, E. P. B.; Lara, S. M. A.; Watanabe, W. M.; A. Filho, M. C.; Fortes, R. P. M. *Usability evaluation of horizontal navigation bar with drop-down menus by middle aged adults*. In Proceedings of the 29th ACM international conference on Design of communication (SIGDOC '11). ACM, New York, NY, USA, 145-150.

(Santos, Gassenferth e Machado, 2008) Santos, R. C., Gassenferth, W., Machado, M. A. S. *Systems usability evaluation metrics review*. Global Business and Technology Association Conference - GBATA, Systems Usability Evaluation Metrics Review, Madrid, Espanha, 2008.

(Scapin e Bastien, 1993) Bastien, J. M. C.; Scapin, D. L. *Ergonomic criteria for the evaluation of human-computer interfaces*. (Technical report No 156), INRIA Rocquencourt, France, 1993.

(Schwabe, Almeida Pontes e Moura, 1999) Schwabe, D.; Almeida Pontes, R.; Moura, I. *OOHDM-Web: an environment for implementation of hypermedia applications in the WWW*. SIGWEB Newsl, v. 8, n. 2, p. 18-34, 1999.

(Shneiderman, 1988) Shneiderman, B. *Designing the User Interface*. Addison-Wesley, Reading, MA. 1998.

(Shneiderman, 1993) Shneiderman, B. *Designing the User Interface: Strategies for Effective Human-Computer Interaction*. Massachusetts, Addison-Wesley Publishing Company, 1993.

(Shull, Carver e Travassos, 2001) Shull, F., Carver, J., Travassos, G. H. *An empirical methodology for introducing software processes*. In 9th ACM SIGSOFT international symposium on Foundations of software engineering (2001), 288-296.

(Sommerville, 2011) Sommerville, I. *Engenharia de Software*, 9. Ed.; São Paulo: Pearson Prentice Hall, 2011.

(Standing, 2002) Standing, C. *Methodologies for developing web applications*. Information and Software Technology, Vol. 44, Issue 3, 2002.

(Sullivan e Matson, 2000) Sullivan, T.; Matson, R. *Barriers to use: usability and content accessibility on the web's most popular sites*. In: Proceedings on the 2000 conference on Universal Usability (CUU '00, v.1), New York, NY, USA: ACM, 2000, p. 139-144.

(SUMI, 2012) SUMI. What is SUMI? Home Page do Questionário. Disponível em: <<http://sumi.ucc.ie/whatis.html>>. Acesso em: outubro de 2014.

(Sun e Zhang, 2009) Sun, Z.; Zhang, J. On Accessibility of Concept, Principle and Model of Educational web Sites Design. In Proceedings of the 2009 International Conference on New Trends in Information and Service Science (NISS '09). IEEE Computer Society, Washington, DC, USA, pp. 730-733, 2009.

(Takagi et al., 2004) Takagi, H.; Asakawa, C.; Fukuda, K.; Maeda, J. Accessibility Designer: Visualizing Usability for the Blind. Proceedings of the Sixth International ACM SIGACCESS Conference on Assistive Technologies, ACM Press, 2004.

(Tanaka e Rocha, 2011) Tanaka, E.H. e Rocha, H.V. Evaluation of web Accessibility Tools. In Proc. IHC&CLIHC'11, ACM (2011), 272-279.

(Thew et al., 2009) Thew, S.; Sutcliffe, A.; Bruijn, O.; McNaught, J.; Protector, R; Venters, C.; Buchan, I. Experience in e-Science Requirements Engineering. IEEE Computer Society, Los Alamitos, CA, USA, pp. 80-87, 2009.

(Tidwell, 2009) Tidwell, J. Common Ground: A Pattern Language for Human-Computer Interface Design. 2009. Disponível em: [http://www.mit.edu/~jtidwell/common\\_ground\\_onefile.html](http://www.mit.edu/~jtidwell/common_ground_onefile.html). Acessado em: Maio de 2013.

(Umbach, 2006) Umbach, J. M. *web 2.0 - the new commons*. Feliciter. Canadian Library Association. HighBeam Research., 2006. Disponível em: <http://www.highbeam.com/doc/1P3-1156039661.html>. Acesso em: março de 2014.

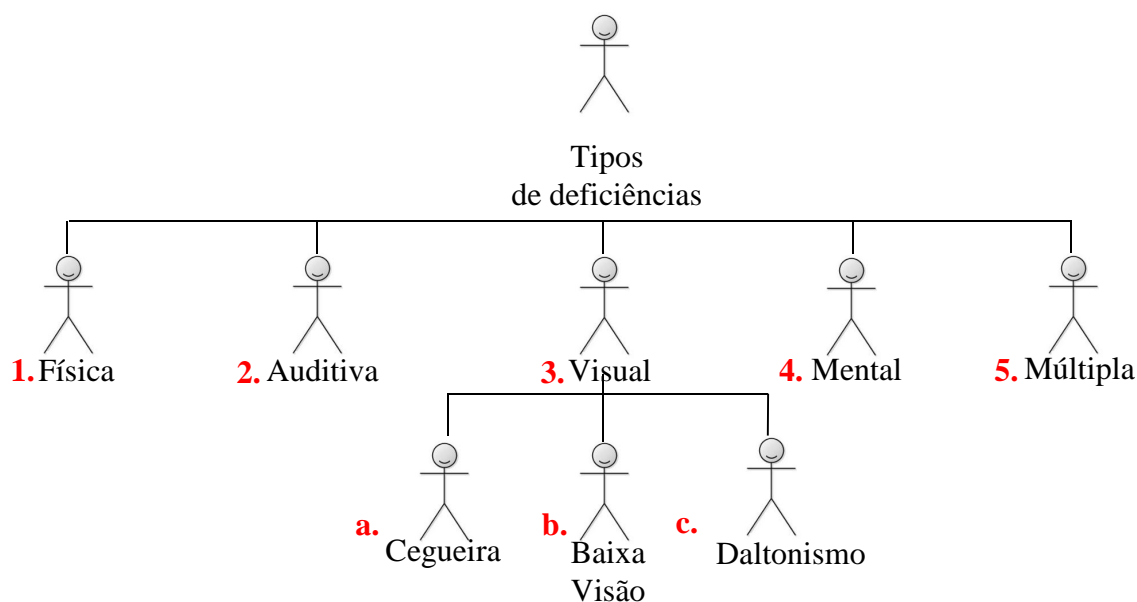
(Usability.gov, 2014) Usability.gov. Disponível em: <http://www.usability.gov/>. Acesso em: janeiro de 2014.

- (Vigo *et al.*, 2007) Vigo, M.; Arrue, M.; Brajnik, G.; Lomuscio, R.; Abascal, J. *Quantitative metrics for measuring web accessibility*. In: W4A '07: Proceedings of the 2007 international crossdisciplinary conference on web accessibility (W4A), New York, NY, USA: ACM, 2007, p. 99-107.
- (W3C, 1999a) W3C. *Web Content Accessibility Guidelines 1.0*, 1999. Disponível em <http://www.w3.org/TR/WCAG10/>. Acesso em: janeiro de 2014.
- (W3C, 1999b) W3C. *HTML 4.01 Specification*, 1999. Disponível em: <http://www.w3.org/TR/html4/>. Acesso em: novembro de 2010.
- (W3C, 2008a) W3C. *Conformance evaluation of web sites for accessibility*. 2008. Disponível em: <http://www.w3.org/WAI/eval/conformance.html>. Acesso em: janeiro de 2014.
- (W3C, 2008b) W3C. *Web content accessibility diretrizs (WCAG) 2.0*. W3C Recommendation, 2008. Disponível em: <http://www.w3.org/TR/WCAG20/>. Acesso em: março de 2014.
- (W3C, 2010) W3C *Involving users in web accessibility evaluation*. 2010. Disponível em: <http://www.w3.org/WAI/eval/users.html>. Acesso em: março de 2014.
- (W3C, 2013) W3C *Preliminary review of web sites for accessibility*. 2013. Disponível em: <http://www.w3.org/WAI/eval/preliminary.html>. Acesso em: março de 2014.
- (WAMMI, 2013) Website Analysis and MeasurMent Inventory, 2013. Disponível em: <http://www.wammi.com>. Acesso em: outubro de 2014.
- (Wang e Zahadat, 2009) Wang, Y. D.; Zahadat, N. *Teaching web development in the web 2.0 era*. In: Proceedings of the 10th ACM conference on SIG-information technology education, New York, NY, USA: ACM, 2009, p. 80–86 (SIGITE '09, v.1).
- (Watanabe, 2007) Watanabe, T. *Experimental evaluation of usability and accessibility of heading elements*. In: W4A '07: Proceedings of the 2007 international cross-disciplinary conference on web accessibility (W4A), New York, NY, USA: ACM, 2007, p. 157-164.
- (Watanabe, Fortes e Dias, 2012) Watanabe, W.M.; Fortes, R.P.M.; Dias, A. L. *Using acceptance tests to validate accessibility requirements in RIA*. In: Proceedings of the International Cross-Disciplinary Conference on web Accessibility (W4A '12, v.1), New York, NY, USA: ACM, 2012, p. 15:1–15:10.





## *Tipos de deficiências*



### **1. Deficiência Física**

É a alteração completa ou parcial de um ou mais segmentos do corpo humano, acarretando o comprometimento da função física, apresentando-se sob a forma de paraplegia, paraparesia, monoplegia, monoparesia, tetraplegia, tetraparesia, triplegia, triparesia, hemiplegia, hemiparesia, ostomia, amputação ou ausência de membro, paralisia cerebral, nanismo, membros com deformidade congênita ou adquirida, exceto as deformidades estéticas e as que não produzam dificuldades para o desempenho de funções (Decreto nº 5.296/04, art. 5º, §1º, I, “a”, c/c Decreto nº 3.298/99, art. 4º, I).

As Deficiências Físicas podem ser classificadas em dois tipos: lesões traumáticas (alguns exemplos: amputação, paraplegia, tetraplegia etc.) e doenças congênitas e

condições análogas (alguns exemplos: paralisia cerebral, distrofia muscular, artrite, doença de Parkinson etc.). O conceito de deficiência inclui a incapacidade relativa, parcial ou total para o desempenho da atividade dentro do padrão considerado normal para o ser humano. O usuário com deficiência pode desenvolver atividades laborais desde que tenha condições e apoios adequados às suas características.

- Dificuldades:

Na computação, os usuários deste perfil sentem várias dificuldades na utilização do mouse e também do teclado comum, além disso, podem ter dificuldades em posicionar o mouse em pequenos elementos ou *links* devido à falta de coordenação. A maioria dos recursos de tecnologia assistiva para pessoas com deficiência motora usam o teclado para emular as funcionalidades do teclado comum. Desta maneira, é necessário garantir que os conteúdos estejam acessíveis por meio do teclado e assegurar que o sistema seja navegável com o uso de poucas teclas.

## 2. Deficiência Auditiva

É a perda bilateral, parcial ou total, de 41 decibéis (dB) ou mais, aferida por audiograma nas frequências de 500hz, 1.000Hz, 2.000Hz e 3.000Hz (Decreto nº 5.296/04, art. 5º, §1º, I, “b”, c/c Decreto nº 5.298/99, art. 4º, II). Embora existam pessoas que são completamente surdas, há também pessoas com diversos graus de perda auditiva funcional. Graus de perda auditiva muitas vezes são classificados como leve (A incapacidade de ouvir sons abaixo de 30 dB. Ex: Discursos podem ser de difícil audição especialmente se estiverem presentes ruídos de fundo), moderada (a incapacidade de ouvir sons abaixo de cerca de 50 dB. Ex: Aparelho ou prótese auditiva pode ser necessário), grave ou severa (A incapacidade de ouvir sons abaixo de cerca de 80 dB. Ex: Próteses auditivas são úteis em alguns casos, mas são insuficientes em outros. Alguns indivíduos com perda auditiva severa se comunicam principalmente por meio de linguagem gestual, outros usam técnicas de leitura labial) e profunda (A ausência da capacidade de ouvir, ou a incapacidade de ouvir sons abaixo de cerca de 95 dB. Ex: Alguns indivíduos com perda auditiva profunda se comunicam principalmente por meio de linguagem gestual, outros com uso das técnicas de leitura labial).

Habitualmente, aqueles que se dizem surdos têm a perda auditiva severa ou profunda. Aquelles com menor grau de perda auditiva são comumente referidos como audição difícil.

- Dificuldades:

Os usuários de sistemas computacionais deste perfil sentem dificuldades em acessar (perceber/ouvir) determinadas informações que se encontram disponíveis por meio de dispositivos de áudio. Para fazer conteúdo de áudio da web acessível aos surdos, alguns desenvolvedores pensaram que o melhor meio seria fazer uma versão da linguagem gestual para o conteúdo de áudio. Mas existem alguns problemas com esta opção: (I) Nem todas as pessoas surdas utilizam as línguas de sinais; (II) A imagem do Vídeo na web muitas vezes não é suficientemente grande ou com imagens claras para tornar a linguagem escrita compreensível; (III) Nem todo mundo fala a mesma linguagem gestual. Dessa maneira, a forma de tornar acessíveis os conteúdos de áudio para os que são surdos é fornecer legendas ou transcrições.

### **3. Deficiência Visual**

De acordo com o Decreto nº 3.298/99 e o Decreto nº 5.296/04, conceitua-se como deficiência visual: Cegueira (na qual a acuidade visual é igual ou menor que 0,05 no melhor olho, com a melhor correção óptica); Baixa Visão (significa acuidade visual entre 0,3 e 0,05 no melhor olho, com a melhor correção óptica. Ex: Glaucoma, Catarata etc.); Daltonismo (os casos nos quais a somatória da medida do campo visual em ambos os olhos for igual ou menor que 60°); ou a ocorrência simultânea de quaisquer das condições anteriores.

#### **a. Cegueira**

Usuários cegos podem acessar a web usando Leitores de Tela e teclados, podendo ainda encontrar alguns obstáculos ao tentar acessar conteúdos da web. Um sistema web acessível significa liberdade e independência para alguém que é cego. Quando desenvolvedores da web não fornecem conteúdos acessíveis, a única maneira que uma pessoa cega pode obter as informações a partir do sistema web é por meio de outra pessoa.

- Dificuldades:

Alguns dos desafios deste tipo de usuário podem ser citados: (I) os usuários não podem ver (perceber) as informações visuais (ex: imagens, leiaute, ou informações coloridas); (II) eles normalmente precisam de um teclado para navegar no conteúdo disponível no sistema web; (III) eles não podem compreender o conteúdo que é apresentado em ordem lógica linear ou que contenha textos dispensáveis (ex: longos endereços da web, palavra por palavra etc.); e (IV) eles são ajudados por técnicas que nem sempre são capazes de acessar um vasto leque de tecnologias, especialmente se essas tecnologias são novas.

## **b. Baixa Visão**

As pessoas com baixa visão são aquelas que, mesmo usando óculos comum, lentes de contato ou implantes de lentes intraoculares, não conseguem ter uma visão nítida.

- Dificuldades:

Elas podem ter sensibilidade ao contraste, percepção das cores e intolerância à luminosidade, dependendo da patologia causadora da perda visual. O usuário não consegue ver (perceber) o conteúdo disponível no sistema web por vários motivos, entre eles: o conteúdo é muito pequeno, não pode ser ampliado, ou não têm suficiente contraste.

A tecnologia mais comum que as pessoas com baixa visão podem usar são os ampliadores de tela. No entanto, para que o texto se torne mais legível quando ampliado, é necessário usar textos reais ao invés de texto em gráficos ou imagens. Além disso, sistemas web com baixo contraste podem ser difíceis de ler para pessoas com baixa visão. Então, o desenvolvedor deve aumentar o contraste dos sistemas web, incluindo gráficos, fontes e origens. A regra geral para desenvolver quando os usuários têm baixa visão é fazer com que tudo possa ser configurável (Ex: no caso de texto, os usuários devem poder aumentá-lo, mudar sua cor, mudar a cor de fundo da página, etc. Se o documento está em porcentagens, a tela pode ser ampliada ou diminuída de acordo com as necessidades do usuário).

## **c. Daltonismo**

As cores com que as pessoas daltônicas têm dificuldade em distinguir dependem de seu tipo de daltonismo, mas a deficiência para as cores vermelha e verde é a mais comum.

- Dificuldades:

Ao conceber o conteúdo da web para as pessoas que possuem daltonismo, o desenvolvedor não precisa converter todas as suas imagens para preto e branco ou se livrar de todas as imagens, mas sim, certificar-se de que as cores não são únicas no método de transmissão para informações importantes. Os usuários não veem (percebem) a diferença entre as combinações de certas cores.

#### **4. Deficiência Mental**

Conceitua-se como deficiência mental o funcionamento intelectual significativamente inferior à média, com manifestação antes dos 18 anos e limitações associadas a duas ou mais áreas de habilidades adaptativas, tais como: comunicação; cuidado pessoal; habilidades sociais; utilização dos recursos da comunidade; saúde e segurança; habilidades acadêmicas; lazer; e trabalho (Decreto nº 5.296/04, art. 5º, §1º, I, “d”; e Decreto nº 3.298/99, art. 4º, I).

Existem pelo menos duas maneiras de classificar deficiências cognitivas: pela incapacidade funcional (por exemplo: autismo, Síndrome de Down, traumatismos crânio-encefálico - TCE, até a demência) ou por incapacidade clínica (déficit de atenção e desordem - ADD, dislexia - dificuldade em leitura, discalculia - dificuldade com a matemática, e dificuldades de aprendizagem em geral). Algumas das principais categorias dos déficits funcionais e cognitivos incluem deficiência ou dificuldades de: Memória; Resolução de problemas; Atenção; Compreensão Verbal, de Leitura e Linguística; Compreensão Matemática; e Compreensão Visual.

- Dificuldades:

Alguns conteúdos podem ser muito complexos para determinados públicos-alvo. Isso é inevitável. No entanto, ainda existem algumas coisas que podem ser feitas pelos desenvolvedores para aumentar a acessibilidade ao conteúdo da web a pessoas com deficiências cognitivas menos graves.

Os usuários não podem compreender o conteúdo que é apresentado da mesma maneira para usuários sem deficiência. Eles precisam de ajuda para reter o aprendizado, para não usarem o sistema erroneamente etc. Em muitos casos, a maneira de tornar o conteúdo web mais acessíveis às pessoas com deficiências cognitivas não é nada mais do que técnicas para uma comunicação eficaz.

## 5. Deficiência Múltipla

De acordo com o Decreto nº 3.298/99, conceitua-se como deficiência múltipla a associação de duas ou mais deficiências (mental, visual, auditiva e física). Neste caso, a interação do usuário com o sistema poderá ser feita de várias maneiras (citadas anteriormente) dependendo das deficiências do usuário.

### Referências consultadas:

Akhter, F.; Buzzi, M. C.; Buzzi, M.; Leporini, B. Conceptual Framework: How to Engineer Online Trust for Disabled Users. *Web Intelligence and Intelligent Agent Technologies*, 2009. WI-IAT '09. IEEE/WIC/ACM International Joint Conferences on. Vol. 3, pp. 614-617, 2009.

Bailey, C., & Pearson, E. *A Web Accessibility Auditing Instrument to Assist Novice Evaluators*. World Conference on E-Learning in Corporate, Government, Healthcare, and Higher Education (E-LEARN). Vancouver, Canada, 2009.

Bailey, C., Pearson, E. *Development and Trial of an Educational Tool to Support the Accessibility Evaluation Process*. International Cross Disciplinary Conference on web Accessibility (W4A). Hyderabad, India, 2011.

Christiernin, L.G. *Guiding the designer: A radar diagram process for applications with multiple layers*. *Interacting with Computers*, 22(2), 107-122, 2010.

Decreto nº 3.298. Política Nacional para a Integração da Pessoa Portadora de Deficiência. Disponível em: [http://www.planalto.gov.br/ccivil\\_03/decreto/d3298.htm](http://www.planalto.gov.br/ccivil_03/decreto/d3298.htm). 20 de Dezembro de 1999, Brasil.

eMAG. Recomendações de acessibilidade para a construção e adaptação de conteúdos do governo brasileiro na Internet. eMAG, Modelo de acessibilidade do Governo Eletrônico, Versão 2.0, 2005. Disponível em: <http://www.mp.mt.gov.br/storage/webdisco/2009/10/06/outros/154c3af281354a82f7c59aca9ffbf87f.pdf>. Acesso em: agosto de 2010.

Gregg, D. *Developing a collective intelligence application for special education*. *Decision Support Systems*, Vol. 47, No. 4, pp. 455-465, 2009.

Sun, Z.; Zhang, J. *On Accessibility of Concept, Principle and Model of Educational Web Sites Design*. *New Trends in Information and Service Science*, 2009. NISS '09. International Conference on, 2009.

WCAG 1.0. *W3C User agent accessibility diretrizs 1.0*. W3C Recommendation, 2002. Disponível em: <http://www.w3.org/TR/WCAG10/>. Acesso em: janeiro de 2010.

WCAG 2.0. *W3C Web content accessibility diretrizs 2.0*. W3C Recommendation, 2008. Disponível em: <http://www.w3.org/TR/WCAG20/>. Acesso em: janeiro de 2010.

---

# *Requisitos de usabilidade e acessibilidade*

---

## **Possíveis requisitos de usabilidade**

Para guiar o desenvolvimento de aplicações visando usabilidade, Nielsen e Molich (1990) definiram dez heurísticas que trata de alguns dos principais pontos que devem ser verificados durante o desenvolvimento de um sistema de software:

1. Visibilidade do status do sistema: o sistema precisa manter os usuários informados sobre o que está acontecendo, fornecendo um *feedback* adequado dentro de um tempo razoável;
2. Compatibilidade do sistema com o mundo real: o sistema precisa falar a linguagem do usuário, com palavras, frases e conceitos familiares ao usuário, ao invés de termos orientados ao sistema. O sistema deve seguir convenções do mundo real, fazendo com que a informação apareça numa ordem natural e lógica;
3. Controle do usuário e liberdade: usuários frequentemente escolhem funções do sistema por engano e precisam ter claras as saídas de emergência, para que possa sair do estado indesejado sem ter que percorrer um extenso diálogo. Para tanto, é necessário prover funções de *undo* (desfaz) e *redo* (faz de novo);

4. Consistência e padrões: usuários não precisam adivinhar que diferentes palavras, situações ou ações significam a mesma coisa. Portanto, é necessário seguir convenções da plataforma computacional;
5. Prevenção de erros: melhor do que uma boa mensagem de erro é um design cuidadoso, que previne o erro antes dele acontecer;
6. Reconhecimento ao invés de relembração: tornar objetos, ações e opções visíveis. O usuário não deve ter que lembrar informações de uma parte para outra de um diálogo; ao contrário, as instruções para uso do sistema devem estar visíveis e facilmente recuperáveis quando necessário;
7. Flexibilidade e eficiência de uso: usuários novatos se tornam peritos com o uso. Prover aceleradores de forma a aumentar a velocidade de interação, assim como permitir que usuários experientes possam “cortar caminho” em ações frequentes é um aspecto importante para usabilidade;
8. Estética e Design Minimalista: diálogos não devem conter informação irrelevante ou raramente necessária. Qualquer unidade de informação extra no diálogo irá competir com unidades relevantes de informação e diminuir sua visibilidade relativa;
9. Ajudar os usuários a reconhecer, diagnosticar e corrigir erros: mensagens de erro devem ser expressas em linguagem clara (sem códigos) indicando precisamente o problema e sugerindo uma solução construtivamente;
10. Help e Documentação: Embora o ideal seja um sistema que possa ser usado sem auxílio nenhum, é necessário prover a opção de *help* e documentação. Essas informações devem ser de fácil localização, focalizadas na tarefa do usuário e não muito extensas.

### **Possíveis requisitos de acessibilidade**

O WCAG 1.0 (*Web Content Accessibility Guidelines*) (W3C, 1999) contém diretrizes para acessibilidade do conteúdo na web, é uma das principais referências mundial em termos de acessibilidade na web.

Essas diretrizes incluem diversos direcionamentos para serem utilizados durante o desenvolvimento de sistemas web, como prover texto alternativo para conteúdo



audiovisual, prover marcação semântica, utilizar mecanismos de navegação, entre outros. Basicamente, as diretrizes são relacionadas a dois temas principais:

- Garantir a transformação harmoniosa de páginas: páginas que garantam uma transformação harmoniosa permanecem acessíveis independentemente das restrições do usuário (por exemplo: restrições de percepção visual, auditiva ou motora) ou restrições tecnológicas (por exemplo: dispositivos com restrições de exibição ou restrição de velocidade de conexão).
- Facilitar a compreensão do conteúdo e a navegação através dele: o desenvolvedor deve utilizar uma linguagem simples e clara, e também prover mecanismos para facilitar a navegação dentro de uma página e a navegação entre as páginas que compõem o conteúdo.

O documento contendo as WCAG 1.0 é composto por um conjunto de 14 diretrizes de acessibilidade, conforme descritas a seguir:

1. Fornecer alternativas para conteúdo sonoro e visual: esta diretriz indica a necessidade de disponibilizar conteúdo que, ao ser apresentado ao usuário, transmita, em essência, as mesmas funções e finalidade que o conteúdo sonoro ou visual.
2. Não recorrer apenas à cor: deve-se garantir a percepção do texto e dos elementos gráficos do documento, mesmo quando são vistos sem cores.
3. Utilizar corretamente marcações e folhas de estilo: esta diretriz indica a necessidade de se utilizar marcação nos documentos com os elementos estruturais adequados e controlar a apresentação por meio de folhas de estilo, em vez de elementos de apresentação e atributos.
4. Indicar claramente qual o idioma utilizado: utilizar marcações que facilitem a pronúncia e a interpretação de abreviaturas ou de texto em língua estrangeira é imprescindível. Deve-se identificar claramente quaisquer mudanças de idioma no texto de um documento, bem como nos equivalentes textuais.
5. Criar tabelas passíveis de transformação harmoniosa: deve-se assegurar que as tabelas têm as marcações necessárias para poderem ser transformadas de forma harmoniosa por navegadores acessíveis e outros agentes do usuário.

6. Assegurar que as páginas dotadas de novas tecnologias sejam transformadas harmoniosamente: as páginas devem ser acessíveis mesmo quando as tecnologias mais recentes não estejam disponíveis ou tenham sido desativadas.
7. Assegurar o controle do usuário sobre as alterações temporais do conteúdo: deve-se possibilitar a interrupção momentânea ou definitiva do movimento, intermitência, transcurso ou atualização automática de objetos e páginas.
8. Assegurar a acessibilidade direta em interfaces integradas pelo usuário: a interface com o usuário deve obedecer a princípios de design para a acessibilidade: acesso independente de dispositivos, operacionalidade pelo teclado, emissão automática de voz (verbalização).
9. Projetar páginas considerando a independência de dispositivos: é imprescindível utilizar funções que permitam a ativação de elementos de página por meio de uma grande variedade de dispositivos de entrada.
10. Utilizar soluções provisórias ou de transição: deve-se utilizar soluções de acessibilidade transitórias para que as tecnologias de apoio e os navegadores mais antigos funcionem corretamente.
11. Utilizar tecnologias e recomendações do W3C: deve-se utilizar tecnologias do W3C (de acordo com suas especificações) e seguir as recomendações de acessibilidade. Aconselha-se utilizar sempre as últimas versões das recomendações do W3C, quando possível, evitando, assim, a utilização de tecnologias consideradas obsoletas.
12. Fornecer informações de contexto e orientações: o fornecimento de informações de contexto e orientações de navegação é fundamental para auxiliar os usuários a compreender e utilizar o conteúdo de um sistema.
13. Fornecer mecanismos de navegação claros: deve-se fornecer mecanismos de navegação coerentes e sistematizados de modo a aumentar a probabilidade de uma pessoa encontrar o que procura em um dado sistema.
14. Assegurar a clareza e a simplicidade dos documentos: deve-se assegurar a produção de documentos com linguagem clara e simples para que sejam mais fáceis de compreender.

O WCAG 1.0 foi melhorado e com isso formalizou-se o WCAG 2.0. Os quatro princípios de acessibilidade definidos nessa segunda versão são os seguintes:

- O conteúdo deve ser perceptível: usuários devem ser capazes de perceber a informação sendo apresentada, ou seja, o conteúdo não pode ser invisível a todos os seus sentidos;
- Os componentes de interface com o usuário devem ser operáveis: a interface não pode exigir interações com as quais um usuário não possa realizar;
- O conteúdo e os controles do usuário devem ser fáceis de entender: o conteúdo e as operações não podem ir além do conhecimento do usuário;
- O conteúdo deve ser suficientemente robusto para funcionar com as tecnologias atuais e futuras.

Os quatro princípios de acessibilidade apontados pelo WCAG 2.0 já estavam de certa forma contidos nas recomendações do WCAG 1.0. Contudo, a classificação, como um princípio, reforça a sua importância e redefine a forma da organização da nova versão da recomendação do W3C. Embora as diretrizes referenciem uma grande quantidade de cenários, elas podem não ser suficientes para garantir o acesso à informação por pessoas com todos os tipos, níveis e combinações de deficiências. Essas diretrizes também ajudam no desenvolvimento de conteúdo mais usável por indivíduos de idade avançada e com melhor usabilidade por usuários em geral. Cada diretriz apresenta um determinado número de critérios de sucesso, que descrevem especificamente o que deve ser alcançado para satisfazer o padrão (W3C, 2008b). Todos os critérios de sucesso devem considerar:

- Problemas de acessibilidade que afetam pessoas com necessidades especiais, diferentemente dos problemas de usabilidade que podem ser apresentados a todos os usuários;
- Aspectos testáveis na interface, ou seja, deve ser possível determinar, objetivamente, se a interface satisfaz ou não o critério de sucesso.

As diretrizes do conjunto WCAG 2.0 são apresentadas a seguir, agrupadas nos seus respectivos princípios (W3C, 2008b):

### **1. Perceptível**

- a) Alternativas textuais: disponibilizar alternativas textuais para conteúdos não textuais;

- b) Mídias temporais: disponibilizar alternativas para mídias temporais;
- c) Adaptabilidade: criar conteúdo que possa ser disponibilizado de diferentes maneiras sem perder a informação ou estrutura;
- d) Distintivos: facilitar aos usuários ver e ouvir o conteúdo, apresentando foco ao conteúdo principal sendo disponibilizado.

## 2. Operável

- a) Acessível pelo teclado: tornar todas as funcionalidades acessíveis pelo teclado;
- b) Tempo suficiente: disponibilizar tempo suficiente de leitura e utilização do conteúdo;
- c) Apreensibilidade: não estruturar o sistema com conteúdos que possam causar apreensão nos usuários, como por exemplo: *flash* de frequência superior a três vezes por segundo;
- d) Navegabilidade: disponibilizar meios que auxiliem a navegação do usuário, busca por conteúdos e localização.

## 3. Compreensível

- a) Legível e compreensível: disponibilizar o conteúdo de forma legível e compreensível aos usuários;
- b) Previsibilidade: os sistemas devem aparecer e operar por meios previsíveis;
- c) Assistência de entrada: auxiliar os usuários a evitar e corrigir erros;

## 4. Robusto

- a) Compatibilidade: maximizar a compatibilidade com agentes de usuário e tecnologias disponibilizados atualmente ou no futuro.

### Referências consultadas:

Nielsen, J. Usability Engineering. Academic Press, 1993.

Nielsen, J.; Molich, R. Heuristic evaluation of user interfaces. Proc. ACM CHI'90 Conf., Seattle, EUA, 1-5 abril, p. 249-256, 1990.

W3C. Web Content Accessibility Guidelines 1.0, 1999.

W3C. Web Content Accessibility Guidelines 2.0, 2008.

---

# ***TCLE - Termo de Consentimento Livre e Esclarecido***

---

## **Termo de Consentimento Livre e Esclarecido**

1. Você está sendo convidado para participar da pesquisa “Um método para garantir a acessibilidade no desenvolvimento de sistemas web”.
2. Esta pesquisa tem como intuito observar a interação de usuários com sistemas web para coletar indícios de usabilidade e acessibilidade.
  - a. Os sistemas web selecionados, “Agendamento” (<http://agua.intermedia.icmc.usp.br/agendamento>) e “AgendAloca” (<http://garapa.intermedia.icmc.usp.br:3000/agendaloca>), possuem a mesma finalidade. São sistemas que tem um Calendário em Grupo que visa suprir a necessidade de integração de informações para o uso de alunos e professores da Universidade de São Paulo (USP).
  - b. Você foi selecionado para desenvolver ou o papel de aluno agendando uma apresentação de monografia de fim de curso ou o papel de professor, agendando a presença para avaliar uma ou mais apresentações agendadas pelos alunos.
  - c. Sua participação nesta pesquisa consistirá em interagir com os sistemas web citados como aluno e/ou professor. Após a utilização desses sistemas, você poderá relatar a sua experiência e relatar os pontos positivos e negativos do mesmo. O relato será por meio de questionários e entrevistas.
3. A sua participação neste estudo de caso envolve riscos mínimos (como desconforto moral, ético ou físico) e é voluntária (você não terá nenhum benefício financeiro por participar desse estudo), podendo você recusar-se a participar ou retirar seu consentimento, em qualquer fase da pesquisa, sem penalização alguma.
  - a. A qualquer momento você pode desistir de participar e retirar seu consentimento.
  - b. Sua recusa não trará nenhum prejuízo em sua relação com o pesquisador ou com a Instituição.

4. A pesquisa ocorrerá no Laboratório Intermídia que fica nas dependências do ICMC-USP, em um horário previamente definido. Durante a pesquisa você será assistido pelos condutores do estudo de caso. O recurso a ser usado é o computador e o tempo do estudo de caso não ultrapassará 60 minutos.
5. Antes e durante o curso da pesquisa, você poderá solicitar esclarecimentos a respeito dos procedimentos ou qualquer outra questão relacionada com a pesquisa.
6. Seus dados pessoais envolvidos na pesquisa serão confidenciais.
  - a. Os dados coletados no estudo de caso serão analisados e todos os participantes receberão por e-mail os resultados da Análise dos dados coletados, por meio de Artigo publicado ou um Capítulo específico da Tese de Doutorado sobre o estudo de caso desenvolvido.
  - b. Toda e qualquer informação coletada durante o estudo é tratada como confidencial. Os dados não serão divulgados de forma a possibilitar sua identificação, para isso usaremos nomes e siglas fictícias para a apresentação dos dados.
7. Os resultados obtidos através dessa pesquisa serão utilizados para investigar melhorias no framework de processo de desenvolvimento de sistemas web com acessibilidade e usabilidade, desenvolvido também por pesquisadores do ICMC.
8. Você receberá uma cópia deste termo onde consta o contato dos pesquisadores, podendo tirar suas dúvidas sobre o projeto e sua participação, agora ou a qualquer momento. Você poderá entrar em contato com os pesquisadores no endereço Avenida Trabalhador são-carlense, 400 – Centro - São Carlos - SP, pelo telefone (16) 3373-9700, ou poderá entrar em contato com o Comitê de Ética em Pesquisa – CEP/EACH, na Av. Arlindo Bértio, 1000, Ermelino Matarazzo, São Paulo-SP, telefone (11) 3091-1046.

<b>Nome</b>	<b>E-mail</b>
Ana Luiza Dias	anadias@icmc.usp.br
Profa. Dra. Renata Pontin de Mattos Fortes	renata@icmc.usp.br
Prof. Dr. Paulo Cesar Masiero	masiero@icmc.usp.br

\_\_\_\_\_  
Assinatura do Participante

\_\_\_\_\_  
Assinatura do Pesquisador responsável

São Carlos, \_\_/\_\_/\_\_.

## *Questionário pré-sessão*

Neste questionário o perfil dos usuários (professores e alunos) que estão participando do estudo de caso “Um método para garantir a acessibilidade no desenvolvimento de sistemas web” será coletado. Portanto, pedimos a gentileza de responder as perguntas abaixo.

**1. Sexo:**

- Feminino  
 Masculino

**2. Idade:** \_\_\_\_\_ anos

**3. Você possui algum tipo de deficiência?**

- Sim       Não

**Se sim, assinale a(s) opção(ões) e o nível de cada deficiência.**

- |                                   |  |                               |                                   |                                 |
|-----------------------------------|--|-------------------------------|-----------------------------------|---------------------------------|
| <input type="checkbox"/> Auditiva |  | <input type="checkbox"/> leve | <input type="checkbox"/> moderada | <input type="checkbox"/> severa |
| <input type="checkbox"/> Física   |  | <input type="checkbox"/> leve | <input type="checkbox"/> moderada | <input type="checkbox"/> severa |
| <input type="checkbox"/> Mental   |  | <input type="checkbox"/> leve | <input type="checkbox"/> moderada | <input type="checkbox"/> severa |
| <input type="checkbox"/> Visual   |  | <input type="checkbox"/> leve | <input type="checkbox"/> moderada | <input type="checkbox"/> severa |

**Se sim, utiliza algum tipo de aparelho ou acessório?**

- Não  
 Sim. Qual(ais)? \_\_\_\_\_

**Se sim, necessita de algum recurso especial para desenvolver suas atividades do dia-a-dia, incluindo seu trabalho?**

- Não  
 Sim. Qual(ais)? \_\_\_\_\_

**4. Assinale o seu Grau Acadêmico mais alto:**

- Graduação Incompleta  
 Pós Graduação Incompleta  
 Pós Graduação Completa

**5. Há quanto tempo você utiliza o computador?**

- Menos de 1 ano  
 Entre 1 e 5 anos  
 Mais de 5 anos

**6. Qual(is) tipo(s) de computador você geralmente utiliza?**

- Desktop* (monitor, gabinete, teclado e mouse)  
 Notebook  
 Celular  
 *Tablet* (Ipad, etc.)  
 Outros. Quais? \_\_\_\_\_

**7. Quanto tempo você navega na web (Internet) por semana?**

- Menos de 4 horas  
 Entre 4 e 8 horas  
 Entre 8 e 24 horas  
 Mais de 24 horas

**8. Quais atividades você geralmente desenvolve em cada tipo de computador?**

- |                                  |  |   |                                 |
|----------------------------------|--|---|---------------------------------|
| <input type="checkbox"/> Desktop |  | <input type="checkbox"/> trabalho             | <input type="checkbox"/> estudo |
|                                  |  | <input type="checkbox"/> compras              | <input type="checkbox"/> bancos |
|                                  |  | <input type="checkbox"/> jogos/entretenimento |                                 |
|                                  |  | <input type="checkbox"/> informativo          |                                 |
|                                  |  | <input type="checkbox"/> outros               | _____                           |



<input type="checkbox"/> Notebook	<input type="checkbox"/> trabalho <input type="checkbox"/> estudo <input type="checkbox"/> compras <input type="checkbox"/> bancos <input type="checkbox"/> jogos/entretenimento <input type="checkbox"/> informativo <input type="checkbox"/> outros _____
<input type="checkbox"/> Celular	<input type="checkbox"/> trabalho <input type="checkbox"/> estudo <input type="checkbox"/> compras <input type="checkbox"/> bancos <input type="checkbox"/> jogos/entretenimento <input type="checkbox"/> informativo <input type="checkbox"/> outros _____
<input type="checkbox"/> Tablet (Ipad, etc.)	<input type="checkbox"/> trabalho <input type="checkbox"/> estudo <input type="checkbox"/> compras <input type="checkbox"/> bancos <input type="checkbox"/> jogos/entretenimento <input type="checkbox"/> informativo <input type="checkbox"/> outros _____
<input type="checkbox"/> Outros.    Quais? _____	<input type="checkbox"/> trabalho <input type="checkbox"/> estudo <input type="checkbox"/> compras <input type="checkbox"/> bancos <input type="checkbox"/> jogos/entretenimento <input type="checkbox"/> informativo <input type="checkbox"/> outros _____

**9. Você sente medo ou insegurança ao utilizar a web?**

Sim                       Não

**10. Entre as suas dificuldades durante o uso de sistemas web, quais você destaca:**

<input type="checkbox"/>	Navegar nos Menus
<input type="checkbox"/>	Memorizar o caminho das tarefas
<input type="checkbox"/>	Percepção da localização em que está navegando
<input type="checkbox"/>	Reverter a ação anterior

**11. Você gosta de aprender coisas novas?**

Sim       Não

**12. Você já utilizou algum sistema web que tivesse características de usabilidade e acessibilidade?**

Sim       Não

**Se sim, você pode mencionar/citar alguma(s) característica(s) que foi(ram) utilizada(s)?**

(1) \_\_\_\_\_

(2) \_\_\_\_\_

---

## *Roteiro de Tarefas*

---

Há dois roteiros de tarefas: um para o sistema AgendAloca e outro para o sistema Agendamento.

### **Roteiro de Tarefas - AgendAloca**

**Participante** \_\_

**Horário inicial: 9:30hs**

**Horário final: 10:30hs**

#### **Instruções:**

- Você fará o papel de administrador do sistema e deve tentar executar as 6 tarefas listadas abaixo, na seção “Tarefas”.
- As tarefas serão executadas no sistema AgendAloca, no *link*:  
<http://garapa.intermidia.icmc.usp.br:3000/agendaloca/> que estará aberto no *browser* na máquina em que será realizado o teste.

#### **Tarefas:**

1- Ajustar o sistema AgendAloca conforme sua necessidade. Para isso, você poderá utilizar as opções de acessibilidade disponíveis.

2 - Fazer *login* no sistema, utilizando as seguintes informações:

Número USP: 7007001

Senha: professor

- 3- Visualizar os dados da própria conta
- 4- Consultar o calendário de bancas (onde estão expostos os horários agendados)
- 5- Cadastrar duas contas de aluno, utilizando os seguintes dados:

**Aluno 1:**

Número USP: 6006001

Nome completo: “Primeiro Aluno”

Email: [aluno1agendaloca@gmail.com](mailto:aluno1agendaloca@gmail.com)

Curso: Bacharelado em Matemática

Data de bloqueio: 25/12/2012

**Aluno 2:**

Número USP: 6006002

Nome completo: “Segundo Aluno”

Email: [aluno2agendaloca@gmail.com](mailto:aluno2agendaloca@gmail.com)

Curso: BCC

Data de bloqueio: 26/12/2012

- 6- Fazer *logoff* (sair do sistema)

**Roteiro de Tarefas - Agendamento**

**Participante** \_\_

**Horário inicial: 09:30hs**

**Horário final: 10:30hs**

**Instruções:**

- Você fará o papel de administrador do sistema e deve tentar executar as 5 tarefas listadas abaixo, na seção “Tarefas”.
- As tarefas serão executadas no sistema Agendamento, no *link*:  
<http://agua.intermidia.icmc.usp.br/agendamento/> que estará aberto no *browser* na máquina em que será realizado o teste.

**Tarefas:**

1- Fazer *login* no sistema, utilizando as seguintes informações:

Usuário: admin

Senha: lalala123

2- Visualizar os dados da própria conta

3- Consultar o calendário de bancas (onde estão expostos os horários agendados)

4- Cadastrar duas contas de aluno, utilizando os seguintes dados:

**Aluno 1:**

Usuário: aluno1

Endereço de email: aluno1agendamento@gmail.com

Senha: aluno1senha

Status: ativo

Papéis: aluno

Língua: Português brasileiro

Nome: "Primeiro Aluno"

Número USP: 6006001

**Aluno 2:**

Usuário: aluno2

Endereço de email: aluno2agendamento@gmail.com

Senha: aluno2senha

Status: ativo

Papéis: aluno

Língua: Português brasileiro

Nome: "Segundo Aluno"

Número USP: 6006002

5- Fazer *logout* (sair do sistema)



---

## *Questionário pós-sessão*

---

**OBS = Este questionário deve ser preenchido após o uso dos sistemas web.**

Por favor, em cada questão abaixo, indique com um “x” a alternativa que mais define a sua impressão sobre o uso do sistema web para agendar bancas.

**1. Você utilizou qual sistema web?**

- Agendamento - Sistema de Agendamento de Bancas (SAB)
- AgendAloca

**2. Que nível de facilidade você achou quanto aos passos para chegar na tela de agendamento de bancas:**

- Muito fácil
- Fácil
- Indiferente
- Difícil
- Muito difícil
- Não tenho como opinar

**3. Que impressão você teve quanto ao processo para agendar uma banca:**

- Muito estimulante
- Estimulante
- Indiferente
- Tedioso
- Muito tedioso

Não tenho como opinar

**4. Em relação ao caminho para agendar uma banca, você o considera:**

- Muito adequado
- Adequado
- Indiferente
- Inadequado
- Muito inadequado
- Não tenho como opinar

**5. Quanto às informações apresentadas nas telas do sistema web para agendar uma banca, você as considera:**

- Muito claras
- Claras
- Indiferente
- Confusas
- Muito confusas
- Não tenho como opinar

**6. A forma com que os elementos das telas (botões, campos, informações, etc.) estavam organizados foi:**

- Muito fácil de entender
- Fácil de entender
- Indiferente
- Difícil de entender
- Muito difícil de entender
- Não tenho como opinar

**7. Liste as dificuldades e/ou desvantagens que você identificou durante o uso do sistema web. Justifique**

---

---

---

---

---

**8. Liste as facilidades e/ou vantagens que você sentiu durante o uso do**



---

**sistema web. Justifique**

---

---

---

---

---

**9. Dentre as 10 características de usabilidade abaixo, quais delas você considerou importantes no uso do sistema?**

**(1 – muito importante, 2 – importante,**

**3 – pouco importante, 4 – não importante):**

- O sistema oferece *feedback* sobre o que está acontecendo no sistema
- Possui contextualização adequada das informações no sistema – apresenta coerentemente as informações
- Possibilita controle para retornar ou retomar ações já realizadas
- Apresenta coisas similares da mesma maneira
- Possibilita reverter erros ao realizar ações
- O sistema evita que lhe seja exigida muita memorização, por meio de reconhecimento familiar
- O sistema é fácil para usuários leigos, mas flexível o bastante para se tornar ágil a usuários avançados
- Os textos e o design são simples, diretos e naturais, presentes nos momentos em que são necessários (nada em excesso)
- Ajuda os usuários a reconhecer, diagnosticar e sanar erros
- O sistema possui documentação visível, facilmente acessada, e oferece ferramenta de busca na ajuda

**10. Entre as tecnologias de acessibilidade abaixo, quais delas você considerou importantes no uso do sistema?**

**(1 – muito importante, 2 – importante**

**3 – pouco importante, 4 – não importante):**

- Aumentar/diminuir o fonte
- Contraste de texto e fundo de tela

- Auxílio por *breadcrumbs* (caminho percorrido na interação com o sistema)
- Ir direto ao conteúdo
- Mudar o idioma
- Utilizar o TAB para navegar
- Descrição das figuras

**11. Faça suas sugestões e/ou comentários a respeito da interface do sistema que você interagiu.**

---

---

---

---

## Questionário HEUA

### Questionário de avaliação de usabilidade e acessibilidade

Q1) O sistema mantém o usuário sempre informado sobre o que está ocorrendo por meio de <i>feedbacks</i> em tempo real?	Atende	Não atende	Não se aplica
<p><b>1.1</b> O sistema deve fornecer o caminho percorrido (clicável) pelo usuário.  <i>Exemplo: breadcrumb.</i>  <i>Motivação:</i> é importante que os usuários possam rever suas experiências de sucesso como forma de incentivá-los a realizar novas interações, reduzindo a insegurança e a quantidade de erros durante a navegação ou execução de uma tarefa.  <i>Fontes:</i> Location – Montero / WCAG 2.4.8 / Lara 2.4.19</p>			
<p><b>1.2</b> O usuário deve ser capaz de perceber qual é a proposta do sistema.  <i>Exemplo:</i> nome do sistema, logotipo e título da página, imagem que identifique o sistema e seu propósito.  <i>Motivação:</i> é importante que os usuários saibam e percebam onde eles estão localizados e qual é a proposta do sistema.  <i>Fontes:</i> Tagline – Montero / WCAG 2.4.2</p>			
<p><b>1.3</b> O sistema deve informar qual é o idioma utilizado.  <i>Exemplo:</i> definir o idioma do sistema na <i>tag</i> &lt;html&gt;.  <i>Motivação:</i> para usuários que utilizam recursos de tecnologia assistiva é fundamental definir o idioma na <i>tag</i> &lt;html&gt; para que eles consigam perceber isso logo no início da navegação.  <i>Fontes:</i> WCAG 3.1.1</p>			
<p><b>1.4</b> O sistema deve informar quais partes podem ser acessadas com ou sem autenticação.  <i>Exemplo:</i> mensagem sobre a necessidade de autenticação.  <i>Motivação:</i> adultos mais velhos tendem a apresentar dificuldades de compreensão do estado do sistema e normalmente confundem mensagens de erro do sistema com as mensagens a respeito da necessidade de autenticação.  <i>Fontes:</i> Welcome – Montero / Lara 3.2.6</p>			

<p><b>1.5</b> O sistema deve informar aos usuários as condições de segurança.</p> <p><i>Exemplo:</i> fornecer mecanismos de segurança.</p> <p><i>Motivação:</i> quando os usuários fornecem informações privadas, eles precisam saber sobre a política de segurança do sistema.</p> <p><i>Fontes:</i> Secret – Montero</p>			
<p><b>1.6</b> O sistema deve fornecer áudio descrição ou mídia alternativa para conteúdos multimídia.</p> <p><i>Exemplo:</i> descrever conteúdos multimídia mesmo quando sem áudio principal usando áudio descrição ou uma alternativa para vídeo.</p> <p><i>Motivação:</i> durante as pausas existentes no diálogo, a áudio-descrição fornece informações sobre ações, personagens, mudanças de cenário e texto na tela que são importantes e que não são descritas ou faladas no áudio principal.</p> <p><i>Fontes:</i> Lara 1.3.4 / WCAG 1.2.3</p>			
<p><b>1.7</b> O sistema deve fornecer <i>áudio descrição</i> para a totalidade do vídeo.</p> <p><i>Exemplo:</i> descrever conteúdos multimídia mesmo quando sem o vídeo principal.</p> <p><i>Motivação:</i> durante as pausas existentes no diálogo, a áudio-descrição fornece informações sobre ações, personagens, mudanças de cenário e texto na tela que são importantes e que não são descritas ou faladas no áudio principal.</p> <p><i>Fontes:</i> WCAG 1.2.5</p>			
<p><b>1.8</b> O sistema deve fornecer áudio descrição estendida para vídeos quando as pausas foram insuficientes.</p> <p><i>Exemplo:</i> descrever conteúdos multimídia mesmo quando sem áudio principal usando áudio descrição estendida para vídeos no caso em que as pausas do primeiro plano do áudio sejam insuficientes para inserir a áudio descrição.</p> <p><i>Motivação:</i> pessoas cegas, pessoas com baixa visão que não pode ver a tela, bem como aqueles com limitações cognitivas que têm dificuldade em interpretar visualmente o que está acontecendo, muitas vezes, usam a descrição áudio da informação visual. No entanto, se houver um longo diálogo, a descrição de áudio é insuficiente. A áudio descrição estendida pode fornecer as informações adicionais necessárias para entender o vídeo.</p> <p><i>Fontes:</i> WCAG 1.2.7</p>			
<p><b>1.9</b> O sistema deve fornecer <i>uma mídia alternativa</i> para a totalidade do conteúdo multimídia.</p> <p><i>Exemplo:</i> oferecer uma alternativa para mídia baseada no tempo para a totalidade do conteúdo existente em mídia sincronizada e/ou pré-gravada.</p> <p><i>Motivação:</i> Pessoas que não podem ver ou ouvir muito bem podem ter acesso às informações com apresentações áudio-visuais, em forma de texto. Uma alternativa para multimídia baseada no tempo fornece uma descrição da execução de tudo o que está acontecendo no conteúdo em multimídia sincronizada.</p> <p><i>Fontes:</i> WCAG 1.2.8</p>			
<p><b>1.10</b> O sistema deve fornecer legenda de textos ao vivo para conteúdos multimídia.</p> <p><i>Exemplo:</i> esse serviço utiliza um operador humano treinado, que escuta para o que está sendo dito e utiliza um teclado especial para introduzir o texto com apenas um pequeno atraso.</p>			

<p><i>Motivação:</i> possibilita ao usuário um <i>feedback</i> em tempo real do que está ocorrendo no conteúdo multimídia disponível.</p> <p><i>Fontes:</i> WCAG 1.2.9</p>			
<b>Q2) O sistema utiliza a linguagem e o modelo mental do usuário com características na interface que são correspondentes ao mundo real?</b>	<b>Atende</b>	<b>Não atende</b>	<b>Não se aplica</b>
<p><b>2.1</b> O sistema deve utilizar uma linguagem simples e clara para o usuário.</p> <p><i>Exemplo:</i> conceitos familiares ao usuário ao invés de termos orientados ao sistema, evitar expressões idiomáticas e jargões.</p> <p><i>Motivação:</i> os usuários se diferem amplamente nos termos que utilizam para descrever conceitos, é fundamental usar uma linguagem apropriada e clara para o conteúdo do sistema.</p> <p><i>Fontes:</i> Polite – Montero / WCAG 3.1.3</p>			
<p><b>2.2</b> O sistema deve oferecer ao usuário a possibilidade de compreender a finalidade de cada <i>link</i> a partir do texto do <i>link</i> ou juntamente com o respectivo contexto do <i>link</i>.</p> <p><i>Exemplo:</i> <i>links</i> significativos.</p> <p><i>Motivação:</i> os usuários necessitam saber onde eles podem ir e o que eles podem fazer a partir do ponto onde estão, portanto é fundamental fornecer <i>links</i> significativos para que o usuário identifique facilmente a finalidade do <i>link</i>. Sempre que possível, o sistema deve fornecer um texto que identifique a finalidade do <i>link</i> sem a necessidade de contexto adicional.</p> <p><i>Fontes:</i> Indication – Montero / WCAG 2.4.4</p>			
<p><b>2.3</b> O sistema deve oferecer ao usuário a possibilidade de compreender a finalidade de cada <i>link</i> a partir do texto do <i>link</i>.</p> <p><i>Exemplo:</i> lista de <i>links</i>.</p> <p><i>Motivação:</i> o usuário deve ser capaz de compreender a finalidade de cada <i>link</i> mesmo quando eles estão fora de contexto, por exemplo, quando o sistema fornece uma lista de todos os <i>links</i> em uma página.</p> <p><i>Fontes:</i> WCAG 2.4.9</p>			
<p><b>2.4</b> O sistema deve oferecer um mecanismo para identificar a forma completa ou o significado de abreviaturas.</p> <p><i>Exemplo:</i> abreviatura significativa.</p> <p><i>Motivação:</i> os usuários necessitam entender o significado de abreviaturas existentes no sistema para navegarem tranquilamente.</p> <p><i>Fontes:</i> WCAG 3.1.4</p>			
<p><b>2.5</b> O sistema deve fornecer cabeçalhos (<i>headings</i>) e rótulos (<i>labels</i>) que descrevem a sua finalidade.</p> <p><i>Exemplo:</i> cabeçalhos e rótulos significativos.</p> <p><i>Motivação:</i> os usuários precisam perceber rapidamente a finalidade dos cabeçalhos e rótulos para uma navegação segura e preventiva.</p> <p><i>Fontes:</i> WCAG 2.4.6</p>			
<p><b>2.6</b> O sistema deve apresentar ícones que possam ser facilmente associados a objetos do mundo real.</p>			

<p><i>Exemplo:</i> impressora para imprimir, disquete para salvar.  <i>Motivação:</i> adultos mais velhos tendem a relacionar as informações que visualizam com objetos do mundo real na tentativa de construção de um modelo mental que os façam lembrar a informação ou tarefa em momentos posteriores.  <i>Fontes:</i> Lara, 2.4.18</p>			
<p><b>2.7</b> O sistema deve apresentar conteúdo adicional para auxiliar a compreensão de textos difíceis ou complexos.  <i>Exemplo:</i> informar sobre conteúdo adicional disponível.  <i>Motivação:</i> conteúdos adicionais podem ajudar os usuários com dificuldades de leitura e ao mesmo tempo permitir que os desenvolvedores publiquem conteúdo web difícil ou complexo.  <i>Fontes:</i> WCAG 3.1.5</p>			
<p><b>2.8</b> O sistema deve fornecer ao usuário linguagem gestual para multimídias.  <i>Exemplo:</i> incluir linguagem textual em vídeos.  <i>Motivação:</i> permitir que pessoas surdas ou com deficiência auditiva que são fluentes em uma língua de sinais entendam o conteúdo da faixa de áudio de apresentações em multimídia sincronizada.  <i>Fontes:</i> WCAG 1.2.6</p>			
<b>Q3) O sistema oferece controle e liberdade ao usuário para sair de estados indesejáveis?</b>			
	<b>Atende</b>	<b>Não atende</b>	<b>Não se aplica</b>
<p><b>3.1</b> O sistema deve oferecer opções de controle aos usuários.  <i>Exemplo:</i> desfazer, refazer, voltar e limpar.  <i>Motivação:</i> os usuários precisam sentir que estão no controle de suas operações.  <i>Fontes:</i> Second Chance – Montero</p>			
<p><b>3.2</b> O sistema deve oferecer a possibilidade de que toda funcionalidade seja operável utilizando somente o teclado sem necessidade de qualquer espaço de tempo.  <i>Exemplo:</i> navegação por TAB.  <i>Motivação:</i> usuários que fazem uso de recursos de tecnologia assistiva precisam que todas as funcionalidades estejam disponíveis via teclado.  <i>Fontes:</i> WCAG 2.1.1</p>			
<p><b>3.3</b> O sistema deve oferecer a possibilidade de navegação utilizando somente o teclado.  <i>Exemplo:</i> navegação por TAB.  <i>Motivação:</i> usuários que fazem uso de recursos de tecnologia assistiva precisam que todas as funcionalidades estejam disponíveis via teclado.  <i>Fontes:</i> WCAG 2.1.2</p>			
<p><b>3.4</b> O sistema deve oferecer a possibilidade de navegação utilizando o teclado dentro de subseções de conteúdo do sistema.  <i>Exemplo:</i> Um <i>widget</i> de calendário permite aos usuários adicionar, remover ou atualizar itens usando o teclado.</p>			

<p><i>Motivação:</i> usuários que dependem de uma interface de teclado ou o teclado para usar a web, incluindo pessoas cegas e pessoas com deficiência física.</p> <p><i>Fontes:</i> WCAG 2.1.3</p>			
<p><b>3.5</b> O sistema deve oferecer suporte para que o usuário possa ignorar blocos de conteúdo (repetidos ou não) do sistema.</p> <p><i>Exemplo:</i> link “pular para o conteúdo”.</p> <p><i>Motivação:</i> quando há essa ancora para ir direto ao conteúdo do sistema, a navegação de usuários que utilizam recursos de tecnologia assistiva é muito facilitada.</p> <p><i>Fontes:</i> WCAG 2.4.1</p>			
<p><b>3.6</b> O sistema deve oferecer ao usuário a possibilidade de navegar sequencialmente pelo sistema, com ou sem tecnologia assistiva, sem afetar o significado ou a operação dos elementos.</p> <p><i>Exemplo:</i> navegação da esquerda para direita e/ou da parte superior para a parte inferior.</p> <p><i>Motivação:</i> ajudar pessoas que dependem de leitores de tela. O significado da informação na apresentação padrão será o mesmo quando o conteúdo for apresentado em forma falada.</p> <p><i>Fontes:</i> WCAG 1.3.2</p>			
<p><b>3.7</b> O sistema deve oferecer ao usuário a possibilidade de navegar sequencialmente utilizando a tabulação do teclado.</p> <p><i>Exemplo:</i> deixar a sequência da navegação por TAB do teclado, a mesma da navegação visual e ordem dos elementos.</p> <p><i>Motivação:</i> beneficiar os usuários que utilizam o teclado para navegar nos documentos. Pessoas com deficiência visual podem ficar desorientadas se a tabulação levar o foco a algum lugar inesperado ou então quando não puderem encontrar facilmente o conteúdo desejado.</p> <p><i>Fontes:</i> WCAG 2.4.3</p>			
<p><b>3.8</b> O sistema deve oferecer ao usuário a opção de alterar o idioma em que o conteúdo está escrito.</p> <p><i>Exemplo:</i> opções de internacionalização para alterar todo o conteúdo do sistema.</p> <p><i>Motivação:</i> quando há opções de internacionalização em um sistema, ele se torna mais acessível a inúmeros usuários. Falar a linguagem do usuário é projetar para todos.</p> <p><i>Fontes:</i> Polyglot – Montero / WCAG 3.1.2</p>			
<p><b>3.9</b> O sistema deve oferecer um <i>link</i> (ou caminho direto) para que o usuário consiga voltar ao estado inicial do sistema.</p> <p><i>Exemplo:</i> link “voltar à página inicial”.</p> <p><i>Motivação:</i> um sistema pode ser acessado de diversas maneiras, mas o usuário precisa estar seguro de que independente de sua navegação ele tem “a segurança” de voltar ao estado inicial caso se sinta perdido pelo sistema.</p> <p><i>Fontes:</i> Homepage – Montero</p>			
<p><b>3.10</b> O sistema deve oferecer opções para que o usuário possa controlar conteúdos multimídia.</p> <p><i>Exemplo:</i> controle da velocidade de execução.</p> <p><i>Motivação:</i> conteúdo multimídia afeta o foco de atenção dos adultos mais velhos desviando-a da execução da tarefa e</p>			

frequentemente não é bem compreendido devido à alta velocidade de execução. <i>Fontes:</i> WCAG 1.4.2 / Lara 2.2.7			
<b>3.110</b> sistema deve oferecer opções para áudio pré-gravado como som de fundo. <i>Exemplo:</i> desligar ou abaixar o volume do áudio. <i>Motivação:</i> qualquer som de fundo de um sistema (seja ele música, voz etc.) deve ter a possibilidade de ser desligado, pois muitos usuários acabam se distraindo da navegação e cometendo possíveis erros que provavelmente não cometeriam caso não houvesse som. <i>Fontes:</i> WCAG 1.4.7			
<b>3.120</b> sistema deve oferecer controles para informações que estejam em movimento, em modo intermitente. <i>Exemplo:</i> pausar, parar ou ocultar. <i>Motivação:</i> os usuários devem conseguir controlar informações em movimentos e que estejam atrapalhando/dificultando sua interação com o sistema. <i>Fontes:</i> WCAG 2.2.2			
<b>3.130</b> sistema deve oferecer ao usuário maneira de interromper/desativar atualizações, exceto alertas de emergências (que incluem mensagens de alerta de perigo de saúde, segurança ou direito de propriedade, incluindo perda de dados, perda de conexão etc.). <i>Exemplo:</i> interromper atualizações. <i>Motivação:</i> interrupções permitem o acesso de pessoas com limitações cognitivas ou distúrbios de atenção, permitindo-lhes concentrar no conteúdo, além disso, colabora para que usuários cegos ou com baixa visão para manter a sua "visão" foco no conteúdo que eles estão lendo no momento. <i>Fontes:</i> WCAG 2.2.4			
<b>Q4) O sistema é consistente e segue o mesmo padrão em toda a interface a fim de facilitar o reconhecimento do usuário?</b>	<b>Atende</b>	<b>Não atende</b>	<b>Não se aplica</b>
<b>4.1</b> O sistema deve fornecer informações sobre o tamanho de qualquer elemento que o usuário possa baixar (fazer download). <i>Exemplo:</i> tamanho do elemento. <i>Motivação:</i> <i>downloads</i> podem demorar muito tempo, criando atrasos significativos e os usuários precisam ser informados sobre o tamanho do elemento para download. <i>Fontes:</i> Busy – Montero			
<b>4.2</b> O sistema deve destacar os elementos importantes. <i>Exemplo:</i> colocar no alto/início do sistema elementos como <i>links</i> , menus e botões. <i>Motivação:</i> adultos mais velhos apresentam dificuldades para identificar e distinguir os componentes da interface de um sistema web e, portanto é necessário que seja oferecido algum meio de destacá-los. Além disso, adultos mais velhos apresentam dificuldades para diferenciar elementos navegacionais do conteúdo da página. <i>Fontes:</i> Lara 1.4.11			



<p><b>4.3</b> O sistema deve utilizar outros meios além da cor para distinguir um elemento.  <i>Exemplo:</i> informação textual.  <i>Motivação:</i> a cor não pode ser utilizada como único meio visual de transmitir informações, indicar uma ação, pedir uma resposta ou distinguir um elemento visual, pois pode prejudicar os usuários com deficiência visual.  <i>Fontes:</i> WCAG 1.4.1</p>			
<p><b>4.4</b> O sistema deve deixar visível o foco nos elementos quando eles são selecionados via teclado.  <i>Exemplo:</i> destacar com uma borda os elementos quando são selecionados.  <i>Motivação:</i> deixar visível os elementos quando são selecionados ajuda muito a navegação de usuários que fazem uso de recursos de tecnologia assistiva.  <i>Fontes:</i> WCAG 2.4.7</p>			
<p><b>4.5</b> O sistema deve manter o contexto quando um elemento apenas recebe o foco.  <i>Exemplo:</i> permanecer apenas com o foco visível.  <i>Motivação:</i> os elementos precisam alterar o contexto somente quando os elementos são ativados/clicados, caso contrário, devem permanecer da maneira como estão.  <i>Fontes:</i> WCAG 3.2.1</p>			
<p><b>4.6</b> O sistema deve diferenciar a apresentação de <i>links</i> de outros componentes que não aceitam ativação.  <i>Exemplo:</i> sublinhar apenas <i>links</i>.  <i>Motivação:</i> usuários tendem a apresentar dificuldades na distinção entre componentes que possibilitam e os que não possibilitam interação.  <i>Fontes:</i> Lara 3.2.8</p>			
<p><b>4.7</b> O sistema deve fornecer um mecanismo para que o usuário possa identificar a pronúncia de palavras específicas.  <i>Exemplo:</i> fornecer arquivos de som da pronúncia, incluir informações de pronúncia no glossário.  <i>Motivação:</i> se a pronúncia correta de uma palavra puder ser determinada a partir do contexto da frase, então nada é necessário. Se não puder, algum mecanismo para determinar a pronúncia correta é necessário para ajudar os usuários com deficiência visual que dependem da pronúncia do conteúdo para compreensão.  <i>Fontes:</i> WCAG 3.1.6</p>			
<p><b>4.8</b> O sistema deve alterar o contexto somente quando um elemento é ativado.  <i>Exemplo:</i> apresentar claramente que o elemento pode ser ativado.  <i>Motivação:</i> pode ajudar a eliminar possíveis confusões causadas por mudanças inesperadas de contexto como a abertura automático de novas janelas, a submissão automática de formulários após a seleção de um item de uma lista, etc.  <i>Fontes:</i> WCAG 3.2.5</p>			
<p><b>4.9</b> O sistema deve garantir que todos os elementos dispõem de marcações de início e fim completas.  <i>Exemplo:</i> deixar o código HTML bem formado.  <i>Motivação:</i> garantir que recursos de tecnologia assistiva possam interpretar com precisão e analisar o conteúdo, facilitando o entendimento/navegação do usuário.</p>			

<i>Fontes: WCAG 4.4.1</i>			
<p><b>4.10</b> O sistema deve garantir que toda mídia pré-gravada possua legenda.</p> <p><i>Exemplo:</i> legendas em vídeos, legendas em áudios, identificando quem está falando.</p> <p><i>Motivação:</i> permitir que pessoas surdas ou com deficiência auditiva possam assistir apresentações em multimídia sincronizada e as legendas podem proporcionar isso.</p> <p><i>Fontes: WCAG 1.2.2</i></p>			
<p><b>4.11</b> O sistema deve garantir que todo áudio ao vivo possua legenda.</p> <p><i>Exemplo:</i> legendas em áudios ao vivo, identificando quem está falando.</p> <p><i>Motivação:</i> permitir que pessoas surdas ou com deficiência auditiva possam assistir apresentações em tempo real e as legendas podem proporcionar isso.</p> <p><i>Fontes: WCAG 1.2.4</i></p>			
<b>Q5) O sistema apresenta um projeto (design) preventivo e cuidadoso que pode ser capaz de evitar algum problema durante a interação do usuário?</b>			
	<b>Atende</b>	<b>Não atende</b>	<b>Não se aplica</b>
<p><b>5.1</b> O sistema deve inibir a entrada de dados incorretos pelos usuários.</p> <p><i>Exemplo:</i> menu de seleção, máscara, <i>labels</i> de exemplo.</p> <p><i>Motivação:</i> na página web que contém campos para a entrada de dados deve ser explicitada informações sobre o formato de dado e seu domínio, ou seja, informações de como o usuário deve preencher o campo. Usuários, tais como adultos mais velhos apresentam dificuldades no preenchimento de campos de formulário e necessitam de visualização de exemplos e explicações de como completar a tarefa.</p> <p><i>Fontes: Form – Montero / WCAG 3.3.2 /Lara 3.3.7</i></p>			
<p><b>5.2</b> O sistema deve utilizar cabeçalhos para organizar seções de conteúdos.</p> <p><i>Exemplo:</i> utilizar <i>tags HTML</i>, tais como: &lt;h1&gt;, &lt;h2&gt;.</p> <p><i>Motivação:</i> além de ser mais fácil visualmente para perceber as organizações das seções, é muito mais fácil a navegação para usuários que utilizam recursos de tecnologia assistiva.</p> <p><i>Fontes: WCAG 2.4.10</i></p>			
<p><b>5.3</b> O sistema deve manter o foco quando o usuário faz uma seleção por filtro.</p> <p><i>Exemplo:</i> destacar com bordas o foco após a ativação do filtro.</p> <p><i>Motivação:</i> o usuário não fica perdido no sistema, pois o foco permanece inalterado.</p> <p><i>Fontes: WCAG 3.2.2</i></p>			
<p><b>5.4</b> O sistema deve permitir ao usuário revisar os dados de entrada antes de submetê-los para sistemas web com responsabilidades jurídicas ou transações</p> <p><i>Exemplo:</i> apresentar uma página resumo.</p> <p><i>Motivação:</i> uma página resumo pode evitar erros futuros, pois o usuário já pode detectar algum problema antes mesmo da submissão. Isso torna um projeto mais preventivo.</p> <p><i>Fontes: WCAG 3.3.4</i></p>			

<p><b>5.5</b> O sistema deve permitir ao usuário revisar os dados de entrada antes de submetê-los para sistemas web que exijam que o usuário envie informações.</p> <p><i>Exemplo:</i> apresentar uma página resumo das informações inseridas pelo usuário.</p> <p><i>Motivação:</i> uma página resumo pode evitar erros futuros, pois o usuário já pode detectar algum problema antes mesmo da submissão. Isso torna um projeto mais preventivo.</p> <p><i>Fontes:</i> WCAG 3.3.6</p>			
<p><b>5.6</b> O sistema deve diferenciar <i>links</i> visitados de <i>links</i> não visitados.</p> <p><i>Exemplo:</i> alterar a cor do <i>link</i> visitado.</p> <p><i>Motivação:</i> Alguns usuários, como os adultos mais velhos, tendem a esquecer de <i>links</i> que já visitaram, devido às deficiências cognitivas.</p> <p><i>Fontes:</i> RecognizeYou – Montero / Lara 2.4.14</p>			
<p><b>5.7</b> O sistema deve utilizar textos para transmitir informações ao invés de imagens de texto.</p> <p><i>Exemplo:</i> utilizar apenas texto sempre que possível.</p> <p><i>Motivação:</i> o uso de imagens de texto pode prejudicar a interação dos usuários que fazem uso de alguma tecnologia assistiva ou mesmo quando o usuário tem um monitor com configuração diferente da configuração de quem desenvolveu a imagem, pois ela acaba ficando distorcida.</p> <p><i>Fontes:</i> WCAG 1.4.5</p>			
<p><b>5.8</b> O sistema SEMPRE deve utilizar textos para transmitir informações ao invés de imagens de texto.</p> <p><i>Exemplo:</i> utilizar apenas texto.</p> <p><i>Motivação:</i> o uso de imagens de texto pode prejudicar a interação dos usuários que fazem uso de alguma tecnologia assistiva ou mesmo quando o usuário tem um monitor com configuração diferente da configuração de quem desenvolveu a imagem, pois ela acaba ficando distorcida.</p> <p><i>Fontes:</i> WCAG 1.4.9</p>			
<p><b>5.9</b> O sistema deve utilizar <i>flash</i> com frequência inferior a três vezes por segundo ou abaixo dos limites de <i>flash</i> universal e <i>flash</i> vermelho.</p> <p><i>Exemplo:</i> um filme com uma cena envolvendo muito brilhantes relâmpagos deve ser editado para que o <i>flash</i> pisque somente três vezes no período de um segundo.</p> <p><i>Motivação:</i> permitir aos usuários acessar o conteúdo completo de um sistema sem induzir convulsões, devido à foto sensibilidade.</p> <p><i>Fontes:</i> WCAG 2.3.1</p>			
<p><b>5.10</b> O sistema deve utilizar <i>flash</i> APENAS com frequência inferior a três vezes por segundo, independente do brilho ou do tamanho do elemento.</p> <p><i>Exemplo:</i> <i>flash</i> simples.</p> <p><i>Motivação:</i> reduzir ainda mais o risco de convulsões.</p> <p><i>Fontes:</i> WCAG 2.3.2</p>			
<p><b>5.11</b> O sistema deve ser independente do recurso de barra de rolagem.</p> <p><i>Exemplo:</i> <i>flash</i> simples.</p> <p><i>Motivação:</i> alguns usuários, tais como adultos mais velhos, apresentam dificuldades para perceber a existência de barra de</p>			

<p>rolagem e geralmente não completam a tarefa que requer o uso desse componente da interface.</p> <p><i>Fontes:</i> Lara 2.1.4</p>			
<p><b>5.12</b> O sistema deve evitar o uso de janelas pop-up.</p> <p><i>Exemplo:</i> colocar avisos/mensagens na mesma página.</p> <p><i>Motivação:</i> alguns usuários, tais como adultos mais velhos, tendem a apresentar uma reação de resistência e confusão quando uma janela pop-up aparece no meio da tela, pois normalmente acreditam que causaram algum erro no sistema.</p> <p><i>Fontes:</i> Lara 2.1.5</p>			
<p><b>5.13</b> O sistema deve apresentar menus do tipo <i>drop-down</i> com velocidade reduzida o suficiente para serem ativados por usuários que apresentam pouca coordenação motora.</p> <p><i>Exemplo:</i> colocar um atraso na volta dos menus após o usuário passar o mouse sobre ele.</p> <p><i>Motivação:</i> alguns usuários, tais como adultos mais velhos, apresentam dificuldades na manipulação de menus móveis rápidos e geralmente acabam selecionando a opção errada.</p> <p><i>Fontes:</i> Lara 2.2.6</p>			
<p><b>5.14</b> O sistema deve apresentar <i>links</i> adjacentes na interface com espaço suficiente entre eles para evitar erros.</p> <p><i>Exemplo:</i> distanciar ao máximo os <i>links</i> adjacentes.</p> <p><i>Motivação:</i> adultos mais velhos, em especial os idosos, apresentam dificuldades de precisão ao clicar em pequenas áreas de <i>links</i>.</p> <p><i>Fontes:</i> Lara 2.4.12</p>			
<p><b>5.15</b> O sistema deve apresentar ícones ou imagens adjacentes na interface como parte da área de ativação dos <i>links</i>.</p> <p><i>Exemplo:</i> reservar uma área clicável maior.</p> <p><i>Motivação:</i> imagens pequenas ou ícones adjacentes aos <i>links</i> confundem os adultos mais velhos, visto que eles tendem a clicar nas imagens ao invés dos <i>links</i>.</p> <p><i>Fontes:</i> Lara 2.4.13</p>			
<p><b>5.16</b> O sistema deve apresentar de forma distinta as opções de menu que já foram visitadas daquelas que ainda não foram selecionadas.</p> <p><i>Exemplo:</i> alterar a cor ou a fonte do menu visitado.</p> <p><i>Motivação:</i> Alguns usuários, como os adultos mais velhos, tendem a esquecer quais opções de menu eles já visitaram.</p> <p><i>Fontes:</i> Lara 2.4.15</p>			
<p><b>5.17</b> O sistema deve apresentar menus do tipo <i>drop-down</i> com apenas um nível de opções.</p> <p><i>Exemplo:</i> menus mais simples.</p> <p><i>Motivação:</i> alguns usuários, tais como adultos mais velhos, apresentam grande dificuldade ao interagir com vários níveis de menus do tipo <i>drop-down</i>, acionando as opções erradas frequentemente.</p> <p><i>Fontes:</i> Lara 2.4.16</p>			
<p><b>5.18</b> O sistema deve apresentar as opções intermediárias do menu, quando selecionadas, em uma página com elementos subsequentes e elementos alternativos para o seu acionamento.</p>			

<p><i>Exemplo:</i> botões e links.  <i>Motivação:</i> é comum adultos mais velhos acionarem as opções intermediárias dos menus quando estão tentando alcançar uma opção do último nível.  <i>Fontes:</i> Lara 2.4.17</p>			
<p><b>5.19</b> O sistema deve ser capaz de oferecer ao usuário a opção de continuar com a atividade sem perder os dados quando uma sessão autenticada expirar.  <i>Exemplo:</i> opção de re-autenticação durante uma atividade.  <i>Motivação:</i> quando os usuários são desconectados no meio de uma transação, é importante que seja dada a capacidade de re-autenticar e continuar com a transação sem a perda de todos os dados já inseridos.  <i>Fontes:</i> WCAG 2.2.5</p>			
<b>Q6) O sistema evita a sobrecarga de memória do usuário fornecendo informações contextuais para cada ação?</b>			
	<b>Atende</b>	<b>Não atende</b>	<b>Não se aplica</b>
<p><b>6.1</b> O sistema deve utilizar a mesma terminologia para referenciar somente as mesmas ações.  <i>Exemplo:</i> termos idênticos nos menus e help.  <i>Motivação:</i> os usuários, bem como os adultos mais velhos podem encontrar dificuldades para identificar e distinguir corretamente as funcionalidades do sistema.  <i>Fontes:</i> WCAG 3.2.4 / Lara 3.2.7</p>			
<p><b>6.2</b> O sistema deve utilizar as mesmas cores, fontes, localização de navegação e leiautes em toda a interface.  <i>Exemplo:</i> usar uma única folha de estilo para todas as páginas.  <i>Motivação:</i> a padronização da interface ajuda os usuários a perceber que estão navegando no mesmo sistema mesmo quando navegam por vários caminhos.  <i>Fontes:</i> Similarity – Montero</p>			
<p><b>6.3</b> O sistema deve manter a mesma ordem de elementos que são repetidos.  <i>Exemplo:</i> organizar os elementos por ordem de prioridade.  <i>Motivação:</i> os elementos que aparecem diversas vezes em um mesmo sistema devem permanecer no mesmo local sempre, para que os usuários possam reconhecê-los.  <i>Fontes:</i> WCAG 3.2.3</p>			
<p><b>6.4</b> O sistema deve utilizar nomes diferentes para referenciar URLs distintas.  <i>Exemplo:</i> utilizar um nome para cada URL disponível no sistema.  <i>Motivação:</i> os usuários, bem como os adultos mais velhos se acostumam com a interface apresentada e qualquer renomeação de referências deve ser evitada.  <i>Fontes:</i> Lara 2.4.11</p>			
<p><b>6.5</b> O sistema deve fornecer informações necessárias para a conclusão de tarefas específicas.  <i>Exemplo:</i> explicações de como completar uma tarefa específica.  <i>Motivação:</i> adultos mais velhos tendem a necessitar de mais tempo que os jovens para realizar uma tarefa complexa, tais</p>			

<p>como o preenchimento de formulários, confirmações de pagamentos, entre outras, o que acaba gerando a expiração da sessão do sistema. Dessa forma a página web deve apresentar as explicações de como completar a tarefa previamente.</p> <p><i>Fontes:</i> Lara 3.3.8</p>			
<p><b>6.6</b> O sistema deve fornecer ajuda contextualizada ao usuário.</p> <p><i>Exemplo:</i> ajuda contextualizada.</p> <p><i>Motivação:</i> usando ajuda sensível ao contexto, os usuários descobrem como executar uma operação sem perder de vista o que eles estão fazendo, evitando assim possíveis erros.</p> <p><i>Fontes:</i> WCAG 3.3.5</p>			
<b>Q7) O sistema oferece flexibilidade e eficiência aos usuários, agilizando o uso para usuários experientes e mantendo a facilidade para os novatos?</b>			
	<b>Atende</b>	<b>Não atende</b>	<b>Não se aplica</b>
<p><b>7.1</b> O sistema deve oferecer tempo suficiente para leitura e utilização de conteúdo</p> <p><i>Exemplo:</i> evitar o uso de temporização.</p> <p><i>Motivação:</i> os usuários precisam do tempo que lhes for necessário para realizar leituras e utilizar o conteúdo disponível no sistema.</p> <p><i>Fontes:</i> WCAG 2.2.3</p>			
<p><b>7.2</b> O sistema deve oferecer a opção de imprimir ou de salvar documentos grandes.</p> <p><i>Exemplo:</i> fornecer a mesma informação de diversas maneiras.</p> <p><i>Motivação:</i> a leitura de textos em um sistema é diferente da leitura de textos impressos, portanto o sistema deve fornecer a mesma informação de diversas maneiras para que o usuário possa escolher a que mais lhe agrada.</p> <p><i>Fontes:</i> Print – Montero</p>			
<p><b>7.3</b> O sistema deve oferecer descrição textual em todos os seus elementos não textuais.</p> <p><i>Exemplo:</i> alternativa textual para imagens.</p> <p><i>Motivação:</i> alternativas textuais são soluções fundamentais para usuários que tem deficiência visual e dependem de leitores de tela para navegarem em sistemas web.</p> <p><i>Fontes:</i> WCAG 1.1.1</p>			
<p><b>7.4</b> O sistema deve oferecer opção de contraste para textos e imagens de texto no mínimo de 4.5:1.</p> <p><i>Exemplo:</i> opção de contraste de no mínimo 4.5:1.</p> <p><i>Motivação:</i> ajudar usuários com baixa acuidade visual, deficiências de cor congênitas ou adquiridas, ou a perda de sensibilidade ao contraste, que normalmente acompanha o envelhecimento.</p> <p><i>Fontes:</i> WCAG 1.4.3</p>			
<p><b>7.5</b> O sistema deve oferecer opção de contraste para textos e imagens de texto no mínimo de 7:1.</p> <p><i>Exemplo:</i> opção de contraste de no mínimo 7:1.</p> <p><i>Motivação:</i> fornecer contraste suficiente entre o texto e o fundo de modo que possa ser lido por pessoas com baixa visão moderada (que não usam contraste de melhoria de tecnologia assistiva).</p> <p><i>Fontes:</i> WCAG 1.4.6</p>			

<p><b>7.6</b> O sistema deve oferecer opção de zoom para textos e imagens de texto.  <i>Exemplo:</i> opção de zoom.  <i>Motivação:</i> o sistema deve oferecer opção de zoom sem tecnologia de apoio para usuários com baixa visão.  <i>Fontes:</i> WCAG 1.4.4</p>			
<p><b>7.7</b> O sistema deve garantir que todos os elementos são acessíveis por tecnologia de apoio.  <i>Exemplo:</i> seguir ao máximo o padrão HTML.  <i>Motivação:</i> para todos os componentes de interface deve haver uma maneira de interpretá-los utilizando recursos de tecnologia assistiva. Dessa maneira, o sistema estará acessível a todos os tipos de usuários.  <i>Fontes:</i> WCAG 4.1.2</p>			
<p><b>7.8</b> O sistema deve garantir que todos os elementos estão disponíveis mesmo se scripts forem desabilitados.  <i>Exemplo:</i> utilizar descrição textual para todos os elementos do sistema.  <i>Motivação:</i> a identificação textual para todos os elementos do sistema é fundamental, pois se houver uma desativação de scripts, mesmo assim ainda será possível navegar pelo sistema.  <i>Fontes:</i> WCAG 1.3.1</p>			
<p><b>7.9</b> O sistema deve oferecer conteúdo independente de características sensoriais.  <i>Exemplo:</i> não utilizar apenas sons como informativos.  <i>Motivação:</i> o conteúdo do sistema deve ser independente de características sensoriais, pois alguns usuários podem ter dificuldades sensoriais, o que inviabiliza a interação com o sistema.  <i>Fontes:</i> WCAG 1.3.3</p>			
<p><b>7.10</b> O sistema deve oferecer uma ferramenta de busca para que os usuários localizem as opções disponíveis  <i>Exemplo:</i> campo de busca.  <i>Motivação:</i> busca é um dos principais elementos que um sistema deve ter para que os usuários localizem as opções disponíveis de forma fácil e sem muito esforço. Além disso, facilita e agiliza a interação de usuários mais experientes.  <i>Fontes:</i> Search – Montero / WCAG 2.4.5</p>			
<p><b>7.11</b> O sistema deve fornecer opções para que o usuário consiga informações automaticamente.  <i>Exemplo:</i> <i>checkboxlist</i> de informações que podem ser enviadas por e-mail.  <i>Motivação:</i> os usuários podem não querer visitar o sistema web a todo o momento, no entanto, podem desejar saber quando novos produtos ou novidades aparecem, automaticamente.  <i>Fontes:</i> Subscription – Montero</p>			
<p><b>7.12</b> O sistema deve disponibilizar diferentes meios de preenchimento de informações em campos de formulário além do teclado.  <i>Exemplo:</i> <i>checkboxlist</i> de informações que podem ser enviadas por e-mail.  <i>Motivação:</i> adultos mais velhos tendem a apresentar grandes dificuldades na digitação de sequências de números pelo teclado.  <i>Fontes:</i> Lara 3.3.10</p>			
<p><b>7.13</b> O sistema deve oferecer ajustes de temporização para cada limite de tempo definido pelo conteúdo</p>			

<p><i>Exemplo:</i> desligar, ajustar e prolongar o tempo.  <i>Motivação:</i> os usuários precisam do tempo que lhes for necessário, por isso devem poder desligar ou ajustar o limite de tempo antes de atingi-lo e serem avisados antes de o tempo expirar, tendo no mínimo, 20 segundos para prolongar o limite de tempo com uma simples ação.  <i>Fontes:</i> WCAG 2.2.1</p>			
<p><b>7.14</b> O sistema deve oferecer ao usuário uma transcrição equivalente ao que é apresentado visualmente em vídeos.  <i>Exemplo:</i> transcrição de vídeos disponíveis.  <i>Motivação:</i> alternativas textuais para multimídia podem tornar a informação acessível, pois o texto pode ser processado por qualquer modalidade sensorial (por exemplo, visual, auditiva ou tátil) para atender às necessidades do usuário.  <i>Fontes:</i> WCAG 1.2.1</p>			
<b>Q8) O sistema oferece estética e projeto (design) minimalista, mantendo apenas informações úteis, diretas e claras?</b>			
	<b>Atende</b>	<b>Não atende</b>	<b>Não se aplica</b>
<p><b>8.1</b> O sistema deve utilizar cores adequadas nas fontes, fundos de tela e imagens.  <i>Exemplo:</i> considerar a cor no início do projeto.  <i>Motivação:</i> proporcionar melhorias no <i>feedback</i>, funcionalidade, consistência e clareza visual.  <i>Fontes:</i> Colour – Montero</p>			
<p><b>8.2</b> O sistema deve utilizar um tamanho de fonte legível.  <i>Exemplo:</i> escolher a fonte do sistema de acordo com o público alvo.  <i>Motivação:</i> proporcionar melhorias no controle, consistência e clareza visual.  <i>Fontes:</i> Size – Montero</p>			
<p><b>8.3</b> O sistema deve fornecer ferramentas ou informações necessárias para que o usuário acesse no sistema web de modo adequado.  <i>Exemplo:</i> mensagem de aviso sobre uma tecnologia que o usuário precisa ter instalado.  <i>Motivação:</i> os sistemas precisam detectar se o usuário possui tudo o que é necessário para navegar no sistema e caso não haja, deve fornecer <i>links</i> para download.  <i>Fontes:</i> Ready – Montero</p>			
<p><b>8.4</b> O sistema deve oferecer sugestões e novidades de uma maneira simples e intuitiva.  <i>Exemplo:</i> destacar as novidades no início do sistema.  <i>Motivação:</i> usuários gostam de saber se existem novas funcionalidades, promoções, ofertas, notícias no sistema web, portanto, o sistema deve fornecer sugestões e novidades de uma maneira simples e intuitiva.  <i>Fontes:</i> Novelty – Montero</p>			
<p><b>8.5</b> O sistema deve conter blocos de texto com largura inferior a 80 caracteres.  <i>Exemplo:</i> oferecer opção de redimensionamento de texto.  <i>Motivação:</i> quando há blocos de textos menores, os usuários têm mais facilidade para ler e entender o conteúdo.</p>			



<i>Fontes: WCAG 1.4.8</i>			
<b>Q9) O sistema ajuda o usuário a reconhecer, diagnosticar e corrigir erros?</b>	<b>Atende</b>	<b>Não atende</b>	<b>Não se aplica</b>
<p><b>9.1</b> O sistema deve detectar e avisar ao usuário se ele tem todos os <i>plug-ins</i> necessários para que o sistema funcione normalmente.</p> <p><i>Exemplo:</i> exibir uma mensagem sobre os <i>plug-ins</i> necessários logo que o usuário acessa o sistema.</p> <p><i>Motivação:</i> usuários gostam de saber se existem novas funcionalidades, promoções, ofertas, notícias no sistema web, portanto, o sistema deve fornecer sugestões e novidades de uma maneira simples e intuitiva.</p> <p><i>Fontes:</i> Danger – Montero</p>			
<p><b>9.2</b> O sistema deve fornecer mensagens de erro ao usuário que devem ser as mais específicas possíveis.</p> <p><i>Exemplo:</i> No caso de envio de formulário sem sucesso, reexibir o formulário e indicar os campos de erro.</p> <p><i>Motivação:</i> mensagens de erro específicas ajudam o usuário a identificar e corrigir o possível erro.</p> <p><i>Fontes:</i> WCAG 3.3.1</p>			
<p><b>9.3</b> O sistema deve fornecer ao usuário uma descrição textual caso algum erro de entrada seja identificado.</p> <p><i>Exemplo:</i> exibir uma mensagem de erro.</p> <p><i>Motivação:</i> mensagens de erro são mais acessíveis se descritivas, pois independente do usuário utilizar ou não recursos de tecnologia assistiva ele consegue perceber/identificar o problema.</p> <p><i>Fontes:</i> WCAG 3.3.3</p>			
<b>Q10) O sistema fornece uma ajuda documentada que pode ser facilmente encontrada em caso de necessidade?</b>	<b>Atende</b>	<b>Não atende</b>	<b>Não se aplica</b>
<p><b>10.1</b> O sistema deve fornecer informações sobre as tarefas que podem ser realizadas.</p> <p><i>Exemplo:</i> mapa do sistema.</p> <p><i>Motivação:</i> quando o sistema web é muito grande os usuários podem precisar de um guia para chegar mais rápido à tarefa ou até mesmo perceber as possíveis tarefas que eles podem realizar.</p> <p><i>Fontes:</i> About this – Montero</p>			
<p><b>10.2</b> O sistema deve fornecer ao usuário informações adicionais sobre o proprietário/desenvolvedor/responsável pelo sistema.</p> <p><i>Exemplo:</i> opção de “fale conosco”.</p> <p><i>Motivação:</i> o usuário se sente mais seguro quando sabe que pode contatar o proprietário ou responsável pelo sistema, por meio de formulários ou <i>links</i> no sistema.</p> <p><i>Fontes:</i> Contact Us – Montero</p>			
<p><b>10.3</b> O sistema deve fornecer um canal simples e pragmático para que o usuário possa informar sobre erros verificados.</p> <p><i>Exemplo:</i> canal de ajuda.</p>			

---

<p><i>Motivação:</i> os usuários precisam se sentir confiantes para reportarem suas experiências e dificuldades para o desenvolvedor web.</p> <p><i>Fontes:</i> Lara, 3.3.11</p>			
<p><b>10.4</b> O sistema deve fornecer recursos interativos para auxiliar o preenchimento de campos de formulário.</p> <p><i>Exemplo:</i> mensagens de texto e voz.</p> <p><i>Motivação:</i> adultos mais velhos geralmente apresentam mais erros durante o preenchimento de formulários, de forma que a página web deve ser capaz de prever suas dificuldades e então auxiliá-los por meio de apresentação de mensagens de texto e voz.</p> <p><i>Fontes:</i> Lara, 3.3.9</p>			

---

## *Critérios de sucesso a partir das demandas de adultos mais velhos*

---

- **Critério de sucesso 1.3.4:** a página web deve oferecer uma funcionalidade que leia o conteúdo por meio de sintetizador de voz.  
Motivação: reduzir o esforço visual e manter o foco da atenção no conteúdo da página.
- **Critério de sucesso 1.4.11:** os principais elementos que compõem a página web devem ser visualizados em destaque, tais como *links*, menus, botões e outros.  
Motivação: adultos mais velhos apresentam dificuldades para identificar e distinguir os componentes da interface de um sistema web e, portanto é necessário que seja oferecido algum meio de destacá-los. Além disso, adultos mais velhos apresentam dificuldades para diferenciar elementos navegacionais do conteúdo da página.
- **Critério de sucesso 2.1.4:** a página web não deve depender do recurso de barras de rolagem.  
Motivação: adultos mais velhos apresentam dificuldades para perceber a existência de barras de rolagem e geralmente não completam a tarefa que requer o uso desse componente da interface.
- **Critério de sucesso 2.1.5:** a página web deve evitar o uso de janelas do tipo pop-up.

Motiva o: adultos mais velhos tendem a apresentar uma rea o de resist ncia e confus o quando uma janela pop-up aparece no meio da tela, pois normalmente acreditam que causaram algum erro no sistema.

- **Cr terio de sucesso 2.2.6**: os menus do tipo *drop-down* devem ser apresentados com velocidade reduzida o suficiente para serem ativados por usu rios que apresentam pouca coordena o motora.

Motiva o: adultos mais velhos apresentam dificuldades na manipula o de menus m veis r pidos e geralmente acabam selecionando a op o errada.

- **Cr terio de sucesso 2.2.7**: conte do multim dia deve disponibilizar op es para controlar a sua execu o e a velocidade de apresenta o.

Motiva o: conte do multim dia afeta o foco de aten o dos adultos mais velhos desviando-a da execu o da tarefa e frequentemente n o   bem compreendido devido   alta velocidade de execu o.

- **Cr terio de sucesso 2.4.11**: n o utilizar nomes id nticos para referenciar URLs (*Uniform Resource Locator*) distintas.

Motiva o: adultos mais velhos se acostumam com a interface apresentada e qualquer renomea o de refer ncias deve ser evitada.

- **Cr terio de sucesso 2.4.12**: *links* adjacentes devem ser apresentados na interface com espa o suficiente entre eles para evitar erros involunt rios.

Motiva o: adultos mais velhos, em especial os idosos, apresentam dificuldades de precis o ao clicar em pequenas  reas de *links*.

- **Cr terio de sucesso 2.4.13**:  cones ou imagens adjacentes devem fazer parte da  rea de ativa o dos *links*.

Motiva o: imagens pequenas ou  cones adjacentes aos *links* confundem os adultos mais velhos, visto que eles tendem a clicar nas imagens ao inv s dos *links*.

- **Cr terio de sucesso 2.4.14**: a p gina web deve diferenciar os *links* visitados dos *links* n o visitados.

Motiva o: adultos mais velhos tendem a esquecer os *links* que eles j  visitaram, devido  s defici ncias cognitivas.

- **Cr terio de sucesso 2.4.15**: a p gina web deve apresentar de forma distinta as op es de menu que j  foram visitadas daquelas que ainda n o foram selecionadas.

Motivação: adultos mais velhos tendem a esquecer quais opções de menu eles já visitaram.

- **Critério de sucesso 2.4.16**: os menus do tipo *drop-down* devem apresentar apenas um nível de opção.

Motivação: adultos mais velhos apresentam grande dificuldade ao interagir com vários níveis de menus do tipo *drop-down*, acionando as opções erradas frequentemente.

- **Critério de sucesso 2.4.17**: as opções intermediárias do menu, quando selecionadas, devem apresentar uma página contendo suas opções subsequentes e elementos alternativos para o seu acionamento, tais como botões e *links*.

Motivação: é comum adultos mais velhos acionarem as opções intermediárias dos menus quando estão tentando alcançar uma opção do último nível.

- **Critério de sucesso 2.4.18**: ícones devem ser facilmente associados com objetos do mundo real.

Motivação: adultos mais velhos tendem a relacionar as informações que visualizam com objetos do mundo real na tentativa de construção de um modelo mental que os façam lembrar a informação ou tarefa em momentos posteriores.

- **Critério de sucesso 2.4.19**: a página deve apresentar meios que permitam ao usuário visualizar um resumo de suas interações anteriores.

Motivação: é importante que os usuários possam rever suas experiências de sucesso como forma de incentivá-los a realizar novas interações, reduzindo a insegurança e a quantidade de erros durante a navegação ou execução de uma tarefa.

- **Critério de sucesso 3.2.6**: a página inicial deve identificar claramente quais partes do sistema web podem ser acessadas com ou sem autenticação.

Motivação: adultos mais velhos tendem a apresentar dificuldades de compreensão do estado do sistema e normalmente confundem mensagens de erro do sistema com as mensagens a respeito da necessidade de autenticação.

- **Critério de sucesso 3.2.7**: a página web não deve apresentar funcionalidades distintas com descrições e títulos já utilizados em contextos diferentes.

Motivação: adultos mais velhos podem encontrar dificuldades para identificar e distinguir corretamente as funcionalidades do sistema.

- **Critério de sucesso 3.2.8**: a página web não deve ter a mesma forma de apresentação para *links* e outros componentes da interface que não aceitem ativação.

Motiva o: adultos mais velhos tendem a apresentar dificuldades na distin o entre componentes que possibilitam e os que n o possibilitam intera o.

- **Cr terio de sucesso 3.3.7**: na p gina web que cont m campos para a entrada de dados, devem ser explicitadas informa es sobre o formato de dado e seu dom nio, ou seja, informa es de como o usu rio deve preencher o campo.

Motiva o: adultos mais velhos apresentam dificuldades no preenchimento de campos de formul rio e necessitam de visualiza o de exemplos e explica es de como completar a tarefa.

- **Cr terio de sucesso 3.3.8**: a p gina web deve fornecer todas as informa es necess rias para a conclus o de tarefas espec ficas.

Motiva o: adultos mais velhos tendem a necessitar de mais tempo que os jovens para realizar uma tarefa complexa, tais como o preenchimento de formul rios, confirma es de pagamentos, entre outras, o que acaba gerando a expira o da sess o do sistema. Dessa forma a p gina web deve apresentar as explica es de como completar a tarefa previamente.

- **Cr terio de sucesso 3.3.9**: a p gina web deve fornecer recursos interativos para auxiliar o preenchimento de campos de formul rio.

Motiva o: adultos mais velhos geralmente apresentam mais erros durante o preenchimento de formul rios, de forma que a p gina web deve ser capaz de prever suas dificuldades e ent o auxili -los por meio de apresenta o de mensagens de texto e voz.

- **Cr terio de sucesso 3.3.10**: a p gina web deve disponibilizar diferentes meios de preenchimento de informa es em campos de formul rios al m do teclado.

Motiva o: adultos mais velhos tendem a apresentar grandes dificuldades na digita o de sequ ncias de n meros pelo teclado.

- **Cr terio de sucesso 3.3.11**: a p gina web deve fornecer um canal simples e pragm tico para que o usu rio informe sobre erros verificados na p gina.

Motiva o: adultos mais velhos precisam se sentir confiantes para reportarem suas experi ncias e dificuldades para o desenvolvedor web.



---

## *Padrões web de Montero: descrição (problema e solução)*

---

- 1) **Welcome**: o usuário necessita saber onde ele está, o que pode fazer e o que ele necessita para visitar o lugar.
  - Forneça um lugar para recepção em que as condições de acesso possam ser avaliadas.
- 2) **Polyglot**: permitir o acesso a todos.
  - Fale a linguagem do usuário e procure “projetar para todos”.
- 3) **Ready**: os usuários que querem visitar um sistema web específico, precisam ter instalado os *plug-ins* necessários.
  - Forneça ferramentas ou informações necessárias para visitar o sistema de modo adequado.
- 4) **Indication**: os usuários necessitam saber onde eles podem ir e o que eles podem fazer a partir do ponto onde estão.
  - Forneça o mecanismo necessário (links significativos) que permita a qualquer usuário se mover de um lugar para outros lugares.
- 5) **Similarity**: quando está navegando pela Internet, o usuário necessita saber se ele está no mesmo sistema web ou não.
  - O sistema deve ser projetado usando os mesmos critérios: cores, fontes, localização de navegação e leiaute.

- 6) **Homepage**: um sistema web é acessado de várias maneiras, entretanto deve haver um ponto de referência, que responda questões como: Quem? O que? Quando? Onde?
  - Forneça uma página inicial onde o usuário se sinta como “se estivesse em casa” e saiba onde ele está.
- 7) **Second Chance**: o usuário quer sentir que está no controle de suas operações.
  - Forneça elementos para desfazer/refazer, voltar e limpar.
- 8) **Form**: o usuário deve fornecer informações.
  - Forneça “brancos” apropriados para serem preenchidos, com indicativo claro e correto de qual informação deve ser fornecida.
- 9) **Busy: downloads** podem demorar muito tempo, criando atrasos significativos ou podem ser completados de modos diferentes.
  - Forneça feedback ao usuário.
- 10) **Polite**: as pessoas diferem amplamente nos termos que utilizam para descrever conceitos.
  - Use uma linguagem apropriada de modo mais simples e clara para o conteúdo do sistema web.
- 11) **Danger**: existe um excesso de *plug-ins*. Não se pode assumir que qualquer um irá tê-los ou usar uma particular configuração do computador.
  - Seja cuidadoso ao usar componentes desorientadores.
- 12) **Tagline**: é necessário saber a proposta do sistema web.
  - Forneça um logotipo ou imagem que identifique o sistema web e seu propósito.
- 13) **About this**: todo sistema web deve fornecer uma maneira fácil de encontrar informações sobre a empresa etc.
  - Inclua, por exemplo, um link para uma seção “sobre o sistema”.
- 14) **Search**: o campo de busca é um dos principais elementos de um sistema web e é essencial que os usuários o utilizem de uma forma fácil e sem esforço.
  - Forneça uma ferramenta de busca na página inicial ou uma página com uma visão geral do sistema (ex: mapa do sistema).
- 15) **Novelty**: usuários gostam de saber se existem novas funcionalidades, promoções, ofertas, notícias no sistema web.
  - Forneça sugestões e novidades do sistema web de uma maneira limpa e intuitiva.



**16) Contact Us:** todo sistema web necessita fornecer um meio de contato.

- Forneça um formulário, um local, ou um link no sistema web onde o usuário possa conseguir informações adicionais sobre o proprietário e os produtos do sistema web.

**17) Secret:** se o usuário fornece informações privadas, ele necessita saber a confiabilidade do sistema.

- Forneça mecanismos de segurança (acesso e privacidade) necessários para proteger o usuário, informando a ele as condições de segurança.

**18) Subscription:** os usuários não querem visitar o sistema web a todo o momento. Eles desejam saber quando novos produtos ou novidades aparecem.

- Forneça um formulário no qual o usuário possa conseguir a informação que ele deseja, automaticamente.

**19) Recognise You:** quando um usuário volta a um sistema web ele necessita saber das ações executadas na sua última visita.

- Guarde informação sobre as ações do usuário, locais visitados, *logins*, documentos salvos, etc. (ex: uso de cookies e links visitados).

**20) Location:** quando um usuário acessa um sistema web, ele precisa saber onde está.

- Forneça informação sobre a localização do usuário no sistema web. Ex: Apresente o rastreamento da navegação do usuário no sistema (ex: breadcrumb).

**21) Colour:** a cor deve ser considerada no início do projeto de um sistema web.

- Forneça a informação usando cores adequadas nas fontes, fundos de tela e imagens.

**22) Size:** balanceamento entre gráficos e tempo real.

- Forneça a informação usando fontes adequadas nas imagens, fundos de tela e páginas.

**23) Print:** a leitura de textos num sistema web é diferente da leitura em textos impressos. A maioria das pessoas lê blocos de texto ao invés de ler palavra por palavra.

- Forneça a informação de diversas maneiras e formatos e dê a possibilidade de imprimir ou salvar documentos grandes.





---

## *Documento de requisitos do sistema AgendAloca*

---

Atualmente há um sistema de Agendamento de Bancas que apoia alunos e professores no agendamento de bancas para conclusão de curso, cujo endereço eletrônico é: <http://agua.intermidia.icmc.usp.br/scc/>. Este sistema foi desenvolvido pelo CMS (*Content Management Systems*) Drupal (<http://www.drupal.org>), adotado principalmente pelo potencial das funcionalidades fornecidas. No entanto, este Sistema não foi desenvolvido a partir de um processo pré-estabelecido e também não foi considerado acessibilidade. Sendo assim, ele está sendo submetido a um processo de reengenharia.

O objetivo da reengenharia é incorporar acessibilidade ao sistema seguindo o Arcabouço do processo de desenvolvimento de sistemas web desenvolvido por Pressman e Lowe (2009). Essa reengenharia servirá para gerar *insights* para a redação do Arcabouço do processo de desenvolvimento de sistemas web acessível, ou seja, Arcabouço por Pressman e Lowe (2009) adaptado para incluir acessibilidade. Além disso, espera-se avaliar e comparar o Sistema de Agendamento de bancas atual com o Sistema acessível a ser desenvolvido.

Para facilitar a reengenharia deste Sistema estão sendo elicitados requisitos com a ajuda de *stakeholders* (alunos e professores) e também por meio do sistema atual, já que o sistema atual não tem documentação. O documento de requisitos do Sistema de Agendamento de Bancas foi especificado de acordo com o modelo proposto por (Turine

& Masiero, 1996). Ele está organizado em três principais Seções, a saber: Visão Geral, Requisitos Funcionais e Requisitos Não Funcionais.

### **A. Visão Geral**

O Sistema de Agendamento de Bancas tem como objetivo principal auxiliar o agendamento de bancas referentes aos trabalhos de conclusão de curso desenvolvidos no Instituto de Ciências Matemáticas e de Computação da Universidade de São Paulo. Os usuários deste sistema são alunos e professores (membros da banca). Os alunos poderão acessar o sistema para agendar sua data e horário de apresentação e os professores poderão acessar o sistema para escolher em quais bancas participarão como avaliadores.

O Sistema de Agendamento de Bancas apresenta três áreas para interação com os usuários:

- a) Área de Administrador do Sistema, responsável por gerenciar as bancas e os usuários do sistema de horários, atribuindo-lhes permissões para o uso do sistema,
- b) Área para Alunos cadastrados no sistema, onde eles podem fazer buscas e registrar banca em data e horário específicos,
- c) Área para Professores cadastrados no sistema, onde eles podem fazer buscas, indicar, modificar e excluir participação em bancas com datas data e horários específicos.

### **B. Requisitos Funcionais**

R1. O Sistema deve ter um usuário Administrador, que será responsável pela criação da categoria de usuários, que são: alunos e professores.

R2. O Sistema deve pedir que no cadastramento de um usuário sejam informados: o número USP, nome completo, email e categoria.

R3. O Sistema deve gerar automaticamente após um novo registro, uma senha que deverá ser enviada ao e-mail do usuário registrado.

R4. O Sistema deve ter controle de acesso aos usuários, a saber: alunos terão domínio sobre o registro de bancas e professores poderão indicar uma ou mais bancas para participar.

R5. O Sistema deve permitir que o Administrador possa registrar, alterar e excluir quaisquer registros contidos no sistema.

R6. O Sistema deve permitir que alunos façam buscas por horários ainda disponíveis antes de registrar a banca.

R7. O Sistema deve exibir os horários disponíveis na forma de calendários semanais, com possibilidade de navegação entre eles.

R8. O Sistema deve pedir que os seguintes dados sejam informados para o registro de bancas: Título do trabalho, Descrição do trabalho, Data, Horário inicial, Horário final, Tipo do trabalho, Área de concentração, Orientador na empresa e/ou Universidade.

R9. O Sistema deve apresentar como opções de registro de bancas os tipos do trabalho, a saber: estágio supervisionado e projeto de graduação.

R10. O Sistema deve apresentar como opções de registro de bancas as áreas de concentração, a saber: arquitetura de computadores, banco de dados/sistemas de informação, computação gráfica/realidade virtual/processamento de imagens, engenharia de software/manutenção/teste/implantação/ desenvolvimento de sistemas, inteligência artificial/processamento de linguagem natural/computação bioinspirada, jogos e entretenimento, sistemas web/interação humano computador/multimídia/computação ubíqua, telecomunicação/computação móvel, redes de computadores/sistemas distribuídos/segurança/infraestrutura, sistemas embarcados.

R11. O Sistema deve permitir que os alunos possam registrar, alterar e excluir a banca desde que não ultrapasse a data limite estipulada pelo Administrador.

R12. O Sistema deve armazenar no momento do registro de uma banca, a data, hora e a pessoa que o efetuou.

R13. O Sistema deve permitir que sejam feitas buscas por bancas já escolhidas por professores ou ainda disponíveis para indicação, filtradas por datas e horários, área de conhecimento, tipo de trabalho.

R14. O Sistema deve exibir as bancas agendadas e/ou disponíveis na forma de calendários semanais, com possibilidade de navegação entre eles.

R15. O Sistema deve possibilitar a exibição de detalhes da banca quando o usuário estiver navegando por ela no calendário, tais como: Título do trabalho, Descrição do trabalho, Data, Horário inicial, Horário final, Tipo do trabalho, Área de concentração, Orientador na empresa e/ou Universidade.

R16. O Sistema deve permitir aos professores que sejam marcadas em uma mesma iteração quantas indicações para participação de bancas forem necessárias.

R17. O Sistema deve permitir que professores possam registrar, alterar e excluir a participação em bancas previamente indicadas, desde que não ultrapasse a data limite estipulada.

R18. O Sistema deve permitir que professores visualizem a agenda pessoal de compromissos de acordo com as reservas indicadas por eles.

R19. O Sistema deve oferecer ajuda detalhada a todos os tipos de usuários, tais como as opções disponíveis a ele a partir do ponto em que ele está.

R20. O Sistema deve oferecer Informação sobre o Sistema a todos os usuários, tais como: versão do Sistema, responsável pelo Sistema, contato do responsável etc.

R21. O Sistema deve ser integrado à Intranet do ICMC.

R22. O Sistema deve emitir ao Administrador um relatório contendo todos os registros contidos no Sistema.

R23. O Sistema deve emitir mensagens de erro que sejam tão explanatórias quanto possíveis para que os usuários fiquem sabendo claramente o que foi feito de errado e como solucionar o problema.

R24. O Sistema deve permitir que todos os usuários possam modificar seus dados pessoais.

R25. O Sistema deve permitir que todos os usuários possam sair do Sistema sempre que desejar.

R26. O Sistema deve permitir que os alunos possam visualizar as informações sobre o trabalho agendado por eles.

R27. O Sistema deve permitir que o administrador cadastre os dias e os horários que estarão disponíveis aos alunos para cadastrarem seus trabalhos.

### **C. Requisitos Não-Funcionais**

#### usabilidade

R28. O Sistema deve oferecer uma interface gráfica baseada em padrões de interface e navegação.

R29. O Sistema deve oferecer visibilidade do estado do sistema, ou seja, deixar claro ao usuário o local por onde ele está navegando [H1].

R30. O Sistema deve oferecer frases e conceitos familiares ao usuário, ao invés de termos orientados ao sistema [H2].

R31. O Sistema deve oferecer ao usuário controle e liberdade para que ele possa sair de um estado indesejado do sistema sem ter que percorrer por um extenso caminho [H3].

R32. O Sistema deve oferecer consistência e padrões nos termos e funcionalidades disponíveis. O usuário não precisa adivinhar que diferentes palavras, situações ou ações significam a mesma coisa [H4].

R33. O Sistema deve oferecer um design cuidadoso que possa prevenir a ocorrência de erros [H5].

R34. O Sistema deve oferecer reconhecimento ao invés de relembração, ou seja, as opções e instruções do sistema deve estar visíveis e facilmente recuperáveis quando necessário [H6].

R35. O Sistema deve oferecer flexibilidade e eficiência de uso, por exemplo, o sistema deve possibilitar, opcionalmente, o uso por linhas de comandos, alternativamente para usuários avançados [H7].

R36. O Sistema deve oferecer estética e design minimalista, ou seja, as informações disponíveis não devem conter informações irrelevantes ou raramente necessárias [H8].

R37. O Sistema deve oferecer o reconhecimento, diagnóstico e correção de erros, ou seja, mensagens de erros devem indicar precisamente o problema e sugerir uma solução construtivamente [H9].

R38. O Sistema deve oferecer ajuda e documentação. Essas informações devem ser de fácil localização, focalizadas na tarefa do usuário e não muito extensas [H10].

### **Legenda para os requisitos de usabilidade**

[H\_] = Heurística de usabilidade de Nielsen e Molich (1990).

#### acessibilidade

R39. O Sistema deve fornecer alternativa textual para conteúdo sonoro e visual [D1-WCAG 1.0].

R40. O Sistema deve garantir a percepção do conteúdo e dos elementos gráficos mesmo quando são vistos sem cores [D2-WCAG 1.0].

R41. O Sistema deve utilizar corretamente marcações HTML com elementos estruturais adequados e folhas de estilo para controlar a apresentação [D3-WCAG 1.0].

R42. O Sistema deve assegurar que as tabelas disponíveis tenham marcações necessárias para poderem ser transformadas de maneira harmoniosa por navegadores acessíveis e outros agentes dos usuários [D5-WCAG 1.0].

R43. O Sistema deve garantir que o sistema seja acessível mesmo quando as tecnologias mais recentes não estejam disponíveis ou tenham sido desativadas [D6-WCAG 1.0].

R44. O Sistema deve possibilitar a interrupção momentânea ou definitiva do movimento, intermitência, transcurso ou atualização automática de objetos [D7-WCAG 1.0].

R45. O Sistema deve assegurar a acessibilidade direta em interfaces integradas ao usuário [D8-WCAG 1.0].

R46. O Sistema deve ser projetado considerando a independência de dispositivos [D9-WCAG 1.0].

R47. O Sistema deve estar apto a atualizações das recomendações do W3C, evitando assim, a utilização de tecnologias obsoletas [D11-WCAG 1.0].

R48. O Sistema deve fornecer informações de contexto e orientações de navegação para auxiliar os usuários a compreender e utilizar o conteúdo do sistema [D12-WCAG 1.0].

R49. O Sistema deve garantir mecanismos de navegação coerentes e sistematizados de modo a aumentar a probabilidade do usuário encontrar o que procura [D13-WCAG 1.0].

R50. O Sistema deve disponibilizar conteúdo de diferentes maneiras sem perder a informação ou estrutura [D1c-WCAG 2.0].

### **Legenda para os requisitos de acessibilidade**

[D\_-WCAG 1.0] = Diretriz do *Web Content Accessibility Guidelines* (WCAG) 1.0 (W3C, 1999)

[D\_\_-WCAG 2.0] = Diretriz do *Web Content Accessibility Guidelines* (WCAG) 2.0 (W3C, 2008)

### Confiabilidade

R51. O sistema deve, em caso de falha, ter capacidade para recuperar os dados perdidos até a última operação que realizou.

R52. O sistema deve fornecer facilidades para a realização de backups dos arquivos do sistema.



*Desempenho*

R53. O tempo de processamento de uma operação de consulta não deve exceder três segundos para uma quantidade inferior a 10 bancas agendadas.

R54. O tempo de resposta para as operações de inserção, alteração e exclusão não deve exceder a três segundos.

*Manutenibilidade*

R55. O sistema deve ser manutenível.

**Referência consultada:**

Apêndice B

PDWAU - Processo de desenvolvimento de sistemas web com usabilidade e acessibilidade.



# *Descrição dos Casos de Uso do sistema AgendAloca*

## 1 Fazer login

<b>Caso de uso:</b>	Fazer <i>login</i>
<b>Ator principal:</b>	Aluno ou Professor
<b>Objetivo no contexto:</b>	Este caso de uso descreve as ações do aluno ou do professor para efetuar o <i>login</i> no sistema.
<b>Precondições:</b>	O usuário necessita estar cadastrado para possuir o nome de usuário e senha.
<b>Acionador:</b>	O usuário decide acessar o sistema.
<b>Identificador(es) dos requisitos satisfeitos:</b>	R04
<b>Cenário:</b>	<ol style="list-style-type: none"> <li>1. Esse caso de uso começa quando o usuário acessa o sistema através do navegador de internet.</li> <li>2. O sistema exibe a página de <i>login</i> com os campos “nome de usuário” e “senha”, um botão “Entrar” e um <i>link</i> “Esqueci a senha”.</li> <li>3. O usuário preenche seu <i>nome de usuário</i> e <i>senha</i> nos campos especificados e pressiona o botão “Entrar”.</li> <li>4. O sistema exibe a página inicial.</li> </ol>
<b>Extensões:</b>	<p><u>Esquecer a senha</u></p> <ul style="list-style-type: none"> <li>• Ver caso de uso “Esquecer nome de usuário e/ou senha</li> <li>• ”</li> </ul> <p><u>Errar nome de usuário ou senha</u></p> <ol style="list-style-type: none"> <li>3.1. O usuário preenche um “nome de usuário” ou “senha” que não conferem e pressiona o botão “Entrar”.</li> <li>3.2. O sistema informa explicitamente sobre o erro ocorrido.</li> <li>3.3. Volta para o passo 3.</li> </ol>
<b>Prioridade:</b>	Baixa
<b>Quando estará disponível:</b>	Primeiro incremento
<b>Frequência de uso:</b>	Baixa

<b>Canal para ator:</b>	<ul style="list-style-type: none"> <li>• Via navegador web baseado em PC e conexão à internet.</li> <li>• Acessibilidade: Via navegador web com leitor de tela.</li> <li>• Acessibilidade: Via teclado.</li> </ul>
<b>Questões em aberto:</b>	

## 2 Esquecer nome de usuário e/ou senha

<b>Caso de uso:</b>	Esquecer nome de usuário e/ou senha
<b>Ator principal:</b>	Administrador, Aluno ou Professor
<b>Objetivo no contexto:</b>	Este caso de uso descreve as ações de algum usuário para recuperar seu nome de usuário e senha.
<b>Precondições:</b>	O usuário necessita estar cadastrado para possuir o nome de usuário e senha.
<b>Acionador:</b>	O usuário decide recuperar seu nome de usuário e/ou senha.
<b>Identificador(es) dos requisitos satisfeitos:</b>	R34
<b>Cenário:</b>	<ol style="list-style-type: none"> <li>1. Esse caso de uso começa quando o usuário acessa o sistema através do navegador de internet.</li> <li>2. O sistema exibe a página de <i>login</i> com os campos nome de usuário e senha e um <i>link</i> “Esqueci a senha”.</li> <li>3. O usuário pressiona o <i>link</i> “Esqueci a senha”.</li> <li>4. O sistema exibe uma caixa de texto para preenchimento do e-mail cadastrado e um botão de confirmação “Recuperar usuário e senha”.</li> <li>5. O usuário preenche a caixa de texto com seu e-mail e pressiona o botão “Recuperar usuário e senha”.</li> <li>6. O sistema envia um e-mail contendo o nome de usuário e senha do usuário ao e-mail especificado.</li> <li>7. O sistema apresenta uma mensagem de sucesso ao usuário.</li> </ol>
<b>Extensões:</b>	<p><b>E-mail não cadastrado</b></p> <ol style="list-style-type: none"> <li>5.1. O usuário preenche o campo “E-mail” com um e-mail não cadastrado e pressiona o botão “Recuperar usuário e senha”.</li> <li>5.2. O sistema exibe explicitamente uma mensagem de erro.</li> <li>5.3. Volta ao passo 5.</li> </ol>
<b>Prioridade:</b>	Baixa
<b>Quando estará disponível:</b>	Primeiro incremento
<b>Frequência de uso:</b>	Baixa
<b>Canal para ator:</b>	<ul style="list-style-type: none"> <li>• Via navegador web baseado em PC e conexão à internet.</li> <li>• Acessibilidade: Via navegador web com leitor de tela.</li> <li>• Acessibilidade: Via teclado.</li> </ul>
<b>Questões em aberto:</b>	<ul style="list-style-type: none"> <li>• E se o usuário não tiver mais acesso ao e-mail cadastrado?</li> <li>• Para o reenvio de senhas vamos ter um campo CAPTCHA como <i>anti-spam</i>?</li> </ul>

## 3 Modificar dados pessoais

<b>Caso de uso:</b>	Modificar dados pessoais
<b>Ator principal:</b>	Administrador, Aluno ou Professor
<b>Objetivo no contexto:</b>	Este caso de uso descreve as ações de algum usuário para modificar seus dados pessoais cadastrados no Sistema.
<b>Precondições:</b>	O usuário necessita estar cadastrado para possuir dados pessoais no sistema.

<b>Acionador:</b>	O usuário decide modificar seus dados pessoais.
<b>Identificador(es) dos requisitos satisfeitos:</b>	R24
<b>Cenário:</b>	<ol style="list-style-type: none"> <li>1. Esse caso de uso começa quando o usuário acessa o sistema através do navegador de internet.</li> <li>2. O sistema exibe a página de <i>login</i> com os campos nome de usuário e senha e um <i>link</i> “Esqueci a senha”.</li> <li>3. O usuário faz o <i>login</i> no sistema.</li> <li>4. O sistema exibe o nome do usuário do lado superior direito da tela com possibilidade de clicar para visualizar seus dados pessoais.</li> <li>5. O usuário pode optar por modificar os seus dados pessoais cadastrados.</li> <li>6. O sistema apresenta os dados do usuário que foram alterados.</li> </ol>
<b>Extensões:</b>	
<b>Prioridade:</b>	Baixa
<b>Quando estará disponível:</b>	Primeiro incremento
<b>Frequência de uso:</b>	Baixa
<b>Canal para ator:</b>	<ul style="list-style-type: none"> <li>• Via navegador web baseado em PC e conexão à internet.</li> <li>• Acessibilidade: Via navegador web com leitor de tela.</li> <li>• Acessibilidade: Via teclado.</li> </ul>
<b>Questões em aberto:</b>	

#### 4 Exibir informações do sistema

<b>Caso de uso:</b>	Exibir informações do sistema
<b>Ator principal:</b>	Aluno, Professor, Administrador
<b>Objetivo no contexto:</b>	Este caso de uso descreve as ações do usuário para consultar as informações sobre o sistema.
<b>Precondições:</b>	
<b>Acionador:</b>	O usuário decide consultar as informações sobre o sistema de marcação de bancas.
<b>Identificador(es) dos requisitos satisfeitos:</b>	R20
<b>Cenário:</b>	<ol style="list-style-type: none"> <li>1. Este caso de uso começa quando o usuário acessa a opção “Sobre” do menu.</li> <li>2. O sistema apresenta na página informações como: versão do sistema, responsável pelo sistema, contato do responsável, etc.</li> </ol>
<b>Extensões:</b>	
<b>Prioridade:</b>	Baixa
<b>Quando estará disponível:</b>	Segundo incremento
<b>Frequência de uso:</b>	Baixa
<b>Canal para ator:</b>	<ul style="list-style-type: none"> <li>• Via navegador web baseado em PC e conexão à internet.</li> <li>• Acessibilidade: Via navegador web com leitor de tela.</li> <li>• Acessibilidade: Via teclado.</li> </ul>
<b>Questões em aberto:</b>	<ul style="list-style-type: none"> <li>• Quais informações serão apresentadas nesta página?</li> </ul>

#### 5 Obter ajuda

<b>Caso de uso:</b>	Obter ajuda
---------------------	-------------

<b>Ator principal:</b>	Aluno, Professor, Administrador
<b>Objetivo no contexto:</b>	Este caso de uso descreve as ações de algum usuário, aluno ou professor, para obter ajuda em determinada operação.
<b>Precondições:</b>	
<b>Acionador:</b>	O usuário decide consultar os tópicos de ajuda da operação que está efetuando.
<b>Identificador(es) dos requisitos satisfeitos:</b>	R19
<b>Cenário:</b>	<ol style="list-style-type: none"> <li>1. Este caso de uso começa quando o usuário necessita de ajuda em determinada operação e pressiona o botão de ajuda presente na página atual.</li> <li>2. O sistema exibe, na mesma página, uma relação de tópicos explanatórios sobre as operações contidas na página atual.</li> </ol>
<b>Extensões:</b>	
<b>Prioridade:</b>	Baixa
<b>Quando estará disponível:</b>	Segundo incremento
<b>Frequência de uso:</b>	Moderada
<b>Canal para ator:</b>	<ul style="list-style-type: none"> <li>• Via navegador web baseado em PC e conexão à internet.</li> <li>• Acessibilidade: Via navegador web com leitor de tela.</li> <li>• Acessibilidade: Via teclado.</li> </ul>
<b>Questões em aberto:</b>	

## 6 Sair do Sistema

<b>Caso de uso:</b>	Sair do Sistema
<b>Ator principal:</b>	Aluno, Professor, Administrador
<b>Objetivo no contexto:</b>	Este caso de uso descreve a ação de qualquer usuário para sair do sistema.
<b>Precondições:</b>	O usuário deve estar logado no sistema.
<b>Acionador:</b>	O usuário decide sair do sistema.
<b>Identificador(es) dos requisitos satisfeitos:</b>	R25
<b>Cenário:</b>	<ol style="list-style-type: none"> <li>1. Este caso de uso começa quando o usuário está logado no sistema.</li> <li>2. O sistema exibe a opção de Sair do sistema no canto superior direito da tela.</li> <li>3. O usuário decide Sair do sistema.</li> <li>4. O sistema responde à interação do usuário, saindo do sistema.</li> </ol>
<b>Extensões:</b>	<u>Confirmação de Saída</u> <ol style="list-style-type: none"> <li>3.1. O usuário pressiona o botão “Sair do sistema” erroneamente.</li> <li>3.2. O sistema pergunta ao usuário se deseja realmente sair do sistema.</li> <li>3.3. O usuário opta por realmente sair do sistema ou cancelar a operação.</li> </ol>
<b>Prioridade:</b>	Baixa
<b>Quando estará disponível:</b>	Segundo incremento.
<b>Frequência de uso:</b>	Baixa
<b>Canal para ator:</b>	<ul style="list-style-type: none"> <li>• Via navegador web baseado em PC e conexão à internet.</li> <li>• Acessibilidade: Via navegador web com leitor de tela.</li> <li>• Acessibilidade: Via teclado.</li> </ul>

<b>Questões em aberto:</b>	
----------------------------	--

## 7 Criar usuário

<b>Caso de uso:</b>	Criar usuário
<b>Ator principal:</b>	Administrador
<b>Objetivo no contexto:</b>	Este caso de uso descreve as ações do administrador para cadastrar um novo usuário no sistema.
<b>Precondições:</b>	
<b>Acionador:</b>	O administrador decide cadastrar um novo professor ou um novo aluno no sistema.
<b>Identificador(es) dos requisitos satisfeitos:</b>	R01, R02, R03, R04, R05
<b>Cenário:</b>	<ol style="list-style-type: none"> <li>1. Este caso de uso começa quando o administrador acessa a página de cadastro de usuário através do botão “Cadastrar usuário” do menu principal.</li> <li>2. O administrador escolhe a opção “Cadastrar um único usuário”.</li> <li>3. O sistema exibe os campos a serem preenchidos: nome completo, número USP, e-mail e categoria (aluno ou professor).</li> <li>4. O administrador preenche os campos necessários.</li> <li>5. O administrador pressiona o botão “Confirmar cadastro”.</li> <li>6. O sistema salva o cadastro.</li> <li>7. O sistema gera uma senha e envia um e-mail de confirmação para o usuário contendo seu nome de usuário e sua senha gerada automaticamente</li> <li>8. O sistema exibe uma mensagem de sucesso ao administrador.</li> </ol>
<b>Extensões:</b>	<p><u>Importar arquivo CSV</u></p> <ol style="list-style-type: none"> <li>2.1. O administrador escolhe a opção “Importar múltiplos usuários de um arquivo CSV”</li> <li>2.2. O sistema exibe uma caixa para seleção do arquivo.</li> <li>2.3. O administrador seleciona o arquivo e pressiona o botão “Importar”.</li> <li>2.4. Volta ao passo 6.</li> </ol> <p><u>Campo em branco</u></p> <ol style="list-style-type: none"> <li>4.1. O administrador deixa um ou mais campos em branco e pressiona o botão de “Confirmar cadastro”.</li> <li>4.2. O sistema exibe uma mensagem de erro próxima aos campos com problema.</li> <li>4.3. Volta ao passo 4.</li> </ol>
<b>Prioridade:</b>	Alta
<b>Quando estará disponível:</b>	Terceiro incremento
<b>Frequência de uso:</b>	Baixa
<b>Canal para ator:</b>	<ul style="list-style-type: none"> <li>• Via navegador web baseado em PC e conexão à internet.</li> <li>• Acessibilidade: Via navegador web com leitor de tela.</li> <li>• Acessibilidade: Via teclado.</li> </ul>
<b>Questões em aberto:</b>	

## 8 Cadastrar datas e horários disponíveis

<b>Caso de uso:</b>	Cadastrar datas e horários disponíveis
<b>Ator principal:</b>	Administrador

<b>Objetivo no contexto:</b>	Este caso de uso descreve a ação do administrador para cadastrar as datas e horários disponíveis para que os alunos possam cadastrar suas apresentações.
<b>Precondições:</b>	O administrador deve estar logado.
<b>Acionador:</b>	O administrador decide cadastrar datas e horários disponíveis.
<b>Identificador(es) dos requisitos satisfeitos:</b>	R27
<b>Cenário:</b>	<ol style="list-style-type: none"> <li>1. Este caso de uso começa quando o administrador acessa a opção cadastrar datas e horários que estarão disponíveis.</li> <li>2. O sistema exibe em forma de calendário todas as datas possíveis.</li> <li>3. O administrador escolhe as datas e posteriormente os horários que estarão disponíveis para que os alunos cadastrem suas apresentações.</li> <li>4. O sistema fixa o calendário nas datas escolhidas pelo administrador.</li> </ol>
<b>Extensões:</b>	8.3
<b>Prioridade:</b>	Baixa
<b>Quando estará disponível:</b>	Terceiro incremento.
<b>Frequência de uso:</b>	Baixa
<b>Canal para ator:</b>	<ul style="list-style-type: none"> <li>• Via navegador web baseado em PC e conexão à internet.</li> <li>• Acessibilidade: Via navegador web com leitor de tela.</li> <li>• Acessibilidade: Via teclado.</li> </ul>
<b>Questões em aberto:</b>	

## 9 Emitir relatório

<b>Caso de uso:</b>	Emitir relatório
<b>Ator principal:</b>	Administrador
<b>Objetivo no contexto:</b>	Este caso de uso descreve as ações do administrador para obter o relatório com os registros contidos no sistema.
<b>Precondições:</b>	O sistema deve possuir algum registro.
<b>Acionador:</b>	O administrador decide consultar os registros do sistema.
<b>Identificador(es) dos requisitos satisfeitos:</b>	R22
<b>Cenário:</b>	<ol style="list-style-type: none"> <li>5. Este caso de uso começa quando o administrador acessa a opção do menu “Emitir relatório”.</li> <li>6. O sistema exibe uma página com as opções de relatório, de filtragem e o botão “Emitir relatório”.</li> <li>7. O administrador escolhe algum filtro para o relatório e pressiona o botão “Emitir relatório”.</li> <li>8. O sistema apresenta o relatório na tela e o botão “Exportar”.</li> </ol>
<b>Extensões:</b>	<p><u>Exportar</u></p> <ol style="list-style-type: none"> <li>8.1 O administrador pressiona o botão “Exportar”.</li> <li>8.2 O administrador escolhe o formato do arquivo e pressiona o botão de confirmação.</li> <li>8.4 O sistema gera o arquivo segundo as escolhas do administrador.</li> </ol>
<b>Prioridade:</b>	Baixa
<b>Quando estará disponível:</b>	Quarto incremento.
<b>Frequência de uso:</b>	Baixa



<b>Canal para ator:</b>	<ul style="list-style-type: none"> <li>• Via navegador web baseado em PC e conexão à internet.</li> <li>• Acessibilidade: Via navegador web com leitor de tela.</li> <li>• Acessibilidade: Via teclado.</li> </ul>
<b>Questões em aberto:</b>	

#### 10 Buscar horários vagos

<b>Caso de uso:</b>	Buscar horários vagos
<b>Ator principal:</b>	Aluno
<b>Objetivo no contexto:</b>	Este caso de uso descreve as ações de um aluno para buscar algum horário vago no calendário.
<b>Precondições:</b>	
<b>Acionador:</b>	O aluno decide buscar algum horário vago para, possivelmente, agendar sua apresentação.
<b>Identificador(es) dos requisitos satisfeitos:</b>	R06, R07
<b>Cenário:</b>	<ol style="list-style-type: none"> <li>1. Este caso de uso começa quando o aluno acessa no menu “Consultar” a opção “Agenda geral”.</li> <li>2. O sistema apresenta um calendário contendo todos os horários vagos disponíveis.</li> </ol>
<b>Extensões:</b>	
<b>Prioridade:</b>	Alta
<b>Quando estará disponível:</b>	Quarto incremento
<b>Frequência de uso:</b>	Alta
<b>Canal para ator:</b>	<ul style="list-style-type: none"> <li>• Via navegador web baseado em PC e conexão à internet.</li> <li>• Acessibilidade: Via navegador web com leitor de tela.</li> <li>• Acessibilidade: Via teclado.</li> </ul>
<b>Questões em aberto:</b>	

#### 11 Agendar data e hora da apresentação

<b>Caso de uso:</b>	Agendar data e hora da apresentação
<b>Ator principal:</b>	Aluno
<b>Objetivo no contexto:</b>	Este caso de uso descreve as ações do aluno para agendar o horário de uma apresentação.
<b>Precondições:</b>	É necessário que haja um ou mais horários vagos para agendar a apresentação.
<b>Acionador:</b>	O aluno decide agendar sua apresentação.
<b>Identificador(es) dos requisitos satisfeitos:</b>	R08, R09, R10, R11, R12
<b>Cenário:</b>	<ol style="list-style-type: none"> <li>1. Este caso de uso começa quando o aluno escolhe um horário disponível no calendário.</li> <li>2. O sistema exibe um formulário contendo os campos: “Título do trabalho”, “Descrição do trabalho”, “Tipo de Trabalho” (verificar requisito R09), “Área de concentração” (verificar requisito R10), “Orientador na empresa e/ou Universidade” e o botão “Agendar apresentação”.</li> <li>3. O aluno preenche todos os campos especificados e pressiona o botão “Agendar apresentação”.</li> </ol>

	4. O sistema salva as alterações e apresenta uma mensagem de sucesso da operação.
<b>Extensões:</b>	<p><u>Data limite extrapolada</u> 2.1. O sistema informa ao aluno que a data limite para inscrição de apresentações expirou e que não é possível agendar mais nenhum trabalho.</p> <p><u>Trabalho já cadastrado</u> 2.1. O aluno já possui um horário agendado pra apresentação do trabalho. 2.2. O sistema apresenta uma mensagem especificando o erro.</p> <p><u>Campos em branco</u> 3.1. O aluno deixa um ou mais campos em branco e pressiona o botão “Agendar apresentação”. 3.2. O sistema exibe mensagens de erro próximas aos campos em branco. 3.3. Volta ao passo 3.</p> <p><u>Agendamento simultâneo</u> 3.1. O aluno pressiona o botão “Agendar apresentação”, porém, antes disso outro aluno agenda sua apresentação no mesmo horário. 3.2. O sistema exibe uma mensagem de erro ao aluno explicitando que o horário da apresentação está ocupado.</p>
<b>Prioridade:</b>	Moderada
<b>Quando estará disponível:</b>	Quarto incremento
<b>Frequência de uso:</b>	Baixa
<b>Canal para ator:</b>	<ul style="list-style-type: none"> <li>• Via navegador web baseado em PC e conexão à internet.</li> <li>• Acessibilidade: Via navegador web com leitor de tela.</li> <li>• Acessibilidade: Via teclado.</li> </ul>
<b>Questões em aberto:</b>	

## 12 Visualizar apresentação agendada

<b>Caso de uso:</b>	Visualizar apresentação agendada
<b>Ator principal:</b>	Aluno
<b>Objetivo no contexto:</b>	Este caso de uso descreve a ação do aluno para visualizar a apresentação do trabalho que foi agendado por ele.
<b>Precondições:</b>	O aluno cadastrou seu trabalho e sua apresentação no sistema.
<b>Acionador:</b>	O aluno decide conferir os dados inseridos na apresentação agendada.
<b>Identificador(es) dos requisitos satisfeitos:</b>	R26
<b>Cenário:</b>	<ol style="list-style-type: none"> <li>1. Este caso de uso começa quando o aluno acessa no menu a opção “Minha apresentação”.</li> <li>2. O sistema apresenta em forma de texto as informações cadastradas pelo aluno.</li> </ol>
<b>Extensões:</b>	<p><u>Alteração das informações:</u> 2.1 O usuário</p>
<b>Prioridade:</b>	Baixa
<b>Quando estará disponível:</b>	Quarto incremento

<b>Frequência de uso:</b>	Moderada
<b>Canal para ator:</b>	<ul style="list-style-type: none"> <li>• Via navegador web baseado em PC e conexão à internet.</li> <li>• Acessibilidade: Via navegador web com leitor de tela.</li> <li>• Acessibilidade: Via teclado.</li> </ul>
<b>Questões em aberto:</b>	

### 13 Buscar trabalhos

<b>Caso de uso:</b>	Buscar trabalhos
<b>Ator principal:</b>	Professor
<b>Objetivo no contexto:</b>	Este caso de uso descreve as ações do professor para buscar informações sobre as apresentações de trabalhos disponíveis.
<b>Precondições:</b>	
<b>Acionador:</b>	O professor decide buscar apresentações para, possivelmente, marcar participação na banca avaliadora, ou simplesmente para visualizar as informações sobre a apresentação ou sobre o trabalho.
<b>Identificador(es) dos requisitos satisfeitos:</b>	R13, R14, R15
<b> Cenário:</b>	<ol style="list-style-type: none"> <li>1. Este caso de uso começa quando o professor acessa o menu “Consultar” e escolhe entre as opções “Todos os trabalhos”, “Tipo de trabalho” e “Área de concentração”.</li> <li>2. O sistema exibe, em formato de calendário, os horários que possuem as apresentações segundo os filtros escolhidos pelo professor.</li> <li>3. O professor pressiona sobre algum dos trabalhos apresentados.</li> <li>4. O sistema apresenta uma página com os detalhes do trabalho.</li> </ol>
<b>Extensões:</b>	<p><u>Filtragem de trabalhos disponíveis</u></p> <ol style="list-style-type: none"> <li>3.1. O professor escolhe em um menu situado acima do calendário a opção de listagem “Apenas disponíveis”.</li> <li>3.2. O sistema apresenta no calendário apenas os trabalhos que ainda estão disponíveis para escolha do professor.</li> <li>3.3. Volta ao passo 3.</li> </ol> <p><u>Pesquisa sem resultados</u></p> <ol style="list-style-type: none"> <li>4.1. Se a pesquisa não retornar nenhum resultado o sistema informa claramente ao professor.</li> </ol>
<b>Prioridade:</b>	Baixa
<b>Quando estará disponível:</b>	Quarto incremento
<b>Frequência de uso:</b>	Moderada
<b>Canal para ator:</b>	<ul style="list-style-type: none"> <li>• Via navegador web baseado em PC e conexão à internet.</li> <li>• Acessibilidade: Via navegador web com leitor de tela.</li> <li>• Acessibilidade: Via teclado.</li> </ul>
<b>Questões em aberto:</b>	

### 14 Marcar participação na apresentação de trabalhos

<b>Caso de uso:</b>	Marcar participação na apresentação de trabalhos
<b>Ator principal:</b>	Professor

<b>Objetivo no contexto:</b>	Este caso de uso descreve as ações do professor para marcar sua participação em uma banca avaliadora para apresentação de um trabalho.
<b>Precondições:</b>	É necessário que haja uma ou mais apresentações agendadas.
<b>Acionador:</b>	O professor decide fazer parte da banca de uma apresentação.
<b>Identificador(es) dos requisitos satisfeitos:</b>	R16, R17
<b>Cenário:</b>	<ol style="list-style-type: none"> <li>1. Este caso de uso começa quando o professor escolhe uma apresentação disponível no calendário.</li> <li>2. O sistema exibe as informações sobre o trabalho escolhido como o “Título do trabalho”, “Descrição do trabalho”, “Data, Horário inicial”, “Horário final”, “Tipo do trabalho”, “Área de concentração”, “Orientador na empresa e/ou Universidade” e um botão “Participar da apresentação”.</li> <li>3. O professor pressiona o botão “Participar da apresentação” para indicar sua participação na apresentação.</li> <li>4. O sistema salva as alterações e apresenta uma mensagem de sucesso da operação.</li> </ol>
<b>Extensões:</b>	<p><u>Data limite extrapolada</u></p> <ol style="list-style-type: none"> <li>1.1. O sistema informa ao professor que a data limite para participar das bancas expirou e que não é possível participar, ou alterar participações, da banca de avaliação de uma apresentação.</li> </ol> <p><u>Agendamento simultâneo</u></p> <ol style="list-style-type: none"> <li>3.1. O professor pressiona o botão “Participar da apresentação”, porém, antes disso outro professor já indicou sua participação na mesma apresentação.</li> <li>3.2. O sistema exibe uma mensagem de erro ao professor explicitando que a participação na apresentação não está mais disponível.</li> </ol> <p><u>Participação já efetuada</u></p> <ol style="list-style-type: none"> <li>4.1. O professor já está participando da banca em questão.</li> <li>4.2. O sistema apresenta uma mensagem de erro explicitando o ocorrido.</li> </ol>
<b>Prioridade:</b>	Moderada
<b>Quando estará disponível:</b>	Quarto incremento
<b>Frequência de uso:</b>	Baixa
<b>Canal para ator:</b>	<ul style="list-style-type: none"> <li>• Via navegador web baseado em PC e conexão à internet.</li> <li>• Acessibilidade: Via navegador web com leitor de tela.</li> <li>• Acessibilidade: Via teclado.</li> </ul>
<b>Questões em aberto:</b>	

## 15 Visualizar agenda de compromissos

<b>Caso de uso:</b>	Visualizar agenda de compromissos
<b>Ator principal:</b>	Professor
<b>Objetivo no contexto:</b>	Este caso de uso descreve as ações do professor para visualizar a agenda com as apresentações em que ele participará.
<b>Precondições:</b>	O professor precisa participar de no mínimo uma banca.
<b>Acionador:</b>	O professor decide consultar as apresentações das quais faz parte da banca.

<b>Identificador(es) dos requisitos satisfeitos:</b>	R18
<b>Cenário:</b>	<ol style="list-style-type: none"> <li>3. Este caso de uso começa quando o professor acessa no menu a opção “Minha agenda”.</li> <li>4. O sistema apresenta, em forma de calendário, as apresentações das quais o professor faz parte da banca examinadora.</li> </ol>
<b>Extensões:</b>	
<b>Prioridade:</b>	Baixa
<b>Quando estará disponível:</b>	Quarto incremento
<b>Frequência de uso:</b>	Moderada
<b>Canal para ator:</b>	<ul style="list-style-type: none"> <li>• Via navegador web baseado em PC e conexão à internet.</li> <li>• Acessibilidade: Via navegador web com leitor de tela.</li> <li>• Acessibilidade: Via teclado.</li> </ul>
<b>Questões em aberto:</b>	

**Glossário:**

CSV: *Comma-separated values*, do Português “Valores separados por vírgula”, é um formato de arquivo utilizado para armazenar dados tabelados suportados pela maioria dos editores de planilhas.



---

***Parecer substanciado e aprovado  
pelo Comitê de Ética EACH-USP***

---

---

A seguir o parecer substanciado e aprovado pelo Comitê de Ética em Pesquisa envolvendo Seres Humanos - Escola de Artes, Ciências e Humanidades EACH-USP pode ser encontrado.

ESCOLA DE ARTES, CIÊNCIAS  
E HUMANIDADES - EACH/USP



**PARECER CONSUBSTANCIADO DO CEP**

**DADOS DO PROJETO DE PESQUISA**

**Título da Pesquisa:** Um método para garantir a acessibilidade no desenvolvimento de sistemas Web

**Pesquisador:** Ana Luíza Dias

**Área Temática:**

**Versão:** 3

**CAAE:** 05700912.5.0000.5390

**Instituição Proponente:** PRÓ-REITORIA DE GRADUAÇÃO DA UNIVERSIDADE DE SÃO PAULO

**DADOS DO PARECER**

**Número do Parecer:** 154.681

**Data da Relatoria:** 27/11/2012

**Apresentação do Projeto:**

O projeto refere-se a uma tese de Doutorado sobre desenvolvimento de sistemas computacionais com características de acessibilidade e usabilidade. A pesquisa específica submetida ao Comitê de Ética consiste em avaliar dois sistemas existentes destinados a agendamentos de banca, a fim de validar o quanto as características de acessibilidade e usabilidade podem apoiar os usuários durante a interação. De acordo com os autores, serão coletados dados referentes a: tempo de execução de tarefas; quantidade de erros na execução de tarefas; severidade dos erros encontrados (baseado na severidade de avaliações heurísticas); quantidade de tarefas concluídas e número de cliques para uma tarefa.

**Objetivo da Pesquisa:**

O objetivo do projeto de Doutorado é formalizar um framework para o desenvolvimento de sistemas Web com acessibilidade e usabilidade. A pesquisa com usuários servirá como base para atingir o objetivo do Doutorado.

**Endereço:** Av. Arlindo Bettio, nº 1000

**Bairro:** Ermelino Matarazzo

**UF:** SP

**Município:** SAO PAULO

**CEP:** 03.828-000

**Telefone:** (11)3091-1046

**E-mail:** cep-each@usp.br



ESCOLA DE ARTES, CIÊNCIAS  
E HUMANIDADES - EACH/USP



**Avaliação dos Riscos e Benefícios:**

De acordo com os autores: "Sabe-se que todas as pesquisas envolvendo seres humanos possuem algum tipo de risco para os participantes. Para este Estudo de Caso, acredita-se que haja dois riscos principais a exposição à radiação e a violação da privacidade. Para amenizar o risco a exposição à radiação, os participantes não ficarão mais do que 4 (quatro) horas em frente ao computador e esse tempo pode ser considerado um tempo desprezível. Já para

amenizar o risco de violação da privacidade dos participantes, os dados coletados durante o experimento não estarão expostos na Web e somente a equipe de pesquisa terá acesso aos dados para análise posterior. Além disso, os dados que serão publicados não identificarão os participantes devido ao uso de siglas e números para se referir a eles."

Assim sendo, acredito que os riscos da pesquisa são mínimos e que estão sendo devidamente tratados pelos pesquisadores.

**Comentários e Considerações sobre a Pesquisa:**

Nesta versão do projeto os pesquisadores deixam explícitos os benefícios da pesquisa:

"Os participantes poderão se sentir beneficiados por participarem da pesquisa, uma vez que eles serão convidados a participar de uma atividade especial em que podem interagir com um sistema que facilitará seu trabalho no final do semestre. Também estarão utilizando uma ferramenta que a instituição tem interesse de utilizar não apenas para esta pesquisa, mas durante o ano letivo. Além disso, sentir-se-ão beneficiados por colaborar com o avanço das pesquisas, levando-os conseqüentemente à satisfação interna e também por saber que pessoas com deficiências serão beneficiadas com os resultados desta pesquisa. Ressalta-se que será informado a todos os participantes que o estudo de caso se deve ao fato de perceber quão usável, fácil e contextualizado é o software/sistema, pelo ponto de vista dos participantes,

Endereço: Av. Arlindo Bettio, nº 1000

Bairro: Ermelino Matarazzo

UF: SP

Município: SAO PAULO

CEP: 03.828-000

Telefone: (11)3091-1046

E-mail: cep-each@usp.br

ESCOLA DE ARTES, CIÊNCIAS  
E HUMANIDADES - EACH/USP



portanto, em

nenhum momento a avaliação será baseada no comportamento dos sujeitos. O grupo de pesquisadores se vê obrigado a suspender a pesquisa imediatamente ao perceber que algum participante da pesquisa está constrangido ou desconfortável, por participar do estudo de caso, bem como se observar algum risco ou dano à saúde do sujeito participante da pesquisa. Adicionalmente, o grupo de pesquisadores se compromete com o Comitê de Ética, informá-lo de todos os efeitos adversos ou fatos relevantes que alterem o curso normal do estudo e cumprir todas as consequências que vierem a ter, assumindo a responsabilidade de dar assistência integral às complicações e danos decorrentes dos riscos previstos."

A partir das afirmações reproduzidas, acredito que a pesquisa contribuirá não somente com os voluntários que executarão o experimento aqui descrito, mas com desenvolvedores que terão subsídios para construir sistemas computacionais mais adequados do ponto de vista de usabilidade e acessibilidade.

Nesta versão do projeto, foram esclarecidos os critérios de inclusão e exclusão de forma adequada: "a seleção dos sujeitos do experimento será feita por conveniência entre alunos e professores por meio de um convite que será enviado por e-mail. Participarão desta pesquisa professores e alunos da Graduação, sendo essa população do Instituto de Ciências Matemáticas e de Computação da Universidade de São Paulo (ICMC/USP), localizado na cidade de São Carlos/SP, público alvo escolhido devido à motivação principal dos sistemas Web desenvolvido, que é apoiá-los durante o final de curso para agendar e organizar as monografias de final de Curso e apresentações.

Se mais de 12 (doze) sujeitos se candidatarem, a equipe de pesquisa dará preferência aos sujeitos com algum tipo de deficiência, os mais velhos, os menos escolarizados e os de classe social mais baixo e que

Endereço: Av. Arlindo Bettio, nº 1000

Bairro: Ermelino Matarazzo

UF: SP

Município: SAO PAULO

CEP: 03.828-000

Telefone: (11)3091-1046

E-mail: cep-each@usp.br

ESCOLA DE ARTES, CIÊNCIAS  
E HUMANIDADES - EACH/USP



tenham menos contato com o computador. Essa prioridade será seguida, pois acredita-se que esses perfis citados poderão utilizar em maior escala as características de acessibilidade e usabilidade presente no sistema. Para a exclusão, os sujeitos que forem mais jovens e também os experientes na área de (IHC) Interação Humano Computador estarão na lista de espera. No entanto, se as vagas não forem preenchidas, a equipe de pesquisa incluirá todos os sujeitos. "

Foram detalhados no projeto os sistemas que serão utilizados, as variáveis a serem coletadas, as tarefas a serem executadas e os questionários que os usuários deverão responder. De acordo com os pesquisadores, a participação do usuário basicamente é descrita como: "...enquanto o usuário estiver interagindo com os sistemas, haverá um programa no computador gravando todos os movimentos do mouse e teclado. Se quiser, durante a interação com os sistemas Web, o usuário também poderá comunicar seus sentimentos, se está gostando ou não, como também dizer tudo o que lhe vem à cabeça, por exemplo, 'gostei', 'não entendi', 'agora ficou mais claro', ou qualquer outro comentário que queira fazer. "

**Considerações sobre os Termos de apresentação obrigatória:**

O TCLE está adequado.

**Recomendações:**

O pesquisador apresentou uma declaração de que o cronograma está adequado.

**Conclusões ou Pendências e Lista de Inadequações:**

Sem pendências, sugerimos a aprovação deste protocolo.

**Situação do Parecer:**

Aprovado

**Necessita Apreciação da CONEP:**

Não

**Considerações Finais a critério do CEP:**

Consideramos o presente protocolo de pesquisa aprovado.

Endereço: Av. Artur de Brito, nº 1000

Bairro: Ermelino Matarazzo

UF: SP

Município: SAO PAULO

CEP: 03.828-000

Telefone: (11)3091-1046

E-mail: cep-each@usp.br

ESCOLA DE ARTES, CIÊNCIAS  
E HUMANIDADES - EACH/USP



SAO PAULO, 26 de Novembro de 2012

---

**Assinador por:**  
**Rosa Yuka Sato Chubaci**  
**(Coordenador)**

Endereço: Av. Arlindo Bettio, nº 1000  
Bairro: Ermelino Matarazzo CEP: 03.828-000  
UF: SP Município: SAO PAULO  
Telefone: (11)3091-1045 E-mail: cep-each@usp.br

## *Aplicação de HEUA no sistema Agendamento*

**Q1) O sistema mantém o usuário sempre informado sobre o que está ocorrendo por meio de *feedbacks* em tempo real?**

- 1.1 Não atende, pois mesmo com a presença do elemento de *breadcrumb* no sistema essa não está corretamente implementada, isso foi notado ao navegar pelo sistema e não visualizar o exato nome das páginas visitadas no *breadcrumb*, e em alguns casos ter as páginas diferentes daquelas visitadas pelo usuário expostas no mesmo.
- 1.2 Atende devido à presença de tais componentes no topo da página na maior parte da interação com o sistema.
- 1.3 Atende devido à presença do atributo lang="pt" na *tag* HTML no código fonte.
- 1.4 Não atende devido ao fato do sistema poder ser acessado sem autenticação e em nenhum momento informar que a mesma é necessária para a realização de qualquer tarefa de edição por exemplo. Outro ponto agravante é o fato do sistema de *login* estar localizado em um menu superior pouco intuitivo o que pode deixar muitos usuários confusos.
- 1.5 Não se aplica, não só pelo fato de que todas as informações requeridas pelo sistema são básicas, mas também pelo fato de que será um administrador que irá realizar o cadastro do usuário, deixando para o usuário apenas a opção de atualizar alguns dados caso necessário.
- 1.6 , 1.7, 1.8, 1.9, 1.10 Não se aplicam devido à ausência de conteúdo multimídia no sistema.

**Q2) O sistema utiliza a linguagem e o modelo mental do usuário com características na interface que são correspondentes ao mundo real?**

2.1 Não atende devido à presença de termos em Inglês como “Calendar” e “Open Panel” e principalmente devido à grande quantidade de termos técnicos da plataforma de desenvolvimento presentes na interface voltada para administradores que caso não ter o domínio sobre a tecnologia encontrarão muitas dificuldades ao manusear o sistema.

2.2 Não atende devido à presença de *links* cujo usuário em um primeiro momento não consegue prever para onde irão se tal *link* for selecionado como a opção “Open Panel”.

2.3 Não atende, pois nas páginas com listas de *links* os mesmos não se encontram totalmente em uma única língua e em todos existe a necessidade de um texto adicional para o correto entendimento de para onde aquele *link* aponta.

2.4 Não atende, pois não existe nenhum mecanismo para identificar as muitas abreviações encontradas principalmente na interface de administrador.

2.5 Não atende devido à presença de rótulos em mais de uma única língua também notamos pouca utilização do componente cabeçalho dentro do sistema.

2.6 Atende, pois apesar dos *links* não possuem uma boa descrição textual, eles possuem representação de objetos do mundo real auxiliando o entendimento do *link*.

2.7 Atende, pois mesmo não sendo muito frequente algumas vezes dentro do sistema encontramos *links* que apontam para informações externas ao sistema onde o usuário pode adquirir maiores informações sobre um determinado assunto.

2.8 Não se aplica devido à ausência de conteúdo multimídia no sistema.

**Q3) O sistema oferece controle e liberdade ao usuário para sair de estados indesejáveis?**

3.1 Não atende, pois o sistema não oferece tal controle para o usuário, sendo necessário acessar novamente a tela de edição caso queira corrigir algum dado incorreto.

3.2 Atende, mas apesar de conseguir operar todas as funcionalidades do sistema não existe o foco na grande maioria dos elementos o que dificulta muito a operação utilizando apenas o teclado.

3.3 Atende, mas apesar de conseguir navegar em todo o sistema não existe o foco da grande maioria dos elementos o que dificulta muito a navegação utilizando apenas o teclado.

3.4 Atende, mas mais uma vez encontramos dificuldades em navegar nas subseções do conteúdo uma vez que alguns elementos não mostram o foco.

- 3.5 Não atende, exigindo que o usuário toda vez percorra todo o conteúdo sequencialmente, mesmo que sejam elementos repetidos como o cabeçalho e o menu, até chegar ao ponto que ele deseja.
- 3.6 Atende, pois é possível navegar sequencialmente pelo sistema sem afetar o significado ou a operação dos elementos.
- 3.7 Não atende, pois foram encontrados diversos problemas relacionados a elementos que fora de ordem sequencial utilizando tabulação.
- 3.8 Não atende uma vez que o idioma é definido para um determinado usuário apenas pelo administrador do sistema, não podendo ser alterado pelo usuário comum do sistema.
- 3.9 Não atende, pois somente em alguns casos relacionados às funções de administrador é exibida uma caixa no canto superior esquerdo contendo um texto em Inglês que redireciona o usuário para a página inicial. Nos demais casos o usuário deve recorrer ao *breadcrumb* para retornar a página inicial.
- 3.10 Não se aplica devido à ausência de conteúdo multimídia no sistema.
- 3.11 Não se aplica devido à ausência de conteúdo em áudio no sistema.
- 3.12 Não se aplica devido à ausência de conteúdo em movimento no sistema.
- 3.13 Não se aplica devido à ausência de atualizações no sistema.

**Q4) O sistema é consistente e segue o mesmo padrão em toda a interface a fim de facilitar o reconhecimento do usuário?**

- 4.1 Não se aplica devido à ausência de conteúdo para download no sistema.
- 4.2 Atende, pois todos os elementos importantes estão em destaque no topo da página.
- 4.3 Atende, pois no sistema não foram encontrados elementos que se distinguem apenas pela cor.
- 4.4 Não atende, pois foram encontrados vários elementos que não exibem o foco durante a seleção pelo teclado.
- 4.5 Atende, pois não foram encontrados problemas com relação ao contexto de um elemento quando o mesmo recebe o foco.
- 4.6 Não atende, pois foram encontrados *links* no sistema que estão representados da mesma forma que textos simples, ou seja, não são sublinhados até que o usuário posicione o mouse sobre o elemento.
- 4.7 Não atende, pois não existe um mecanismo que permita que o usuário identifique a pronúncia de certas palavras.

4.8 Não atende, pois existem elementos no sistema como os *links* que não demonstram serem ativados ou não sem o uso do mouse, dificultado a utilização do sistema caso o mouse não possa ser utilizado.

4.9 Atende, pois se constatou que o código fonte gerado é corretamente formatado.

4.10 Não se aplica devido à ausência de conteúdo multimídia no sistema.

4.11 Não se aplica devido à ausência de conteúdo em áudio no sistema.

**Q5) O sistema apresenta um projeto (design) preventivo e cuidadoso que pode ser capaz de evitar algum problema durante a interação do usuário?**

5.1 Não atende, pois apesar de em alguns elementos existirem máscaras e *labels* de exemplo, ainda existem muitos outros que não possuem e que evitariam entradas incorretas de dados como o campo de e-mail por exemplo.

5.2 Não atende devido à exagerada utilização de textos em negrito e caixas tituladas, elementos que seriam melhores representados através de cabeçalhos.

5.3 Não atende, pois ao realizar a pesquisa o foco se perde exigindo que o usuário selecione novamente o campo utilizado na pesquisa.

5.4 Não se aplica devido à inexistência de informações jurídicas ou de transações.

5.5 Atende devido à existência de uma página de resumo que pode ser visualizada antes de submeter informações para o sistema.

5.6 Não atende, pois além de manter os *links* na mesma cor mesmo após os mesmos terem sido selecionados, muitos *links* não são sublinhados dificultando ainda mais a identificação de *links* selecionados.

5.7 Atende, pois tirando a imagem contendo o nome do sistema presente no cabeçalho principal, não existem outras imagens de texto, inclusive com ícones possuindo conteúdo textual.

5.8 Não atende, pois o texto descrevendo o nome do sistema presente no cabeçalho principal é representado através de uma imagem.

5.9 Não se aplica uma vez que o sistema não possui nenhum tipo de *flash* luminoso.

5.10 Não se aplica uma vez que o sistema não possui nenhum tipo de *flash* luminoso.

5.11 Não atende, pois no sistema existem elementos de caixa de texto que possuem barras de rolagem próprias.

5.12 Atende uma vez que não foram constatadas janelas de pop-up no sistema.

5.13 Não se aplica uma vez que não existem menus do tipo *drop-down* no sistema.



5.14 Não atende devido ao pouco espaço existente entre os *links* do sistema tanto no cabeçalho quanto no conteúdo do sistema.

5.15 Atende uma vez que os ícones e imagens selecionáveis possuem pelo menos a área da imagem para ser selecionada.

5.16 Não atende uma vez que as opções de menu já visitadas são apresentadas da mesma forma que as opções de menu não visitadas.

5.17 Não se aplica uma vez que não existem menus do tipo *drop-down* no sistema.

5.18 Atende embora tal funcionalidade não é constante em todas as opções do sistema, pelo menos nas opções expostas para os administradores, que por sua vez possuem diversas configurações, tal funcionalidade de menu existe auxiliando o usuário caso cometa algum erro.

5.19 Não atende uma vez os dados são perdidos caso ocorra uma queda de conexão ou a sessão autenticada expire.

**Q6) O sistema evita a sobrecarga de memória do usuário fornecendo informações contextuais para cada ação?**

6.1 Não atende uma vez que foi constatado termos com nomes idênticos, mas com mais de uma funcionalidade como, por exemplo: Usuários é utilizado tanto para agrupar todas as operações que podem ser realizadas sobre os usuários como edição das “roles” dos mesmos, quanto para ativar a operação de busca e alteração de um usuário específico.

6.2 Não atende, pois apesar do sistema apresentar praticamente o mesmo leiaute, em algumas opções acessíveis para o administrador o leiaute é completamente diferente tendo, por exemplo, duas colunas de texto ao invés de uma que é o padrão do sistema.

6.3 Atende uma vez que o sistema apresenta uma interface bastante consistente com os elementos principais da navegação sempre nos mesmos lugares.

6.4 Atende uma vez que foi constatado ao menos abas com nomes diferentes dentro de uma página com nomes diferentes para referenciar URLs distintas.

6.5 Não atende uma vez que a opção de ajuda contém conteúdo exclusivo sobre como utilizar as opções presentes na tecnologia e não no sistema, e exclusive somente o usuário que é administrador tem acesso a tais informações.

6.6 Não atende uma vez que para o usuário comum não existe nenhum tipo de ajuda para realizar determinada tarefa, e administradores existe um *link* normalmente no topo do formulário em questão, mas que obriga o usuário parar o que está fazendo consultar a página de ajuda para só então retomar suas atividades.

**Q7) O sistema oferece flexibilidade e eficiência aos usuários, agilizando o uso para usuários experientes e mantendo a facilidade para os novatos?**

7.1 Atende uma vez em todo o sistema não encontramos problemas de insuficiência de tempo para utilizar o conteúdo.

7.2 Não atende uma vez que se o usuário desejar imprimir um conteúdo presente no sistema terá que recorrer às funcionalidades presentes no *browser* que irão imprimir a página toda e não apenas o conteúdo que o usuário deseja.

7.3 Não atende uma vez que existem imagens sem descrição textual como, por exemplo, a imagem com o título do sistema presente no cabeçalho do sistema.

7.4 Não atende uma vez que não existe tal funcionalidade no sistema.

7.5 Não atende uma vez que não existe tal funcionalidade no sistema.

7.6 Não atende uma vez que não existe tal funcionalidade no sistema. Para realizar tal operação o usuário necessita de ferramentas externas ao sistema.

7.7 Atende uma vez que o sistema segue o padrão HTML facilitando a utilização de recursos de tecnologia assistiva.

7.8 Não atende uma vez que o sistema utiliza *JavaScript* para realizar suas operações. Caso o *JavaScript* esteja desabilitado o usuário não conseguirá logar no sistema.

7.9 Atende uma vez que o sistema é apresentado quase que em sua completude por textos o que facilita tanto a leitura por usuários quanto por recursos de tecnologia assistiva de leitura de tela.

7.10 Atende, pois possui uma ferramenta de busca escondida dentro da opção “Open Panel” que apesar de não estar sempre visível possibilita que o usuário a utilize com um atalho para acessar algum conteúdo do sistema.

7.11 Não se aplica uma vez que os usuários irão acessar o sistema esporadicamente, não havendo a necessidade de enviar novidades para o usuário uma vez que elas quase não irão existir.

7.12 Não atende uma vez que o sistema não possui tal funcionalidade, exigindo do usuário a utilização de ferramentas externas ao sistema.

7.13 Não se aplica uma vez que o sistema não limita a visualização de conteúdo por tempo.

7.14 Não se aplica uma vez que o sistema não possui conteúdo no formato de vídeos.

**Q8) O sistema oferece estética e projeto (design) minimalista, mantendo apenas informações úteis, diretas e claras?**

8.1 Atende uma vez que é possível constatar uma consistência na escolha das cores para o sistema, utilizando o cinza escuro, o branco e o laranja para proporcionar um bom contraste.

8.2 Atende uma vez que possui um tamanho de fonte legível para estudantes e professores, que são o público alvo do sistema.

8.3 Não atende uma vez que o sistema não informou o usuário sobre a falta de alguma tecnologia para o correto funcionamento do mesmo. Isso pode ser verificado quando os scripts foram desligados no *browser*.

8.4 Não se aplica devido ao propósito do sistema não contemplar a apresentação de novidades para o usuário.

8.5 Não atende uma vez que o sistema possui páginas cujos blocos de textos excedem os 80 caracteres.

**Q9) O sistema ajuda o usuário a reconhecer, diagnosticar e corrigir erros?**

9.1 Não atende uma vez que o sistema não exibiu nenhuma mensagem ao usuário informando que está faltando alguma tecnologia para o correto funcionamento do mesmo.

9.2 Não atende, pois diversas vezes foi possível verificar mensagens de erro genéricas e inclusive mensagens não tratadas para o usuário contendo erros de código. Algumas vezes mensagens de erro não eram apresentadas.

9.3 Atende, pois todas as mensagens de erro são mostradas através de uma mensagem dentro de uma caixa vermelha no canto superior do contexto do sistema.

**Q10) O sistema fornece uma ajuda documentada que pode ser facilmente encontrada em caso de necessidade?**

10.1 Não atende uma vez que o sistema possui uma sessão denominada “ajuda” que contém somente informações a respeito das operações referentes à tecnologia e além do mais apenas o administrador pode ter acesso a essa sessão. Aos demais usuários não são disponibilizados nenhum tipo de ajuda para a realização das operações do sistema.

10.2 Não atende uma vez que não foi encontrado nenhum tipo de informação desse tipo dentro do sistema.

10.3 Não atende uma vez que não foi encontrada nenhuma opção que permitisse que os usuários entrassem em contato com os desenvolvedores para reportar possíveis erros e melhorias no sistema.

10.4 Não atende uma vez que o sistema nativamente permite apenas a inserção de conteúdo através do teclado e mouse, sendo necessária a utilização de ferramentas externas para realizar essa atividade.



## *Aplicação de HEUA no sistema AgendAloca*

**Q1) O sistema mantém o usuário sempre informado sobre o que está ocorrendo por meio de *feedbacks* em tempo real?**

1.1 Atende devido à presença do *breadcrumb* no sistema.

1.2 Atende devido à presença de tais componentes no topo da página independente de qualquer interação com o sistema.

1.3 Atende devido à presença do atributo lang="pt" na *tag* HTML no código fonte.

1.4 Não se aplica devido ao fato do sistema não possuir partes acessíveis sem autenticação e mesmo depois de logado o sistema só permite que usuário interaja com aquilo que ele tem permissão para fazer.  
OBS: Se o perfil do usuário é professor e ele tenta marcar participação em uma apresentação ele é direcionado para a mesma tela de cadastro na participação realizadas por alunos. Nesse caso o sistema exibe uma mensagem de acesso negado. Entretanto estou levando em consideração esse caso como um *bug* do sistema, logo acredito que esse requisito ainda não é aplicável.

1.5 Não se aplica, não só pelo fato de que todas as informações requeridas pelo sistema são básicas, mas também pelo fato de que será um administrador que irá realizar o cadastro do usuário, deixando para o usuário apenas a opção de atualizar alguns dados caso necessário.

1.6, 1.7, 1.8, 1.9, 1.10 Não se aplicam devido à ausência de conteúdo multimídia no sistema.

**Q2) O sistema utiliza a linguagem e o modelo mental do usuário com características na interface que são correspondentes ao mundo real?**

2.1 Atende, pois de modo geral a linguagem utilizada no sistema é semelhante aos demais sistemas de agenda e se adequa ao dia-a-dia dos usuários alvo do sistema que são professores e alunos, logo o usuário não encontrará muitas dificuldades.

OBS: o único ponto destoante seria a presença do *label* para o botão “criar novo” que apensar de estar de acordo com o contexto de criar novos usuários e horários, não está de acordo com o contexto de marcar apresentações, tarefa realizada pelo aluno.

2.2 Atende uma vez que existem poucos *links* textuais e estes estão bem claros.

2.3 Atende uma vez que a lista de *links* na página “*help*” está bastante clara e organizada.

2.4 Não atende, pois mesmo existindo um número reduzido de abreviações, e estão presentes no dia-a-dia dos usuários não existe nenhum mecanismo para identificar estas e futuras abreviações.

2.5 Atende, pois não foi encontrado nenhum problema de entendimento com relação aos cabeçalhos e rótulos.

2.6 Não atende devido aos fatos dos ícones de acessibilidade presentes ao longo de todo o sistema, não são facilmente associados a nenhum objeto do mundo real.

2.7 Não atende, pois na sessão de ajuda que é onde se encontra os textos com alguma complexidade não existe nenhuma referência a um conteúdo externo que de certa forma poderia auxiliar na elaboração de textos mais sucintos.

2.8 Não se aplica devido à ausência de conteúdo multimídia no sistema.

**Q3) O sistema oferece controle e liberdade ao usuário para sair de estados indesejáveis?**

3.1 Não atende, pois o sistema não oferece tal controle para o usuário, sendo necessário acessar novamente a tela de edição caso queira corrigir algum dado incorreto.

3.2 Atende, pois não foram encontrados problemas ao operar as funcionalidades de sistema.

3.3 Atende, pois não foram encontrados problemas ao navegar pelo sistema.

3.4 Atende, pois não foram encontrados problemas ao navegar pelas sub sessões de conteúdo do sistema.

3.5 Não atende, exigindo que o usuário toda vez percorra todo o conteúdo sequencialmente, mesmo que sejam elementos repetidos como o cabeçalho e o menu, até chegar ao ponto que ele deseja.

3.6 Atende, pois não foram encontrados problemas ao navegar sequencialmente pelo sistema sem afetar o significado ou a operação dos elementos.

3.7 Atende, pois não foram encontrados problemas ao navegar sequencialmente pelo sistema utilizando tabulação.

3.8 Não atende, pois apensar de existir esta opção em todas as telas através de um ícone com a bandeira do país da língua, não é todo o conteúdo do sistema que é transformado para a nova língua, como por exemplo, a página de ajuda.

3.9 Atende sendo necessário selecionar o cabeçalho da página como ocorre em demais sistemas da web.

3.10 Não se aplica devido à ausência de conteúdo multimídia no sistema.

3.11 Não se aplica devido à ausência de conteúdo em áudio no sistema.

3.12 Não se aplica devido à ausência de conteúdo em movimento no sistema.

3.13 Não se aplica devido à ausência de atualizações no sistema.

**Q4) O sistema é consistente e segue o mesmo padrão em toda a interface a fim de facilitar o reconhecimento do usuário?**

4.1 Não se aplica devido à ausência de conteúdo para download no sistema.

4.2 Atende, pois todos os elementos importantes, com exceção do menu lateral estão em destaque no topo da página.

4.3 Atende, pois mesmo na troca de perfil que possui um maior enfoque na mudança de cor da interface, é exibida no topo da tela o novo perfil no qual o usuário se encontra. No restante do sistema não foram encontrados elementos que se distinguem apenas pela cor.

4.4 Atende, pois não foram encontrados problemas com relação ao foco dos elementos durante a seleção pelo teclado.

4.5 Atende, pois não foram encontrados problemas com relação ao contexto de um elemento quando o mesmo recebe o foco.

4.6 Atende, pois apenas os *links* são sublinhados dentro do sistema.

4.7 Não atende, pois não existe um mecanismo que permita que o usuário identifique a pronúncia de certas palavras.

4.8 Atende, pois não foram encontrados problemas com relação à alteração do contexto de modo inesperado sem a ativação prévia de um elemento.

4.9 Atende, pois constatou-se que o código fonte gerado é corretamente formatado.

4.10 Não se aplica devido à ausência de conteúdo multimídia no sistema.

4.11 Não se aplica devido à ausência de conteúdo em áudio no sistema.

**Q5) O sistema apresenta um projeto (design) preventivo e cuidadoso que pode ser capaz de evitar algum problema durante a interação do usuário?**

5.1 Não atende, pois apesar de em alguns elementos existirem máscaras e *labels* de exemplo, ainda existem muitos outros que não possuem e que evitariam entradas incorretas de dados como campos de horário e e-mail.

5.2 Atende devido à clara presença dos cabeçalhos no sistema.

5.3 Não atende, pois ao realizar a pesquisa o foco se move para o primeiro campo da pesquisa, mesmo que a mesma tenha sido realizada utilizando um segundo campo logo abaixo.

5.4 Não se aplica devido a inexistência de informações jurídicas ou de transações.

5.5 Não atende devido a inexistência de tal funcionalidade antes de submeter informações para o sistema.

5.6 Não atende, pois apensar de ocorrer tal diferenciação em todo o sistema na página de ajuda, no cabeçalho tal comportamento não é reproduzido, mantendo os *links* na mesma cor mesmo após os mesmos terem sido selecionados.

5.7 Atende, pois não foram encontradas imagens de texto no sistema com exceção de ícones como o logotipo do ICMC e os ícones da W3C e do DaSilva.

5.8 Não atende, pois foram encontradas imagens de texto nos ícones como o logotipo do ICMC e os ícones da W3C e do DaSilva.

5.9 Não se aplica uma vez que o sistema não possui nenhum tipo de *flash* luminoso.

5.10 Não se aplica uma vez que o sistema não possui nenhum tipo de *flash* luminoso.

5.11 Atende, pois no sistema não existe nenhuma barra de rolagem com exceção da do *browser*.

5.12 Atende uma vez que não foram constatadas janelas de pop-up no sistema.

5.13 Atende uma vez que foi constatado um atraso significativo na exibição de menus do tipo *drop-down*.

5.14 Não atende devido ao pouco espaço existente entre os *links* do sistema tanto no cabeçalho quanto no rodapé.

5.15 Atende uma vez que os ícones e imagens selecionáveis possuem pelo menos a área da imagem para ser selecionada.



5.16 Não atende uma vez que as opções de menu já visitadas são apresentadas da mesma forma que as opções de menu não visitadas.

5.17 Atende uma vez que os menus do sistema apresentam apenas um nível.

5.18 Não atende uma vez que tal funcionalidade não foi constatada no sistema.

5.19 Não atende uma vez os dados são perdidos caso ocorra uma queda de conexão ou a sessão autenticada expire.

**Q6) O sistema evita a sobrecarga de memória do usuário fornecendo informações contextuais para cada ação?**

6.1 Atende uma vez que foram constatados termos idênticos nos menus, na ajuda e nos títulos das páginas.

6.2 Atende uma vez que o estilo é o mesmo independente do contexto do sistema.

6.3 Atende uma vez que o sistema apresenta uma interface bastante consistente com os elementos principais da navegação sempre nos mesmos lugares.

6.4 Atende uma vez que foi constatado páginas com nomes diferentes para referenciar URLs distintas.

6.5 Atende uma vez que na opção de ajuda existem textos descrevendo em detalhes e passo-a-passo as tarefas específicas do sistema.

6.6 Não atende uma vez que o usuário precisar parar o que está fazendo consultar a página de ajuda para só então retomar suas atividades.

**Q7) O sistema oferece flexibilidade e eficiência aos usuários, agilizando o uso para usuários experientes e mantendo a facilidade para os novatos?**

7.1 Atende uma vez em todo o sistema não encontramos problemas de insuficiência de tempo para utilizar o conteúdo.

7.2 Não atende uma vez que se o usuário desejar imprimir um conteúdo presente no sistema terá que recorrer às funcionalidades presentes no *browser* que irão imprimir a página toda e não apenas o conteúdo que o usuário deseja.

7.3 Atende todos os elementos não textuais apresentam descrições textuais quando selecionados.

7.4 Atende uma vez que permite que o usuário altere o contraste da página caso necessário.

7.5 Atende uma vez que permite que o usuário altere o contraste da página caso necessário.

7.6 Atende uma vez que existe uma opção no sistema que possibilita o a alteração no tamanho da fonte de todos os textos do sistema. Tal funcionalidade não se aplica

para as imagens de texto uma vez que as mesmas são praticamente ícones para algum tipo de conteúdo externo.

7.7 Atende uma vez que o sistema segue ao máximo o padrão HTML sendo facilitando a utilização de recursos de tecnologia assistiva.

7.8 Não atende uma vez que o sistema utiliza *JavaScript* para realizar suas operações.

7.9 Atende uma vez que o sistema é apresentado quase que em sua completude por textos o que facilita tanto a leitura por usuários quanto por recursos de tecnologia assistiva de leitura de tela.

7.10 Atende, pois possui uma ferramenta de busca logo abaixo do cabeçalho da página que o usuário pode utilizar com o atalho para acessar algum conteúdo no contexto atual do sistema.

7.11 Não se aplica uma vez que o usuário irão acessar o sistema esporadicamente, não havendo a necessidade de enviar novidades para o usuário uma vez que elas quase não irão existir.

7.12 Não atende uma vez que o sistema não possui tal funcionalidade, exigindo do usuário a utilização de ferramentas externas ao sistema.

7.13 Não se aplica uma vez que o sistema não limita a visualização de conteúdo por tempo.

7.14 Não se aplica uma vez que o sistema não possui conteúdo no formato de vídeos.

**Q8) O sistema oferece estética e projeto (design) minimalista, mantendo apenas informações úteis, diretas e claras?**

8.1 Atende uma vez que as cores principais do sistema são azul e branco que cansam menos a visão do usuário e proporcionam um bom contraste.

8.2 Atende devido ao tamanho padrão da fonte ser legível e o sistema também permite que o usuário adeque o tamanho da fonte para aquela que mais o agrada.

8.3 Atende uma vez que o sistema informa o usuário caso esteja faltando alguma tecnologia para o correto funcionamento do mesmo. Isso pode ser verificado na mensagem mostrada quando os scripts foram desligados no *browser*.

8.4 Não se aplica devido ao propósito do sistema não contemplar a apresentação de novidades para o usuário.

8.5 Não atende uma vez que na página de ajuda mesmo aumentando a fonte no máximo os blocos de textos excedem os 80 caracteres.

**Q9) O sistema ajuda o usuário a reconhecer, diagnosticar e corrigir erros?**

9.1 Atende uma vez que o sistema informa o usuário com uma mensagem no canto superior da tela caso esteja faltando alguma tecnologia para o correto funcionamento do mesmo.

9.2 Atende, pois nas diversas mensagens de erro testadas todas retornaram mensagens bem específicas sobre qual foi o problema ocorrido.

9.3 Atende, pois todas as mensagens de erro são mostradas através de uma mensagem dentro de uma caixa vermelha no canto superior do contexto do sistema.

**Q10) O sistema fornece uma ajuda documentada que pode ser facilmente encontrada em caso de necessidade?**

10.1 Atende uma vez que o sistema possui uma sessão denominada “ajuda” que tem o intuito de descrever como realizar cada tarefa dentro do sistema.

10.2 Atende uma vez que o sistema possui uma sessão denominada “sobre o sistema” que tem o intuito de exibir o e-mail do responsável pelo sistema.

10.3 Atende uma vez que na sessão denominada “sobre o sistema” o usuário tem acesso ao e-mail do responsável para juntamente reportar possíveis erros e melhorias no sistema.

10.4 Não atende uma vez que o sistema nativamente permite apenas a inserção de conteúdo através do teclado e mouse, sendo necessária a utilização de ferramentas externas para realizar essa atividade.



