**UNIVERSIDADE DE SÃO PAULO
INSTITUTO DE FÍSICA
INSTITUTO DE QUÍMICA
INSTITUTO DE BIOCÊNCIAS
FACULDADE DE EDUCAÇÃO**

MARCOS MATSUKUMA

**PRODUÇÃO E AVALIAÇÃO DE INTERFACES, SEGUNDO
PRINCÍPIOS DA INTERAÇÃO HUMANO-COMPUTADOR**

**São Paulo
2012**

MARCOS MATSUKUMA

**PRODUÇÃO E AVALIAÇÃO DE INTERFACES, SEGUNDO
PRINCÍPIOS DA INTERAÇÃO HUMANO-COMPUTADOR**

Dissertação de Mestrado apresentada ao
Programa de Pós-Graduação Interunidades
em Ensino de Ciências para obtenção do
título de Mestre em Ensino de Ciências.

Área de concentração: Ensino de Física

Orientador: Prof. Dr. Mikiya Muramatsu

São Paulo

2012

Autorizo a reprodução e divulgação total ou parcial deste trabalho, por qualquer meio convencional ou eletrônico, para fins de estudo e pesquisa, desde que citada a fonte.

FICHA CATALOGRÁFICA

Preparada pelo Serviço de Biblioteca e Informação
do Instituto de Física da Universidade de São Paulo

Matsukuma, Marcos

Produção e avaliação de interfaces, segundo princípios da
interação humano-computador. - São Paulo, 2012.

Dissertação (Mestrado) – Universidade de São Paulo.
Faculdade de Educação, Instituto de Física, Instituto de Química
e Instituto de Biociências

Orientador: Prof. Dr. Mikiya Muramatsu

Área de Concentração: Ensino de Ciências

Unitermos: 1. Usabilidade de Software; 2. Interação
Usuário-Computador; 3. Tecnologia da Comunicação;
4. Tecnologia da Informação; 5. Física – Estudo e Ensino.

USP/IF/SBI-075/2012

FOLHA DE APROVAÇÃO

Nome: MATSUKUMA, Marcos

Título: Produção e avaliação de interfaces, segundo princípios da Interação Humano-Computador

Dissertação de Mestrado apresentada ao Programa de Pós-Graduação Interunidades em Ensino de Ciências para obtenção do título de Mestre em Ensino de Ciências.

Aprovado em: ____/____/____

Banca Examinadora

Prof. Dr.: _____ Instituição: _____

Julgamento: _____ Assinatura: _____

Prof. Dr.: _____ Instituição: _____

Julgamento: _____ Assinatura: _____

Prof. Dr.: _____ Instituição: _____

Julgamento: _____ Assinatura: _____

Prof. Dr.: _____ Instituição: _____

Julgamento: _____ Assinatura: _____

Prof. Dr.: _____ Instituição: _____

Julgamento: _____ Assinatura: _____

DEDICATÓRIAS

À Dirce Pranzetti, pelo apoio, pelo incentivo e por sempre ter acreditado no meu trabalho.

Aos meus pais, Emiko e Kasuyoshi, e aos meus irmãos, Ciro, Marcello e Yumi, que possibilitaram meus primeiros passos para essa longa caminhada que me trouxe até esse ponto da jornada.

A todos aqueles que lutam diária e incansavelmente para promover o diálogo, o exercício da tolerância e a democratização de todas as formas de conhecimento.

AGRADECIMENTOS

Ao Prof. Dr. Mikiya Muramatsu, pela confiança depositada em mim e em meu trabalho.

Ao Prof. Dr. Ernst W. Hamburger, pelos anos de ensinamento sobre divulgação científica e por ter apoiado a criação do Laboratório Virtual.

À Profa. Amélia Império Hamburger (*in memoriam*), pelas várias conversas inspiradoras.

À Flora Paul, por ter me ajudado na fase final da conclusão desse trabalho.

À Andrea Boneti, à Camila Denleschi, à Desiree Della Monica, ao Duílio Gentilli, ao Eliano Marques, ao Felipe Bondezan, ao Fernando Goldenstein, ao Chico Bryan, à Isabela Gonçalves, à Juliana Parollo, à Lívia Sena, à Mariana Blatner, ao Rodrigo da Silva e ao Ronaldo Montalvão. Meus agradecimentos aos estagiários que fizeram parte da equipe do Laboratório Virtual da Estação Ciência / USP.

Aos meus colegas de pós-graduação, em especial ao Edson Cesar e ao Marcelo Pereira, pelos diálogos e pelos trabalhos inspiradores feitos em conjunto.

Ao Prof. Dr. Luís Carlos de Menezes, ao Prof. Dr. Manoel Oriosvaldo de Moura, ao Prof. Dr. Osvaldo Frota Pessoa Junior, à Profa. Dra. Stela Conceição Bertholo Piconez e à Profa. Dra. Silvia de Mattos Gasparian Colello, pelos melhores momentos oferecidos nesse curso.

À Comissão de Pós-Graduação Interunidades, pela oportunidade de realização do meu curso de pós-graduação.

RESUMO

MATSUKUMA, M. **Produção e avaliação de interfaces, segundo princípios da Interação Humano-Computador**. 2012. 191 f. Dissertação (Mestrado). Faculdade de Educação, Instituto de Física, Instituto de Química e Instituto de Biociências, Universidade de São Paulo, 2012.

Com a expansão do uso de ferramentas tecnológicas, principalmente representadas pelo computador, por seus programas e pelo acesso à Internet, percebe-se a sua presença cada vez maior em diferentes setores da sociedade, inclusive, no meio educacional. No entanto, por ainda ser uma novidade no ambiente tradicional de ensino, essas ferramentas podem ser utilizadas de modo descontextualizado, não favorecendo seu uso correto na aprendizagem. Portanto, é necessário que as políticas de acesso a esses recursos informáticos sejam bem coordenadas para permitir que os usuários, principalmente professores e alunos, possam contar com uma infraestrutura que realmente possibilite o apoio ao ensino. Por outro lado, as interfaces interativas também devem ser desenvolvidas para que sejam utilizadas de modo adequado pelo seu público-alvo. Dessa forma, procurando contribuir para avaliação dos objetos de aprendizagem nos meios virtuais, essa pesquisa apresenta o estudo da construção de interfaces interativas centrada no usuário, apoiando-se em propostas originadas da Interação Humano-Computador (IHC). Assim, partindo do princípio de que o indivíduo é parte integrante e mais importante dos sistemas computacionais, o presente trabalho fez com que a investigação do uso das interfaces pelo usuário interferisse em seu próprio processo de produção. Para tanto, utilizou-se a técnica de observação indireta denominada *Log* de dados, ou seja, a coleta remota de informações e sua gravação em banco de dados, para posterior resgate e análise das interações. A obtenção contínua de dados coletados permitiu fazer um acompanhamento detalhado da utilização do *software*, comprovando a pertinência do método. O resultado mais importante da avaliação, derivado desse processo, foi verificar que, apesar de orientações presentes na interface, as ações executadas pelo usuário se basearam mais na distribuição dos elementos no *software* do que nas informações ali presentes. Assim, percebeu-se a importância da produção prévia de protótipos para que situações de Usabilidade fossem avaliadas, permitindo a obtenção de um produto final mais adequado ao usuário do ambiente virtual de ensino e aprendizagem.

Palavras-chave: Interação Humano-Computador. Produção de *Software*. Usabilidade. Objetos de Aprendizagem. Ensino de Física.

ABSTRACT

MATSUKUMA, M. **Production and evaluation of interfaces, according to the principles of Human-Computer Interaction**. 2012. 191 f. Dissertation (Master's degree). Faculdade de Educação, Instituto de Física, Instituto de Química e Instituto de Biociências, Universidade de São Paulo, 2012.

In the increase of the use of technological tools, mainly represented by the computer, its applications and the access to the Internet, it's observed by its growing presence in different sectors of society, including the educational environment. However, since it still is a novelty in the traditional teaching environment, these tools can be used in a non-contextualized form, not allowing its correct use in the learning process. Therefore, it is necessary that the access policies to these computer resources are well coordinated so the users, especially teachers and students, can count on an infrastructure that really allows the support for the true learning. On the other hand, interactive interfaces should also be developed to be used properly by the target audience. Thus, to contribute to the evaluation of the learning objects in virtual environments, this research presents a study of interactive interfaces built centered on the user, supported by the proposals originated by the Human-Computer Interaction (HCI). Assuming that the person is participative and the most important part of computer systems, this work researches how the users interactions with the interface interferes in its own production process. For this purpose, an indirect observation technique named Logging was used, that is, the remote collection of information and its recording in a database for later recovery and analysis of the interactions. The continuous collection of data allowed a detailed monitoring of the software usage, proving the utility of the method. The most important result of the evaluation, derived by this process, was verifying that, in spite the orientations at the interface, the actions taken by the users were based more on the distribution of the graphic elements in the software than by the orientation displayed. Thus, the research verified the importance of pre-produced prototypes to evaluate Usability situations, allowing the final product to be best suited for the user of the virtual environment for teaching and learning.

Keywords: Human-Computer Interaction. Software Production. Usability. Learning Objects. Teaching of Physics.

LISTA DE FIGURAS

Figura 1 –	Um modelo simples de <i>design</i> de interação.....	37
Figura 2 –	O modelo Cascata de Ciclo de Vida do desenvolvimento de <i>software</i>	38
Figura 3 –	O modelo Espiral de Ciclo de Vida do desenvolvimento de <i>software</i>	39
Figura 4 –	O modelo de Ciclo de Vida Estrela	41
Figura 5 –	O Ciclo de Vida da Engenharia de Usabilidade	42
Figura 6 –	O Ciclo de Vida da Engenharia de Usabilidade	43
Figura 7 –	Áreas de trabalho do programa <i>Adobe Flash</i>	64
Figura 8 –	A estrutura do Laboratório Virtual	65
Figura 9 –	<i>Print screen</i> da página do Laboratório Virtual	66
Figura 10 –	Quadros que compõem o <i>website</i> Laboratório Virtual	67
Figura 11 –	Menu do <i>website</i>	68
Figura 12 –	<i>Layout</i> líquido	69
Figura 13 –	<i>Print screen</i> da interface de animação Reflexão	72
Figura 14 –	<i>Print screen</i> da interface de animação Refração	74
Figura 15 –	<i>Print screen</i> da interface de animação Lentes	75
Figura 16 –	<i>Print screen</i> da interface de animação Espelhos esféricos	77
Figura 17 –	<i>Print screen</i> da interface de animação sobre a formação da imagem no olho ...	78
Figura 18 –	<i>Print screen</i> da interface de animação Formação da imagem no olho	79
Figura 19 –	<i>Print screen</i> da interface de animação Miopia	81
Figura 20 –	<i>Print screen</i> da interface de animação Hipermetropia	82
Figura 21 –	<i>Print screen</i> da interface de animação Correção de defeitos da visão	83
Figura 22 –	Feixes da lente divergente	84
Figura 23 –	Feixes da lente convergente	84
Figura 24 –	Correção da miopia e da hipermetropia	84

Figura 25 – Lentes trocadas. Sem correção	84
Figura 26 – <i>Print screen</i> da interface de animação Cores – cores primárias de luz	86
Figura 27 – Misturando cores primárias de luz: ao deslocar um dos círculos de cor para o centro, o círculo negro vai recebendo a luz da cor correspondente	87
Figura 28 – Misturando cores primárias de luz: ao deslocar todas as cores para o centro, o círculo, antes negro, fica branco	87
Figura 29 – A combinação das cores aditivas pode ser visualizada pela codificação hexadecimal, o sistema numérico dos computadores	88
Figura 30 – <i>Print screen</i> da interface de animação Cores – mistura das cores aditivas	89
Figura 31 – A animação do círculo cromático mostra a relação entre as cores primárias aditivas e subtrativas	90
Figura 32 – <i>Print screen</i> da interface de animação Cores – cores primárias de luz	91
Figura 33 – A combinação de pigmentos de cor: ao mover as cores para o centro, a interface apresenta o resultado da mistura	92
Figura 34 – A combinação de pigmentos de cor: ao juntar todas as cores de pigmento, o resultado é o preto	92
Figura 35 – Mistura de cores de pigmento mostrando variações em porcentagem como em processos gráficos	93
Figura 36 – <i>Print screen</i> da interface de animação da tarefa para o usuário: que cores combinadas formam a luz branca?	94
Figura 37 – A mensagem para a resposta correta	95
Figura 38 – A mensagem para a resposta não esperada	95
Figura 39 – <i>Print screen</i> da interface de animação Construindo onda	96
Figura 40 – Aumentando o valor de k , diminui-se o comprimento de onda	96
Figura 41 – Aumentando o valor da amplitude, a intensidade da onda aumenta	96
Figura 42 – <i>Print screen</i> da interface de animação Interferência de onda	98
Figura 43 – Interferência construtiva: ondas idênticas com a mesma fase	98
Figura 44 – Interferência destrutiva: ondas idênticas com fases opostas	98

Figura 45 – <i>Print screen</i> da primeira página do questionário	100
Figura 46 – <i>Print screen</i> do questionário para professor	101
Figura 47 – <i>Print screen</i> do questionário para aluno	102
Figura 48 – Ao escolher a opção ensino fundamental abre-se novas opções	102
Figura 49 – <i>Print screen</i> do questionário para usuário comum	103
Figura 50 – Aspecto da tela do <i>Google Analytics</i> para o Laboratório Virtual	104
Figura 51 – Ordem dos blocos da animação Cores em sua primeira versão	109
Figura 52 – Três grupos de cores com quantidade considerável de escolhas	114
Figura 53 – Alteração da distribuição das cores (versão de março de 2011)	115
Figura 54 – Dois grupos de cores que começaram a ter mais escolhas	116
Figura 55 – Nova ordem dos blocos da animação Cores	118
Figura 56 – O processo de produção das animações do Laboratório Virtual na perspectiva do modelo Cascata de Ciclo de Vida	132
Figura 57 – O processo de produção da animação Cores do Laboratório Virtual na perspectiva do modelo de Ciclo de Vida da Engenharia de Usabilidade	133

LISTA DE GRÁFICOS

Gráfico 1 – Proporção de domicílios que possuem equipamentos	24
Gráfico 2 – Proporção de domicílios com computador e Internet	24
Gráfico 3 – Percentual das pessoas que utilizaram a Internet para cada finalidade, na população de 10 anos ou mais de idade	25
Gráfico 4 – Evolução das escolhas dos grupos de cores no tempo	113
Gráfico 5 – Detalhe mostrando que cada registro de dado é formado por um quadrado formado pelas três cores escolhidas	113
Gráfico 6 – Evolução das escolhas de cores no tempo. Período selecionado: julho de 2010 a maio de 2011	116
Gráfico 7 – Gráfico da evolução das escolhas de cores no tempo. Período selecionado: julho de 2010 a outubro de 2011	119
Gráfico 8 – Evolução das escolhas de cores no tempo. Período selecionado: julho de 2010 a junho de 2012	120
Gráfico 9 – Perfil dos usuários do Laboratório Virtual obtidos entre 25 de novembro de 2010 a 27 de junho de 2012	125
Gráfico 10 – O que cursam os alunos que preencheram o questionário do Laboratório Virtual	125
Gráfico 11 – Quantidade de acessos dos alunos, segundo suas séries	126
Gráfico 12 – Onde os professores lecionam	126
Gráfico 13 – Áreas de ensino dos professores segundo os PCNs	127
Gráfico 14 – Formação dos usuários que não eram alunos nem professores	128
Gráfico 15 – Visualizações de páginas e Visualizações únicas entre 1º de janeiro a 30 de julho de 2012	129
Gráfico 16 – Tempo médio de acessos das páginas entre 1º de janeiro a 30 de julho de 2012.....	130

LISTA DE TABELAS

Tabela 1 –	Parte das amostragens das escolhas de cores dos usuários	110
Tabela 2 –	Exibição da combinação do trio de cores escolhida pelo usuário	111
Tabela 3 –	Total de seleções de cores. Período: 22 de julho a 30 de junho de 2012	120
Tabela 4 –	Parte das amostragens das respostas da correção de defeitos da visão. Amostragem de 22 de junho de 2011 a 30 de junho de 2012	122
Tabela 5 –	Resultado parcial dos questionários preenchidos pelos usuários	124

LISTA DE ABREVIATURAS E SIGLAS

ACM – SIGCHI	<i>Association for Computing Machinery - Special Interest Group on Computer–Human Interaction</i> (Grupo de Interesse Especial sobre Interação Humano-Computador)
ARPANET	<i>Advanced Research and Projects Agency</i> (Agência de Pesquisas em Projetos Avançados)
AVI	<i>Audio Video Interleave</i>
CGI	Comitê Gestor da Internet no Brasil
CMYK	<i>Cyan - Magenta - Yellow – Key (Black)</i>
CRT	<i>Cathode ray tube</i> (tubo de raios catódicos)
ECUSP	Estação Ciência da Universidade de São Paulo
EDUSP	Editora da Universidade de São Paulo
GIF	<i>Graphics Interchange Format</i>
HTML	<i>HyperText Markup Language</i>
IBGE	Instituto Brasileiro de Geografia e Estatística
IEEE	<i>Institute of Electrical and Electronics Engineers</i>
IFUSP	Instituto de Física da Universidade de São Paulo
IHC	Interação Humano-Computador
IP	<i>Internet Protocol</i>
JPEG	(ou JPG) <i>Joint Photographic Experts Group</i>
LO	<i>Learning Object</i>
MIDI	<i>Musical Instrument Digital Interface</i>
MP3	<i>Moving Picture Experts Group Audio Layer 3</i>
OA	Objeto de Aprendizagem
PC	<i>Personal Computer</i>

PHP	<i>Hypertext Preprocessor</i> ou, originalmente, <i>Personal Home Page</i>
RGB	<i>Red - Green - Blue</i>
SESC	Serviço Social do Comércio
SWF	<i>Shockwave Flash</i>
TIC	Tecnologia de Informação e Comunicação
WMA	<i>Windows Media Audio</i>
WWW	<i>World Wide Web</i>

SUMÁRIO

1	INTRODUÇÃO	18
1.1	O Laboratório Virtual da Estação Ciência	19
1.2	Uso das tecnologias de informação e comunicação	22
1.3	Objetivos	27
1.4	Principais referências bibliográficas consultadas	28
2	ASPECTOS TEÓRICOS	32
2.1	A Interação Humano-Computador	33
2.1.1	Histórico da IHC	35
2.2	O processo da produção da interface e modelos de Ciclo de Vida	36
2.2.1	Modelo de Ciclo de Vida simples	36
2.2.2	Modelos de Ciclo de Vida em Engenharia de <i>Software</i>	37
2.2.2.1	O modelo Cascata de Ciclo de Vida	38
2.2.2.2	O modelo Espiral de Ciclo de Vida	39
2.2.3	Modelos de Ciclo de Vida em IHC	40
2.2.3.1	Modelo de Ciclo de Vida Estrela	40
2.2.3.2	O Ciclo de Vida da Engenharia de Usabilidade	41
2.3	Usabilidade	45
2.4	Avaliação das interfaces: inspeção de Usabilidade e teste de Usabilidade	46
2.5	Testes de Usabilidade	47
2.5.1	Método etnográfico de coleta de dados	48
2.5.2	Coleta de dados	50
2.5.3	Rastreamento remoto de atividades dos usuários (<i>Log</i> de dados)	51
2.6	Objeto de aprendizagem	51
2.6.1	Características dos Objetos de Aprendizagem	53
2.7	Produção de OA no modelo de Ciclo de Vida de <i>Software</i>	55
2.8	Usabilidade na <i>web</i>	58
2.9	Resumo	60

3	METODOLOGIA	62
3.1	Linguagens de programação utilizadas	62
3.2	O <i>website</i> Laboratório Virtual	66
3.3	As interfaces do Laboratório Virtual	69
3.3.1	Reflexão	72
3.3.2	Refração	73
3.3.3	Lentes (Como se formam as imagens com as lentes)	75
3.3.4	Espelhos esféricos (Como se formam as imagens com espelhos esféricos)	76
3.3.5	O olho (formação da imagem no olho e defeitos da visão)	78
3.3.6	O olho (Formação de imagem pelo olho – olho normal)	79
3.3.7	Miopia	81
3.3.8	Hipermetropia	82
3.3.9	Correção (Correção de defeitos da visão)	83
3.3.10	Cores	85
3.3.10.1	Cores primárias de luz	86
3.3.10.2	Mistura das cores aditivas	89
3.3.10.3	Cores primárias de pigmento	90
3.3.10.4	Tarefa para o usuário: que cores combinadas formam a luz branca?	93
3.3.11	Construindo ondas (onda transversal)	95
3.3.12	Interferência de ondas	97
3.4	Obtendo informações do usuário da interface	99
3.4.1	Questionário de perfil dos usuários	99
3.5	<i>Google Analytics</i>	104
3.6	Resumo	105
4	RESULTADOS	106
4.1	Objetivos dos testes realizados na interface de animação Cores	106
4.2	Primeira avaliação da interface de animação Cores	109
4.3	Alteração no <i>layout</i> da interface	115
4.4	Nova alteração no <i>layout</i> da interface	117
4.5	Resultados obtidos com a interface de animação Correção de defeitos da visão	122
4.6	Conhecendo o usuário	123
4.7	<i>Google Analytics</i>	129

5	CONCLUSÃO	131
6	CONSIDERAÇÕES FINAIS	136
6.1	Perspectivas futuras	137
7	REFERÊNCIAS	139
	GLOSSÁRIO	149
	APÊNDICE A – Animações do Laboratório Virtual selecionadas para integrar o Banco Internacional de Recursos Multimídia do MEC em 2008	152
	APÊNDICE B – Principais trechos de código <i>action script</i> das animações do Laboratório Virtual	153
	APÊNDICE C – Código <i>action script</i> da interface de visualização da Evolução das Escolhas dos Grupos de Cores	167
	APÊNDICE D – Códigos PHP das interfaces de visualização de resultados do método de <i>Log</i> de dados	169
	APÊNDICE E – Códigos PHP das interfaces de visualização do Resultado do questionário preenchido pelos usuários	178
	ANEXO A – As diretrizes de Usabilidade de Jakob Nielsen	184

1. INTRODUÇÃO

A presente pesquisa aborda a análise do desenvolvimento e do uso das animações do Laboratório Virtual da Estação Ciência / USP (ECUSP)¹. O estudo se baseia no processo de produção de interfaces centrado no usuário, apoiando-se nas propostas originadas da Ciência denominada Interação Humano-Computador (IHC). Para verificar como o usuário interagia com a animação e, dessa forma, poder fazer avaliações sobre o seu uso, foi utilizado o método de *Log* de dados, ou seja, a coleta remota de informações e sua gravação em banco de dados. Apesar de este método ser orientado para apoiar outras técnicas de observação, como a entrevista ou a gravação de vídeo, o uso isolado do *Log* de dados permitiu obter informações relevantes sobre a dinâmica da interação do usuário com a interface.

Os resultados obtidos poderiam ter sido mais expressivos se também fosse realizada uma pesquisa em sala de aula com observações diretas sobre as ações e reações dos alunos e dos professores com as animações. Mas, apesar do projeto ter enfatizado apenas os aspectos relacionados ao processo de produção de interfaces, o volume de material desenvolvido foi relativamente grande e complexo, permitindo a avaliação da criação de *softwares*, além de apontar questões críticas sobre o seu uso nos ambientes de ensino e aprendizagem.

Toda a produção desse trabalho pode ser acessada pela Internet, no endereço do Laboratório Virtual (www.ideiasnacaixa.com/laboratoriovirtual²), assim como os resultados da pesquisa realizada.

¹ Estação Ciência - Centro de Difusão Científica, Tecnológica e Cultural da Pró-Reitoria de Cultura e Extensão Universitária da Universidade de São Paulo. Foi inaugurada em 24 de junho de 1987 e em seu espaço são realizadas exposições e atividades nas áreas de Astronomia, Meteorologia, Física, Geologia, Geografia, Biologia, História, Informática, Tecnologia, Matemática, Humanidades, além de cursos, eventos e outras atividades, com o objetivo de popularizar a ciência e promover a educação científica (Estação Ciência. Disponível em: <www.eciencia.usp.br>. Acesso em: 14.jul.2012).

² Até agosto de 2012, o Laboratório Virtual podia ser acessado pelo endereço www.eciencia.usp.br/laboratoriovirtual. No entanto, este foi desvinculado do site da Estação Ciência por determinação da gestão em exercício.

1.1 O Laboratório Virtual da Estação Ciência

O Laboratório Virtual é fruto de uma ideia gestada no período em que trabalhava na Editora da Universidade de São Paulo (EDUSP). De 1995 a 1998, fui Arte-Finalista nessa editora e, concomitantemente, estava em processo de conclusão da minha graduação em bacharelado em física no Instituto de Física da USP (IFUSP).

Durante o tempo em que trabalhei na EDUSP produzi ilustrações e capas para livros, em sua maior parte relacionadas a temas científicos. Era frequente, durante a produção editorial, auxiliar no desenvolvimento de gráficos e esquemas, principalmente quando se referiam a um conteúdo ligado à física.

Naquele ambiente de trabalho iniciei o uso profissional de programas de computador para editoração e ilustração eletrônica. Juntamente com outros colegas, procurávamos outras formas de produzir conteúdo editorial. Foi assim que comecei a desenvolver páginas em HTML e pequenas animações interativas em *Macromedia Flash*.

Apesar de não ter tido oportunidade de usar essas linguagens nas produções editoriais da EDUSP, continuei explorando suas potencialidades. Ao concluir minha graduação, em 1997, decidi criar animações simples explorando conceitos de física. Em 1998 levei esse projeto à ECUSP, onde ingressei como Especialista em Laboratório.

Na ECUSP fui responsável por gerenciar o site da instituição (www.eciencia.usp.br) e desenvolver o Laboratório Virtual (www.eciencia.usp.br/laboratoriovirtual), explorando a linguagem da *web* para promover a divulgação científica.

Assim, o Laboratório Virtual nasceu oficialmente em 1998 e originalmente era constituído por animações interativas que reproduziam o funcionamento de equipamentos expostos na ECUSP. Embora criado inicialmente com temas voltados à física, aos poucos foram introduzidos trabalhos de outras áreas, como geologia, geografia, história, biologia, astronomia, matemática, entre outros, com a colaboração de estagiários que formaram a equipe de produção³.

Entre março de 1999 e março de 2000 foi incluído no site da ECUSP um formulário para verificar o perfil do público visitante (MATSUKUMA, 2000, p. 477). O resultado

³ Em 2010, a gestão em exercício da ECUSP suspendeu a contratação de estagiários para o Laboratório Virtual.

indicou que grande parte era formada por professores (32%) e estudantes (25%). Ou seja, 57% dos indivíduos tinham vínculo com o ambiente escolar.

Em meados de 2008, doze animações do Laboratório Virtual (ver Apêndice A) foram selecionadas para integrar o Banco Internacional de Recursos Multimídia⁴ do Ministério de Educação do Brasil (MEC).

Os dados mais recentes mostram que no primeiro semestre de 2012 (1.jan a 30.jun), o Laboratório Virtual recebeu 15.150 visitas⁵, em sua maior parte formada por professores e alunos do ensino médio e fundamental⁶.

No entanto, apesar de mais de uma década desenvolvendo e publicando animações, nenhuma pesquisa aprofundada havia sido feita para verificar como as interfaces eram utilizadas pelo público. A oportunidade de realizar um estudo desse tipo surgiu quando ingressei no Programa de Pós-Graduação Internunidades em Ensino de Ciências da Universidade de São Paulo (2009). O método que pretendia utilizar na pesquisa era a observação através do monitoramento remoto (posteriormente compreendi que esse processo denominava-se *Log* de dados).

A primeira vez que observei uma técnica semelhante foi em um projeto, desvinculado da USP, do qual participei, denominado *Amazing Amazon* (GARCIA, 2002). O trabalho começou em meados de 2000, quando fui convidado pelo arquiteto e *designer* José Wagner Garcia para desenvolver uma interface para simular as alterações do curso de um rio por meio de recursos computacionais. Essa interface desenhada em *Flash*, constituída por diversos elementos interativos, deveria ser modificada ao ser manipulada pelo usuário. O programa detectava os elementos clicados e enviava a informação a um banco de dados. Um programa desenvolvido em PHP calculava as interações coletivas dos usuários e reformulava o padrão dos elementos, alterando o desenho da interface. Em outubro de 2002 o simulador foi concluído e apresentado no Museu de Arte Moderna (MAM) de São Paulo, em uma exposição que incluía a exibição de fotos de satélite mostrando as alterações das margens do rio Amazonas ao longo dos anos.

⁴ Repositório de objetos de aprendizagem do MEC (disponível em <<http://objetoseducacionais2.mec.gov.br>>. Acesso em: 13.jul.2012).

⁵ De acordo com o serviço do *Google Analytics* (disponível em: <www.google.com/analytics>. Acesso em: 10.jul.2012).

⁶ De acordo com a pesquisa realizada para esse projeto, entre 25 de novembro de 2010 a 27 de junho de 2012, 38% do público correspondeu a alunos e 51%, a professores.

Nessa época, porém, ainda não dominava a técnica de coleta e armazenamento de informações em banco de dados, procedimento que foi realizado por outro programador. No entanto, foi a partir desse trabalho que me interessei pelo método e comecei a estudá-lo.

Em 2009 tive oportunidade de me integrar a uma equipe para o desenvolvimento de um *website* para o SESC Santo Amaro, denominado SESC Santo Amaro em Rede⁷. Nessa ocasião já dominava o processo de integração das animações em *Flash* com Banco de Dados, através do uso da linguagem PHP. Esse foi um conhecimento importante, já que era necessário desenvolver, no *website* do SESC Santo Amaro, uma interface gráfica que reproduzia a região sul da cidade de São Paulo e onde eram localizados dispositivos culturais de instituições cadastradas e armazenados em um grande banco de dados. O projeto foi concluído em maio de 2010 e o resultado pode ser conferido no endereço www.mostrasescdeartes.com.br/stoamaroemrede.

Esses trabalhos foram de grande relevância, já que me forneceram o conhecimento técnico para o desenvolvimento de um sistema computacional para a observação remota de interações de usuários. Assim, em 2010, já no Programa de Pós-Graduação Internunidades, comecei a desenvolver os primeiros protótipos do Laboratório Virtual para testar o recurso.

As animações interativas desenvolvidas para o Programa tiveram como tema a Óptica. Foram realizadas ao todo doze interfaces, dentre as quais duas com o recurso de *Log* de dados. Para essas duas animações foram desenvolvidos programas de captura de dados em PHP que enviavam as informações para um banco de dados. Para a análise dos dados foram criados programas específicos que somavam números de eventos e fazia a sua exibição para análise. Também foi desenvolvido um questionário para determinar o perfil dos usuários, cujos dados também eram enviados a um banco de dados e suas informações, apresentadas através de outro programa.

O desenvolvimento dessas interfaces e a sua relação com o usuário é o tema principal desse projeto.

⁷ Disponível em <www.mostrasescdeartes.com.br/stoamaroemrede>.

1.2 Uso das tecnologias de informação e comunicação

A aprendizagem mediada por computador tem sido discutida com muita relevância pelas políticas de ensino público⁸. No entanto, é preciso perceber que o uso da tecnologia não se constitui um facilitador da atividade pedagógica por si só. A formação das novas competências do professor, a compreensão do perfil do usuário, o processo do desenvolvimento de interfaces, são fundamentais para que os ambientes virtuais possam ser estruturados para dar suporte adequado à aprendizagem.

Sendo assim, a aproximação do sujeito com a tecnologia deve ser realizada dentro de uma perspectiva social. A própria tecnologia, em particular a informática, para ser assimilada de forma contextualizada, deve ser compreendida como fruto de um processo social e histórico.

Os computadores surgiram em meados do século vinte. Aparentemente apenas máquina, instrumento com funções de processamento matemático de grande velocidade e precisão, o computador foi introduzido no cotidiano das pessoas, promovendo relações sociais significativas (CASTELLS, 2003; LÉVY, 2005; COLL, 2010; NEGROPONTE, 1995; PAPERT, 2004).

A partir dos sistemas computacionais, aos poucos foram surgindo uma sociedade virtual, estruturando um novo espaço de informações. As primeiras redes de computadores foram criadas para fins militares (ARPANET⁹) e posteriormente para o compartilhamento de

⁸ Um exemplo da importância da introdução do uso das tecnologias de informação e comunicação (TIC) na educação pode ser verificado nas ações da Secretaria de Educação a Distância (SEED) do Ministério da Educação do Governo Federal do Brasil, criada em 1996. Ela atua como um agente de inovação tecnológica nos processos de ensino e aprendizagem, fomentando a incorporação das TIC e das técnicas de educação a distância aos métodos didático-pedagógicos. Entre ações subordinadas a essa secretaria estão:

- O Programa Um Computador por Aluno (Prouca), de 2010.
- O Programa Banda Larga nas Escolas (PBLE), de 2008.
- O Banco Internacional de Objetos Educacionais, de 2008.
- E o Programa Nacional de Tecnologia Educacional (ProInfo), de 1997.

(BRASIL. Ministério da Educação. Disponível em:

<http://portal.mec.gov.br/index.php?option=com_content&view=article&id=289&Itemid=822>.
Acesso em: 15.jul.2012).

⁹ A ARPANET foi desenvolvida pela agência americana ARPA (Advanced Research and Projects Agency - Agência de Pesquisas em Projetos Avançados) em 1969, e tinha o objetivo de interligar as

trabalhos acadêmicos nas universidades norte-americanas (origem da Internet). Nesse ambiente surgiram indivíduos que criavam e compartilhavam programas livremente; o desejo de se conectar com todos e trocar dados deu origem à grande rede mundial (WWW¹⁰ na sigla em inglês).

Dessa forma, a sociedade em rede (CASTELLS, 2003) foi sendo construída, promovendo o compartilhamento de informações produzidas e acessadas pelos usuários, criando um relacionamento do tipo *todos-todos* (LÉVY, 1999, p. 63). Ao contrário dos meios de comunicação tradicionais em que um centro emissor levava a informação a muitos (*um-todos*), sem que esses pudessem interferir na fonte da mensagem (jornal, rádio, televisão), nessa rede, o compartilhamento do conhecimento de muitos para muitos, proporcionaria a participação mais totalizante e democrática dos indivíduos na sociedade (LÉVY; MICHEL, 2000, p. 188).

O Gráfico 1¹¹ apresenta a evolução da participação de equipamentos de tecnologias de comunicação e informação (TIC) nas residências brasileiras, segundo pesquisa do Comitê Gestor da Internet (CGI). Observa-se como a presença dos equipamentos mais novos, como o computador de mesa, o *notebook* e o celular vêm crescendo ao longo dos últimos anos, enquanto aqueles tradicionais, como a televisão, o rádio e o telefone fixo têm estabilizado ou diminuído sua participação (Comitê Gestor da Internet no Brasil. Pesquisa TIC Domicílios, 2011, p. 13).

No Gráfico 2¹², observa-se o crescimento de domicílios com computador e a expansão do uso da Internet no país. Mesmo assim, segundo o CGI, ainda há 4,6 milhões de domicílios com computador e sem acesso a Internet (Comitê Gestor da Internet no Brasil. Pesquisa TIC Domicílios, 2011, p. 11).

bases militares e os departamentos de pesquisa do governo americano (disponível em: <<http://pt.wikipedia.org/wiki>>. Acesso em: 20.jul.2012).

¹⁰ A rede mundial de computadores, ou World Wide Web (WWW) no original em inglês, foi criada por Berners-Lee, cientista do Conselho Europeu de Pesquisas Nucleares (CERN, na sigla em francês), na década de 1990 (disponível em: <<http://pt.wikipedia.org/wiki>>. Acesso em: 20.jul.2012).

¹¹ Fonte: Comitê Gestor da Internet no Brasil. Pesquisa TIC Domicílios, 2011. Disponível em: <<http://cetic.br/usuarios/tic/2011-total-brasil/analises.htm>>. Acesso em: 2.jul.2012.

¹² Fonte: Comitê Gestor da Internet no Brasil. Pesquisa TIC Domicílios, 2011. Disponível em: <<http://cetic.br/usuarios/tic/2011-total-brasil/analises.htm>>. Acesso em: 2.jul.2012.

Gráfico 1 – Proporção de domicílios que possuem equipamentos (percentual sobre o total de domicílios).

Gráfico 2 – Proporção de domicílios com computador e Internet (percentual sobre o total de domicílios).

O Gráfico 3¹³ apresenta a finalidade de uso da Internet no Brasil, na população com idade igual ou superior a 10 anos, em 2005 e 2008. Observa-se que, apesar da diminuição na finalidade em Educação e aprendizado (era a principal em 2005), ela ainda é tema citado por 65,9% dos indivíduos que acessaram a Internet (IBGE, Diretoria de Pesquisas, Coordenação de Trabalho e Rendimento, Pesquisa Nacional por Amostra de Domicílios 2008, p. 42).

¹³ Fonte: IBGE, Diretoria de Pesquisas, Coordenação de Trabalho e Rendimento, Pesquisa Nacional por Amostra de Domicílios 2008. Disponível em: <<http://www.ibge.gov.br/home/estatistica/populacao/acessoainternet2008/internet.pdf>>. Acesso em: 13.jul.2012.

Gráfico 3 – Percentual das pessoas que utilizaram a Internet para cada finalidade, na população de 10 anos ou mais de idade, no período de referência dos últimos três meses, por situação de ocupação na semana de referência, segundo a finalidade do acesso à Internet.

Compreende a natureza social da tecnologia, no entanto, não minimiza as dúvidas sobre sua utilização nas mais diversas áreas das interações humanas, sobretudo na educação. Apesar do crescimento de acesso aos equipamentos tecnológicos estimular a sua utilização em sala de aula, muitas vezes essa prática é apoiada em propostas descontextualizadas, o que pode motivar críticas por parte dos educadores.

As novas tecnologias não podem ser utilizadas apenas porque elas existem. Suas potencialidades devem ser investigadas e adaptadas à realidade da escola, dos professores e dos alunos. Recursos como livros, cadernos, papel e lápis continuarão sendo fundamentais na sala de aula e não deverão ser substituídos. O computador e a Internet, portanto, devem ser introduzidos para complementar esses elementos, além de apoiar diferentes perspectivas de ensino originadas da sociedade em rede.

Um exemplo de recurso que poderia ser explorado nos meios computacionais é a simulação, destacada por Lévy (1995, 1999) como uma ferramenta que permitiria a construção de um conhecimento diferente do científico (apoiado pela linguagem escrita) e da

tradição oral e, portanto, complementar aos recursos utilizados tradicionalmente em sala de aula.

A simulação possibilita uma exploração rápida e repetida de fenômenos, permitindo a formulação de um grande número de hipóteses, ao contrário das ações praticadas no espaço real. Nas áreas científicas e técnicas auxilia na análise, compreensão e previsão do funcionamento de sistemas (COLL; MONEREO, 2010, p.234). Permite explorar locais inalcançáveis pelos meios físicos, como o núcleo atômico ou o interior das estrelas; ou a observação de objetos imaginados apenas na mente humana, como estruturas moleculares, elétrons em movimento e ondas eletromagnéticas.

Mesmo assim, a utilização da simulação não é garantia de aprendizagem. Para que isso ocorra, ela deve ser elaborada especificamente para esse fim. Ela deve ter características de um Objeto de Aprendizagem (OA), ou seja, poder ser combinada com outros objetos para proporcionar a criação de ambientes diferentes de aprendizagem (granularidade e modularidade) e poder ser localizada e acessada facilmente na rede (identificação por metadados).

Além disso, para garantir que o indivíduo possa utilizar o objeto com facilidade (Usabilidade), o OA ou qualquer interface computacional devem ter sua produção centrada no usuário. Esta, por sua vez, é uma preocupação da disciplina denominada Interação Humano-Computador (IHC), responsável por avaliar e implementar os sistemas computacionais para uso humano, e que engloba conhecimentos tão diversos como a Ciência da Computação, a Psicologia, a Sociologia, a Antropologia e o Desenho Industrial (ACM SIGCHI, 1992).

Assim, através de orientações derivadas da IHC, o processo de produção de uma interface pode ser estruturado em etapas que são submetidas à avaliação do usuário para a verificação de diretrizes de Usabilidade (NILSEN; TAHIR, 2002). Testes são aplicados em protótipos, sendo possível verificar a ocorrência de alguma falha de Usabilidade da interface que pode tanto acarretar na correção de algum desenho ou programa específico, como alterar alguma definição inicial do projeto. Sucessões de modificações e testes (processo de iteração) podem fornecer respostas importantes para se produzir um produto final que possa ser melhor adaptado às necessidades do usuário.

Diante da crescente presença do computador no cotidiano é importante que sua introdução nos meios educacionais ocorra observando o indivíduo e o seu ambiente. A produção de um OA deve ser centrada no usuário e avaliada por especialistas de diferentes

áreas (educação, informática e demais profissionais envolvidos no tema explorado). No entanto, de nada adianta o *software* e o acesso aos equipamentos de informática se o ambiente de ensino não oferece condições mínimas para que ocorra a orientação à aprendizagem. Também o professor, adaptado aos recursos tradicionais de ensino, deve ser apoiado nesse momento de transição, oferecendo a ele não apenas treinamento, mas oportunidade da construção conjunta desse novo espaço de aprendizagem.

A introdução e o uso das ferramentas tecnológicas não podem ser realizados de modo simplista. Isso apenas colaboraria para descontextualizar a utilização dos computadores nas salas de aula. Por outro lado, deve-se encorajar que o processo ocorra de forma crítica e com a participação de todas as partes interessadas. Dessa forma, será possível construir um espaço de aprendizagem integrando o tradicional com o virtual e centrado efetivamente no indivíduo e na sociedade.

1.3 Objetivos

De modo geral, essa pesquisa visa à criação de interfaces segundo uma perspectiva de produção de *software* centrada no usuário, procurando realizar avaliações sobre sua utilização durante seu processo de desenvolvimento.

De modo específico, os objetivos desse trabalho são:

1. A construção de interfaces tendo como referência a IHC;
2. Introduzir o método de *Log* de dados nas interfaces, ou seja, enviar informações remotamente para um banco de dados para posterior consulta e avaliação;
3. Produzir interfaces de leitura das informações armazenadas pelo método de *Log* de dados;
4. Verificar como as informações obtidas contribuem para a construção das interfaces.

Além disso, um importante requisito para a produção de *softwares* é conhecer o seu público alvo. Assim, outros objetivos do projeto são:

5. A construção de um questionário *on-line* que armazene as informações em um banco de dados.
6. A construção de uma interface de leitura das informações armazenadas pelo questionário *on-line*.

Um último objetivo é acompanhar a audiência das interfaces produzidas através da utilização do método de *Log* de dados do serviço do *Google Analytics*.

Enfim, procura-se com esse trabalho apresentar um processo de construção de *software* em que o usuário não faz apenas o uso de um produto finalizado, mas que contribui e participa do ciclo de produção dessa interface.

1.4 Principais referências bibliográficas consultadas

O percurso para se chegar às bibliografias consultadas nesse trabalho sofreu alterações de acordo com a evolução e a análise dos estudos sobre a criação de interfaces interativas para o uso no ambiente de ensino e de aprendizagem.

Sendo as produções do Laboratório Virtual as principais ferramentas desse estudo, era necessário que se fizesse uma revisão sobre os Objetos de Aprendizagem (OA). Além disso, um dos focos do estudo se baseava na utilização de métodos para verificar como os usuários interagiam com as interfaces produzidas, o que poderia ser revelado pelos estudos de Usabilidade.

Inicialmente foram pesquisados na Biblioteca Digital de Teses e Dissertações da USP¹⁴ trabalhos relacionados à palavra-chave “Objetos Virtuais de Aprendizagem”. Entre os títulos encontrados, buscou-se aqueles que se aprofundavam em suas explicações, definindo-os e caracterizando-os.

Os títulos encontrados referem-se às teses de doutorado de Nunes (2011), *Ensino de conceitos físicos no ensino médio e as contribuições dos objetos de aprendizagem*, de Salve (2010), *Modelo de planejamento para repositório de objetos de aprendizagem em*

¹⁴ Disponível em: <<http://www.teses.usp.br>>.

organizações educacionais, de Rivas (2009), *Objetos de aprendizagem no contexto das comunidades virtuais auto-organizadas para a produção de software livre e de código aberto* e Vaz (2007), *Modelagem e arquitetura de sistemas para monitoração e acompanhamento da aprendizagem eletrônica*, além da dissertação de mestrado de Baptista (2007), *Msys: uma ferramenta de acompanhamento de atividades para sistemas de aprendizado eletrônico*.

A partir desses títulos foi possível compreender como eram caracterizados os OA. Outros trabalhos, referenciados nesses textos, tratavam de características específicas dos OA, ampliando o entendimento sobre o uso desse recurso no ensino e na aprendizagem. Na dissertação de mestrado de Oliveira (2009), intitulado *Segmentação de objetos de aprendizagem e abordagens para sua utilização*, o autor destaca as características de granularidade e reusabilidade dos OA; no artigo de Betio (2002), *Objetos de aprendizado: um novo modelo direcionado ao ensino a distância*, e principalmente no de Gibbons e Nelson (2000), *The nature and origin of instructional objects*, os autores definem e contextualizam a origem dos OA; Pimenta e Baptista, em *Das plataformas de e-learning aos objetos de aprendizagem*, Filho e Machado, em *O computador como agente transformador da educação e o papel do objeto de aprendizagem*, Tarouco e Fabre (2003), em *Reusabilidade de objetos educacionais*, Jacobson (2002), em *Reusable Learning Objects – What does the future hold?*, e Muzio, Heins e Mundell, em *Experiences with reusable elearning objects: from theory to practice*, destacam as características da modularidade, da reusabilidade e o uso de metadados para a construção de repositórios de OA.

Um importante documento que define padrões técnicos do uso de metadados é o *Draft standard for learning object metadata*, elaborado pelo IEEE Learning Technology Standards Committee (2002). Finalmente, um autor muito referenciado é Wiley, cujo artigo *Connecting learning objects to instructional theory: a definition, a metaphor and a taxonomy*, define o OA e traça um detalhado histórico sobre sua origem.

Já em artigos como o de Monteiro (2006), *Metodologia de desenvolvimento de objetos de aprendizagem com foco na aprendizagem significativa*, de Souza (2005), *Objetos de aprendizagem colaborativos*, e de Mendes, Souza e Caregnato (2005), *Propriedade intelectual na elaboração de objetos de aprendizagem*, além da descrição de características dos OA, os autores fazem referência à produção e à avaliação desses objetos, analisadas sob perspectivas educacionais. Porém, apesar de constituírem análises importantes, procurava-se a avaliação do *software* segundo suas características de Usabilidade, tema de particular interesse dessa pesquisa.

Dessa forma, recorreu-se novamente à Biblioteca Digital de Teses e Dissertações da USP para procurar trabalhos com foco na Usabilidade.

Foram encontradas referências sobre o tema nas teses de doutorado de Neto (2011), *Uma abordagem para projeto de aplicações com interação multimodal na web*, de Ferecini (2011), *Desenvolvimento e avaliação do objeto digital de aprendizagem sobre o aleitamento materno do prematuro*, de Góes (2010), *Desenvolvimento e avaliação de objeto virtual de aprendizagem interativo sobre o raciocínio diagnóstico em enfermagem aplicado ao recém-nascido pré-termo*, e de Silva (2007), *Design e arquitetura de informação para web sites educacionais: um estudo de usabilidade*.

Um autor muito referenciado nestes textos é Jacob Nielsen, autor que desenvolveu as diretrizes de Usabilidade para serem aplicadas nos *websites* (Nielsen e Loranger, 2002).

A leitura dos textos sobre a avaliação da usabilidade mostrou a importante relação entre o usuário e o *software*. Esta por sua vez, teria seu estudo fundamentado pela ciência denominada Interface Humano-Computador (IHC). Seria através da IHC que se poderia explicar a produção de *softwares* centrada no usuário, como apresentados nas dissertações de Filardi (2007), *Análise e avaliação de técnicas de interação humano-computador para sistemas de recuperação de imagens por conteúdo baseadas em um estudo de caso*, e de Jacober (2007), *Proposta e implementação de uma interface para motores de jogos interativa e centrada no usuário*.

Um documento fundamental, citado nesses trabalhos, e que define a ciência da IHC é o *Curricula for Human-Computer Interaction* (ACM SIGCHI, 1992, 2009). Outras duas bibliografias, referenciadas por aqueles autores, e que descrevem o processo de produção de uma interface (Ciclo de Vida de *software*) centrada no usuário, são de Preece, Rogers e Sharp (2005), *Design de interação: além da informação homem-computador*, e de Rocha e Baranauskas (2003), *Design e avaliação de interfaces humano-computador*.

Ao apresentar os processos para avaliação das produções de *softwares*, esses autores descrevem os métodos de coletas de dados das ações dos usuários sobre as interfaces. Foi na leitura desses textos que se encontrou a descrição do método denominado *Log* de dados, que seria um referencial importante para explicar a forma de obtenção de informações nas interações dos usuários com as animações do Laboratório Virtual. O método de *log* de dados é descrito em detalhes no artigo de Hartson e Castilho, *Remote evaluation: the network as an extension of the usability laboratory*.

Além desses textos, também foram consultados os trabalhos do Programa de Pós-Graduação Interunidades em Ensino de Ciências¹⁵. Entre esses trabalhos foram verificados aqueles cujo eixo principal de pesquisa era o uso das TIC.

Assim, encontraram-se as dissertações de mestrado de Ayres (2011), *O uso do recurso multimídia no ensino de química para alunos de ensino médio sobre o conteúdo de ligações intermoleculares*, de Moraes (2011), *Aplicação de recursos de ambiente virtual de aprendizagem em curso de biologia do ensino médio*, de Moreira (2009), *Ambiente virtual interativo no ensino de ciências: uma abordagem sociocultural*, de Oliveira (2009), *O software modellus e sua possibilidade para desafiar as concepções de senso comum em óptica*, de Bonetti (2008), *A linguagem de vídeo e a natureza da aprendizagem*, de Colagrande (2008), *Desenvolvimento de um jogo didático virtual para a aprendizagem do conceito mol*, de Silva (2007), *Desenvolvimento de um ambiente virtual para estudo sobre representação estrutural em Química*, de Severino (2005), *Recursos virtuais em aulas de laboratório de física*, e de Tofoli (2003), *Utilização e compreensão do computador: um olhar no dia-a-dia do professor*, além da tese de doutorado de Vinha (2007), *Criatividade em ação: roteiros de animações virtuais elaborados por alunos de ensino médio em físico*.

Muitos desses trabalhos propõem a avaliação de *software* com foco na aprendizagem, mas não pelas diretrizes de Usabilidade de Nielsen. Apesar da maior parte desses textos tratarem da análise de *softwares* já existentes, dois deles trabalharam com o desenvolvimento de animações em *Shockwave Flash* (Moreira, 2009 e Silva, 2007). Além disso, esses trabalhos utilizaram o método de observação indireta de vídeo para a geração de relatórios. Porém não são citados nessas pesquisas metodologias de produção de interface centrada no usuário ou a utilização de *Log* de dados.

Dessa forma, a presente bibliografia procura contribuir com os trabalhos sobre avaliações de interfaces para o uso no ensino, observando a produção de *softwares* apoiada na perspectiva da IHC, compondo, assim, um cenário mais amplo de estudo da construção dos OA.

¹⁵ Disponível em: <<http://web.if.usp.br/cpgi/pt-br/node/61>>.

2. ASPECTOS TEÓRICOS

O processo de avaliação de interfaces produzidas para o ambiente da internet, como é o caso dos objetos de aprendizagem (OA), pode se relacionar tanto ao seu conteúdo, quanto a aspectos ligados ao seu desenho ou facilidade de uso. Além disso, a maneira como as interfaces são produzidas também podem passar por diferentes tipos de análise.

Pesquisas relacionadas a essas interfaces têm sido realizadas pelo campo da ciência denominada Interação Humano-Computador (IHC). Assim, a IHC estuda como as pessoas interagem com sistemas computacionais, bem como a cadeia de produção dessas interfaces, tendo seu processo centrado no usuário.

A avaliação do modo como os indivíduos se relacionam com a interface pode ser amparada pelas diretrizes de Usabilidade. Estas podem garantir interfaces mais eficientes e mais fáceis de serem utilizadas por eles (NIELSEN; TAHIR, 2002).

Os testes de Usabilidade são aplicados no processo de produção da interface e durante a sua utilização pelo usuário. O *feedback* pode orientar um novo desenho ou uma programação diferente para melhorar sua interação com o indivíduo.

Diferentes métodos de pesquisa podem ser utilizados nos testes de Usabilidade. As avaliações das interfaces são realizadas mediante coleta de dados, que podem ser por observação direta, como a realizada por um avaliador a partir de questionários, ou indireta, através de gravações de vídeo, por exemplo.

Uma das técnicas utilizadas nos testes de Usabilidade é a coleta remota de dados (*Log de dados*). Esse é um método que auxilia a coleta tradicional de dados e que ocorre no local natural de trabalho do usuário. Nessa técnica, a interface recebe uma programação especial que captura diferentes informações, como cliques de mouse, teclas acionadas, tempo de permanência, entre outras ações.

As interfaces produzidas para esse trabalho, que se caracterizam como OA, tiveram sua interação com o usuário observada mediante o método de *Log de dados*. Sua utilização principal foi realizada a partir da introdução de elementos de programação na interface que forneciam informações para a avaliação do OA estudado. Os resultados obtidos nesse estudo

revelaram como o desenho da interface promovia ações não esperadas realizadas pelos usuários.

O uso de *Log* de dados também esteve presente no serviço de audiência do *Google Analytics*, ferramenta gratuita oferecida pela empresa especializada em busca na Internet. A partir desse serviço foi possível obter informações sobre quantidade de acessos e o tempo de permanência médio de cada interface.

2.1 A Interação Humano-Computador

A Interação Humano-Computador (IHC) pode ser definida como “a disciplina relativa ao *design*, à avaliação e à implementação de sistemas computacionais interativos para uso humano e com o estudo de fenômenos importantes que os rodeiam” (ACM SIGCHI¹⁶, 1992).

Ao contrário da Ciência da Computação, que tem foco na interação entre um ou mais seres humanos e uma ou mais máquinas computacionais, a IHC se destaca por estudar as relações mais aprofundadas entre as máquinas e o homem.

Segundo o ACM SIGCHI (1992), a IHC é uma grande área disciplinar que engloba a Ciência da Computação, a Psicologia, a Sociologia, a Antropologia e o Desenho Industrial. Rocha e Baranauskas (2003, p. 3) ainda acrescentam a essa lista a Psicologia Educacional, o *Design* Instrucional e Gráfico e a Ergonomia.

Algumas preocupações da IHC são apresentadas pelo ACM SIGCHI para caracterizá-la como um campo de estudo:

- Desempenho conjunto de tarefas por seres humanos e máquinas;
- Estrutura de comunicação entre seres humanos e máquinas;
- A capacidade humana de utilizar máquinas (incluindo a capacidade de aprendizado de interfaces);
- Algoritmos e programação da interface;

¹⁶ SIGCHI - Special Interest Group on Computer-Human Interaction (Grupo de Interesse Especial sobre Interação Humano-Computador, em tradução livre), faz parte da ACM - Association for Computing Machinery. O SIGCHI é a criadora do campo de pesquisa de IHC (Disponível em: <http://old.sigchi.org/cdg/cdg2.html#2_1>. Acesso em: may.2012).

- Problemas de engenharia que possam surgir na concepção e construção de interfaces;
- O processo de especificação, projeto e implementação de interfaces.

Observa-se nas questões apresentadas acima a presença do termo interface. Segundo Rocha e Baranauskas (2003, p. 3) a interface não deve ser entendida apenas como lugar físico onde ocorre o contato entre duas entidades. O conceito pode ser ampliado, considerando como interface uma superfície de contato que reflete as propriedades físicas das partes que interagem, as funções a serem executadas e o balanço entre poder e controle. Ou seja, a superfície de contato não precisa ser necessariamente real, mas pode ser virtual e, mesmo assim, ser controlada pelos usuários para que ações desejadas sejam executadas.

Conforme as preocupações apresentadas pela IHC, não é possível imaginar a interface sem considerar o ser humano que irá utilizá-lo. Segundo Rocha e Baranauskas (2003, p. 13), produzir ambientes virtuais para a internet não consiste apenas em “criar tipos mais legíveis, melhores barras de rolagem, integrar cor, som e voz”. Apesar de serem importantes, são secundários. “A preocupação primeira deve ser melhorar o modo como as pessoas podem usar o computador para pensar e comunicar, observar e decidir, calcular e simular, discutir e projetar” (ROCHA; BARANAUSKAS, 2003, p. 13).

Os algoritmos gráficos de computador, por exemplo, são apenas os algoritmos que dão certas experiências para o aparelho de percepção humano. A concepção de muitas aplicações informáticas inevitavelmente irá requerer a construção de componentes que irão interagir com o usuário. Portanto, é fundamental entender sobre a funcionalidade que um sistema terá, como trazer isso para o usuário, como construir o sistema e como testar o projeto (ACM SIGCHI, 1992).

Como os estudos da IHC se relacionam com o ser humano e a máquina, é necessário ter conhecimentos a respeito de ambos os lados. Do lado da máquina devem-se conhecer técnicas de computação gráfica, sistemas operacionais, linguagens de programação e ambientes de desenvolvimento. No lado humano, teoria da comunicação, disciplinas de *design* gráfico e industrial, linguística, ciências sociais, psicologia cognitiva e desempenho humano.

Nesse contexto, é importante que várias etapas relacionadas ao desenvolvimento das interfaces sejam estudadas e analisadas de forma a manter o indivíduo como objeto central do processo. Dessa forma, além da preocupação relacionada ao potencial de uso das interfaces

pelo usuário (avaliação de Usabilidade), o próprio processo de sua produção (projeto de *software*) deve ser avaliado.

2.1.1 Histórico da IHC

A IHC surgiu como um campo ligado à computação gráfica, a sistemas operacionais, a fatores humanos, à ergonomia, à engenharia industrial, à psicologia cognitiva e a sistemas de informática. A computação gráfica, por sua vez, nasceu do uso do CRT (do inglês, *cathode ray tube* - tubo de raios catódicos) e de canetas eletrônicas, desde cedo na história dos computadores, o que conduziu a várias técnicas utilizadas na IHC. Muitas dessas técnicas datam de 1963, quando uma tese de Sketchpad Sutherland marcou o início da computação gráfica como uma disciplina (ACM SIGCHI, 1992).

Uma série de artefatos contribuiu como importantes elementos para a IHC, como o *mouse*, a tela de *bitmaps*, os PCs, o *windows(r)* e a metáfora da mesa de trabalho (*desktop*) (BAECKER; BUXTON, 1987, *apud* ACM SIGCHI, 1992).

Enquanto isso, os trabalhos em sistemas operacionais desenvolviam técnicas para a interface de entrada e saída, para o ajuste do tempo de resposta do sistema aos tempos de interação humana, para o multiprocessamento e para o suporte a ambientes com janelas e animações.

O crescimento do mercado de massa de PCs influenciou na produção de computadores mais diretamente ligada à qualidade de suas interfaces do que ocorria no passado. O resultado foi a evolução gradual de uma arquitetura de *softwares* e, conseqüentemente, a expansão de pesquisas voltadas ao desenvolvimento de técnicas de produção e testes de interfaces.

2.2 O processo da produção da interface e modelos de Ciclo de Vida

O processo de produção de uma interface¹⁷ ou o *design* de interação, como é denominado por Preece, Rogers e Sharp (2005, p. 29), ou ainda, o *design* de *software*, como é apresentado por Rocha e Baranauskas (2003, p. 114), é determinado pelo modelo de Ciclo de Vida. Este termo compreende um conjunto de atividades e maneiras como elas se relacionam e também como ocorre o fluxo de trabalho entre os blocos de produção. Os diferentes modelos existentes apresentam níveis diferenciados de sofisticação e complexidade e permitem à equipe de produção uma visão geral do desenvolvimento do projeto (PREECE; ROGERS; SHARP, 2005, p. 203).

Esses modelos podem ser classificados basicamente como derivados da Engenharia de *Software* e dos estudos originados da IHC.

2.2.1 Modelo de Ciclo de Vida simples

Um modelo de Ciclo de Vida simples é constituído por diferentes etapas interligadas, como mostra a Figura 1¹⁸. Esse modelo incorpora a iteração (repetição de ciclos de operações dentro de uma cadeia de processos) e encoraja o foco centrado no usuário. O início do Ciclo de Vida da produção ocorre na identificação de necessidades e requisitos¹⁹. Em seguida, alguns *designs* alternativos são criados a fim de alcançar essas necessidades e requisitos. Na sequência, são desenvolvidas e avaliadas versões iterativas.

O *feedback* das avaliações proporciona a identificação de novas necessidades ou refinamento dos requisitos. As etapas se repetem até que o produto final se concretize. Esse

¹⁷ Produção de interface, *design* de interação ou *design* de *software* são denominações dadas por diferentes autores para a produção do mesmo objeto, que é o *software*. Assim, quando surgirem no texto as expressões interface ou *software*, estaremos nos referindo ao mesmo produto. Na produção de interfaces no *Adobe Flash*, esse objeto também será denominado animação.

¹⁸ Fonte: Adaptado de Preece, Rogers e Sharp (2005, p. 206).

¹⁹ *Requisito* consiste em uma declaração sobre um produto pretendido que especifique o que ele deveria fazer ou como deveria operar (Rocha; Baranauskas, 2003).

ciclo é limitado pelos recursos disponíveis, no entanto, o *software* deve respeitar os critérios de Usabilidade prescritos no projeto.

Figura 1 – Um modelo simples de *design* de interação.

2.2.2 Modelos de Ciclo de Vida em Engenharia de *Software*

Antes do modelo de Ciclo de Vida do tipo Cascata ter sido proposto, em 1970, não havia uma abordagem consensual sobre o desenvolvimento de um *software*. Mas com o passar do tempo, muitos modelos foram sendo criados, alguns deles com foco centrado no usuário. Os modelos de ciclos apresentados a seguir são representativos daqueles utilizados no mercado e mostram como o desenvolvimento de *softwares* foi mudando gradativamente, a fim de incorporar um processo mais iterativo centrado no usuário.

2.2.2.1 O modelo Cascata de Ciclo de Vida

Foi o primeiro modelo de engenharia de *software* e forma a base de muitos modelos de Ciclo de Vida em uso hoje em dia. É um modelo linear em que cada passo deve ser completado antes que o próximo possa ser dado. Uma das principais falhas dessa abordagem é o de ignorar o fato de que os requisitos se alteram durante o processo de trabalho.

Figura 2 – O modelo Cascata de Ciclo de Vida do desenvolvimento de *software* ²⁰.

Uma ideia incorporada a esse modelo é o de *feedback* relativas às etapas anteriores. No entanto, não se encontram nesse modelo o processo de iteração nem a participação do usuário para a avaliação do sistema.

²⁰ Fonte: Adaptado de Preece, Rogers e Sharp (2005, p. 208).

2.2.2.2 O modelo Espiral de Ciclo de Vida

Em 1988, Barry Boehm (1988, *apud* PREECE; ROGERS; SHARP, 2005, p.207) sugeriu o modelo Espiral de desenvolvimento de *software*. Como características desse modelo, encontram-se a análise de risco e a prototipagem (produção de protótipos do *software*). Estes são incorporados à estrutura de trabalho de modo iterativo, permitindo que as ideias e o progresso sejam repetidamente verificados e avaliados.

Diferente do modelo Cascata, o Espiral promove a consideração de diferentes alternativas durante o projeto e a reavaliação de estágios onde problemas são verificados.

Figura 3 – O modelo Espiral de Ciclo de Vida do desenvolvimento de *software* ²¹.

²¹ Fonte: Adaptado de Preece, Rogers e Sharp (2005, p. 209).

2.2.3 Modelos de Ciclo de Vida em IHC

Além dos modelos de Ciclo de Vida derivados da engenharia de *software*, o *design* de interação também emergiu de trabalhos da IHC, sendo que nesse campo o foco no usuário é tradicionalmente mais forte.

A seguir são apresentados dois deles, o modelo de Ciclo de Vida Estrela, centrado na avaliação, e o Ciclo de Vida da Engenharia de Usabilidade.

2.2.3.1 Modelo de Ciclo de Vida Estrela

Em 1989, o modelo de Ciclo de Vida Estrela foi proposto por Hartson e Hix (1989 *apud* PREECE; ROGERS; SHARP, 2005). A ideia emergiu de um trabalho que ambos realizaram, observando como *designers* de interfaces trabalhavam. Eles identificaram dois diferentes modelos de trabalho: o analítico e o sintético. No analítico havia uma organização do tipo de baixo para cima, no sentido do sistema para o usuário. Por outro lado, no sintético, a produção era mais livre e criativa, partindo de uma visão do usuário para o sistema. No trabalho desses pesquisadores observou-se que os *designers* alternavam de um modelo para outro enquanto produziam o projeto.

O Ciclo de Vida Estrela foi inspirado nessa dinâmica de trabalho, porém, sem ordenamento específico das atividades. Elas são interconectadas, podendo ir de uma para a outra, mas passando obrigatoriamente pela etapa de avaliação, que é a atividade central desse modelo de Ciclo de Vida.

Figura 4 – O modelo de Ciclo de Vida Estrela ²².

2.2.3.2 O Ciclo de Vida da Engenharia de Usabilidade

O Ciclo de Vida da Engenharia de Usabilidade foi proposto por Deborah Mayhew em 1999 (MAYHEW, 1999, *apud* PREECE; ROGERS; SHARP, 2005, p. 214). Esse ciclo se caracteriza por apresentar uma descrição bastante detalhada de como realizar testes de Usabilidade em cada parte do projeto. Por esse motivo, esse modelo é considerado útil para aqueles que apresentam pouco conhecimento no processo de avaliação de Usabilidade.

Para Mayhew (1999, *apud* PREECE; ROGERS; SHARP, 2005, p. 214) esse Ciclo de Vida apresenta três blocos de atividades:

- Análise de requisitos;
- Projeto / teste / desenvolvimento (maior estágio e que envolve mais subtarefas);
- Instalação.

²² Fonte: Adaptado de Preece, Rogers e Sharp (2005, p. 213).

Figura 5 – O Ciclo de Vida da Engenharia de Usabilidade, segundo Mayhew (1999 *apud* PREECE; ROGERS; SHARP, 2005). Esse modelo é caracterizado pelo detalhamento dos passos de avaliação de Usabilidade. As guias de estilo, indicadas na Figura, consistem em uma coleção de regras e princípios de *design* específicos que orienta a produção do *software*.

A Figura 5²³ ilustra o nível de detalhamento do modelo do Ciclo de Vida de Engenharia. Várias etapas são submetidas a testes de Usabilidade e, caso não haja aprovação, retorna-se a etapas anteriores até que ocorra avaliação positiva para dar prosseguimento ao processo.

Porém, dada a quantidade de etapas desse diagrama, a própria autora reconheceu que, devido a alguns projetos não possuírem o nível de estrutura apresentado no Ciclo de Vida completo, alguns passos poderiam ser pulados no sistema desenvolvido (PREECE; ROGERS; SHARP, 2005, p. 214).

Figura 6 – O Ciclo de Vida da Engenharia de Usabilidade.

O modelo de Engenharia de Usabilidade é apresentado de modo mais simplificado por Rocha e Baranauskas (2003, p. 120), porém contém os mesmos procedimentos internos do modelo apresentado por Mayhew. Para as autoras, esse modelo (Figura 6²⁴) é constituído por quatro fases:

²³ Fonte: Adaptado de Preece, Rogers e Sharp (2005, p. 215).

²⁴ Fonte: Adaptado de Rocha e Baranauskas (2003, p. 123).

- *Pré-design*
- *Design* inicial
- Desenvolvimento iterativo
- *Pós-design*

A fase de *pré-design* é a etapa preparatória para o desenvolvimento do *software* centrado no usuário. Por isso, se preocupa em conhecer o indivíduo, obter suas características particulares e compreender qual será a sua relação com a interface. É nesta fase, que se deve compreender quais serão os requisitos, os padrões de interface, os *guidelines* (princípio e regras do *design*), quais serão as ferramentas de desenvolvimento etc.

Para atingir o objetivo de conhecer o usuário, vários métodos podem ser utilizados, como visitar o seu local de trabalho, observá-lo em seu ambiente, fazer registros por meio de gravações, coletas remotas de dados, entre outras técnicas.

Durante a fase de *design* inicial é aplicada a especificação inicial da interface a um desenho de *software* que possa ser utilizado pelo usuário. Ou seja, um protótipo que siga os princípios de Usabilidade deve ser verificado empiricamente com usuários reais.

O desenvolvimento iterativo é a fase de *feedback* de testes que orienta a produção do *software* até que o objetivo do projeto seja alcançado. Nessa ação, caso algum problema seja verificado, há a recondução do trabalho ao *design* inicial, levando à criação de novos protótipos, que são submetidos novamente a testes, em mais um ciclo de desenvolvimento.

No *pós-design*, com a interface concluída, é necessário realizar a sua instalação no local de trabalho para o acompanhamento de medidas de reação e aceitação do sistema pelo usuário final.

2.3 Usabilidade

A produção de *softwares* baseado na perspectiva da IHC é centrada no usuário e, como visto no tópico anterior, pode ser orientada a partir dos princípios de Usabilidade.

Para Nielsen e Loranger (2007, p. xvi), a Usabilidade é definida como:

Um atributo de qualidade relacionado à facilidade do uso de algo. Mais especificamente, refere-se à rapidez com que os usuários podem aprender a usar alguma coisa, a eficiência deles ao usá-la, o quanto lembram daquilo, seu grau de propensão a erros e o quanto gostam de utilizá-la. Se as pessoas não puderem ou não utilizar um recurso, ele pode muito bem não existir.

Já para Preece, Rogers e Sharp (2005, p. 35), aplicar os princípios de Usabilidade “implica otimizar as interações estabelecidas pelas pessoas com produtos interativos, de modo a permitir que realizem suas atividades no trabalho, na escola e em casa”.

De modo resumido, para essas autoras, a Usabilidade estaria dividida nas seguintes metas:

- ser eficaz no uso (eficácia)
- ser eficiente no uso (eficiência)
- ser de boa utilidade (utilidade)
- ser fácil de aprender (*learnability*)
- ser fácil de lembrar como se usa (*memorability*)

Outra meta que poderia ser incluída, e que é descrita por Tubbs (2003), seria:

- ser receptivo ao ensino (*teachability*)

As metas de Usabilidade, aplicadas ao processo de desenvolvimento de interfaces, garantiriam a sua produção vinculada às necessidades do usuário.

2.4 Avaliação das interfaces: inspeção de Usabilidade e teste de Usabilidade

A avaliação de uma interface não é uma atividade a ser aplicada apenas ao final do processo de sua produção. Ela deve ser realizada nos diferentes momentos do Ciclo de Vida do projeto.

Uma avaliação busca principalmente conhecer o que querem e quais problemas enfrentados pelos usuários quando eles usam um *software*. Em muitos estágios da produção é necessário verificar se o modelo idealizado no projeto inicial é realmente aquele que o usuário necessita e se está funcionando como o planejado.

Segundo Rocha e Baranauskas (2003, p. 161) a avaliação pode ser dividida em três objetivos: avaliar a funcionalidade do sistema, avaliar o efeito da interface junto ao usuário e identificar problemas específicos do sistema.

A funcionalidade do sistema está relacionada ao desenho da interface, que deve estar adequado às tarefas do usuário, ou seja, permitir ele efetue a tarefa pretendida de modo fácil e eficiente.

Avaliar o impacto do *design* junto ao usuário significa avaliar a Usabilidade da interface, ou seja, verificar quão fácil é aprender com o sistema; observar a atitude do usuário ao utilizar a interface; identificar se há regiões do *software* com problemas de interação, etc.

A identificação de problemas específicos com o *design* significa avaliar aspectos da interface que quando usados causam resultados inesperados ou confusos entre os usuários.

Segundo Rocha e Baranauskas (2003, p. 161), para alcançar esses objetivos é necessário realizar avaliações que podem ser apresentados em dois grupos de métodos:

1. *Inspeção de Usabilidade*, que não envolve o usuário e que pode ser realizado em qualquer momento do desenvolvimento do *software*; e
2. *Testes de Usabilidade*, que são as avaliações centradas no usuário e aplicadas a partir de métodos experimentais ou empíricos, métodos de observação ou questionamento, etc.

A inspeção de Usabilidade é baseada em um conjunto de ações para exame da Usabilidade da interface e é realizada por especialistas, como consultores de desenvolvimento

de *software* ou de usuários. Esse processo ocorre para que sejam verificados problemas durante a produção da interface e que sejam feitas recomendações para a melhoria de Usabilidade do sistema. Existem diferentes métodos de inspeção, sendo que os principais são a avaliação heurística, a revisão de *guidelines*, a inspeção de consistência e o percurso cognitivo.

A avaliação heurística se baseia na inspeção da interface a partir de uma lista pré-definida de heurísticas de Usabilidade. Essa lista é dividida em grupos que descrevem propriedades comuns de interfaces usáveis. Segundo Nielsen (2007) essa metodologia se caracteriza por ser fácil de ser aplicada, rápida de ser executada e de custo reduzido.

Por outro lado, na revisão de *guidelines* a interface é checada com uma lista de itens de Usabilidade que pode formar uma sequência de mil *guidelines*. Por isso, é um método muito raro de ser utilizado.

Na inspeção de consistência a avaliação ocorre checando o *software* a partir de sua constituição estrutural, como cores, *layout*, formatos de entrada e saída de dados.

No caso do percurso cognitivo, o avaliador simula a ação do usuário atuando com a interface e executando tarefas típicas. As informações sobre o funcionamento e erro são coletadas pela exploração que se faz em todo *software*.

2.5 Testes de Usabilidade

Os testes de Usabilidade visam avaliar a interface segundo metodologias centradas no indivíduo. Para realizar a avaliação é necessário conhecer o usuário, o que eles querem e quais os problemas que experimentam. Quanto mais informações forem obtidas dos usuários, melhor será o *design* da interface.

Segundo Rocha e Baranauskas (2003, p.202) ao aplicar os testes de Usabilidade, as seguintes questões devem ser respondidas:

- O objetivo do teste: o que se deseja obter?
- Quando e onde o teste irá acontecer?
- Qual a duração prevista de cada sessão de teste?

- Qual o suporte computacional necessário?
- Qual *software* precisa estar à disposição?
- Qual será o estado do sistema no início do teste?
- Quem serão os experimentadores?
- Quem serão os usuários e como serão conseguidos?
- Quantos usuários são necessários?
- Quais as tarefas que serão solicitadas aos usuários?
- Qual critério será utilizado para definir que os usuários terminaram cada tarefa corretamente?
- Quanto o experimentador poderá ajudar o usuário durante o teste?
- Quais dados serão coletados e como serão analisados uma vez que tenham sido coletados?
- Qual o critério para determinar que a interface é um sucesso?

Os testes de Usabilidade devem ainda ser verificados segundo os critérios de confiabilidade e de validade.

Por confiabilidade deve-se entender se o mesmo resultado será obtido caso o teste seja repetido outras vezes. Nesse caso, questões de diferenças individuais devem ser relevadas, já que usuários diferentes podem ter tempos de respostas diferentes, por exemplo, nos casos em que o teste se refere à velocidade de manipulações.

Os testes também devem assegurar sua validade, ou seja, os resultados devem refletir os aspectos de Usabilidade que se quer testar. Ou seja, o produto avaliado no laboratório deve atender às expectativas do usuário real.

2.5.1 Método etnográfico de coleta de dados

Para Suchman (1987, *apud* ROCHA; BARANAUSKAS, 2003, p. 142) o ideal da investigação da tecnologia é o método etnográfico (aquele em que a pesquisa é realizada onde o trabalho ocorre naturalmente em cenários construídos pelos próprios participantes). Nessa

perspectiva é possível realizar uma avaliação centrada no indivíduo e, dessa forma, compreender o trabalho que se faz junto à interface do ponto de vista do próprio usuário.

A investigação no local de trabalho também permite que a avaliação se direcione a um objeto, para verificar como a interface se relaciona com o usuário. Na análise de Usabilidade de um protótipo de sistemas, por exemplo, uma câmera pode captar toda a sequência de ações de um usuário interagindo com o sistema. O objetivo, neste caso, é analisar a interface com respeito à adequação ao trabalho realizado (ROCHA; BARANAUSKAS, 2003, p. 142).

Quanto à observação das ações que ocorrem no ambiente de trabalho avaliado, esta pode ser realizada direta ou indiretamente²⁵.

Na observação direta, os usuários são observados fazendo seu trabalho normal ou desenvolvendo tarefas específicas. Fazer anotações é a técnica mais simples de se coletar dados. Nesse método, o observador faz registros de como o usuário utiliza a interface, possibilitando investigar a rotina de uso do *software* ou anotando comportamentos interessantes realizados pelo indivíduo.

No entanto, a observação direta é considerada invasiva, já que o usuário tem conhecimento de que está sendo observado e sua performance, sendo monitorada. Com isso, o indivíduo pode ter seu comportamento afetado, acarretando em um desvio dos dados observados em relação àqueles desejados.

Na observação indireta os registros são gravados (em vídeo, por exemplo) para uma posterior análise. Esse processo cria uma distância maior entre o observador e o usuário, minimizando o efeito invasivo da observação direta. O registro da observação indireta pode ainda ser realizado com outras câmeras, que permitiriam focalizar diferentes aspectos da interação do usuário com a interface, como por exemplo, a ação no teclado ou olhar do indivíduo para tela do computador, informações essas que comporiam um cenário mais completo para a avaliação da interação do usuário com o *software*.

²⁵ Preece, Rogers e Sharp (2005, p. 396) utilizam o termo “observação indireta” em outro contexto. Para as autoras esse método se refere à coleta de dados a partir de rastreamento de informações, que pode ser obtido a partir de dados registrados em diários ou então a partir de *Logs* das interações. Aqui foi adotado o termo no contexto utilizado por Rocha e Baranauskas (2003, p. 144).

2.5.2 Coleta de dados

A coleta de dados constitui uma parte importante da atividade de identificação de requisitos e também da avaliação. O propósito da coleta de dados é reunir informações suficientes, relevantes e apropriadas, de forma que um conjunto de requisitos definidos possa ser produzido.

Existem diferentes técnicas de coleta de dados que podem ser combinadas e estendidas de muitas formas.

Tais técnicas constituem-se de questionários, entrevistas, grupo de foco e *workshops*, observação natural e estudo da documentação. Preece, Rogers e Sharp (2005, p. 230) enumeram as seguintes técnicas de coleta de dados:

1. Questionário. Série de questões para obter informações específicas do usuário. Pode ser apresentada em papel, ser requisitada via e-mail ou então estar presente em uma página da *web*.
2. Entrevista. Perguntas feitas geralmente face a face, ainda que não necessariamente. Objetivam descobrir o que o usuário gosta ou não gosta. Quando realizada diretamente com o avaliador, a entrevista encoraja as pessoas a responder as questões e pode tornar a atividade mais agradável.
3. Grupos de estudos específicos e *workshops*. Ao contrário da entrevista, esta técnica reúne mais pessoas o que permite encontrar uma visão mais consensual e ressaltar áreas de conflito e discordância.
4. Observação natural (método etnográfico). Avaliar o usuário no local em que interage com a interface normalmente, enquanto realizam suas tarefas diárias, observando o trabalho como realmente acontece. Esse é um importante método de registrar comportamentos peculiares que podem complementar outras investigações.
5. Estudo da documentação. Técnica baseada no estudo dos registros dos procedimentos anotados em relatórios de trabalho.

2.5.3 Rastreamento remoto de atividades dos usuários (*Log de dados*)

Algumas vezes os métodos de observação direta não permitem um registro completo das ações do usuário, já que nem sempre é possível manter um observador em tempo integral no local de trabalho do indivíduo. Gravações em vídeo também podem não ser totalmente eficazes na observação de uma particular interação do usuário com a interface.

Além dos métodos tradicionais de coleta de dados, há testes que não se limitam à observação direta do usuário no espaço dos laboratórios ou em seus locais de trabalho. Trata-se de métodos de rastreamento remoto que usam a rede como uma ligação ao ambiente de trabalho do indivíduo. Nesses mecanismos o observador e o usuário estão separados em tempo e espaço (Hartson et al, 1996).

Um método de rastreamento remotos atualmente em uso é o *Log de dados*, *software* especificamente construído para coletar dados de interação. Uma interface, por exemplo, pode ser preparada para registrar informações sobre as atividades dos usuários, que são armazenadas em um banco de dados, e que podem ser examinadas posteriormente.

O *Log de dados* não requer a presença do pesquisador e parte do processo e análise pode ser automatizada. Já existem várias ferramentas que utilizam essa técnica, como as que registram cada tecla que o usuário pressiona e o tempo exato de um evento. Os contadores que registram o número de visitas a um *website* são bastante familiares e são um exemplo de *software* que utiliza o *Log de dados* (como é o caso do *Google Analytics*²⁶).

2.6 Objeto de aprendizagem

No âmbito das interfaces produzidas com o objetivo de serem utilizadas no ensino, destaca-se o objeto de aprendizagem. O objeto de aprendizagem (*learning object*) é definido, segundo o Institute of Electrical and Electronics Engineers - Learning Technology Standards

²⁶ O *Google Analytics* é uma ferramenta do *Google* que realiza a análise de audiência de páginas da internet. O uso desse recurso é gratuito e pode ser obtido através do endereço <http://www.google.com/analytics>.

Committee ²⁷, como “qualquer entidade, digital ou não digital, que possa ser usada na aprendizagem, na educação e no treinamento” (IEEE, 2002, tradução nossa).

Diferentes autores apresentam diferentes definições para os objetos de aprendizagem (OA²⁸). Essas definições procuram evidenciar as características desses objetos como elementos apoiadores no processo de ensino e aprendizagem.

Muzio, Heins e Mundell (2001, p. 2), por exemplo, relatam os seguintes exemplos de abordagens de OA:

- A Cisco²⁹ (2001) define um OA como “um pedaço granular e reutilizável de informação, que é uma mídia independente. “
- Sul e Monson (2001) na Universidade Brigham Young usam o termo “objeto de mídia” e o define como “um mídia digital que foi concebida e / ou utilizada para finalidade de ensino. Esses objetos vão desde mapas e gráficos até demonstrações em vídeo e simulações interativas”.
- Wiley (2001) define objetos de aprendizagem como “elementos de um novo tipo de instrução baseada em computação”.
- Os Internettime.com (2001) define como “pequenos pedaços (grânulos, objetos)”, que “são rotulados para que sistemas possam automaticamente misturá-los e combiná-los para montar e propiciar uma experiência individualizada de aprendizagem”. Esses objetos apresentam uma hierarquia, que vão desde simulação por unidade tópica e objeto de aprendizagem reutilizável até objetos de informação.

²⁷ IEEE – LTSC desenvolve padrões técnicos, recomendações práticas e guias para a aprendizagem tecnológica. Foi criada em 1996 para desenvolver e promover padrões de tecnologia instrucional (IEEE, 2002).

²⁸ Nos ambientes digitais os objetos de aprendizagem são denominados *elearnig objects* (ELO, por exemplo, em Muzio, Heins e Mundell, 2001, p. 2). Porém, para todos os efeitos, nesse trabalho, ao nos referirmos aos *elearning objects*, trataremos essa entidade simplesmente como objetos de aprendizagem (OA).

²⁹ A Cisco Systems é uma companhia multinacional cuja atividade principal é o oferecimento de soluções para redes e comunicações (disponível em :<<http://pt.wikipedia.org/wiki>>. Acesso em: 20.jul.2012).

Para Willey (2001, p. 2) a idéia fundamental por trás dos OA é a possibilidade de construção de pequenos componentes instrucionais que podem ser reutilizados várias vezes em diferentes contextos de aprendizagem. Além disso, os objetos de aprendizagem são compreendidos normalmente como entidades digitais veiculados pela Internet, o que significa que qualquer número de pessoas pode acessá-los e usá-los simultaneamente (Wiley, 2001, p. 3).

Para Pimenta e Batista (2004, p. 102), o OA é uma “unidade de aprendizagem de pequena dimensão, desenhada e desenvolvida de forma a fomentar a sua reutilização, eventualmente em mais do que um curso ou em contextos diferenciados, e passível de combinação e/ou articulação com outros objetos de aprendizagem de modo a formar unidades mais complexas e extensas.” (destacando a característica da reusabilidade do OA, esses autores denominam essa entidade como Objeto de Aprendizagem Reutilizável - OAR).

A percepção de unidade que pode ser combinada para compor estruturas mais complexas é apoiada por Soares, Sá e Machado (2007). Para esses autores, os OA são “recursos digitais, usados, reutilizados e combinados com outros objetos para formar um ambiente de aprendizado rico e flexível”. Dessa forma, a ideia é que os objetos sejam blocos com os quais será construído o contexto de aprendizagem (TAROUÇO, 2003, p. 2).

Soares, Sá e Machado (2007) destacam ainda que os OA podem ser produzidos em qualquer mídia ou formato. Monteiro (2006, p. 7) exemplifica algumas tecnologias para a construção dos OA: *Java Applets*, *Adobe Flash*, *Modellus* e *Javascript*.

Tarouco (2003) observa que os recursos de aprendizagem digital são construídos a partir da combinação de HTML, Java e outras linguagens e ferramentas de autoria, sendo que nelas podem ser incluídos jogos, textos, áudio, vídeos, gráficos, imagens etc.

2.6.1 Características dos Objetos de Aprendizagem

Para que possam ser utilizados como ferramentas no ensino e fazerem parte de um ambiente de aprendizagem, onde os usuários possam encontrá-los e acessá-los facilmente, os OA devem apresentar características próprias que os definam.

Nunes (2011, p. 112), a partir de estudos de trabalhos de Assis (2005), Silva (2006), Machion (2007) e Oliveira (2009), e Mendes, Souza e Caragnato (2007), levanta as seguintes características de um AO:

Reusabilidade

Permite que o OA seja reutilizado em contextos diferentes daquele para o qual foi construído, por diferentes usuários e com variados propósitos.

Flexibilidade ou adaptabilidade

Os OA devem ser construídos de forma que possuam início, meio e fim. Possibilita que os OA possam ser criados para um determinado contexto e depois reutilizados em contextos similares.

Granularidade

Evidencia os diferentes tamanhos que deverá ter um OA, de forma que ele possa ser agrupado a outros OA, fazendo blocos maiores de conteúdos, disciplinas e/ou estruturas tradicionais de cursos. Quanto menor for a granularidade de um OA maior é seu potencial de reuso.

Modularidade

Possibilidade de conter outros OA ou estar contido em um ou mais objetos ou cursos.

Interatividade

Possibilita ao usuário interagir com o conteúdo do OA de alguma forma, vendo, ouvindo e/ou respondendo a alguma solicitação proposta pelo objeto.

Conceituação

Quando há estabelecimento de vínculos entre OA e os conteúdos a serem abordados no processo de ensino-aprendizagem.

Acessibilidade ou recuperabilidade

Possibilidade do OA ser de facilmente acessado, via Internet, para serem usados em diversos locais pelos usuários.

Identificação por metadados³⁰

Capacidade do OA ser facilmente localizado por mecanismos de busca. Para isso o conteúdo do objeto deve ser descrito através de metadados.

Interoperabilidade ou portabilidade

Habilidade do OA operar através de uma variedade de *hardware*, sistemas operacionais e *browsers*, sem necessidade de alteração.

Durabilidade

Possibilidade de continuar a ser usado, independente da mudança de tecnologia.

Os OA são normalmente armazenados em repositórios, que são bancos de dados disponíveis na Internet. Disponibilizados dessa maneira, ficam facilmente acessíveis para serem utilizados para comporem ambientes virtuais de aprendizagem.

2.7 Produção de OA no modelo de Ciclo de Vida de *Software*

Apesar de diversos objetos acessíveis pela internet poderem ser caracterizados como OA, nem todos correspondem aos exemplos de interfaces, na perspectiva dos modelos de Ciclo de Vida de *Software* apresentados.

Por isso, apesar de documentos de texto (formato DOC, por exemplo), apresentações (por exemplo, PPT), imagens (JPG, GIF), áudio (mp3, midi, wma) e vídeo (mp4, avi) serem objetos com possibilidade de conterem metadados, poderem ser usados e reutilizados e possuírem uma granularidade que permita a sua combinação para a formação de um ambiente de aprendizagem, não são objetos que propiciam uma interação como imaginada para as interfaces desenvolvidas para esse projeto.

³⁰ Metadados são informações sobre um objeto, seja ele físico ou digital, que aumentam a possibilidade para encontrá-los, gerenciá-los e usá-los, a partir de um mecanismo de busca ou de um repositório. Como o número de objetos cresce exponencialmente e as necessidades de aprendizagem expandem dramaticamente o uso de metadados para a identificação de um objeto se torna cada vez mais essencial. (IEEE, 2002).

Exemplos de objetos típicos, produzidos de acordo com os modelos de Ciclo de Vida de *Software*, são os arquivos HTML, os arquivos PHP e os aplicativos em *Shockwave Flash*³¹, que podem ser observados e analisados segundo diretrizes de Usabilidade.

O HTML (*HyperText Markup Language* ou Linguagem de Marcação de Hipertexto) é uma utilizada para produzir páginas na *web* e que são interpretadas por diferentes tipos *browsers*³² (Internet Explorer, Chrome, Firefox, Safari, Opera, entre outros). A linguagem utilizada possibilita construir *links* para acessar outros textos HTML ou outros objetos, dando origem a uma leitura hipertextual. As páginas em HTML podem ser editadas simplesmente a partir do bloco de notas (*windows notepad*), mas com a incorporação de elementos denominados *tags*, é possível desenvolver *layouts* de diferentes graus de complexidade e incorporar vários elementos midiáticos, como texto, imagem, som, vídeo e animação.

PHP (*Hypertext Preprocessor* ou originalmente *Personal Home Page*) é uma linguagem de *script* de fonte aberta³³, utilizada para o desenvolvimento de aplicações *web* e embutível dentro do HTML. Possui sintaxe de programação muito parecida com a linguagem C e com a C++ e, assim como o HTML, pode ser editada em um bloco de notas. O PHP se caracteriza por ter um código que é executado no servidor³⁴ de rede, gerando uma página em HTML que é então enviada para o usuário. O cliente, portanto, recebe os resultados da execução desse *script* sem saber como é o código fonte. Isto torna o programa seguro para a comunicação com bancos de dados, cujo acesso deve ser privado e cuja entrada e saída de informação devem ser controladas mediante o uso de senhas que não devem ser observados na página do usuário.

³¹ Há diversos outros tipos de objetos cuja produção pode ser classificada pelos modelos de ciclo de vida, como aplicativos construídos em Java ou C++. No entanto, o HTML, o PHP e o *Flash* são os exemplos dos recursos utilizados nesse projeto.

³² Navegadores de Internet.

³³ Fonte aberta, do inglês *open source* é um conjunto de princípios e práticas sobre como escrever um *software*. Significa que o código fonte do *software* está disponível aos interessados, tanto programadores como para o público em geral, possuindo restrições intelectuais de cópia e reprodução reduzidas ou inexistentes quanto aos valores referente à propriedade intelectual ou ao direito autoral (disponível em: <<http://pt.wikipedia.org>>. Acesso em: 20.jul.2012).

³⁴ O servidor é um sistema de computação que fornece serviços a uma rede de computadores. Esses serviços podem ser o envio de dados de uma página da internet ou o do correio eletrônico. Os computadores que acessam os serviços de um servidor são chamados clientes (disponível em: <<http://pt.wikipedia.org>>. Acesso em: 20.jul.2012).

O *Adobe Flash* (assim denominado após a fabricante de *softwares Adobe* ter adquirido a marca da *Macromedia*) é um *software* proprietário³⁵ para a produção de animações denominadas *Shockwave Flash* (identificadas pela extensão *swf*). Uma animação em *Shockwave* se caracteriza por ser interativa (recursos acionados por mouse ou teclado), multimodal (permite o uso de texto, imagem, vídeo, som), programável (através da linguagem *action script*) e poder ser incluída em páginas HTML. As animações podem ser visualizadas em uma página *web* usando um navegador (através de um *plug-in* especial) ou através do *Flash Player*, que é um aplicativo de leitura, distribuído gratuitamente pelo *Adobe*.

O uso do *Flash* para aplicações na Internet e para a produção de objetos de aprendizagem não é unânime. Conforme Reis (2009, p. 4), há profissionais que o adotam, pois acreditam nas potencialidade dessa ferramenta e desenvolvem aplicações para diversos tipos de trabalho na Internet. Por outro lado, existem aqueles que não aceitam o seu uso, principalmente devido a problemas de Usabilidade³⁶ – limitações com acessibilidade e com a incorporação de metadados que permitam a sua busca pela internet (Reis, 2009, p. 4).

No entanto, o *Flash* é muito utilizado para facilitar a produção de projetos que não estão naturalmente incorporadas à plataforma HTML, como é o caso da produção de multimídia. Esse *software* permite a realização de animações de modo rápido a partir de um assistente de edição incorporado e, com o *action script*, é possível elaborar aplicativos complexos, como por exemplo, aqueles que necessitam de cálculos numéricos (como é o caso de simulações na área da física).

³⁵ *Software* proprietário ou não livre é aquele cuja cópia, redistribuição ou modificação são em alguma medida restritos pelo seu criador ou distribuidor (disponível em: <<http://pt.wikipedia.org>>. Acesso em: 20.jul.2012).

³⁶ No ano de 2012 a Adobe pretende descontinuar o suporte ao *Flash*, principalmente devido a problemas de Usabilidade em plataformas móveis (celulares e *tablets*), como apontado por representantes da Apple, fabricante que não permitia a exibição dessa ferramenta em seus equipamentos (ROMANI, 2012).

2.8 Usabilidade na web

A construção de um *website* também deve ser submetida a diretrizes de Usabilidade, conforme orienta Nielsen (2002, p. 17). Essas diretrizes são baseadas em três níveis de pesquisa:

- Comportamento geral do usuário diante da maioria dos *websites*, observado como um padrão repetidas vezes;
- Descobertas especializadas sobre gêneros específicos de sites ou áreas dos sites, identificadas a partir da análise de relatórios;
- Descobertas detalhadas sobre um site específico e seus clientes: algumas observações são tão específicas de um site que não são abrangidas em nenhum tipo de relatório e por isso merecem estudos que não estão compreendidos nas diretrizes gerais.

Baseado no comportamento dos usuários diante de *websites*, Nielsen (2002) descreveu 113 diretrizes baseadas em diversos aspectos relacionadas a esse tipo de interface, tais como *design*, conteúdo, navegabilidade, pesquisa, recursos multimídia, publicidade e relação com o cliente (ver Anexo A).

Diretrizes de Usabilidade também são suscetíveis às mudanças tecnológicas e a adaptação dos usuários às interfaces. Por exemplo, com o aprimoramento da tecnologia, dificuldades sofridas pelos usuários teriam sido atenuadas e, portanto, algumas diretrizes passaram a oferecer menos problemas aos usuários.

Diante desse fato, Nielsen (2007, p. 57, p. 351) fez revisões em sua lista, constatando melhora no padrão de Usabilidade de algumas de suas diretrizes. Essas observações são importantes, já que um OA produzido hoje não seria tão eficiente se fosse utilizado com uma tecnologia ultrapassada. A seguir são apresentadas algumas dessas diretrizes que tiveram seu conceito melhorado e que são relevantes para a produção de um OA.

Tempo de *download*

A largura de banda da rede tem aumentado e com isso, a velocidade de transferência de dados na Internet. Assim, arquivos que antes demoravam para ser carregados fluem com mais rapidez.

Flash

Para o autor, a novidade do *Flash* (em meados da década de 1990) encorajava abusos no *design* de interfaces. *Websites* inteiros eram construídos com essa ferramenta e os recursos de multimídia, utilizados para enfeitar e carregar a página com informações desnecessárias. Contudo, o tempo foi passando e os *designers* foram abandonando o *Flash* em troca de páginas mais leves e com melhores critérios de Usabilidade.

Multimídia e vídeos

O aumento de largura de banda favoreceu o uso de recursos multimídia e de vídeos, constituídos por arquivos mais pesados. No entanto, para Nielsen (2007, p. 92), a mudança principal ocorreu com o surgimento de produções especialmente realizadas para o ambiente *web*, ao contrário do que ocorria anteriormente, quando eram apenas adaptados conteúdos televisivos para a Internet.

Layouts congelados

Uma das diretrizes de Usabilidade faz referência ao rolamento da barra do *browser*. Alguns conteúdos são mais extensos do que a visualização permitida. Assim, o usuário é obrigado a fazer rolagens de página para ter acesso ao resto da informação presente. Segundo Nielsen (2002, p. 23), no entanto, os usuários tendem a visualizar, com maior frequência, apenas o conteúdo contido “acima da dobra³⁷”. Assim, *layouts* congelados, ou seja, de tamanhos fixos, poderiam ter sua Usabilidade comprometida, pois alguns usuários tenderiam a desprezar o resto da informação não visualizável.

³⁷ Essa é uma referência à dobra do jornal que divide a leitura em duas partes: acima da dobra e abaixo da dobra (NIELSEN, 2002, p. 23).

Em algumas situações, por outro lado, é possível utilizar o “*layout líquido*” de página. Essa técnica permite que o conteúdo se ajuste para qualquer tamanho do *browser*, possibilitando a visualização completa da informação. Esse é um recurso desejável, por exemplo, para a observação de imagens inteiras.

Incompatibilidade entre várias plataformas

Atualmente a linguagem HTML, o PHP e o *Shockwave Flash* podem ser lidos em diversos tipos de *browsers*, reduzindo o risco de determinado conteúdo não ser visualizado ou de ter sua aparência modificada em diferentes plataformas. Essa é uma diretriz, portanto que teve seu risco reduzido.

No entanto, essas são revisões aplicadas aos ambientes oferecidos pelos computadores pessoais (*desktop* ou *notebook*). Com a presença cada vez maior de dispositivos móveis, tal como celulares e *tablets*, que possuem diferentes sistemas operacionais, tamanhos de tela e tecnologias de interação, novas diretrizes de Usabilidade devem ser verificados.

2.9 Resumo

A avaliação de uma interface, como um OA, por exemplo, pode ser realizada a partir de diferentes perspectivas. Uma delas se baseia na análise dos métodos de produção de *softwares*, ou seja, no Ciclo de Vida de *Softwares*. Diferentes modelos, de complexidades variadas, têm seu processo centrado no usuário, elemento para o qual deve ser desenvolvido o modo de interação com o produto, como propõe o campo da IHC.

Modelos de Ciclo de Vida centrados no indivíduo implicam em metodologias de avaliação que respeitam o modo como o usuário percebe o *software*, o que é definido por diretrizes de Usabilidade.

Os métodos de coletas de dados, que fornecem subsídios para fazer a avaliação das interfaces, devem objetivar a realidade de uso desses *softwares* e, portanto, os usuários devem ser observados no seu ambiente natural de trabalho.

A estrutura completa para a avaliação do *software*, desde a sua produção até sua utilização pelo usuário, conforme o modelo teórico apresentado, pode ser complexa, composta por várias etapas, envolver diversos profissionais e despende um grande orçamento e tempo de trabalho.

No entanto, a construção das animações desse projeto não seguiu todas as etapas descritas nos modelos de Ciclo de Vida de *Software* apresentados, devido à estrutura em escala reduzida da produção das interfaces, com prazos e custos limitados (além de ter sido realizado por um único profissional).

Mesmo assim, foi possível distinguir algumas linhas metodológicas discutidas nesse capítulo e coletar e analisar dados a partir dos métodos descritos. O processo completo da produção das interfaces desse projeto e as particularidades de cada animação são apresentados no capítulo seguinte.

3. METODOLOGIA

O processo de construção de interfaces pode ser estruturado conforme os modelos de Ciclo de Vida de *Software* (PREECE; ROGERS; SHARP, 2005, p. 203). Quando centrado no usuário, sua avaliação pode se basear em diretrizes de Usabilidade (Nielsen, 2007) aplicadas em diferentes etapas da produção da interface (ROCHA; BARANAUSKAS, 2003, p. 105). Na etapa do desenvolvimento iterativo (em que a produção entra em ciclo de (re) criação e (re) avaliação) pode-se adotar o método de coleta de *Log* dados para obter registros remotos da interação do usuário com o *software*.

A produção do Laboratório Virtual da Estação Ciência segue, em menor ou maior grau, os métodos descritos, o que permite avaliar a criação de suas interfaces dentro da perspectiva de Ciclo de Vida.

Para melhor compreensão da análise do processo de sua produção, o percurso de criação do Laboratório é dividido em três blocos:

1. Construção do *website* Laboratório Virtual;
2. Produção das interfaces para o projeto apresentado;
3. Preparação para a coleta de informações dos usuários.

Antes de apresentar cada uma dessas etapas, no entanto, é necessário conhecer quais linguagens de programação³⁸ utilizadas possibilitaram a realização desse trabalho.

3.1 Linguagens de programação utilizadas

Foram utilizadas nesse trabalho três linguagens distintas para a programação do *website* e para a produção das interfaces.

³⁸ Uma linguagem de programação é um método padronizado para comunicar instruções para um computador (disponível em: <<http://pt.wikipedia.org>>. Acesso em: 20.jul.2012).

Uma delas foi o HTML (*HyperText Markup Language* ou Linguagem de Marcação de Hipertexto), usada para a construção das páginas de visualização de conteúdo *web*. Essas páginas dão suporte para acesso a *links*, execução de *softwares* e apoio a coletas de dados. A linguagem de programação do HTML baseia-se em etiquetas (*tags*) que permitem a formatação de diferentes elementos gráficos de uma página *web*, como quantidade e tamanho de tabelas, tipologias, cores, assim como a inclusão de imagens, de sons, de vídeos e de animações.

Para a produção das interfaces de animação foi utilizado o *Adobe Flash*. Apesar de vários dos objetos criados anteriormente para o Laboratório Virtual terem sido realizados com o auxílio do assistente de animação automatizada *Motion Tween* e *Shape Tween*, as interfaces produzidas especificamente para esse trabalho foram criadas com o auxílio do *action script*, que é a linguagem de programação incorporada ao *Flash*. A utilização dessa linguagem foi essencial para permitir a conversão de expressões matemáticas (como as equações geométricas para a formação de imagem usadas nesse trabalho) em movimento de objetos.

Além disso, outra possibilidade oferecida pelo *action script* é a troca de dados com o servidor, o que significa que informações internas à animação podem ser direcionadas para outros objetos de uma página *web*, e vice e versa.

A Figura 7³⁹ apresenta o aspecto de uma publicação aberta no *Flash*. Como esse programa foi criado originalmente para o desenvolvimento de animações, o trabalho é estruturado em quadros e camadas na guia denominada *timeline* (parte 1 da Figura 7⁴⁰). Essa organização permite distribuir os elementos em diferentes planos, sendo que as camadas superiores da *timeline* contêm os objetos que estão acima daqueles que estão nas camadas inferiores. Em uma animação clássica⁴¹, são criados desenhos em cada um dos quadros abertos na *timeline*, sendo cada um ligeiramente diferente do anterior. Um dos recursos originais do *Flash* foi a de executar a exibição desses desenhos em sequência, criando a ilusão de uma cena animada.

³⁹ Fonte: Laboratório Virtual. Disponível em: <www.ideiasnacaixa.com/laboratoriovirtual>. Acesso em: 19.jun.2012.

⁴⁰ A configuração dos elementos da área de trabalho do *Flash* é variável, sendo que cada desenvolvedor os distribui conforme a sua necessidade.

⁴¹ Animação denominada *Stop Motion*. As cenas são fotografadas quadro a quadro, podendo ser constituídas por desenhos, personagens de massa de modelar, recorte de papel, etc.

O local em que os desenhos (denominados gráficos) são criados chama-se palco (parte 2 da Figura 7). Nele podem-se aplicar ferramentas básicas de edição, como construção de linhas, figuras geométricas, aplicação de tipologias, alteração de cores de contorno e preenchimento, para criar diferentes tipos de desenhos. Pode-se ainda incluir imagens, sons e vídeo.

Por fim, a área de trabalho indicada pelo número 3 apresenta o editor do *action script*. Uma característica do *script* de programação do *Flash* é que ele pode ser realizado em diferentes locais e sua execução pode ocorrer em diferentes momentos ou em diferentes objetos simultaneamente. Dessa forma, o *script* de programa pode ser feito em qualquer quadro de qualquer camada da *timeline*; caso haja quadros em sequência e dentro deles houver programas editados, eles serão executados um após o outro. O *script* de programação denomina-se *action-frame* quando é realizado dentro dos quadros da *timeline*. A programação também pode ser feita diretamente em um elemento presente no palco; nesse caso, teremos o *action-movie*.

Figura 7 – Áreas de trabalho do programa *Adobe Flash*. No exemplo acima, a edição de uma das animações do Laboratório Virtual.

A terceira linguagem utilizada no trabalho foi o PHP (*Hypertext Preprocessor* ou *Personal Home Page*). Nesse projeto, essa linguagem foi utilizada para que algumas informações das animações em *shockwave* pudessem ser enviadas para um banco de dados MySQL⁴² devidamente configurado⁴³. A visualização dos dados coletados para análise também foi realizada através da programação em PHP, que nesse caso recolhia as informações do banco e as enviava para uma página HTML (apresentando os dados brutos ou então já formatados para amostragens específicas).

A Figura 8 apresenta o esquema de como o *shockwave*, o PHP, o HTML e o banco de dados se intercalam para formar a estrutura de funcionamento do Laboratório Virtual.

Figura 8 – A estrutura do Laboratório Virtual. O usuário interage com a interface em *swf* que está inserida em uma página HTML (bloco à esquerda). As ações realizadas pelo usuário podem resultar em informações que são enviadas para um programa PHP, inserido na página HTML, mas que é executado no servidor. O PHP processa essas informações e as envia para um banco de dados (MySQL). Em sentido reverso, o PHP pode coletar as informações do banco de dados e enviar para a interface *swf* que é visualizada pelo usuário. No bloco à direita, o pesquisador pode solicitar uma informação ao banco de dados por intermédio de outro programa em PHP. Assim os dados são lidos e o PHP estrutura as informações no servidor que é então enviado como arquivo de HTML para ser lida pelo pesquisador.

⁴² O MySQL é um sistema de gerenciamento de banco de dados e é atualmente um dos mais populares em uso. Trata-se de um *software* livre com compatibilidade com o PHP. É um sistema de fácil utilização com suporte através de uma interface gráfica (*MySQL Toolkit*) (disponível em: <<http://pt.wikipedia.org>>. Acesso em: 20.jul.2012).

⁴³ Através do *MySQL Toolkit* são abertas as tabelas onde os dados serão inseridos. Essa tabela é constituída por colunas, em que os nomes das variáveis do sistema serão definidos e pelas linhas, onde os dados serão inseridos (disponível em: <<http://pt.wikipedia.org>>. Acesso em: 20.jul.2012).

3.2 O *website* Laboratório Virtual

O Laboratório Virtual da Estação Ciência está hospedado em um servidor privado (Uol Host) no endereço virtual denominado Ideias na Caixa⁴⁴ (www.ideiasnacaixa.com). A hospedagem não pode ser feita na Estação Ciência porque o servidor da instituição não tem configuração para operar o programa PHP e o banco de dados MySQL, ferramentas necessárias para o funcionamento dos projetos apresentados.

Apesar disso, até agosto de 2012, o Laboratório Virtual podia ser acessado através do site da Estação Ciência, já que o *website* oficial www.ideiasnacaixa.com/laboratoriovirtual era redirecionado para o endereço www.eciencia.usp.br/laboratoriovirtual.

Figura 9 – *Print screen* da página do Laboratório Virtual.

O *layout* da página do Laboratório Virtual é mostrado na Figura 9⁴⁵. Como pode ser observado na Figura 10⁴⁶, o *website* é dividido em quatro *frames* (quadros). O primeiro, que

⁴⁴ O *website* ideiasnacaixa.com é um domínio registrado em nome de Marcos Matsukuma.

⁴⁵ Fonte: Laboratório Virtual. Disponível em: <www.ideiasnacaixa.com/laboratoriovirtual>. Acesso em: 19.jun.2012.

ocupa a faixa superior, apresenta, à esquerda, o logo da Estação Ciência / USP e, à direita, o título da página: Laboratório Virtual. Ao se clicar o logo da Estação Ciência o *website* da instituição é aberto em uma nova página *web*. A altura⁴⁷ desse *frame* é fixa, permitindo a visualização total do título, independentemente do tamanho da janela do *browser*.

O segundo quadro, localizado à esquerda da página, contém o menu do *website*. Ele foi construído para apresentar sempre a mesma largura e permitir a leitura de cada item em apenas uma linha. Para visualizar os itens abaixo da dobra da página⁴⁸, é necessário usar a barra de rolagem. A Figura 11⁴⁹ mostra o conteúdo completo deste menu.

Figura 10 – Quadros que compõem o *website* Laboratório Virtual.

⁴⁶ Fonte: Laboratório Virtual. Disponível em: <www.ideiasnacaixa.com/laboratoriovirtual>. Acesso em: 19.jun.2012.

⁴⁷ As dimensões utilizadas no *design* de páginas da *web* são dadas em *pixels*. O pixel é o menor elemento de imagem digital.

⁴⁸ Dobra da página é um termo técnico utilizado em impressos como o jornal em que o caderno é dobrado ao meio, apresentando o conteúdo acima e abaixo dessa dobra (NIELSEN, 2002, p. 23).

⁴⁹ Fonte: Laboratório Virtual. Disponível em: <www.ideiasnacaixa.com/laboratoriovirtual>. Acesso em: 19.jun.2012.

Figura 11 – Menu do *website*.

O terceiro *frame*, à direita da página, exibe as interfaces do Laboratório Virtual. Este quadro possui um *layout líquido*, ou seja, a interface se ajusta ao tamanho do quadro e sempre é visualizada integralmente, independentemente do ajuste de dimensão do *browser*. A Figura 12⁵⁰ reproduz o efeito da página quando a largura da janela de exibição do *website* é alterada. Observe que apenas a interface de animação reduz de tamanho, enquanto os demais quadros apresentam seus conteúdos nas dimensões originais. Essa configuração permite a apresentação da interface sempre acima da dobra da página.

A seleção de um dos itens do menu faz abrir a interface escolhida nesse terceiro *frame*. Uma exceção de Usabilidade aplicada ao padrão de escolha foi a de não marcar os *links* visitados. A diretriz de Usabilidade aplicada a esse item (Nielsen, 2007) sugere a marcação de *links* visitados como forma de auxiliar a memória do usuário, quando este precisa consultar vários textos disponíveis em um site. Isso evita que o usuário visite novamente um texto já lido. Todavia, a exceção se aplica a um conteúdo que pode ser visto diversas vezes por um mesmo usuário, o que é o caso da lista de *links* do Laboratório Virtual.

O quarto *frame* é um quadro oculto onde ocorrem as operações de envio e recepção de informações entre a interface de animação e o banco de dados configurado, executados por um programa PHP. Este quadro apenas é utilizado quando ocorre a coleta por *Log* de dados em algumas das interfaces selecionadas.

⁵⁰ Fonte: Laboratório Virtual. Disponível em: <www.ideiasnacaixa.com/laboratoriovirtual>. Acesso em: 19.jun.2012.

Figura 12 – *Layout* líquido: o quadro da interface ajusta a dimensão da animação, mostrando-a sempre inteira, independente da largura do *browser*.

3.3 As interfaces do Laboratório Virtual

O conteúdo das interfaces produzidas para esse projeto teve como base a Óptica. Esse e outros ramos da Física se beneficiam do uso da programação em *action script* do *Flash*, pois as expressões matemáticas podem ter seus resultados visualizados simbolicamente segundo os valores calculados pela manipulação de suas variáveis. Dessa forma, em vez de mostrar apenas grandezas numéricas (como faria uma conta ou uma calculadora), a animação reproduziria visualmente os eventos associados às alterações das propriedades físicas dos objetos. Os detalhes dessas programações podem ser vista no Apêndice B.

Como referência para a produção das interfaces, foi utilizada a apostila “Óptica”, do curso FGE-160 do Instituto de Física da USP (MURAMATSU, 2008). Esse material oferece

um cenário amplo do tema, explorando situações do cotidiano como tema gerador para o aprofundamento das questões teóricas.

A partir dessa apostila foram elaboradas animações contemplando aspectos da Óptica geométrica, princípios da teoria das cores e uma breve introdução à ondulatória. No total foram produzidas doze interfaces, cujos títulos são apresentados a seguir⁵¹:

1. Reflexão
2. Refração
3. Lentes (como se formam as imagens com as lentes)
4. Espelhos esféricos (como se formam as imagens com espelhos)
5. O olho (formação da imagem no olho e defeitos da visão)
6. Olho normal (formação da imagem no olho)
7. Miopia
8. Hipermetropia
9. Correção (correção de defeitos da visão)
10. Cores
11. Construindo ondas (onda transversal)
12. Interferência de ondas

Essas animações interativas foram construídas seguindo as características dos OA, podendo ser consideradas como tal. Elas foram elaboradas para conter a informação mínima necessária para a apresentação de um determinado tópico, constituindo-se em um objeto de granularidade adequada para possibilitar sua utilização em diferentes contextos de aprendizagem (reusabilidade, flexibilidade e modularidade). Essas animações também podem ser pesquisadas na Internet com auxílio de mecanismos de busca (como o Google), pois são incorporados metadados em seu corpo de programação.

Pela perspectiva do modelo de Ciclo de Vida de *Software*, pode-se relatar que a maior parte das animações produzidas seguiu o modelo Cascata. Assim, o processo se iniciava com a definição de requisitos que consistia em traduzir as expressões matemáticas dos temas abordados em ações realizadas por elementos gráficos dentro das interfaces; fazer com que determinados elementos pudessem ser manipulados pelo usuário; converter essas

⁵¹ Os títulos se referem aos nomes apresentados no menu. Os que aparecem entre parênteses, ocorrem quando o título da interface difere daquele do menu.

manipulações em alterações dos parâmetros das variáveis do sistema e, consequentemente, agir sobre elementos específicos.

Durante o esboço dos projetos, o desenho dos elementos gráficos da interface deveria corresponder aos diagramas dos aspectos físicos explorados (por exemplo, eixo óptico, objeto, imagem, lente e as linhas geométricas presentes na representação da formação da imagem em lentes). Esse desenho ajudava na construção de um *storyboard* onde as ações de interação eram esquematizadas passo a passo e limitadas aos contornos do cenário imaginado (os elementos não podiam ser movidos ou ser arrastados além dos limites desse cenário, por exemplo).

A partir desses esquemas, iniciava-se a elaboração da programação em *action script*. Como no *Flash* a programação pode ser editada em diferentes locais, ela foi desenvolvida em várias etapas e testada para verificar sua funcionalidade em diferentes momentos do trabalho. Alguns testes implicavam na necessidade de modificação de uma determinada programação, bem como a revisão de algum passo do projeto, num processo de *feedback* à etapa anterior. Quando os testes indicavam o funcionamento de acordo com o projeto inicial, a animação era publicada no *website* do Laboratório Virtual.

Devido à estrutura em escala reduzida da produção das animações para esse projeto, essas interfaces não se enquadram completamente aos modelos de Ciclo de Vida centrados em avaliação e estão muito distantes daqueles mais complexos constituídos por diferentes etapas (modelo Espiral de Engenharia de *Software* e o modelo de Engenharia de Usabilidade com foco na IHC). Por esse motivo, a produção delas se assemelha mais ao modelo Cascata.

Na interface das Cores, no entanto, a introdução do método de coleta de *Log* de dados, promoveu um processo de avaliação que não constava nos requisitos iniciais. Realizada inicialmente para verificar o grau de acerto de uma tarefa solicitada ao usuário (perguntava-se a ele quais as três cores primárias que originavam a luz branca), a avaliação registrou uma resposta não esperada, reflexo do desenho da interface. A situação levou à realização de mais testes com diferentes desenhos de interface.

Observa-se assim, que essa animação, em particular, se assemelha bastante ao modelo simplificado de *design* de interação ilustrado na Figura 1, em que o processo de produção é realizado em ciclos e a avaliação é introduzida como importante mediador para a construção do produto final. Muitos dos detalhes dessa produção também se enquadram no modelo de Engenharia de Usabilidade descrito por Rocha e Baranauskas (2003, p.120).

Os detalhes da produção e as particularidades relacionadas a cada uma das animações são descritos nos capítulos seguintes.

3.3.1 Reflexão

Figura 13 – *Print screen* da interface de animação Reflexão.

A animação da Reflexão mostra como a imagem se forma a partir do uso de espelhos planos. Nos espelhos, quando um raio incide sobre sua superfície sofre uma reflexão para o mesmo meio de onde veio. A lei da reflexão afirma que o ângulo de incidência (aquele formado pelo raio incidente e a normal do espelho) é igual ao ângulo de reflexão (aquele formado pelo raio refletido e a normal):

$$\theta_i = \theta_r \quad (1)$$

Além disso, a imagem se forma na direção do prolongamento do raio refletido e em uma linha que se liga ao objeto e que está perpendicular ao plano do espelho. Esses dados foram suficientes para transformar as expressões matemáticas, derivadas dessas regras, em linguagem de programação.

Dessa forma, o cálculo da programação agiu de forma a variar a posição da imagem e do ponto onde ocorria a reflexão. Esse cálculo não ofereceu grandes obstáculos, já que se utilizou o princípio de semelhanças de triângulos, situação comum nos problemas de geometria e com soluções baseadas em regras de proporcionalidade.

Nessa interface, o usuário podia alterar a posição do objeto e a do observador, necessitando para tanto clicar e arrastar esses elementos. Além disso, para se verificar a igualdade entre ângulo de incidência e de reflexão, foi colocado um transferidor, o que possibilitava a mensuração do ângulo em graus, e que acompanhava o ponto onde ocorria a reflexão.

3.3.2 Refração

A refração é o fenômeno óptico no qual um feixe de luz, ao passar de um meio para outro, sofre um desvio em sua trajetória. Isso ocorre porque os meios possuem índices de refração diferentes, o que afeta a velocidade de propagação da luz. Como a luz percorre a trajetória de menor intervalo de tempo (não de espaço!), o resultado é a sua refração. A expressão que fornece a relação entre o ângulo de incidência (formado pelo raio incidente e a normal da superfície que separa os dois meios) e o ângulo de refração (formado pelo raio refratado e a normal) é dada pela lei de Snell-Descartes:

$$n_i \cdot \text{sen } \theta_i = n_r \cdot \text{sen } \theta_r \quad (2)$$

Figura 14 – Print screen da interface de animação Refração.

Um dos requisitos para o projeto dessa animação, do mesmo modo que na interface da Reflexão, é a possibilidade da alteração das posições do objeto e do observador. Partindo desse critério, o problema central do *script* foi o cálculo da posição onde o feixe de luz sofre a refração. Nessa situação, em que os ângulos de incidência e de refração não são nem congruentes e nem complementares, o princípio de semelhanças de triângulos não se aplicava ao cálculo do projeto.

A solução foi variar valores de θ_i para encontrar valores de θ_r , a partir da lei de Snell-Descartes, desde que resultado se ajustasse à distância entre o objeto e o observador. Quando as condições eram respeitadas, o ponto de refração era determinado.

A posição da imagem era localizada pelo prolongamento do raio refratado com a intersecção da linha perpendicular originada do ponto do objeto à linha de separação dos dois meios. Da mesma forma que na Reflexão, um transferidor graduado em graus acompanhava o ponto de refração, permitindo a leitura dos ângulos de incidência e refração.

Além da possibilidade de mover o objeto e o observador (clitando e arrastando os elementos), um dos requisitos do projeto previa a variação dos valores dos índices de refração dos meios, entre o mínimo de um e o máximo de dois. Com isso era possível fazer com que o meio superior tivesse índice maior que o do meio inferior, e vice e versa, e verificar como a refração do feixe se alterava com essas mudanças.

3.3.3 Lentes (Como se formam as imagens com as lentes)

Figura 15 – Print screen da interface de animação Lentes.

Uma lente convergente é aquela que faz com que raios paralelos que incidem sobre ela sofram refração, causando uma convergência para um mesmo ponto (denominado foco). Por outro lado, uma lente divergente faz com que raios paralelos sofram refração, causando uma divergência, parecendo que os raios provêm de um mesmo ponto (foco virtual). A imagem

que se forma tem sua posição, tamanho e sentido determinados pelo tipo de lente, pela sua distância focal e pela posição do objeto.

É possível determinar geometricamente a posição d da imagem (distância em relação à lente) a partir da posição D do objeto (em relação à lente) e da distância f em que se encontra o foco dessa lente. Para tanto é necessário seguir as seguintes regras:

1. Um feixe de luz paralelo ao eixo principal da lente sofre refração e passa pelo foco;
2. Um feixe de luz que passa pelo centro da lente não sofre desvio.

A expressão, que pode ser determinada a partir dos parâmetros geométricos, é denominada equação da lente e pode ser escrita da seguinte forma:

$$\frac{1}{d} + \frac{1}{D} = \frac{1}{f} \quad (3)$$

Essa expressão foi convertida em *script* de programação sendo que a posição do objeto e a da lente podia ser alterada pelo usuário (clicando e arrastando-as), fazendo com que o cálculo indicasse a posição final da imagem. Outra variável disponível para que o usuário pudesse alterar foi a distância focal.

As duas animações foram colocadas juntas para que o usuário pudesse comparar a formação de imagem por uma lente convergente e divergente ao mesmo tempo.

3.3.4 Espelhos esféricos (Como se formam as imagens com espelhos esféricos)

Um espelho convexo é um espelho esférico em que a reflexão ocorre na superfície externa. Por outro lado, um espelho côncavo tem a sua reflexão na parte interna da calota esférica. A imagem que se forma a partir de espelhos esféricos tem sua posição, tamanho e sentido determinados pelo tipo do espelho, pela sua distância focal e pela posição do objeto.

Figura 16 – *Print screen* da interface de animação Espelhos esféricos.

É possível determinar geometricamente a posição d da imagem (distância em relação ao espelho) a partir da posição D do objeto (em relação ao espelho) e da distância f em que se encontra o foco desse espelho. Para tanto é necessário seguir as seguintes regras:

1. Um feixe de luz paralelo ao eixo principal do espelho sofre reflexão e passa pelo foco;
2. Um feixe de luz que passa pelo centro de curvatura do espelho não sofre desvio.

A expressão, que pode ser determinada a partir dos parâmetros geométricos, pode ser escrita da seguinte forma:

$$\frac{1}{d} + \frac{1}{D} = \frac{1}{f} \quad (4)$$

Do mesmo modo como as lentes, as informações do diagrama geométrico foram convertidas em *script* de programação sendo que a posição do objeto e da lente podiam ser alteradas para que cálculo indicasse a posição final da imagem. Outra variável que podia ser alterada pelo usuário era o centro de curvatura. A essa variação foi incorporada, a efeito de animação, a alteração da curvatura do espelho.

As duas animações foram colocadas juntas para que o usuário pudesse comparar a formação de imagem por um espelho convexo e um côncavo ao mesmo tempo.

3.3.5 O olho (formação da imagem no olho e defeitos da visão)

Figura 17 – *Print screen* da interface de animação sobre a formação da imagem no olho.

Essa interface reúne quatro animações sobre a visão: a formação da imagem no olho, a hipermetropia, a miopia e a correção desses dois defeitos. Elas foram criadas separadamente,

mas devido à afinidade de tema foi possível criar uma apresentação única. Trata-se de um exemplo da característica de modularidade e de granularidade de um OA, em que um objeto pode conter outros menores para formar uma unidade de trabalho.

A seguir são apresentados detalhes de cada uma dessas interfaces.

3.3.6 O olho (Formação de imagem pelo olho – olho normal)

O olho é o órgão responsável por focalizar a luz proveniente de um objeto e convertê-la em impulsos elétricos para enviá-los ao cérebro, onde é decodificada.

Figura 18 – Print screen da interface de animação *Formação da imagem no olho*.

A córnea e o cristalino são os elementos do olho responsáveis pela focalização, funcionando como uma lente convergente; a retina é uma camada formada por células responsáveis pela transformação da luz em estímulo elétrico e se localiza no fundo do olho, onde a imagem se forma.

Em um olho normal (emétrope) a imagem sempre se forma na retina, independentemente da distância onde se localiza o objeto⁵². Isso é possível porque o cristalino pode se contrair, alterando a sua distância focal de tal modo a alterar a posição em que a imagem se forma sobre a retina.

Nessa interface os elementos de programação utilizados foram os mesmos da animação da lente convergente. Porém, alguns requisitos foram revisados para mostrar ao usuário como o cristalino se adapta para focalizar a imagem na retina.

Assim, para essa animação, ainda é possível alterar a posição do objeto clicando-o e arrastando-o. Porém, como a distância em que a imagem se forma deve se manter constante (na posição da retina), não há necessidade de mover o foco, já que ele se desloca automaticamente, segundo o cálculo da expressão de formação de imagem. Além disso, nessa animação, ao mesmo tempo em que se varia a posição do foco, varia também o formato do cristalino.

A efeito de curiosidade, foi acrescentado a esse programa um recurso para desabilitar a função do cálculo que dá a variação automática da distância focal do cristalino. Com isso a interface mostra onde se formaria a imagem caso a estrutura do cristalino não pudesse sofrer alterações.

⁵² Na verdade, existe uma distância mínima na qual o olho normal enxerga sem desconforto e essa distância varia com a idade. Os bebês enxergam a centímetros dos olhos. Os jovens, de 10 a 15 cm. O olho perfeito, a cerca de 25 cm. Após os 40 anos, com a perda da acomodação visual, há dificuldade de enxergar objetos próximos (MURAMATSU, 2008, p. 20).

3.3.7 Miopia

Figura 19 – Print screen da interface de animação *Miopia*.

No olho míope a imagem do objeto se forma em uma posição anterior a da retina. Isso ocorre porque o globo ocular tem um formato mais longo que o normal (oval) e, apesar do cristalino alterar seu formato para ajustar sua distância focal, a imagem não consegue ser formada na retina. Nessa situação, o indivíduo míope tem dificuldade de enxergar objetos distantes.

O programa utilizado para calcular a posição da imagem foi o mesmo empregado no do olho normal. No entanto, foi acrescentada uma função na equação da lente que alterava a posição final do foco. Essa função projetava a imagem para uma posição anterior à retina para objetos distantes do olho, mas mantinha a imagem sobre a retina para objetos mais próximos, simulando o funcionamento de um olho míope.

Nessa animação apenas o objeto podia ser movido pelo usuário.

3.3.8 Hipermetropia

Figura 20 – Print screen da interface de animação *Hipermetropia*.

No olho hipermétrepe a imagem do objeto se forma depois da retina. Isso ocorre porque o globo ocular tem um formato mais achatado que o normal e, apesar do cristalino alterar seu formato para ajustar a sua distância focal, a imagem não consegue se formar na retina. Nessa situação, o indivíduo hipermétrepe tem dificuldade de enxergar objetos próximos.

O programa utilizado para calcular a posição da imagem foi o mesmo do olho normal. No entanto, foi acrescentada uma função na equação da lente que alterava a posição final do foco. Essa função projetava a imagem para uma posição posterior à retina para objetos próximos do olho, mas mantinha a imagem sobre a retina para objetos mais distantes, simulando o funcionamento de um olho míope.

Nessa animação apenas o objeto podia ser movido pelo usuário.

3.3.9 Correção (Correção de defeitos da visão)

Figura 21 – Print screen da interface de animação *Correção de defeitos da visão*.

A correção da miopia e da hipermetropia pode ser realizada com o auxílio de lentes divergentes e convergentes, respectivamente. Nessa simulação o usuário pode clicar e arrastar qualquer uma dessas lentes para próximo de cada olho com defeito e verificar em que situação haverá a correção da visão.

A Figura 22 mostra o feixe de luz quando o mouse passa sobre a lente divergente. A Figura 23 mostra o feixe quando o mouse está sobre na lente convergente. Ao ser colocada na frente do olho, a interface mostra como seria o novo trajeto do feixe de luz, que passa inicialmente pela lente e depois pelo cristalino do olho míope ou hipermetrópe.

Ao colocar a lente divergente em frente do olho míope, o feixe se abre antes de chegar ao cristalino. Isto faz com que o foco se afaste, atingindo a região da retina do olho míope, proporcionando a sua correção.

Figura 22 – Feixes da lente divergente.

Figura 23 – Feixes da lente convergente.

Por outro lado, a lente convergente aproximaria os feixes antes de eles chegarem ao cristalino do olho hipermetrópe. O resultado seria a aproximação do foco, que atingiria a retina do olho hipermetrópe.

Essa interface também possibilita a troca de lentes. Ou seja, a convergente para a miopia e a divergente para a hipermetropia. Mas essas opções apenas acentuariam os defeitos.

Figura 24 – Correção da miopia e da hipermetropia.

Figura 25 – Lentes trocadas. Sem correção.

Nessa interface não foi necessário um *script* especial para o cálculo de posição. Os desenhos apresentados em cada situação já estavam pré-definidos e apenas eram exibidos quando o usuário realizava determinada ação (mouse sobre as lentes ou lentes na frente dos olhos).

Log de dados

Essa foi a primeira interface em que se realizou a coleta de informação por *Log* de dados. O que se procurou determinar foi quais eram as escolhas que o usuário fazia para correção da visão. Assim, para cada defeito de visão havia duas respostas possíveis para a correção: *sim* ou *não* (formato das respostas possíveis: *miopiaSim*, *miopiaNao*, *hiperSim* e *hiperNao*). Essas informações eram enviados para fora do sistema e um programa PHP as enviava para o banco de dados MySQL juntamente com o endereço IP⁵³ da máquina do usuário, além da data e horário da interação. O propósito da coleta do endereço IP foi a de fornecer uma identificação ao grupo de dados armazenados.

3.3.10 Cores

Essa interface apresenta uma sequência de quatro trabalhos relacionados ao estudo da formação das cores. O primeiro deles é sobre as cores aditivas, ou seja, aquelas originadas das emissões de luz. O segundo é uma animação que relaciona as cores aditivas e subtrativas. O terceiro explica o que são as cores subtrativas, ou seja, aquelas provenientes da interação da luz com pigmentos coloridos. O último é um *software* em que o usuário tem como tarefa responder quais as cores que formam a luz branca.

Na tarefa desse último bloco foi introduzido o método de *Log* de dados para observar as cores escolhidas pelos usuários. A análise das respostas, que foram diferentes da esperada, levou a avaliar a interface e influenciou reformulações do desenho do projeto.

Uma das alterações realizadas foi a reorganização da ordem dos blocos de trabalho, sendo que a sequência dos quadros apresentada corresponde à última modificação realizada. Nos tópicos a seguir são apresentados os detalhes da construção e funcionamento desses blocos. No capítulo Resultados procurou-se dar continuidade à ênfase nas alterações da interface Cores, segundo a análise das informações coletadas, em um processo cuja descrição se assemelha ao modelo de Engenharia de Usabilidade, como descrito por Rocha e Baranauskas (2003, p. 120).

⁵³ Endereço IP ou *Internet Protocol address* é um rótulo numérico que identifica cada equipamento que participa de uma rede de computadores e que usa o protocolo de internet para se comunicar (disponível em: <<http://en.m.wikipedia.org>>. Acesso em: 20.jul.2012).

3.3.10.1 Cores primárias de luz

Figura 26 – Print screen da interface de animação *Cores – cores primárias de luz*.

Esse bloco mostra o vermelho, o verde e o azul como as cores primárias que combinadas produzem a cor branca. A simulação permite misturar essas três cores em qualquer proporção e visualizar o resultado. Quando as três cores são combinadas em suas intensidades máximas obtemos a cor branca.

Em computação, o vermelho (*red*), o verde (*green*) e o azul (*blue*) também são cores utilizadas para formar imagens na tela do computador. Em programação, a representação da composição das cores, a partir das componentes primárias, pode ser dada por:

$$\text{cor} = (\text{R}, \text{G}, \text{B}) \quad (5)$$

Esse sistema é denominado RGB, sendo R a intensidade da cor vermelha, o G, a da cor verde e o B, a da cor azul.

Figura 27 – Misturando cores primárias de luz: ao deslocar um dos círculos de cor para o centro, o círculo negro vai recebendo a luz da cor correspondente.

Figura 28 – Misturando cores primárias de luz: ao deslocar todas as cores para o centro, o círculo, antes negro, fica branco.

Na interface, o *script* de programação, faz com que o movimento dos círculos coloridos, da extremidade onde se encontram para o centro, altere a intensidade da cor correspondente para valores entre 0 e 255 (ou entre 00 e FF, em código hexadecimal⁵⁴). A linguagem *action script* já possui uma ferramenta própria para soma dessas cores. Orientando o resultado para um objeto no palco, possibilita a sua mudança de cor (efeito que ocorre no círculo inicialmente negro no centro da interface).

Podemos utilizar a Expressão 5 para representar as cores primárias aditivas:

$$\text{Vermelho} = (\text{FF}, 0, 0) \quad (6)$$

$$\text{Verde} = (0, \text{FF}, 0) \quad (7)$$

$$\text{Azul} = (0, 0, \text{FF}) \quad (8)$$

⁵⁴ No código hexadecimal um número qualquer é representado por 16 algarismos. Nos sistemas computacionais esses algarismos são os seguintes: 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F. Dessa forma, em uma representação numérica de base hexadecimal, cada ordem corresponde a 16 valores possíveis de algarismos (de 0 a F). Para um número de duas ordens, poderemos ter até 16 x 16 valores, ou seja, 256 valores possíveis: começando pelo número 00 e indo até o FF, que em base decimal correspondendo ao número 255.

Combinando aos pares, obtemos as cores denominadas secundárias:

Amarelo = (FF, FF, 0) (9)

Magenta = (FF, 0, FF) (10)

Ciano = (0, FF, FF) (11)

O branco é a combinação de todas as cores:

Branco = (FF, FF, FF) (12)

Figura 29 – A combinação das cores aditivas pode ser visualizada pela codificação hexadecimal, o sistema numérico dos computadores.

As explicações sobre a combinação das cores primárias de luz com variações de intensidade dada por valores hexadecimais foram adicionadas à interface em caráter de curiosidade. Para acessar essa informação o usuário deve clicar sobre o símbolo (+) que aparece abaixo do círculo de soma de cores (Figura 29).

3.3.10.2 Mistura das cores aditivas

Figura 30 – Print screen da interface de animação *Cores – mistura das cores aditivas*.

O segundo trabalho da interface das Cores é uma animação que mostra as cores primárias de luz se fundindo, duas a duas, pra dar origem a um outro grupo denominado cores secundárias (como as indicadas pelas somas (9), (10) e (11)). Essa animação começa com apenas as três cores primárias de luz e em seguida círculos de mesma cor se afastam das originais percorrendo uma trajetória circular até encontrar outra, vindo de direção oposta, e se fundindo a ela. Do vermelho com o verde surge o amarelo, do verde com o azul, o ciano e do vermelho com o azul, o magenta.

Apesar dessa combinação poder ser obtida pela manipulação da primeira interface de mistura de cores de luz, a animação introduz o círculo cromático para mostrar a relação entre as cores primárias de luz e as de pigmento, detalhadas no terceiro trabalho dessa série.

Figura 31 – A animação do círculo cromático mostra a relação entre as cores primárias aditivas e subtrativas.

3.3.10.3 Cores primárias de pigmento

As cores de pigmento são aquelas usadas em pinturas e impressões. Como explicado anteriormente, as cores básicas de pigmento correspondem às cores secundárias de luz: o ciano, o magenta e o amarelo.

Essa interface permite a mistura dessas três cores em proporções diferentes e verificar o seu resultado. Quando as três cores são combinadas em suas intensidades máximas obtemos o preto. Isso ocorre, pois a mistura de pigmentos vai absorvendo cada vez mais luz⁵⁵.

O *action script* não possui um código específico que manipula diretamente as cores de pigmento. Para obtê-las a programação utilizava o código RGB, conforme as representações já mostradas em (9), (10) e (11).

$$\text{Amarelo} = (\text{FF}, \text{FF}, 0) \quad (9)$$

$$\text{Magenta} = (\text{FF}, 0, \text{FF}) \quad (10)$$

$$\text{Ciano} = (0, \text{FF}, \text{FF}) \quad (11)$$

⁵⁵ Na verdade isso apenas ocorre em teoria. Nos processos gráficos o preto é utilizado como uma quarta cor básica independente, sendo que o sistema cromático de impressão é denominado CMYK (Cyan, Magenta, Yellow e Key- Black).

Figura 32 – Print screen da interface de animação *Cores – cores primárias de luz*.

Assim, para obter uma dessas cores, o *script* subtraia do branco (FF, FF, FF) o valor complementar da cor desejada. Por exemplo, para a cor amarela, representada por (FF, FF, 00), a sua cor complementar, é o azul, representado por (00, 00, FF). Assim, ao subtrair a cor azul do branco obtém-se o amarelo:

$$\text{Branco} - \text{azul} = (\text{FF}, \text{FF}, \text{FF}) - (00, 00, \text{FF}) = (\text{FF}, \text{FF}, 00) = \text{Amarelo}. \quad (12)$$

O mesmo ocorre com as demais cores:

$$\text{Branco} - \text{verde} = (\text{FF}, \text{FF}, \text{FF}) - (00, \text{FF}, 00) = (\text{FF}, 00, \text{FF}) = \text{Magenta}. \quad (13)$$

$$\text{Branco} - \text{vermelho} = (\text{FF}, \text{FF}, \text{FF}) - (\text{FF}, 00, 00) = (00, \text{FF}, \text{FF}) = \text{Ciano}. \quad (14)$$

No programa, o deslocamento dos círculos da extremidade para o centro, subtraia do branco as intensidades seleccionadas para simular a mistura de cores de pigmento. Assim, misturar amarelo e magenta, significava subtrair do branco as cores azul e verde:

Amarelo + Magenta = Branco – azul – verde →

→ (FF, FF, FF) – (00, 00, FF) – (00, FF, 00) = (FF, 00, 00) →

→ Amarelo + Magenta = Vermelho. (15)

O mesmo ocorria para os outros pares de cores:

Amarelo + Ciano = Verde (16)

Magenta + Ciano = Azul. (17)

Finalmente, a combinação das três cores fornecia o preto:

Amarelo + Magenta + Ciano = Branco – Azul – Verde – Vermelho →

→ (FF, FF, FF) – (00, 00, FF) – (00, FF, 00) – (FF, 00, 00) = (00, 00, 00) →

→ Amarelo + Magenta + Ciano = Preto. (18)

Figura 33 – A combinação de pigmentos de cor: ao mover as cores para o centro, a interface apresenta o resultado da mistura.

Figura 34 – A combinação de pigmentos de cor: ao juntar todas as cores de pigmento, o resultado é o preto.

Em caráter de curiosidade, foi adicionada à interface uma explicação sobre a combinação de cores pelo método de impressão no qual suas intensidades são expressas em porcentagem. Para tanto, foram convertidos os valores hexadecimais para decimais. Para acessar essa informação o usuário deve clicar sobre o símbolo (+) que aparece abaixo do círculo de soma de cores (Figura 35).

Figura 35 – Mistura de cores de pigmento mostrando variações em porcentagem como em processos gráficos.

3.3.10.4 Tarefa para o usuário: que cores combinadas formam a luz branca?

Nessa interface pergunta-se ao usuário quais cores primárias que combinadas formam a luz branca. O desenho desse trabalho seguiu o esquema das relações cromáticas.

Para escolher a cor o usuário deve clicar sobre uma delas e levá-la até um dos círculos vazios, à direita da interface. Caso queira, pode trocar as cores. Porém, quando os três círculos vazios são completados, o programa para e é verificado se a escolha está certa ou não. A verificação ignora a ordem de escolha das cores.

A Figura 37 mostra a mensagem para a escolha certa enquanto a Figura 38 mostra a mensagem para a errada.

Figura 36 – *Print screen* da interface de animação da tarefa para o usuário: que cores combinadas dão o branco?

Um programa foi acrescentado a essa interface para que um *Log* de dados fosse acionado caso fossem preenchidos completamente os círculos vazios. O *Log* enviava três informações, denominadas *cor1*, *cor2* e *cor3*. Cada uma delas poderia receber o código associado a cada uma das cores escolhidas:

Vermelho: *corRR*,

Verde: *corGG*,

Azul: *corBB*,

Amarelo: *corRG*,

Ciano: *corGB*,

Magenta: *corBR*.

Figura 37 – A mensagem para a resposta correta.

Figura 38 – A mensagem para a resposta não esperada.

Um programa PHP, desenvolvido especialmente para essa interface, recebia essas três informações. Em seguida, ele abria o banco de dados MySQL e enviava esses dados para uma tabela, além do endereço IP da máquina utilizada, da data e da hora em que houve a interação do usuário. O propósito da coleta do endereço IP foi a de fornecer uma identificação ao grupo de dados armazenados.

Os dados coletados a partir desse processo foram de grande importância para avaliar e estudar o processo de interação da interface. Modificações foram feitas no protótipo para compreender como elas influenciavam os usuários. Esse processo é descrito em detalhes no capítulo Resultados.

3.3.11 Construindo ondas (onda transversal)

Além do campo da Óptica geométrica, a luz pode ser estudada pela sua natureza eletromagnética. Assim, é importante compreender alguns aspectos associados à ondulatória.

A onda eletromagnética é formada por uma componente formada por um campo elétrico e outra por um campo magnético que oscilam transversalmente em relação à direção de propagação e ortogonalmente entre si.

Figura 39 – Print screen da interface de animação Construindo onda.

A interface produzida apresenta a onda do campo elétrico, que é dada pela expressão abaixo:

$$E(x,t) = E_0 \cdot \cos (k \cdot x - w \cdot t) \quad (19)$$

Figura 40 – Aumentando o valor de k , diminui-se o comprimento de onda.

Figura 41 – Aumentando o valor da amplitude, a intensidade da onda aumenta.

Sendo E_0 a amplitude dessa onda, k , o parâmetro definido como número de onda (relacionado à propagação no espaço) e w , o parâmetro definido como velocidade angular (relacionada à propagação no tempo). As variáveis x e t são, respectivamente, as coordenadas de espaço e de tempo da função de onda.

O *action script* possui uma biblioteca de funções trigonométricas e os seus argumentos podem ser programados de modo idêntico à função de onda. Todos os parâmetros podem ser alterados facilmente e sobre a variável tempo pode ser aplicada o recurso de *looping* de programação, sendo possível simular a passagem do tempo e, conseqüentemente, o movimento da onda.

Nessa interface o usuário pode alterar a amplitude, o número de onda k e a velocidade w . Inicialmente todos esses três parâmetros encontram-se em seus valores mínimos. Ao aumentar a amplitude, o gráfico mostra uma linha se elevando acima do nível 0 (comprimento de onda longo). Aumentando o valor de k o comprimento de onda vai ficando cada vez menor⁵⁶. Finalmente, ao variar o valor de w a onda começa a se movimentar.

3.3.12 Interferência de ondas

Um princípio importante na Óptica ondulatória é o conceito de interferência de onda. Esse fenômeno ocorre quando mais de uma onda se encontram e, como consequência, temos uma onda resultante diferente daquelas que deram origem.

Nessa interface temos duas ondas em que é possível variar, em cada uma delas, seus parâmetros de amplitude, número de onda k e fase de onda. O resultado da soma das duas ondas pode ser visto na parte inferior desse trabalho.

A alteração dos valores desses parâmetros permite observar dois fenômenos importantes da interferência de onda: a interferência construtiva e a destrutiva. Para verificá-los, o usuário deve manter os valores dos parâmetros iguais para as duas ondas. Caso a fase seja igual, as ondas se somam resultando em uma com o dobro da amplitude original; temos assim uma interferência construtiva. Por outro lado, se for feito uma modificação para que as

⁵⁶ Comprimento de onda é a distância entre dois picos ou dois vales de onda.

ondas estejam defasadas em meia onda, elas se anularão, dando origem a uma interferência destrutiva.

Figura 42 – Print screen da interface de animação Interferência de onda.

Figura 43 – Interferência construtiva: ondas idênticas com a mesma fase.

Figura 44 – Interferência destrutiva: ondas idênticas com fases opostas.

3.4 Obtendo informações do usuário da interface

A parte preparatória para o desenvolvimento do *software* ocorre na fase de *pré-design* (ROCHA; BARANAUSKAS, 2003, p. 120) e nessa etapa procura-se conhecer o indivíduo, obter suas características particulares e compreender qual será a sua relação com a interface.

Já se conhecia o perfil do público do Laboratório Virtual desde 2000, pois entre março de 1999 e março de 2000 foi realizada uma pesquisa online em que se obteve registro de 409 usuários, apontando que esse público era formado por professores (32%) e estudantes (25%) (MATSUKUMA, 2000, p. 477).

Assim, já se sabia que as animações produzidas seriam dirigidas a esses usuários. No entanto, aproveitou-se esse trabalho para confirmar os dados anteriores, o que foi feito mediante um questionário hospedado no *website* do Laboratório Virtual.

Esse questionário foi o primeiro método para obtenção de informações do usuário. Outro foi o de *Log* de dados que foi usado na animação sobre Correção de Defeitos da Visão e sobre Cores. Nessas animações foi possível coletar informações sobre tarefas que deveriam ser realizadas e identificar problemas específicos com o *design*, a partir de resultados inesperados dados pelos usuários.

3.4.1. Questionário de perfil dos usuários

O questionário de perfil do usuário pode ser acessado a partir do menu do Laboratório Virtual. Nele se pergunta se o indivíduo é um *aluno*, um *professor* ou um *usuário* comum. O questionário possui campos para seleção de opções e um campo de texto para que se escrevam observações. Caso seja selecionada uma das opções, a interface abre novas opções para que o indivíduo refine seu perfil.

A página de abertura do questionário é mostrada na Figura 45.

pesquisa sobre o usuário		comentário
	eu sou...		<input type="text"/>
	professor	área de atuação	
	aluno	() ensino fundamental	
	usuário	() ensino médio	
		() outra	
		() Língua Portuguesa	() Política
		() Matemática	() Língua Estrangeira
		() Ciências Naturais	() Arte
		() Biologia	() Informática
	() Física	() Educação Física	
	() Química	() Temas Transversais	
	() História	() Meio Ambiente	
	() Geografia	() Saúde	
	() Sociologia	() Pluralidade Cultural	
	() Antropologia	() Orientação Sexual	
	() Filosofia		

Figura 46 – Print screen do questionário para professor.

Quando a opção *ensino fundamental* ou *ensino médio* são escolhidas, abrem-se mais opções, como mostrado na Figura 48. Caso seja escolhido *ensino fundamental*, o aluno deve escolher as séries 1ª, 2ª, 3ª, 4ª, 5ª, 6ª, 7ª, 8ª ou 9ª. Se for escolhido *ensino médio*, deve ser escolhida as séries 1ª, 2ª ou 3ª. Apenas uma opção pode ser escolhida.

Se a for escolhida a opção *outro* se abre uma caixa de texto para livre preenchimento.

Quando a opção *usuário* é selecionada pergunta-se qual a sua escolaridade e qual sua profissão (caixa de texto para livre preenchimento).

A escolaridade pode ser selecionada entre *ensino fundamental*, *ensino médio*, *ensino superior* e *outro* (Figura 49). Apenas é possível selecionar uma dessas opções. Caso seja selecionada *outro*, uma caixa de texto se abre para livre preenchimento.

pesquisa sobre o usuário			comentário
eu sou...			
professor			
aluno	matriculado no	idade	
usuário	☐ ensino fundamental ☐ ensino médio ☐ outro		
matérias preferidas			

Figura 47 – Print screen do questionário para aluno.

pesquisa sobre o usuário			comentário	enviar pesquisa
eu sou...				
professor				
aluno	matriculado no	idade		
usuário	☒ ensino fundamental 1ª 2ª 3ª 4ª 5ª 6ª 7ª 8ª 9ª ☐ ensino médio ☐ outro			
matérias preferidas				

Figura 48 – Ao escolher a opção ensino fundamental, abre-se novas opções.

pesquisa sobre o usuário		comentário
	eu sou...		<input type="text"/>
	professor		
	aluno		
	usuário	escolaridade	
		() ensino fundamental	
		() ensino médio	
		() ensino superior	
		() outro	
		profissão	
	<input type="text"/>		

Figura 49 – Print screen do questionário para usuário comum.

Em todas as situações escolhidas é possível escrever algo na caixa de texto *comentário*. Após a identificação do perfil do usuário e após ser selecionada alguma opção, um botão “enviar pesquisa” fica disponível (Figura 48).

Ao clicar o botão de envio, os dados selecionados e preenchidos são enviados a uma página PHP. Essa página abre um banco de dados MySQL, armazenando essas informações, juntamente com o endereço IP do computador do usuário, a data e hora do registro. Os dados armazenados foram analisados posteriormente por uma página PHP devidamente programada para resgatar as informações necessárias.

3.5. Google Analytics

Um recurso acessório para acompanhar remotamente o acesso das interfaces de animação foi o uso dos serviços do *Google Analytics*⁵⁷. Essa ferramenta, da empresa de busca *Google*, oferece a possibilidade de consultar informações sobre a audiência de páginas na *web*.

O uso do *Google Analytics* é gratuito e para utilizá-lo é necessário fazer um cadastro e informar o nome do *website* que terão suas páginas monitoradas. Em seguida o *Google* fornece um código fonte que deve ser inserido em cada página HTML que deve ser rastreada.

Assim, cada vez que um usuário abrir uma página em um *browser*, informações sobre o acesso do usuário alimentarão o banco de dados do sistema, que poderão ser consultados através de relatórios gráficos, fornecendo quantidade de visitas diárias ao *website*, quantidade de visitantes novos, quantidade de retorno de visitantes, país de acesso, sistema operacional utilizado, entre outros dados.

Figura 50 – Aspecto da tela do *Google Analytics* para o Laboratório Virtual.

⁵⁷ <http://www.google.com/analytics>

3.6 Resumo

A produção das interfaces de animação do Laboratório Virtual pode ter seu processo explicado pelo modelo de Ciclo de Vida de *Software*. Praticamente todos se enquadraram no modelo Cascata: se iniciaram a partir de alguns requisitos iniciais e o trabalho foi sendo desenvolvido praticamente de modo linear até chegar ao protótipo final.

No entanto, isso não significou que o projeto foi realizado de modo simples. No caso da refração, por exemplo, a lei de Snell Descartes, que define como uma trajetória de um feixe de luz se desvia ao atravessar um meio, tem solução simples apenas quando se conhece o ponto em que ocorre a refração. No entanto, como objeto e observador se movem, esse ponto não se encontra fixo, tornando a solução mais complexa. Esse fato não era evidente no início e só foi verificado quando a lei de Snell Descartes teve de ser traduzido para a linguagem de programação.

Em alguns casos, também foram feitas modificações no desenho da interface mesmo após a sua conclusão e publicação no *website*. As animações do espelho e do olho, por exemplo, apresentavam elementos ópticos com formatos fixos, mesmo com a variação da posição dos focos. Apesar de não interferir no resultado da formação da imagem, essa condição foi alterada para dar mais veracidade à interatividade e tanto os espelhos esféricos como o cristalino dos olhos passaram a ter seu desenho modificado quando a posição do foco era alterada.

Esses foram problemas específicos identificados mediante inspeção de Usabilidade (ROCHA; BARANAUSKAS, 2003, p. 161) e que procuraram melhorar as características finais da interface.

Mas a interface que melhor desenvolveu as características do modelo de Ciclo de Vida de *Software* centrado no usuário foi o das Cores. Durante o desenvolvimento iterativo (fase do modelo de Ciclo de Vida de Engenharia de Usabilidade descrito por Rocha e Baranauskas, 2003, p. 120) foi feita a avaliação de Usabilidade do projeto a partir do método de coleta de *Log* de dados, cuja análise de informações verificou respostas inesperadas e provocou a necessidade de realizar alterações na interface para analisar como esse *design* influenciava nas escolhas dos usuários.

O processo de resgate dos *Logs* de dados da interface de animação Cores e as modificações sofridas em seu desenho, assim como os resultados do questionário sobre o perfil dos usuários, são temas detalhados no capítulo seguinte.

4. RESULTADOS

A maior parte da produção das interfaces do Laboratório Virtual seguiu o modelo de Ciclo de Vida Cascata, como relatado no capítulo anterior. A interface de animação Cores, no entanto, ao introduzir o método de coleta de *Log* de dados, desencadeou um processo de reavaliação do seu *design*, gerando uma fase de *desenvolvimento iterativo*, característico do modelo de *Engenharia de Usabilidade*, descrito por Rocha e Baranauskas (2003, p. 120).

Apesar do projeto inicial da interface Cores ter previsto a utilização da coleta de *Log* de dados, seu intuito original era apenas registrar a estatística do seu uso. Não se imaginava que haveria respostas inesperadas na sua interação com o usuário. Essa ocorrência motivou o desenvolvimento de interfaces para resgatar os dados registrados no banco de dados e apresentá-los de modo a auxiliar na percepção de como o usuário interagia com a animação Cores. Com essas ferramentas em mãos, realizou-se uma série de testes para verificar se alterações no *layout* da interface mudariam o modo como os usuários utilizavam o *software*.

Dessa forma, esse projeto incorpora a coleta e a análise de dados como parte de um processo de *feedback* para avaliação e *re-design* da interface, como explicitado no funcionamento dos modelos de Ciclo de Vida de *Softwares*. Assim, nesse trabalho, essa etapa não determina um fim, mas ao contrário, o início de uma nova fase dentro do projeto.

4.1 Objetivos dos testes realizados na interface de animação Cores

Em um dos blocos da interface Cores o usuário é convidado a responder quais são as três cores primárias que formam a luz branca. É possível fazer qualquer combinação entre as seis cores disponíveis no círculo cromático dessa interface. As respostas coletadas pelo método de *Log* de dados revelaram que os usuários⁵⁸ selecionavam algumas combinações de cores esperadas (o que incluía a resposta correta: as cores vermelho, verde e azul); porém

⁵⁸ O perfil dos usuários que acessavam a interface Cores poderia ser obtido ao se verificar o número IP registrado a partir do formulário preenchido por eles. No entanto, não foi possível produzir um programa para realizar esse procedimento no período de elaboração dessa pesquisa.

algumas delas, escolhidas de modo muito frequente, foram inesperadas, o que provocou o início de testes de Usabilidade nesse trabalho.

A Usabilidade avaliada nesse bloco de trabalho não faz parte das diretrizes de Usabilidade padrão descrita por Nielsen (2002, p.17); na verdade, trata-se de uma questão específica originada das respostas inesperadas efetuadas pelos usuários.

A estrutura de trabalho realizado nesse bloco, que faz parte do *desenvolvimento iterativo* do modelo de *Engenharia de Usabilidade*, pode ser resumida a partir das respostas às questões apresentadas por Rocha e Baranauskas (2003, p. 202), para delinear os objetivos dos testes de Usabilidade empregados nesse projeto.

- O objetivo do teste: o que se deseja obter?

Verificar como a disposição das cores da interface pode afetar a escolha dos usuários. O objetivo principal é a avaliação do *software*, não do usuário.

- Quando e onde o teste irá acontecer?

O teste é realizado pelo método de *Log* de dados. Portanto, ele ocorre no próprio local de trabalho do usuário, no período por ele determinado.

- Qual a duração prevista de cada sessão de teste?

A duração dos testes variou em cada sessão. A princípio, dependeu do tempo de confecção de *softwares* de análise e também de um período mínimo (em torno de dois meses) para verificar como cada alteração da interface afetava a escolha das cores feita pelos usuários.

- Qual o suporte computacional necessário?

Os testes podiam ser realizados em qualquer computador com um *browser* (Internet Explorer, Chrome, Firefox, Safari, Opera, entre outros) que pudesse exibir as animações *shockwave* do Laboratório Virtual.

- Qual *software* precisa estar à disposição?

O *browser* precisa ter o *plugin shockwave flash* instalado.

- Quem serão os experimentadores?

Os testes foram supervisionados pelo próprio pesquisador desse projeto.

- Quem serão os usuários e como serão conseguidos?

Quaisquer usuários que visitassem o Laboratório Virtual puderam realizar a escolha de cores e, portanto participaram do teste. Seu perfil foi determinado a partir de um questionário disponível no *website* do Laboratório Virtual.

- Quantos usuários são necessários?

Não houve limite de usuários para os testes serem realizados.

- Quais as tarefas que serão solicitadas aos usuários?

Os usuários deveriam responder que cores primárias formavam a luz branca. Para isso, escolhiam três, entre seis cores disponíveis na interface, e as arrastava para três círculos vazios. Era possível realizar a tarefa quantas vezes fosse preciso.

- Qual critério será utilizado para definir que os usuários terminaram cada tarefa corretamente?

O *software* informa ao usuário se escolheu as cores corretas. Caso queira, é possível repetir a tarefa.

- Quanto o experimentador poderá ajudar o usuário durante o teste?

A interface dá explicações sobre as cores primárias de luz e sobre as cores primárias de pigmento.

- Quais dados serão coletados e como serão analisados uma vez que tenham sido coletados?

A cada tarefa realizada foram coletados, pelo método de *Log* de dados, três códigos correspondentes a cada uma das cores escolhidas pelo usuário. Estas informações eram armazenadas em um banco de dados, juntamente com o endereço IP, a data e a hora da interação. Os dados foram analisados a partir tabelas e gráficos que apresentavam a quantidade de vezes que cada conjunto de cores era escolhida.

- Qual o critério para determinar que a interface é um sucesso?

O sucesso não foi a questão focada no trabalho. No entanto, o método de *Log* de dados e as ferramentas criadas para a análise de dados coletados, mostraram-se eficazes para compreender como a interface influenciava as escolhas do usuário.

4.2 Primeira avaliação da interface de animação Cores

A interface de animação Cores foi publicada no *website* do Laboratório Virtual em julho de 2010. Ela era composta de quatro blocos e, em sua primeira versão, estava organizada como mostra a Figura 51.

Figura 51 – Ordem dos blocos da animação Cores em sua primeira versão (jul. 2010): a) tarefa das cores; b) mistura de cores primárias de luz; c) animação do círculo cromático e d) mistura de cores primárias de pigmento.

No primeiro bloco da interface, os usuários tinham como tarefa responder quais as três cores primárias formava a luz branca. As cores eram selecionadas e colocadas nos círculos vazios à direita da interface e quando elas estavam todas ocupadas seus códigos correspondentes (*corRR*, *corGG*, *corBB*, *corRG*, *corGB*, *corBR*) eram enviados a um banco de dados, por meio de um programa escrito em PHP, alimentando uma tabela previamente formatada. Além das três cores selecionadas também eram enviados o endereço IP da máquina utilizada pelo usuário, a data e a hora em que a ação era realizada.

Para consultar os registros gravados através do método de *Log* de dados, a primeira forma imaginada foi desenvolver um programa em PHP para calcular as quantidades de cada trio de cores e em seguida exibir uma tabela da amostragem em uma página HTML.

id	ip	cor 1	cor 2	cor 3	data	horário
1851	177.33.121.8	Amarelo	Vermelho	Azul	12-06-2012	09:25:34
1852	177.33.121.8	Amarelo	Magenta	Ciano	12-06-2012	09:25:46
1853	177.33.121.8	Amarelo	Magenta	Ciano	12-06-2012	09:25:54
1854	201.68.5.46	Verde	Vermelho	Azul	13-06-2012	04:02:05
1855	200.206.254.230	Amarelo	Verde	Vermelho	13-06-2012	11:23:32
1856	200.206.254.230	Verde	Verde	Vermelho	13-06-2012	11:23:57
1857	187.35.89.246	Verde	Verde	Vermelho	14-06-2012	03:11:13
1858	187.35.89.246	Ciano	Amarelo	Magenta	14-06-2012	03:11:28
1859	189.37.64.10	Magenta	Amarelo	Vermelho	14-06-2012	03:19:22
1860	177.42.43.180	Vermelho	Amarelo	Verde	14-06-2012	06:26:28
1861	177.42.43.180	Ciano	Amarelo	Verde	14-06-2012	06:27:53
1862	177.144.195.222	Vermelho	Verde	Verde	14-06-2012	09:03:35
1863	189.102.6.214	Amarelo	Vermelho	Magenta	16-06-2012	10:06:44
1864	189.102.6.214	Ciano	Magenta	Amarelo	16-06-2012	10:07:48
1865	189.107.254.114	Vermelho	Verde	Amarelo	19-06-2012	11:32:29
1866	200.129.186.253	Verde	Vermelho	Amarelo	20-06-2012	10:57:40
1867	186.212.22.185	Verde	Vermelho	Verde	25-06-2012	01:35:25
1868	186.212.22.185	Verde	Vermelho	Verde	25-06-2012	01:41:46
1869	189.79.32.23	Vermelho	Amarelo	Magenta	25-06-2012	03:45:22
1870	95.7.229.182	Amarelo	Vermelho	Magenta	26-06-2012	11:41:20

Tabela 1 – Parte das amostragens das escolhas de cores dos usuários.

Esse programa denominado “tabela de escolha dos grupos de cores” (<http://www.ideiasnacaixa.com/pesquisa/resultadoOpticao1Cor.php>) exibia todas os grupos de cores selecionados e organizados em ordem cronológica, como apresentada na Tabela 1⁵⁹. Além da escolha do trio de cor 1, cor 2 e cor 3 (correspondendo a cada um dos círculos vazios

⁵⁹ Fonte: Laboratório Virtual. Disponível em:
<<http://www.ideiasnacaixa.com/pesquisa/resultadoOpticao1Cor.php>>. Acesso em: 26.jun.2012.

da interface), a tabela exibida em outras colunas o identificador de ordem numérica id, o endereço IP do computador utilizado pelo usuário, a data e a hora da interação. Detalhes desse programa podem ser observados no Apêndice D.

No final dessa página era exibida uma tabela de cores mostrando a quantidade que cada uma das trincas havia sido escolhida. Esses grupos resultaram da combinação das seis cores do círculo cromático. Foram considerados eventos iguais as trincas que apresentavam as mesmas cores, independente de suas ordens. Disso resultaram 20 combinações possíveis, como mostrados na Tabela 2⁶⁰.

107	Red	Green	Blue	grupo 1
27	Yellow	Green	Blue	grupo 2
143	Red	Yellow	Blue	grupo 3
49	Red	Green	Yellow	grupo 4
38	Yellow	Magenta	Blue	grupo 5
5	Yellow	Green	Magenta	grupo 6
15	Red	Yellow	Magenta	grupo 7
28	Magenta	Green	Blue	grupo 8
124	Red	Yellow	Blue	grupo 9
11	Red	Green	Magenta	grupo 10
22	Yellow	Magenta	Cyan	grupo 11
63	Magenta	Green	Blue	grupo 12
8	Magenta	Green	Cyan	grupo 13
11	Red	Magenta	Cyan	grupo 14
11	Yellow	Cyan	Blue	grupo 15
13	Yellow	Green	Cyan	grupo 16
9	Red	Yellow	Cyan	grupo 17
9	Yellow	Green	Blue	grupo 18
8	Red	Cyan	Blue	grupo 19
5	Red	Green	Cyan	grupo 20

Tabela 2 – Exibição da combinação do trio de cores escolhida pelo usuário. Período selecionado: 22 de julho a 31 de dezembro de 2010.

Logo após os primeiros meses de coleta de dados já era possível verificar a relação das quantidades de respostas escolhidas. A Tabela 2 apresenta o resultado das coletas realizadas entre o dia 22 de julho a 31 de dezembro de 2010, com um total de 706 de respostas registradas (<http://www.ideiasnacaixa.com/pesquisa/resultadoOpticao1Cor.php?dataInicio=>

⁶⁰ Fonte: Laboratório Virtual. Disponível em:

<<http://www.ideiasnacaixa.com/pesquisa/resultadoOpticao1Cor.php?dataInicio=2010-07-22&dataFim=2010-12-31>>. Acesso em: 26.jun.2012.

2010-07-22&dataFim=2010-12-31). Nessa tabela verifica-se que a resposta correta (grupo 1: vermelho, verde e azul) não foi a mais selecionada (107 escolhas). Antes dela escolheu-se o grupo de cores vermelho, amarelo e azul (grupo 3: 143 escolhas) e o grupo vermelho, magenta e azul (grupo 9: 124 escolhas). A tabela é organizada em ordem dos grupos citados e não em ordem de quantidades.

O número de escolhas do grupo 3 (vermelho, amarelo e azul) não foi uma surpresa, já que estas são as utilizadas com frequência como cores primárias nas aulas de artes em escolas. No entanto, a quantidade de seleções feitas para o grupo 9 (vermelho, magenta e azul) foi inesperada. Esse fato poderia ser explicado pela disposição dos elementos dentro do *layout* da interface, já que essas cores correspondem àsquelas mais próximas aos círculos vazios de escolha (Figura 51a).

Foi para poder analisar melhor como as cores eram escolhidas e responder com mais segurança o que ocorria com o uso da interface Cores, que foi desenvolvida outra maneira de apresentação de dados, diferente daquela mostrada na Tabela 1.

Nesse novo gráfico o número de escolhas de cada grupo de cores é apresentado como uma evolução no tempo. Feito em *Shockwave*, essa interface acessa a base de dados via PHP, somando os eventos e distribuindo-os ao longo da linha de tempo diária. Detalhes desse programa podem ser observados no Apêndice C.

O Gráfico 4⁶¹ apresenta a aparência da distribuição dos dados (<http://www.ideiasnacaixa.com/pesquisa/leituraPesquisaOptica01.html>). Nele, cada grupo de cores é representado por um quadrado contendo as três cores do conjunto escolhido (Gráfico 5⁶²). O eixo vertical mostra a quantidade de escolhas feitas e o eixo horizontal mostra o intervalo de tempo expresso em dias. Nessa interface, ao aproximar o mouse de um dos dados, é possível obter informações das cores escolhidas, do endereço IP, da data e do total de escolhas até essa data (detalhe na Gráfico 5).

⁶¹ Fonte: Laboratório Virtual. Disponível em:
<<http://www.ideiasnacaixa.com/pesquisa/resultadoOpticao1Cor.php?dataInicio=2010-07-22&dataFim=2010-12-31>>. Acesso em: 26.jun.2012.

⁶² Fonte: Laboratório Virtual. Disponível em:
<<http://www.ideiasnacaixa.com/pesquisa/resultadoOpticao1Cor.php?dataInicio=2010-07-22&dataFim=2010-12-31>>. Acesso em: 26.jun.2012.

Gráfico 4 – Evolução das escolhas dos grupos de cores no tempo. Período selecionado: julho a dezembro de 2010.

Gráfico 5 – Detalhe mostrando que cada registro de dado é formado por um quadrado formado pelas três cores escolhidas (programa adaptado para o período entre julho e agosto de 2010).

A exibição da evolução das escolhas pelo tempo, fornecida por esse gráfico, permitiu observar o comportamento geral de como os grupos de cores foram selecionados pelos usuários. Grupos com mais escolhas (como os grupos 1, 3 e 9), por terem sido realizados em dias muito próximos, acabam se destacando pela formação de uma curva ascendente. Por

outro lado, grupos pouco escolhidos apresentam-se separados ao longo dos dias, formando um arranjo disperso, não tendo destaque muito especial no gráfico.

Observando o Gráfico 4 percebeu-se que, além dos grupos de cores 1, 3 e 9, as combinações formadas pelo magenta, ciano e azul (grupo 12) e pelo vermelho, verde e amarelo (grupo 4), também formavam uma curva perceptível.

Figura 52 – Três grupos de cores com quantidade considerável de escolhas.

Excetuando o grupo cuja resposta era a correta (grupo 1) e aquele relativas às “cores primárias” usadas na escola (grupo 3), é possível indicar no círculo cromático da interface (Figura 51a) os grupos de cores mais selecionados. O destaque é mostrado na Figura 52. O padrão parece revelar que o conjunto de cores mais próximas aos círculos vazios são os mais escolhidos, além de serem formados por cores adjacentes umas às outras.

A questão que se seguiu foi saber se esse padrão se repetiria, caso o arranjo de cores fosse alterado no *layout* da interface.

4.3 Alteração no *layout* da interface

A Figura 53 mostra a alteração feita no *layout* da interface em março de 2011. Em relação à Figura 51(a), o disco cromático recebeu uma rotação de 120° no sentido horário, sem que sua ordem fosse alterada.

Figura 53 – Alteração da distribuição das cores (versão de março de 2011).

O gráfico 6⁶³ mostra a evolução de escolhas de grupos de cores entre julho de 2010 a maio de 2011. Nesse período considerado, foram realizados 1.193 escolhas de cores na interface (“tabela de escolha dos grupos de cores”, <http://www.ideiasnacaixa.com/pesquisa/resultadoOpticao1Cor.php?dataInicio=2010-07-22&dataFim=2011-05-31>).

A linha vermelha indica o período em que a modificação no *layout* foi realizada.

⁶³ Fonte: Laboratório Virtual. Disponível em: <http://www.ideiasnacaixa.com/pesquisa/resultadoOpticao1Cor.php?dataInicio=2010-07-22&dataFim=2010-12-31>. Acesso em: 26.jun.2012.

Gráfico 6 – Evolução das escolhas de cores no tempo. Período selecionado: julho de 2010 a maio de 2011.

Figura 54 – Dois grupos de cores que começaram a ter mais escolhas.

Neste gráfico é possível perceber que a taxa de aumento de escolhas do grupo 1 (as cores da escolha correta) e do grupo 3 (a das “cores primárias” da escola) continuaram praticamente inalteradas. No entanto, os grupos 9 e 12 tiveram diminuição na frequência de escolhas, o que é constatada pelo aumento de distância entre esses pontos.

Por outro lado, o grupo 4, formado pelas cores vermelho, verde e amarelo, seguido pelo grupo 7, formado pelas cores vermelho, amarelo e magenta, começaram a se destacar devido à proximidade dos seus pontos (aumento na frequência de escolha). A Figura 54 apresenta a localização desses dois grupos na nova disposição do *layout*. Novamente, para esse novo desenho de interface, o padrão se repete: os grupos mais escolhidos são aqueles mais próximos aos círculos vazios de seleção.

4.4 Nova alteração no *layout* da interface

Outro detalhe para ser explorado na interface de animação das Cores era a ordem dos seus blocos. Na sequência mostrada na Figura 51 o usuário era convidado a realizar a tarefa antes de ser apresentado às explicações sobre as cores primárias de luz e as cores primárias de pigmento.

Dessa forma, a nova questão levantada era se uma nova organização desses blocos teria alguma influência no modo como as cores seriam escolhidas pelos usuários. Assim, em maio de 2011, foi feita uma nova versão, em que o bloco das escolhas das cores foi transferido para o fim da interface, como mostra a Figura 55. O resultado das quantidades dos grupos de cores escolhidas devido a essa alteração é mostrado no Gráfico 7⁶⁴.

Esse gráfico apresenta a evolução de escolhas de cores entre julho de 2010 a outubro de 2011. A linha vermelha indica o período em que a última modificação no *layout* foi realizada (maio de 2011). A linha amarela indica a modificação anterior (março de 2011).

⁶⁴ Fonte: Laboratório Virtual. Disponível em:

<<http://www.ideiasnacaixa.com/pesquisa/leituraPesquisaOptica01.html>>. Acesso em: 26.jun.2012.

Figura 55 – Nova ordem dos blocos da animação Cores: a) mistura de cores primárias de luz; b) animação do círculo cromático; c) mistura de cores primárias de pigmento e d) tarefa das cores. Versão de maio de 2011.

No período considerado, foram realizadas 1.406 escolhas de cores na interface (“tabela de escolha dos grupos de cores”, <http://www.ideiasnacaixa.com/pesquisa/resultadoOpticao1Cor.php?dataInicio=2010-07-22&dataFim=2011-10-31>).

Há um detalhe técnico que deve ser observado nesse gráfico. Entre meados de junho e início de julho de 2011 o *website* Laboratório Virtual ficou fora do ar. Isso é facilmente identificado no Gráfico 7, no intervalo de dados logo após a linha vermelha. Essa foi uma ocorrência relacionada à suspensão temporária do serviço de hospedagem⁶⁵. Esse problema acarretou a impossibilidade do público acessar as animações nesse período. A contagem retornou à normalidade assim que o serviço foi retomado.

⁶⁵ O período de renovação do domínio www.ideiasnacaixa.com havia vencido e a perda do prazo legal acarretou na suspensão do seu serviço. Situações como essas não são frequentes, mas os trabalhos na web estão sujeitos a diferentes imprevistos técnicos que podem causar a interrupção da exibição de um *website*.

Gráfico 7 – Evolução das escolhas de cores no tempo. Período selecionado: julho de 2010 a outubro de 2011.

No Gráfico 7 observa-se que, apesar das explicações terem sido colocadas antes da tarefa, os grupos de cores, anteriormente relatados, continuaram sendo escolhidas tanto quanto antes. O grupo 4, relativo às cores próximas aos círculos vazios, teve uma queda na taxa de crescimento, mesmo assim continuou sendo expressiva. O grupo 3, das “cores primárias” usadas nas escolas, também sofreu uma ligeira queda, no entanto, permaneceu como uma opção escolhida pelos usuários.

O Gráfico 8⁶⁶ apresenta a variação das escolhas dos grupos entre julho de 2010 a junho de 2012. Como o gráfico representa o histórico de 24 meses apresentado no mesmo espaço que os gráficos anteriores, os pontos dos grupos aparecem mais próximos, evidenciando suas curvas de crescimento. Assim, após 12 meses depois da última alteração, os grupos 1 e 3 continuaram sendo os mais escolhidos; os grupos de cores próximos aos círculos vazios mantiveram suas taxas de crescimento (grupo 4 e 7, e também o grupo 9).

⁶⁶ Fonte: Laboratório Virtual. Disponível em: <http://www.ideiasnacaixa.com/pesquisa/leituraPesquisaOptica01.html>. Acesso em: 30.jun.2012.

Gráfico 8 – Evolução das escolhas de cores no tempo. Período selecionado: julho de 2010 a junho de 2012.

Outra observação se refere ao grupo 11, formado pelas cores ciano, magenta e amarelo, que começou a se destacar no gráfico. Esse grupo não era muito selecionado antes da última modificação da interface. Provavelmente isso se reverteu por ele representar as cores primárias de pigmento, informação que passou a aparecer antes do usuário realizar a tarefa de escolha de cores.

394		grupo 1
60		grupo 2
376		grupo 3
177		grupo 4
60		grupo 5
18		grupo 6
120		grupo 7
43		grupo 8
203		grupo 9
29		grupo 10
82		grupo 11
105		grupo 12
21		grupo 13
27		grupo 14
31		grupo 15
37		grupo 16
38		grupo 17
20		grupo 18
21		grupo 19
12		grupo 20

Tabela 3 – Total de seleções de cores. Período: 22 de julho a 30 de junho de 2012.

Desde julho de 2010 até o final das análises desse trabalho, que se encerraram em junho de 2012, foram registradas 1.878 escolhas de grupos de cores (“tabela de escolha dos grupos de cores”, <http://www.ideiasnacaixa.com/pesquisa/resultadoOptica01Cor.php?dataInicio=2010-07-22&dataFim=2012-06-30>). Finalmente, Na Tabela 3⁶⁷ (“tabela de escolha dos grupos de cores”), é apresentado o total dessas escolhas, distribuídas pelos diferentes grupos de cores.

⁶⁷ Fonte: Laboratório Virtual. Disponível em: <<http://www.ideiasnacaixa.com/pesquisa/resultadoOptica01Cor.php?dataInicio=2010-07-22&dataFim=2012-06-30>>. Acesso em: 30.jun.2012.

4.5 Resultados obtidos com a interface de animação Correção de defeitos da visão

Na animação Correção de defeitos da visão o usuário era convidado a responder como se corrigia a miopia e a hipermetropia arrastando uma lente divergente e uma convergente para a parte frontal de cada um dos olhos que apresentavam esses defeitos. A resposta era coletada e enviada para um banco de dados, por meio de um programa PHP.

resultado de escolhas de correção de visão

formato do requisito:
?dataInicio=AAAA-MM-DD&dataFim=AAAA-MM-DD

id	ip	resposta	data	horário
3146	201.79.53.49	hiperSim	27-06-2012	01:41:53
3147	201.79.53.49	hiperNao	27-06-2012	01:42:17
3148	201.79.53.49	hiperNao	27-06-2012	01:42:25
3149	201.79.53.49	miopiaNao	27-06-2012	01:42:29
3150	201.79.53.49	hiperSim	27-06-2012	01:42:42
3151	201.79.53.49	miopiaSim	27-06-2012	01:42:56
3152	201.79.53.49	miopiaSim	27-06-2012	01:45:36
3153	201.79.53.49	hiperSim	27-06-2012	01:46:08
3154	201.79.53.49	hiperNao	27-06-2012	01:46:14
3155	187.126.169.122	miopiaSim	28-06-2012	09:22:09
3156	187.126.169.122	miopiaNao	28-06-2012	09:22:16
3157	187.126.169.122	hiperSim	28-06-2012	09:22:26
3158	187.126.169.122	hiperSim	28-06-2012	09:22:31
3159	187.126.169.122	hiperNao	28-06-2012	09:22:36
3160	187.126.169.122	miopiaSim	28-06-2012	09:22:42
3161	187.116.119.50	hiperNao	29-06-2012	01:52:20
3162	187.116.119.50	hiperSim	29-06-2012	01:52:24
3163	187.116.119.50	miopiaSim	29-06-2012	01:52:27
3164	187.116.119.50	hiperNao	29-06-2012	01:52:35
3165	187.116.119.50	miopiaSim	29-06-2012	01:52:37

Tabela 4 – Parte das amostragens das respostas da correção de defeitos da visão. Amostragem de 22 de junho de 2011 a 30 de junho de 2012.

Os resultados da coleta de *Log* de dados dessa animação podem ser observados a partir da página eletrônica intitulada *Resultado de escolhas de correção de visão* (<http://www.ideiasnacaixa.com/pesquisa/resultadoOlhoCorrecao.php>). Essa página apresenta uma tabela contendo a informação de identificação numérica do dado (id), o endereço IP do computador

utilizado pelo usuário (ip), a resposta das interações (*miopiaSim*, *miopiaNao*, *hiperSim* e *hiperNao*), a data e o horário da interação do usuário com a interface (Tabela 4⁶⁸).

Entre o período de 22 de junho de 2011 a 30 de junho de 2012 foram contabilizados 3.165 respostas. Destas, 1.076 corresponderam à resposta correta da correção da miopia (lente divergente) e 1.028, à resposta correta da correção da hipermetropia (lente convergente). Do total, 606 foram respostas erradas na correção da miopia e 456, na da hipermetropia. Considerando apenas as respostas corretas, 2.104 (66%) corresponderam à opção esperada. Por outro lado, 1.062 (34%) corresponderam à resposta errada. Esses resultados podem ser consultados na tabela *Resultado de escolhas de correção de visão* (<http://www.ideiasnacaixa.com/pesquisa/resultadoOlhoCorrecao.php>), na parte inferior da página.

4.6 Conhecendo o usuário

Conhecer o usuário faz parte da etapa de *pré-design* da produção de um *software* (no modelo de *Engenharia de Usabilidade*, descrito por Rocha e Baranauskas, 2003).

Para determinar o perfil dos usuários do Laboratório Virtual foi utilizado um formulário *online* hospedado no próprio *website* do projeto e identificado no menu de *links*. O usuário acessava a página e respondia espontaneamente às questões formuladas, que eram enviadas ao banco de dados.

As informações armazenadas contendo as respostas dos usuários foram resgatadas por meio de um programa PHP para que fossem apresentados em forma de uma tabela. Essa página tem o aspecto mostrado na Tabela 5⁶⁹, podendo ser acessada no endereço <http://www.ideiasnacaixa.com/pesquisa/resultadoPesquisaUsuario.php>.

As informações exibidas por essa tabela estão organizadas do seguinte modo:

⁶⁸ Fonte: Laboratório Virtual. Disponível em: <<http://www.ideiasnacaixa.com/pesquisa/resultadoOlhoCorrecao.php>>. Acesso em: 08.jul.2012.

⁶⁹ Fonte: Laboratório Virtual. Disponível em: <<http://www.ideiasnacaixa.com/pesquisa/resultadoPesquisaUsuario.php>>. Acesso em: 01.jul.2012.

1. id (índice, corresponde a contagem de usuários listados na tabela);
2. ip (identificação do endereço do computador que foi feito o preenchimento do questionário);
3. usuário (podendo ser aluno, professor ou usuário);
4. fundamental (se o aluno for do ensino fundamental a célula será preenchida com a série que indicou no questionário);
5. médio (se o aluno for do ensino médio a célula será preenchida com a série que indicou no questionário);
6. superior (caso o usuário tenha formação superior esse campo aparecerá preenchido);
7. outro (campo para livre preenchimento da formação do usuário);
8. comentário (campo para livre preenchimento de comentários do usuário);
9. data (mostra a data e a hora em que a informação do questionário foi enviada).

No final dessa página ainda são contabilizadas as informações específicas de cada grupo de usuários (aluno, professor e usuário) e que alimentaram a construção dos gráficos apresentados nesse capítulo. Os detalhes da programação podem ser vistos no Apêndice E.

resultado do questionário preenchido pelos usuários

id	ip	usuario	fundamental	medio	superior	outro	comentario	data
1	201.7.143.5	aluno	0					2012-08-27 11:23:21
2	201.9.204.189	aluno	3				o que e uma nebulosa	2012-08-27 09:29:55
3	201.73.164.98	aluno	9					2012-08-25 09:21:34
4	189.110.18.71	aluno		2				2012-08-25 02:50:35
5	177.83.239.241	aluno				Mestrado		2012-08-18 05:21:58
6	189.102.6.214	aluno	2				bom	2012-08-16 10:28:15
7	187.126.161.252	aluno		1				2012-08-16 04:35:46
8	186.218.225.126	aluno		1				2012-08-14 02:51:57
9	189.12.117.182	aluno		1				2012-08-14 01:45:08
10	187.35.155.240	aluno			superior		Queria mais informações sobre o raio laser	2012-08-13 12:33:58

Tabela 5 – Resultado parcial dos questionários preenchidos pelos usuários.

Gráfico 9 – Perfil dos usuários do Laboratório Virtual obtidos entre 25 de novembro de 2010 a 27 de junho de 2012.

Os dados obtidos entre 25 de novembro de 2010 a 27 de junho de 2012 (“resultado do questionário preenchido pelos usuários”, <http://www.ideiasnacaixa.com/pesquisa/resultadoPesquisaUsuario.php>. Acesso em 01.jul.2012) mostram o resultado do preenchimento do questionário realizado por 3.907 indivíduos. Destes, 1.997 (51%) se identificaram como professor, 1.501 (38%) se identificaram como aluno e 429 (11%), como usuário (Gráfico 9⁷⁰).

Em relação aos alunos, de um total de 1.501 registros, 706 (47%) informaram que cursavam o ensino fundamental e 486 (32%), o ensino médio. Os registros de 309 (21%) alunos correspondiam a outros cursos (ensino superior, técnico, pré-vestibular ou outros) (Gráfico 10⁷¹).

Gráfico 10 – O que cursam os alunos que preencheram o questionário do Laboratório Virtual.

⁷⁰ Fonte: Laboratório Virtual. Resultado do questionário preenchido pelos usuários. Disponível em: <<http://www.ideiasnacaixa.com/pesquisa/resultadoPesquisaUsuario.php>>.

⁷¹ Fonte: Laboratório Virtual. Resultado do questionário preenchido pelos usuários. Disponível em: <<http://www.ideiasnacaixa.com/pesquisa/resultadoPesquisaUsuario.php>>.

Gráfico 11 – Quantidade de acessos dos alunos, segundo suas séries.

O Gráfico 11⁷² mostra a distribuição dos alunos em relação às séries. Para o ensino fundamental, parece haver uma tendência do acesso ao Laboratório Virtual ser maior para séries mais adiantadas. No ensino médio os acessos parecem que ocorrem de modo semelhante nas três séries.

Gráfico 12 – Onde os professores lecionam.

⁷² Fonte: Laboratório Virtual. Resultado do questionário preenchido pelos usuários. Disponível em: <<http://www.ideiasnacaixa.com/pesquisa/resultadoPesquisaUsuario.php>>.

Em relação aos professores, dos 1.977 registros, 640 (32%) corresponderam exclusivamente aos que lecionavam no ensino fundamental; 541 (27%) lecionavam exclusivamente no ensino médio; 489 (25%) lecionavam tanto no ensino fundamental como no médio; 72 (4%) corresponderam a professores de outros cursos (superior, técnico, pré-vestibular, etc.); 27 (1%) davam aulas em outros cursos além do ensino fundamental e/ou médio (Gráfico 12⁷³). Os que não deram informações corresponderam a 208 (11%) registros.

Também foi possível relacionar as áreas de ensino dos professores, segundo estrutura dos Parâmetros Curriculares Nacionais (PCNs) para o ensino fundamental e médio (Gráfico 13⁷⁴).

Gráfico 13 – Áreas de ensino dos professores segundo os PCNs.

⁷³ Fonte: Laboratório Virtual. Resultado do questionário preenchido pelos usuários. Disponível em: <<http://www.ideiasnacaixa.com/pesquisa/resultadoPesquisaUsuario.php>>.

⁷⁴ Fonte: Laboratório Virtual. Resultado do questionário preenchido pelos usuários. Disponível em: <<http://www.ideiasnacaixa.com/pesquisa/resultadoPesquisaUsuario.php>>.

Finalmente, em relação aos usuários que não eram alunos e nem professores, dos 429 registros, 160 (37%) se identificaram tendo formação superior; 116 (27%) eram formados no ensino médio; e 97 (23%), tinham o ensino fundamental (Gráfico 14⁷⁵). Usuários que se identificaram com outro tipo de formação foram 30 (7%) e 26 (6%) não relacionaram nenhum tipo de formação.

Gráfico 14 – Formação dos usuários que não eram alunos nem professores.

Não foi produzida nenhuma interface para a exibição de informações qualitativas do questionário, como matérias preferidas dos alunos, profissão dos usuários ou comentários. Por isso, esses dados não foram avaliados nessa pesquisa.

⁷⁵ Fonte: Laboratório Virtual. Resultado do questionário preenchido pelos usuários. Disponível em: <<http://www.ideiasnacaixa.com/pesquisa/resultadoPesquisaUsuario.php>>.

4.7 Google Analytics

A partir do *Google Analytics* foram obtidos os dados de visualizações das páginas e o tempo de médio da visita nas interfaces de animação do Laboratório Virtual, realizadas entre 1º de janeiro a 30 de junho de 2012.

O Gráfico 15⁷⁶ apresenta a “visualização de página” e “visualização única de página” de cada uma das doze animações. Segundo o Google, uma “visualização de página” é aquela que contabiliza todos os acessos, mesmo aqueles feitos pelo mesmo usuário quando a página é atualizada. Por outro lado, a “visualização única de página” representa o número de sessões durante as quais a página foi visualizada uma ou mais vezes pelo mesmo usuário.

Gráfico 15 – Visualizações de páginas e Visualizações únicas entre 1º de janeiro a 30 de julho de 2012.

Considerando o grupo das doze animações do Laboratório Virtual produzidas para esse projeto, a média de visualização de páginas para o período considerado, de 1º de janeiro a 30 de julho de 2012, foi de 1.013 vezes; por sua vez, houve 757 visualizações únicas de páginas.

⁷⁶ Fonte: *Google Analytics*.

O Gráfico 16⁷⁷ apresenta o tempo médio de visitas dos usuários nas animações do Laboratório Virtual. Considerando o grupo das doze animações, a média de tempo de permanência foi de 40 segundos.

Gráfico 16 – Tempo médio de permanência das páginas entre 1º de janeiro a 30 de julho de 2012.

⁷⁷ Fonte: *Google Analytics*.

5. CONCLUSÃO

Um das propostas iniciais desse projeto foi a produção de animações interativas cujo tema foi a Óptica. As interfaces produzidas com o programa *Adobe Flash* ofereceram um ambiente apropriado para traduzir as equações de formação da imagem em diagramas animados, utilizando para isso, a linguagem de programação *action script*. Posteriormente, foram introduzidos elementos de programação para realizar coletas de dados e fornecer alguma informação sobre o modo como os usuários interagem com a interface. Esse recurso, denominado *Log* de dados e realizado no ambiente natural de trabalho do usuário, levou à avaliação da produção da interface, em um processo descrito pelo Ciclo de Vida de *Softwares*. Também contribuiu para avaliar as animações do Laboratório Virtual a audiência verificada através do serviço do *Google Analytics*.

Ao todo foram produzidas doze animações para o Laboratório Virtual para essa pesquisa. A média de visualizações de páginas dessas animações, realizadas entre os dias 1º de janeiro a 30 de julho de 2012, foi de 1.013 visualizações, sendo que a média de tempo de cada uma das visualizações foi de 40 segundos (dados do *Google Analytics*, acesso em: 1.jan.2012 a 30.07.2012).

Dessas doze animações, onze tiveram seu processo de produção realizado de modo semelhante ao modelo Cascata de Ciclo de Vida. Essas produções praticamente não tiveram alterações desde a sua concepção até sua publicação no Laboratório Virtual (Figura 56⁷⁸).

Em duas das produções foi incluído o método de coleta de *Log* de dados para obter informações sobre escolhas do usuário sobre a interface: a animação *Correção de defeitos da visão* e *Cores*. Esse método foi adotado isoladamente, sem que outros processos de observação sobre o usuário tivessem sido realizados (gravação de vídeo, por exemplo). Essa opção foi decorrente das escalas de desenvolvimento reduzidas no projeto (apenas um profissional envolvido para a programação e implementação do sistema). Além disso, desejava-se conhecer quais informações o método de *Log* poderia fornecer.

⁷⁸ Fonte: Adaptado de Preece, Rogers e Sharp (2005, p. 208).

Figura 56 – O processo de produção das animações do Laboratório Virtual na perspectiva do modelo Cascata de Ciclo de Vida.

Na *Correção de defeitos da visão* o resultado da coleta de *Log* de dados correspondeu à etapa final da produção de sua interface. Ou seja, os dados obtidos não interferiram na construção do *software* (por isso, o Ciclo de Vida da animação *Correção de defeitos da visão* também foi considerado como Cascata).

Já na interface *Cores* o método de *Log* de dados proporcionou uma revisão da produção dessa animação, com alterações no desenho do projeto e nova análise de dados. Apesar de não contar com tantas etapas e complexos processos de avaliação, comparados aos modelos de Ciclo de Vida de *software*, ela se assemelha ao ciclo de Engenharia de Usabilidade, conforme proposto por Rocha e Baranauskas (2003, p. 120).

As escolhas de grupos de cores realizadas pelos usuários despertaram a atenção devido ao seu resultado não esperado (esse resultado criou a necessidade de uma avaliação de Usabilidade específica, sem descrição nas diretrizes propostas por Nielsen (2002)). Em vez das respostas indicarem um padrão relacionado às orientações contidas na animação ou, então, a alguns agrupamentos de cores aleatórios, os usuários apresentaram uma tendência a fazer escolhas relacionadas à disposição dos elementos na interface. Muitos grupos

selecionados eram formados por cores adjacentes e próximos aos círculos vazios de escolha. Esse resultado foi verificado mesmo quando se alterou a posição das cores. A avaliação dessa Usabilidade específica, portanto, revelou a influência do *design* nas escolhas do usuário. Também não influenciou positivamente nos resultados as explicações dadas na interface sobre a diferença entre cores primárias de luz e de pigmento antes da realização da tarefa solicitada ao usuário.

Assim, o método de *Log* de dados demonstrou ser uma técnica viável e funcional para o registro das informações que auxiliam na avaliação da produção de uma interface.

Figura 57 – O processo de produção da animação Cores do Laboratório Virtual na perspectiva do modelo de Ciclo de Vida da Engenharia de Usabilidade⁷⁹.

Essa pesquisa foi complementada com a determinação do perfil do usuário do Laboratório Virtual. Dos preenchimentos do questionário realizados entre 25 de novembro de 2010 a 27 de junho de 2012, verificou-se que 38% correspondeu a alunos e 51%, a professores. Ou seja, de um total de 3.907 questionários preenchidos, 89% se relacionou a

⁷⁹ Fonte: Adaptado de Preece, Rogers e Sharp (2005, p. 123).

indivíduos pertencentes ao meio escolar, resultado que contribui para vincular as animações do Laboratório Virtual ao seu público alvo.

Todos os trabalhos desse projeto estão disponíveis na *web* nos endereços relacionados abaixo:

Endereço principal do site do Laboratório Virtual:

www.ideiasnacaixa.com/laboratoriovirtual

Reflexão:

www.ideiasnacaixa.com/laboratoriovirtual/pequisaOpticaReflexao.html

Refração:

www.ideiasnacaixa.com/laboratoriovirtual/pequisaOpticaRefracao.html

Lentes:

www.ideiasnacaixa.com/laboratoriovirtual/pequisaOptica01.html

Espelhos esféricos:

www.ideiasnacaixa.com/laboratoriovirtual/pequisaOpticaEspelhos.html

O olho:

www.ideiasnacaixa.com/laboratoriovirtual/pequisaOpticaOlhoImagem.html

Olho normal:

www.ideiasnacaixa.com/laboratoriovirtual/pequisaOptica03olho.html

Miopia:

www.ideiasnacaixa.com/laboratoriovirtual/pequisaOptica04olhoMiope.html

Hipermetropia:

www.ideiasnacaixa.com/laboratoriovirtual/pequisaOptica05olhoHipermetrope.html

Correção:

www.ideiasnacaixa.com/laboratoriovirtual/pequisaOpticaOlhoImagemCorrecao.html

Cores:

www.ideiasnacaixa.com/laboratoriovirtual/pequisaOptica02cores.html

Construindo onda:

www.ideiasnacaixa.com/laboratoriovirtual/ondulatoriaParte01.html

Interferência de ondas:

www.ideiasnacaixa.com/laboratoriovirtual/ondulatoriaInterferencia.html

Os dados coletados na animação Correção de defeitos da visão podem ser consultados em:

www.ideiasnacaixa.com/pesquisa/resultadoOlhoCorrecao.php

Os dados coletados na animação Cores podem ser consultados em:

www.ideiasnacaixa.com/pesquisa/resultadoOpticao1Cor.php

A evolução das escolhas dos grupos de cores na animação Cores pode ser visualizado em gráfico em:

www.ideiasnacaixa.com/pesquisa/leituraPesquisaOptica01.html

O questionário do perfil de usuário pode ser acessado em:

www.ideiasnacaixa.com/laboratoriovirtual/pesquisaUsuario.html

Os detalhes dos *scripts* de programação no *Flash* e em PHP podem ser observados nos Apêndices B, C, D e E. As tabelas de banco de dados e as informações de audiência do Laboratório Virtual fornecidas pelo serviço *Google Analytics* são de acesso privado.

6. CONSIDERAÇÕES FINAIS

Um OA pode ser avaliado pelas características que o definem e que permitem que seja utilizado em um ambiente virtual de aprendizagem (granularidade, modularidade, metadados, etc). A arquitetura de programação e as áreas interativas, no entanto, também devem ter sua ação verificada quando utilizadas pelo usuário. Como exemplo, os resultados apresentados nesse trabalho mostram como a disposição de elementos em simulações pode influenciar uma resposta não esperada.

O método de *Log* de dados é apenas um dos recursos para se avaliar um OA e deve ser utilizado como apoio a outras técnicas, como por exemplo, questionários, entrevistas, relatórios ou gravações de vídeo. O uso desses métodos pode proporcionar a melhora da Usabilidade de uma interface, porém, é preciso perceber que o potencial pedagógico de um OA será ampliado quando o indivíduo tiver apoio de um mediador, como um professor em sala de aula. Nesse caso, vários problemas relacionados à construção de interfaces podem ser minimizados.

O principal resultado desse trabalho, as respostas inesperadas dos usuários, serve de alerta a uma importante questão que deve ser discutida ao introduzir as TIC no ensino. Caso um OA seja utilizado sem que ocorra uma avaliação de Usabilidade e sem mediação adequada, poderá não ser aplicada adequadamente como uma ferramenta na aprendizagem de conceitos. É claro que a adoção da perspectiva da IHC, no qual a produção de uma interface é centrada no usuário, faz com que o suporte tecnológico tenha de ser construído de tal modo a facilitar seu uso e sua aprendizagem; caso ocorra algum problema de Usabilidade, este será do sistema computacional e não do indivíduo. Esta prática, em si, já poderia evitar problemas na introdução das TIC no ambiente de ensino.

Entretanto, não basta apenas se preocupar em melhorar o sistema computacional para utilizá-lo como ferramenta auxiliadora na aprendizagem em sala de aula. É necessário que a sua introdução no ambiente escolar seja realizada de modo crítico, evitando que o computador se contraponha ao método tradicional de ensino de um professor, ou que seja utilizado de modo indiscriminado, interferindo no processo de aprendizagem.

Um exemplo disso pode ser observado em matéria publicada na Folha de S. Paulo que aborda a introdução do *tablet* em escolas americanas (HU, 2011):

Modas tecnológicas vêm e vão nas escolas, e outras experiências cujo objetivo era propiciar melhor experiência a crianças criadas na era dos videogames apresentaram resultados contraditórios. Os educadores, por exemplo, continuam divididos quanto aos resultados das iniciativas para oferecer um *laptop* a cada aluno.

No momento no qual os distritos escolares tentam conseguir aprovação para seus orçamentos, gastar dinheiro na compra de computadores *tablet* pode parecer uma extravagância. E alguns pais e pesquisadores expressaram preocupação por as escolas investirem no equipamento antes que seu valor educacional tenha sido comprovado.

Com a iminente presença do computador e do acesso à Internet no âmbito escolar, o uso das TIC deve ser realizado de modo a manter compromissos pedagógicos consistentes com a realidade da sala de aula e a de seus membros. Isso significa observar as características do ambiente social em relação às situações de infra-estrutura, econômica, política e histórica. Medidas descontextualizadas podem representar mais perdas do que ganhos no espaço de ensino.

Assim, em um ambiente que promova um efetivo diálogo, a introdução e o uso das ferramentas computacionais deve ser discutida com clareza, sendo que o método de investigação de Usabilidade de OA pode se tornar um aliado ao favorecer a participação de alunos e professores na produção de interfaces que se aproximem das reais necessidades do espaço de aprendizagem.

6.1 Perspectivas futuras

Essa pesquisa explorou o método de coleta de *Log* de dados para permitir a avaliação e a reestruturação do projeto de uma interface. A técnica demonstrou ser funcional e pode ser utilizada em projetos futuros que incluam outros meios de coletas de dados, como a observação direta, a entrevista ou a gravação de vídeo, para subsidiar uma avaliação mais ampla do uso de um *software*.

Essas investigações poderiam ser realizadas tanto nos espaços de ensino formal como nos de ensino não formal (em centros de ciências, por exemplo) e verificar como as respostas poderiam ser influenciadas, não apenas pelo desenho da interface, mas principalmente pela abordagem de mediação realizada em cada um desses locais. Por outro lado, também seria

interessante observar que contribuições um espaço promoveria ao outro para a construção do conhecimento e quanto isso seria auxiliado pelas ferramentas tecnológicas.

Novamente é importante alertar que interfaces interativas nem sempre podem substituir instrumentos tracionais de ensino, como as salas de laboratório, por exemplo. No entanto, na educação a distância o aluno pode não ter acesso a equipamentos para a realização de experiências e, portanto, os *softwares* representariam um importante suporte para suprir as necessidades dessa modalidade de ensino, sem jamais substituí-las.

Dessa forma, seria interessante explorar o método do *Log* de dados para registrar tipos de interações mais complexas, como desenhos realizados pelos usuários em uma interface. Uma animação virtual, por exemplo, poderia oferecer como tarefa o traçado de um diagrama de formação de imagem através de lentes ou de espelhos. As diversas etapas dessa interação poderiam ser gravadas durante a execução do exercício. Assim, além de poder ser utilizada para a avaliação da própria interface, essas informações forneceria ao professor um importante instrumento de registro do processo da atividade realizada pelo usuário, contribuindo para estruturar uma orientação mais adequada a esse aluno virtual.

7. REFERÊNCIAS⁸⁰

ACM SIGCHI. **Curricula for Human-Computer Interaction**. 1992. Última atualização: 29.jul.2009. Disponível em: <http://old.sigchi.org/cdg/cdg2.html#2_1>. Acesso em: may.2012.

ALVES, A. J. **O planejamento de pesquisas qualitativas em educação**. CAD. Pesq., São Paulo (77): 53-61, maio 1991.

AYRES, C. **O uso do recurso multimídia no ensino de química para alunos de ensino médio sobre o conteúdo de ligações intermoleculares**. 2011. 319 f. Dissertação (Mestrado) – Faculdade de Educação, Instituto de Física, Instituto de Química e Instituto de Biociências, Universidade de São Paulo, São Paulo. 2011.

BAPTISTA, C. M. **Msys: uma ferramenta de acompanhamento de atividades para sistemas de aprendizado eletrônico**. 2007. 149 p. Dissertação (Mestrado) – Escola Politécnica da Universidade de São Paulo, São Paulo. 2007.

BETTIO, R. W.; MARTINS, A. **Objetos de aprendizado: um novo modelo direcionado ao ensino a distância**. Disponível em: <<http://www.abed.org.br/congresso2002/trabalhos/texto42.htm>>. Acesso em: 30.abr.2005.

BONETTI, M. C. **A linguagem de vídeo e a natureza da aprendizagem**. 2008. 172 f. Dissertação (Mestrado) – Faculdade de Educação, Instituto de Física, Instituto de Química e Instituto de Biociências, Universidade de São Paulo, São Paulo. 2008.

BRASIL. Ministério da Educação. **Secretaria de Educação a Distância**. Disponível em: <http://portal.mec.gov.br/index.php?option=com_content&view=article&id=289&Itemid=822>. Acesso em: 15.jul.2012).

CASTELLS, M. **A galáxia da internet**. Rio de Janeiro, Jorge Zahar Eds, 2003.

_____. **La dimensión cultural de Internet**. Disponível em: <<http://www.uoc.edu/culturaxxi/esp/articles/castells0502/castells0502.html>>. Acesso em: 14.fev.2011.

⁸⁰ De acordo com a Associação Brasileira de Normas Técnicas. NBR 6023.

COLAGRANDE, E. A. **Desenvolvimento de um jogo didático virtual para a aprendizagem do conceito mol**. 2008. 110 f. Dissertação (Mestrado) – Faculdade de Educação, Instituto de Física, Instituto de Química e Instituto de Biociências, Universidade de São Paulo, São Paulo. 2008.

COLELLO, S. **Para onde vai o ensino da escrita**. In: A escola que (não) ensina a escrever. São Paulo: Paz e Terra, 2007.

COLL, C.; MONEREO, C. **Psicologia da Educação Virtual**. Porto Alegre: Artmed, 2010.

COMITÊ GESTOR DA INTERNET NO BRASIL. **Pesquisa TIC Domicílios 2011 - Área Urbana**. Disponível em: <<http://cetic.br/usuarios/tic/2010/apresentacao-tic-domicilios-2010.pdf>>. Acesso em: 24.jul.2011.

CORRADI, M. I. **Construção de objetos de aprendizagem para apoio ao processo ensino-aprendizagem do exame físico em enfermagem**. 2008. 146 f. Dissertação (Mestrado) – Pontifícia Universidade Católica, Paraná, Curitiba. 2008.

COX, M. I. P. **Oralidade e escrita segundo Walter J. Ong**. Polifonia, n. 1, p. 166-169. Cuiabá, 1994.

DANIELS, H. **Vigotsky e a Pedagogia**. São Paulo: Loyola, 2003.

DESCARTES, R. **O Mundo ou Tratado da Luz**. São Paulo: Hedra, 2008.

ECO, U. **A Arte Perdida da Caligrafia**. Revista da Cultura, São Paulo, p. 29, novembro. 2009.

ECO, U.; CARRIERE, J. C. **Não Contem com o Fim do Livro**. Rio de Janeiro: Record, 2010.

ESTAÇÃO CIÊNCIA. Disponível em: <www.eciencia.usp.br>.

FERECINI, G. M. **Desenvolvimento e avaliação do objeto digital de aprendizagem sobre o aleitamento materno do prematuro**. 2011. 158 f. Tese (Doutorado) - Escola de Enfermagem de Ribeirão Preto, Universidade de São Paulo, Ribeirão Preto. 2011.

FERREIRO, E. **Ler e escrever num mundo em transformação**. In: Passado e Presente dos Verbos Ler e Escrever. São Paulo: Cortez, 2002.

FILARDI, A. L. **Análise e avaliação de técnicas de interação humano-computador para sistemas de recuperação de imagens por conteúdo baseadas em um estudo de caso**. 2007. 149 f. Dissertação (Mestrado) – Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo, São Carlos. 2007.

FILHO, C. S. S.; MACHADO, E. C. **O computador como agente transformador da educação e o papel do objeto de aprendizagem**. 2003. Disponível em: <<http://www.abed.org.br/seminario2003/texto11.htm>>. Acesso em: maio.2012.

FREIRE, M. **A Paixão de Conhecer o Mundo**. São Paulo: Paz e Terra, 1983.

FREITAS, M. T. A. **Vygotsky e Bakhtin – psicologia e educação: um intertexto**. São Paulo: Ática, 1994.

GALVÃO, A. M. O.; BATISTA, A. A. G. **Oralidade e escrita: uma revisão**. Cadernos de Pesquisa, v. 36, n. 128, p. 403-432, maio/ago. 2006.

GARCIA, J. W. (org). **Amazing Amazon: estética evolucionaria**. Catálogo da exposição realizada no Museu de arte Moderna. São Paulo, 2002.

GARNIER, C. et. al. **Após Vygotsky e Piaget**. Porto Alegre: Artes médicas, 1996.

GERALDI, W. **Portos de passagem**. São Paulo: Martins Fontes, 1993.

GIBBONS, A. S.; NELSON, J.; Richards, R. **The nature and origin of instructional objects**. Idaho National Engineering and Enviromental Laboratory, Idaho Falls, 2000. Disponível em: <http://dlsystems.us/readings/NTFL_1/Learning_Objects/1_2_gibbons.pdf>. Acesso em: may.2012.

GÓES, F. S. N. **Desenvolvimento e avaliação de objeto virtual de aprendizagem interativo sobre o raciocínio diagnóstico em enfermagem aplicado ao recém-nascido pré-termo**. 2010. 188 f. Tese (Doutorado) – Escola de Enfermagem de Ribeirão Preto, Universidade de São Paulo, Ribeirão Preto. 2010.

HARTSON, H. R.; CASTILHO, J. C.; KELSO, J. **Remote evaluation: the network as an extension of the usability laboratory**. Disponível em: <http://www.sigchi.org/chi96/proceedings/papers/Hartson/hrh_txt.htm>. Acesso em: 30.may.2012.

HECHT, E. **Óptica**. Lisboa: Calouste Gulbenkian, 2002.

HU, W. Escolas americanas adotam iPad para ser usado em casa e em sala de aula. **Folha de São Paulo**, São Paulo, 6.jan.2011. Caderno Cotidiano. Disponível em: <<http://www1.folha.uol.com.br/fsp/cotidian/ff0601201111.htm>>. Acesso em: 20.jul.2012.

IEEE Learning Technology Standards Committee (LTSC). **Draft standard for learning object metadata**. (IEEE 1484.12.1-2002). New York, NY. 2002. Disponível em: <http://ltsc.ieee.org/wg12/files/LOM_1484_12_1_v1_Final_Draft.pdf>. Acesso em: 29.may.2012.

INSTITUTO BRASILEIRO DE GEOGRAFIA E ESTATÍSTICA. **Pesquisa Nacional por Amostra de Domicílios 2008**. Diretoria de Pesquisas, Coordenação de Trabalho e Rendimento. Disponível em: <http://www.ibge.gov.br/home/estatistica/populacao/acessoainternet2008/internet.pdf>. Acesso em: 13.jul.2012.

JACOBBER, E. C. **Proposta e implementação de uma interface para motores de jogos interativa e centrada no usuário**. 2007. 175 f. Dissertação (Mestrado) – Escola Politécnica da Universidade de São Paulo, São Paulo. 2007.

JACOBSEN, P. **Reusable Learnig Objects – What does the future hold?** E-learnig Magazine. Nov. 2002. Disponível em: <<http://www.mcli.dist.maricopa.edu/ocotillo/retreat02/docs/rlos.pdf>>. Acesso em: may.2012.

LACASA, P. **La escuela: uma sociedad dentro de outra?** In: Aprender en la escuela, aprender en la calle. Espanha: Editorial Antonio Machado Libros, 1994.

LEONTIEV, A. N. **O desenvolvimento do psiquismo**. Lisboa: Livros Horizonte, 1978.

LÉVY, P. **Cibercultura**. São Paulo: Editora 34, 1999.

_____. **O ciberespaço como um passo metaevolutivo**. Revista Famecos, Porto Alegre, n.o 13. p. 59-67, dezembro. 2000.

_____. **A máquina universo**. Lisboa: Instituto Piaget, 1987.

LÉVY, P. **O que é virtual**. São Paulo: Editora 34, 2005.

_____. **As tecnologias da inteligência**. São Paulo: Editora 34, 1995.

LÉVY, P.; MICHEL, A. **As árvores de conhecimentos**. São Paulo: Escuta, 2000.

MATSUKUMA, M. **Divulgação científica da Estação Ciência pela internet**. In: CRESTANA, Silvério et al. Educação para a Ciência: Curso para Treinamento em Centros e Museus de Ciência. Editora Livraria da Física, 2001.

_____. **Laboratório virtual**. Disponível em:
<<http://www.ideiasnacaixa.com/laboratoriovirtual>>. Acesso em: jul.2011.

MATTELART, A. **A origem do mito da Internet**. Disponível em:
<http://amaivos.uol.com.br/amaivos09/noticia/noticia.asp?cod_noticia=3648&cod_canal=37>
. Acesso em: jun.2011.

MENDES, R. M.; SOUZA, V. I.; CAREGNATO, S. E. **Propriedade intelectual na elaboração de objetos de aprendizagem**. Disponível em:
<<http://wiki.sj.ifsc.edu.br/wiki/images/7/7d/Propriedintelec.pdf>>. Acesso em 29.maio.2012.

MINISTÉRIO DA EDUCAÇÃO. **Banco internacional de objetos educacionais**. Disponível em: <<http://objetoseducacionais2.mec.gov.br/handle/mec/26/search>>. Acesso em: ago.2011.

_____. **Rede Interativa Virtual de Educação (RIVED)**. 2007. Disponível em:
<<http://rived.mec.gov.br/projeto.php>>. Acesso em: 29.maio.2012.

MONTEIRO, B. S. et al. **Metodologia de desenvolvimento de objetos de aprendizagem com foco na aprendizagem significativa**. 2006. Disponível em
<http://www.fisica.ufpb.br/~romero/pdf/2006_XVIISBIE.pdf>. Acesso em: 29.maio.2012.

MORAES, M. J. **Aplicação de recursos de ambiente virtual de aprendizagem em curso de biologia do ensino médio**. 2011. 125 f. Dissertação (Mestrado em Ciências) – Faculdade de Educação, Instituto de Física, Instituto de Química e Instituto de Biociências, Universidade de São Paulo, São Paulo. 2011.

MOREIRA, C. A. O. **Ambiente virtual interativo no ensino de ciências: uma abordagem sociocultural**. 2009. 152 f. Dissertação (Mestrado) – Faculdade de Educação, Instituto de Física, Instituto de Química e Instituto de Biociências, Universidade de São Paulo, São Paulo. 2009.

MURAMATSU, M. (coord). **FGE 160 – ótica**. São Paulo: Instituto de Física, 2008.

MUZIO, J.; HEINS, T.; MUNDELL, R. **Experiences with reusable elearning objects: from theory to practice**. Victoria, Canadá. 2001. Disponível em: <<http://www.udutu.com/pdfs/eLearning-objects.pdf>>. Acesso em: may.2012.

NEGROPONTE, N. **A vida digital**. São Paulo: Companhia das Letras, 1995.

NETO, A. T. **Uma abordagem para projeto de aplicações com interação multimodal na web**. 2011. 215 f. Tese (Doutorado) - Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo, São Carlos, 2011.

NILSEN, J. **Projetando websites**. São Paulo: Editora Campus, 2000.

NILSEN, J.; LORANGER, H. **Usabilidade na web**. Rio de Janeiro: Elsevier Editora, 2007.

NILSEN, J.; TAHIR, M. **Homepage usabilidade: 50 websites desconstruídos**. São Paulo: Editora Campus, 2002.

NUNES, E. R. **Ensino de conceitos físicos no ensino médio e as contribuições dos objetos de aprendizagem**. 2011. 302 f. Tese (Doutorado). Faculdade de Educação da Universidade de São Paulo, São Paulo. 2011.

OLIVEIRA, A. B. **O software modellus e sua possibilidade para desafiar as concepções de senso comum em ótica**. 2009. 169 f. Dissertação (Mestrado) – Faculdade de Educação, Instituto de Física, Instituto de Química e Instituto de Biociências. Universidade de São Paulo, São Paulo. 2009.

OLIVEIRA, E. W. **Segmentação de objetos de aprendizagem e abordagens para sua utilização**. 2009. 131 f. Dissertação (Mestrado) - Departamento de Informática Aplicada, UNIRIO, Rio de Janeiro. 2009.

OLIVEIRA, M. K. **Pensar a educação: contribuições de Vigotski**. In.: LERNER, D. et al. Piaget-Vigotski: novas contribuições para o debate. São Paulo: Ática, 2000.

PABLOS, J. **A visão disciplinar no espaço das tecnologias da informação e comunicação**. In: SANCHO, J. M.; HERNÁNDEZ, F. Tecnologias para transformar a educação. Porto Alegre: Artmed, 2006.

PALLOFF, R. M.; PRATT, K. **Construindo comunidades de aprendizagem no ciberespaço**. Porto Alegre: Artmed, 2002.

PAPERT, S. **A máquina das crianças**. Porto Alegre: Artmed, 2004.

PERISSE, G. **Filosofia, ética e literatura**. São Paulo: Manole, 2003.

PESSOA Junior, O. **Conceitos de física quântica**. Volume 1. São Paulo: Editora Livraria da Física, 2006.

PETERS, O. **Um modelo pedagógico para a utilização de espaços virtuais de aprendizagem**. In: A educação a distância em transição. São Leopoldo: Editora Unisinos, 2004.

_____. **Novos espaços de aprendizagem**. In: A educação a distância em transição. São Leopoldo: Editora Unisinos, 2004.

PIMENTA, P.; BAPTISTA, A. A. **Das plataformas de e-learning aos objetos de aprendizagem**. In: Dias, A. A. S.; Gomes, M. J. (coord.). E-learning para e-formadores. TecMinho, 2004. Disponível em: <<http://repositorium.sdum.uminho.pt/bitstream/1822/8723/3/dos%20lms%20aos%20objectos.pdf>>. Acesso em: maio.2012.

PREECE, J.; ROGERS, Y.; SHARP, H. **Design de interação: além da informação homem-computador**. Porto Alegre : Bookman, 2005.

PRIMO, A. **Interação mútua e reativa: uma proposta de estudo**. Revista da Famecos, n. 12, p. 81-92, jun. 2000. Disponível em: <http://www6.ufrgs.br/limc/PDFs/int_mutua_reativa.pdf>. Acesso em: jun.2011.

REIS, A. A. **Flash, Usabilidade e Web 2.0**. Centro Universitário UMA, Belo Horizonte, Minas Gerais. 2009. Disponível em: <http://www.aareis.com.br/arquivos_posts/flash_usabilidade_web2.pdf>. Acesso em: maio.2012.

RIVAS, T. **Objetos de aprendizagem no contexto das comunidades virtuais auto-organizadas para a produção de software livre e de código aberto**. 2009. 327 f. Tese (Doutorado) - Escola de Engenharia de São Carlos, Universidade de São Paulo, São Carlos. 2009.

ROCHA, H. V.; BARANAUSKAS, M. C. C. **Design e avaliação de interfaces humano-computador**. Campinas: Nied/Unicamp, 2003.

ROMANI, B. Adeus, Flash. **Folha de São Paulo**, São Paulo, 9.jul.2012. Tec. Disponível em: <<http://www1.folha.uol.com.br/fsp/tec/53441-adeus-flash.shtml>>. Acesso em: 20.jul.2012.

SANTAELLA, L. **Navegar no ciberespaço: o perfil cognitivo do leitor imersivo**. São Paulo: Paulus, 2004.

SALVE, G. B. **Modelo de planejamento para repositório de objetos de aprendizagem em organizações educacionais**. 2010. 226f. Tese (Doutorado) - Escola de Engenharia de São Carlos, Universidade de São Paulo, São Carlos. 2010.

SANTAELLA, L.; LEMOS, R. **Redes sociais digitais**. São Paulo: Paulus Editora, 2010.

SCHLEMMER, E. **Ambiente virtual de aprendizagem (AVA): uma proposta para a sociedade em rede na cultura da aprendizagem**. In: Valentini, C. B.; Schlemmer, E. (org.). *Aprendizagem em ambientes virtuais: compartilhando ideias e construindo cenários*. Caxias do Sul, RS: Educs, 2005.

SESC. **Sesc santo amaro em rede**. Disponível em: <<http://www.mostrasescdeartes.com.br/stoamaroemrede>>. Acesso em: jun.2011.

SEVERINO, E. Z. G. **Recursos virtuais em aulas de laboratório de física**. 2006. 155 f. Dissertação (Mestrado) – Faculdade de Educação, Instituto de Física, Instituto de Química e Instituto de Biociências. Universidade de São Paulo, São Paulo. 2006.

SILVA, A. S. **Design e arquitetura de informação para web sites educacionais: um estudo de usabilidade**. 2007. 170 f. Tese (Doutorado) – Faculdade de Arquitetura e Urbanismo, Universidade de São Paulo. São Paulo. 2007.

SILVA, J. G. **Desenvolvimento de um ambiente virtual para estudo sobre representação estrutural em Química**. 2007. 172 f. Dissertação (Mestrado) – Faculdade de Educação, Instituto de Física, Instituto de Química e Instituto de Biociências, Universidade de São Paulo, São Paulo. 2007.

SOUZA, A. C. S. **Objetos de Aprendizagem Colaborativos**. Data de envio: maio.2005. Disponível em: <<http://www.abed.org.br/congresso2005/por/pdf/024tcc4.pdf>>. Acesso em: maio.2012.

SUCHMAN, L. ;TRIGG, R. H. **Understanding practice**: video as a medium for reflection and design. In: Human-computer interaction: toward the year 2000. Baecker , R. et al. (eds). Morgan Kaufmann Publishers, Inc. 1995.

TAROUCO, L. M. R.; FABRE, M. C. J. M.; TAMUSIUNAS, F. R. **Reusabilidade de objetos educacionais**. Novas Tecnologias na Educação CINTED, UFRGS, v. 1, n. 1, fev. 2003. Disponível em: <http://www.nuted.ufrgs.br/oficinas/criacao/marie_reusabilidade.pdf>. Acesso em: maio.2012.

TOFOLI, M. R. **Utilização e compreensão do computador**: um olhar no dia-a-dia do professor. 2003. 132 f. Dissertação (Mestrado em Ciências) – Faculdade de Educação, Instituto de Física, Instituto de Química e Instituto de Biociências, Universidade de São Paulo, São Paulo. 2003.

TUBBS, N. **The Concept of Teachability**. In: Educational Theory, University of Illinois, v. 53, n. 1, p. 75-90, winter 2003. Disponível em: <<http://onlinelibrary.wiley.com/doi/10.1111/j.1741-5446.2003.00075.x/full>>. Acesso em: sep. 2012.

VAZ, F. R. V. **Modelagem e arquitetura de sistemas para monitoração e acompanhamento da aprendizagem eletrônica**. 2007. 164 f. Tese (Doutorado) - Escola Politécnica da Universidade de São Paulo, São Paulo. 2007.

VIGOSTSKI, L. S. **A formação social da mente**. São Paulo: Martins Fontes, 1989.

_____. **A imaginação e seu desenvolvimento na infância**. In: O desenvolvimento psicológico na infância. São Paulo: Ed. 34, 2009.

_____. **Pensamento e linguagem**. São Paulo: Martins Fontes, 1987.

VINHA, M. L. V. **Criatividade em ação**: roteiros de animações virtuais elaborados por alunos de ensino médio em físico. 2007. 258 f. Tese (Doutorado) – Faculdade de Educação, Instituto de Física, Instituto de Química e Instituto de Biociências, Universidade de São Paulo, São Paulo. 2007.

VYGOTSKY, L.; LURIA, A. R. **Linguagem, desenvolvimento e aprendizagem**. São Paulo, Icone, 1988.

WILEY, D. A. **Conecting learning objects to instructional theory**: A definition, a methaphor and a taxonomy. 2000. Disponível em: <<http://www.reusability.org/read/chapters/wiley.doc>>. Acesso em: may.2012.

ZILIO, S. C. **Óptica moderna, fundamentos e aplicações**. São Carlos: Fotônica - Instituto de Física de São Carlos – USP. Disponível em: <<http://www.fotonica.if.sc.usp.br/ebook/e-book1.php>>. Acess em: out.2010.

GLOSSÁRIO

AVI	Formato de arquivo de vídeo que significa <i>Audio Video Interleave</i> (http://pt.wikipedia.org).
<i>Browser</i>	Navegador de páginas da Internet.
Cliente	Computadores que acessam os serviços de um servidor (http://pt.wikipedia.org).
CMYK	Abreviatura do sistema de cores subtrativas formado pelo Ciano (<i>Cyan</i>), Magenta (<i>Magenta</i>), Amarelo (<i>Yellow</i>) e Preto (denominado como <i>Key</i>).
CRT	Tubo de raios catódicos (do inglês, <i>cathode ray tube</i>). É o elemento eletrônico presente nos antigos aparelhos de televisão e monitores de computadores que produziam a imagem em uma tela mediante a emissão de feixes de elétrons.
Dobra de página	Referência dada às informações visualizadas na tela do computador, à semelhança do conteúdo apresentado acima ou abaixo da dobra de um jornal impresso (NIELSEN, 2002, p. 23).
DOC	Extensão de nome de arquivos de editor de texto (http://pt.wikipedia.org).
Fonte aberta	Do inglês <i>open source</i> , é um conjunto de princípios e práticas sobre como escrever um <i>software</i> . Significa que o código fonte do <i>software</i> está disponível aos interessados, tanto programadores como para o público em geral, possuindo restrições intelectuais de cópia e reprodução reduzidas ou inexistentes quanto aos valores referentes à propriedade intelectual ou ao direito autoral (http://pt.wikipedia.org).
GIF	Extensão de arquivo digital que significa <i>Graphics Interchange Format</i> . É um formato de imagem de mapa de bits muito usado na <i>world wide web</i> , quer para imagens fixas, quer para animações (http://pt.wikipedia.org).

<i>Google Analytics</i>	Ferramenta do <i>Google</i> que realiza a análise de audiência de páginas da internet (http://www.google.com/analytics).
Hexadecimal	Código numérico representado por 16 algarismos. Nos sistemas computacionais esses algarismos são os seguintes: 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F (http://pt.wikipedia.org).
Interface	Superfície de contato que reflete as propriedades físicas das partes que interagem, as funções a serem executadas e o balanço entre poder e controle (Laurel, 1993 apud Rocha e Baranauskas, 2003).
IP	Do inglês <i>Internet Protocol</i> , é um rótulo numérico que identifica cada equipamento que participa de uma rede de computadores e que usa o protocolo de internet para se comunicar (http://en.m.wikipedia.org).
Iteração	Repetição de ciclos de operação dentro de uma cadeia de processos.
JPEG	(ou JPG). Extensão de arquivo digital que significa <i>Joint Photographic Experts Group</i> . Refere-se a um método usado para comprimir imagens fotográficas criado em 1983 (http://pt.wikipedia.org).
Linguagem de programação	Método padronizado para comunicar instruções para um computador (http://pt.wikipedia.org).
<i>Log</i>	Registro de informações coletadas remotamente.
Metadados	Informações sobre um objeto, seja ele físico ou digital, que aumentam a possibilidade para encontrá-los, gerenciá-los e usá-los, a partir de um mecanismo de busca ou de um repositório (IEEE, 2002).
MIDI	Formato de áudio digital que significa <i>Musical Instrument Digital Interface</i> (http://pt.wikipedia.org).
MP3	Extensão de arquivo digital que significa <i>Moving Picture Experts Group Audio Layer 3</i> . Trata-se de uma codificação de imagens em movimento e áudio (http://pt.wikipedia.org).
MySQL	Sistema de gerenciamento de banco de dados que utiliza a linguagem SQL (Linguagem de Consulta Estruturada, do inglês <i>Structured Query Language</i>) como interface (http://pt.wikipedia.org).

Observação indireta

Coleta de dados a partir de rastreamento de informações, que pode ser obtido a partir de dados registrados em diários ou então a partir de *Logs* das interações (Preece, Rogers e Sharp, 2005, p. 396).

Observação indireta

Registros gravados (em vídeo, por exemplo) para uma posterior análise (Rocha e Baranauskas 2003, p. 144).

Pixel

Menor elemento de imagem digital.

PPT

Formato nativo do programa de apresentação *PowerPoint* da *Microsoft* (<http://pt.wikipedia.org>).

Requisito

Declaração sobre um *software* que se deseja produzir, que especifique o que ele deveria fazer ou como deveria operar (Rocha e Baranauskas, 2003).

RGB

Abreviatura do sistema de cores aditivas formado pelo Vermelho (*Red*), Verde (*Green*) e Azul (*Blue*).

Servidor

Sistema de computação que fornece serviços a uma rede de computadores. Esses serviços podem ser o envio de dados de uma página da Internet ou o do correio eletrônico (<http://pt.wikipedia.org>).

Software proprietário

Software não livre; é aquele cuja cópia, redistribuição ou modificação são em alguma medida restritos pelo seu criador ou distribuidor (<http://pt.wikipedia.org>).

Stop Motion

Forma de animação em que as cenas são fotografadas quadro a quadro, podendo ser constituídas por desenhos, personagens de massa de modelar, recorte de papel, etc.

SWF

O *Shockwave Flash* é um formato de arquivo de animações interativas criado pela *Macromedia*.

WMA

Extensão de arquivo digital que significa *Windows Media Audio*. Formato de áudio produzido pela *Microsoft* (<http://pt.wikipedia.org>).

APÊNDICE A

Animações do Laboratório Virtual selecionadas para integrar o Banco Internacional de Recursos Multimídia do MEC em 2008.

1. Trenzinho da Estação (1999)
Animação: Marcos Matsukuma
2. Motorzinho elétrico (1999)
Animação: Marcos Matsukuma
3. Tangram (2001)
Programação: Marcos Matsukuma
4. Dilatação (2001)
Animação: Fernando Goldenstein Carvalhaes / Ilustração: Eliano Silva Marques /
Coordenação: Marcos Matsukuma
5. Arquimedes (2002)
Animação: Duilio Gentili / Ilustração: Eliano Silva Marques /
Coordenação: Marcos Matsukuma
6. Pêndulo (2002)
Programação: Desirée Della Mônica Ferreira / Coordenação: Marcos Matsukuma
7. Máquina de Heron (2002)
Animação: Desirée Della Mônica Ferreira / Ilustração: Eliano Silva Marques /
Coordenação: Marcos Matsukuma
8. Chuvas (2002)
Animação: Francisco Bryan / Ilustração: Eliano Silva Marques /
Coordenação: Marcos Matsukuma
9. Interior da Terra (2002)
Animação: Desirée Della Mônica Ferreira / Ilustração: Eliano Silva Marques /
Coordenação: Marcos Matsukuma
10. Relevo do Brasil (2002)
Animação: Francisco Bryan / Coordenação: Marcos Matsukuma
11. Cortinas do tempo (2002)
Programação: Marcos Matsukuma /
Texto: Francisco Bryan, Isabela Cunha Gonçalves, Rodrigo da Silva /
Fotos: Renato Cury / Coordenação: Marcos Matsukuma
12. Sistema solar (2005)
Programação: Marcos Matsukuma / Pesquisa: Fernando Camelier
Locução: Tereza Marinho / Apoio técnico: Francisco Bryan, Juliana Parollo /
Coordenação: Marcos Matsukuma

APÊNDICE B

Principais trechos de código *action script* das animações do Laboratório Virtual

1. Reflexão

```

1  superficie._x = centro._x;
2  superficie._y = centro._y;
3
4  objeto.swapDepths(10);
5
6  objeto.onPress = function() {
7      startDrag(this, false, 0, 0, observador._x-50, centro._y-50);
8      rotacionarDispositivo();
9  };
10 objeto.onRelease = function() {
11     stopDrag();
12     pararDispositivo();
13 };
14 observador.onPress = function() {
15     startDrag(this, false, objeto._x+50, 0, 800, centro._y-50);
16     rotacionarDispositivo();
17 };
18 observador.onRelease = function() {
19     stopDrag();
20     pararDispositivo();
21 };
22
23 rotacionarDispositivo();
24
25 function rotacionarDispositivo() {
26     this.onEnterFrame = function() {
27         //trace("ok");
28         attachMovie("objeto","imagem",8);
29         imagem._alpha = 50;
30         imagem._x = objeto._x;
31         imagem._y = objeto._y+2*(centro._y-objeto._y);
32
33         dX = observador._x-imagem._x;
34         dY = observador._y-imagem._y;
35         sinalX = dX/Math.abs(dX);
36         sinalY = dY/Math.abs(dY);
37         angulo = 180*Math.atan(dX/dY)/Math.PI;
38         if (sinalY>0) {
39             angulo = angulo-180;
40         }
41         observador._rotation = 90-angulo;
42
43         // posicao do eixo de reflexao
44         centro._x = objeto._x-(centro._y-objeto._y)*dX/dY;
45
46         // feixe emergente
47         this.createEmptyMovieClip("feixe1",2);
48         feixe1.lineStyle(2,0xFF0000,100);
49         feixe1.moveTo(objeto._x,objeto._y);
50         feixe1.lineTo(centro._x,centro._y);
51         //feixe refletido
52         this.createEmptyMovieClip("feixe2",3);
53         feixe2.lineStyle(2,0xFF0000,100);
54         feixe2.moveTo(centro._x,centro._y);
55         feixe2.lineTo(observador._x,observador._y);
56         //feixe refletido
57         this.createEmptyMovieClip("feixe3",4);
58         feixe3.lineStyle(2,0xFF0000,50);
59         feixe3.moveTo(imagem._x,imagem._y);
60         feixe3.lineTo(centro._x,centro._y);
61         // eixo perpendicular
62         this.createEmptyMovieClip("eixo",5);
63         eixo.lineStyle(1,0x666666,100);
64         eixo.moveTo(centro._x,100);
65         eixo.lineTo(centro._x,centro._y);
66
67         legendaObjeto._x = objeto._x;
68         legendaObjeto._y = objeto._y;
69         legendaImagem._x = imagem._x;

```

Reprodução da parte principal do *script* de programação da animação Reflexão. Cálculo para posicionamento dos elementos na interface segundo diagrama de formação de imagem (script 1/2).

```

70     legendaImagem._y = imagem._y;
71     legendaObservador._x = observador._x;
72     legendaObservador._y = observador._y;
73
74     seta1._rotation = 90+angulo;
75     seta1._x = objeto._x+(centro._x-objeto._x)/2;
76     seta1._y = objeto._y+(centro._y-objeto._y)/2;
77
78     seta2._rotation = 270-angulo;
79     seta2._x = centro._x+(observador._x-centro._x)/2;
80     seta2._y = centro._y+(observador._y-centro._y)/2;
81
82     seta3._rotation = 270-angulo;
83     seta3._x = imagem._x+(centro._x-imagem._x)/2;
84     seta3._y = imagem._y+(centro._y-imagem._y)/2;
85 };
86 }
87 function pararDispositivo() {
88     delete this.onEnterFrame;
89 }

```

Reprodução da parte principal do *script* de programação da animação Reflexão. Cálculo para posicionamento dos elementos na interface segundo diagrama de formação de imagem (script 2/2).

2. Refração

```

1  superficie._x = centro._x;
2  superficie._y = centro._y;
3
4  objeto.swapDepths(10);
5
6  objeto.onPress = function() {
7      startDrag(this, false, 100, centro._y+50, observador._x-150, 500);
8      rotacionarDispositivo();
9  };
10 objeto.onRelease = function() {
11     stopDrag();
12     pararDispositivo();
13 };
14 objeto.onReleaseOutside = function() {
15     stopDrag();
16     pararDispositivo();
17 };
18 observador.onPress = function() {
19     startDrag(this, false, objeto._x+150, 50, 600, centro._y-20);
20     rotacionarDispositivo();
21 };
22 observador.onRelease = function() {
23     stopDrag();
24     pararDispositivo();
25 };
26 observador.onReleaseOutside = function() {
27     stopDrag();
28     pararDispositivo();
29 };
30
31 rotacionarDispositivo();
32
33 function rotacionarDispositivo() {
34     this.onEnterFrame = function() {
35
36         indice1 = botao1.indice;
37         indice2 = botao2.indice;
38         //trace(indice1 + " ... " + indice2)
39         attachMovie("objeto", "imagem", 8);
40         imagem._alpha = 50;
41         imagem._x = objeto._x;
42         imagem._y = centro._y - (centro._x - objeto._x) * (centro._y - observador._y) / (centro._x - observador._x);
43
44         // rotacoes dos objetos
45         dX = observador._x - imagem._x;
46         dY = observador._y - imagem._y;
47         sinalX = dX/Math.abs(dX);
48         sinalY = dY/Math.abs(dY);
49         angulo = 180*Math.atan(dX/dY)/Math.PI;
50         if (sinalY>0) {
51             angulo = angulo-180;
52         }
53         observador._rotation = 90-angulo;
54
55         dX2 = objeto._x - centro._x;
56         dY2 = objeto._y - centro._y;
57         sinalX2 = dX2/Math.abs(dX2);
58         sinalY2 = dY2/Math.abs(dY2);
59         angulo2 = 180*Math.atan(dX2/dY2)/Math.PI;
60         if (sinalY2>0) {
61             angulo2 = angulo2-180;
62         }
63         // posicao do eixo de reflexao
64         //centro._x = objeto._x - (centro._y - objeto._y) * dX/dY;
65
66         baseI = Math.pow((centro._x - objeto._x), 2) + Math.pow((centro._y - objeto._y), 2);
67         HipI = Math.pow(baseI, 0.5);
68         xI = Math.abs(centro._x - objeto._x);
69         sinI = xI/HipI;

```

Reprodução da parte principal do *script* de programação da animação Refração. Cálculo para posicionamento dos elementos na interface segundo diagrama de formação de imagem (script 1/2).

```

70
71 baseR = Math.pow((centro._x-observador._x), 2)+Math.pow((centro._y-observador._y), 2);
72 HipR = Math.pow(baseR, 0.5);
73 xR = Math.abs(centro._x-observador._x);
74 sinR = xR/HipR;
75
76 diferencaIndice = indice1*sinI-indice2*sinR;
77 //trace(">>> ... "+diferencaIndice);
78 if (Math.abs(diferencaIndice)<0.0001) {
79     //delete this.onEnterFrame;
80 }
81 centro._x += -50*diferencaIndice;
82 // feixe emergente
83 this.createEmptyMovieClip("feixe1",2);
84 feixe1.lineStyle(2,0xFF0000,100);
85 feixe1.moveTo(objeto._x,objeto._y);
86 feixe1.lineTo(centro._x,centro._y);
87 //feixe refletido
88 this.createEmptyMovieClip("feixe2",3);
89 feixe2.lineStyle(2,0xFF0000,100);
90 feixe2.moveTo(centro._x,centro._y);
91 feixe2.lineTo(observador._x,observador._y);
92 //feixe refletido
93 this.createEmptyMovieClip("feixe3",4);
94 feixe3.lineStyle(2,0xFF0000,50);
95 feixe3.moveTo(imagem._x,imagem._y);
96 feixe3.lineTo(centro._x,centro._y);
97 // eixo perpendicular
98 //this.createEmptyMovieClip("eixo",5);
99 eixo.lineStyle(1,0x666666,100);
100 eixo.moveTo(centro._x,100);
101 eixo.lineTo(centro._x,centro._y);
102
103 legendaObjeto._x = objeto._x;
104 legendaObjeto._y = objeto._y;
105 legendaImagem._x = imagem._x;
106 legendaImagem._y = imagem._y;
107 legendaObservador._x = observador._x;
108 legendaObservador._y = observador._y;
109
110 seta1._rotation = 90-angulo2;
111 seta1._x = objeto._x+(centro._x-objeto._x)/2;
112 seta1._y = objeto._y+(centro._y-objeto._y)/2;
113
114 seta2._rotation = 270-angulo;
115 seta2._x = centro._x+(observador._x-centro._x)/2;
116 seta2._y = centro._y+(observador._y-centro._y)/2;
117
118 seta3._rotation = 270-angulo;
119 seta3._x = imagem._x+(centro._x-imagem._x)/2;
120 seta3._y = imagem._y+(centro._y-imagem._y)/2;
121 };
122 }
123 function pararDispositivo() {
124     //delete this.onEnterFrame;
125 }

```

Reprodução da parte principal do *script* de programação da animação Refração. Cálculo para posicionamento dos elementos na interface segundo diagrama de formação de imagem (script 2/2).

3. Lentes (como se formam as imagens com as lentes)

```

1  this.onEnterFrame = function() {
2      H_obj = objeto._height;
3      D_obj = Math.abs(lenteConv._x-objeto._x);
4      f = Math.abs(lenteConv._x-foco._x);
5      mult = f/(D_obj-f);
6      h_img = H_obj*mult;
7      d_img = D_obj*mult;
8      posicao._x = lenteConv._x+d_img;
9      posicao._y = lenteConv._y+h_img;
10
11      ampliacao = 100*h_img/H_obj;
12
13      imagem._xscale = ampliacao;
14      imagem._yscale = -ampliacao;
15      imagem._x = posicao._x;
16
17      // desenhar linhas
18      this.createEmptyMovieClip("linha1",1);
19      linha1.lineStyle(1,0xFF6666,100);
20      linha1.moveTo(objeto._x,objeto._y-objeto._height);
21      linha1.lineTo(lenteConv._x,objeto._y-objeto._height);
22
23      this.createEmptyMovieClip("linha2",2);
24      linha2.lineStyle(1,0xFF6666,100);
25      linha2.moveTo(lenteConv._x,objeto._y-objeto._height);
26      linha2.lineTo(posicao._x,posicao._y);
27
28      this.createEmptyMovieClip("linha3",3);
29      linha3.lineStyle(1,0xFF6666,100);
30      linha3.moveTo(objeto._x,objeto._y-objeto._height);
31      linha3.lineTo(posicao._x,posicao._y);
32
33      this.createEmptyMovieClip("linha4",4);
34      linha4.lineStyle(1,0xFF6666,100);
35      linha4.moveTo(lenteConv._x,posicao._y);
36      linha4.lineTo(imagem._x,posicao._y);
37
38      this.createEmptyMovieClip("linha5",5);
39      linha5.lineStyle(1,0xFF6666,100);
40      linha5.moveTo(lenteConv._x,posicao._y-5);
41      linha5.lineTo(lenteConv._x,posicao._y+5);
42
43      this.createEmptyMovieClip("linha6",6);
44      linha6.lineStyle(1,0xFF6666,50);
45      linha6.moveTo(lenteConv._x,objeto._y-objeto._height);
46      linha6.lineTo(2*foco._x-lenteConv._x,objeto._y+objeto._height);
47
48      // posicao das letras
49      d_obj._x = objeto._x+(lenteConv._x-objeto._x)/2;
50      d_obj._y = objeto._y-objeto._height-10;
51      //
52      p_foco._x = foco._x;
53      p_foco._y = foco._y-20;
54      //
55      h_obj._x = objeto._x-20;
56      h_obj._y = objeto._y-objeto._height/2;
57      //
58      h_imgm._x = imagem._x+20;
59      h_imgm._y = foco._y-(foco._y-posicao._y)/2;
60      //
61      d_imgm._x = imagem._x+(lenteConv._x-imagem._x)/2;
62      d_imgm._y = posicao._y+10;
63  };

```

Reprodução da parte principal do *script* de programação da animação Lentes. Cálculo para posicionamento dos elementos na interface segundo diagrama de formação de imagem.

4. Espelhos esféricos (como se formam as imagens com espelhos)

```

1  this.onEnterFrame = function() {
2      // posicoes geometricas
3      // raio da esfera
4      raioEsfera = Math.abs(espelhoConv._x-centro._x);
5      focoValor = raioEsfera/2;
6      // posicao do foco
7      foco._y = centro._y;
8      foco._x = centro._x-raioEsfera/2;
9      // inflexão no espelho
10     posicaoFeixeEspelho._x = centro._x-Math.pow(Math.pow(raioEsfera, 2)-Math.pow(alturaObjeto, 2), 0.5);
11     posicaoFeixeEspelho._y = alturaObjeto;
12
13     // calculo de interceccão dos feixes
14     // parametro a:
15     a = focoValor;
16     // parametro b:
17     b = centro._x-posicaoFeixeEspelho._x-focoValor;
18     // parametro L:
19     L = centro._x-objeto._x;
20     // parametro c:
21     c = a/(1-(b/L));
22     // parametro de altura da imagem:
23     h_img = c*objeto._height/L;
24     // posicao da imagem em x:
25     posicao._x = centro._x-c;
26     // posicao da imagem em y:
27     posicao._y = centro._y-h_img;
28
29
30     H_obj = objeto._height;
31     //D_obj = Math.abs(espelhoConv._x-objeto._x);
32     //f = Math.abs(espelhoConv._x-centro._x);
33     //mult = f/(D_obj-f);
34     //h_img = H_obj*mult;
35     //d_img = D_obj*mult;
36
37     ampliacao = 100*h_img/H_obj;
38
39     imagem._xscale = ampliacao;
40     imagem._yscale = ampliacao;
41     imagem._x = posicao._x;
42
43     // desenhar linhas
44
45     alturaObjeto = objeto._y-objeto._height;
46     this.createEmptyMovieClip("linha1",1);
47     linha1.lineStyle(1,0xFF6666,100);
48     linha1.moveTo(objeto._x,alturaObjeto);
49     linha1.lineTo(posicaoFeixeEspelho._x,alturaObjeto);
50
51     this.createEmptyMovieClip("linha2",2);
52     linha2.lineStyle(1,0xFF6666,100);
53     linha2.moveTo(posicaoFeixeEspelho._x,alturaObjeto);
54     linha2.lineTo(foco._x,foco._y);
55
56     this.createEmptyMovieClip("linha3",3);
57     linha3.lineStyle(1,0xFF6666,100);
58     linha3.moveTo(objeto._x,alturaObjeto);
59     linha3.lineTo(centro._x,centro._y);
60
61     d_obj._x = objeto._x+(espelhoConv._x-objeto._x)/2;
62     d_obj._y = alturaObjeto-10;
63     //
64     p_centro._x = centro._x;
65     p_centro._y = centro._y-20;
66     //
67     h_obj._x = objeto._x-20;
68     h_obj._y = alturaObjeto/2;
69     //

```

Reprodução da parte principal do *script* de programação da animação Espelhos Esféricos. Cálculo para posicionamento dos elementos na interface segundo diagrama de formação de imagem (script 1/2).

```

70 h_imgm._x = imagem._x+20;
71 h_imgm._y = centro._y-(centro._y-posicao._y)/2;
72 //
73 d_imgm._x = imagem._x+(espelhoConv._x-imagem._x)/2;
74 d_imgm._y = posicao._y+10;
75 //
76 p_foco._x = foco._x;
77 p_foco._y = foco._y+20;
78 //
79 p_centro._x = centro._x;
80 p_centro._y = centro._y+20;
81
82 esfera._width = 2*raioEsfera;
83 esfera._height = esfera._width;
84 esfera._x = centro._x;
85 esfera._y = centro._y;
86 //espelhoConv.setMask(esfera);
87 espelhoMask._x = espelhoConv._x;
88 espelhoMask._y = espelhoConv._y;
89 esfera.setMask(espelhoMask);
90
91 esfera2._width = esfera._width-10;
92 esfera2._height = esfera._height-10;
93 esfera2._x = centro._x;
94 esfera2._y = centro._y;
95 espelhoMask2._x = espelhoConv._x;
96 espelhoMask2._y = espelhoConv._y;
97 //espelhoConv2._x = espelhoConv._x;
98 //espelhoConv2._y = espelhoConv._y;
99 //espelhoConv2.setMask(esfera2);
100 esfera2.setMask(espelhoMask2);
101 };

```

Reprodução da parte principal do *script* de programação da animação Espelhos Esféricos. Cálculo para posicionamento dos elementos na interface segundo diagrama de formação de imagem (script 2/2).

5. O olho (formação da imagem no olho e defeitos da visão)

Não possui nenhum *script* de programação relevante.

6. Olho normal (formação da imagem no olho)

A parte principal do *script* de programação dessa animação é semelhante à da Lente (animação 3).

7. Miopia

A parte principal do *script* de programação dessa animação é semelhante à da Lente (animação 3).

8. Hipermetropia

A parte principal do *script* de programação dessa animação é semelhante à da Lente (animação 3).

9. Correção (correção de defeitos da visão)

```

1  onClipEvent (load) {
2      this.onPress = function() {
3          startDrag(this, true, 100, 150, 570, 500);
4          posicaoOculos();
5      };
6      this.onRelease = function() {
7          acao();
8      };
9      this.onReleaseOutside = function() {
10         acao();
11     };
12     this.onRollOver = function() {
13         if (!posicaoOculosMiopia and !posicaoOculosHiper) {
14             _parent.feixeConvergente._alpha = 100;
15         }
16     };
17     this.onRollOut = function() {
18         _parent.feixeConvergente._alpha = 0;
19     };
20     function acao() {
21         if (posicaoOculosMiopia) {
22             _x = _parent.oculosMiopia._x;
23             _y = _parent.oculosMiopia._y;
24             _parent.feixeMiopiaErro._alpha = 100;
25             enderecoURL = "http://www.ideiasnacaixa.com/pesquisa/pesquisaOpticaOlhoCorrecao.php?resposta=miopiaNao";
26             getUrl(enderecoURL, "upload");
27         }
28         if (posicaoOculosHiper) {
29             _x = _parent.oculosHiper._x;
30             _y = _parent.oculosHiper._y;
31             _parent.feixeHiperCorrecao._alpha = 100;
32             enderecoURL = "http://www.ideiasnacaixa.com/pesquisa/pesquisaOpticaOlhoCorrecao.php?resposta=hiperSim";
33             getUrl(enderecoURL, "upload");
34         }
35         delete this.onEnterFrame;
36         stopDrag();
37     }
38     function posicaoOculos() {
39         this.onEnterFrame = function() {
40             if (posicaoOculosMiopia or posicaoOculosHiper) {
41                 _parent.feixeConvergente._alpha = 0;
42             } else {
43                 _parent.feixeConvergente._alpha = 100;
44             }
45             if (this.hitTest(_parent.oculosMiopia)) {
46                 posicaoOculosMiopia = true;
47                 _parent.feixeMiopia._alpha = 0;
48             } else {
49                 posicaoOculosMiopia = false;
50                 if (!_parent.lenteDivergente.posicaoOculosMiopia) {
51                     _parent.feixeMiopia._alpha = 100;
52                 }
53                 _parent.feixeMiopiaErro._alpha = 0;
54             }
55             if (this.hitTest(_parent.oculosHiper)) {
56                 posicaoOculosHiper = true;
57                 _parent.feixeHiper._alpha = 0;
58             } else {
59                 posicaoOculosHiper = false;
60                 if (!_parent.lenteDivergente.posicaoOculosHiper) {
61                     _parent.feixeHiper._alpha = 100;
62                 }
63                 _parent.feixeHiperCorrecao._alpha = 0;
64             }
65         };
66     }
67 }

```

Reprodução da parte principal do *script* de programação da animação Correção de Defeitos da Visão. O método de Log de dados para envio da seleção do usuário pode ser identificado nas linhas em verde.

10. Cores

Cores - parte 1. Cores primárias de luz (RGB)

```

1 // criar cor
2 hexaRR = "00";
3 hexaGG = "00";
4 hexaBB = "00";
5 var my_color:Color = new Color("mistura");
6 my_color.setRGB(0x000000);
7
8 function misturarCor(nome, valor) {
9     if (nome == "RR") {
10         hexaRR = "00"+valor;
11         hexaRR = hexaRR.slice(-2);
12     } else {
13         if (nome == "GG") {
14             hexaGG = "00"+valor;
15             hexaGG = hexaGG.slice(-2);
16         } else {
17             if (nome == "BB") {
18                 hexaBB = "00"+valor;
19                 hexaBB = hexaBB.slice(-2);
20             }
21         }
22     }
23     resultadoHexa = hexaRR+hexaGG+hexaBB;
24     resultado = "0x"+hexaRR+hexaGG+hexaBB;
25     //trace(resultado);
26     var my_color:Color = new Color("mistura");
27     my_color.setRGB(resultado);
28 }

```

Reprodução da parte principal do *script* de programação da animação Cores / cores de luz. O programa soma as intensidades e calcula a cor resultante em RGB (vermelho, verde e azul).

Cores - parte 2. Cores primárias de pigmento (CMY)

```

1 // criar cor
2 hexaRR = "ff";
3 hexaGG = "ff";
4 hexaBB = "ff";
5 var my_color:Color = new Color("mistura");
6 my_color.setRGB(0xffffffff);
7
8 function misturarCor(nome, valor, decimal) {
9     if (nome == "RG") {
10         hexaBB = "00"+valor;
11         hexaBB = hexaBB.slice(-2);
12         valorRG = int(100-100*decimal/255)+"%";
13     } else {
14         if (nome == "GB") {
15             hexaRR = "00"+valor;
16             hexaRR = hexaRR.slice(-2);
17             valorGB = int(100-100*decimal/255)+"%";
18         } else {
19             if (nome == "RB") {
20                 hexaGG = "00"+valor;
21                 hexaGG = hexaGG.slice(-2);
22                 valorRB = int(100-100*decimal/255)+"%";
23             }
24         }
25     }
26     resultado = "0x"+hexaRR+hexaGG+hexaBB;
27     //trace(resultado);
28     var my_color:Color = new Color("mistura");
29     my_color.setRGB(resultado);
30 }

```

Reprodução da parte principal do *script* de programação da animação Cores / cores de pigmento. O programa soma as porcentagens e calcula a cor resultante em CMY (ciano, magenta e amarelo).

Cores - parte 3. Tarefa: quais as cores primárias que combinadas resultam na luz branca?

```

1  var my_color:Color = new Color("discoCor");
2  my_color.setRGB(cor);
3
4  x0 = _x;
5  y0 = _y;
6  this.onPress = function() {
7      this.swapDepths(1);
8      startDrag(this);
9      if (local) {
10         ultimoLocal = local;
11         //trace("ultimo > "+ultimoLocal);
12     }
13 };
14 this.onRelease = function() {
15     stopDrag();
16     encaixar = false;
17     if (this.hitTest(_parent.comp01)) {
18         xfinal = _parent.comp01._x;
19         yfinal = _parent.comp01._y;
20         encaixar = true;
21         local = "comp01";
22     }
23     if (this.hitTest(_parent.comp02)) {
24         xfinal = _parent.comp02._x;
25         yfinal = _parent.comp02._y;
26         encaixar = true;
27         local = "comp02";
28     }
29     if (this.hitTest(_parent.comp03)) {
30         xfinal = _parent.comp03._x;
31         yfinal = _parent.comp03._y;
32         encaixar = true;
33         local = "comp03";
34     }
35     if (ultimoLocal) {
36         eval("_parent."+ultimoLocal)._alpha = 100;
37     }
38     if (encaixar) {
39         if (eval("_parent."+local)._alpha == 100) {
40             _x = xfinal;
41             _y = yfinal;
42             eval("_parent."+local)._alpha = 50;
43             eval("_parent."+local).cor = _name;
44         } else {
45             _x = x0;
46             _y = y0;
47             local = undefined;
48             ultimoLocal = undefined;
49         }
50     } else {
51         _x = x0;
52         _y = y0;
53         eval("_parent."+local)._alpha = 100;
54         local = undefined;
55         ultimoLocal = undefined;
56     }
57     if ((_parent.comp01._alpha+_parent.comp02._alpha+_parent.comp03._alpha) == 150) {
58         //this._visible = false;
59         _parent.cor1 = _parent.comp01.cor;
60         _parent.cor2 = _parent.comp02.cor;
61         _parent.cor3 = _parent.comp03.cor;
62         _parent.play();
63     }
64 };

```

Reprodução da parte principal do *script* de programação da animação Cores / tarefa. O programa recebe as cores seleccionadas pelo usuário (“comp01”, “comp02” e “comp03”) e as prepara para envio para o banco de dados.

11. Construindo ondas (onda transversal)

Construindo ondas - parte 1

```

1 function criarobjeto(kk) {
2   matriz = eval("ponto"+kk);
3   attachMovie("objeto","objeto"+kk,kk);
4   setProperty("objeto"+kk, _x, matriz[0]+centro._x);
5   setProperty("objeto"+kk, _y, matriz[1]+centro._y);
6   set("objeto"+kk+".z1", matriz[2]);
7 }
8 //
9 function criarobjetoEstatico(kk) {
10  matriz = eval("ponto"+kk);
11  attachMovie("objetoEstatico","objeto"+kk,kk);
12  setProperty("objeto"+kk, _x, matriz[0]+centro._x);
13  setProperty("objeto"+kk, _y, matriz[1]+centro._y);
14  set("objeto"+kk+".z1", matriz[2]);
15 }
16 //
17 function moverobjeto() {
18   if (chave) {
19     totalvertices = vertice.length;
20     for (vtc=0; vtc<=totalvertices; vtc++) {
21       eval("objeto"+eval("vtc"+vtc)).play();
22     }
23   }
24 }
25 //
26 function basef(tet, phi, gam) {
27   singam = Math.sin(gam);
28   sinphi = Math.sin(phi);
29   sintet = Math.sin(tet);
30   cosgam = Math.cos(gam);
31   cosphi = Math.cos(phi);
32   costet = Math.cos(tet);
33   //
34   a = cosphi*costet;
35   b = cosphi*sintet;
36   c = sinphi;
37   d = cosgam*sinphi*costet-singam*sintet;
38   e = cosgam*sinphi*sintet+singam*costet;
39   f = -cosgam*cosphi;
40   g = -singam*sinphi*costet-cosgam*sintet;
41   h = -singam*sinphi*sintet+cosgam*costet;
42   i = singam*cosphi;
43 }

```

Reprodução da parte principal do *script* de programação da animação Construindo Ondas. Nessa primeira parte são criados os pontos que irão compor o desenho da onda. Há ainda um *script* destinado a observação da onda em 3D que não foi utilizado nesse programa (função “basef”).

Construindo ondas - parte 2

```

1  tet = -0.02+Math.PI/2;
2  gam = 0.04+Math.PI/2;
3  for (i=0; i<=iMax; i++) {
4      //px = 10*Math.cos(i/4);
5      px = 0;
6      //py = 10*Math.sin(i/4);
7      py = 0;
8      pz = i;
9      set("ponto"+i, Array(px, py, pz));
10     criarobjeto(i);
11 }
12 for (i=0; i<=iMax; i += 3) {
13     pz = i;
14     set("ponto"+(1000+i), Array(px, py, pz));
15     criarobjetoEstatico(1000+i);
16 }
17 //
18 ponto100 = Array(0, 0, 0);
19 ponto101 = Array(-15, -15, 0);
20 ponto102 = Array(-15, 15, 0);
21 ponto103 = Array(15, 15, 0);
22 ponto104 = Array(15, -15, 0);
23 criarobjetoEstatico(100);
24 criarobjetoEstatico(101);
25 criarobjetoEstatico(102);
26 criarobjetoEstatico(103);
27 criarobjetoEstatico(104);
28 //
29 ponto200 = Array(0, 0, 80);
30 ponto201 = Array(-15, -15, 80);
31 ponto202 = Array(-15, 15, 80);
32 ponto203 = Array(15, 15, 80);
33 ponto204 = Array(15, -15, 80);
34 criarobjetoEstatico(200);
35 criarobjetoEstatico(201);
36 criarobjetoEstatico(202);
37 criarobjetoEstatico(203);
38 criarobjetoEstatico(204);
39 //
40 this.onEnterFrame = function() {
41     t = getTimer()/1000;
42     // nova base f:
43     //gam+=0.01
44     phi = phiX-Math.PI/2;
45     basef(tet,phi,gam);
46
47     createEmptyMovieClip("poligono",20000);
48     // vertices do dodecaedro
49     for (i=0; i<iMax; i++) {
50         profundidade = 10000000+i;
51         tellTarget ("poligono") {
52             createEmptyMovieClip("forma"+_parent.profundidade,_parent.profundidade);
53         }
54         with (eval("poligono.forma"+profundidade)) {
55             lineStyle(3,0xff0000,100);
56             moveTo(eval("objeto"+i).xo+centro._x,eval("objeto"+i).yo+centro._y);
57            .lineTo(eval("objeto"+(i+1)).xo+centro._x,eval("objeto"+(i+1)).yo+centro._y);
58             //lineTo(eval("objeto"+(i+1)).xo+centro._x,eval("objeto"+(i+1)).yo+centro._y);
59         }
60     }
61     //
62     // campo eletrico
63     for (m=0; m<(iMax-1); m += 3) {
64         tellTarget ("poligono") {
65             createEmptyMovieClip("campoEletrico"+_parent.m,20000000+_parent.m);
66         }

```

Reprodução da parte principal do *script* de programação da animação Construindo Ondas. Nessa segunda parte os pontos são unidos por linhas para dar o traçado da onda. A variável tempo (t) é introduzida na programação para dar movimento à onda (script 1/2).

```

67     with (eval("poligono.campoElettrico"+m)) {
68         lineStyle(1,0xffffffff,30);
69         moveTo(eval("objeto"+m).xo+centro._x,eval("objeto"+m).yo+centro._y);
70         lineTo(eval("objeto"+(1000+m)).xo+centro._x,eval("objeto"+(1000+m)).yo+centro._y);
71     }
72 }
73 //
74 tellTarget ("poligono") {
75     createEmptyMovieClip("linha",5000);
76 }
77 with (poligono.linha) {
78     lineStyle(1,0xffffffff,100);
79     moveTo(objeto100.xo+centro._x,objeto100.yo+centro._y);
80     lineTo(objeto200.xo+centro._x,objeto200.yo+centro._y);
81 }
82 tellTarget ("poligono") {
83     createEmptyMovieClip("quadrado1",110001);
84 }
85 with (poligono.quadrado1) {
86     lineStyle(1,0xffffffff,100);
87     beginFill(0xffffffff,20);
88     moveTo(objeto101.xo+centro._x,objeto101.yo+centro._y);
89     lineTo(objeto102.xo+centro._x,objeto102.yo+centro._y);
90     lineTo(objeto103.xo+centro._x,objeto103.yo+centro._y);
91     lineTo(objeto104.xo+centro._x,objeto104.yo+centro._y);
92     lineTo(objeto101.xo+centro._x,objeto101.yo+centro._y);
93     endFill();
94 }
95 tellTarget ("poligono") {
96     createEmptyMovieClip("quadrado2",20000000);
97 }
98 with (poligono.quadrado2) {
99     lineStyle(1,0xffffffff,100);
100    beginFill(0xffffffff,40);
101    moveTo(objeto201.xo+centro._x,objeto201.yo+centro._y);
102    lineTo(objeto202.xo+centro._x,objeto202.yo+centro._y);
103    lineTo(objeto203.xo+centro._x,objeto203.yo+centro._y);
104    lineTo(objeto204.xo+centro._x,objeto204.yo+centro._y);
105    lineTo(objeto201.xo+centro._x,objeto201.yo+centro._y);
106    endFill();
107 }
108 };
109 stop();

```

Reprodução da parte principal do *script* de programação da animação Construindo Ondas. Nessa segunda parte os pontos são unidos por linhas para dar o traçado da onda. A variável tempo (t) é introduzida na programação para dar movimento à onda (script 2/2).

Construindo ondas - parte 3

```

1  stop();
2  this.onEnterFrame = function() {
3      //
4      AmpX = _parent.Amp*Math.cos(_parent.polarizacao); //-AmpY0*Math.sin(_parent.polarizacao);
5      AmpY = _parent.Amp*Math.sin(_parent.polarizacao); //+AmpY0*Math.cos(_parent.polarizacao);
6
7      k = _parent.k;
8      w = _parent.w;
9      dtX = _parent.dtX;
10     dtY = _parent.dtY;
11     t = _parent.t;
12     //
13     z = zi;
14     if (_parent.alterarZ) {
15         z = zi+_parent.zi0;
16     }
17     x0 = AmpX*Math.cos(k*zi-w*t+Math.PI+_parent.fase);
18     y0 = AmpY*Math.sin(k*zi-w*t+Math.PI+_parent.fase);
19
20     xi = x0*Math.cos(-_parent.rotacao)-y0*Math.sin(-_parent.rotacao);
21     if (_parent.filtro) {
22         yi = 0;
23     } else {
24         yi = x0*Math.sin(-_parent.rotacao)+y0*Math.cos(-_parent.rotacao);
25     }
26     if (_parent.torcaao<>0) {
27         xi = x0*Math.cos(-_parent.torcaao*(zi-_parent.z0))-y0*Math.sin(-_parent.torcaao*(zi-_parent.z0));
28         yi = x0*Math.sin(-_parent.torcaao*(zi-_parent.z0))+y0*Math.cos(-_parent.torcaao*(zi-_parent.z0));
29     }
30     //_parent.alvo.zi;
31     a = _parent.a;
32     b = _parent.b;
33     c = _parent.c;
34     d = _parent.d;
35     e = _parent.e;
36     f = _parent.f;
37     g = _parent.g;
38     h = _parent.h;
39     i = _parent.i;
40     novacoordenada(xi,yi,z,a,b,c,d,e,f,g,h,i);
41     projecao(xf,yf,zf);
42     //
43     l = 250*Math.abs(_parent.dz/(zf-_parent.zo));
44     raio = Math.sqrt(xf*xf+yf*yf+(zf-_parent.zo)*(zf-_parent.zo));
45     prof = 100000-int(100*raio);
46 };

```

Reprodução da parte principal do *script* de programação da animação Construindo Ondas. Nessa terceira parte os parâmetros da onda aparecem na programação, assim como as funções trigonométricas, que dão o seu aspecto característico.

12. Interferência de ondas

A parte principal do *script* de programação dessa interface é semelhante à da animação Construindo Ondas (animação 11).

APÊNDICE C

Código *action script* da interface de visualização da Evolução das Escolhas dos Grupos de Cores

```

1  //////////////////////////////////////////////////
2  // programa de leitura de dados //
3  //////////////////////////////////////////////////
4  stop();
5  //var listaVerbete:Array = new Array();
6  // formato do requisito:
7  // ?dataInicio=AAAA-MM-DD&dataFim=AAAA-MM-DD";
8
9  lerDados("http://www.ideiasnacaixa.com/pesquisa/resultadoOpticaCor3.php?dataInicio=2010-07-01&dataFim=2012-06-30");
10
11 function lerDados(arquivo) {
12
13
14     var variaveis:LoadVars = new LoadVars();
15     variaveis.onLoad = function(success:Boolean) {
16         if (success) {
17
18             totalDados = variaveis.total;
19             //trace(totalDados);
20
21             dataVar = eval("variaveis.data1");
22             dia = Number(dataVar.substr(0, 2));
23             mes = 30*(Number(dataVar.substr(3, 2))-1);
24             ano = 365*(Number(dataVar.substr(6, 4))-2010);
25             diaTotal0 = dia+mes+ano;
26             XO = 50;
27             TX = 300;
28             //
29             YO = 550;
30             TY = 1.2;
31             eixoX._y = YO;
32             eixoX._x = 0;
33             for (ki=0; ki<9; ki++) {
34                 attachMovie("eixoX", "eixoX"+ki, 1000000+ki);
35                 attachMovie("numero", "numero"+ki, 2000000+ki);
36                 eval("eixoX"+ki)._y = YO-TY*ki*50;
37
38                 eval("numero"+ki).numero = ki*50;
39                 eval("numero"+ki)._y = YO-TY*ki*50;
40             }
41
42             for (i=1; i<=totalDados; i++) {
43
44                 //trace(i+" .. "+eval("variaveis.cor"+i));
45                 //trace(i+" .. "+eval("variaveis.selecionado"+i));
46                 //trace(i+" .. "+eval("variaveis.data"+i));
47
48                 cor = eval("variaveis.cor"+i);
49                 if (cor == "rgb") {
50                     attachMovie("pontoBBGRRR", "ponto"+i, i);
51                 }
52                 if (cor == "ygb") {
53                     attachMovie("pontoRBBGG", "ponto"+i, i);
54                 }
55                 if (cor == "ryb") {
56                     attachMovie("pontoRGBRRR", "ponto"+i, i);
57                 }
58                 if (cor == "rgy") {
59                     attachMovie("pontoRGGRRR", "ponto"+i, i);
60                 }
61                 if (cor == "ymb") {
62                     attachMovie("pontoBRRGBB", "ponto"+i, i);
63                 }
64                 if (cor == "ygm") {
65                     attachMovie("pontoBRRGGG", "ponto"+i, i);
66                 }
67                 if (cor == "rym") {
68                     attachMovie("pontoBRRGRR", "ponto"+i, i);
69                 }

```

Reprodução da parte principal do *script* de programação do gráfico da Evolução das Escolhas dos Grupos de Cores (script 1/2).


```

70     if (cor == "mgb") {
71         attachMovie("pontoBRBBGG", "ponto"+i,i);
72     }
73     if (cor == "rmb") {
74         attachMovie("pontoBRBBRR", "ponto"+i,i);
75     }
76     if (cor == "rgm") {
77         attachMovie("pontoBRGGRR", "ponto"+i,i);
78     }
79     if (cor == "ymc") {
80         attachMovie("pontoGBBRRG", "ponto"+i,i);
81     }
82     if (cor == "mcb") {
83         attachMovie("pontoGBBRRB", "ponto"+i,i);
84     }
85     if (cor == "mgc") {
86         attachMovie("pontoGBBRGG", "ponto"+i,i);
87     }
88     if (cor == "rmc") {
89         attachMovie("pontoGBBRRR", "ponto"+i,i);
90     }
91     if (cor == "ych") {
92         attachMovie("pontoGBRGGB", "ponto"+i,i);
93     }
94     if (cor == "ygc") {
95         attachMovie("pontoGBRGGG", "ponto"+i,i);
96     }
97     if (cor == "ryc") {
98         attachMovie("pontoGBRGRR", "ponto"+i,i);
99     }
100    if (cor == "cgb") {
101        attachMovie("pontoGBBBGG", "ponto"+i,i);
102    }
103    if (cor == "rcb") {
104        attachMovie("pontoGBBBRR", "ponto"+i,i);
105    }
106    if (cor == "rgc") {
107        attachMovie("pontoGBGGRR", "ponto"+i,i);
108    }
109    dataVar = eval("variaveis.data"+i);
110    IP = eval("variaveis.IP"+i);
111    corSelec = eval("variaveis.cor"+i);
112    selecao = eval("variaveis.selecionado"+i);
113    dia = Number(dataVar.substr(0, 2));
114    mes = 30.4*(Number(dataVar.substr(3, 2))-1);
115    ano = 365*(Number(dataVar.substr(6, 4))-2010);
116    diaTotal = dia+mes+ano;
117    //diaEixoX = 1.6*diaTotal-300;
118
119    diaEixoX = X0+IX*(diaTotal-diaTotal0)/diaTotal0;
120
121    quantidade = eval("variaveis.selecionado"+i);
122    //if (diaTotal == {1+90+365} or diaTotal == {1+150+365} or diaTotal == {1+245+365}) {
123    //attachMovie("divisoria", "divisoria"+i,100000+i);
124    //eval("divisoria"+i)._x = diaEixoX;
125    //eval("divisoria"+i)._y = Y0;
126    //}
127    // marcar a data na absissa
128    DT = int(diaTotal);
129    if (DT%30 == 0) {
130        DTO = DT;
131    } else {
132        if ((DT-1)%30 == 0) {
133            DTO = DT-1;
134        } else {
135            if ((DT+1)%30 == 0) {
136                DTO = DT+1;
137            }
138        }
139    }

```

Reprodução da parte principal do *script* de programação do gráfico da Evolução das Escolhas dos Grupos de Cores (script 2/2).

APÊNDICE D

Códigos PHP das interfaces de visualização de resultados do método de *Log* de dados

1. Tabela de Resultados de Escolhas de Correção da Visão

```

<?php
echo "<head>";
echo "<title>pesquisa - forças</title>";
echo "</head>";

echo "<body bgcolor=#666666>";

// conexao ao banco de dados
$conexao=mysql_connect("mysql://mysql:root@localhost:3306", "root@localhost", "root@localhost");
or die ("falhou 1");
// selecionar bd
$db=mysql_select_db("ideiasnaca")
or die ("falhou 2");

echo "<div align=center>";
echo "<table width=750 cellpadding=0 border=0 bgcolor=#ffffff>";
echo "</table>";
echo "</div>";

echo "<div align=center>";
echo "<table border=0 cellpadding=5 cellspacing=0 width=750 height=40 bgcolor=#ffffff>";
echo "<tr>";
echo "<td>";

echo "<p align=center><font face=Arial size=3><b>";
echo "resultado de escolhas de correção de visão</b><br>";

echo "<p align=center><font face=Arial size=2>";
echo "formato do requisito:<br>";
echo "?dataInicio=AAAA-MM-DD&dataFim=AAAA-MM-DD<br>";

////////////////////////////////////
// correcao de formato de data //
////////////////////////////////////

$dataok = mktime(0, 0, 0, $mesInicio, $diaInicio, $anoInicio);
//$dataInicio = date ("Y-m-d", $dataok);

$dataok = mktime(0, 0, 0, $mesFim, $diaFim, $anoFim);
//$dataFim = date ("Y-m-d", $dataok);

////////////////////////////////////

echo "<div align=center>";
echo "<table width=600 border=1 cellpadding=2 cellspacing=0>";

echo "<tr>";
echo "<td>";
echo "<font face=Arial size=1>id";
echo "</td>";
echo "<td>";
echo "<font face=Arial size=1>ip";
echo "</td>";
echo "<td>";
echo "<font face=Arial size=1>resposta";
echo "</td>";
echo "<td>";
echo "<font face=Arial size=1>data";
echo "</td>";
echo "<td>";
echo "<font face=Arial size=1>horário";
echo "</td>";
echo "</tr>";

$contadorMiopiaSim=0;
$contadorMiopiaNao=0;
$contadorHiperSim=0;
$contadorHiperNao=0;

if ($dataInicio and $dataFim){
    $sql="SELECT * FROM pesquisaopticaOlhoCorrecao where data>='$dataInicio' and data<='$dataFim' ORDER BY data ASC";
} else {
    $sql="SELECT * FROM pesquisaopticaOlhoCorrecao ORDER BY data ASC";
}

$resultado=mysql_query($sql);
while ($linha=mysql_fetch_array($resultado)) { //while 1
    $id = $linha["id"];
    $ip = $linha["ip"];
    $resposta = $linha["resposta"];
    $data = $linha["data"];

```

Reprodução do *script* de programação da tabela de Resultados de Escolhas de Correção da Visão (script 1/2).

```

if ($resposta=="miopiaSim"){ $contadorMiopiaSim++;}
if ($resposta=="miopiaNao"){ $contadorMiopiaNao++;}
if ($resposta=="hiperSim"){ $contadorHiperSim++;}
if ($resposta=="hiperNao"){ $contadorHiperNao++;}

// yyyy-mm-dd hh:mm:ss
$dataDMA = substr($data, 8, 2)."-".substr($data, 5, 2)."-".substr($data, 0, 4);
$horario = substr($data, 10, 17);

    $contadorDados++;

    echo "<tr>";
        echo "<td>";
            echo "<font face=Arial size=1>$contadorDados";
        echo "</td>";
        echo "<td>";
            echo "<font face=Arial size=1>$ip";
        echo "</td>";
        echo "<td>";
            echo "<font face=Arial size=1>$resposta";
        echo "</td>";
        echo "<td>";
            echo "<font face=Arial size=1>$dataDMA";
        echo "</td>";
        echo "<td>";
            echo "<font face=Arial size=1>$horario";
            //echo "<font face=Arial size=1>$sequencia";
        echo "</td>";
    echo "</tr>";
} // while 1

    echo "</table>";
echo "</div>";

echo "<p align=center><font face=Arial size=1>";
echo "miopia resposta correta: $contadorMiopiaSim<br>";
echo "miopia resposta errada: $contadorMiopiaNao<br>";
echo "hipermetropia resposta correta: $contadorHiperSim<br>";
echo "hipermetropia resposta errada: $contadorHiperNao<br>";

        echo "</td>";
    echo "</tr>";
echo "</table>";
echo "</div>";

echo "<div align=center>";
echo "<table width=750 cellpadding=0 border=0 bgcolor=#ffffff>";
echo "</table>";
echo "</div>";

echo "</body>";

?>

```

Reprodução do *script* de programação da tabela de Resultados de Escolhas de Correção da Visão (script 2/2).

2. Tabela de Escolha dos Grupos de Cores

```

<?php
echo "<head>";
echo "<title>pesquisa - óptica</title>";
echo "</head>";

echo "<body bgcolor=#666666>";

// conexao ao banco de dados
$conexao=mysql_connect("mysql122.ufeservidor.com", "ideiasnaca", "null@escol1")
or die ("falhou 1");
// selecionar bd
$db=mysql_select_db("ideiasnaca")
or die ("falhou 2");

// tabela de cores
$coresArray = array(
    'corRR'=>'#ff0000',
    'corGG'=>'#00ff00',
    'corBB'=>'#0000ff',
    'corRG'=>'#ffff00',
    'corBR'=>'#ff00ff',
    'corGB'=>'#00ffff'
);

$coresSeqArray = array(
    'corRR'=>1,
    'corGG'=>10,
    'corBB'=>100,
    'corRG'=>1000,
    'corBR'=>10000,
    'corGB'=>100000
);

$ocorrenciaaBBGGRR=0; // 111
$ocorrenciaaRGBBGG=0; // 1110
$ocorrenciaaRGBBRR=0; // 1101
$ocorrenciaaRGGBRR=0; // 1011
$ocorrenciaaBRGGBB=0; // 11100
$ocorrenciaaBRGGGG=0; // 11010
$ocorrenciaaBRGGRR=0; // 11001
$ocorrenciaaBRBBGG=0; // 10110
$ocorrenciaaBRBBRR=0; // 10101
$ocorrenciaaBRGGRR=0; // 10011
$ocorrenciaaGBBRRG=0; // 111000
$ocorrenciaaGBBRRB=0; // 110100
$ocorrenciaaGBBRRG=0; // 110010
$ocorrenciaaGBBRRR=0; // 110001
$ocorrenciaaGBRGGBB=0; // 101100
$ocorrenciaaGBRGGGG=0; // 101010
$ocorrenciaaGBRGGRR=0; // 101001
$ocorrenciaaGBBBGG=0; // 100110
$ocorrenciaaGBBBRR=0; // 100101
$ocorrenciaaGBGGRR=0; // 100011

echo "<div align=center>";
echo "<table width=750 cellpadding=0 border=0 bgcolor=#ffffff>";
echo "</table>";
echo "</div>";

echo "<div align=center>";
    echo "<table border=0 cellpadding=5 cellspacing=0 width=750 height=40 bgcolor=#ffffff>";
        echo "<tr>";
            echo "<td>";

////////////////////////////////////
// formulário //
////////////////////////////////////

////////////////////////////////////
// correcao de formato de data //
////////////////////////////////////

$dataok = mktime(0, 0, 0, $mesInicio, $diaInicio, $anoInicio);
//$dataInicio = date ("Y-m-d", $dataok);

$dataok = mktime(0, 0, 0, $mesFim, $diaFim, $anoFim);
//$dataFim = date ("Y-m-d", $dataok);

////////////////////////////////////

```

Reprodução do *script* de programação da Tabela de Escolha dos Grupos de Cores (script 1/7).

```

echo "<p align=center><font face=Arial size=3><b>";
echo "tabela de escolha dos grupos de cores</b><br>";

echo "<p align=center><font face=Arial size=2>";
echo "Formato do requisito:<br>";
echo "?dataInicio=AAAA-MM-DD&dataFim=AAAA-MM-DD<br>";

    echo "<div align=center>";
        echo "<table width=600 border=1 cellpadding=2 cellspacing=0>";

echo "<tr>";
    echo "<td>";
        echo "<font face=Arial size=1>id";
    echo "</td>";
    echo "<td>";
        echo "<font face=Arial size=1>ip";
    echo "</td>";
    echo "<td>";
        echo "<font face=Arial size=1>cor 1";
    echo "</td>";
    echo "<td>";
        echo "<font face=Arial size=1>cor 2";
    echo "</td>";
    echo "<td>";
        echo "<font face=Arial size=1>cor 3";
    echo "</td>";
    echo "<td>";
        echo "<font face=Arial size=1>data";
    echo "</td>";
    echo "<td>";
        echo "<font face=Arial size=1>horário";
    echo "</td>";
echo "</tr>";

$contadorDados=0;
//$dataInicio = "2010-07-26";
//$dataFim = "2010-07-30";
if ($dataInicio and $dataFim){
    $sql="SELECT * FROM pesquisaoptica01 where data>='$dataInicio' and data<='$dataFim' ORDER BY data ASC";
} else {
    $sql="SELECT * FROM pesquisaoptica01 ORDER BY data ASC";
}
$resultado=mysql_query($sql);
while ($linha=mysql_fetch_array($resultado)) { //while 1
    $id = $linha["id"];
    $ip = $linha["ip"];
    $cor1 = $linha["cor1"];
    $cor2 = $linha["cor2"];
    $cor3 = $linha["cor3"];
    $data = $linha["data"];

    $sequencia = $coresSeqArray[$cor1] + $coresSeqArray[$cor2] + $coresSeqArray[$cor3];
    if ($sequencia == 111){
        $ocorrenciaBBGGRR++;
    }
    if ($sequencia == 1110){
        $ocorrenciaRGBBGG++;
    }
    if ($sequencia == 1101){
        $ocorrenciaRGBBRR++;
    }
    if ($sequencia == 1011){
        $ocorrenciaRGGGRR++;
    }
    if ($sequencia == 11100){
        $ocorrenciaBRRGBB++;
    }
    if ($sequencia == 11010){
        $ocorrenciaBRRGGG++;
    }
    if ($sequencia == 11001){
        $ocorrenciaBRRGRR++;
    }
    if ($sequencia == 10110){
        $ocorrenciaBRBBGG++;
    }
    if ($sequencia == 10101){
        $ocorrenciaBRBBRR++;
    }
}

```

Reprodução do *script* de programação da Tabela de Escolha dos Grupos de Cores (script 2/7).

```

if ($sequencia == 10011){
    $ocorrenciaBRGGRR++;
}
if ($sequencia == 111000){
    $ocorrenciaGBBRRG++;
}
if ($sequencia == 111000){
    $ocorrenciaGBBRRG++;
}
if ($sequencia == 110100){
    $ocorrenciaGBBRRB++;
}
if ($sequencia == 110010){
    $ocorrenciaGBBRRG++;
}
if ($sequencia == 110001){
    $ocorrenciaGBBRRR++;
}
if ($sequencia == 101100){
    $ocorrenciaGBRGGB++;
}
if ($sequencia == 101010){
    $ocorrenciaGBRGGG++;
}
if ($sequencia == 101001){
    $ocorrenciaGBRGRR++;
}
if ($sequencia == 100110){
    $ocorrenciaGBBBGG++;
}
if ($sequencia == 100101){
    $ocorrenciaGBBBRR++;
}
if ($sequencia == 100011){
    $ocorrenciaGBGGRR++;
}
}

// yyyy-mm-dd hh:mm:ss
$dataDMA = substr($data, 8, 2)."-".substr($data, 5, 2)."-".substr($data, 0, 4);
$horario = substr($data, 10, 17);

$contadorDados++;

echo "<tr>";
    echo "<td>";
        echo "<font face=Arial size=1>$contadorDados";
    echo "</td>";
    echo "<td>";
        echo "<font face=Arial size=1>$ip";
    echo "</td>";
    echo "<td bgColor=$coresArray[$cor1]>";
        //echo "<font face=Arial size=1>$coresArray[$cor1]";
    echo "</td>";
    echo "<td bgColor=$coresArray[$cor2]>";
        //echo "<font face=Arial size=1>$coresArray[$cor2]";
    echo "</td>";
    echo "<td bgColor=$coresArray[$cor3]>";
        //echo "<font face=Arial size=1>$coresArray[$cor3]";
    echo "</td>";
    echo "<td>";
        echo "<font face=Arial size=1>$dataDMA";
    echo "</td>";
    echo "<td>";
        echo "<font face=Arial size=1>$horario";
        //echo "<font face=Arial size=1>$sequencia";
    echo "</td>";
echo "</tr>";
} // while 1

echo "</table>";
echo "</div>";

echo "<div align=center>";
    echo "<table height=20 width=600 border=0 cellpadding=20 cellspacing=0>";
        echo "<tr>";
            echo "<td>";
                echo "</td>";
            echo "</tr>";
        echo "</table>";
echo "</div>";

```

Reprodução do *script* de programação da Tabela de Escolha dos Grupos de Cores (script 3/7).

```

        echo "<div align=center>";
        echo "<table width=200 border=1 cellpadding=2 cellspacing=0>";
echo "<tr>";
    echo "<td>";
        echo "<p align=right><font face=Arial size=1>$ocorrenciaBBGGRR";
    echo "</td>";
    echo "<td bgColor=#ff0000>";
    echo "</td>";
    echo "<td bgColor=#00ff00>";
    echo "</td>";
    echo "<td bgColor=#0000ff>";
    echo "</td>";
    echo "<td>";
        echo "<p align=right><font face=Arial size=1>grupo 1";
    echo "</td>";
echo "</tr>";
//
echo "<tr>";
    echo "<td>";
        echo "<p align=right><font face=Arial size=1>$ocorrenciaRGBGG";
    echo "</td>";
    echo "<td bgColor=#ffff00>";
    echo "</td>";
    echo "<td bgColor=#00ff00>";
    echo "</td>";
    echo "<td bgColor=#0000ff>";
    echo "</td>";
    echo "<td>";
        echo "<p align=right><font face=Arial size=1>grupo 2";
    echo "</td>";
echo "</tr>";
//
echo "<tr>";
    echo "<td>";
        echo "<p align=right><font face=Arial size=1>$ocorrenciaRGGGRR";
    echo "</td>";
    echo "<td bgColor=#ff0000>";
    echo "</td>";
    echo "<td bgColor=#00ff00>";
    echo "</td>";
    echo "<td bgColor=#ffff00>";
    echo "</td>";
    echo "<td>";
        echo "<p align=right><font face=Arial size=1>grupo 4";
    echo "</td>";
echo "</tr>";
//
echo "<tr>";
    echo "<td>";
        echo "<p align=right><font face=Arial size=1>$ocorrenciaBRRGBB";
    echo "</td>";
    echo "<td bgColor=#ffff00>";
    echo "</td>";
    echo "<td bgColor=#ff00ff>";
    echo "</td>";
    echo "<td bgColor=#0000ff>";
    echo "</td>";
    echo "<td>";
        echo "<p align=right><font face=Arial size=1>grupo 5";
    echo "</td>";
echo "</tr>";
//
echo "<tr>";
    echo "<td>";
        echo "<p align=right><font face=Arial size=1>$ocorrenciaBRRGGG";
    echo "</td>";
    echo "<td bgColor=#ffff00>";
    echo "</td>";
    echo "<td bgColor=#00ff00>";
    echo "</td>";
    echo "<td bgColor=#ff00ff>";
    echo "</td>";
    echo "<td>";
        echo "<p align=right><font face=Arial size=1>grupo 6";
    echo "</td>";
echo "</tr>";
//

```

Reprodução do *script* de programação da Tabela de Escolha dos Grupos de Cores (script 4/7).

```

echo "<tr>";
echo "    <td>";
echo "        <p align=right><font face=Arial size=1>$ocorrenciaBRRGRR";
echo "    </td>";
echo "    <td bgColor=#ff0000>";
echo "    </td>";
echo "    <td bgColor=#ffff00>";
echo "    </td>";
echo "    <td bgColor=#ff00ff>";
echo "    </td>";
echo "    <td>";
echo "        <p align=right><font face=Arial size=1>grupo 7";
echo "    </td>";
echo "</tr>";
//
echo "<tr>";
echo "    <td>";
echo "        <p align=right><font face=Arial size=1>$ocorrenciaBRBBGG";
echo "    </td>";
echo "    <td bgColor=#ff00ff>";
echo "    </td>";
echo "    <td bgColor=#00ff00>";
echo "    </td>";
echo "    <td bgColor=#0000ff>";
echo "    </td>";
echo "    <td>";
echo "        <p align=right><font face=Arial size=1>grupo 8";
echo "    </td>";
echo "</tr>";
//
echo "<tr>";
echo "    <td>";
echo "        <p align=right><font face=Arial size=1>$ocorrenciaBRBBRR";
echo "    </td>";
echo "    <td bgColor=#ff0000>";
echo "    </td>";
echo "    <td bgColor=#ff00ff>";
echo "    </td>";
echo "    <td bgColor=#0000ff>";
echo "    </td>";
echo "    <td>";
echo "        <p align=right><font face=Arial size=1>grupo 9";
echo "    </td>";
echo "</tr>";
//
echo "<tr>";
echo "    <td>";
echo "        <p align=right><font face=Arial size=1>$ocorrenciaBRGGRR";
echo "    </td>";
echo "    <td bgColor=#ff0000>";
echo "    </td>";
echo "    <td bgColor=#00ff00>";
echo "    </td>";
echo "    <td bgColor=#ff00ff>";
echo "    </td>";
echo "    <td>";
echo "        <p align=right><font face=Arial size=1>grupo 10";
echo "    </td>";
echo "</tr>";
//
echo "<tr>";
echo "    <td>";
echo "        <p align=right><font face=Arial size=1>$ocorrenciaGBBRRG";
echo "    </td>";
echo "    <td bgColor=#ffff00>";
echo "    </td>";
echo "    <td bgColor=#ff00ff>";
echo "    </td>";
echo "    <td bgColor=#00ffff>";
echo "    </td>";
echo "    <td>";
echo "        <p align=right><font face=Arial size=1>grupo 11";
echo "    </td>";
echo "</tr>";
//

```

Reprodução do *script* de programação da Tabela de Escolha dos Grupos de Cores (*script* 5/7).


```

echo "<tr>";
echo "    <td>";
echo "        <p align=right><font face=Arial size=1>$ocorrenciaGGBRBB";
echo "    </td>";
echo "    <td bgColor=#ff00ff>";
echo "    </td>";
echo "    <td bgColor=#00ffff>";
echo "    </td>";
echo "    <td bgColor=#0000ff>";
echo "    </td>";
echo "    <td>";
echo "        <p align=right><font face=Arial size=1>grupo 12";
echo "    </td>";
echo "</tr>";
//
echo "<tr>";
echo "    <td>";
echo "        <p align=right><font face=Arial size=1>$ocorrenciaGGBRGG";
echo "    </td>";
echo "    <td bgColor=#ff00ff>";
echo "    </td>";
echo "    <td bgColor=#00ff00>";
echo "    </td>";
echo "    <td bgColor=#00ffff>";
echo "    </td>";
echo "    <td>";
echo "        <p align=right><font face=Arial size=1>grupo 13";
echo "    </td>";
echo "</tr>";
//
echo "<tr>";
echo "    <td>";
echo "        <p align=right><font face=Arial size=1>$ocorrenciaGGBRRR";
echo "    </td>";
echo "    <td bgColor=#ff0000>";
echo "    </td>";
echo "    <td bgColor=#ff00ff>";
echo "    </td>";
echo "    <td bgColor=#00ffff>";
echo "    </td>";
echo "    <td>";
echo "        <p align=right><font face=Arial size=1>grupo 14";
echo "    </td>";
echo "</tr>";
//
echo "<tr>";
echo "    <td>";
echo "        <p align=right><font face=Arial size=1>$ocorrenciaGGBRBB";
echo "    </td>";
echo "    <td bgColor=#ffff00>";
echo "    </td>";
echo "    <td bgColor=#00ffff>";
echo "    </td>";
echo "    <td bgColor=#0000ff>";
echo "    </td>";
echo "    <td>";
echo "        <p align=right><font face=Arial size=1>grupo 15";
echo "    </td>";
echo "</tr>";
//
echo "<tr>";
echo "    <td>";
echo "        <p align=right><font face=Arial size=1>$ocorrenciaGGBRGG";
echo "    </td>";
echo "    <td bgColor=#ffff00>";
echo "    </td>";
echo "    <td bgColor=#00ff00>";
echo "    </td>";
echo "    <td bgColor=#00ffff>";
echo "    </td>";
echo "    <td>";
echo "        <p align=right><font face=Arial size=1>grupo 16";
echo "    </td>";
echo "</tr>";
//

```

Reprodução do *script* de programação da Tabela de Escolha dos Grupos de Cores (*script* 6/7).

```

echo "<tr>";
echo "    <td>";
echo "        <p align=right><font face=Arial size=1>$ocorrenciaGGBRGR";
echo "    </td>";
echo "    <td bgColor=#ff0000>";
echo "    </td>";
echo "    <td bgColor=#ffff00>";
echo "    </td>";
echo "    <td bgColor=#00ffff>";
echo "    </td>";
echo "    <td>";
echo "        <p align=right><font face=Arial size=1>grupo 17";
echo "    </td>";
echo "</tr>";
//
echo "<tr>";
echo "    <td>";
echo "        <p align=right><font face=Arial size=1>$ocorrenciaGGBBGG";
echo "    </td>";
echo "    <td bgColor=#00ffff>";
echo "    </td>";
echo "    <td bgColor=#00ff00>";
echo "    </td>";
echo "    <td bgColor=#0000ff>";
echo "    </td>";
echo "    <td>";
echo "        <p align=right><font face=Arial size=1>grupo 18";
echo "    </td>";
echo "</tr>";
//
echo "<tr>";
echo "    <td>";
echo "        <p align=right><font face=Arial size=1>$ocorrenciaGBBBRR";
echo "    </td>";
echo "    <td bgColor=#ff0000>";
echo "    </td>";
echo "    <td bgColor=#00ffff>";
echo "    </td>";
echo "    <td bgColor=#0000ff>";
echo "    </td>";
echo "    <td>";
echo "        <p align=right><font face=Arial size=1>grupo 19";
echo "    </td>";
echo "</tr>";
//
echo "<tr>";
echo "    <td>";
echo "        <p align=right><font face=Arial size=1>$ocorrenciaGGBGRR";
echo "    </td>";
echo "    <td bgColor=#ff0000>";
echo "    </td>";
echo "    <td bgColor=#00ff00>";
echo "    </td>";
echo "    <td bgColor=#00ffff>";
echo "    </td>";
echo "    <td>";
echo "        <p align=right><font face=Arial size=1>grupo 20";
echo "    </td>";
echo "</tr>";
//
echo "    </table>";
echo "</div>";

echo "        </td>";
echo "    </tr>";
echo "</table>";
echo "</div>";

echo "<div align=center>";
echo "<table width=750 cellpadding=0 border=0 bgcolor=#ffffff>";
echo "</table>";
echo "</div>";

echo "</body>";

?>

```

Reprodução do *script* de programação da Tabela de Escolha dos Grupos de Cores (*script 7/7*).

APÊNDICE E

Códigos PHP das interfaces de visualização do Resultado do questionário preenchido pelos usuários

```

<?php
echo "<head>";
echo "<title>pesquisa de usuário</title>";
echo "</head>";

echo "<body bgcolor=#666666>";

// conexao ao banco de dados
$conexao=mysql_connect("mysql1022.whserv102r.com", "ideiasnaca", "n0ll2wex1")
or die ("falhou 1");
// selecionar bd
$db = mysql_select_db("ideiasnaca")
or die ("falhou 2");

echo "<div align=center>";
echo "<table border=0 cellpadding=5 cellspacing=0 width=100% height=40
bgcolor=#ffffff>";
echo "<tr>";
echo "<td>";

////////////////////
// correcao de formato de data
////////////////////

$dataok = mktime(0, 0, 0, $mesInicio, $diaInicio, $anoInicio);
$dataInicio = date ("Y-m-d", $dataok);

$dataok = mktime(0, 0, 0, $mesFim, $diaFim, $anoFim);
$dataFim = date ("Y-m-d", $dataok);

////////////////////
echo "<div align=center>";
echo "<table width=100% border=1 cellpadding=2 cellspacing=0>";

echo "<p align=center><font face=arial size=2><b>";
echo "resultado do questionário preenchido pelos usuários<br>";

echo "<tr>";
echo "<td>";
echo "<font face=Arial size=1>id";
echo "</td>";
echo "<td>";
echo "<font face=Arial size=1>ip";
echo "</td>";
echo "<td>";
echo "<font face=Arial size=1>usuario";
echo "</td>";
echo "<td>";
echo "<font face=Arial size=1>fundamental";
echo "</td>";
echo "<td>";
echo "<font face=Arial size=1>medio";
echo "</td>";
echo "<td>";
echo "<font face=Arial size=1>superior";
echo "</td>";
echo "<td>";
echo "<font face=Arial size=1>outro";
echo "</td>";
echo "<td>";
echo "<font face=Arial size=1>comentario";
echo "</td>";
echo "<td>";
echo "<font face=Arial size=1>data";
echo "</td>";
echo "</tr>";

$contadorAluno=0;
$contadorProfessor=0;
$contadorUsuario=0;

$contadorAlunoFundamental = 0;
$contadorAlunoMedio = 0;
$contadorAlunoOutro = 0;
$contadorAlunoSuperior = 0;
$contadorAlunoNenhum = 0;

$contadorAlunoFundamental0 = 0;
$contadorAlunoFundamental1 = 0;
$contadorAlunoFundamental2 = 0;
$contadorAlunoFundamental3 = 0;
$contadorAlunoFundamental4 = 0;
$contadorAlunoFundamental5 = 0;
$contadorAlunoFundamental6 = 0;
$contadorAlunoFundamental7 = 0;
$contadorAlunoFundamental8 = 0;
$contadorAlunoFundamental9 = 0;

```

Reprodução do *script* de programação do resultado do questionário preenchido pelos usuários (script 1/6).

```

$contadorAlunoMedio0 = 0;
$contadorAlunoMedio1 = 0;
$contadorAlunoMedio2 = 0;
$contadorAlunoMedio3 = 0;

$contadorProfFundamental = 0;
$contadorProfMedio = 0;
$contadorProfOutro = 0;

$contadorProfFundamentalMedio = 0;
$contadorProfFundamentalOutro = 0;
$contadorProfMedioOutro = 0;
$contadorProfFundamentalMedioOutro = 0;

$contadorProfNenhum = 0;

    $contadorProfEnsinoPort=0;
    $contadorProfEnsinoMat=0;
    $contadorProfEnsinoCien=0;
    $contadorProfEnsinoBio=0;
    $contadorProfEnsinoFis=0;
    $contadorProfEnsinoQui=0;
    $contadorProfEnsinoHis=0;
    $contadorProfEnsinoGeo=0;
    $contadorProfEnsinoSoc=0;
    $contadorProfEnsinoAnt=0;
    $contadorProfEnsinoFil=0;
    $contadorProfEnsinoPol=0;
    $contadorProfEnsinoLin=0;
    $contadorProfEnsinoArt=0;
    $contadorProfEnsinoInf=0;
    $contadorProfEnsinoEdf=0;
    $contadorProfEnsinoTran=0;
    $contadorProfEnsinoMeio=0;
    $contadorProfEnsinoSau=0;
    $contadorProfEnsinoCul=0;
    $contadorProfEnsinoSex=0;

$contadorUsuarioFundamental = 0;
$contadorUsuarioMedio = 0;
$contadorUsuarioSuperior = 0;
$contadorUsuarioOutro = 0;
$contadorUsuarioNenhum = 0;

$sql="SELECT * FROM pesquisausuario ORDER BY usuario, data desc";
// where datavisita>='$dataInicio' and datavisita<='$dataFim' ORDER BY data ASC";

$resultado=mysql_query($sql);

while ($linha=mysql_fetch_array($resultado)) { //while 1

    $id = $linha["id"];
    $ip = $linha["ip"];
    $data = $linha["data"];

    $usuario = $linha["usuario"];

    ///////////////////////////////////
    // verificar ultimo IP //
    if ($ultimoIp<>$ip){
    ///////////////////////////////////

    $ensinoFundamental = "";
    $ensinoMedio = "";
    $ensinoSuperior = "";
    $ensinoOutro = "";

    if ($usuario=="aluno"){
        $contadorAluno++;
        $ensinoAluno = $linha["alunoMatricula"];
        if ($ensinoAluno=="fundamental"){
            $alunoEnsinoFundamental = $linha["alunoEnsinoFundamental"];
            $ensinoFundamental = $alunoEnsinoFundamental;

            if ($ensinoFundamental==1){
                $contadorAlunoFundamental1++;
            }
            if ($ensinoFundamental==2){
                $contadorAlunoFundamental2++;
            }
            if ($ensinoFundamental==3){
                $contadorAlunoFundamental3++;
            }
            if ($ensinoFundamental==4){
                $contadorAlunoFundamental4++;
            }
            if ($ensinoFundamental==5){
                $contadorAlunoFundamental5++;
            }
        }
    }
}

```

Reprodução do *script* de programação do resultado do questionário preenchido pelos usuários (script 2/6).

```

        if ($ensinoFundamental==6){
            $contadorAlunoFundamental6++;
        }
        if ($ensinoFundamental==7){
            $contadorAlunoFundamental7++;
        }
        if ($ensinoFundamental==8){
            $contadorAlunoFundamental8++;
        }
        if ($ensinoFundamental==9){
            $contadorAlunoFundamental9++;
        }
        if ($ensinoFundamental==0){
            $contadorAlunoFundamental0++;
        }
        $contadorAlunoFundamental++;
    }
    if ($ensinoAluno=="medio"){
        $alunoEnsinoMedio = $linha["alunoEnsinoMedio"];
        $ensinoMedio = $alunoEnsinoMedio;

        if ($ensinoMedio==1){
            $contadorAlunoMedio1++;
        }
        if ($ensinoMedio==2){
            $contadorAlunoMedio2++;
        }
        if ($ensinoMedio==3){
            $contadorAlunoMedio3++;
        }
        if ($ensinoMedio==0){
            $contadorAlunoMedio0++;
        }
        $contadorAlunoMedio++;
    }
    if ($ensinoAluno=="outra"){
        $alunoOutroEnsino = $linha["alunoOutroEnsino"];
        if ($alunoOutroEnsino){
            $ensinoOutro = $alunoOutroEnsino;
        }
        $contadorAlunoOutro++;
    }
    if (!$ensinoAluno){
        $contadorAlunoNenhum++;
    }
    $alunoIdade = $linha["alunoIdade"];
    $alunoMaterias = $linha["alunoMaterias"];
}

if ($usuario=="professor"){
    $contadorProfessor++;

    $ensinoFundamental = $linha["profEnsinoFundamental"];
    $ensinoMedio = $linha["profEnsinoMedio"];
    $profOutroArea = $linha["profOutroArea"];

    if ($ensinoFundamental and !$ensinoMedio and !$profOutroArea){
        $contadorProfFundamental++;
    }
    if (!$ensinoFundamental and $ensinoMedio and !$profOutroArea){
        $contadorProfMedio++;
    }
    if (!$ensinoFundamental and !$ensinoMedio and $profOutroArea){
        $contadorProfOutro++;
    }
    if ($ensinoFundamental and $ensinoMedio and !$profOutroArea){
        $contadorProfFundamentalMedio++;
    }
    if ($ensinoFundamental and !$ensinoMedio and $profOutroArea){
        $contadorProfFundamentalOutro++;
    }
    if (!$ensinoFundamental and $ensinoMedio and $profOutroArea){
        $contadorProfMedioOutro++;
    }
    if ($ensinoFundamental and $ensinoMedio and $profOutroArea){
        $contadorProfFundamentalMedioOutro++;
    }
    if (!$ensinoFundamental and !$ensinoMedio and !$profOutroArea){
        $contadorProfNenhum++;
    }
}

```

Reprodução do *script* de programação do resultado do questionário preenchido pelos usuários (script 3/6).

```

$profEnsinoPort = $linha["profEnsinoPort"];
if ($profEnsinoPort){
    $contadorProfEnsinoPort++;
}
$profEnsinoMat = $linha["profEnsinoMat"];
if ($profEnsinoMat){
    $contadorProfEnsinoMat++;
}
$profEnsinoCien = $linha["profEnsinoCien"];
if ($profEnsinoCien){
    $contadorProfEnsinoCien++;
}
$profEnsinoBio = $linha["profEnsinoBio"];
if ($profEnsinoBio){
    $contadorProfEnsinoBio++;
}

$profEnsinoFis = $linha["profEnsinoFis"];
if ($profEnsinoFis){
    $contadorProfEnsinoFis++;
}
$profEnsinoQui = $linha["profEnsinoQui"];
if ($profEnsinoQui){
    $contadorProfEnsinoQui++;
}
$profEnsinoHis = $linha["profEnsinoHis"];
if ($profEnsinoHis){
    $contadorProfEnsinoHis++;
}
$profEnsinoGeo = $linha["profEnsinoGeo"];
if ($profEnsinoGeo){
    $contadorProfEnsinoGeo++;
}

$profEnsinoSoc = $linha["profEnsinoSoc"];
if ($profEnsinoSoc){
    $contadorProfEnsinoSoc++;
}
$profEnsinoAnt = $linha["profEnsinoAnt"];
if ($profEnsinoAnt){
    $contadorProfEnsinoAnt++;
}
$profEnsinoFil = $linha["profEnsinoFil"];
if ($profEnsinoFil){
    $contadorProfEnsinoFil++;
}
$profEnsinoPol = $linha["profEnsinoPol"];
if ($profEnsinoPol){
    $contadorProfEnsinoPol++;
}

$profEnsinoLin = $linha["profEnsinoLin"];
if ($profEnsinoLin){
    $contadorProfEnsinoLin++;
}
$profEnsinoArt = $linha["profEnsinoArt"];
if ($profEnsinoArt){
    $contadorProfEnsinoArt++;
}
$profEnsinoInf = $linha["profEnsinoInf"];
if ($profEnsinoInf){
    $contadorProfEnsinoInf++;
}
$profEnsinoEdf = $linha["profEnsinoEdf"];
if ($profEnsinoEdf){
    $contadorProfEnsinoEdf++;
}
$profEnsinoTran = $linha["profEnsinoTran"];
if ($profEnsinoTran){
    $contadorProfEnsinoTran++;
}

$profEnsinoMeio = $linha["profEnsinoMeio"];
if ($profEnsinoMeio){
    $contadorProfEnsinoMeio++;
}
$profEnsinoSau = $linha["profEnsinoSau"];
if ($profEnsinoSau){
    $contadorProfEnsinoSau++;
}
$profEnsinoCul = $linha["profEnsinoCul"];
if ($profEnsinoCul){
    $contadorProfEnsinoCul++;
}
$profEnsinoSex = $linha["profEnsinoSex"];
if ($profEnsinoSex){
    $contadorProfEnsinoSex++;
}
}

```

Reprodução do *script* de programação do resultado do questionário preenchido pelos usuários (script 4/6).

```

if ($usuario=="usuario"){
    $contadorUsuario++;

    $usuarioEscolaridade =$linha["usuarioEscolaridade"];
    if ($usuarioEscolaridade=="fundamental"){
        $ensinoFundamental = true;
        $contadorUsuarioFundamental++;
    }
    if ($usuarioEscolaridade=="medio"){
        $ensinoMedio = true;
        $contadorUsuarioMedio++;
    }
    if ($usuarioEscolaridade=="superior"){
        $ensinoSuperior = true;
        $contadorUsuarioSuperior++;
    }
    if ($usuarioEscolaridade=="outra"){
        $ensinoOutro = true;
        $contadorUsuarioOutro++;
    }
    if (!$usuarioEscolaridade){
        $contadorUsuarioNenhum++;
    }

    $usuarioOutraEscolaridade =$linha["usuarioOutraEscolaridade"];
    $usuarioProfissao =$linha["usuarioProfissao"];
}

$comentario =$linha["comentario"];

// yyyy-mm-dd hh:mm:ss
$dataDMA = substr($data, 8, 2)."-".substr($data, 5, 2)."-".substr($data, 0, 4);
$horario = substr($data, 10, 17);

$dataDMA0 = substr($data, 0, 10);
if ($dataDMA1<>$dataDMA0){
    $contadorData=0;
    $horario = $dataDMA0;
}

$contadorData++;

$dataDMA=$dataDMA." (".$contadorData.")";

$dataDMA1=$dataDMA0;

$contadorDados++;

echo "<tr>";
    echo "<td>";
        echo "<font face=Arial size=1>$contadorDados";
    echo "</td>";
    echo "<td>";
        echo "<font face=Arial size=1>$ip";
    echo "</td>";
    echo "<td>";
        echo "<font face=Arial size=1>$usuario";
    echo "</td>";
    echo "<td>";
        echo "<font face=Arial size=1>$ensinoFundamental";
    echo "</td>";
    echo "<td>";
        echo "<font face=Arial size=1>$ensinoMedio";
    echo "</td>";
    echo "<td>";
        echo "<font face=Arial size=1>$ensinoSuperior";
    echo "</td>";
    echo "<td>";
        echo "<font face=Arial size=1>$ensinoOutro";
    echo "</td>";
    echo "<td>";
        echo "<font face=Arial size=1>$comentario";
    echo "</td>";
    echo "<td>";
        echo "<font face=Arial size=1>$data";
    echo "</td>";
    //echo "<td>";
        echo "<font face=Arial size=1>$horario";
    //echo "<td>";
        echo "<font face=Arial size=1>$sequencia";
    echo "</td>";
echo "</tr>";
}

```

Reprodução do *script* de programação do resultado do questionário preenchido pelos usuários (script 5/6).

```

$ultimoip = $ip;

} // while 1

        echo "</table>";
        echo "</div>";

echo "<br><font face=arial size=2><b>";
echo "aluno: $contadorAluno<br>";
echo "> fundamental: $contadorAlunoFundamental<br>";
echo ">>> 1: $contadorAlunoFundamental1<br>";
echo ">>> 2: $contadorAlunoFundamental2<br>";
echo ">>> 3: $contadorAlunoFundamental3<br>";
echo ">>> 4: $contadorAlunoFundamental4<br>";
echo ">>> 5: $contadorAlunoFundamental5<br>";
echo ">>> 6: $contadorAlunoFundamental6<br>";
echo ">>> 7: $contadorAlunoFundamental7<br>";
echo ">>> 8: $contadorAlunoFundamental8<br>";
echo ">>> 9: $contadorAlunoFundamental9<br>";
echo ">>> nenhum: $contadorAlunoFundamental0<br>";
echo "<br>";
echo "> médio: $contadorAlunoMedio<br>";
echo ">>> 1: $contadorAlunoMedio1<br>";
echo ">>> 2: $contadorAlunoMedio2<br>";
echo ">>> 3: $contadorAlunoMedio3<br>";
echo ">>> nenhum: $contadorAlunoMedio0<br>";
echo "<br>";
echo "> outro: $contadorAlunoOutro<br>";
echo "> Nenhum: $contadorAlunoNenhum<br>";

echo "<br>";
echo "professor: $contadorProfessor<br>";
echo "> fundamental: $contadorProfFundamental<br>";
echo "> médio: $contadorProfMedio<br>";
echo "> outro: $contadorProfOutro<br>";

echo "> fundamental e médio: $contadorProfFundamentalMedio<br>";
echo "> fundamental e outro: $contadorProfFundamentalOutro<br>";
echo "> médio e outro: $contadorProfMedioOutro<br>";

echo "> fundamental, médio e outro: $contadorProfFundamentalMedioOutro<br>";

echo "> Nenhum: $contadorProfNenhum<br>";

echo "<br>";
echo ">>> port: $contadorProfEnsinoPort<br>";
echo ">>> mat: $contadorProfEnsinoMat<br>";
echo ">>> cien: $contadorProfEnsinoCien<br>";
echo ">>> bio: $contadorProfEnsinoBio<br>";
echo ">>> fis: $contadorProfEnsinoFis<br>";
echo ">>> qui: $contadorProfEnsinoQui<br>";
echo ">>> his: $contadorProfEnsinoHis<br>";
echo ">>> geo: $contadorProfEnsinoGeo<br>";
echo ">>> soc: $contadorProfEnsinoSoc<br>";
echo ">>> ant: $contadorProfEnsinoAnt<br>";
echo ">>> fil: $contadorProfEnsinoFil<br>";
echo ">>> pol: $contadorProfEnsinoPol<br>";
echo ">>> lin: $contadorProfEnsinoLin<br>";
echo ">>> art: $contadorProfEnsinoArt<br>";
echo ">>> inf: $contadorProfEnsinoInf<br>";
echo ">>> edf: $contadorProfEnsinoEdf<br>";
echo ">>> tran: $contadorProfEnsinoTran<br>";
echo ">>> meio: $contadorProfEnsinoMeio<br>";
echo ">>> sau: $contadorProfEnsinoSau<br>";
echo ">>> cul: $contadorProfEnsinoCul<br>";
echo ">>> sex: $contadorProfEnsinoSex<br>";

echo "<br>";
echo "usuario: $contadorUsuario<br>";
echo ">>> fundamental: $contadorUsuarioFundamental<br>";
echo ">>> medio: $contadorUsuarioMedio<br>";
echo ">>> superior: $contadorUsuarioSuperior<br>";
echo ">>> outro: $contadorUsuarioOutro<br>";
echo ">>> nenhum: $contadorUsuarioNenhum<br>";

        echo "</td>";
        echo "</tr>";
echo "</div>";

echo "<div align=center>";
echo "<table width=750 cellspacing=0 border=0 bgcolor=#ffffff>";
echo "</table>";
echo "</div>";

echo "</body>";
?>

```

Reprodução do *script* de programação do resultado do questionário preenchido pelos usuários (script 6/6).

ANEXO A - As diretrizes de Usabilidade de Jakob Nielsen

No livro *50 websites desconstruídos*, Jakob Nielsen e Marie Tahir apresentam 113 diretrizes de Usabilidade para serem aplicados e analisados nos *websites*. Os autores classificam essas diretrizes em diferentes temáticas e são prescritos para que a construção de um *website* ofereça um ambiente de utilização mais adequada ao usuário. Apesar de algumas dessas diretrizes terem sido revisadas em publicações posteriores, a lista que se segue é um resumo que se baseia no título original.

Informando o objetivo do site

1. Exibir o nome da empresa e/ou logotipo em um tamanho razoável em um local de destaque.
2. Incluir um *slogan* resumindo explicitamente o que o site ou a empresa faz.
3. Enfatizar o que o site faz de importante sob a perspectiva dos usuários, assim como a diferença entre seu site e os dos principais concorrentes.
4. Enfatizar as tarefas de mais alta prioridade, para que os usuários tenham um ponto de partida definido na *homepage*.
5. Designar explicitamente uma página do site como a *homepage* oficial.
6. No *Website* da empresa principal, não usar palavra “*Website*” para se referir a qualquer outro aspecto à totalidade da presença da empresa na *Web*.

Transmitindo informações sobre sua empresa

7. Estruturar a *homepage* de modo diferente de todas as outras páginas existentes no site.
8. Agrupar informações da empresa, como Sobre Nós, Relações com Investidores, Sala de Imprensa, Empregos e outras informações sobre a empresa, em uma única área reservada.
9. Incluir um link da *homepage* para uma seção “Sobre Nós”, que oferece aos usuários uma visão geral sobre a empresa e *links* para todos os detalhes relevantes sobre seus produtos, serviços, valores da empresa, proposta de negócios, equipe de gerenciamento e outros pormenores.
10. Para obter cobertura da imprensa para sua empresa, incluir um *link* “Sala de Imprensa” ou “Sala de Notícias” em sua *homepage*.
11. Apresentar uma face unificada ao cliente, em que o *Website* seja um dos pontos de toque em vez de uma entidade em si.
12. Incluir um link “Fale Conosco” na *homepage*, que acessa uma página com todas as informações de contato de sua empresa.

13. Ao fornecer um mecanismo de “*feedback*” (resposta), especificar o objetivo do link e se será lido pelo atendimento ao cliente ou pelo *Webmaster*, e outras informações pertinentes.
14. Não incluir informações internas da empresa (destinadas aos funcionários e que devem permanecer na Intranet) no *Website* público.
15. Se o site reunir informações de qualquer cliente, é recomendável incluir um *link* “Política Privada” na *homepage*.
16. Explicar como o *Website* gera dinheiro se essa informação não estiver muito clara.

Criação do conteúdo

17. Usar seções e categorias de rótulo, com idioma centrado no cliente, de acordo com a importância dessas seções e categorias para o cliente e não para a empresa.
18. Evitar conteúdo redundante.
19. Não utilizar frases eruditas nem dialeto de *marketing* que fazem com que as pessoas tenham um trabalho para descobrir o que está sendo dito.
20. Empregar letras maiúsculas e outros padrões de estilo com consistência.
21. Não rotular uma área nitidamente definida da página se o conteúdo for suficientemente auto-explicativo.
22. Evitar as categorias e as listas de marcadores de um único item.
23. Utilizar espaços não-separáveis entre as palavras nas frases, que precisam permanecer juntas para serem vistas e entendidas.
24. Usar somente o discurso imperativo, como em “Insira uma Cidade ou CEP” nas tarefas obrigatórias, ou qualificar a declaração adequadamente.
25. Explicar o significado de abreviações, iniciais maiúsculas, acrônimos e segui-los imediatamente com abreviações, na primeira ocorrência.
26. Evitar pontos de exclamação.
27. Empregar raramente todas as letras maiúsculas e nunca como um estilo de formatação.
28. Evitar usar inadequadamente espaços e pontuação para dar ênfase.

Revelando o conteúdo por meio de exemplos

29. Usar exemplos para revelar o conteúdo do site, em vez de apenas descrevê-lo.
30. Para cada exemplo, disponibilizar um *link* para acessar diretamente a página detalhada desse exemplo, em vez de saltar para uma página de categoria geral a que o item pertence.
31. Inserir um *link* para uma categoria mais abrangente, ao lado do exemplo específico.

32. Indicar claramente quais *links* conduzem a informações de acompanhamento sobre cada exemplo e quais *links* direcionam para informações gerais sobre a categoria como um todo.
33. Facilitar o acesso aos itens apresentados recentemente na *homepage*, como nas duas últimas semanas ou no mês anterior, fornecendo uma lista das últimas apresentações e inserindo itens recentes em arquivos permanentes.

Links

34. Diferenciar os *links* e torná-los fácil de visualizar.
35. Não usar instruções genéricas como “Clique aqui” como um nome de *link*.
36. Não usar *links* genéricos, como “Mais...”, no final de uma lista de itens.
37. Permitir *links* coloridos para indicar os estados visitados e não-visitados.
38. Não usar a palavra “*Links*” para indicar *links* existentes na página, indicar a presença de links com sublinhado e cor azul.
39. Se um link não fizer nada mais do que ir para outra página da *Web*, como vincular a um arquivo PDF ou acionar um equipamento de áudio e vídeo, aplicativo de mensagens de e-mail ou outro aplicativo, certificar-se de que o link indique explicitamente o que acontecerá.

Navegação

40. Alocar a área de navegação principal em um local bastante destacado, de preferência imediatamente ao lado do corpo principal da página.
41. Agrupar itens na área de navegação, de modo que os itens semelhantes fiquem próximos entre si.
42. Não disponibilizar diversas áreas de navegação para o mesmo tipo de *links*.
43. Não incluir um *link* ativo para a *homepage* na *homepage*.
44. Não inventar termos para as opções de navegação de categorias. As categorias devem ser diferenciáveis entre si. Se os usuários não entenderem sua terminologia inventada, não conseguirão distinguir as categorias.
45. Se existir um recurso de carrinho de compras em seu site, incluir um *link* para esse recurso na *homepage*.
46. Usar ícones na navegação somente se ajudarem aos usuários a reconhecer imediatamente uma classe de itens, como novos itens, itens em liquidação ou conteúdo do vídeo.

Pesquisa

47. Disponibilizar para os usuários uma caixa de entrada na *homepage* para inserir consultas de pesquisa, em vez de oferecer apenas um link para uma página de pesquisa.
48. As caixas de entrada devem ser suficientemente grandes para os usuários verem e editarem consultas padrão no site.

49. Não rotular a área de pesquisa com um título, em vez disso, usar o botão “*Search*” (Busca/Pesquisa), à direita da caixa.
50. A menos que as pesquisas avançadas sejam regra geral em tal site, forneça pesquisa simples na *homepage*, com um link para acessar a pesquisa avançada ou dicas de pesquisa, se existirem.
51. A pesquisa na *homepage* deve pesquisar o site inteiro, por *default*.
52. Não oferecer um recurso para “Pesquisar na *Web*” na função de pesquisa do site.

Ferramentas e atalhos para tarefas

53. Oferecer aos usuários acesso direto às tarefas de alta prioridade na *homepage*.
54. Não incluir ferramentas que não estejam relacionadas com as tarefas que os usuários costumam fazer no site.
55. Não oferecer ferramentas que reproduzem funções do navegador, como definir uma página como página inicial default do navegador ou marcar um site.

Gráficos e animação

56. Usar gráficos para apresentar o conteúdo real, não somente para decorar a *homepage*.
57. Rotular gráficos e fotos se os respectivos significados não estiverem claros no contexto da história que complementam.
58. Editar fotos e diagramas adequadamente, segundo o tamanho de exibição.
59. Evitar gráficos e marca d’água (imagens de plano de fundo com texto sobreposto).
60. Não usar a animação para o único propósito de chamar a atenção para um item na *homepage*. Raramente, a animação tem um local na página porque distrai a atenção voltada para outros elementos.
61. Jamais animar elementos críticos da página, como logotipo, slogan ou título principal.
62. Permitir que usuários decidam se desejam ver uma introdução animada de seu site – não deixar a opção de animação predefinida.

Design gráfico

63. Limitar os estilos de fonte e outros atributos de formatação de texto, como tamanhos, cores, etc., na página, porque o texto com *design* muito pesado pode se desviar do significado das palavras.
64. Usar texto com muito contraste e cores de plano de fundo, para que os caracteres fiquem o mais legíveis possível.
65. Evitar a rolagem horizontal.

- 66. Os elementos mais críticos da página devem estar visíveis “acima da dobra” (na primeira tela de conteúdo, sem rolar), no tamanho de janela mais predominante.
- 67. Usar um *layout* fluído para permitir o ajuste do tamanho da *homepage* a diversas resoluções de tela.
- 68. Usar logotipos criteriosamente.

Componentes da interface com o usuário

- 69. Nunca utilizar componentes da interface como parte da tela em que as pessoas não deverão clicar.
- 70. Evitar utilizar diversas caixas de entrada de texto na *homepage*, principalmente na parte superior da página em que as pessoas geralmente procuram o recurso de pesquisa.
- 71. Usar raramente menus suspensos, principalmente se os itens neles contidos não forem auto-explicativos.

Títulos de janela

- 72. Iniciar o título da janela com a palavra que resume a informação (geralmente o nome da empresa).
- 73. Não incluir o nome de domínio de nível superior como “.com”, no título da janela, a menos que faça realmente parte do nome da empresa, como em “Amazon.com”.
- 74. Não incluir a palavra *homepage* no título. É uma verbosidade sem importância.
- 75. Incluir uma descrição resumida do site no título da janela.
- 76. Limitar os títulos das janelas a não mais do que sete ou oito palavras e a menos de 64 caracteres.

URLs

- 77. As *homepages* para *Websites* comerciais devem ter o URL <http://www.empresa.com> (ou o equivalente em seu país ou no domínio de nível superior não-comercial).
- 78. Para qualquer *Website* que tenha uma identidade conectada a um país específico, diferente dos Estados Unidos, usar o domínio de nível superior desse país.
- 79. Se disponível, registrar os nomes de domínio com grafias alternativas, abreviações ou erros comuns do nome do site.
- 80. Se existirem grafias alternativas de nomes de domínio, selecionar uma delas como a versão autorizada e redirecionar os usuários para essa versão a partir de todas as demais grafias.

Notícias e comunicados à imprensa

81. Os títulos devem ser sucintos, mas descritivos, para transmitir o máximo de informações com o mínimo de palavras possível.
82. Escrever e editar sinopses específicas de comunicados à imprensa e das novas histórias apresentadas na *homepage*.
83. Vincular o título e não a sinopse à história completa da notícia.
84. Desde que todas as novas histórias da *homepage* tenham ocorrido dentro da semana, não há necessidade de listar a data e hora na sinopse de cada história, a não ser que seja realmente um item do noticiário de última hora, que tem atualizações frequentes.

Janelas pop-up e páginas intermediárias

85. Conduzir os usuários à “verdadeira” *homepage* quando digitarem a URL principal ou clicarem em um *link* para seu site.
86. Evitar janelas *pop-up*.
87. Não usar páginas de roteamento para os usuários selecionarem as respectivas localizações geográficas, a não ser que existam versões de seu site em diversos idiomas.

Publicidade

88. Manter os anúncios de empresas externas nas bordas da página.
89. Manter os anúncios externos (anúncios de empresas diferentes das suas) pequenos e o mais discretos possível em relação ao conteúdo central da *homepage*.
90. Se você posiciona anúncios fora da área de *banner* padrão, no início da página, rotule-os como publicidade, para que os usuários não os confundam com o conteúdo de seu site.
91. Evitar usar convenções para anúncios para acomodar recursos regulares do *site*.

Boas-vindas

92. Não dê boas-vindas aos usuários do site. Antes que você renuncie ao patrimônio da *homepage* principal e a utilize para dar saudações, experimente usá-la para um *slogan*.

Comunicando problemas técnicos e tratando de emergências

93. Se o *Website* ficar paralisado ou partes importantes do *Website* não estiverem funcionando, informar isso claramente na *homepage*.
94. Ter um plano para lidar com o conteúdo crítico do *Website*, para o caso de uma emergência.

Créditos

95. Não desperdiçar espaço com créditos relacionados ao mecanismo de pesquisa, empresa de *design*, empresa do navegador favorito ou com a tecnologia utilizada por trás dos bastidores.
96. Limitar a exibição de prêmios recebidos por seu *Website*.

Recarregamento e atualização de página

97. Não atualizar automaticamente a *homepage* para acionar atualizações para os usuários.
98. Ao fazer uma atualização, atualizar somente o conteúdo realmente modificado, como as atualizações de notícias.

Personalização

99. Se a sua *homepage* tiver áreas que fornecerão informações personalizadas assim que você souber algo sobre o usuário, não disponibilize uma versão genérica do conteúdo para os novos usuários – crie um conteúdo diferente para esse espaço.
100. Não disponibilizar para os usuários recursos para personalizar a aparência básica da interface com o usuário da *homepage*.

Obtendo dados do cliente

101. Não fornecer *links* para registro na *homepage*, em vez disso, explique (ou, pelo menos, faça uma associação) as vantagens do registro para o cliente.
102. Explicar para os usuários os benefícios e a frequência de publicação, antes de solicitar seus endereços de *e-mail*.

Favorecendo uma comunidade

103. Se existir um suporte para comunidades de usuários com bate-papo (*chat*) ou outros recursos para discussão, não apresentar *links* genéricos para esses recursos.
104. Não oferecer uma entrada em “Livro de Visitantes” para *sites* comerciais.

Datas e horas

105. Mostrar datas e horas somente para informações relacionadas ao tempo, como itens de notícias, bate-papos ao vivo, cotações de ações e outros itens.
106. Mostrar aos usuários a hora da última atualização do conteúdo, não a hora atual gerada pelo computador.
107. Incluir o fuso horário, sempre que fizer referência a uma hora.
108. Usar abreviações padrão, como p.m ou P.M.
109. Usar o nome do mês inteiro ou abreviações, mas não números.

Cotações de ações e exibição de números

- 110. Fornecer a porcentagem de mudança, não apenas os pontos ganhos ou perdidos em cotações de ações.
- 111. Explicar as abreviações das ações, a não ser que a abreviação seja totalmente explícita, como “IBM”.
- 112. Usar um separador de milhares adequado à sua localidade, para os números com cinco ou mais dígitos.
- 113. Alinhar os pontos decimais ao exibir colunas de números.