

UNIVERSIDADE FEDERAL DE SANTA CATARINA  
PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA DE  
PRODUÇÃO E SISTEMAS (PPGEP / UFSC)

Marcelo Morandini

**Ergo-Monitor: Monitoramento da Usabilidade  
em Ambiente Web por Meio  
da Análise de Arquivos de Log**

Tese apresentada ao Programa de Pós-Graduação  
em Engenharia de Produção da  
Universidade Federal de Santa Catarina  
como requisito para a obtenção do  
título de Doutor Engenharia de Produção e Sistemas

**Orientador:** Prof. Dr. Walter de Abreu Cybis

**Florianópolis**

**Novembro de 2003**

Ficha Catalográfica

Morandini, Marcelo

Ergo-Monitor: Monitoramento da Usabilidade em Ambiente Web por meio da Análise de Arquivos de Log

Xxx págs

Orientador: Prof. Dr. Walter de Abreu Cybis

Tese (doutorado) – Universidade Federal de Santa Catarina, Programa de Pós-Graduação em Engenharia de Produção e Sistemas.

1. Avaliação da Usabilidade de Sistemas Interativos; 2 - Interações Humano-Computador; 3 – Web Sites; 4 – Arquivos de Log.

**Marcelo Morandini**

**Ergo-Monitor: Monitoramento da Usabilidade  
em Ambiente Web por meio da Análise de Arquivos de Log**

Esta tese foi  **julgada e aprovada**  para a obtenção do título de Doutor em Engenharia de Produção e Sistemas junto ao PPGEPP – UFSC

Florianópolis, 28 de Novembro de 2003

**Banca Examinadora:**

Prof. Dr. Walter de Abreu Cybis (orientador) \_\_\_\_\_  
Universidade Federal de Santa Catarina (assinatura)

Prof. Dra. Heloísa Vieira Rocha \_\_\_\_\_  
Universidade Estadual de Campinas (assinatura)

Prof. Dr. Marcelo Soares Pimenta \_\_\_\_\_  
Universidade Federal do Rio Grande do Sul (assinatura)

Prof. Dr. Neri dos Santos \_\_\_\_\_  
Universidade Federal de Santa Catarina (assinatura)

Prof. Dra. Patrícia Vilain \_\_\_\_\_  
Universidade Federal de Santa Catarina (assinatura)

## **Dedicatória**

**Aos meus pais, minha esposa Luciana  
e minha filha Rafaela**

## Agradecimentos

Ao meu **orientador**, Prof. Dr. Walter de Abreu Cybis, pela indispensável orientação na realização deste trabalho, pela confiança dispensada, pelo incentivo e apoio sempre prestados e por todas as oportunidades oferecidas, que tanto contribuíram para minha formação acadêmica.

Ao meu **amigo** Walter de Abreu Cybis, pelo acolhimento em sua família nas minhas idas a Florianópolis e por todas as conversas agradáveis que tivemos oportunidade de ter. A Rose e ao Pedro por terem me recebido tão bem em sua casa e tratarem como um membro da família.

A Profa. Dra. Heloísa Rocha, não apenas pela sua participação como membro da banca deste trabalho, mas por todas as sugestões fornecidas ao longo do desenvolvimento do mesmo, que muito contribuíram para minha formação. Também pela oportunidade de trabalhar junto ao grupo de pesquisadores do NIED – Unicamp.

Aos Profs. Drs. Marcelo Soares Pimenta, Patrícia Vilain e Neri dos Santos, componentes da banca examinadora deste trabalho.

Ao Prof. Dr. Ivan Rizo Guilherme, do DEMAC - Unesp, campus de Rio Claro, pela amizade por ter fornecido condições para que eu pudesse trabalhar no desenvolvimento das atividades desta tese em seu laboratório. Ao amigo Marcus Vinícius Maltempi, também professor do DEMAC – Unesp, campus de Rio Claro, por ter me apoiado em vários momentos, inclusive, como um dos usuários que participaram dos ensaios de interação que constituíram uma das atividades deste trabalho.

Ao amigo e orientando Ângelo Ferecini Neto, pelo apoio nas atividades de programação envolvidas nesta tese, pela solicitude e enorme capacidade que sempre demonstrou.

Aos colegas professores e funcionários do DIN – UEM que sempre me apoiaram para que eu chegasse à conclusão de mais uma etapa da minha vida e muito colaboraram no desenvolvimento desta tese e na minha adaptação à UEM e à cidade de Maringá. Ainda, pelo apoio dos professores do DIN que estão com cargas letivas maiores para que alguns professores, como eu, possam estar afastados de suas atividades docentes para se dedicarem única e exclusivamente às atividades de pesquisa.

Aos funcionários da secretaria do PPGEP, em especial à Neiva, pela amizade e excelente auxílio prestado.

Aos colegas dos laboratórios em que desenvolvi minhas pesquisas e realizei as implementações computacionais necessárias ao desenvolvimento desta tese. Para mim, foi uma felicidade poder atuar junto aos grupos de pesquisadores e profissionais do LabIUtil da UFSC, do NIED da Unicamp e do GIA da Unesp – Rio Claro. Obrigado pelas discussões e sugestões tão importantes, pelo auxílio prestado na condução de diversos experimentos e pela amizade conquistada.

A CAPES e à UEM pelo apoio financeiro.

À minha esposa Luciana pelo incentivo constante e por me mostrar diariamente exemplos de dedicação, capacidade e paixão, não apenas pela família, mas também pelo trabalho. E em especial, pela sua exuberante capacidade de conseguir unir essas duas atividades e cumpri-las tão bem. À minha filha Rafaela, pela oportunidade da paternidade e todas as alegrias que me proporciona a cada dia.

Aos meus pais, Arcênio e Ivete, pelo apoio e incentivo incondicional em todos os momentos de minha vida e serem grandes exemplos de dedicação e trabalho. Aos meus sogros, Marcos e Ana, por me mostrarem as alegrias e responsabilidades de ser um educador.

Aos meus irmãos, Mateus e Giovana (e à minha sobrinha Amanda). Aos meus cunhados, Cleber, Júnia, Lucimara, Daniel e Juliana. Às tias Tata, Tita, Anete e Jane. Obrigado pelos bons momentos vividos e por todas as palavras de incentivo e carinho.

## Resumo

Os projetistas de *web sites* são continuamente confrontados com um desafio: tornar seus *web sites* atraentes aos usuários e garantir que estes sejam capazes de realizar suas interações de maneira fácil, eficiente e agradável. Em muitos casos, a rentabilidade financeira das empresas que utilizam a *internet* como um meio de difusão, ou até mesmo de comércio, de seus produtos está fortemente dependente da qualidade das interfaces humano-computador desenvolvidas. Nesse contexto, uma técnica de avaliação da usabilidade de *web sites* tem um importante papel, que é o de fornecer aos avaliadores as melhores definições para que estes sejam capazes de influir no ciclo de desenvolvimento da usabilidade dos produtos avaliados. Este trabalho propõe a definição e implementação de um protótipo para um Sistema de Monitoramento (batizado por Ergo-Monitor) para a coleta de informações que possam nortear a elaboração de laudos quantitativos acerca da usabilidade de *web sites* (com ênfase em sistemas de comércio eletrônico). O ambiente Ergo-Monitor baseia-se na coleta de dados diretamente das interações dos usuários que se encontram armazenados nos *server log files*. Após essa coleta, as ferramentas do ambiente fazem o tratamento desses dados produzindo taxas e métricas de usabilidade que devem ser, na seqüência, confrontadas pelo avaliador com valores característicos de boas e más interações para *web sites* que estejam na mesma categoria do *web site* sendo avaliado. O ambiente Ergo-Monitor apóia uma interação produtiva entre laudos quantitativos e qualitativos de avaliação de usabilidade de *web sites* e os laudos quantitativos produzidos pelo Ergo-Monitor podem vir a confirmar diagnósticos qualitativos elaborados previamente, ou vir a suscitar novas avaliações qualitativas da usabilidade dos *web sites* monitorados. Acima de tudo, o Ergo-Monitor está sendo proposto como um ambiente de avaliação “invisível” aos usuários e cujos tratamentos dos dados coletados sejam feitos de forma objetiva, automática e econômica. Assim, seus resultados não são influenciados por ruídos nas observações, como a presença de avaliadores e equipamentos, os usuários não se sentem eles mesmos o foco da avaliação e os custos de uma avaliação *in loco* e *a posteriori* podem ser minimizados.

**Palavras-Chave:** Avaliação da Usabilidade de Sistemas Interativos; Interações Humano-Computador; Web Sites; Arquivos de Log

## Abstract

Web Site designers are regularly dealing with a challenge: making their web sites attractive to the users and guarantee that the users can be able to interact easily, efficiently and with satisfaction. In several cases, the financial profitability of companies that use internet as a merchandising or even e-commerce way is strongly dependant on the quality of the human-computer interfaces developed. So, a web site usability evaluation technique has an important role in this context, that is supporting the evaluators with the best definition so they can be able to influence in the evaluated products' usability development cycle. This work proposes the definition and a prototype implementation of a Monitoring System (named as Ergo-Monitor) to collect informations that can be useful to the elaboration of quantitative reports concerning the usability of *web sites* (with emphasis in e-commerce systems). The Ergo-Monitor environment is based on the collection of data done directly from the users' interactions stored at the server log files. After this collection, this environment tool will perform the treatment of these data generating usability taxes and metrics that must be, in a next step, compared by the evaluator with values that define good and bad interactions of web sites that are in the same category of the one being evaluated. The quantitative reports generated by the Ergo-Monitor can confirm the qualitative diagnosis previously produced, or may ask new qualitative evaluations for the monitored web sites. Above all, the Ergo-Monitor is being proposed as an evaluation environment "invisible" by the users, where the data collected treatment is done in an objective, automatic and economic way. So, the results produced are not influenced by noises in the observations such as evaluators and equipments presence, the users may not be considering themselves the evaluation target and the costs of "in loco" and postponed evaluations can be minimized.

**Keywords:** Interactive Systems Usability Evaluation; Human-Computer Interaction; Web Sites; Log Files

# Sumário

	Página
1 - Introdução	1
1.1 - Contextualização do Problema	1
1.2 - Objetivos	8
1.2.1 - Objetivos Específicos	9
1.3 - Justificativas do Projeto	9
1.4 - Divisão dos Capítulos	12
2 - Revisão Bibliográfica	13
2.1 - Conceitos Básicos sobre Engenharia de Usabilidade	19
2.2 - Norma ISO/IEC 9241 - parte 11	21
2.2.1 - Objetivos da Norma ISO/IEC 9241 - parte 11	22
2.2.1.1 - Medindo Eficácia	24
2.2.1.2 - Medindo Eficiência	24
2.2.1.3 - Medidas de Satisfação	25
2.2.2 – Características da Norma ISO/IEC que Motivaram a Determinação das Taxas	25
2.3 - Questionários (Checklists)	27
2.4 - Avaliação Heurística	30
2.4.1 - Heurísticas de Usabilidade (Nielsen)	30
2.4.2 - Heurísticas para a Web (Instone)	32
2.4.3 - Critérios Ergonômicos (Scapin)	33
2.5 - Observação do Usuário	37

2.6 - Técnica de Co-Inspeção Avaliativa	39
2.6.1 - Descrição do Método, Documentos e Ferramentas	40
2.6.2 - Etapas do Método	40
2.6.3 - Reconhecimento-Descrição Assistida Global	41
2.6.4 - Entrevista com os Projetistas	41
2.6.5 - Entrevista com os Usuários	42
2.6.6 - Integração de Cenários	43
2.6.7 - Reconhecimento-Descrição Assistida Local	43
2.6.8 - Inspeção Analítica	44
2.6.9 - Configuração das Listas de Verificação	44
2.6.10 - Realização das Inspeções Avaliativas	44
2.7 – A Avaliação de Usabilidade na <i>Web</i>	45
2.7.1 - Algumas Ferramentas Automatizadas	47
2.7.2 - Ferramentas de Destaque	52
2.7.2.1 - WebSAT	52
2.7.2.2 - Lift	53
2.7.2.3 - Max WebCriteria	53
2.7.2.4 - Bobby	54
2.7.2.5 - Considerações Gerais Sobre Ferramentas Automatizadas	55
3 – Ferramentas e Estruturas Empregadas no Ergo-Monitor	57
3.1 - Sistemas de Comércio Eletrônico	58
3.1.1 - Servidores de Comércio Eletrônico	59
3.2 - Server Log Files	60
3.3 - Cookies	63

3.4 – Algumas Características de Ferramentas de Análise de Log Files	65
3.4.1 - A Ferramenta Fast Stats Analyzer	66
4 - Definição da Estrutura do Ambiente Ergo-Monitor	74
4.1 - Motivação da Proposição do Ergo-Monitor	75
4.1.1 - Principais Dificuldades para o Emprego de Ferramentas Automatizadas	76
4.2 - O Ambiente Ergo-Monitor	78
4.2.1 - Etapas de Aplicação do Ergo-Monitor	81
4.2.1.1 – Análise para o Monitoramento	85
4.2.1.2 – Filtragem dos Dados de <i>Log</i> e Determinação Comportamentos Verificados	87
4.2.1.3 – Cálculo das Taxas de Usabilidade	89
4.2.1.4 – Elaboração de Laudos	92
4.2.1.5 – Montagem da Base de Parâmetros Admissíveis para Usabilidade	92
4.2.2 – O Log Control	98
4.2.2.1 - Filtro.java	99
4.2.2.2 - Busca.java	101
4.2.2.3 - Buscaseq.java	102
4.2.2.4 - Buscatar.java	103
4.3 - Validação do Ergo-Monitor	111
4.3.1 - O Web Site de Comércio Eletrônico da Ruth Fraldas	112
4.3.2 - Avaliação da Usabilidade do Web Site da Ruth Fraldas	118
4.4 - Considerações sobre o Ergo-Monitor	122
5 – Conclusões e Trabalhos Futuros	125
5.1 – Conclusões	125
5.2 – Atividades Futuras	127

5.2.1 – Montagem de Base de Valores Admissíveis para uma Boa Usabilidade	127
5.2.2 – Definição e Implementação do Monitoramento do Lado Cliente	127
5.2.3 – Caracterização dos Usuários Independentemente de Seu Endereço IP	128
5.2.4 – Realizar Manutenções nos Programas do Log Control	129
5.2.5 – Desenvolvimento do Modelo de Tarefas Fechadas e Objetivas	130
6 – Referências Bibliográficas	131
Apêndice A - Listagem dos Programas do Log Control	136
A.1 - Filtro.Java	136
A.2 - Buscar.Java	138
Apêndice B - Códigos-Fonte das Páginas do Web Site da Ruth Fraldas	143
Apêndice C - Arquivos-Texto de Entrada dos Programas do Log Control	146
Apêndice D - Tabelas (Excel) Elaboradas Com Dados Observados Em Interações Para Apoiar a Determinação dos Parâmetros Admissíveis	156
Apêndice E - Resultados Obtidos para o Monitoramento de Tarefas Empregando-se o Log Control	163

## Lista de Figuras

Figura 2.1 - Fase de Co-Descrição Avaliativa	42
Figura 2.2 - Fase de Avaliação Inspetiva	43
Figura 3.1 - Formato de um Registro de Log File	62
Figura 3.2 - Exemplo de Log File	62
Figura 3.3 - Tela Inicial da ferramenta Fast Stats Analyzer	67
Figura 3.4 - Imagem da Ferramenta depois da Coleta dos Dados	68
Figura 3.5 - Número de Acessos	71
Figura 3.6 - Mensagens de Erro	71
Figura 3.7 - Caminhos Acessados e Quantidades de Acessos por Componentes	72
Figura 3.8 - Relacionamento entre Dias da Semana e Número de Acessos	73
Figura 4.1 - Legenda das Estruturas Apresentadas na Descrição do Ergo-Monitor	82
Figura 4.2 – Visão Geral do Ergo-Monitor	84
Figura 4.3 - Descrição do Log Control	89
Figura 4.4 - Tela do Programa Buscarar.java	106
Figura 4.5 - Seqüência Páginas para a Determinação de Taxas Relativas ao Cadastro	107
Figura 4.6 - Seqüência Páginas para a Determinação Taxas Relativas Busca Avançada	108
Figura 4.7 - Resultado do Programa Buscarar.Java para a Busca Avançada	109
Figura 4.8 - Planilha Excel que Apóia a Determinação de Taxas para a Atividade de Cadastro	110
Figura 4.9 - Tela Inicial do Cadastro de Usuários	114
Figura 4.10 - Tela de Mensagem de Erro no Cadastro	114

Figura 4.11 - Tela Inicial da Busca Avançada	115
Figura 4.12 - Tela Inicial das Compras	116
Figura 4.13 - Tela de Requisição da Forma de Pagamento	117
Figura 4.14 - Tela de Confirmação das Compras	118
Figura A.1 - Código-Fonte do Programa filtro.java	137
Figura A.2 - Código-Fonte do Programa buscar.java	142
Figura B.1 - Código-Fonte do Programa index.php	144
Figura B.2 - Código-Fonte do Programa login.php	145
Figura C.1 - Definição das Páginas Acessadas para a Tarefa Busca Avançada	147
Figura C.2 - Definição das Páginas Acessadas para a Tarefa Busca Simples	148
Figura C.3 - Definição das Páginas Acessadas para a Tarefa Cadastro	149
Figura C.4 - Definição das Páginas Acessadas para a Tarefa Fale Conosco	150
Figura C.5 - Definição das Páginas Acessadas para a Tarefa Trocar Senha	151
Figura C.6 - Definição das Páginas Acessadas para a Tarefa Compras (escolha dos produtos)	152
Figura C.7 - Definição das Páginas Acessadas para a Tarefa Compra (Ir para o Caixa)	153
Figura C.8 - Definição das Páginas Acessadas para a Tarefa Compra (forma pagamento - Cartão)	154
Figura C.9 - Definição das Páginas Acessadas para a Tarefa Compra (forma pagamento - depósito bancário)	155
Figura E.1 - Planilha Excel que Apóia a Determinação de Taxas para a Atividade de Busca Avançada	165
Figura E.2 – Planilha Excel que Apóia a Determinação de Taxas para a Atividade de Busca Simples	166
Figura E.3 – Planilha Excel que Apóia a Determinação de Taxas para a Atividade de Compras – Definição do Usuário Comprador ( <i>login</i> )	167

Figura E.4 – Planilha Excel que Apóia a Determinação de Taxas para a Atividade de Compras – Ir para o Caixa	168
Figura E.5 – Planilha Excel que Apóia a Determinação de Taxas para a Atividade de Compras – Forma de Pagamento: Depósito	169
Figura E.6 – Planilha Excel que Apóia a Determinação de Taxas para a Atividade de Compras – Forma de Pagamento: Cartão de Crédito	170
Figura E.7 – Planilha Excel que Apóia a Determinação de Taxas para a Atividade de Compras – Fale Conosco	171
Figura E.8 – Planilha Excel que Apóia a Determinação de Taxas para a Atividade de Compras – Troca de Senha	172

## Lista de Tabelas

Tabela 2.1 - Exemplo de Medidas de Usabilidade [ISO 9241 - parte 11]	26
Tabela 2.2 - Exemplos de Medidas para Propriedades Desejáveis [ISO 9241 - parte 11]	27
Tabela 4.1 - Valores de Referência para a Busca Avançada no Cadastro	94
Tabela 4.2 - Valores de Referência para o Enfrentamento de Erros no Cadastro	95
Tabela 4.3 - Valores de Referência para a Busca Avançada nas Compras	96
Tabela 4.3 - Valores de Referência para os Enfrentamentos de Erros nas Compras	97
Tabela 4.5 – Definição Inicial dos Parâmetros Admissíveis	98
Tabela 4.6 – Resultados Obtidos pelo Ergo-Monitor – Tarefa de Cadastro	120
Tabela 4.7 - Resultados Obtidos da Análise das Gravações – Tarefa de Cadastro	121
Tabela D.1 - Tempo (em segundos) para Realizar a Tarefa Busca Avançada	157
Tabela D.2 - Quantidade de Acessos ao Help para Realizar a Tarefa Busca Avançada	157
Tabela D.3 - Quantidade de Mensagens de Erro Apresentadas na Tarefa de Busca Avançada	157
Tabela D.4 - Tempo (em segundos) para Realizar a Tarefa Busca Simples	158
Tabela D.5 – Quantidade de Acessos ao Help para Realizar a Tarefa Busca Simples	158
Tabela D.6 - Quantidade de Mensagens de Erro Apresentadas na Tarefa Busca Simples	158
Tabela D.7 - Tempo (em segundos) para Realizar a Tarefa Cadastro	159
Tabela D.8 - Quantidade de Acessos ao Help para Realizar a Tarefa Cadastro	159
Tabela D.9 - Quantidade de Mensagens de Erro Apresentadas na Tarefa Cadastro	159
Tabela D.10 - Tempo (em segundos) para Realizar a Tarefa Compra	160
Tabela D.11 - Quantidade de Acessos ao Help para Realizar a Tarefa Compra	160
Tabela D.12 - Quantidade de Mensagens de Erro Apresentadas na Tarefa Compra	160
Tabela D.13 - Tempo (em segundos) para Realizar a Tarefa Fale Conosco	161

Tabela D.14 - Quantidade de Acessos ao Help para Realizar a Tarefa Fale Conosco	161
Tabela D.15 - Quantidade de Mensagens de Erro Apresentadas na Tarefa Fale Conosco	161
Tabela D.16 - Tempo (em segundos) para Realizar a Tarefa Login	162
Tabela D.17 - Quantidade de Acessos ao Help para Realizar a Tarefa Login	162
Tabela D.18 - Quantidade de Mensagens de Erro Apresentadas na Tarefa Login	162
Tabela D.19 - Tempo (em segundos) para Realizar a Tarefa Troca Senha	163
Tabela D.20 - Quantidade de Acessos ao Help para Realizar a Tarefa Troca Senha	163
Tabela D.21 - Quantidade de Mensagens de Erro Apresentadas na Tarefa Troca Senha	163

# 1 – Introdução

## 1.1 – Contextualização do Problema

Os engenheiros de software e os projetistas de interfaces devem se defrontar certamente com o problema de avaliar e redefinir a usabilidade de sistemas baseados na *internet*. Entre suas características particulares, tais sistemas apresentam, de maneira geral, Interfaces Humano-Computador (IHCs) em evolução constante sendo que estas podem facilmente serem analisadas como os próprios *web sites*.

A principal qualidade em uma interface humano-computador é a sua *usabilidade*, que estabelece o quanto os sistemas são projetados de forma a serem fáceis de utilizar e aprender [Nielsen, 2000]. Deve-se pensar na usabilidade em termos de qualidade de uso que um sistema interativo apresenta para seus usuários (pretendidos) de modo a que possam atingir um conjunto específico de objetivos e tarefas em um ambiente particular de trabalho [ISO 9241]. A usabilidade é, ou deveria ser, o foco central no desenvolvimento de uma IHC, e é, por intermédio dela, que se tenta eliminar a distância entre os objetivos do usuário e as possibilidades do sistema informatizado. A usabilidade, conforme aponta a Norma ISO/IEC 9241, pode ser definida como um recorte de métricas derivadas das interações, entre elas: a efetividade, a eficiência e a satisfação com que cada usuário específico alcança suas metas definidas em ambientes particulares. *Efetividade* significa a precisão e perfeição com que usuários podem alcançar metas desejadas em ambientes particulares, *eficiência* significa os recursos gastos em relação à precisão e perfeição de metas alcançadas, e *satisfação* significa o conforto e aceitabilidade do sistema de trabalho para seus usuários e outras pessoas afetadas por seus usos [ISO 9241]. A parte 11 da norma ISO 9241 fornece indicações sobre como quantificar a usabilidade a partir de métricas de eficácia, eficiência e satisfação, principalmente, e sugere também outras métricas como intuitividade, facilidade de aprender, facilidade de lembrar, etc.

Sob a perspectiva dos usuários, a usabilidade é importante, pois afeta diretamente sua produtividade, seu desempenho e sua carga de trabalho. Para os projetistas de IHC, a usabilidade, ou a falta dela, pode significar sucesso ou fracasso em seu trabalho. A falta de

usabilidade tem custos elevados tanto para a empresa que desenvolve como para a que adquire um sistema interativo [Nielsen, 2000].

A *Engenharia de Usabilidade* visa favorecer o desenvolvimento de sistemas com boa usabilidade através da aplicação estruturada e sistemática de diferentes métodos em diversos estágios do processo de projeto (*design*) e desenvolvimento dos sistemas interativos. As técnicas de Engenharia de Usabilidade direcionam os projetistas a focarem mais os usuários do que a tecnologia [Sears, 1997]. O entendimento das necessidades dos usuários e das tarefas que eles estão desejando realizar está se tornando um conhecimento cada vez mais importante quando comparado unicamente com o associado às últimas novidades tecnológicas [Shneidermann, 1992].

Sempre resta a dúvida do quão bem projetadas são as interfaces de *web sites*, e para tanto, métodos que favoreçam uma boa avaliação devem ser empregados. No ciclo de desenvolvimento da usabilidade, a avaliação busca apontar resultados relativos à satisfação de uso por parte dos usuários e os pontos positivos ou negativos da interação no que se refere a diversas características, tais como: acessibilidade, funcionalidade, adaptatividade, entre outras [Shneidermann, 1992].

Em estágios iniciais do processo de projeto, a avaliação de usabilidade é usada para selecionar projetos alternativos de interface com o usuário e, então, identificar o projeto preferível. Na seqüência dos desenvolvimentos, a avaliação de usabilidade é realizada para determinar se o projeto alcança os requisitos de usabilidade estabelecidos [ISO 9241-11], [Lacerof and Paternó, 1998].

Em particular, a análise da usabilidade de *web sites* que envolvem atividades comerciais pode ser sempre vista como uma estratégia financeira, uma vez que o desempenho das empresas envolvidas pode estar fortemente vinculado à facilidade com que os usuários realizem suas interações com o *web site*.

Antes de uma avaliação ser realizada, é importante conhecer qual é o objetivo dessa avaliação. Preece em [Preece et alli. 1994] identifica os principais objetivos de uma avaliação de usabilidade sob os seguintes eixos:

- Engenharia: o projeto é suficientemente bom?
- Comparação: qual é o melhor projeto?
- Entendimento do mundo real: quão bem o projeto funciona em aplicações reais?

Algumas técnicas de avaliação da usabilidade de sistemas interativos encontradas na literatura [Downton, 1992], [Lea, 1988], [Treu, 1994], [Sears, 1997] abordam os aspectos qualitativos da interface avaliada, e não chegam a ser suficientemente objetivas para quantificar a eficiência, a eficácia e a satisfação do usuário ao utilizar a interface. Tal observação indica que devem ser desenvolvidas pesquisas que busquem apresentar características quantitativas a respeito da usabilidade dos sistemas interativos, sendo que mais particularmente, neste trabalho, o de *web sites*.

Os *web sites* tornam-se “obsoletos” por razões externas como, quando uma nova característica de interação se torna disponível e outros *web sites* que a utilizem se tornam mais desejados que os que não a empreguem. Um *web site* que esteja de acordo com o “estado-da-arte” ontem, pode estar ultrapassado e não mais ser chamativo aos usuários [Nielsen, 2002]. Assim, o tempo de desenvolvimento de um *web site* é encurtado, uma vez que os ciclos de projeto (*design*), teste de usabilidade e iteração do projeto são difíceis de serem implantados em escalas de tempo reduzidas [Scholtz, 2001]. Trazer os conceitos, metodologias e técnicas relativas à avaliação da usabilidade para o contexto do desenvolvimento de *web sites* e de aplicações *web* é uma tarefa bastante difícil.

A usabilidade na *World Wide Web* pode significar não apenas a satisfação do usuário, mas a própria sobrevivência financeira da empresa. Jakob Nielsen indaga porque os usuários deveriam desperdiçar seu tempo em *sites* confusos, lentos ou que não satisfaçam às suas necessidades uma vez que existem diversos *web sites* similares facilmente encontráveis na *web* [Nielsen, 2002]. Tal afirmativa torna-se irrefutável ao se analisar a quantidade de *sites.com* existentes atualmente no mercado [Netcraft, 2003]. Como resultado dessa quantidade enorme de opções e da facilidade de se fazer o acesso a outros sites, os usuários da *web* demonstram uma notável impaciência e insistência na gratificação instantânea. Segundo Nielsen, se os usuários não conseguem descobrir como interagir com um *web site* em aproximadamente um minuto, eles concluem que não vale a pena gastar mais seu tempo nesse acesso [Nielsen, 2000].

De maneira geral, os desenvolvedores de softwares colocam à disposição dos usuários um centro de suporte voltado a atender problemas de usabilidade. Entretanto, diferentemente do que ocorre para produtos de software, tal quadro normalmente não é verdadeiro para os *web sites*, pois o usuário não é uma pessoa que tenha comprado o *web site*, tal como ocorre com um produto de software. Neste caso, os usuários experimentam a usabilidade de um *web site* antes mesmo de se comprometerem a usá-lo e mesmo antes de gastarem dinheiro com possíveis aquisições. Esta equação é bastante simples [Nielsen, 2000]:

- no projeto de produtos genéricos, ou mesmo de produtos de software, os clientes pagam primeiro e experimentam a usabilidade depois;
- na *web*, os usuários experimentam a usabilidade primeiro, para virem a pagar depois.

Além disso, a barreira competitiva na *web* é bastante alta. Tal concorrência não se limita apenas às outras empresas de seu setor. Com todos os milhões de *web sites* existentes, há uma concorrência pelo tempo e pela atenção dos usuários.

Os métodos para avaliação de usabilidade podem ter várias finalidades, como por exemplo: auxiliar em futuros projetos da interface (do *web site*) para atingir as necessidades do usuário; identificar e diagnosticar problemas; e avaliar a implementação (para comparações com outros projetos e sistemas, e também para o teste de aceitação). Os dados coletados durante essas avaliações podem ser qualitativos (descrições que qualifiquem a usabilidade) ou quantitativos (medidas que a quantifiquem).

Em uma perspectiva qualitativa de avaliação de usabilidade, a procura principal deve ser a satisfação das necessidades dos usuários, sendo que tais necessidades podem estar sendo representadas de forma explícita e/ou implícita [ISO 9126]. Elas se referem tanto às restrições comportamentais e sociais dos usuários dentro do contexto ambiental que o mesmo se encontra, quanto aos padrões esperados para a categoria em que o *web site* se encontra. Proporcionam diagnósticos fundamentalmente comparativos entre as observações realizadas nas avaliações e os padrões esperados para usabilidade. Já a perspectiva quantitativa se refere aos valores que podem ser obtidos através de medições da interação entre usuários e os componentes de um *web site*, em termos de eficiência, eficácia,

satisfação, intuitividade, facilidade de aprender, facilidade de lembrar, etc [Shneidermann, 1992].

Diversas técnicas de avaliação qualitativas são apresentadas na literatura, entre elas as avaliações por especialistas (avaliações heurísticas), as inspeções por *checklists* e os testes de interação.

As avaliações heurísticas representam a técnica mais conhecida e mais utilizada nas avaliações de usabilidade. Baseados em seus padrões de qualidade (muitas vezes implícitos), construídos por meio de experiência prática e conhecimento teórico, os avaliadores realizam um julgamento de valor da usabilidade de uma interface e de seus componentes. Em geral, as avaliações heurísticas são reveladoras de uma grande quantidade e variedade de diagnósticos de problemas (quanto ao nível de severidade). Muitas vezes, em seu aprendizado, os avaliadores valem-se de grades de análise que os orientam quanto aos aspectos a verificar na interface e no seu contexto de uso.

Dentre as grades de heurísticas conhecidas, destacam-se os critérios ergonômicos definidos por Scapin e Bastien [Scapin, 1997]. Tais critérios são apresentados como uma forma de se fornecer diretrizes para que os projetistas elaborem interfaces ergonomicamente corretas. Scapin e Bastien mostraram que o emprego deste conjunto de critérios pode trazer resultados satisfatórios, em termos de interfaces que não causem constrangimentos e aborrecimentos ao usuário devido a dificuldades de utilização, erros de manuseio, problemas de *feedback* ou, até mesmo, desistência de utilização. Os critérios ergonômicos representam as qualidades mínimas que um sistema interativo (IHC ou *web site*) deve ter. Eles provaram ser completos (cobrindo os *guidelines* correntes), distintos (independentes um do outro) e aplicáveis (usáveis em diferentes situações) [Scapin, 1997]. Além disso, eles se mostraram suficientemente bons em revelar um casamento um-para-um na relação problema x critério. Em alguns experimentos conduzidos [Bastien, 1998], todos os problemas diagnosticados foram realmente relacionados a um critério específico. Além disso, nenhum critério foi rejeitado ou classificado como inapropriado ou dependente do contexto. Ainda, os critérios ergonômicos melhoraram o desempenho das avaliações e se mostraram melhores que os princípios de diálogo propostos na Norma ISO/IEC 9241 [ISO 9241]. Os critérios ergonômicos podem, ainda, apoiar a organização do conhecimento sobre

a usabilidade e a criação de *checklists* empregáveis na avaliação de *web sites*, tal como os propostos no ambiente Ergo-CoIn [Cybis et alli, 2002] para o apoio à avaliação de *web sites*.

Uma avaliação por *checklists* pode ser definida como uma técnica diagnóstica, que tem por finalidade comparar as características de usabilidade de um sistema com padrões qualitativos explícitos. Como um exemplo de *checklist* que vem sendo bastante empregado no Brasil podemos citar o ERGO-LIST, que está disponível livremente na Internet ([www.labiutil.inf.ufsc.br/ergolist](http://www.labiutil.inf.ufsc.br/ergolist)). Dentre as vantagens que uma avaliação por *checklist* proporciona temos a sistematização da metodologia (uma vez que as mesmas questões podem ser apresentadas a diversos avaliadores); a facilidade de aplicação, uma vez que o *checklist* pode ser respondido pelo avaliador na forma e horário que lhe for mais conveniente; a possibilidade de se produzir uma classificação (*ranking*) final da interface como um todo, entre outras. Já como desvantagens, podemos destacar a baixa efetividade em termos de problemas mais sérios que dificilmente podem ser descobertos com o uso dos *checklists*. De fato, os problemas detectados por meio de *checklists* se apresentam como sendo relativamente superficiais e bastante gerais [Preece et alli. 1994]. Existe, ainda, uma carga de trabalho muito repetitiva e maçante por parte do avaliador que precisa responder às diversas questões formuladas.

Já as técnicas baseadas na observação de interações reais, os chamados testes de usabilidade, são conhecidos por sua efetividade, isto é, pela capacidade de identificar problemas reais que se manifestam claramente (não são diagnósticos) durante a interação entre usuário e sistema. Elas representam uma maneira de se comprovar os diagnósticos qualitativos produzidos por avaliações heurísticas ou pela aplicação de *checklists*, principalmente quando estes diagnósticos forem indicadores de problemas importantes que possam prejudicar significativamente a interação. Um teste de usabilidade permite também a obtenção de métricas quantitativas de usabilidade, como as propostas pela metodologia MUSiC [Macleod et alli, 1997] que analisa algumas características tais como a efetividade, eficácia, produtividade, entre outras. Os testes de usabilidade de softwares tradicionais são aplicados para “usuários típicos” realizando “tarefas típicas” e executando o software em uma plataforma típica.

Entretanto, as populações de usuários, quantidades de tarefas e plataformas de uso podem ser significativamente grande para os *web sites*. Uma solução a esse impasse é a de se recrutar voluntários para a atividade de teste. Como isso iria resultar na necessidade de se ter usuários de diversas regiões, esta hipótese de se trazer um grande grupo de usuários para um único laboratório e treiná-los para o teste de usabilidade desejado, pode ser uma atividade demasiadamente cara e, portanto, praticamente, inviável. Apesar dessa técnica ser comumente conhecida pela comunidade como "teste de usabilidade", para os usuários, entretanto, a palavra "teste" oferece uma conotação negativa e que causa problemas de constrangimento dos usuários. Não se deve deixar que os usuários pensem que eles estão sendo testados, mas sim insistir que o alvo do teste é o próprio *web site* e a forma de interação do usuário. Os usuários estão apenas auxiliando na identificação de problemas, cujas soluções podem tornar a interação mais prática e produtora. Ainda, outros fatores podem ser considerados limitantes na realização de um teste de usabilidade, tais como custo e competência da equipe de avaliadores. Outros podem ser considerados como indutores de ruídos nas observações, como a sobrecarga cognitiva para realização das tarefas e a necessidade de verbalização das ações tomadas, por outro.

As avaliações de usabilidade de *web sites*, com um enfoque especial para o comércio eletrônico, representam, então, o contexto deste trabalho. Os problemas a serem atacados neste contexto são basicamente os seguintes:

- Dificuldades de acompanhar a evolução da usabilidade de *web sites* em constante alteração;
- Dificuldades de comprovar avaliações diagnósticas qualitativas da usabilidade de *web sites* destinados a um público geral numeroso e variado. Em particular, estas dificuldades estão ligadas ao alto custo para envolver usuários em quantidade e variedade necessárias e às interferências na avaliações quantitativas comprobatórias.

A hipótese de base para este trabalho é de que se pode, por meio da mineração de dados de arquivos de *log*, obter taxas e métricas objetivas que quantifiquem a usabilidade de interfaces de determinadas tarefas, viabilizando, assim, avaliações confiáveis, objetivas, rápidas e de baixo custo.

## 1.2 – Objetivos

O foco principal deste trabalho é a avaliação da usabilidade de *web sites*. Neste contexto, o objetivo geral é tornar as avaliações quantitativas comprobatórias abrangentes, rápidas e baratas, além de invisíveis aos usuários. Para tanto, pretende-se definir uma abordagem avaliativa e um ambiente informatizado que apóie o monitoramento da usabilidade real desses *sites* por meio de métricas quantitativas objetivas, determinadas a partir da análise de dados de arquivos de *log* em servidores *web*.

Visando, então, auxiliar este processo, são apresentadas a proposição e a implementação de um ambiente de apoio à avaliação de *web sites*. Este ambiente, que em algumas partes está sendo descrito na forma de protótipo é denominado por Ergo-Monitor. Como apresentado nesta tese, ele é capaz de determinar taxas e métricas que quantifiquem a usabilidade de interfaces que apóiam as tarefas suportadas pelo *web site* em monitoramento. Tais taxas e métricas são calculadas com base nas informações que são coletadas a partir de utilizações reais do *web site*, de forma automatizada e praticamente “imperceptível” pelo usuário. Tais informações podem ser obtidas de maneira relativamente simples, uma vez que elas se encontram armazenadas nos *server log files*. Depois de determinadas, as taxas e métricas da usabilidade verificadas no monitoramento são confrontadas pelo avaliador com valores característicos de boas e más interações para *web sites* que estejam na mesma categoria do *web site* sendo avaliado.

Desse modo, espera-se que os resultados obtidos não sofram influências externas como ruídos nas observações, ou mesmo constrangimentos que os avaliadores podem sentir quando fazem parte de avaliações “típicas” empregando-se métodos “tradicionais”. Também se espera que o uso do Ergo-Monitor venha auxiliar na solução de um problema bastante grande em avaliações de usabilidade: seu alto custo.

### 1.2.1 – Objetivos Específicos

Ao objetivo geral deste trabalho estão ligados os seguintes objetivos específicos:

- 1) Especificar um ambiente tecnológico capaz de coletar e analisar dados sobre a interação humano-computador, sendo que tais dados estão gravados em arquivos de *log*. A partir desses dados, gerar taxas e métricas verificadas quando do monitoramento da usabilidade de um *web site*;
- 2) Construir versões prototipadas de programas capazes de realizar tal coleta, desprezar os dados que não sejam significativos para análises acerca da usabilidade e apresentar mecanismos para a obtenção das taxas e métricas relacionadas às interações;
- 3) Observar interações entre usuários reais e outros *web sites* que estejam na mesma categoria do *web site* a ser avaliado, para determinar valores (quantitativos) admissíveis para a usabilidade, capazes de balizar os níveis de usabilidade verificados para os *sites* monitorados. Como *valor admissível* entende-se uma unidade de medida que represente uma delimitação numérica para a execução de tarefas com patamares de qualidade na interação bons ou ruins;
- 4) Mostrar um estudo de caso da aplicação desta abordagem e do ambiente para o monitoramento.

### 1.3 – Justificativas do Projeto

O emprego dos dados coletados automaticamente através do Sistema de Monitoramento Ergo-Monitor pode ser visto como uma solução interessante para auxiliar a tarefa de se avaliar a usabilidade de um *web site*, uma vez que evita constrangimentos ao usuário e busca reduzir as possíveis subjetividades nas interpretações desses dados, uma vez que apresenta valores numéricos como indicadores da qualidade nas interações e as comparações com outros valores podem servir como base para as análises acerca das características de usabilidade do *web site*. As ferramentas que compõem o Ergo-Monitor estão baseadas na interpretação dos *server log files* para identificar páginas visitadas, cancelamento de acesso, mensagens de erro ocorridas, acesso às páginas de ajuda, etc.

Entre os dados que são coletados, encontram-se:

- quantidade de acessos ao *web site*;
- quantidade de acessos a páginas ou atividades específicas (tais como realizar uma compra, fazer uma consulta, etc.);
- tempo de acesso para o *web site* e para cada página que o compõe;
- quantidade de mensagens de erro apresentadas por página (e geral);
- quantidade de páginas de ajuda consultadas.

Especificamente, neste trabalho, foram empregados apenas os *server log files* (arquivos mantidos por servidores *web*, contendo um registro de todos os eventos solicitados por um cliente) como fonte para a coleta dos dados relativos às interações, visto que eles são suficientes para apresentar uma certa porção de informações quantitativas relativas ao uso de um *web site*. Até o momento, estas informações têm sido compiladas e interpretadas para análises estatísticas sobre [Jones, 1997]:

- a quantidade de acessos;
- quais foram os *browsers* empregados;
- tempos de acesso e problemas técnicos encontrados;

A análise destes dados também proporciona algumas informações técnicas relativas ao carregamento do servidor, às requisições sem sucesso, e outras que podem auxiliar em atividades de divulgação (*marketing*), além de proporcionar indícios de melhorias no *web site* [Jones, 1997].

Com o emprego do Ergo-Monitor pretende-se obter taxas e métricas capazes de caracterizar a usabilidade do *web site*. Algumas são listadas abaixo:

- taxas de pedidos de ajuda;
- taxas de mensagens de erro;
- taxas de repetições de tarefas (i.e., uma proporção que apresente quantas vezes a mesma tarefa, tal como o preenchimento de um formulário, teve que ser repetida);

- taxas de desistências de acesso;
- taxas de cancelamentos de acessos;

A arquitetura conceitual deste sistema pode ser inicialmente apresentada, como composta por:

- **Pré-diagnósticos** - obtidos como resultado de técnicas diagnósticas capazes de apontar para possíveis problemas em interfaces que apóiam determinadas tarefas. Por exemplo, em consequência da falta de condução da interface, o usuário pode ser levado a hesitações, a buscar ajuda, a ficar inativo, ou a cometer erros;

- **Modelo de Tarefas a monitorar** – com base em diagnósticos e nas características do *web site* a monitorar, o avaliador deve montar um modelo de tarefas nas quais os comportamentos de sucesso, desistência, cancelamento, abandono, busca de ajuda e enfrentamento de erros sejam identificáveis, clara e objetivamente por meio da análise de arquivos de *log*.

- **Taxas de usabilidade verificadas** – calculadas através de ferramentas de filtro e de busca de informações específicas nos *server log files*. Estas quantificam características da interação, como por exemplo: taxa específica de sucesso em um cadastro, taxas de desistências em uma busca, etc.; e

- **Parâmetros de usabilidade admissíveis** – valores associados às interações com *web sites* de boa e de má usabilidade, quantificando as taxas de efetividade, de desistências, de chamadas de ajuda, de mensagens de erro, de retrabalho que podem ser admitidas para o site em monitoramento.

Os resultados fornecidos pelo emprego do ambiente Ergo-Monitor podem fornecer indicações que comprovem diagnósticos qualitativos feitos previamente, ou venham a suscitar a realização de novas avaliações deste tipo.

## 1.4 – Divisão dos Capítulos

Este trabalho está dividido da seguinte forma: no Capítulo 1 é apresentada uma introdução do contexto de avaliação da usabilidade de sistemas interativos, mais especificamente de *web sites*, além dos objetivos que este trabalho de doutoramento buscou realizar visando solucionar os problemas apontados.

O Capítulo 2 apresenta uma revisão bibliográfica da literatura pesquisada e empregada na realização deste trabalho, visando descrever as principais metodologias, técnicas e critérios que apoiem a avaliação de sistemas interativos. Ainda, neste Capítulo é apresentada a técnica Co-Inspeção Avaliativa como exemplo de técnica diagnóstica orientada a tarefas.

O Capítulo 3 apresenta os conceitos e ferramentas específicas para coleta e filtragem de dados relativos à interação (provenientes basicamente dos *log files*), e que são usados na definição do ambiente Ergo-Monitor. Ainda, neste Capítulo são apresentadas algumas particularidades a respeito dos sistemas de comércio eletrônico.

No Capítulo 4 são apresentados o ambiente Ergo-Monitor e seus módulos, destacando os programas que compõem o Log Control e as limitações encontradas em sua implementação atual. Além disso, é apresentado um estudo de caso que visa validar este ambiente. Para tanto foi desenvolvido um *web site* específico para comércio eletrônico de produtos infantis (da empresa Ruth Fraldas, da cidade de Rio Claro – SP).

Finalmente, o Capítulo 5 traz conclusões relativas ao emprego do Ergo-Monitor e os trabalhos futuros que ainda devem ser realizados para se obter resultados mais significativos no que diz respeito à aplicação desta técnica para o apoio à avaliação de usabilidade de *web sites*.

## 2 - Revisão Bibliográfica

Este Capítulo apresenta uma revisão bibliográfica das principais técnicas e metodologias empregadas para a avaliação da usabilidade de sistemas interativos, mais especificamente, para avaliação de *web* sites, além das ferramentas automatizadas que dão suporte a esta atividade. Ainda, é apresentado um detalhamento acerca da técnica de avaliação Co-Inspetiva, por se tratar da técnica que serviu como base para o desenvolvimento do Ergo-Monitor.

Uma avaliação não deve ser vista como uma fase independente e única dentro do processo de projeto de uma interface, e não deve ser encarada como uma atividade a ser cumprida caso os prazos de tempo e custo não estejam “estourados”. As interfaces devem ser gradativamente avaliadas e, com os resultados obtidos nessas avaliações, procurar melhorá-las.

Não se pode afirmar que exista um consenso de qual metodologia ou técnica deve ser empregada para se obter uma avaliação sempre eficiente. Diferentes tipos de avaliações que empreguem métodos específicos devem ser utilizados nas diversas fases do desenvolvimento da interface [Rocha e Baranauskas, 2000]. Uma avaliação de interfaces possui três grandes objetivos:

1. Avaliar a funcionalidade do sistema, que deve estar adequada aos requisitos das tarefas dos usuários, que deve poder executá-las de modo fácil e eficiente;
2. Avaliar o efeito da interface junto aos usuários;
3. Identificar problemas específicos com o projeto, isto é, identificar aspectos do projeto que, quando usados no contexto envolvido, causam resultados inesperados ou confusão entre os usuários.

A aplicação de testes de usabilidade é o principal meio para se avaliar sistemas interativos, e certamente, o mais tradicional. Contudo, usuários reais são difíceis de serem recrutados, além de serem custosos [Rocha e Baranauskas, 2000].

Ao avaliar um sistema interativo é importante saber o que a usabilidade significa para este sistema [Lacerof and Paternó, 1998]. Por exemplo, um programa de controle de voo deve possuir algumas características que um processador de texto não requeira, ou então um *web site* voltado à apresentação de informações econômicas deve requisitar particularidades que não sejam necessárias em um *web site* voltado ao público infantil. Este é o motivo pelo qual uma análise dos usuários e de suas necessidades é importante. Se não se conhece o que o usuário quer e precisa, não se pode conhecer quais tarefas ele(a) está apto a realizar.

É importante considerar aspectos relacionados à realização das tarefas em uma avaliação de usabilidade. Uma *tarefa* é realizada visando atingir um objetivo. Pode-se pensar nas tarefas em diferentes níveis de abstração, variando desde tarefas de alto nível (tais como obter informações em projeções de filmes) a tarefas de baixo nível (tais como selecionar um botão na tela). Em ambos os casos, pode-se pensar nas tarefas que podem ou não ser realizadas e na ordem temporal através dessas tarefas. Mais especificamente, no caso de tarefas de alto nível, as relações temporais são determinadas pelas dependências lógicas entre as tarefas (por exemplo, imprimir um arquivo pode ser realizado apenas após a identificação do nome do arquivo a ser impresso). Enquanto que no caso de tarefas de baixo nível, as relações temporais podem depender também das restrições apresentadas pela implementação da interface [Lacerof and Paternó, 1998].

De modo geral, uma avaliação de sistemas interativos deve analisar:

- a) as tarefas que os usuários são capazes de realizar;
- b) as tarefas que os usuários não são capazes de realizar;
- c) quantas vezes cada tarefa deve (pode) ser realizada;
- d) em qual ordem as tarefas devem (podem) ser realizadas;
- e) os possíveis erros que podem ser cometidos;
- f) quanto tempo o usuário deve levar para realizar uma determinada tarefa;
- g) entre outras.

A Seção 2.1, a seguir, apresenta os conceitos básicos relacionados à Engenharia de Usabilidade, enquanto que as demais apresentam as principais características das metodologias e técnicas mais comumente empregadas para a realização de avaliações de usabilidade de sistemas interativos.

Dentro desse contexto de avaliações, pode-se afirmar que um *web site* é um sistema de software interativo, pois ele interage com pelo menos dois tipos diferentes de usuários: usuários finais que tentam alcançar alguma meta, e desenvolvedores se esforçando para manter o funcionamento do sistema, melhorando-o [Nielsen, 2000]. Além disso, dado ao altíssimo número de *web sites .com* que podem ser acessados atualmente, sempre haverá uma “disputa” pelo interesse do usuário e o *web site* que se apresentar com características que facilitem a interação apresenta um atrativo a mais para chamar e prender a atenção dos usuários. Tais características, quando analisadas sob o enfoque de considerações financeiras, pode representar o sucesso ou fracasso de uma empresa em um mercado tão competitivo como o atual [Nielsen et alli, 2002]. O *web site* da Netcraft apresenta que em agosto de 2003 foram catalogados 47.807.275 *web sites* com extensão *.com* [Netcraft, 2003], o que comprova o quão acirradas são as disputas pela atenção dos usuários.

Portanto, os usuários finais podem ser caracterizados em termos de [Smith et alli, 1997]:

- *metas e tarefas*: por exemplo, busca de informação, escolhendo onde comprar algum produto específico, comprando produtos, escrevendo uma revisão de livro, etc.

- *contexto*: o comportamento do usuário durante o processo de busca da informação é fortemente afetado pela cultura dos usuários, idioma, conhecimento prévio do assunto, experiência prévia em utilizar a *web*, etc.

- *tecnologia*: usuários finais interagem com o *web site* através de uma tecnologia que não está sob controle do *web designer*: *browsers*, protocolos, *plug-ins*, plataformas de sistemas operacionais, dispositivos de interação (telas, dispositivos de fala, canetas, teclados de telefone reduzidos, etc.), conexões de rede, etc.

A busca por informação através de um *browser* é um processo que quase todos *web sites* têm que apoiar. Infelizmente, também é uma tarefa difícil para modelar porque cercam complexos processos cognitivos, sociais e culturais medidos por interpretação de

mensagens textuais, visuais, auditivas, além da seleção da informação pertinente [Usability, 2001].

Por outro lado, existem desenvolvedores e mantenedores para os *web sites*. Entre suas atividades, um papel importante é representado pelas ações que incluem: manutenção corretiva (corrigindo problemas com o comportamento do *web site* ou inserindo assuntos perdidos), manutenção adaptável (melhorando o *site* com respeito a novas tecnologias, como novos *browsers*, capacidades, ...), manutenção perfectiva (melhorando o comportamento ou o conteúdo do *web site*), e manutenção preventiva (corrigindo problemas no comportamento ou conteúdo antes que eles afetem os usuários). Uma fração grande destas atividades é apontada com sistemas detectores de falhas, analisando e identificando-as (são representações, dentro do sistema, de erros humanos que aconteceram durante o desenvolvimento; ou de *bugs* que normalmente ocorrem durante a interação).

Manutenção significa melhorar a qualidade do *web site*. A Norma ISO/IEC 9126 define qualidade como a totalidade de características de um produto de software que afetam sua habilidade para satisfazer ou implicam necessidades, e inclui propriedades como manutenibilidade (*maintainability*), robustez, confiabilidade e usabilidade, uma vez que são particularmente importantes para os *web sites* [ISO9126].

As propriedades gerais que se referem à avaliação de usabilidade de *web sites* como estas não são independentes: por exemplo, uma falha de robustez de um *web site*, tal como incompatibilidade do *browser* também resultará em um fracasso ou em algum nível de frustração por parte do usuário, tal como, a inabilidade do usuário para completar uma tarefa e/ou sentir algum tipo de descontentamento na realização da mesma.

Para serem operacionalizadas, estas propriedades precisam ser decompostas e mais detalhadas, visando uma avaliação mais simplificada e, talvez, de um modo mais genérico. Por exemplo, a manutenibilidade pode ser decomposta em complexidade do código HTML, seu tamanho, número absoluto de *páginas*, etc.

Como a usabilidade é o foco principal deste trabalho, um enfoque maior, é, então, dado a esta característica, uma vez que ela pode ser descrita em termos de fatores (como velocidade de uso, taxa de erro, facilidade de recuperação de erro, etc) e estes fatores, dentro de contextos específicos, podem ser reduzidos em outras propriedades de menor

nível. As propriedades mais importantes para usabilidade de *web site* incluem itens relacionados com "navegabilidade", a maioria deles provenientes de [Chack, 2002]:

- consistência de apresentação e controles
- retorno (*feedback*) adequado
- organização natural da informação (rótulos sistemáticos, estrutura hierárquica clara)
- navegação contextual (em cada estado tudo e só as possíveis opções de navegação estão disponíveis)
- navegação eficiente (em termos de tempo e esforço precisaram completar uma tarefa)
- rótulos claros e significantes.

Outras propriedades pertinentes à usabilidade de um *web site* são [Lacerof and Paternó, 1998]:

- robustez (i.e. como o *web site* emprega a tecnologia usada por usuários, mas que podem não ter sido previstas pelos desenvolvedores)
- flexibilidade (por exemplo: disponibilidade de gráficos e de versões textuais),
- índices redundantes e mapas do *site*,
- funcionalidade (i.e. suporte às metas dos usuários)

Devido ao fato de que o ambiente Ergo-Monitor foi originalmente projetado para apoiar a avaliação da usabilidade de *web sites* específicos para comércio eletrônico (*e-commerce*) ainda podem ser citados outros atributos relevantes a serem investigados, tais como:

- a segurança e a privacidade serem garantidas e apresentar o quão fácil é a aquisição de informações sobre essas características tão importantes;
- quão fácil e efetivo é achar um item desejado;

- quão fácil e efetivo é procurar o catálogo para um item desconhecido a priori;
- quão fácil e efetivo é prever um item; e
- o que são políticas de retorno e como elas são comunicadas.

A Iniciativa de Acessibilidade na *Web* [W3C, 2000] é um esforço da W3C, que é uma organização criada para procurar padronizar e melhorar a acessibilidade de *web sites*. Os redatores da W3C publicam uma série de diretrizes [WAI, 1999] onde **acessibilidade** é definida como a habilidade do *web site* ser usado por alguém com inaptidão ou empregando alguma tecnologia obsoleta. Um *web site* acessível possui, entre outras, as seguintes características:

- assegura transformação amigável: deveria permanecer acessível apesar das inaptidões físicas, sensoriais e intuitivas e que trabalhe constrangimentos e barreiras tecnológicas; e
- apresenta conteúdo compreensível e navegável: deve empregar conteúdo em um idioma claro e simples e prover mecanismos compreensíveis para navegação dentro e entre páginas.

Enquanto usabilidade implica acessibilidade (pelo menos quando uma população de usuários sem restrições é considerada), o contrário não é necessariamente verdade. Por exemplo, a falta de um *link* para a página principal pode ser uma falha que afeta a usabilidade, apesar de não afetar a acessibilidade.

Todas estas propriedades (inclusive as que se relacionam com usabilidade ou as que se relacionam com acessibilidade) podem ser decompostas e detalhadas no que se refere aos atributos específicos de implementação do *web site*. Por exemplo, para determinar quão flexível é um *web site*, é necessário que se inspecione a implementação (ou talvez as especificações do projetista) para determinar diversas características como: se há uma versão textual da página, se existem *links* textuais e *links* embutidos em imagens (duplicados com relação aos *links* textuais), etc.

Algumas dessas propriedades são relacionadas aos atributos que só dependem de como o *web site* foi desenvolvido. Por exemplo: duplicatas textuais de *links* embutidos em

imagens são atributos internos, enquanto saber o quão significativo é um *label* é um atributo externo. Este é sempre o caso para propriedades que se referem ao conteúdo que requeira algum tipo de interpretação que nomeie o significado de símbolos para serem avaliados.

Nas seções a seguir são discutidos alguns conceitos sobre Engenharia de Usabilidade e apresentadas algumas metodologias, técnicas e critérios mais comumente empregados para a atividade de se avaliar um sistema interativo. Além disso, uma seção específica sobre a Norma ISO/IEC 9241 – parte 11 (que trata exclusivamente de orientações sobre usabilidade) também é apresentada, pois seu estudo foi relevante na definição das atividades deste trabalho.

## 2.1 - Conceitos Básicos da Engenharia de Usabilidade

O objetivo principal da *Engenharia de Usabilidade* é aperfeiçoar a interface com o usuário. A usabilidade está relacionada com a criação de sistemas fáceis de serem usados e de serem aprendidos. Mas isso não é tudo. Por exemplo, se os usuários não podem realizar todas as tarefas que desejam por uma característica estar ausente no sistema, não é provável que eles venham a rotular este sistema como usável. Entretanto, uma definição mais completa é preferível, e tal definição deve incluir, ao menos [Nielsen, 2000]:

- a) *relevância*: quão bem o sistema serve para as necessidades dos usuários;
- b) *eficiência*: quão eficientemente bem os usuários podem realizar suas tarefas usando o sistema;
- c) *atitude* dos usuários com relação ao sistema: seus sentimentos subjetivos;
- d) *aprendizado*: quão fácil o sistema é de ser aprendido em um uso inicial e quão bem os usuários podem se lembrar de como usar o sistema;
- e) *segurança* do sistema: dando aos usuários o direito de "desfazer" (*undo*) as ações realizadas e não permitir que o sistema aja de modo destrutivo (por exemplo, apagar arquivos sem notificar o usuário)

A parte mais importante da usabilidade depende ainda do próprio sistema. Para algumas aplicações, como por exemplo, para crianças, é muito importante que o sistema

seja fácil de ser usado; enquanto que para outros sistemas, como sistemas bancários, a eficiência é uma das partes mais importantes [Nielsen, 2000].

Em Engenharia de Usabilidade deve-se tentar operacionalizar uma *própria* definição de usabilidade para as ações que o sistema se propõe a realizar. Isso significa que a usabilidade pode ser definida em termos de fatores mensuráveis, tais como:

- desempenho do usuário na realização de tarefas específicas, medido em termos de taxa de realização, tempo para completá-las ou mesmo quantidade de erros;
- aprendizagem, medida em termos de taxa ou tempo para a conclusão das tarefas, número de erros ou uso de documentação e *help*;
- flexibilidade, medida em quão bem o sistema pode ser adequado às trocas de requisitos durante a utilização.

Esses e outros fatores similares podem ser o resultado de uma avaliação de usabilidade de um sistema interativo. Alguns dos benefícios de se ter termos mensuráveis são o fato de ser comum que os recursos inseridos no projeto relativos à usabilidade venham a aumentar, visando com isso, uma maior satisfação de uso por parte do usuário, ou o fato de que será mais fácil comparar requisitos de projetos alternativos.

A avaliação de usabilidade é uma tarefa que exige conhecimento e experiência. Para a maioria dos peritos em usabilidade, grande parte dos métodos de avaliação é “simples”. Para não-experientes, entretanto, esses mesmos métodos podem ser considerados “complicados”. Os *guidelines*, por exemplo, podem ser uma forma de reduzir essa falta de experiência, uma vez que algumas pesquisas mostraram que a aplicação de *guidelines* teve um impacto positivo [Scapin, et. alli., 2000]. Para aplicações *web*, a avaliação de usabilidade não é apenas importante na identificação de problemas nas fases iniciais do processo de desenvolvimento, mas também no gerenciamento das freqüentes atualizações que se fazem necessárias ou requisitadas [Kaiser, 2001]. A avaliação automática pode ser proposta também como uma abordagem para avaliação de usabilidade de *web sites*. Considerando a diversidade de estratégias que lidam com avaliação automática, Ivory e Hearst [Ivory et. al., 2001] apresentaram uma taxonomia compreensiva para 57 métodos aplicáveis para a *web*. A avaliação automática parece ser barata, fácil de se conduzir,

mesmo para pessoas não-experientes em usabilidade e fácil de ser integrada em um projeto iterativo.

G. Brajnik [Brajnik, 2000] comparou 11 ferramentas atualmente disponíveis para avaliação automatizada para a *Web*, tais como *Bobby* [Kaiser, 2001], *Lift* [Lift, 2002] e *WebSat* [WebSAT, 2002]. A maioria delas proporciona informações úteis para a descoberta de problemas relacionados ao código HTML, tais como falta de *tags* alternativos para imagens e verificações sintáticas. Algumas ferramentas são capazes de identificar *links* quebrados e calcular o tempo de *download* de arquivos e/ou figuras. *Bobby*, por exemplo, pode checar automaticamente 27% dos *guidelines* propostos pela W3C [W3C, 2002] proporciona alertas para outros 55%.

Entretanto, a automatização da avaliação apresenta problemas bastante conhecidos relativos à própria automatização, tais como [Farenc et alli, 1996]:

- na maioria dos casos, a inspeção automatizada das regras é feita nos níveis sintáticos e léxicos a partir do código de implementação da interface;
- a automatização da avaliação de usabilidade pode não levar em consideração algumas informações qualitativas e subjetivas que podem ser importantes, como, por exemplo as preferências dos usuários ou má-interpretações dos textos ou mecanismos de interação;
- algumas regras ergonômicas requerem descrições de alto nível (ou funcionais) dos objetos usados, mas, em geral, tais informações não podem ser recuperadas de forma automática.

Na seção a seguir são analisadas algumas considerações sobre a Norma ISSO/IEC 9241 parte 11, salientando as recomendações que nortearam as atividades deste doutoramento.

## **2.2 – Norma ISO/IEC 9241 – parte 11**

A Norma ISO 9241-11 esclarece os benefícios de medir usabilidade em termos de desempenho e satisfação do usuário. Estes são medidos pela extensão na qual os objetivos pretendidos de uso são alcançados, pelos recursos gastos para alcançar os objetivos pretendidos e pela extensão na qual o usuário considera aceitável o uso do produto.

A Norma ISO 9241-11 enfatiza que a usabilidade dos computadores é dependente do contexto de uso e que o nível de usabilidade alcançado dependerá das circunstâncias específicas nas quais o produto é usado. O contexto de uso consiste de usuários, tarefas, equipamentos (*hardware*, *software* e materiais), e do ambiente físico e social, pois todos esses podem influenciar a usabilidade de um produto dentro de um sistema de trabalho. As medidas de desempenho e satisfação do usuário avaliam o sistema de trabalho como um todo, e, quando um produto é o foco de interesse, estas medidas fornecem informações sobre a usabilidade daquele produto no contexto particular de uso proporcionado pelo restante do sistema de trabalho. Os efeitos das mudanças em outros componentes do sistema de trabalho, tal como: tempo de treinamento do usuário ou melhoria de iluminação, podem também ser medidos pelo desempenho e satisfação do usuário.

### **2.2.1 – Objetivos da Norma ISO/IEC 9241 – parte 11**

A Norma ISO 9241-11 define usabilidade e explica como identificar a informação necessária a ser considerada na especificação ou avaliação de usabilidade de um computador em termos de medidas de desempenho e satisfação do usuário. É dada orientação sobre como descrever explicitamente o contexto de uso do produto (*hardware*, *software* ou serviços) e as medidas relevantes de usabilidade. A orientação é dada na forma de princípios e técnicas gerais, em vez de requisitos para usar métodos específicos.

As orientações da Norma ISO 9241-11 podem ser usadas na aquisição, projeto, desenvolvimento, avaliação, e comunicação da informação sobre usabilidade. Esta Norma inclui orientações sobre como a usabilidade de um produto pode ser especificada e avaliada. Ela se aplica tanto a produtos de uso geral quanto a produtos sendo adquiridos ou sendo desenvolvidos dentro de uma organização específica.

Além disso, a Norma ISO 9241-11 também explica como medidas de desempenho e satisfação do usuário podem ser usadas para medir como qualquer componente de um sistema afeta todo o sistema de trabalho em uso.

A orientação inclui procedimentos para medir usabilidade, mas, não detalha todas as atividades a serem realizadas. A especificação de métodos de medidas detalhados baseados no usuário está além do objetivo da ISO 9241-11 [ISO 9241-11].

Esta Norma é aplicada ao trabalho de escritório com computadores e em outras situações onde o usuário está interagindo com um produto para alcançar seus objetivos. As partes 12 a 17 fornecem recomendações condicionais que são aplicadas em contextos de uso específico. As orientações nesta parte da Norma podem ser usadas para ajudar a identificar a aplicabilidade de recomendações individuais.

A Norma ISO 9241 parte 11 enfoca especificamente a usabilidade e não fornece uma ampla cobertura de todos os objetivos de projeto ergonômico citados na ISO 6385. Entretanto, o fato de projetar para a usabilidade irá contribuir positivamente para os objetivos ergonômicos, tal como a redução de possíveis efeitos de uso adversos do sistema de computadores sobre a saúde, segurança e desempenho humano.

Ainda, esta parte da Norma apresenta as medidas de usabilidade que devem ser baseadas em dados que reflitam os resultados de usuários interagindo com o produto ou sistema de trabalho. É possível coletar dados através de meios objetivos, tais como medidas de saída, de velocidade de trabalho ou de ocorrência de eventos particulares. Opcionalmente, os dados podem ser coletados de respostas subjetivas de usuários expressando sentimentos, crenças, atitudes e preferências. As medidas objetivas fornecem indicações diretas de eficácia e eficiência enquanto que as medidas subjetivas podem estar ligadas diretamente com a satisfação.

Convém notar que é possível obter dados relacionados para cada componente de usabilidade a partir de medidas objetivas ou subjetivas. Por exemplo, a satisfação pode também ser inferida de medidas objetivas do comportamento dos usuários, e as estimativas de eficácia e eficiência também podem ser derivadas de opiniões subjetivas que os usuários expressam sobre seus trabalhos e seus resultados.

A validade dos dados obtidos a fim de prever o nível de usabilidade alcançado quando um produto é realmente usado dependerá da extensão com a qual os usuários, tarefas e contexto de uso são representativos da situação real e da natureza das medidas escolhidas. Em um extremo, alguém pode fazer medidas no “campo” usando uma situação real de trabalho como base para a avaliação de usabilidade de um produto. No outro extremo, alguém pode avaliar um aspecto particular do produto em um “laboratório”, onde os aspectos dos contextos de usos relevantes são recriados de modos representativos e

controlados. A vantagem de usar um laboratório baseado na abordagem é que ele oferece oportunidade de exercitar um controle maior sobre as variáveis que podem ter efeitos críticos no nível de usabilidade alcançada, e medidas mais precisas podem ser feitas. A desvantagem é que a natureza artificial de um ambiente de laboratório pode produzir resultados irreais.

#### **2.2.1.1 - Medindo Eficácia**

A eficácia é definida como a acurácia e completude com que os usuários atingem objetivos específicos.

Para medir acurácia e completude é necessário produzir uma especificação operacional de critérios para atingir satisfatoriamente os objetivos. Isto pode ser expresso em termos de qualidade e quantidade de saídas, por exemplo, a especificação de um formato requerido para documentos de saídas juntamente com o número e extensão de documentos a serem processados.

A acurácia pode ser medida pela extensão com a qual a qualidade da saída corresponde com o critério especificado e a completude pode ser medida como a proporção da quantidade alvo que foi alcançada.

Se uma única medida de eficácia é requerida, é possível combinar medidas de acurácia e completude. Por exemplo, completude e acurácia podem ser calculadas como porcentagens e multiplicadas entre si resultando um valor percentual para eficácia (E.2.2, E.2.19). Em casos onde não é apropriado nivelar acurácia com completude, as duas medidas devem ser consideradas independentemente.

#### **2.2.1.2 - Medindo Eficiência**

A eficiência é medida relacionando o nível de eficácia alcançada com os recursos usados. Por exemplo, a eficiência temporal pode ser definida como a proporção entre a medida de eficiência em alcançar um objetivo específico e o tempo para alcançar tal objetivo. As medidas de eficiência devem levar em consideração as demandas físicas causadas pelas altas taxas de entrada e períodos de atividades prolongados.

Os recursos cognitivos gastos na condução de tarefas também podem ser medidos. Os efeitos da carga de trabalho cognitivo têm certas características especiais, em que tanto subcarga, quanto sobrecarga pode resultar em baixa eficiência (queda de eficiência) e problemas de saúde e segurança. Uma tarefa que demande pouquíssimo esforço mental pode resultar em baixa eficiência (queda de eficiência) porque isto leva ao tédio e à falta de vigilância, a qual diretamente reduz a eficiência. Em tal caso, tanto a eficácia quanto a eficiência podem ser melhoradas aumentando a demanda. Uma excessiva carga de trabalho cognitivo também pode resultar em queda de eficiência, se isto causar perda de informação resultando assim, em erros. Isto é uma importante questão em situações onde a segurança é crítica, como por exemplo, o controle de tráfego aéreo e controle de processos.

### **2.2.1.3 - Medidas de Satisfação (Efetividade)**

A satisfação (definida como ausência de desconforto e atitudes positivas para o uso do produto) é uma resposta do usuário na interação com o produto. A satisfação pode ser avaliada/estimada por medidas subjetivas ou objetivas. Medidas objetivas podem ser baseadas na observação do comportamento do usuário (p.ex. postura corporal, movimento do corpo, frequência de distração) ou pode ser baseada no monitoramento de respostas do usuário quando indagado sobre a facilidade e satisfação com que o mesmo realizou suas interações.

As medidas subjetivas de satisfação são produzidas quantificando subjetivamente a intensidade das reações atitudes ou opiniões expressadas por um usuário. Este processo de quantificação pode ser feito de muitas maneiras, por exemplo, pedindo ao usuário para dar uma nota correspondente à intensidade de seu sentimento em um momento particular, ou pedindo ao usuário para classificar produtos na ordem de preferência, ou usando uma escala de atitudes baseadas em um questionário.

### **2.2.2 – Características da Norma ISO 9241 – parte 11 que Motivaram a Determinação das Taxas**

Nesta seção são apresentadas as características particulares que motivaram a determinação de algumas taxas acerca da usabilidade de *web site* que podem ser empregadas no contexto deste trabalho.

De uma forma geral, esta Norma esclarece os benefícios de medir usabilidade em termos de desempenho e satisfação do usuário. Estes são medidos pela extensão na qual os objetivos pretendidos de uso são alcançados, pelos recursos gastos para alcançar os objetivos pretendidos e pela extensão na qual o usuário considera aceitável o uso do produto (ou seja, do *web site*).

A ISO 9241-11 enfatiza que a usabilidade dos computadores é dependente do contexto de uso e que o nível de usabilidade alcançado dependerá das circunstâncias específicas nas quais o produto é usado. O contexto de uso consiste de usuários, tarefas, equipamentos (*hardware, software* e materiais), e do ambiente físico e social, pois todos esses podem influenciar a usabilidade de um produto dentro de um sistema de trabalho. As medidas de desempenho e satisfação do usuário avaliam o sistema de trabalho como um todo, e, quando um produto é o foco de interesse, estas medidas fornecem informações sobre a usabilidade do mesmo no contexto particular de uso.

Como o foco deste trabalho é o apoio à avaliação da usabilidade de forma automatizada, o estudo desta Norma foi focado nas *medidas de usabilidade* uma vez que tal estudo norteou a definição do que deve (e pode!) ser avaliado através de uma coleta automática dos dados provenientes dos *log files*. As Tabelas 2.1 e 2.2 provenientes do Anexo B da ISO 9241 – parte 11 apresentam medidas de usabilidade a respeito da eficácia, eficiência e satisfação que podem ser especificadas para objetivos globais e mais abrangente (p.ex. realizar uma compra) ou mesmo para objetivos menores e mais diretos (p.ex. realizar busca). Exemplos de medidas apropriadas são dados na Tabela 2.1

**Tabela 2.1** – Exemplo de Medidas de Usabilidade [ISO 9241 – parte 11]

Objetivos de usabilidade	Medidas de eficácia	Medidas de eficiência	Medidas de satisfação
Usabilidade global	Porcentagem de objetivos alcançados;  Porcentagem de usuários completando a tarefa com sucesso;  Média da acurácia de tarefas completadas	Tempo para completar uma tarefa;  Tarefas completadas por unidade de tempo;  Custo monetário de realização da tarefa	Escala de satisfação;  Frequência de uso;  Frequência de reclamações

Além disso, podem ser necessárias medidas adicionais para propriedades particulares desejadas do produto que contribuam para a usabilidade. Exemplos de algumas destas propriedades e medidas adicionais especializadas são apresentadas na Tabela 2.2.

**Tabela 2.2** – Exemplos de Medidas para Propriedades Desejáveis [ISO 9241 – parte 11]

<b>Objetivos de usabilidade</b>	<b>Medidas de eficácia</b>	<b>Medidas de eficiência</b>	<b>Medidas de satisfação</b>
Adequado às necessidades de usuários treinados	Número de tarefas importantes realizadas; Porcentagem de funções relevantes usadas	Eficiência relativa comparada com um usuário experiente	Escala para satisfação com características importantes
Adequado às necessidades para usar facilmente	Porcentagem de tarefas completadas com sucesso na primeira tentativa	Tempo gasto na primeira tentativa <sup>1)</sup> ; Eficiência relativa na primeira tentativa	Taxa de uso voluntário
Adequado às necessidades para uso não freqüente ou intermitente	Porcentagem de tarefas completadas com sucesso depois de um período específico sem uso	Tempo gasto reaprendendo funções <sup>1)</sup> ; Número de erros persistentes	Freqüência de reuso
Redução de necessidade de suporte	Número de referências para documentação; Número de chamadas ao suporte; Número de acessos para obter ajuda	Tempo produtivo <sup>1)</sup> ; Tempo para aprender por critério <sup>1)</sup>	Escala para satisfação com recursos de apoio
Facilidade de Aprender	Número de funções aprendidas; Porcentagem de usuários que conseguem aprender por critério	Tempo para aprender por critério <sup>1)</sup> ; Tempo para reaprender por critério <sup>1)</sup> ;	Escala para facilidade de aprendizado
Tolerância a erros	Porcentagem de erros corrigidos ou apresentados pelo sistema; Número tolerado de erros do usuário	Tempo gasto na correção de erros	Escala para tratamento de erros
Legibilidade	Porcentagem de palavras lidas corretamente em uma distância normal de visualização	Tempo para ler corretamente um número especificado de caracteres	Escala para desconforto visual

<sup>1)</sup> Convém que nesses exemplos os recursos sejam medidos em relação a um nível especificado de eficácia.

Na próxima seção são apresentadas as principais características dos questionários, enfocando as vantagens e desvantagens de suas aplicações.

### 2.3 - Questionários (*Checklists*)

O uso de questionários/entrevistas é uma técnica prospectiva, que tem por finalidade avaliar a satisfação dos usuários com o sistema e sua operação. Muitos aspectos de usabilidade podem ser melhor estudados simplesmente perguntando aos usuários [Nielsen, 2000]. Esses

questionários também são conhecidos na literatura como Listas de Verificação, pois visam verificar características particulares acerca do sistema interativo, e, no caso específico desta tese, da usabilidade desses sistemas.

Um exemplo dessa técnica é o QUIS – *Questionnaire for User Interaction Satisfaction*, um questionário de avaliação disponível na Internet através do endereço <http://www.lap.umd.edu/QUIS/index.html> e que se apresenta como uma ferramenta de significativo interesse quando se trata da aplicação de *checklists* para a avaliação da usabilidade de sistemas interativos. Dentre diversos outros questionários, podem ser citados: SUMI (*Software Usability Measurement Inventory*) e o MUMMS (*Measurement of Usability of Multi Media Software*), que possuem questões específicas a fatores humanos [Perlman, 2000].

Geralmente, a aplicação da técnica de se empregar *checklists* é realizada visando um aumento na efetividade de outros tipos de avaliação. Os avaliadores diagnosticam problemas de usabilidade e apoiados pelas respostas do questionário podem centrar as análises em pontos problemáticos apontados pelos usuários [Cybis, 2000]. Neste contexto, uma avaliação realizada com a presença de um usuário após o avaliador ter em mãos os resultados de uma avaliação com o emprego de *checklists* pode ser uma estratégia capaz de apresentar bons resultados [Nielsen, 2000].

Muitos aspectos da usabilidade podem ser melhor analisados simplesmente quando se usa *checklists*, pois profissionais não especializados em ergonomia, como analistas ou programadores, têm facilidade em utilizá-los em uma avaliação. Ressalta-se que a ergonomia é a ciência que tem como objetivo a compreensão das interações entre o ser humano e os outros elementos de um sistema de trabalho (computacional ou não). O ergonomista aplica teorias, princípios, dados e métodos para a concepção e avaliação de produtos e sistemas, visando de forma integrada a saúde, a segurança e o bem estar dos usuários desses sistemas [Cybis, 2000].

Os resultados produzidos através do emprego dos *checklists* como metodologia de avaliação da usabilidade são mais uniformes, os problemas encontrados são gerais e repetitivos. Basicamente as inspeções são feitas por meio de uma lista de questões a

responder sobre a interface, muitas vezes acompanhadas de notas explicativas [Cybis et alli, 2002].

Uma avaliação realizada com *checklist* apresenta as seguintes características:

- a avaliação pode ser feita pelos projetistas,
- não há a necessidade de especialistas de interface humano-computador, devido ao conhecimento ergonômico estar embutido no *checklist*;
- a sistematização da avaliação garante resultados mais estáveis mesmo quando aplicada separadamente por diferentes avaliadores;
- existe uma facilidade na identificação de problemas de usabilidade, pois as questões do *checklist* são mais específicas;
- com a redução da subjetividade relacionada aos processos de avaliação, há um aumento da eficácia da utilização dos *checklists*;
- redução de custos da avaliação, por se tratar de um método rápido.

Um *checklist* é uma ferramenta de inspeção capaz de dar suporte à avaliação preliminar de uma interface, pois consegue identificar a maior parte dos problemas encontrados por uma outra técnica ou método de avaliação mais completa [Matias, 1995].

Existem algumas inspeções que utilizam o processo de verificação da usabilidade por meio de *checklists* como a norma ISO 9241 e o ErgoList. Especificamente, o ErgoList é uma ferramenta proposta pelo LabIUtil (Laboratório de Utilizabilidade) [LabIUtil, 2003] composta de uma base de conhecimento em ergonomia, associada a uma lista de verificação, *checklist*, para a inspeção de interfaces humano-computador.

No ambiente da *Web*, onde está disponível a ferramenta, o avaliador acessa o endereço <http://www.labiutil.inf.ufsc.br> e pode verificar e analisar a interface de um aplicativo, a partir dos seguintes módulos [Andres, 2000]:

- *Checklist*, que ajuda o avaliador a realizar uma inspeção sistemática da qualidade ergonômica da interface com o usuário de seu sistema.
- *Questões*, que dão a possibilidade de conhecer de modo informal as questões que compõem o módulo *checklist*.

- *Recomendações*, onde se localizam as recomendações ergonômicas que irão auxiliar nas decisões de projeto de interfaces com os usuários.

Os usuários ainda têm a possibilidade de verificar termos pouco conhecidos por meio de um glossário, além dos enunciados das questões e de informações complementares. As vantagens dessa ferramenta giram em torno da rapidez de aprendizado, da facilidade de uso e da utilidade para os seus usuários.

## **2.4 - Avaliação Heurística**

Através da Avaliação Heurística, o avaliador inspeciona todo o software livremente tendo em mãos apenas uma lista de heurísticas (grade de avaliação) de usabilidade como guia [Sears, 1997]. Os problemas são documentados junto com as heurísticas que foram violadas, adicionando comentários para posterior compreensão do problema.

As heurísticas propostas por Jakob Nielsen [Nielsen, 1995] na seção 2.3.1 abaixo são amplamente empregadas para projeto e avaliação de interfaces. A seção seguinte (2.3.2) apresenta tais heurísticas modificadas por Instone [Instone, 1997] para sua adequação ao contexto da *web*, bem como novas quatro heurísticas que são propostas.

### **2.4.1 - Heurísticas de Usabilidade (Nielsen)**

Jakob Nielsen [Nielsen, 2002] disponibilizou sua lista de heurística mais recente, conhecida como Lista de Princípios Reconhecidos de Usabilidade, que são regras gerais para descrever propriedades comuns de interfaces utilizáveis e podem ser considerados para todos os elementos de diálogo. A esta lista podem ser acrescentados outros princípios de usabilidade ou resultados de experiências que tornem relevantes para algum diálogo específico. Abaixo, uma breve descrição dessas heurísticas:

1. *Visibilidade do Estado do Sistema* - O sistema deve manter sempre o usuário informado sobre o que está acontecendo, por *feedback* apropriado dentro de um tempo razoável;
2. *Compatibilidade entre o Sistema e o Mundo Real* - O sistema deve falar o idioma do usuário, com palavras, frases e conceitos familiares ao usuário, em lugar de termos orientados ao sistema, seguindo as convenções de mundo real e fazendo que a informação apareça em uma ordem natural e lógica;

3. *Controle e Liberdade do Usuário* - O usuário escolhe freqüentemente por engano funções de sistema e precisará de uma saída "de emergência claramente marcada" para deixar o estado não desejado sem ter que passar por um diálogo longo. O sistema deve apoiar as ações desfazer (undo) e fazer novamente (redo);
4. *Consistência e Padrão* – O usuário não deve encontrar palavras, situações, ou ações diferentes que significam a mesma coisa. Siga as convenções de plataforma;
5. *Prevenção de Erros* - Melhor que boas mensagens de erro é um projeto bem avaliado impedindo que um problema aconteça pela primeira vez;
6. *Reconhecimento* – Torne os objetos, ações, e opções visíveis. O usuário não deve se lembrar da mesma informação de uma parte do diálogo para outro. As instruções para o uso do sistema devem ser visíveis ou facilmente recuperáveis sempre que apropriado;
7. *Flexibilidade e Eficiência de Uso* – Os aceleradores (não visto pelo novato) podem acelerar freqüentemente a interação para o especialista, de maneira que o sistema reconheça os usuários sem experiência e experientes. Permita aos usuários construir aceleradores para as ações freqüentes;
8. *Projeto Estético* – Os diálogos não devem conter informação que sejam irrelevantes ou desnecessários. Toda informação extra tira a importância das informações pertinentes, diminuindo a compreensão das relevantes;
9. *Ajuda ao Reconhecimento, Diagnóstico e Recuperação de Erros* – As mensagens de erro devem ser expressas em linguagem clara (sem código), simplesmente indicando o problema e, construtivamente sugerindo uma solução;
10. *Ajuda e Documentação* – Apesar de ser melhor que o sistema seja usado sem documentação, é necessário fornecer ajuda e documentação. Todas as informações devem ser fáceis de procurar, a sua apresentação deve estar focada na tarefa, possuir uma lista de passos concretos, e não ser muito grande.

### 2.4.2 - Heurísticas para a Web (Instone)

Nesta seção, são descritas 14 (quatorze) heurísticas de usabilidade para avaliação *web sites* propostas por Keith Instone [Instone, 1997] e formuladas a partir das 10 (dez) heurísticas definidas para avaliação de outros tipos de sistema ou software [Nielsen, 2002] e 4 (quatro) novas heurísticas foram definidas especificamente para interação em ambiente *Web*:

1. *Visibilidade do estado do sistema* - o sistema sempre deve informar ao usuário o estado atual em tempo razoável;
2. *Igualdade entre sistema e mundo real* - utilizar a linguagem convencional, ou seja, natural e lógica dentro do sistema;
3. *Controle de usuário e liberdade* - permitir ao usuário liberdade durante a interação. Ações como desfazer e refazer e concisão durante o tratamento de algum erro do usuário são bons exemplos;
4. *Consistência e padrões* - evitar que diferentes palavras, situações ou ações levem a um mesmo resultado. Além disso, a utilização de modelos diferentes dos padrões usualmente empregados nem sempre são atraentes, uma vez que se tornam inutilizáveis pela maioria de usuários;
5. *Prevenção de erros* - o projetista, em primeiro lugar, deve antes de prover o software com técnicas de tratamento de erros, atentar para que o software não possua erros;
6. *Reconhecer ao invés de nova chamada* - o software deve permitir ao usuário situar-se durante a interação sem que tenha que refazer os passos até aquele momento;
7. *Flexibilidade e eficiência de uso* - o projetista deve prover o software com mecanismos que tornem a interação mais atraente ao usuário, como atalhos ou aceleradores, tornando seu uso mais eficiente;
8. *Estética e design mínimo* - disponibilizar ao usuário somente informações pertinentes à situação, sem fortes apelos visuais, mantendo coerência ao dispôr quantidade de alternativas de escolha ao usuário;

9. *Ajuda, diagnóstico e recuperação a partir de erros* - as mensagens de erros deve fornecer precisamente o erro ocorrido, apresentando-se de forma compreensível ao usuário. Além disso, tais mensagens devem oferecer soluções para os referidos problemas;
10. *Ajuda e documentação* - todo software deve possuir documentação, servindo como fonte de ajuda aos usuários em tarefas complexas;
11. *Desorientação minimizada* - o *site* deve proporcionar ao usuário informação necessária para que o mesmo atinja seu objetivo com simples e direcionadas ações;
12. *Suporte a navegação efetiva* - o usuário poderia ser informado sobre onde ele pode chegar a partir de sua localização corrente;
13. *Acessibilidade para diversos tipos de usuário* - o *webmaster* deve levar em consideração as características cognitivas e habilidades físicas do usuário necessárias para as funções que irá dispôr em seu *site*;
14. *Minimizar o tempo de download* - Páginas e imagens disponibilizadas para *download* não devem ser grandes.

A seguir são apresentados os Critérios Ergonômicos propostos por Dominique Scapin como heurísticas aplicáveis não apenas à avaliação de usabilidade, mas também ao projeto de sistemas interativos, e que são aplicáveis também aos *web sites*.

### **2.4.3 - Critérios Ergonômicos (Scapin)**

Os critérios ergonômicos representam as características mínimas que uma interface ou um web site deve ter. Eles se mostraram completos (cobrindo todos os *guidelines* correntes), distintos (independentes um do outro) e aplicáveis (em diferentes situações) [Scapin, 1997].

O conjunto dos principais critérios é apresentado abaixo e resumidos na Tabela 2.3. Tal conjunto consiste de 8 critérios principais e de 18 critérios elementares (sub-critérios) que são apresentados nesta tabela com um asterisco.

Os critérios principais são:

1. **Condução** – *Guidance* : se refere aos mecanismos disponíveis para advertir, orientar, informar, instruir e guiar os usuários durante suas interações com o computador. Este critério é subdividido em: *prompting*, agrupamento e distinção de itens, *feedback* imediato e legibilidade;
2. **Sobrecarga de trabalho** – *Workload* : está relacionado a todos os elementos da interface que têm o papel de reduzir a carga perceptual ou cognitiva do usuário e também de melhorar a eficiência do diálogo com o usuário. Este critério é subdividido em: brevidade e densidade de informação.
3. **Controle Explícito** – *Explicit Control* : refere-se tanto ao processamento do sistema através de ações explícitas do usuário quanto ao controle que os usuários têm no processamento dessas ações na interação com o sistema. Subdivide-se em: ação explícita do usuário e controle do usuário.
4. **Adaptabilidade** – *Adaptability* : a adaptabilidade de um sistema refere-se à sua capacidade de se comportar contextualmente e de acordo com as necessidades e preferências do usuário. Este critério é subdividido em: flexibilidade e experiência do usuário.
5. **Gerenciamento de Erros** – *Error Management* : Este critério refere-se aos mecanismos disponíveis para prevenir ou reduzir erros e recuperá-los (se possível) quando ocorrerem. É dividido em: proteção contra erros, qualidade das mensagens de erros e correção de erros.
6. **Consistência** – *Consistency* : refere-se à forma como as opções da interface (códigos, nomes, formatos, etc.) são mantidas em contextos similares e são diferentes quando aplicadas em contextos diferentes.
7. **Significado dos Códigos** – *Significance of Codes* : qualifica o relacionamento entre um termo e/ou um sinal e suas referências. Os códigos e nomes são significantes para o usuário quando existe uma forte relação semântica entre esses códigos e os itens ou ações a que eles se referem.

8. **Compatibilidade** – *Compatibility* : este critério refere-se ao relacionamento entre as características do usuário (memória, percepções, habilidades, expectativas, etc.) e as características da tarefa a ser realizada.

Ainda, a maioria desses critérios principais é dividida em sub-critérios, tal como descrito abaixo e apresentado na Tabela 2.3.

### 1) **Condução** -

- 1.1) *Prompting*: refere-se aos mecanismos disponíveis capazes de guiar os usuários na realização de suas tarefas;
- 1.2) *Agrupamento/Distinção de Itens*: refere-se à organização visual dos ítems de informação. Este critério leva em consideração a topologia (localização) e algumas características gráficas (formato);
- 1.3) *Feedback Imediato*: a ausência de retorno (feedback) ou um atraso no mesmo pode ser prejudicial aos usuários, que podem suspeitar de uma falha no sistema e podem tomar ações que sejam prejudiciais ao andamento da interação;
- 1.4) *Legibilidade*: refere-se às características da informação apresentada na tela que podem facilitar a leitura e entendimento dessas informações.

### 2) **Sobrecarga de Trabalho**:

- 2.1) *Brevidade*: corresponde ao objetivo de limitar a leitura e sobrecarga de entrada de dados em alguns passos da interação. Divide-se em:
- 2.1.1) *Concisão*: refere-se à sobrecarga perceptiva e cognitiva de cada entrada ou saída do sistema;
- 2.1.2) *Ações Mínimas*: refere-se à sobrecarga relacionada à quantidade de ações necessárias para se atingir um objetivo ou concluir uma tarefa;
- 2.2) *Densidade Informacional*: refere-se à sobrecarga do usuário a partir de um ponto de vista perceptivo ou cognitivo relacionado a todo o conjunto de informações apresentadas, analisando-se elemento por elemento;

### **3) Controle Explícito -**

3.1) *Ação Explícita do Usuário*: o relacionamento entre o sistema interativo e as ações do usuário devem ser explícitas, ie, o computador deve processar apenas as ações requisitadas pelo usuário e apenas quando ele as solicitar;

3.2) *Controle do Usuário*: refere-se ao fato de que os usuário devem sempre estar no controle do processamento do sistema. Toda ação possível de ser realizada pelo usuário deve ser antecipada e as ações apropriadas devem ser disponibilizadas;

### **4) Adaptabilidade -**

4.1) *Flexibilidade*: é a capacidade do sistema de ser adaptável às necessidades ou desejos particulares do usuário;

4.2) *Experiência do Usuário*: refere-se aos mecanismos disponíveis para se levar em conta a experiência do usuário;

### **5) Gerenciamento de Erros -**

5.1) *Proteção Contra Erros*: refere-se aos mecanismos disponíveis para detectar e prevenir entradas de dados com erros, ou comandos/ações com conseqüências destrutivas;

5.2) *Correção de Erros*: refere-se aos mecanismos disponíveis para os usuários corrigirem os eventuais erros que venham a cometer;

**6) Consistência** - não apresenta sub-critérios;

**7) Significado dos Códigos** - não apresenta sub-critérios;

**8) Compatibilidade** - não apresenta sub-critérios.

A seguir será apresentada a metodologia de Observação do Usuário como uma forma de se realizar avaliações de usabilidade em sistemas interativos.

## 2.5 - Observação do Usuário

A Observação do Usuário na interação é representada por um conjunto de técnicas empíricas que se diferencia por empregar controles experimentais. Ela tem as seguintes características:

- O usuário participa da avaliação como amostra do público alvo;
- São usados cenários com tarefas típicas ou críticas;
- Os dados são originados da observação dos usuários durante a interação.
- Os procedimentos utilizados para envolver o usuário na avaliação distinguem este teste na obtenção de dados.

Na equipe montada para avaliação de um sistema interativo, deve haver no mínimo um especialista em IHC. A participação dos desenvolvedores do aplicativo tanto no planejamento da avaliação como na própria avaliação incrementam subsídios para o reprojeto da interface [Nielsen, 1995].

Deve ficar claro para o usuário que o teste a ser realizado irá avaliar somente o sistema interativo e não a sua competência pessoal. O usuário deve sentir-se livre para abandonar o trabalho quando quiser. Ele pode saber que será observado durante a avaliação, mas a sua imagem e seu nome serão preservados no anonimato quanto ao resultado final da avaliação sempre que solicitado ou necessário.

Devido à necessidade de um observador presente, o constrangimento do usuário pode interferir nos resultados dos ensaios. Este constrangimento poderá ser minimizado com alguns cuidados especiais como o esclarecimento da finalidade do teste, não pressionando o usuário a participar, e realizando os testes em local e hora de pouco movimento. O observador precisa saber o que o usuário está pensando e não somente o que está fazendo, para isso pode ser necessário solicitar que ele verbalize-os durante ou após a interação.

As observações, teoricamente, podem ser realizadas em laboratório ou no próprio ambiente de trabalho. Por ser um ambiente diferente, o laboratório deve parecer o mais próximo possível do ambiente que o usuário fará suas interações, e como possivelmente o

ambiente está equipado especificamente para a avaliação, o observador pode conseguir um controle maior das situações. Para aplicativos ainda em fase de desenvolvimento, a avaliação no laboratório é mais adequada, pois o observador pode testar uma função, fazer correções e voltar aos testes. No ambiente “regular” de trabalho a diferença está no acréscimo de uma pessoa observadora. Apesar de ser trabalhosa, as informações são mais ricas mostrando as interferências alheias às tarefas e suas complicações no sistema avaliado.

Como a interação com o sistema é um processo contínuo que envolve imagens e sons, além da verbalização do usuário é recomendado o uso de câmaras de vídeo onde todas as informações possam ser registradas. Outro método é a anotação com lápis e papel que pode ser usada em qualquer local e com mínimo de custo, mas quando a observação torna-se excessivamente explícita, pode causar algum desconforto ou constrangimento para o usuário, além de requerer prática e habilidade por parte do observador e dificilmente pode ser aplicada sem o apoio de uma outra técnica de registro.

Dentre as principais técnicas empregadas na metodologia de Observação do Usuário podemos destacar:

- **Teste com pares de usuários**

Pares de usuários são colocados na execução das tarefas. Esta técnica fortalece o diálogo entre os usuários, uma vez que a troca de informações auxilia na execução das tarefas, visando tornar claros os problemas e a possível solução do mesmo.

- **Teste com usuário e observador**

A função principal do observador é acompanhar a interação. Em caso de necessidade, ele pode auxiliar o usuário a resolver problemas ou efetuar possíveis interações;

- **Teste de verbalização simultânea (*think aloud*)**

Durante o teste de interação com a interface, o usuário comenta seu raciocínio. Existem, ainda, estratégias onde laboratórios de usabilidade podem utilizar ferramentas de software para captar mais informações do usuário, como o uso de câmaras de vídeo, gravadores de áudio, equipamentos de registro de eventos (*logs*) para captar toda a interação com a interface.

A seção a seguir apresenta algumas considerações sobre avaliações específicas da usabilidade de *web sites*, enfatizando a Técnica de Co-Inspeção Avaliativa como uma ferramenta de suporte à avaliação da usabilidade de *web sites*.

A seguir será apresentada a Técnica de Co-Inspeção Avaliativa como exemplo de uma técnica de avaliação qualitativa-diagnóstica de usabilidade de *web sites* fortemente baseada em tarefas.

## **2.6 - Técnica de Co-Inspeção Avaliativa**

A técnica de **Co-Inspeção Avaliativa** tem características de inspeção, pois o caráter ergonômico das interfaces web é decomposto segundo qualidades ou componentes que podem ser inspecionadas por meio de checklists. Existem, entretanto, inspeções que não são propriamente objetivas, e que demandam julgamentos dos avaliadores, de onde surge a natureza avaliativa da técnica. Finalmente, a participação dos usuários e dos projetistas confere à técnica uma perspectiva participativa. Esta técnica visa apoiar a inspeção ergonômica de *web sites* de comércio eletrônico através da inspeção via *checklists*.

Esta técnica será descrita a seguir segundo procedimentos, documentos e ferramentas que suportam as avaliações de usabilidade de *web sites*. Ainda, esta técnica está sendo desenvolvida em parceria entre a UFSC e o INRIA/Rocquencourt/França com apoio financeiro do próprio INRIA e do CNPq através do Projeto ProTeM – CC.

A seguir são apresentados, com maiores detalhes, suas características principais. Tal apresentação se deve ao fato de que a Técnica de Inspeção Co-Avaliativa norteia este trabalho de doutoramento uma vez que ela permite a classificação dos problemas por critérios ergonômicos e também pelo fato de que ela está associada ao Ergo-Monitor, uma vez que este ambiente define a implementação de uma combinação entre técnicas comprobatórias e diagnósticas. Ainda, a Técnica de Inspeção Co-Avaliativa se restringe à usabilidade, enquanto que outras técnicas analisadas não são tão unicamente focadas na usabilidade.

### **2.6.1 - Descrição do Método, Documentos e Ferramentas**

A definição de etapas explícitas de coleta de informação é feita por meio de questionários / entrevistas e inspeção dirigidas. Esta técnica se baseia na idéia da racionalização das atividades de avaliação / inspeção ergonômica de um software interativo. O princípio da racionalização está na origem de suas características principais:

- uma sistemática racional de coleta de informações sobre o contexto de utilização do software. Não é necessário conhecer todas as informações sobre o sistema, dos usuários, de suas tarefas, e do ambiente de trabalho, mas apenas as que são pertinentes para a configuração e a realização das avaliações.
- uma estratégia racional de avaliação orientada ao binômio tarefa – interação. Não é necessário examinar todas as interações que se estabelecem com o sistema. Apenas as relacionadas com as tarefas frequentes, mais importantes e críticas .
- uma sistemática para a definição da aplicabilidade das questões ou itens de verificação. Não é necessário que o avaliador decida sobre a aplicabilidade das questões dos checklists. Um sistema pode lhe apresentar listas com somente as questões aplicáveis aos componentes técnicos da interface em avaliação e a seu contexto de operação.
- uma sistemática para a definição de ponderações entre os critérios ergonômicos, que repercutem sobre as questões que compõem as listas de verificação. Não é necessário que o avaliador decida sobre a importância das questões dos checklists ou dos problemas detectados. Um sistema pode lhe apresentar listas com questões priorizadas. Os resultados (laudos) da aplicação da técnica podem ser melhorados, na medida em que os avaliadores estejam cientes das importâncias relativas de cada questão que lhe é apresentada.

### **2.6.2 - Etapas do Método**

O procedimento de Co-Inspeção Avaliativa se inicia com a análise do contexto de operação do *web site* em avaliação (Figura 2.1). Esta análise consiste em buscar informações que são pertinentes para a avaliação de usabilidade, em particular: as características do *site*, as intenções de seus projetistas e as opiniões de seus usuários. O analista inicia suas atividades

com o reconhecimento e descrição assistida dos componentes globais do *site* e de sua *homepage*. A seguir são realizadas as entrevistas com os projetistas e com os usuários que fornecem entre outras, informações sobre diferentes cenários de operação do *site*. Estes cenários serão integrados de modo a definir as páginas web que serão alvo das atividades de reconhecimento local de componentes.

Os resultados destas atividades são registrados em uma base de conhecimentos sobre o contexto de operação. A partir da análise destes dados, o avaliador realiza a configuração do material de avaliação, propriamente dito e a alguns julgamentos avaliativos (a partir da opinião dos usuários e do confronto entre o prescrito e o real). O avaliador aplica então os *checklists* configurados apenas com questões aplicáveis e arranjadas segundo níveis de importância. O tratamento das respostas fornecidas permite que os relatórios finais sejam elaborados segundo diferentes perspectivas, como os problemas mais sérios, os problemas de um determinado tipo, os problemas observáveis, etc. Estas últimas etapas aparecem no esquema da Figura 2.2.

### **2.6.3 - Reconhecimento-Descrição Assistida Global**

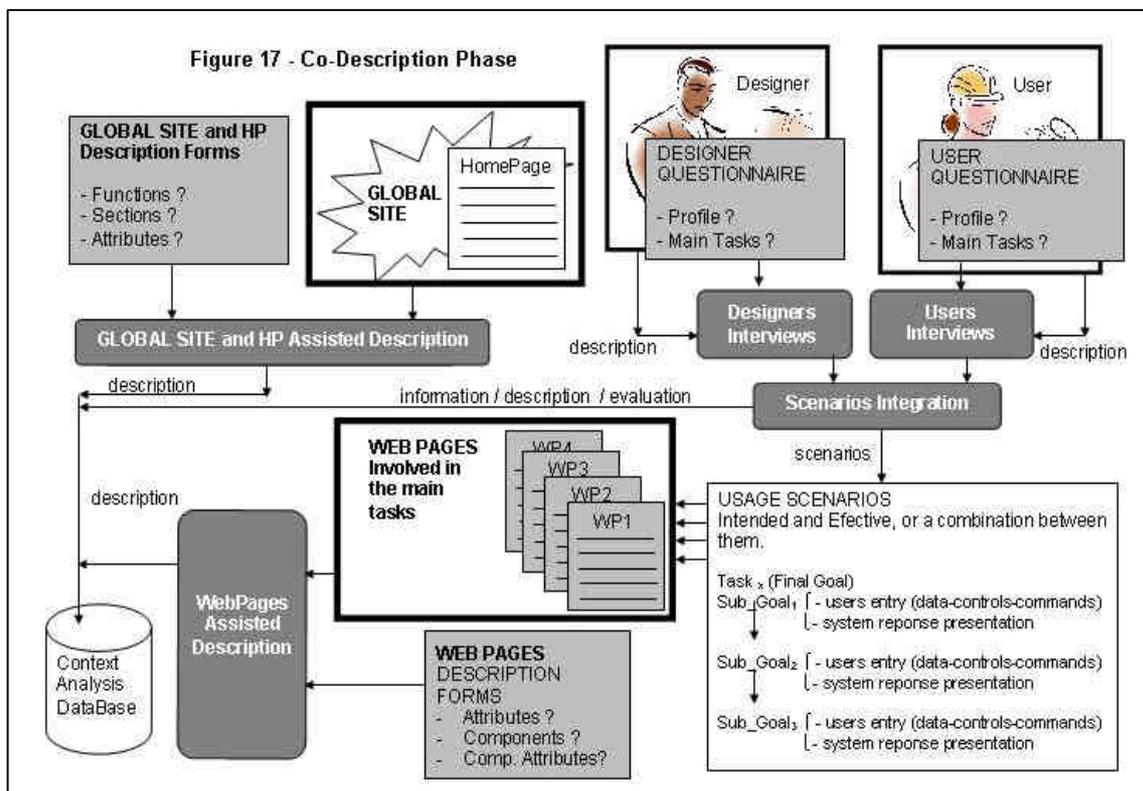
A descrição assistida global aborda dois tipos de características gerais do site. De um lado as funções existentes e a divisão modular e do outro as características da página de acesso ao site. Esta descrição é realizada, portanto, com o apoio de uma ficha de reconhecimento-descrição global e de uma ficha de reconhecimento-descrição da página de acesso.

### **2.6.4 - Entrevista com os Projetistas**

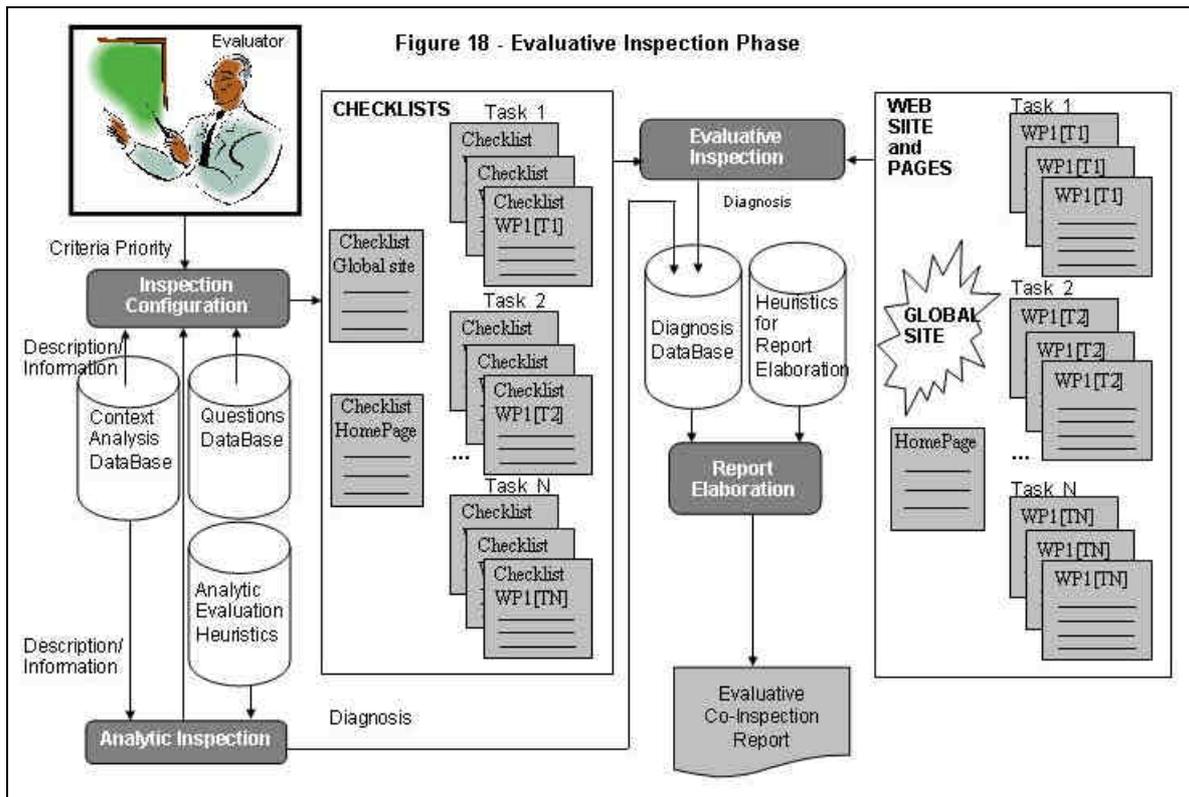
O analista realiza as entrevistas com o projetista visando obter informações referentes ao perfil dos usuários alvo do site. Se o site encontra-se em operação, é possível acessar os registros do serviço de suporte ou apoio ao usuário (*hot line*, *e-mail*, etc.), para elaborar um pré-diagnóstico dos problemas de usabilidade. Então, como parte das descrições assistidas locais às páginas web, o avaliador demanda aos projetistas a identificação dos caminhos (seqüência de páginas) associados às tarefas que podem ser executadas quando da interação com o *web site*. Para esta última atividade, recomenda-se que a entrevista seja realizada diante de um computador dando acesso as páginas do *web site* em avaliação. O avaliador será também conduzido a coletar as coordenadas de alguns usuários do *site*.

## 2.6.5 - Entrevista com os Usuários

Nesta etapa, o analista, assistido por um documento-guia, entrevista alguns usuários, de modo a obter diferentes tipos de informações sobre a operação “real” do *site*. O analista deve, em particular, obter a opinião dos usuários sobre certas características importantes da usabilidade do *site*, como denominações, ícones, organização dos dados, entre outros aspectos. O avaliador solicita também aos usuários, que identifiquem os caminhos (seqüência de páginas) relacionados com as principais tarefas (as mais freqüentes e as mais importantes). Da mesma forma, recomenda-se que esta última atividade seja realizada diante de um computador dando acesso às páginas do *site* em avaliação.



**Figura 2.1** – Fase de Co-Descrição Avaliativa



**Figura 2.2 – Fase de Avaliação Inspetiva**

### 2.6.6 - Integração de Cenários

Nesta etapa o analista integra as formas “prescrita” (informada pelo projetista) e “real” (informada pelos usuários) de realizar as tarefas apoiadas pelo site, de modo a compor o elenco de cenários de operação a serem considerados nas inspeções avaliativas. Cada cenário envolverá uma série de páginas *web* e uma ficha de descrição é gerada para cada uma delas.

### 2.6.7 - Reconhecimento-Descrição Assistida Local

Com a ajuda das respectivas fichas de reconhecimento-descrição locais o analista descreve com detalhes, as características pontuais das páginas web relacionadas com as principais tarefas a serem examinadas na avaliação.

### **2.6.8 - Inspeção Analítica**

A partir dos dados da análise do contexto e de heurísticas de inspeção analítica já é possível obter os primeiros resultados da avaliação. Isto decorre de atividades como, por exemplo:

- validação de denominações/representações do site à partir da entrevista com os usuários do *site*;
- confronto entre representações obtidas do usuário e do projetista com relação às principais tarefas (identificação, estrutura de objetivos, entradas dos usuários/saídas do sistema);
- validação de características dos usuários segundo a perspectiva dos projetistas, tendo em vista a reunião de diversos perfis encontrados entre os próprios usuários (só é possível se forem diversos os usuários ouvidos).

### **2.6.9 - Configuração das Listas de Verificação**

Nesta etapa o avaliador realiza a regulação dos instrumentos de avaliação, mais precisamente, as fichas de inspeção. Estas são definidas a partir de uma base de questões ergonômicas para os *web sites*, indexadas por critério ergonômico e por objeto de interação. Por meio de tabelas (versão manual) ou de ferramentas automáticas (versão informatizada), o avaliador define, portanto, a aplicabilidade das questões desta base, de maneira a montar as fichas para as inspeções: globais ao *site*, locais à página de acesso e locais a cada página em uma tarefa. Os aspectos do contexto de operação determinam também as ponderações entre as questões que compõem as fichas de inspeção. Existem questões que são evidentemente mais importantes que outras, visto as diferenças entre usuários, tarefas, tipos de equipamento, etc. Uma estrutura de ponderação foi definida previamente com base nos critérios ergonômicos, mas ela pode ser redefinida pelo avaliador. As fichas de inspeção trazem, desta forma, apenas questões aplicáveis e ponderadas segundo uma escala de importância ergonômica.

### **2.6.10 - Realização das Inspeções Avaliativas**

As inspeções avaliativas se dão por meio de fichas de inspeções definidas e ponderadas na etapa anterior. O avaliador realiza três tipos de inspeções avaliativas: inspeção avaliativa global do site; inspeção avaliativa local da página de acesso; e inspeções avaliativas locais

relativas às páginas ligadas às principais tarefas. Para tanto, ele deve ter uma boa visão geral do site e percorrê-lo seguindo os caminhos relacionados às tarefas principais, e procurar responder as questões propostas sobre as fichas de verificação. Os problemas de usabilidade do *web site*, bem como dos componentes neles implicados, os resultados das inspeções avaliativas são *registrados* em uma base de dados específica.

A seção a seguir apresenta algumas características e particularidades a respeito da avaliação de usabilidade na *web*, das ferramentas que podem ser empregadas para dar suporte às avaliações desse tipo de sistema interativo, bem como as dificuldades que podem ser encontradas para a condução dessa tarefa.

## **2.7 – A Avaliação de Usabilidade na Web**

Dentro do contexto do teste de usabilidade de um *web site*, abaixo são listados algumas questões e/ou *guidelines* que devem ser considerados como passos básicos para avaliações genéricas que podem ou não contar com a presença de facilitadores, especialistas ou usuários [Graham, 2000]. Apesar de existirem diversas recomendações próprias, nem sempre elas são totalmente empregadas, uma vez que muitos desenvolvedores de *web sites* utilizam-se de heurísticas próprias para avaliarem os seus próprios produtos e normalmente o fazem em curtos períodos de tempo por questões de necessidade de adequação aos prazos [Nielsen, 2000]. Essas recomendações envolvem:

1. Escopo, participantes, localização e orçamentos
  - O que vai ser testado?
  - Qual é o seu interesse com relação a testar o site?
  - Quais usuários devem participar do teste?
  - Onde será conduzido o teste? (em laboratório específico, sala de conferências ou no próprio ambiente de utilização)
  - Qual é o custo estimado do teste?

## 2. Desenvolver Cenários

- Selecionar tarefas relevantes para os usuários testarem.
- Preparar, utilizar e refinar cenários específicos para atingir tais tarefas.
- Nota: Tenha certeza de que os cenários estão bem definidos e que não sejam muito difíceis de interagir.

## 3. Recrute os participantes

- Procure por usuários que representem seus usuários em potencial
- Considere a contratação de uma empresa especializada em recrutamento para testes de usabilidade
- Cadastre os seus usuários em uma base de dados, visando avaliações futuras

## 4. Conduzir o teste de usabilidade

- Tenha um facilitador treinado para interagir com os usuários
- Treine os observadores para assistir, ouvir e tomar notas
- Certifique-se que os participantes saibam que eles estão auxiliando a construção e adequação do *web site*. Enfoque que o site é que está sendo testado, e não eles
- Peça aos participantes que empreguem a metodologia *think aloud* enquanto interagem
- Permita que os participantes expressem suas reações
- Ouça apenas! Não lidere! Certifique-se de estar em uma posição neutra, observando tanto seus movimentos corporais quanto suas palavras
- Tome notas úteis e detalhadas, concentrando-se nas observações do comportamento dos participantes

## 5. Faça um bom uso dos resultados do teste

- Compile os dados de todos os participantes
- Ouça os problemas que os participantes tiveram
- Ordene os problemas em prioridades e frequência de ocorrências
- Procure corrigir os problemas

No próximo tópico são apresentadas de forma bastante genérica as características básicas de algumas ferramentas automatizadas projetadas para apoiar a avaliação da usabilidade de *web sites* e que foram encontradas na literatura e em diversas pesquisas na *web*. A análise das características dessas ferramentas foi de significativa importância para o projeto do ambiente Ergo-Monitor e de seus componentes.

### 2.7.1 – Algumas Ferramentas Automatizadas

Por se tratar do estudo alvo desta tese, esta seção apresenta comparações básicas entre diversas ferramentas que auxiliam o desenvolvedor de *web sites* a achar falhas de usabilidade e corrigi-las em seus projetos ou produtos finais. De uma forma geral, a análise de ferramentas de apoio à avaliação de *web sites* podem ser classificadas de acordo com diversos fatores, entre eles:

- **localização:** acesso *off-line* (onde não há a necessidade de comunicação com servidores) vs *web-based* (que requer tal comunicação)
- **tipo de Serviço:** descobrem fracassos potenciais por simulação de ações de usuário, como preenchendo um formulário;
- **analísadores de falhas:** acham falhas e destacam as suas causas (*faults*), normalmente analisam sistematicamente o código fonte do website, podendo ordenar a lista de falhas de acordo com suas importâncias;
- **análise e ferramentas de reparo:** também ajudam o desenvolvedor a corrigir as falhas;
- **informações sobre o código:** uma análise de usabilidade automática pode ser executada baseada na implementação atual de um *web site* (ou mais especificamente, de seus códigos-fonte);

- **escopo:** uma série de atributos que são considerados durante a análise automática. Uma classificação baseada em escopo pode ser:
  - Validadores de HTML e Cleaners (ajudam removendo aquilo que não é usado pelo idioma-alvo do *web site*);
  - Otimizadores gráficos de HTML (melhoram o tempo de *download* e o desempenho de acessos através de gravações de certas partes do código HTML);
  - Checadores de Links (sondam todos os links que compõem uma página para determinar se as páginas relacionadas a eles existem);

Uma vez apresentadas as características básicas que devem ser analisadas pelas ferramentas de apoio à avaliação de *web sites*, são apresentadas a seguir algumas ferramentas que têm sido desenvolvidas buscando apresentar essas características e que estão disponíveis (ou estarão brevemente disponíveis) através da Web:

- *A-Prompt*: desenvolvido pela Universidade de Toronto [ATRC, 1999]; *off-line*, faz análise de falhas;
- *Bobby*: disponível por CAST [CAST, 1999]; *web based* e *on-line* e analisador de falhas
- *Doctor HTML*: disponível por Imagiware [Imagiware, 1997]; *web based* e *off-line*; analisador de falhas
- *LIFT*: disponível por UsableNet.com [Usablenet, 2000]; *web based* e *off-line*, analisador de falhas e ferramenta de reparos
- *LinkBot*: por WatchFire [WatchFire, 2000], *off-line*, análise de falhas e ferramenta de reparos;
- *MacroBot*: por WatchFire [WatchFire, 2000], *off-line* e identificador de falhas
- *MetaBot*: por WatchFire [WatchFire, 2000], *off-line*, analisador de falhas e ferramenta de reparos;

- *NetMechanic*: por Netmechanic [Netmechanic, 2000], *web based*, analisador de falhas e ferramenta de reparos;
- *WebCriteria*: disponível por WebCriteria [WebCriteria, 2000], *web based*; avaliação comparativa de um *web site* com respeito a um banco de dados de características de *web sites* que se encontrem em categorias semelhantes e com identificador de falhas;
- *WebGarage*: disponível por Netscape [Netscape, 1999]; *web based* e com analisador de falhas;
- *WebSAT*: disponível por NIST [NIST, 1999]; Web-based e off-line; analisador de falhas

A seção 2.5.2 apresenta maiores detalhes a respeito de algumas dessas ferramentas, uma vez que tais ferramentas serviram como base para o projeto do ambiente Ergo-Monitor.

Praticamente, todas essas ferramentas cobrem uma série relativamente grande de testes que podem ser agrupados de acordo com propriedades relacionadas à usabilidade, tais como:

### **1. consistência de apresentação e controles**

- evite misturar texto sublinhado com links sublinhados;
- *links* diferentes que apontam ao mesmo recurso deveriam ter o mesmo rótulo (*label*);
- rótulos associados a um determinado endereço de *e-mail* devem ser consistentes;
- cores usadas para fundos e *links* devem ser consistentes entre páginas;
- devem ser usadas imagens de fundo consistentemente;
- *links* incluídos em barras de navegação devem ser consistentes entre páginas.

## 2. feedback adequado

- páginas devem ter data de criação e autor(es) destacados.

## 3. navegação contextual

- cada página deve conter um *link* para a *homepage*;
- cada página deve conter *links* para cada página intermediária no caminho que conecta a página atual para a página principal;
- páginas não devem conter *links* para elas próprias;
- *frames* devem fixar o título;
- *links* que são locais ao *web site* devem apontar a recursos existentes;
- *links* para recursos externos devem ser conferidos periodicamente.

## 4. navegação eficiente

- o número de *links* que precisam ser seguido a partir da *homepage* para outras páginas não deve ser muito alto;
- a quantidade de páginas não deve exceder um numero pré-determinado;
- componentes de tabelas devem ter largura e altura explícitas;
- as imagens também devem ter largura e altura explícitas;
- as páginas devem ser carregadas dentro de um tempo pré-determinado que não seja muito alto;
- imagens usadas no *web site* devem ser compartilhadas (de forma que os browsers possam gravá-las na área de *cache* de memória)
- as páginas não devem conter elementos que não possam ser mostrados (como mapas não associados a quaisquer imagens)

## 5. rótulos claros e significantes

- *links* que apontem para *plug-ins* pesados devem apresentar tal especificação;
- rótulos de “*mail to*” devem conter endereços corretos;
- todas as páginas devem ter um título;
- tabelas deveriam ter *headers* e resumos;
- dentro de formulários, o texto dos campos devem ter um rótulo;

## 6. robustez

- os códigos HTML não deveriam usar estruturas que não permitam a compatibilidade com os diversos *browsers*;
- os elementos de páginas devem usar cores “*web-safe*”;
- somente códigos HTML padrão devem ser empregados;
- Códigos portáveis: deveriam ser usadas códigos fontes standards;
- cores de fundo e primeiro plano devem proporcionar contrastes suficientes.

## 7. flexibilidade

- imagens devem ter descrições textuais alternativas;
- vídeos, áudios, *applets* e outros objetos devem ter descrições textuais alternativas;
- *links* embutidos em imagens também devem estar disponíveis em formato textual;
- não podem ser projetados *links* a partir de imagens sem serem carregadas as imagens;
- devem ser usados tamanhos adequados para fontes, *frames* e tabelas.

## 8. apoio de usuários

- formulários devem ter botões de submissão (*submit*) e re-preenchimento (*reset*).

## 9. manutenibilidade

- links relativos às páginas locais ao *web site* devem ser sempre checados.

## 10. outros

- faça o aviso do número de mídias diferentes que serão usados no site
- páginas deveriam ter informações apropriadas para serem encontradas por máquinas de busca.

### 2.7.2 – Ferramentas de Destaque

Nesta seção serão sumariamente apresentadas algumas ferramentas que foram analisadas com maior destaque neste trabalho por apresentarem características bastante próximas das inseridas no ambiente Ergo-Monitor. São elas: Bobby, WebSAT, Max e Lift.

#### 2.7.2.1 - WebSAT

##### Custo - Gratuito

A Ferramenta Web Static Analyzer WebSAT [WebSAT, 2002] é uma das ferramentas incluídas no conjunto de *WebMetrics* desenvolvido pelo *National Institute of Standards and Technology* (NIST). Esta ferramenta aponta problemas de usabilidade no *web site* ao checar o código HTML das páginas individuais comparando-os com um conjunto de *guidelines* de usabilidade. WebSAT revisa o código HTML para verificar se sua implementação está de acordo com os padrões de usabilidade.

Trata-se de uma ferramenta gratuita que pode ser tanto usada *online* quanto ter uma cópia “baixada” em um microcomputador (PC ou UNIX). Após a avaliação de usabilidade com o emprego dessa ferramenta, são apresentados problemas encontrados em cada uma das páginas que compõem o *web site*, basicamente de acordo com as seguinte categorias: acessibilidade, desempenho, manutenibilidade, navegação, e confiabilidade..

De forma geral, a WebSAT é útil para tentar garantir que o código HTML segue boas práticas de codificação e também para apoiar os desenvolvedores a compreenderem melhor alguns princípios básicos de usabilidade. Os relatórios podem ser apresentados apontando também recomendações específicas em vez de apenas identificar os problemas.

A principal limitação dessa ferramenta deve-se ao fato de que ela apenas pode checar páginas individuais, o que impede uma avaliação de usabilidade do *web site* em um contexto de interações entre suas páginas.

### **2.7.2.2 - Lift**

**Custo - US\$1,00 por página (máximo de US\$50,00 para um relatório completo do *web site*)**

A empresa UsableNet oferece uma ferramenta chamada *Lift* [Lift, 2002] que analisa as páginas *web* procurando por problemas de usabilidade potenciais. Esta ferramenta também realiza uma checagem no código HTML procurando por trechos que não estejam de acordo com os princípios de usabilidade, similarmente ao que faz a ferramenta WebSAT. Da mesma forma como ocorre para a ferramenta WebSAT, a avaliação é feita por páginas individuais, entretanto uma nova versão já está em desenvolvimento para uma checagem completa e ampla de todo o *web site*. Os desenvolvedores dessa ferramenta definem o final de 2003 como uma data base para que esta nova versão esteja disponível

### **2.7.2.3 - Max WebCriteria**

**Custo - os desenvolvedores requisitam um contato para análise de custos**

A ferramenta *Max Web Criteria* [Max, 2002] utiliza um agente de navegação inteligente chamado *Max* que rastreia o *web site* para obter estatísticas vitais e definir um *ranking* de usabilidade para o mesmo. Esta ferramenta se utiliza de um modelo estatístico para simular a experiência do usuário para definir padrões em três áreas: acessibilidade, tempos de carregamento e conteúdo.

Apesar dos dois últimos padrões poderem ser considerados importantes, o foco principal na questão de avaliação de usabilidade dos *web sites* que esta ferramenta proporciona é a acessibilidade e serão apresentados comentários alusivos a como a acessibilidade pode ser mensurada empregando-se a ferramenta *Max Web Criteria*.

A avaliação de acessibilidade mede o quão rapidamente um usuário pode encontrar a informação que ele procura no *web site*. O *ranking* é determinado em função de um tempo estimado que os usuários devem demorar para encontrar o que eles estão procurando.

O primeiro fator que esta ferramenta leva em consideração ao calcular a acessibilidade é o tempo de análise de um *link*, que é baseado na premissa de que *links*

definidos em regiões “mais altas” da página são mais acessíveis. Este cálculo é contencioso, uma vez que os *links* definidos mais no alto não são necessariamente mais visíveis ou acessíveis, visto que muitos usuários ignoram naturalmente os *links* no topo das páginas evitando itens relativos a navegação ou *banners* para se concentrarem especificamente no conteúdo da *web page*.

O segundo fator ao se medir a acessibilidade é o rastreamento do *browser*, que mede a menor quantidade de *clicks* entre a página inicial e a página em que se encontra a informação procurada (ou a que permite a realização da tarefa desejada).

O tempo de acesso calculado deve ser utilizado basicamente como um *benchmark* inicial, pois ele não considera características de utilização ou efetividade das ações que o *web site* se propõe a cumprir. Esta análise também assume que todos os *links* são igualmente visíveis e compreensíveis pelos usuários, o que nem sempre é verdadeiro.

#### **2.7.2.4 - Bobby**

**Custo: Gratuito**

A ferramenta *Bobby* [Bobby, 2002] é um software “*web-based*” projetado para expor e auxiliar a reparar barreiras na acessibilidade aos *web sites* e encorajar a anuência com *guidelines* existentes relativos à acessibilidade. Essa ferramenta foi desenvolvida pela CAST, uma organização de pesquisa e desenvolvimento que não visa lucros e cujo objetivo principal é a expansão de oportunidades para pessoas com deficiências físicas através do emprego de tecnologias computacionais inovadoras e dos estudos que enfatizam as principais dificuldades encontradas para que pessoas com deficiências possam interagir com os *web sites*.

Com sua primeira versão lançada em Setembro de 1996, *Bobby* foi originalmente baseada nos *guidelines* propostos pelo *Trace Research and Development Center*. *Bobby* foi projetada para testar páginas *web* e gerar relatórios que apontem as características principais que afetem a acessibilidade do *web site* (mais especificamente de cada uma de suas páginas).

*Bobby* é uma ferramenta que apresenta resultados acerca da acessibilidade de *web sites* projetada para auxiliar na exposição (e futura correção) de barreiras à acessibilidade,

além de encorajar a aplicação de *guidelines* relativos à acessibilidade. Ainda, *Bobby* oferece sugestões priorizadas tendo como base o *World Wide Web Consortium's (W3C)* e permite que os desenvolvedores testem suas páginas e gera resultados apresentando relatórios que enfatizem os problemas de acessibilidade nas páginas testadas.

A CAST tem trabalhado em forte consonância com o *World Wide Web Consortium's (W3C) Web Accessibility Initiative (WAI)* para desenvolver uma ferramenta de avaliação que empregue seus *Web Content Accessibility Guidelines* e proporcione suporte à avaliação das páginas [W3C, 2002].

#### **2.7.2.5 - Considerações Gerais sobre Ferramentas Automatizadas**

Essas ferramentas podem ser melhor empregadas se analisadas como indicadores da usabilidade dos *web sites* e as avaliações de usabilidade devem, sempre que necessário, ser completadas com testes de usabilidade “tradicionais” baseados em observações [Chack, 2002].

Entretanto, sabe-se que as ferramentas atualmente disponíveis para avaliação automatizada para a *Web*, tais como *Bobby* [Bobby, 2002], *Lift* [Lift, 2002] e *WebSat* [WebSAT, 2002] basicamente proporcionam informações especificamente úteis para a descoberta de problemas relacionados ao código HTML, enquanto que outras podem ser capazes de identificar *links* quebrados, calcular tempos de *download* de arquivos e/ou figuras. Especificamente, a ferramenta *Bobby*, pode checar automaticamente 27% dos *guidelines* propostos pela W3C [W3C, 2002] e proporciona alertas para outros 55%.

De forma genérica, essas ferramentas buscam cobrir uma série relativamente grande de testes relacionados à usabilidade, tais como: consistência de apresentação, feedback adequado, navegação, rótulos claros e significantes, robustez e flexibilidade. Entretanto, não foram encontradas ferramentas que façam análises comparativas entre diagnósticos comprobatórios qualitativos e quantitativos tal como está propondo o Ergo-Monitor.

O Capítulo a seguir descreve as ferramentas, sistemas e estruturas empregadas no desenvolvimento do Ergo-Monitor, que, como as ferramentas de automação das avaliações

de usabilidade descritas anteriormente, procura auxiliar na obtenção de características a respeito das propriedades do *web site*.

### 3 – Ferramentas e Estruturas Empregadas

Neste Capítulo serão apresentadas as ferramentas e os sistemas empregados na concepção do Ergo-Monitor. Também são apresentadas algumas considerações teóricas que envolvem o desenvolvimento e a aplicação dos mesmos e a forma como tais sistemas e ferramentas são importantes dentro do contexto do Ergo-Monitor. Ainda, uma análise básica sobre algumas particularidades do comércio eletrônico também é apresentada.

Especificamente, este Capítulo visa apresentar estruturas, técnicas e ferramentas *QUE* devem ser consideradas na coleta, filtragem, análise e integração de dados coletados das interações dos usuários (clientes) com um *web site* (servidor). *O COMO* essas ferramentas são executadas (e seus relacionamentos) está sendo descrito no Capítulo 4. Ainda, devido ao fato do desenvolvimento do Ergo-Monitor estar fortemente voltado para *web sites* de comércio eletrônico, a Seção 3.1 visa apresentar algumas características básicas deste tipo de sistemas, enfatizando o porque são encontrados diversos *web sites* que suportem as atividades de comércio eletrônico (ou *e-commerce*).

O Ergo-Monitor é proposto no Capítulo 4, como um sistema “invisível” ao usuário. Ou seja, os usuários interagem normalmente com o *web site* sem terem consciência de que estão participando de testes de usabilidade. Os dados da interação vão sendo armazenados do lado do servidor na conexão em *server log files*, Quando acionado, o Ergo-Monitor coleta, filtra e faz a análise dos dados nos *server log files*.

A Seção 3.2 descreve a estrutura dos *server log files* apresentando o porque do acesso às informações contidas em tais arquivos terem sido suficientemente úteis para a proposição e implementação do Ergo-Monitor. Além disso, também são apresentadas as principais características dos *cookies*, que são arquivos que ficam armazenados do lado do cliente em uma conexão cliente-servidor. Apesar de não terem sido empregados no Ergo-Monitor nesta primeira versão, os *cookies* poderão ser empregados em proposições futuras visando dar maior flexibilidade e credibilidade aos laudos produzidos pelo Ergo-Monitor.

### 3.1 – Sistemas de Comércio Eletrônico

O comércio eletrônico é considerado como uma atividade fundamental para o desenvolvimento do setor produtivo, pois possibilita a ampliação e diversificação dos mercados consumidores, promove o desenvolvimento das atividades comerciais pela introdução de novos mecanismos de comercialização e acelera o desenvolvimento e a difusão de novas tecnologias de informação. Ele é caracterizado pelo uso de meios eletrônicos para a condução de transações comerciais entre empresas, governo e consumidores. Podemos identificar diversas atividades comerciais que vêm sendo executadas por serviços de comércio eletrônico, como localização de produtos, elaboração de pedidos, efetivação de pagamentos, atendimento aos consumidores, vendas em geral, etc [Meira Jr. et alli, 2002].

A realização das atividades de *marketing* e de venda direta ao consumidor através da *internet* é apenas parte das inovações que estão mudando o funcionamento das empresas, o qual tem sido modificado também por tecnologias como mensagens eletrônicas, publicação de documentos, gerência de manufaturas, etc. Assim, um dos principais benefícios do comércio eletrônico é a possibilidade de integrar a cadeia de produção e criar novos serviços que explorem os recursos que a *internet* proporciona.

A motivação tradicional para a implementação de serviços de comércio eletrônico é o uso dos recursos da *internet* para agilizar transações comerciais e aumentar a eficiência dos processos nas organizações. Neste contexto, uma definição mais precisa de comércio eletrônico pode ser apresentada como: *o intercâmbio de informações para facilitar a realização de transações comerciais, como pedidos, cotações e entregas* [Meira Jr., et alli., 2002].

A ampliação do mercado consumidor, de uma região ou país para potencialmente todo o mundo, ressalta a importância que os serviços oferecidos possam acompanhar uma demanda crescente. A facilidade para obtenção de informações sobre os diversos produtos que podem ser comercializados, a praticidade da compra sem a necessidade de que o cliente tenha que se deslocar até uma loja para analisar e adquirir um produto, a facilidade que os clientes podem ter para a busca por informações acerca de um mesmo produto nos diversos fornecedores, entre outras características torna o comércio eletrônico cada dia mais atraente

e com maiores adeptos. Ainda, a facilidade que os usuários de computadores têm encontrado para fazer o acesso aos *web sites* de comércio eletrônico das empresas permite que a prática desse tipo de comércio seja melhor difundida.

### **3.1.1 – Servidores de Comércio Eletrônico**

Os servidores de comércio eletrônico são um elemento essencial desse cenário, pois são os responsáveis pela execução dos vários serviços oferecidos. Pode-se definir servidores de comércio eletrônico como um sistema composto por componentes de *hardware* e *software* que implementa um conjunto de serviços de comércio eletrônico [Meira Jr. et. alli, 2002].

O desempenho dos servidores de comércio eletrônico é um aspecto essencial para o sucesso do serviço prestado e a satisfação dos clientes. Tendo em vista a complexidade dos servidores de comércio eletrônico, suas interações com a *internet* e do comportamento dos clientes nesse novo meio de comercialização, a análise e a modelagem de desempenho desses servidores são tarefas amplas e difíceis que devem ser realizadas de forma contínua em função da evolução dos produtos comercializados e do perfil dos clientes. Neste trabalho não é analisado o desempenho computacional dos servidores, visto que tal atividade, apesar de ser fortemente vinculada à satisfação dos usuários com relação ao acesso aos *web sites*, não está relacionada aos padrões de usabilidade que são exigidos em *web sites* e que são o foco dos estudos e análises apresentadas.

Como forma de contextualizar os servidores com relação ao Ergo-Monitor, eles podem ser definidos como programas que armazenam a informação da *World Wide Web* e a fornecem para os clientes (computadores dos usuários) quando requisitados. As respostas às requisições são documentos que podem ser estáticos (obtidos a partir de arquivos preexistentes) ou dinâmicos (gerados em tempo real em função do estado do servidor e das informações contidas na requisição). Normalmente, páginas dinâmicas exigem o acesso a informações específicas da aplicação em questão a Bancos de Dados próprios.

Os servidores são elementos passivos, operando em função das requisições recebidas. Estudos indicam que a forma e a ordem em que as requisições são atendidas são dependentes de decisões de projeto e podem afetar drasticamente o desempenho percebido pelos clientes. Entretanto, devido à complexidade de se definir e implementar ordenações

especiais, usualmente as requisições são atendidas em ordem de chegada e tratadas concorrentemente pelos servidores até serem completadas [Arlitt et all, 1996].

O principal servidor World Wide Web utilizado atualmente é o *Apache*, responsável por cerca de 60% de todos os *sites* da rede, normalmente em plataformas *Unix*. Seu sucesso se deve a três fatores básicos: (1) o bom desempenho, (2) a estabilidade do sistema operacional tipo Unix e (3) o fato de ser gratuito e ter código aberto. Além do Apache, outros servidores populares são: IIS, da Microsoft; JavaWebServer, da SUN; o Enterpriser Server, da Netscape; além do Jigsaw e NCSA. Em vista da forma de operação concorrente dos programas servidores, as máquinas utilizadas por eles normalmente devem ser muito mais potentes que as máquinas-clientes, já que vários pedidos precisam ser atendidos em paralelo para garantir um bom desempenho. Em termos de *hardware*, fatores importantes que devem ser considerados são a quantidade de memória disponível e as velocidades de acesso a discos e à interface de rede. Além da implementação do programa servidor propriamente dito, o sistema operacional também é um fator importante na determinação do desempenho dos servidores, especialmente sob carga elevada. Os grandes servidores usualmente necessitam de uma configuração minuciosa dos parâmetros do sistema para que atinjam o melhor desempenho possível, Os sistemas operacionais mais comuns atualmente em servidores são as versões do Unix (Linux, FreeBSD, Sun Solaris, etc.) e o Microsoft Windows NT. De uma forma geral, as principais diferenças entre eles são a estabilidade e a escalabilidade. Servidores Unix são em geral mais robustos, operando por períodos extremamente longos sem sofrer falhas de sistema nem exigir religamentos periódicos para garantir seu funcionamento [Meira Jr. et. alli., 2002].

### **3.2 – Server Log Files**

A WWW foi construída baseada em uma premissa simples, mas muito poderosa: todo o material na *Web* foi basicamente formatado em um padrão geral e uniforme denominado por *HTML (Hypertext Markup Language)*, e todas as requisições de chamadas e respostas devem estar de acordo com as similaridades propostas neste protocolo padrão [Nielsen, 2000]. Quando alguém faz o acesso a um servidor na *Web*, tal como uma loja virtual ou um jornal qualquer, o *browser* do usuário irá enviar uma requisição por informações para o computador da loja virtual ou do jornal. Este computador é chamado de servidor (*Web*

*server*). O servidor *Web* irá responder à requisição transmitindo a informação desejada para o computador do usuário. Neste computador, o *browser* do usuário irá, então, apresentar as informações recebidas.

Qualquer comunicação entre o *browser* de um cliente e um servidor *web* resulta em uma entrada no registro do servidor *web* que armazena dados relativos a esta transação. Um *web site* que é bastante acessado gera centenas ou milhares de entradas de *log* por hora e os compila em um *arquivo de log* (daí, o termo em inglês *log file*). Os dados capturados em um arquivo de *log* variam de acordo com o tipo do servidor usado e do formato do arquivo de *log* que este servidor suporta. De forma geral, um arquivo de *log* contém [Haigh & Megarity, 1998]:

- endereço IP do computador que faz a requisição;
- data e horário da requisição;
- *url* requisitada (e o arquivo, quando for o caso);
- tamanho do arquivo requisitado;
- protocolo empregado na requisição;
- *browser* e Sistema Operacional usados pelo computador requisitante.

O formato típico de um *log* pode ser visto na Figura 3.1. Os campos de um *log file* são apresentados abaixo:

- IP address (h) = **195.238.161.136**
- *hostname* (não é normalmente ativado por razões de desempenho)
- *identd* (não é ativado por razões de desempenho e normalmente não é suportado)
- *Date and Time* (l) = - **[06/Nov/1998:14:54:33 +0000]**
- *Request method* (u) = **GET**
- *Request path* (t) = **"/img/navigation/top\_nav/jamba\_dips\_stat.gif HTTP/1.0"**
- *Request protocol* (r) = " "
- *Response status* (s) = **"200"**
- *Response content size* (b) = **"743"**
- *Referrer path (Referer)* = **"/navigation/top\_nav/jamba\_dips\_stat.html HTTP/1.0"**
- *User agent (User-Agent)* = **"Mozilla/4.05"**
- *Cookie values (Cookie)* = **" [en] "**
- *Logging header* (l) = **"(Win95; I)"**

```
LogFormat "%h %l %u %t \"%r\"%>s %b \"/% {Referer}i \"% {User-Agent}i \"% {Cookie}i \" "
```

**Figura 3.1** – Formato de um Registro de *Log File*

Um exemplo de *log file* é exibido na Figura 3.2. Este trecho foi retirado do arquivo *access.log* situado no diretório `\var\logs` do servidor de páginas HTML *Apache*. Nesta entrada de *log* pode-se verificar o IP da máquina, a data e hora em que esta máquina efetuou o acesso, o tipo de requisição por ela efetuada, a localização do componente requerido, o tamanho deste componente, endereço onde ocorreu a requisição do componente, o *browser* utilizado e o sistema operacional utilizado pela máquina do usuário.

```
195.238.161.136 - - [06/Nov/1998:14:54:33 +0000] "GET /img  
/navigation/top_nav/jamba_dips_stat.gif HTTP/1.0" 200 743  
|"/navigation/top_nav/jamba_dips_stat.html HTTP/1.0" "Mozilla/4.05 [en] (Win95; I)"
```

**Figura 3.2** – Exemplo de *Log File*

Os dados armazenados em um *log file* podem ser compilados e combinados em várias formas, proporcionando estatísticas ou listagens, tais como:

- número de requisições feitas (*hints*);
- totais de arquivos e Kilobytes disponibilizados com sucesso;
- número de requisições por tipo de arquivo, tais como visões de páginas HTML;
- endereços IPs distintos que solicitam requisições e quantidade de requisições pedidas;
- número de requisições para arquivos e diretórios específicos;
- número de requisições por estados de código *http* (sucesso, falha, redirecionamento);

- totais e médias para um período específico de tempo (horas, dias, semanas, meses e anos);
- *urls* a partir da(s) qual(is) o(s) usuário(s) entrou(aram) no *web site*;
- *browsers* e versões que solicitaram as requisições.

Destaca-se que o número de requisições pode ser uma medida insuficiente para análises, devido à memória *cache* do lado do cliente. Uma página é automaticamente armazenada na memória *cache* do cliente por um certo período (determinado pela quantidade de memória disponível para esta função). Com isso, um documento que seja freqüentemente requisitado pode ser carregado diretamente da memória *cache*. O exemplo mais claro disso é a revisão em um *browser*: os acessos com o uso dos botões *Back*, *Forward* ou *Go* não são contados no servidor, enquanto o acesso através do botão *Reload* é. Esses fatores reduzem a quantidade de uso armazenada no servidor para uma extensão não conhecida [Haigh & Megarity, 1998]. No *web site* empregado neste trabalho para a validação do Ergo-Monitor (da empresa Ruth Fraldas), houve um esforço de implementação que garante que *todos* os acessos às páginas do *web site* sejam gravados nos *log files*.

Tanto a visualização, quanto a elaboração de medidas estatísticas acerca dos dados provenientes dos *log files* pode se dar por meio de uma ferramenta denominada por *log analyser*. A Seção 3.4.1 apresenta a ferramenta Fast Stats Analyzer [Mach-5, 2002] como uma ferramenta deste tipo. Esta ferramenta é de domínio público e pode ser facilmente acessada via internet.

### **3.3 - Cookies**

Os *cookies* são arquivos gerados pelo servidor *Web* e armazenados no computador do usuário, contendo informações prontas para um acesso futuro. Os *cookies* estão embutidos nas informações HTML fluindo entre os computadores do servidor e do cliente (usuário) e foram implementados para permitir a personalização dos usuários com relação às informações trocadas. Por exemplo, os *cookies* são usados para personalizar ferramentas de busca na *web* ou para armazenar itens de listas de compras enquanto houver uma interação do usuário com a loja virtual.

Na maioria dos casos, não apenas a armazenagem de informações pessoais em um *cookie* é não-notificada ao usuário, mas também o acesso a esse *cookie* e às informações nele contidas. Os servidores *Web* automaticamente fazem o acesso aos *cookies* relevantes sempre que o usuário estabelece uma conexão na forma de requisições *Web*.

Os *cookies* são baseados em um processo de dois estágios: primeiramente, o *cookie* é armazenado no computador do usuário sem o seu consentimento ou conhecimento. Por exemplo, com ferramentas personalizáveis de busca na *Web* tal como o *My Yahoo*, um usuário seleciona categorias de interesses a partir de uma página *web*. O servidor cria, então, um *cookie* específico, que é essencialmente uma cadeia (*tagged string*) de texto contendo as preferências do usuário, e transmite esse *cookie* para o computador do usuário. O *browser*, se suficientemente sensível a esse *cookie*, recebe tal *cookie* e o armazena em um arquivo especial chamado de *cookie list* (lista de *cookies*). Como resultado, as informações pessoais (neste caso, as categorias de preferências do usuário) são formatadas pelo servidor, transmitidas e salvas pelo computador do usuário [Mayer-Schönberger, 2000]

Como exemplo prático empregado na utilização de cookies, sob o prisma da linguagem de programação PHP [PHP, 2002], uma das linguagens especificamente projetada para ser focada para a *web* e que apresenta facilidades para a consulta e manutenção de *cookies*, um *cookie* é enviado ao cliente através da função *setcookie*, que tem o seguinte formato:

```
int setcookie (string name [, string value [, int expire [, string path [, string domain [, int secure]]]]]);
```

Onde o campo *name* recebe a mesma *string* de *value*, e significa o nome do cookie. Este é o único campo obrigatório da função. O campo *expire* refere-se a data em que o cookie expirará. *Path* especifica o endereço do site no qual foi originado o cookie. *domain* está ligado ao domínio de origem do *cookie*. Finalmente, o campo *secure* relaciona-se à segurança. Se este campo for marcado, os *cookies* só são transmitidos em canais de comunicação seguros. Depois de enviado ao cliente, o *cookie* é tratado como uma variável comum. A função *setcookie* da linguagem PHP foi escolhida para representar as comunicações com o emprego dos *cookies* apenas por motivos ilustrativos.

Esses arquivos são usados posteriormente para diversas finalidades, como por exemplo: saber o número de vezes que o usuário fez o acesso ao *web site*, as páginas visitadas, o tempo gasto em cada página, etc.

### **3.4 – Algumas Características das Ferramentas de Análise de *Log Files***

O emprego dos *server log files* como forma de se apoiar a avaliação de *web sites* é uma abordagem bastante interessante, por representar uma estratégia prática e confiável [UsableWeb, 2002]. Os *server log files* ficam armazenados nos servidores por um tempo suficientemente grande, sendo necessário um pequeno esforço computacional adicional para fazer o acesso aos dados contidos nesses arquivos, filtrá-los e modelá-los de acordo com os possíveis interesses de investigação.

A análise de dados acumulados historicamente nos *server log files* pode proporcionar aos projetistas, administradores e profissionais de usabilidade informações sobre a forma como os usuários interagem com seus *web sites* [UsableWeb, 2002]. Compiladas e interpretadas de maneira apropriada, as informações provenientes dos *logs* proporcionam uma base para estatísticas que podem indicar [Haigh & Megarity, 1998]:

- níveis de uso (acesso);
- comparações entre partes (páginas ou rotinas) de um *web site*;
- tempos de acesso e problemas encontrados durante estes acessos;
- entre outras.

Por exemplo, servidores de rede mantêm arquivos de *log* que listam todos os pedidos feitos ao servidor. Com o emprego de ferramentas de análise de *logs* é possível adquirir uma boa idéia da procedência dos visitantes, com que frequência eles retornam, como eles navegam através do *site*, o número de erros de acesso, etc.

A análise desses dados também proporciona algumas informações técnicas relativas ao carregamento do servidor ou requisições sem sucesso e pode auxiliar em atividades de divulgação (*marketing*), além de proporcionar indícios de melhorias e desenvolvimentos no *web site* [Haigh & Megarity, 1998].

A seguir são apresentadas as características e funcionalidades da ferramenta Fast Stats Analyzer enfatizando que a sua utilização está fortemente vinculada à apresentação gráfica dos resultados obtidos das análises aos arquivos de *log* do servidor.

### 3.4.1 – A Ferramenta Fast Stats Analyzer

A ferramenta Fast Stats Analyzer foi obtida através de ferramentas de busca na internet. Entre diversas outras ferramentas analisadas, tais como W3Perl, Net Racker, WebSat, Max Web Criteria, Bobby, entre outras (ver Seção 2.6.2), esta foi a escolhida para a apresentação de dados dos *log files* por se apresentar como uma ferramenta simples de ser utilizada, que não requisita espaço muito grande de armazenamento em disco, não requisita muita memória na máquina em que for ser executada, e produz relatórios de forma simples, rápida e eficaz.

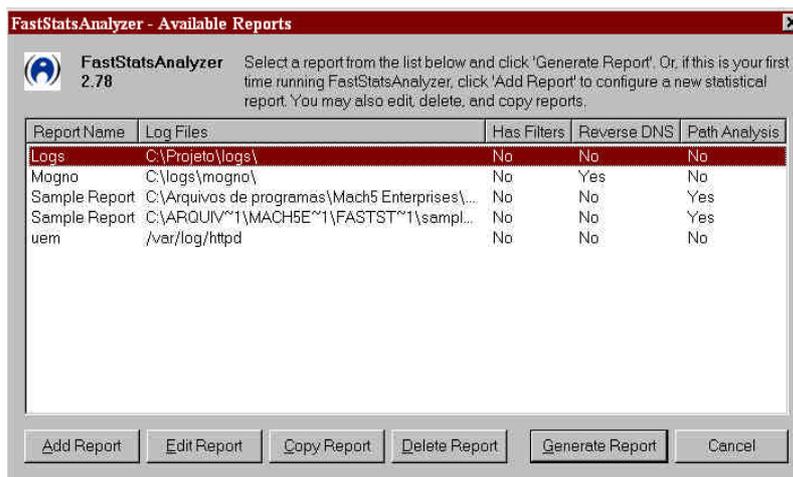
Nesta ferramenta, a solução de acesso aos *log files* é simples, mas requer que seja configurada a localização dos *server log files*, uma vez que duas alternativas são possíveis: (1) os servidores *web* geram seus *log files* utilizando protocolos FTP na internet, tornando-os, dessa forma, disponíveis na internet; (2) caso o *web site* esteja hospedado em um servidor próprio, os *log files* estarão gravados no disco rígido local e somente serão acessíveis através dele (e, nesse caso, pode-se ter problemas de permissão de acesso a tal disco rígido).

Para que a coleta seja feita de maneira correta, algumas configurações são necessárias. Elas são:

1. Se o *web site* estiver hospedado em um servidor próprio, deve-se apenas localizar onde o servidor está salvando os *log files* em seu disco rígido local. Como a maioria dos *log files* possui extensão *.log*, deve-se atentar que podem existir inúmeros arquivos também terminados com a extensão *.log*, mas que tenham outras finalidades. Esta análise deve ser realizada apenas com os *server log files*;
2. Se os *log files* precisarem ser acessados via FTP, deve-se ter acesso ao servidor, e, portanto, esta ferramenta requisitará o “nome” do *site* FTP, além dos possíveis *username*, *password* e caminhos em que os *log files* estão sendo salvos.

Assim, uma vez configurada a localização dos *server log files*, o Fast Stats Analyzer irá buscá-los e, automaticamente, gerar os dados estatísticos relativos à interação. Para facilitar a análise desses dados, a ferramenta possibilita a saída dos dados exportando-os para três tipos de arquivos: *Word*, *Excel* ou *HTML*. Essas saídas facilitam não apenas a obtenção dos resultados, mas também a forma como trabalhar com eles e de como tais saídas do Fast Stats Analyzer podem ser empregadas como entradas para outras ferramentas ou processos.

Cabendo a esta ferramenta a tarefa de realizar a coleta das informações, é possível obter uma análise mais precisa e com maior confiabilidade nos resultados. Pode-se visualizar a tela inicial da ferramenta na Figura 3.3, que apresenta a escolha da localização que se pretende realizar a coleta dos dados.



**Figura 3.3** – Tela Inicial da ferramenta Fast Stats Analyzer

Os testes feitos para a validação desta ferramenta mostraram que seu desempenho diante de grande quantidade de informações é excelente. Em uma análise realizada com 8,8 MB de arquivos de *log*, em um *web site* que recebe 100.000 acessos por dia, a ferramenta gera aproximadamente 39,6 MB de dados relatados nos *logs*. Estes testes foram executados em um Pentium Pro 200 com 96 MB RAM com o Windows NT 4 [Mach 5, 2002].

Na Figura 3.4 pode-se visualizar uma das telas da ferramenta depois de realizada a coleta dos dados, onde são encontradas as imagens mais requisitadas, com o nome do

componente, o número de acessos com sucesso e sem sucesso deste componente como também o tamanho de cada componente em Kbytes.

Image File Name	Hits	Incomplete Requests	Bandwidth
/icons/blank.gif	25	0	1.01 kilobytes
/icons/unknown.gif	25	0	1.67 kilobytes
/icons/back.gif	25	0	1.48 kilobytes
/icons/tolder.gif	18	0	1.54 kilobytes
/icons/image2.gif	14	0	2.11 kilobytes
/imgs/logo-cnc.gif	6	0	12.74 kilobytes
/imgs/openssl.gif	6	0	6.04 kilobytes
/imgs/mod_ssl_sb.gif	6	0	5.88 kilobytes
/imgs/apache_pb.gif	6	0	5.29 kilobytes
/*vnribas/uem.gif	1	0	450.00 bytes
/jpgraph/Examples/tiger_bkg.jpg	1	0	7.91 kilobytes
/jpgraph/Examples/tiger_bkg.gif	1	0	19.20 kilobytes

**Figura 3.4** – Imagem da Ferramenta depois da Coleta dos Dados

Esta ferramenta ainda disponibiliza um número considerável de informações que são divididas nas seguintes categorias:

- **Informações Gerais:** apresenta as características gerais dos acessos como um todo. São medidos neste módulo:
  - a média de dados transferidos por dia, por acesso e por usuário;
  - a média de acessos por dia e por usuário;
  - a média de usuários por dia;
  - o número de acessos por componente;
  - o número de acessos por cliente;

- o número de acessos por página;
  - o número de downloads incompletos;
  - o período de análise dos *logs*;
  - o total de dados transferidos;
  - o total de requisições falhas;
  - o total de usuários;
  - os tipos de arquivos requeridos.
- **Informações dos ambientes dos usuários:** apresenta as características de cada usuário, tais como:
    - acessos recentes;
    - *browser* utilizado;
    - sistema operacional utilizado.
  - **Estatísticas de Acesso:** exibe os dados relativos aos acessos ao *site*, que incluem:
    - páginas mais requisitadas;
    - componentes mais requisitados;
    - dia e hora com maior número de acessos;
    - sistema operacional utilizado por acesso.
  - **Informações Técnicas:** apresenta informações relacionadas às características técnicas do site. Entre elas:
    - número de erros do tipo *404 (missing file)*;
    - componentes em que ocorreram mais erros;
    - número total de erros.

O emprego da ferramenta Fast Stats Analyzer permite que os dados sejam mostrados basicamente de forma bruta, ou seja, da mesma forma como eles são coletados por esta ferramenta. Cabe ao avaliador procurar analisá-los e tirar quaisquer conclusões

quanto à usabilidade do *web site*. Os dados coletados pela ferramenta FastStats Analyzer a partir dos *server log files*, podem ser numerosos, no entanto, em uma análise da usabilidade do *web site*, o analista leva em consideração apenas alguns dos dados relevantes aos componentes pertencentes ao *web site* testado, tais como:

- o número de acessos totais;
- o número de acessos por usuário;
- o número de acessos totais por componente;
- o número de acessos incompletos por componente;
- o número de acessos completos por componente;
- o número e tipo de mensagens de erro;
- o número de *links* com sucesso;
- o número de *links* sem sucesso;
- os tipos de browser utilizados;
- os tipos de arquivos requeridos;

As Figuras 3.5 a 3.8 a seguir apresentam algumas telas que descrevem partes significativas do emprego desta ferramenta, tais como: número de acessos, mensagens de erro.

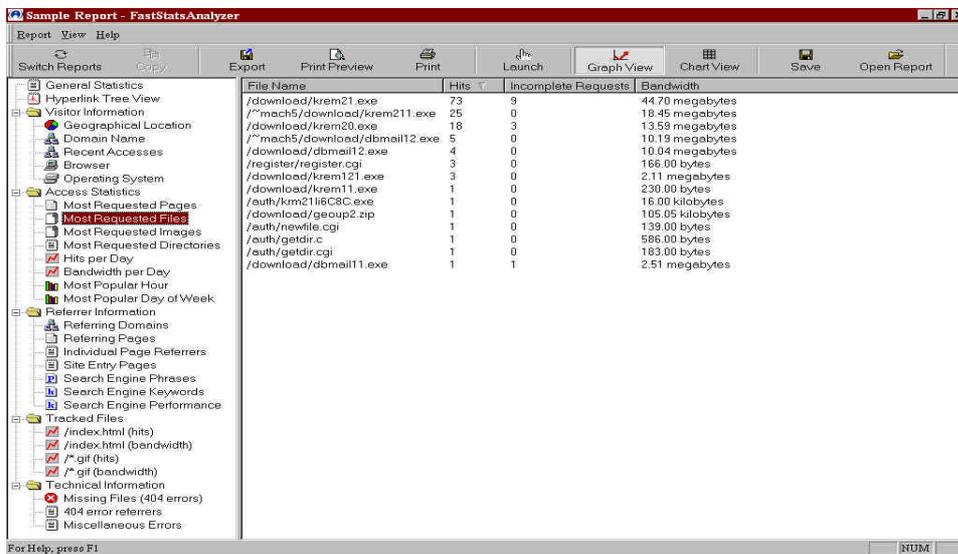


Figura 3.5 – Número de Acessos

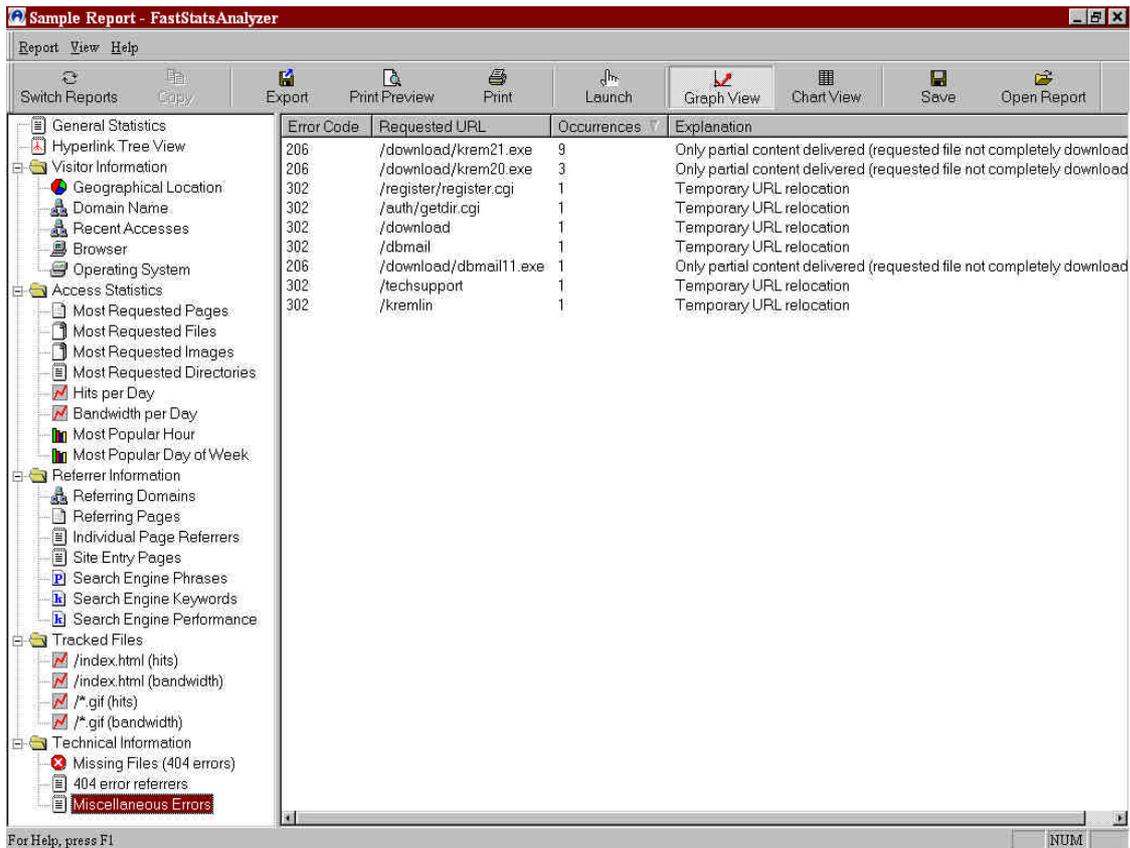


Figura 3.6 – Mensagens de Erro

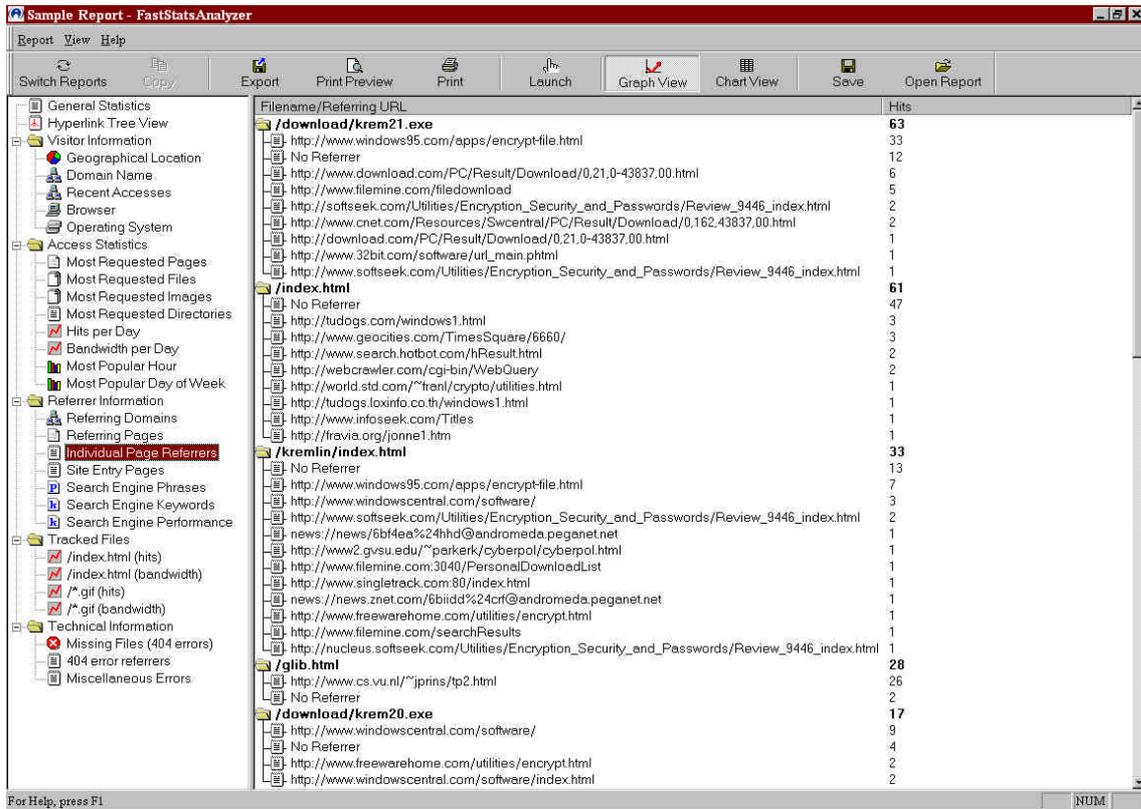
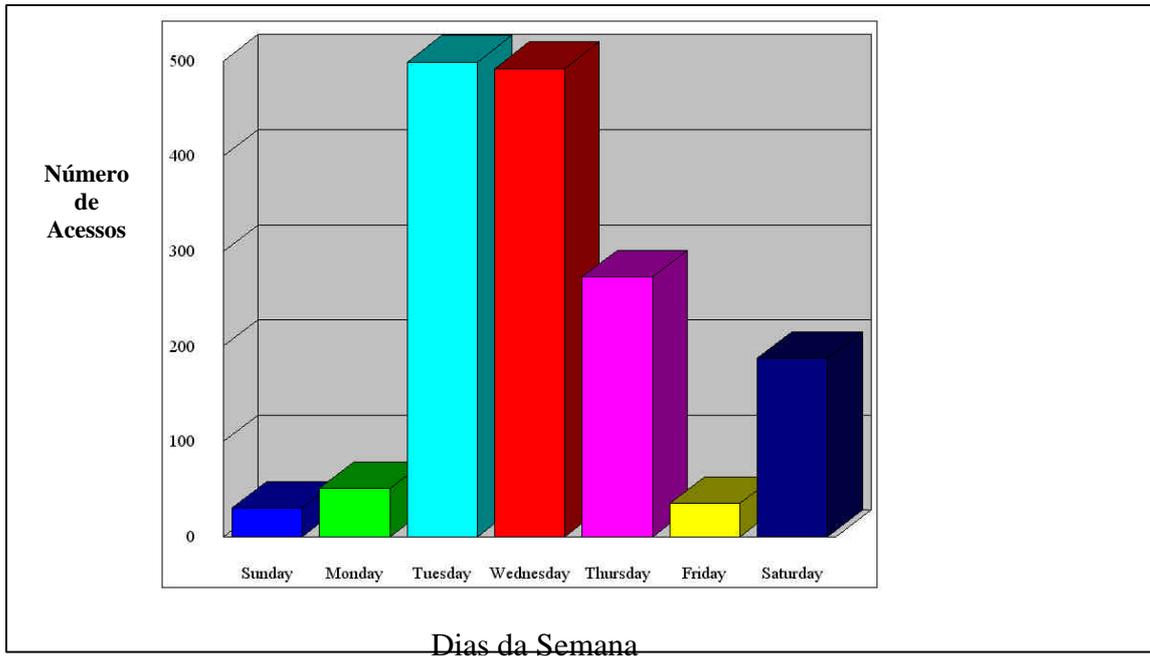


Figura 3.7 – Caminhos Acessados e Quantidades de Acessos por Componentes



**Figura 3.8** – Relacionamento entre Dias da Semana e Número de Acessos

O Capítulo a seguir descreve o ambiente Ergo-Monitor, enfatizando a motivação que levou ao seu desenvolvimento, as ferramentas empregadas na concepção e busca apresentar a forma *como* este ambiente deve ser empregado.

## 4 –Estrutura do Ergo-Monitor

O Ergo-Monitor é proposto como um sistema de monitoramento da usabilidade de *web sites* por meio da coleta seletiva e análise de dados de log (*log files*) referentes às interações reais que se estabelecem entre usuários finais e um *web site* disponibilizado por um servidor *web*. Com base em dados selecionados, o sistema calcula, por meio de modelos de comportamentos esperados para tarefas específicas, as métricas que quantificam a usabilidade efetiva do *web site*. Todos estes procedimentos são realizados de forma “invisível” e “imperceptível” ao usuário, ou seja, o mesmo interage normalmente com o *web site*, e, enquanto suas interações vão ocorrendo, os dados relativos às interações vão sendo armazenados e serão, posteriormente, coletados e analisados.

Esses dados servem, então, como base na elaboração de medidas significativas para uma análise relativa à usabilidade do *web site* e de suas páginas. Além disso, é finalidade do Ergo-Monitor procurar confirmar os diagnósticos inicialmente apontados por um especialista em usabilidade.

Com base nas medidas de usabilidade calculadas pelo Ergo-Monitor, os responsáveis (projetistas) pelos *web sites* devem ser capazes de:

1. providenciar a elaboração de novas versões (projetos) para o *web site* eliminando problemas de usabilidade em interfaces que apóiam algumas atividades específicas (tais como cadastro de usuários, identificação do usuário, busca direta de produtos e compra de produtos); e
2. apoiar a tomada de atitudes gerenciais visando revisões, treinamento de desenvolvedores e manutenções.

Neste Capítulo será apresentada a arquitetura definida para o ambiente Ergo-Monitor, além do estado atual deste ambiente. Algumas ferramentas empregadas estão sendo utilizadas apenas na forma de protótipos e o seu desenvolvimento consta entre as atividades futuras destas ações de pesquisa & desenvolvimento.

Antes dessa apresentação, serão discutidas as motivações básicas que levaram à elaboração deste ambiente, enfatizando as principais dificuldades que são encontradas para

que sejam empregadas ferramentas automatizadas de apoio à atividade de avaliação da usabilidade de sistemas interativos.

#### **4.1 – Motivação da Proposição do Ergo-Monitor**

Os testes de usabilidade são abordagens de avaliação capazes de gerar resultados definitivos, uma vez que pressupõem a participação de usuários utilizando um sistema interativo implementado para a execução de suas tarefas [Nielsen, 2000]. Por outro lado, a aplicação de testes de usabilidade tem seus próprios problemas e dificuldades, destacando-se, segundo [Rocha e Baranauskas, 2000]:

- ruídos, interferências causados pelo constrangimento dos usuários que podem se considerar como sendo o alvo da avaliação e não o *web site*;
- tempo e custo de avaliações de usabilidade, que normalmente são altos devido a necessidade de se empregar especialistas em usabilidade para conduzir a avaliação;
- subjetividade inerente ao próprio processo de avaliação que depende de decisões (*juízo*) de um especialista quanto a condução, coleta e avaliação dos resultados obtidos.

Além disto, existem diversas questões relacionadas à validação dos *web sites* que devem ser indagadas quando do planejamento de qualquer avaliação, e algumas são apresentadas a seguir [Macleod et alli, 1994]:

- Estão sendo analisando os pontos certos? Esta consideração está relacionada à representatividade do uso real em atividades reais nos próprios ambientes de trabalho do sistema interativo sendo avaliado, analisando-se questões como segurança e eficácia das atividades e definindo o escopo da avaliação;
- Estão sendo coletados os dados corretos? Os dados coletados são relevantes para analisar a usabilidade e a qualidade em uso? Os dados quantitativos e qualitativos são facilmente coletados através processos simples e de fáceis de serem utilizados?

- Os dados estão sendo analisados apropriadamente? A análise dos dados deve indicar confiabilidade nos níveis apresentados de usabilidade;
- Estão sendo empregados os usuários certos e/ou em número suficiente? Esta escolha é fundamental para o sucesso da avaliação de usabilidade.

A usabilidade, conforme já destacado anteriormente, não depende apenas das características do sistema interativo, mas principalmente do acordo entre estas características com as dos usuários (e tarefas que os mesmos executam) e do ambiente de trabalho [Rocha e Baranauskas, 2000]. Em particular, constatam-se as dificuldades encontradas ao se tentar avaliar *à priori* a usabilidade de um *web site* por meio de técnicas diagnósticas, uma vez que pouco se sabe sobre o contexto de uma interação que ocorre não se sabe ao certo onde (quais equipamentos e localizações são utilizadas), não se sabe ao certo com quem (qual o perfil dos usuários) e não se sabe ao certo em que condições ambientais, organizacionais e sociais. Pode-se, apenas, esperar um certo padrão mínimo para tais acessos por parte dos usuários, o que já pode determinar algumas características básicas das interações.

A definição do Ergo-Monitor visa reduzir a distância entre o que se espera realmente e o que se pode obter em uma avaliação de usabilidade de *web sites* baseadas na observação da interação. Pretende-se que os laudos finais apresentados por este ambiente sejam provenientes do processamento de dados coletados a partir do uso real do *web site* em seu próprio ambiente de execução cotidiana.

A seção a seguir apresenta algumas dificuldades encontradas na aplicação de ferramentas automatizadas de teste de usabilidade, enfocando que nem sempre os resultados obtidos podem ser considerados como suficientes ou mesmo como de significativa relevância e abrangência para caracterizar a usabilidade do *web site*.

#### **4.1.1 – Principais Dificuldades para o Emprego de Ferramentas Automatizadas**

No Capítulo 2, ao serem introduzidas as principais características que se esperam das ferramentas automatizadas de apoio à avaliação da usabilidade de *web sites*, procurou-se descrever o que se espera como resultado do emprego das mesmas e como elas podem auxiliar os avaliadores na coleta de dados que sirvam como parâmetros para as avaliações.

Entretanto, o emprego dessas ferramentas pode estar sujeito às falhas ou dificuldades de condução devido a diversos fatores que serão apresentados a seguir. Essas falhas e dificuldades foram incentivadoras da proposição desta tese de doutorado, uma vez que se procurou, com o desenvolvimento do Ergo-Monitor, reduzir (ou preferencialmente eliminar) os problemas encontrados, que são:

1. Em termos de análise qualitativa, as ferramentas não oferecem grande apoio. Algumas delas disponibilizam relatórios com a seqüência cronológica das ações ou eventos capturados, mas que na maioria das vezes não possuem um formato que possa efetivamente auxiliar na análise das informações;
2. Constatou-se que a automação do procedimento de análise dos dados capturados restringe-se basicamente à utilização da técnica de medição de desempenho dos usuários durante a interação. As ferramentas calculam taxas e as apresentam aos avaliadores que dependem de suas próprias experiências para a elaboração de seus diagnósticos;
3. Não foram encontradas ferramentas que possibilitaram a comparação de interações de diferentes usuários sobre o mesmo *web site* e suas páginas.;
4. As ferramentas pesquisadas não dispõem de recursos que possibilitem ao avaliador manipular as interações, filtrando apenas as que o interessem. Tal funcionalidade é bastante importante, pois as ações nem sempre são sucintas e relativas apenas ao objeto de avaliação e uma gama muito grande de dados pode atrapalhar a análise dos dados coletados;
5. Mesmo com o emprego de ferramentas automatizadas, existe uma dependência muito grande da capacidade do avaliador em entender os dados coletados e somente ele/ela pode apresentar laudos concludentes;
6. Não foi encontrada uma definição de taxas específicas às considerações que sejam basicamente relativas à usabilidade de *web sites*;

As definições do ambiente Ergo-Monitor visam minimizar esses e outros problemas que podem ser significativos quando da condução e eficácia nas avaliações de usabilidade de *web sites*.

## 4.2 – O Ergo-Monitor

Nesta seção serão apresentadas as características do ambiente Ergo-Monitor, enfatizando suas utilizações e as ferramentas de suporte a este sistema. Serão apresentados, ainda, os conceitos básicos que nortearam o desenvolvimento dessas ferramentas e as características de utilizações das mesmas.

O Ergo-Monitor é proposto como um sistema capaz de monitorar interações *reais* com usuários *reais* e que estejam *realmente* utilizando o *web site* em seus *próprios* ambientes de trabalho. Isso define o foco de atenção das avaliações de usabilidade: problemas de usabilidade que ocorrem efetivamente, sem que se tenha o conhecimento pleno das condições de contexto de uso envolvidas com tais problemas. Existe basicamente uma procura por indicadores da existência dos problemas nas interações, mesmo que não seja possível saber quem é o usuário. Sob outro ponto de vista, o Ergo-Monitor é capaz de identificar, por meio de análises quantitativas, problemas nas interações que se estabelecem entre usuários e *web sites*. As análises qualitativas das causas destes problemas na interface com o usuário podem já haver ocorrido, nos casos em que o Ergo-Monitor é usado para confirmar um diagnóstico, ou podem ocorrer futuramente, em função das métricas quantitativas oferecidas pelo Ergo-Monitor. De qualquer forma, as análises qualitativas estão além do escopo deste sistema, e podem ser empregadas basicamente como um mecanismo de apoio à análise dos resultados apresentados pelo Ergo-Monitor, uma vez que uma avaliação diagnóstica qualitativa pode apresentar indícios de problemas que podem ser melhor investigados.

Outra característica marcante do Ergo-Monitor é a invisibilidade para o usuário, o que permite minimizar, ou até mesmo eliminar, ruídos e interferências que sempre fazem parte de qualquer avaliação de usabilidade.

Para tanto, o Ergo-Monitor baseia o seu funcionamento no monitoramento da usabilidade de *web sites* por meio da coleta seletiva e análise de *log files* referentes às interações reais que se estabelecem entre usuários finais e um *web site*, quantificando métricas da usabilidade efetiva dos mesmos. É importante frisar que estas métricas se referem a interações em tarefas especiais que podem ser classificadas de “fechadas” ou “objetivas”. Estas são tarefas que possuem pontos de início e final claramente definidos e

onde a eficácia (o sucesso e o fracasso, seja por cancelamento ou desistência) e a eficiência (o retrabalho e as interrupções por mensagens de erro ou busca de ajuda) possam ser determinadas de forma objetiva e sem ambigüidades, através da análise dos caminhos trilhados pelo usuário durante as interações com o *web site*. Especificamente, em um *site* de comércio eletrônico, tarefas objetivas incluem o cadastro de usuários, a obtenção de acesso, a busca direta de produtos, a compra de produtos, entre outras.

O avaliador-operador do Ergo-Monitor deve ter uma participação ativa nos seguintes momentos:

- Em revisões no *web site* a ser monitorado. Esta intervenção é opcional, e envolve eventualmente, a implementação de páginas com mensagens de sucesso, mensagens de erro e mensagens de ajuda. Tais recursos, cuja implementação não deve interferir na usabilidade do sistema, servem de marcadores de comportamentos de sucesso, de cancelamento, de desistência, e de ajuda realizados por parte dos usuários,
- Na configuração do Ergo-Monitor, de modo a:
  - Definir os parâmetros de ‘serviço’, como endereço dos *log files*, tipo de servidor *web*, etc;
  - Modelar as tarefas e comportamentos (de sucesso, de cancelamento, de desistência, de ajuda, ...) a serem monitoradas em cada tarefa. Esta modelagem é realizada basicamente com a definição das seqüências de *urls* que o Ergo-Monitor irá monitorar;
- Na interpretação da usabilidade do sistema monitorado, a partir das métricas calculadas pelo Ergo-Monitor. Esta atividade pode se enquadrar em um processo de confirmação de um diagnóstico qualitativo prévio de problema de usabilidade ou disparar a realização de futuras análises qualitativas da interface do sistema em função de métricas de usabilidade determinadas pelo Ergo-Monitor e consideradas inaceitáveis;

- Na montagem de uma base de medidas históricas de métricas de usabilidade. Estas podem ser usadas para monitorar a evolução da usabilidade de um sistema e definir valores de referência. Por exemplo: usabilidade normal, aceitável, e inaceitável para o sistema.

Dentre os dados que se pretende coletar após as interações terem sido realizadas sob o monitoramento do Ergo-Monitor, encontram-se:

- páginas acessadas no *web site*;
- tempo de permanência no *web site* de forma geral,
- tempo de permanência para cada página;
- mensagens de erro apresentadas (se existirem); e
- páginas de ajuda acessadas (se existirem).

A partir deles, e com base em uma modelagem de comportamentos dos usuários em tarefas com início e fim bem definidos, é possível contabilizar comportamentos ligados à *eficácia* na interação, tal como:

- quantidade de sucessos em tarefas fechadas (início e fim com mensagem explícita).
- quantidade de desistências em tarefas fechadas (comportamento de entrada e saída sem mensagem de sucesso);
- quantidade de cancelamentos em tarefas fechadas (comportamento de entrada e saída sem sucesso da tarefa, após mensagem de erro);

Da mesma forma, é possível contabilizar comportamentos ligados a *eficiência* na interação, tais como:

- quantidade de repetições em tarefas fechadas (comportamento com mais de um início para uma mesma tarefa, antes de ocorrer uma mensagem de sucesso); e
- quantidade de interações consideradas improdutivas, como o tratamento de erros e a busca de ajuda.

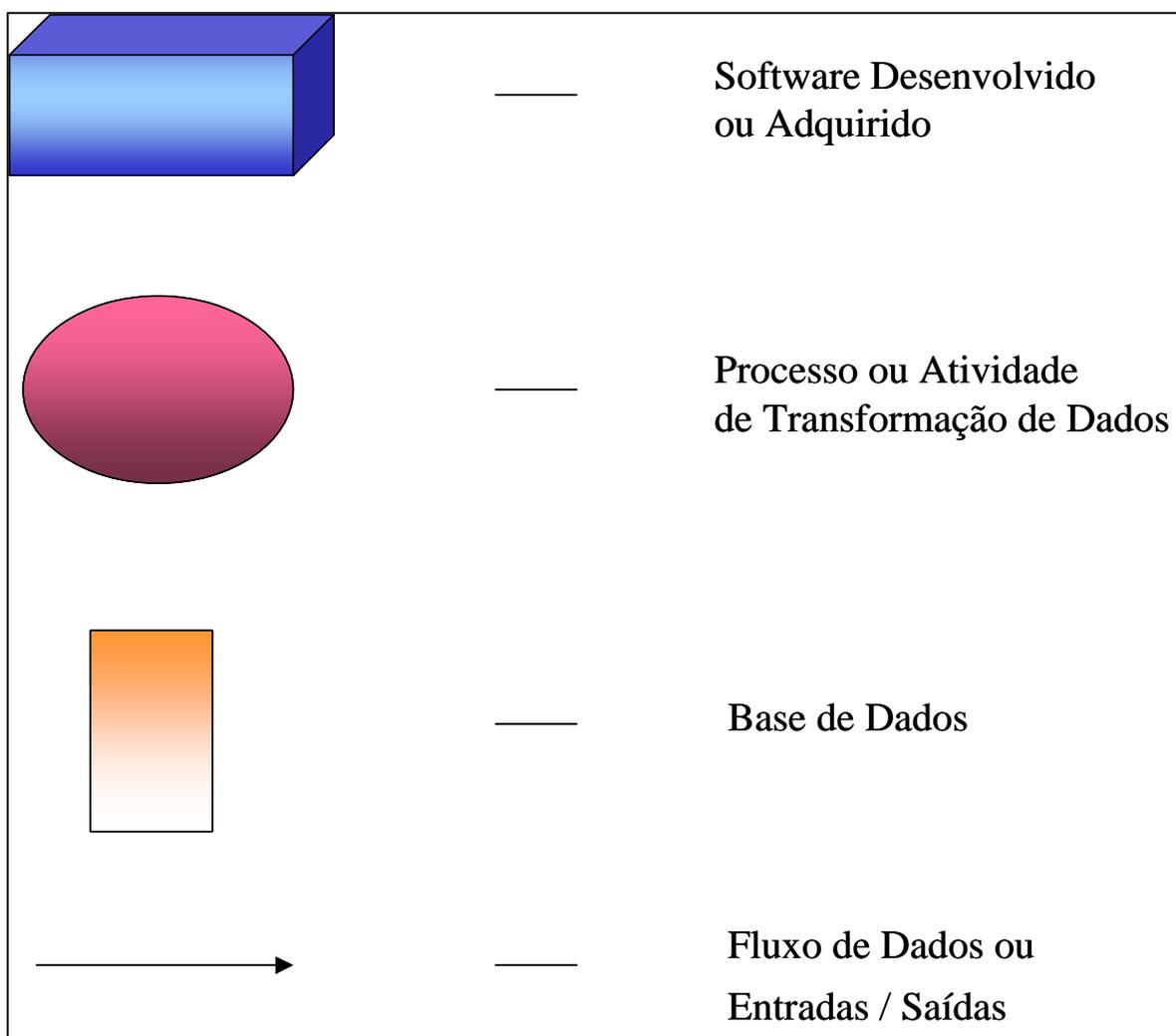
Deve-se destacar que a proposição atual do Ergo-Monitor examina apenas os dados provenientes dos *log files*, uma vez que os *cookies* não foram considerados, não apenas por motivos de praticidade, mas, principalmente, pelo fato das informações obtidas a partir dos *server log files* já serem suficientes para provar a viabilidade do sistema. Entretanto, para versões futuras deste ambiente, pode-se prever a análise dos *cookies* como forma de confirmação dos dados provenientes dos *log files* e de realização de análises mais finas das interações usuário final – *web site*. Também por simplicidade, em sua proposição atual, o Ergo-Monitor associa um único endereço IP a um único usuário (e vice-versa), apesar de que, em algumas situações, mais de um usuário podem interagir com um *web server* a partir de um mesmo número IP.

Para que o Ergo-Monitor seja capaz de gerar resultados ainda mais significativos, uma avaliação de usabilidade diagnóstica prévia do *web site* pode ser realizada empregando-se as técnicas e metodologias tradicionais, tais como as apresentadas no Capítulo 2. Essa avaliação prévia irá possibilitar uma melhor interpretação dos resultados (numéricos) apresentados nos relatórios produzidos pelo Ergo-Monitor.

A seguir serão apresentadas as etapas que devem ser percorridas para a aplicação do Ergo-Monitor dando suporte à avaliação da usabilidade de *web sites*.

#### **4.2.1. Etapas de Aplicação do Ergo-Monitor**

A Figura 4.1 apresenta a legenda das entidades empregadas nas figuras que serão apresentadas neste Capítulo. São elas: (1) *caixa*: representa um software que pode ter sido implementado especificamente para este trabalho; (2) *elipse*: representa alguma atividade específica de transformação de dados, visando a produção de novos dados (tal como é o caso da atividade de Determinação de Taxas de Usabilidade, que tem como entrada o Modelo de Tarefas e Comportamentos Verificados e produz Taxas de Usabilidade específicas que quantifiquem as interações; (3) *retângulo*: que representa uma base de dados (tal como a Base de Dados de Parâmetros Admissíveis); e, finalmente (4) *seta*: que representa fluxos de dados (tal como os Dados sobre os *Log Files*).

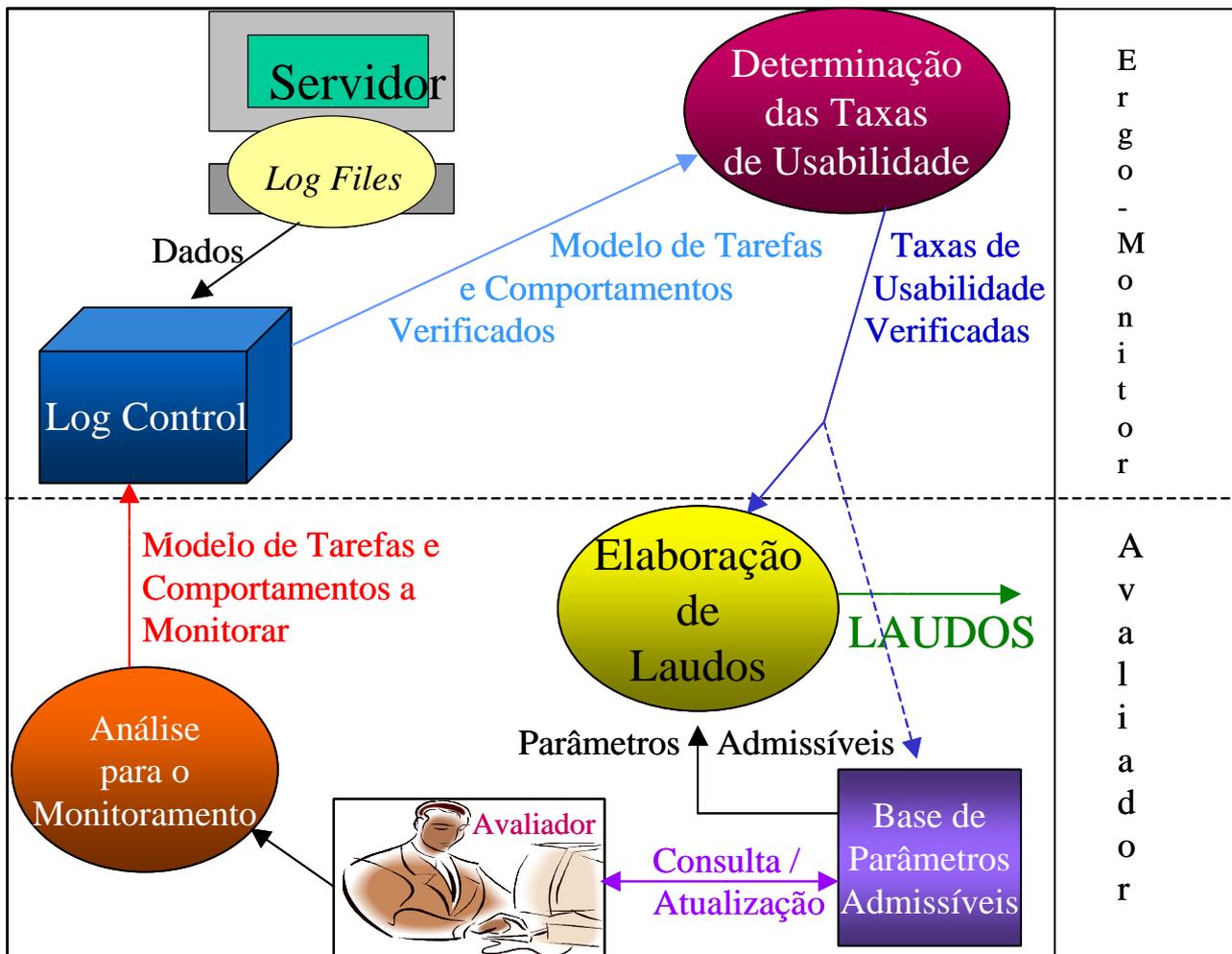


**Figura 4.1** – Legenda das Estruturas Apresentadas na Descrição do Ergo-Monitor

A Figura 4.2 ilustra uma descrição geral das atividades que se passam no ambiente Ergo-Monitor a partir da coleta de dados dos *server log files*, passando pela geração das taxas e métricas, e concluindo com a elaboração de laudos de monitoramento. O grupo de programas denominado Log Control faz a filtragem dos *log files* para eliminação de informações não significativas face o monitoramento da usabilidade do *web site*. Além dos *log files*, a outra entrada para o Log Control é um *Modelo de Tarefas e Comportamentos a Monitorar*. Cabe ao analista-operador do Ergo-Monitor montar este modelo, tendo partindo da observação do *web site* e com base em um eventual diagnóstico de problemas de usabilidade. Este modelo deve ser composto por tarefas fechadas, com início e fim bem definidos, as quais estarão associados comportamentos indicativos da eficácia e da eficiência das interações para a sua realização (sucesso, desistência, cancelamento,

retrabalho, etc.). A saída esperada do Log Control é uma contabilidade da ocorrência de cada comportamento, na forma de um *Modelo de Tarefas e Comportamentos Verificados*, que serve como base para a determinação de taxas e métricas que quantifiquem as interações.

O próximo passo é determinar **taxas** e **métricas** que quantifiquem a qualidade da eficácia e eficiência das interações. Nesse contexto, uma **taxa** é uma medida intermediária da qualidade da interação (taxa de sucesso, de desistência, de cancelamento, etc), enquanto **métrica** representa uma medida de usabilidade, conforme estabelecida pela norma ISO 9241:11 (basicamente as métricas de eficácia e eficiência) [ISO 9241-11]. Os valores verificados para as taxas e métricas associadas às interações são, então, confrontados com parâmetros admissíveis determinados por especialistas, que analisam e quantificam interações entre diferentes tipos de usuários de *sites* de boas e más interfaces. É importante ressaltar que os valores admissíveis devem ser estabelecidos para interações de um mesmo tipo. Por meio do confronto entre o verificado e o admissível, o avaliador é capaz de elaborar um laudo de monitoramento que serve tanto para confirmar a existência de problemas de usabilidade previamente diagnosticados, como para apontar para a existência de outros que deverão ser analisados posteriormente.



**Figura 4.2** – Visão Geral do Ergo-Monitor

Esta figura apresenta as atividades que devem ser realizadas pelo avaliador e as que o próprio ambiente Ergo-Monitor realiza. Dentre as atividades que são realizadas pelo ambiente, destaca-se a coleta e a análise dos dados provenientes dos arquivos de *log*, enquanto que entre as atividades que são de responsabilidade do avaliador, podem ser destacadas a Análise para o Monitoramento (onde o avaliador irá definir os arquivos utilizados como parâmetros de entrada para o Log Control) e a Elaboração de Laudos.

A seção a seguir apresenta as etapas que caracterizam o processo do ambiente Ergo-Monitor para que o monitoramento das tarefas possa ser efetuado. O processo se inicia com as análises e ajustes necessários para que se tenham tarefas fechadas, objetivas, e principalmente, monitoráveis (cujos rastros fiquem armazenados no arquivo de *log*).

#### 4.2.1.1 – Análise para o Monitoramento

O primeiro passo para o monitoramento de interações se refere à análise do que pode e do que não pode ser monitorado pelo Ergo-Monitor. Como mencionado anteriormente, as ferramentas deste ambiente dependem da identificação de tarefas e comportamentos claramente observáveis. De um lado, é necessário que as tarefas tenham características de fechamento e de objetividade, sendo possível determinar claramente o início e final da tarefa, seja com sucesso ou o fracasso por parte usuário. As marcas de início, fim, sucesso ou fracasso, além da busca pela ajuda ou o enfrentamento de um erro devem ficar armazenadas nos arquivos de *log* do servidor *web*, e, para tanto, devem ser apresentadas em uma *url* específica. Estas marcas podem já ser características dos *sites* ou este pode ter de vir a sofrer alterações de modo a que elas existam e fiquem registradas. Tipicamente, nesta etapa, o avaliador deve fazer uma análise de quais tarefas são do tipo fechadas e objetivas e caso necessário, providenciar “marcas” que fiquem registradas no arquivo de *log*, permitindo assim, verificar claramente determinados comportamentos (sucesso, desistência, cancelamento, ...) nas tarefas. Além de pertinentes para a usabilidade do *site*, ajustes como a implementação de mensagens de sucesso, de erro e de ajuda em *urls* específicas são de fundamental importância para que o monitoramento possa ser viabilizado para um grande número de tarefas.

O resultado prático desta etapa é a definição de um *Modelo de Tarefas e Comportamentos a Monitorar*, a partir da observação das características das tarefas com o *web site* e com base em um eventual diagnóstico de problemas de usabilidade. Este modelo será descrito basicamente por meio de seqüências de *urls*.

No *site* empregado para o estudo de caso realizado com um protótipo do Ergo-Monitor as seguintes tarefas foram identificadas como fechadas ou objetivas:

- Cadastro de usuário;
- Identificação de usuário (*login*);
- Trocar senha;
- Busca direta de produtos;
- Busca direta avançada de produtos;

- Efetivar compra (depois que a compra foi decidida);
- Enviar reclamação ou sugestão (fale conosco).

Para este conjunto de tarefas foi definido o seguinte conjunto de comportamentos verificáveis:

- Acessos na Tarefa: ocorrência da *url* de uma página que marque o início de uma tarefa;
- Sucesso na Tarefa: seqüência de páginas demarcando a realização de uma tarefa com sucesso. O comportamento se inicia com a entrada em uma *url* (p.ex. Cadastro de usuários) e tem seu final com a passagem para uma outra que é dependente do sucesso na tarefa (Mensagem de sucesso).
- Desistência na Tarefa: seqüência de páginas demarcando a desistência em uma tarefa. O comportamento se inicia com a entrada em uma *url* (p.ex. Cadastro de usuários) e tem seu final com a passagem para uma outra que NÃO esteja ligada ao sucesso na tarefa (volta para a página inicial sem passar pela Mensagem de Sucesso);
- Abandono na Tarefa: seqüência de páginas demarcando a desistência em uma tarefa. O comportamento se inicia com a entrada em uma *url* (p.ex. Busca Avançada) e se desenvolve em outra *url* definida (p.ex. Resultado da Busca) e tem seu final com a passagem para uma outra que NÃO esteja ligada ao sucesso na tarefa (não acessa a visualização e nem a compra do produto);
- Cancelamento na Tarefa: seqüência de páginas demarcando o cancelamento em uma tarefa após a ocorrência de um erro. O comportamento se inicia com a entrada em uma *url* (p.ex. Cadastro de usuários) e tem seu final com a passagem para uma outra que NÃO esteja ligada ao sucesso na tarefa (volta para a página inicial sem passar pela Mensagem de Sucesso) APÓS a apresentação de uma mensagem de erro;

- Repetição da tarefa: seqüência de páginas demarcando a repetição de uma tarefa. O comportamento apresenta mais de uma passagem pela página que demarca o início da tarefa e tem seu final com a passagem para uma *url* que é dependente do sucesso na tarefa (Mensagem de sucesso).
- Busca de Ajuda na Tarefa: seqüência de páginas demarcando a busca de ajuda em uma tarefa. O comportamento se inicia com a entrada em uma *url* (p.ex. Cadastro de usuários) e tem a qualquer momento uma passagem pela *url* de páginas ajuda.
- Enfrentamento de Incidente na Tarefa: seqüência de páginas demarcando a ocorrência de incidente em uma tarefa. O comportamento se inicia com a entrada em uma *url* (p.ex. Cadastro de usuários) e tem a qualquer momento uma passagem pela *url* de mensagens de erro.

Na versão atual do Ergo-Monitor, podemos caracterizar os seus usuários principais como *especialistas em informática*, uma vez que se faz necessário que este usuário informe as páginas (*urls*) que podem ser acessadas quando da interação com o *site* para a realização das tarefas a serem monitoradas. Tais informações só podem ser fornecidas por usuários que conheçam um pouco a respeito das principais linguagens e conceitos de programação envolvidos na elaboração de *web sites*, além de ser necessária a definição de quais informações devem ser filtradas a partir dos *log files* de modo a serem representativas no que diz respeito à usabilidade do mesmo.

A próxima seção descreve como os dados provenientes dos *log files* devem ser filtrados, bem como o mecanismo que deve ser empregado para a determinação dos comportamentos verificados nas avaliações.

#### **4.2.1.2 – Filtragem de Dados de Log e Determinação de Comportamentos Verificados**

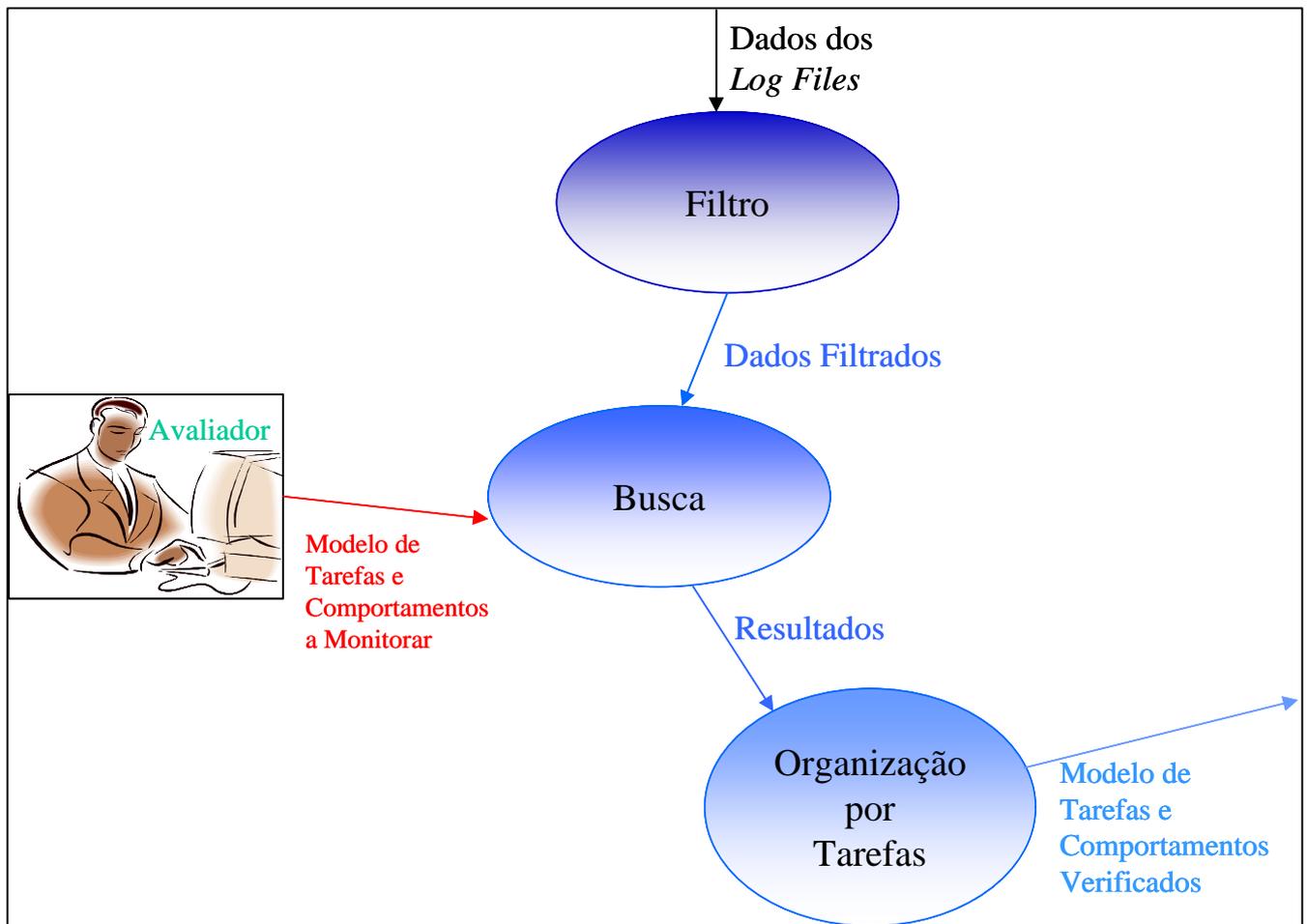
Esta seção apresenta a fase de filtragem dos dados provenientes dos *log files* que devem ser mapeados e analisados visando a determinação de taxas que venham a quantificar a usabilidade do *web site*.

A Figura 4.3 ilustra uma descrição das atividades da primeira etapa de processamentos do Ergo-Monitor, apresentando a coleta de dados a partir dos *server log files* até a determinação de um modelo de tarefas e comportamentos verificados no web site. O grupo de programas denominado Log Control é dividido nas seguintes etapas:

- filtragem dos *log files* para eliminação de informações não significativas face o monitoramento da usabilidade do *web sit*. Este arquivo apresenta apenas as *urls* das páginas acessadas durante a interação;
- busca no arquivo gerado na etapa anterior, pela ocorrência de *urls* que constam do Modelo de tarefas e comportamentos a monitorar;
- organização dos dados sobre a ocorrência de comportamentos nas tarefas de forma a facilitar a determinação das taxas de usabilidade do *web site*.

Logo, além dos *log files*, a outra entrada requisitada pelo Log Control é o *Modelo de Tarefas e Comportamentos a Monitorar*, definido pelo analista-operador do Ergo-Monitor. Ele (ela) irá montar este modelo a partir da observação do *web site* e com base em um eventual diagnóstico de problemas de usabilidade. A saída esperada do Log Control é uma contabilidade de cada um desses comportamentos, organizada segundo as taxas de usabilidade.

A Figura 4.3 ilustra uma descrição das atividades do Log Control, mostrando suas entradas (Dados do Log File e Modelo de Tarefas e Comportamentos a Monitorar) e a saída esperada (Modelo de Tarefas e Comportamentos Verificados). Maiores detalhes acerca do Log Control encontram-se na Seção 4.2.2.



**Figura 4.3** – Descrição do Log Control

#### 4.2.1.3 – Cálculo das Taxas de Usabilidade

O próximo passo a ser executado pelo Ergo-Monitor é a determinação das taxas que quantificam a qualidade das interações (eficácia e eficiência) em tarefas fechadas ou objetivas. Estando contabilizadas as ocorrências destes comportamentos, o sistema é capaz de calcular as seguintes Taxas de Usabilidade na Tarefa:

- Taxa Individual de Sucessos na Tarefa: quantidade de comportamentos de Sucesso sobre a quantidade de Acessos para UM usuário;
- Taxa Individual de Desistências na Tarefa: quantidade de comportamentos de Desistência sobre a quantidade de Acessos para UM usuário;

- Taxa Individual de Abandonos na Tarefa: quantidade de comportamentos de Abandono sobre a quantidade de Acessos para UM usuário;
- Taxa Individual de Cancelamentos na Tarefa: quantidade de comportamentos de Cancelamentos sobre a quantidade de Acessos para UM usuário;
- Taxa Individual de Repetições da tarefa na Tarefa: quantidade de comportamentos de Repetição sobre a quantidade de Acessos para UM usuário;
- Taxa Individual de Busca de Ajuda na Tarefa: quantidade de comportamentos de Busca de Ajuda sobre a quantidade de Acessos para UM usuário; e
- Taxa Individual de Enfrentamento de Incidentes na Tarefa: quantidade de comportamentos de Enfrentamento de Incidente sobre a quantidade de Acessos para UM usuário.

Observa-se que nem todas as taxas podem ser calculadas para todas as tarefas listadas acima, pois, para algumas delas, tais taxas nada representariam ou simplesmente não seriam capazes de serem calculadas. Exemplos dessas taxas são: (1) taxa de repetição de tarefas para buscas - não se pode saber o que especificamente está sendo buscado apenas com os dados coletados dos *log files*, e, portanto, o usuário pode estar realizando duas buscas diferentes o que em hipótese alguma caracteriza uma repetição de tarefa; ou (2) taxa de cancelamentos para buscas, que não pode ser calculada uma vez que normalmente ao ser disparada uma busca, não se pode abortar tal atividade sem o subterfúgio do emprego dos botões de controle do próprio *browser*.

O passo seguinte é calcular as Taxas Coletivas, como a média das taxas individuais, descartando as que se encontram fora do desvio padrão da amostra. Assim, podem ser determinadas:

- Taxa Coletiva de Sucessos na Tarefa: quantidade de comportamentos de Sucesso sobre a quantidade de Acessos para  $(N - \sigma)$  usuários ( $\sigma$  representa o número de usuários que estão fora do desvio padrão, e, portanto, devem ser eliminados);
- Taxa Coletiva de Desistências na Tarefa: quantidade de comportamentos de Desistência sobre a quantidade de Acessos para  $(N - \sigma)$  usuários;
- Taxa Coletiva de Abandonos na Tarefa: quantidade de comportamentos de Abandono sobre a quantidade de Acessos para  $(N - \sigma)$  usuários;
- Taxa Coletiva de Cancelamentos na Tarefa: quantidade de comportamentos de Cancelamentos sobre a quantidade de Acessos para  $(N - \sigma)$  usuários;
- Taxa Coletiva de Repetições da tarefa na Tarefa: quantidade de comportamentos de Repetição sobre a quantidade de Acessos para  $(N - \sigma)$  usuários;
- Taxa Coletiva de Busca de Ajuda na Tarefa: quantidade de comportamentos de Busca de Ajuda sobre a quantidade de Acessos para  $(N - \sigma)$  usuários; e
- Taxa Coletiva de Enfrentamento de Incidentes na Tarefa: quantidade de comportamentos de Enfrentamento de Incidente sobre a quantidade de Acessos para  $(N - \sigma)$  usuários.

As taxas coletivas são as taxas que representam a usabilidade e serão comparadas com os valores admissíveis para a usabilidade, conforme processo descrito no próximo tópico.

O Ergo-Monitor é também capaz de calcular métricas de eficácia e eficiência para a tarefa, seguindo as definições fornecidas pela norma ISO 9241, parte 11. Para tanto ele contabiliza o seguinte evento:

- Usuário com Sucesso na Tarefa: ocorrência do comportamento Sucesso na Tarefa para um determinado IP

A partir destes eventos o sistema calcula:

- Eficácia na tarefa: porcentagem de usuários com sucesso pelo número total de acessos na tarefa; e
- Eficiência na tarefa: média dos tempos do comportamento de sucesso na tarefa.

Atualmente, no contexto de utilização do Ergo-Monitor estas duas métricas não estão sendo calculadas, mas podem ser facilmente implementadas em versões futuras deste sistema.

#### **4.2.1.4 – Elaboração de Laudos**

A partir do confronto entre taxas coletivas e métricas verificadas e os valores admissíveis para a usabilidade em tarefas fechadas e objetivas, o Ergo-Monitor é capaz de apoiar a elaboração de dois tipos de laudos:

- um confirmando a existência de problemas de usabilidade diagnosticados previamente por técnicas de avaliação qualitativas; ou
- um outro suscitando a realização de avaliações qualitativas futuras.

A proposição do Ergo-Monitor é a de proporcionar o confronto entre seus diagnósticos quantitativos e os diagnósticos qualitativos fornecidos pela aplicação de técnicas diagnósticas a priori.

#### **4.2.1.5 – Montagem de uma Base de Parâmetros Admissíveis para a Usabilidade**

Esta atividade se desenvolve de maneira síncrona ao monitoramento da usabilidade de um *web site*. Os resultados do Ergo-Monitor, em termos de taxas verificadas de usabilidade do *site* monitorado, alimentam a base de valores admissíveis para usabilidade, e estas fornecem valores de referência contra os quais confrontar-se a sua usabilidade efetiva. As atividades desta fase, que são de responsabilidade do avaliador, incluem:

- a observação e coleta de dados sobre a interação entre web sites e usuários reais,
- a análise e a determinação de taxas de usabilidade;
- o cadastro e manutenção destas taxas em uma Base de Parâmetros Admissíveis para a usabilidade de *web sites*.

Esta base tem uma característica bastante importante: ela é incremental; ou seja, está em constante atualização. Com isso, a cada nova avaliação realizada, haverá uma manutenção nos dados dessa base. Por outro lado, o especialista utiliza os dados armazenados em uma Base de Parâmetros Admissíveis para apoiar suas decisões sobre a usabilidade do *web site* em monitoramento.

É importante destacar que os parâmetros admissíveis estão ligados a seus contextos específicos. Assim, os *web sites* de comércio eletrônico irão apresentar um conjunto de parâmetros admissíveis no que diz respeito a diversas taxas como: erros, repetições de tarefas ou tempos de resposta, de modo totalmente diferente dos que são relativos a conjuntos de *web sites* de jogos, notícias, ou de outra categoria. Logo, os dados da base de parâmetros admissíveis devem estar categorizados por contextos de aplicação específicos, e faz parte da tarefa do avaliador categorizar o contexto onde se encontra o *web site* que está sendo monitorado.

O foco do Ergo-Monitor é o monitoramento da usabilidade de *web sites* para comércio eletrônico do tipo C2B (consumidor x empresa), apesar de haver diversos contextos de aplicação dentro da categoria de comércio eletrônico. Por exemplo, os *web sites* de restaurantes, de bancos, de lojas, de livrarias, etc. Logo, a determinação dos padrões de aceitação pode estar bastante dependente dos próprios *web sites*, e uma definição de valores próprios para a definição de “boas” ou “más” interações deve ser realizada após uma série de avaliações de *web sites* similares e análises dos resultados obtidos anteriormente. Entretanto, a aplicação deste ambiente para outros *web sites* que estejam em outras categorias depende, basicamente, de uma análise de outros *web sites* dessa mesma categoria para que os Parâmetros Admissíveis possam ser determinados.

De forma geral, o próprio ambiente Ergo-Monitor, por meio do componente Log Control, pode ser usado para a determinação de valores admissíveis, na medida em que, como descrito até agora, ele é capaz de gerar taxas de usabilidade verificada para *sites* monitorados. Para esta tese de doutorado, entretanto, houve a determinação manual de alguns valores, somente para mostrar a viabilidade do ambiente. Foram avaliados alguns *web sites* de comércio eletrônico de boa qualidade para empresas conceituadas como Livraria Saraiva, Lojas Ponto Frio, Fnac, Alô Bebê, entre outras e realizados ensaios de interação com 09 usuários pertencentes a população alvo destes *web sites*. Foram tomadas as devidas precauções para a escolha desse usuários de forma a empregar pessoas que possuam níveis de capacitação experientes, medianos ou iniciantes nas tarefas monitoradas e também na própria utilização da *internet*. O Apêndice D mostra as tabelas (*Excel*) elaboradas com os dados observados dessas interações.

As Tabelas 4.1 a 4.2 apresentadas a seguir reportam os resultados obtidos desses ensaios, especificamente para as tarefas de cadastro e compras.

**Tabela 4.1** – Valores de Referência para a Busca de Ajuda no Cadastro

	Saraiva	Tempo Real	Ponto Frio	Fnac	Shop Fácil	America nas	Alô Bebê	Média do usuário
Usuário 1	00	01	00	00	00	02	00	0,43
Usuário 2	01	00	00	00	00	01	00	0,29
Usuário 3	01	01	00	00	00	01	00	0,43
Usuário 4	02	01	00	00	00	00	00	0,43
Usuário 5	01	01	00	00	00	00	00	0,29
Usuário 6	00	00	00	00	00	01	00	0,14
Usuário 7	01	00	00	00	00	01	00	0,29
Usuário 8	01	01	00	00	00	01	00	0,43
Usuário 9	01	01	00	00	00	02	00	0,57
Média do site	0,89	0,67	0,0	0,0	0,0	1	0,0	0,37

Segundo a Tabela 4.1, os *sites* da Livraria Saraiva, das lojas Tempo Real e Americanas proporcionam interações não tão boas para o cadastro de usuários. Assim, para este tipo de tarefa, um parâmetro admissível para poderia ser de 1 busca de ajuda.

**Tabela 4.2** – Valores de Admissíveis para os Enfrentamentos de Erros no Cadastro

	Saraiva	Tempo Real	Ponto Frio	Fnac	Shop Fácil	America nas	Alô Bebê	Média do usuário
Usuário 1	02	01	02	02	02	02	01	1,71
Usuário 2	03	01	01	02	01	02	01	1,57
Usuário 3	02	01	03	03	03	04	04	2,86
Usuário 4	02	01	02	01	01	01	02	1,43
Usuário 5	01	02	02	03	03	03	03	2,43
Usuário 6	01	01	01	01	01	01	01	1,00
Usuário 7	02	02	02	02	02	02	02	2,00
Usuário 8	01	01	01	01	01	01	01	1,00
Usuário 9	02	01	02	02	01	01	01	1,43
Média do site	1,78	1,22	1,78	1,89	1,67	1,89	1,78	1,71

Segundo a Tabela 4.2, os *sites* da FNAC e Americanas proporcionam interações não tão boas para o cadastro de usuários. Assim, para este tipo de tarefa, um parâmetro admissível para poderia ser de 2 enfrentamentos de erros na tarefa de cadastro de usuário.

**Tabela 4.3** – Valores de Referência para Busca Avançada nas Compras

	Saraiva	Tempo Real	Ponto Frio	Fnac	Shop Fácil	America nas	Alô Bebê	Média do usuário
Usuário 1	01	00	00	00	00	00	00	0,14
Usuário 2	01	00	00	00	00	00	00	0,14
Usuário 3	01	00	00	00	00	00	00	0,14
Usuário 4	01	00	00	00	00	00	00	0,14
Usuário 5	02	00	00	00	00	00	00	0,29
Usuário 6	01	00	00	00	00	00	00	0,14
Usuário 7	01	00	00	00	00	00	00	0,14
Usuário 8	01	00	00	00	00	00	00	0,14
Usuário 9	01	00	00	00	00	00	00	0,14
Média do site	1,11	0,0	0,0	0,0	0,0	0,0	0,0	0,16

Segundo a Tabela 4.3, o *site* da Livraria Saraiva proporciona interações não tão boas para a realização de compras pelos usuários. Assim, para este tipo de tarefa, um parâmetro admissível para poderia ser de 1 a 2 buscas de ajuda.

**Tabela 4.4** – Valores de Referência para os Enfrentamentos de Erros nas Compras

	Saraiva	Tempo Real	Ponto Frio	Fnac	Shop Fácil	America nas	Alô Bebê	Média do usuário
Usuário 1	01	03	03	01	02	01	00	1,57
Usuário 2	02	02	03	00	01	00	00	1,14
Usuário 3	02	01	02	01	04	01	00	1,57
Usuário 4	01	03	02	01	01	02	01	1,57
Usuário 5	03	03	04	01	03	01	01	2,29
Usuário 6	01	01	01	00	01	00	00	0,57
Usuário 7	02	02	02	01	02	01	01	1,57
Usuário 8	01	01	01	00	01	00	01	0,71
Usuário 9	02	01	02	00	02	01	00	1,14
Média do site	1,67	1,89	2,22	0,56	1,89	0,78	0,44	1,35

Segundo a Tabela 4.4, os *sites* das empresas Ponto Frio, Tempo Real e Shop Fácil proporcionam interações não tão boas para as compras dos usuários. Assim, para este tipo de tarefa, um parâmetro admissível poderia ser de 2 a 3 enfrentamentos de erros na tarefa.

Uma visão geral dos valores admissíveis determinados manualmente pode ser vista na Tabela 4.5 abaixo. Eles representam o embrião de uma base de valores admissíveis para usabilidade de *web sites* de *e-commerce*, e já podem ser usados na aplicação piloto do Ergo-Monitor, descrita no final deste capítulo.

**Tabela 4.5** – Definição Inicial de Parâmetros Admissíveis

<b>Valores Admissíveis para a Usabilidade em Web Sites de Comércio Eletrônico (b2c)</b>		
	<b>Busca de Ajuda</b>	<b>Enfrentamento de Erros</b>
<b>Cadastro</b>	1	2
<b>Compra</b>	1 a 2	2 a 3

Como valores admissíveis, eles deveriam ser considerados como limites máximos, além dos quais os valores indicativos da usabilidade de um *web site* não podem estar. De fato, em situação normal, os indicativos da usabilidade deveriam estar bem aquém dos valores admissíveis, talvez um valor correspondente à metade do admissível.

Na seção a seguir são apresentadas as características que compõem o conjunto de programas que define a versão atual do protótipo do Ergo-Monitor. O conjunto de programas implementado foi especialmente denominado por Log Control para facilitar sua descrição ao longo deste trabalho.

#### **4.2.2 – O Log Control**

Nesta Seção são apresentadas as características dos programas envolvidos no Log Control. Esses programas foram implementados visando realizar uma filtragem nos *log files*, buscando remover os trechos que não são significativos para o apoio à avaliação da usabilidade dos *web sites*. Eles são: *Filtro.java*, *Busca.java*, *Buscaseq.java* e *Buscar.java*.

Para desenvolver os programas de filtro e busca nos *log files*, foi empregada a linguagem Java por ser uma linguagem orientada a objetos e multiplataforma, além do fato de que esta linguagem é bastante apropriada para ser empregada no ambiente *World Wide Web*, uma vez que o código fonte é pré compilado e transformado em *byte codes*, que são interpretados pela JVM (Java Virtual Machine).

Em caráter ilustrativo, pode-se ressaltar que a implementação desses programas requisitou a seguinte quantidade de linhas de código:

- Filtro – 186
- Busca Simples - 328
- Busca Sequencial - 376
- Busca de Tarefas – 643

As seções a seguir apresentam as principais características de cada um desses programas.

#### **4.2.2.1 – *Filtro.java***

Este programa tem a finalidade de filtrar o *log file* visando criar um novo arquivo (*Arquivo de Destino*) que contenha apenas as páginas que foram acessadas quando das interações com o *web site*. Essas páginas podem ter extensões ASP, HTML ou PHP e as informações que não são importantes no que se refere às características de usabilidade do *web site* que podem ficar gravadas no *log file* são desprezadas. Como exemplo para a apresentação das informações desprezadas podemos ter os arquivos com extensão GIF cujos carregamentos ficam armazenados nos *log files*, mas que não apresentam relevância com relação às características de usabilidade do *web site*. A única ressalva que pode ser feita com relação aos arquivos .GIF refere-se ao fato de que o tempo de carregamento de alguns desses arquivos pode ser grande, o que determinaria um problema de acesso ao *web site*, mas de forma alguma tais arquivos têm algum relacionamento com características específicas relacionadas às interações do usuário com o *web site*.

Ainda, o programa Filtro.Java grava no Arquivo de Destino apenas os acessos feitos por um número de IP específico, sendo que os outros acessos provenientes de outros IPs são desprezados. Isso se deve ao fato de haver a restrição de um endereço IP corresponder a um usuário específico (e vice-versa) e que os acessos ao *web site* são gravados nos *log files* em ordem cronológica, ficando armazenadas informações de data e hora de acesso, independentemente do IP da máquina que fez o acesso. Se não forem eliminados os IPs que não seja o do monitoramento em questão, o monitoramento dos acessos pode ficar bastante

comprometido. Logo, uma visualização prévia do arquivo de *log* pode se fazer necessária para a determinação exata do endereço IP a ser monitorado.

A seguir é apresentado um trecho de *log file* que pode servir como entrada para o programa de busca.

```
127.0.0.1 - - [02/Sep/2003:11:46:33 -0300] "GET /ruthfraldas/ HTTP/1.1" 200 9297
127.0.0.1 - - [02/Sep/2003:11:46:33 -0300] "GET /ruthfraldas/imagens/busca_av.gif
HTTP/1.1" 200 1352
127.0.0.1 - - [02/Sep/2003:11:46:34 -0300] "GET
/ruthfraldas/imagens/cadastrese.gif HTTP/1.1" 200 1594
127.0.0.1 - - [02/Sep/2003:11:46:34 -0300] "GET /ruthfraldas/imagens/cesta.gif
HTTP/1.1" 200 1596
127.0.0.1 - - [02/Sep/2003:11:46:34 -0300] "GET
/ruthfraldas/imagens/barra_azul.gif HTTP/1.1" 200 134
127.0.0.1 - - [02/Sep/2003:11:46:35 -0300] "GET /ruthfraldas/imagens/dicas.gif
HTTP/1.1" 200 307
127.0.0.1 - - [02/Sep/2003:11:46:35 -0300] "GET /ruthfraldas/imagens/pesquisa.gif
HTTP/1.1" 200 354
127.0.0.1 - - [02/Sep/2003:11:46:36 -0300] "GET /ruthfraldas/imagens/forum.jpg
HTTP/1.1" 200 1105
127.0.0.1 - - [02/Sep/2003:11:47:00 -0300] "GET
/ruthfraldas/imagens/bot_ok_am.gif HTTP/1.1" 200 231
127.0.0.1 - - [02/Sep/2003:11:47:01 -0300] "GET /ruthfraldas/imagens/voltar.jpg
HTTP/1.1" 200 751
127.0.0.1 - - [02/Sep/2003:11:47:04 -0300] "GET /ruthfraldas/imagens/produtos.jpg
HTTP/1.1" 200 3063
127.0.0.1 - - [02/Sep/2003:11:47:04 -0300] "GET /ruthfraldas/index.php?cod_cat=34
HTTP/1.1" 200 9938
127.0.0.1 - - [02/Sep/2003:11:47:09 -0300] "GET
/ruthfraldas/cesta_compras.php?opt=I&cod_produto=13 HTTP/1.1" 200 10823
127.0.0.1 - - [02/Sep/2003:11:47:16 -0300] "POST /ruthfraldas/cesta_compras.php
HTTP/1.1" 200 10827
127.0.0.1 - - [02/Sep/2003:11:47:23 -0300] "POST /ruthfraldas/cesta_compras.php
HTTP/1.1" 200 10827
127.0.0.1 - - [02/Sep/2003:11:47:26 -0300] "GET
/ruthfraldas/ajuda.php?assunto=frete HTTP/1.1" 200 660
127.0.0.1 - - [02/Sep/2003:11:47:26 -0300] "GET /ruthfraldas/imagens/ajuda.jpg
HTTP/1.1" 200 1148
```

E o trecho filtrado deste arquivo de *log*, apenas com as linhas de páginas *.php*, é o seguinte:

```
127.0.0.1 - - [02/Sep/2003:11:47:04 -0300] "GET /ruthfraldas/index.php?cod_cat=34
HTTP/1.1" 200 9938
127.0.0.1 - - [02/Sep/2003:11:47:09 -0300] "GET
/ruthfraldas/cesta_compras.php?opt=I&cod_produto=13 HTTP/1.1" 200 10823
127.0.0.1 - - [02/Sep/2003:11:47:16 -0300] "POST /ruthfraldas/cesta_compras.php
HTTP/1.1" 200 10827
127.0.0.1 - - [02/Sep/2003:11:47:23 -0300] "POST /ruthfraldas/cesta_compras.php
HTTP/1.1" 200 10827
127.0.0.1 - - [02/Sep/2003:11:47:26 -0300] "GET
/ruthfraldas/ajuda.php?assunto=frete HTTP/1.1" 200 660
```

#### **4.2.2.2 – Busca.java**

Este programa tem como entrada o arquivo filtrado gerado pelo *Filtro.java* (que por *default* foi definido como *access.fil*) e tem como finalidade buscar por uma ou mais páginas específicas que possam estar presentes nesse arquivo já filtrado. A entrada desse programa é um arquivo texto que contenha todas as páginas que podem ser acessadas no *web site* (denominado por *paginas.dat*), sendo que a escolha dessas páginas é apresentada ao usuário através de caixas de seleção.

Toda vez que uma dessas páginas for encontrada ela é mostrada, e também é informado o tempo de acesso, bem como o somatório desses tempos.

A seguir são apresentados dois arquivos textos: o primeiro é a saída gerada para a busca de das páginas *index.php* e *cesta\_compras.php* que estão presentes no *log file* (já “filtrado”) apresentado no segundo trecho proveniente de um outro arquivo texto de entrada.

\*\*\*\*\*

Busca da pagina index.php

\*\*\*\*\*

Ocorrencia 1: 0 min. 05 seg.

Tempo total: 05 segundos.

\*\*\*\*\*

Busca da pagina cesta\_compras.php

\*\*\*\*\*

Ocorrencia 1: 0 min. 16 seg.

Ocorrencia 2: 0 min. 13 seg.

Tempo total: 29 segundos.

```
127.0.0.1 - - [02/Sep/2003:11:47:04 -0300] "GET /ruthfraldas/index.php?cod_cat=34 HTTP/1.1" 200 9938
127.0.0.1 - - [02/Sep/2003:11:47:09 -0300] "GET /ruthfraldas/cesta_compras.php?opt=I&cod_produto=13 HTTP/1.1"
200 10823
127.0.0.1 - - [02/Sep/2003:11:47:16 -0300] "POST /ruthfraldas/cesta_compras.php HTTP/1.1" 200 10827
127.0.0.1 - - [02/Sep/2003:11:47:23 -0300] "POST /ruthfraldas/cesta_compras.php HTTP/1.1" 200 10827
127.0.0.1 - - [02/Sep/2003:11:47:26 -0300] "GET /ruthfraldas/ajuda.php?assunto=frete HTTP/1.1" 200 660
127.0.0.1 - - [02/Sep/2003:11:47:34 -0300] "GET /ruthfraldas/cesta_compras.php HTTP/1.1" 200 10832
127.0.0.1 - - [02/Sep/2003:11:47:47 -0300] "GET /ruthfraldas/fale_conosco.php HTTP/1.1" 200 10832
```

#### 4.2.2.3 – *Buscaseq.java*

Este programa também tem como entrada o arquivo filtrado gerado pelo *Filtro.java* e tem como finalidade buscar por uma única seqüência de acesso a uma ou mais páginas específicas que possam estar presentes nesse arquivo já filtrado. A entrada desse programa é a própria seqüência de páginas que compõem o *web site* que devem estar gravadas no mesmo arquivo texto *paginas.dat..* Toda vez que uma seqüência é encontrada, ela é mostrada, informando também o tempo de acesso para cada uma dessas vezes, bem como o somatório desses tempos. Esta seqüência tem que ser exatamente igual à definida, sendo que a ordem de apresentação das páginas deve ser observada.

A seguir é apresentado um trecho do arquivo de saída gerado para a busca da seguinte seqüência exata de páginas: *index.php* + *busca\_avancada.php*. Ou seja, ocorreram duas repetições exatas de *index.php* seguido de *busca\_avancada.php*, a primeira durando 3 minutos e 29 segundos e a segunda 2 minutos e 40 segundos.

Busca da sequencia de paginas: 1: index.php 2: busca_avancada.php  Ocorrencia 1: 3 min. 29 seg. Ocorrencia 2: 2 min. 40 seg.  Tempo total: 6 minuto(s) e 09 segundo(s).
--

#### 4.2.2.4 – *Buscatar.java*

Finalmente, tal como nos dois programas anteriores, o *Buscatar.Java* também tem como entrada o arquivo filtrado gerado pelo programa *Filtro.java*, e tem como finalidade buscar por um ou mais comportamentos específicos. Esses comportamentos são relativos a uma tarefa e, portanto, também é entrada deste programa um arquivo texto que contenha todos os comportamentos que se deseja monitorar em uma tarefa específica.

Assim, devem ser criados diversos arquivos textos, um para cada tarefa. E, em cada arquivo texto, deve-se apresentar as seqüências de páginas que devem ser monitoradas para a determinação das taxas de usabilidade do *web site*. Logo, por exemplo, o comportamento de **Sucesso** em uma tarefa de **Cadastro**, pode ser determinado através da seguinte seqüência de interação:

- acesso à página de cadastro;
- acesso às  $n$  páginas de mensagens de erro ( $n \geq 0$ );
- acesso à página de conclusão do cadastro (com êxito).

Especificamente, para o *web site* empregado para validar o funcionamento do Ergo-Monitor (empresa Ruth Fraldas), foi assumido que esse valor de  $n$  pode variar de 0 a 4 (ou seja,  $0 \leq n \leq 4$ ). Uma parte do arquivo de entrada para o programa *Buscarar.java* empregado no monitoramento das possibilidades de sucesso que devem ser determinadas na tarefa de *cadastro* é apresentada a seguir:

<pre>#Sucesso 0 usr_man_cadastro.php usr_man_cadastro.php?opt=IG #Sucesso 1 usr_man_cadastro.php erro.php usr_man_cadastro.php?opt=IG #Sucesso 2 usr_man_cadastro.php erro.php erro.php usr_man_cadastro.php?opt=IG</pre>	<pre>#Sucesso 3 usr_man_cadastro.php erro.php erro.php erro.php usr_man_cadastro.php?opt=IG #Sucesso 4 usr_man_cadastro.php erro.php erro.php erro.php usr_man_cadastro.php?opt=IG</pre>
---	--

Neste exemplo, a página *usr\_man\_cadastro* é a página de início da tarefa e *usr\_man\_cadastro?opt=IG* é a página de confirmação do sucesso no cadastro. Ainda, pode-se observar que são apresentadas diversas possibilidades de acesso quando do sucesso de um *cadastro*. Essas possibilidades se devem ao fato de poderem ocorrer alguns erros no preenchimento do formulário que compõe o cadastro (tal como inserção errada do número do CPF ou de uma data, não preenchimento de algum campo, etc.). Nesta primeira versão do Ergo-Monitor foi necessária a limitação do número de mensagens de erro que podem ser apresentadas (este limite foi definido como sendo 4) e, por isso, existem 5 possibilidades de sucesso para o *cadastro* (de 0 a 4 erros no preenchimento dos formulários).

Alguns caracteres têm significado especial nestes arquivos texto de entrada do programa *Buscarar.java*. Eles são:

# - serve para definir uma das possibilidades de seqüência de páginas acessadas para a realização de alguma tarefa específica. Como ilustração, pode ser observado no exemplo anterior (o do *cadastro*) as linhas *# Sucesso0*, *# Sucesso1*, ..., *# Sucesso4*. Cada uma delas determina que a seqüência de páginas a seguir representa a possibilidade de ocorrência do

sucesso no cadastro, com nenhuma mensagem de erro no preenchimento do formulário (*Sucesso 0*), com apenas uma mensagem de erro no preenchimento do formulário (*Sucesso 1*), ou com até quatro mensagens de erro no preenchimento do formulário (*Sucesso 4*).

! – significa que, para a ocorrência do evento a ser monitorado, pode ser observado no arquivo de *log* qualquer página que não seja especificamente a apresentada após o símbolo “!”. Como exemplo, pode-se ter:

```
#CancErro1
usr_man_cadastro.php
erro.php
!usr_man_cadastro.php?opt=IG
```

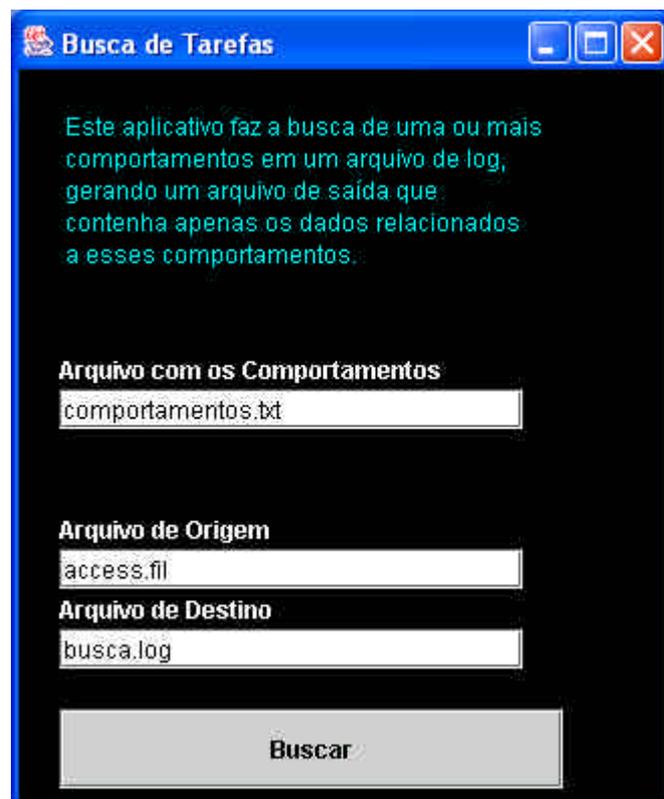
Aqui, a tarefa de cadastro é monitorado o cancelamento com um erro (com uma mensagem de erro), como sendo o acesso à página *usr\_man\_cadastro.php*, seguido do acesso à página *erro.php*, seguindo de qualquer outra página que não seja *usr\_man\_cadastro.php?opt=IG*, que no caso, é a página que apresenta a mensagem de sucesso no cadastro.

?? – significa que a taxa em questão não tem como ser calculada para o comportamento desejado. Essa impossibilidade pode ser devida a dois fatores: (1) ou não se tem como monitorar o acesso às páginas para tal tarefa (como é o caso das *repetições* na tarefa *fale conosco*: o usuário pode estar querendo deixar mais de uma mensagem, e como não se tem acesso à mensagem digitada, não se sabe se o usuário digitou a mesma mensagem), ou (2) a taxa nada representaria (como é o caso da *repetição* na atividade de *trocar senha*: o usuário pode ter desejado escolher uma nova senha logo após a primeira troca, por exemplo).

Esses símbolos “??” foram escolhidos por saber-se que praticamente nunca podem ser encontrados nas definições das páginas *php* que podem compor um *web site*. Entretanto, pode ficar a cargo do avaliador-usuário do Ergo-Monitor a definição de algum símbolo (ou sequência de símbolos) que não possam ser encontrados nas *urls* do *web site* a ser monitorado. Esta versão do Ergo-Monitor requer que todas as possibilidades de

comportamentos sejam apresentadas, pois o arquivo texto gerado pelo programa *Buscar.java* deve estar formatado de forma a ser facilmente importável pelo *Excel* para a determinação numérica das taxas de usabilidade. Portanto, mesmo que sejam apresentados valores nulos para um comportamento específico, ele deve ser apresentado no arquivo de saída na posição correta para uma possível exportação para o Excel (isto é, em suas linhas e colunas pré-definidas).

A Figura 4.4 apresenta a interface do programa *Buscar.java* que requisita a entrada dos arquivos de comportamentos e de origem, além da definição do nome do arquivo de saída que será gerado. Este arquivo irá conter os resultados do monitoramento da tarefa em questão.



**Figura 4.4** – Tela do Programa *Buscar.java*

Existe uma outra dificuldade que pode ser encontrada na elaboração dos arquivos texto que devem servir de entrada para esta versão do programa *Buscar.java*: é necessário que sejam definidas TODAS as possibilidades de acessos às páginas que devem ser

monitoradas quando da realização de uma tarefa. Esta atividade pode ser demorada, difícil de ser realizada e cansativa. Portanto, sujeita a erros na sua condução;

Para ilustrar tais seqüências, as Figuras 4.5 e 4.6 abaixo apresentam as possibilidades de acessos às páginas do *web site* empregado para a avaliação do Ergo-Monitor (Ruth Fraldas) para a realização de duas tarefas importantes: Cadastro e Busca Avançada.

Nestas figuras podem ser observadas as páginas a serem acessadas para dar apoio à determinação de taxas, como sucesso, cancelamentos, repetição de tarefas, entre outras. Logo, uma notificação será gravada no arquivo de saída toda vez que uma seqüência dessas for monitorada pelo Ergo-Monitor (Log Control). Com isso, o avaliador terá dados que permitam determinar as taxas que caracterizem os padrões de usabilidade do *web site*.

Repetição0 ?? #Repetição1 usr_man_cadastro.php erro.php usr_man_cadastro.php #Repetição2 usr_man_cadastro.php erro.php erro.php usr_man_cadastro.php #Repetição3 usr_man_cadastro.php erro.php erro.php erro.php usr_man_cadastro.php #Repetição4 usr_man_cadastro.php erro.php erro.php erro.php erro.php usr_man_cadastro.php	#Sucesso0 (continuação de #Repetição4???) usr_man_cadastro.php usr_man_cadastro.php?opt=IG #Sucesso1 usr_man_cadastro.php erro.php usr_man_cadastro.php?opt=IG #Sucesso2 usr_man_cadastro.php erro.php erro.php usr_man_cadastro.php?opt=IG #Sucesso3 usr_man_cadastro.php erro.php erro.php erro.php usr_man_cadastro.php?opt=IG #Sucesso4 usr_man_cadastro.php erro.php erro.php erro.php erro.php usr_man_cadastro.php?opt=IG
--	---

**Figura 4.5** – Seqüência de Páginas para a Determinação de Taxas Relativas ao Cadastro

#Abandono0 ??	#Repetição2 busca_avancada.php
#Abandono1 busca_avancada.php busca_avancada.php?acao=B !cesta_compras.php	busca_avancada.php?acao=N busca_avancada.php #Repetição3 ??
#Abandono 2 busca_avancada.php busca_avancada.php?acao=B !ver_produto.php	#Repetição4 ??#Sucesso0 ??
#Abandono3 ??	#Sucesso1 busca_avancada.php busca_avancada.php?acao=B
#Abandono4 ??	cesta_compras.php #Sucesso2
#Repetição0 ??	busca_avancada.php busca_avancada.php?acao=B
#Repetição1 busca_avancada.php erro.php busca_avancada.php	ver_produto.php cesta_compras.php #Sucesso3 ?? #Sucesso4 ??

**Figura 4.6** – Sequência de Páginas para a Determinação de Taxas Relativas à Busca Avançada

A Figura 4.7 apresenta um exemplo de saída esperada, e a determinação de taxas específicas, tais como Taxa de Sucessos, Taxa de Cancelamentos, Taxa de Desistências, Taxa de Repetições, etc, devem ser feitas a partir desses dados. Neste exemplo, esses dados são relacionados ao monitoramento da tarefa de *Busca Avançada* no *web site* da Ruth Fraldas.

A planilha apresentada na Figura 4.8 já está formatada de forma adequada a apresentar as taxas determinadas pelo arquivo texto apresentado na Figura 4.7.

TAREFA: buscaavanc			
IPO	TEMPO	QUANTIDADE	
Acesso	178	15	
Help	37	2	
Mensagem de Erro	0	0	
Abandono0	0	0	
Abandono1	172	5	
Abandono2	0	0	
Abandono3	0	0	
Abandono4	0	0	
Soma_Abandono			=soma(B7:B11) =soma(C7:C11)
CancErro0	0	0	
CancErro1	0	0	
CancErro2	0	0	
CancErro3	0	0	
CancErro4	0	0	
Soma_CancErro			=soma(B13:B17) =soma(C13:C17)
Desistência0	56	2	
Desistência1	0	0	
Desistência2	0	0	
Desistência3	0	0	
Desistência4	0	0	
Soma_Desistência			=soma(B19:B23) =soma(C19:C23)
InSucesso0	0	0	
InSucesso1	0	0	
InSucesso2	0	0	
InSucesso3	0	0	
InSucesso4	0	0	
Soma_InSucesso			=soma(B25:B29) =soma(C25:C29)
Repetição0	0	0	
Repetição1	0	0	
Repetição2	0	0	
Repetição3	0	0	
Repetição4	0	0	
Soma_Repetição			=soma(B31:B35) =soma(C31:C35)
Sucesso0	0		
Sucesso1	59	2	
Sucesso2	0		
Sucesso3	0		
Sucesso4	0		
Soma_Sucesso			=soma(B37:B41) =soma(C37:C41)
TAXAS	Help		=(B5/B4)=(C5/C4)
	Mensagem de Erro		=(B6/B4)=(C6/C4)
	Abandono		=(E12/B4) =(F12/C4)
	CancErro		=(E18/B4) =(F18/C4)
	Desistência		=(E24/B4) =(F24/C4)
	InSucesso		=(E30/B4) =(F30/C4)
	Repetição		=(E36/B4) =(F36/C4)
	Sucesso		=(E42/B4) =(F42/C4)

**Figura 4.7** – Resultado Obtido pelo Programa Buscar.java para a Tarefa Busca Avançada

TAREFA: buscaavanc			
TIPO	TEMPO	QUANTIDADE	
Acesso	178	15	
Help	37	2	
Mensagem de Erro	0	0	
Abandono0	0	0	
Abandono1	172	5	
Abandono2	0	0	
Abandono3	0	0	
Abandono4	0	0	
Soma_Abandono			172 5
CancErro0	0	0	
CancErro1	0	0	
CancErro2	0	0	
CancErro3	0	0	
CancErro4	0	0	
Soma_CancErro			0 0
Desistência0	56	2	
Desistência1	0	0	
Desistência2	0	0	
Desistência3	0	0	
Desistência4	0	0	
Soma_Desistência			56 2
InSucesso0	0	0	
InSucesso1	0	0	
InSucesso2	0	0	
InSucesso3	0	0	
InSucesso4	0	0	
Soma_InSucesso			0 0
Repetição0	0	0	
Repetição1	0	0	
Repetição3	0	0	
Repetição4	0	0	
Soma_Repetição			0 0
Sucesso0	0	0	
Sucesso1	59	2	
Sucesso2	0	0	
Sucesso3	0	0	
Sucesso4	0	0	
Soma_Sucesso			59 2
TAXAS		Help	0,207865 0,133333
		Mensagem de Erro	0 0
		Abandono	0,966292 0,333333
		CancErro	0 0
		Desistência	0,314607 0,133333
		InSucesso	0 0
		Repetição	0 0
		Sucesso	0,331461 0,133333

**Figura 4.8** – Planilha Excel que Apóia a Determinação de Taxas para a Busca Avançada

Uma comparação dessas taxas determinadas com os parâmetros admissíveis de boas (ou más) interações com *web sites* similares pode apoiar uma análise das características de usabilidade do *web site* em avaliação.

O programa *Buscatar.java* foi executado para outras tarefas que podem ser monitoradas com o apoio do Ergo-Monitor e os resultados obtidos estão listados no Apêndice E. Essas demais tarefas monitoradas foram: busca simples, fale conosco, compras (dividida nas sub-tarefas: ir para o caixa, *login* do usuário, forma de pagamento via depósito ou forma de pagamento via cartão, entre outras.

É importante ressaltar que os resultados obtidos foram comparados com as gravações dos ensaios de interação e, tal como acontece com os dados apresentados nas Tabelas 4.6 e 4.7, os valores determinados via monitoramento automatizado e via ensaio de interação registrado são praticamente os mesmos. Isso só vem a corroborar a adequação do emprego do Ergo-Monitor para apoiar a avaliação da usabilidade de *web sites* de comércio eletrônico.

Com a finalidade de melhor descrever os programas que compõem o Log Control, o Apêndice A irá apresentar as listagens de dois desses programas, enquanto que o Apêndice C apresenta os arquivos textos empregados como parâmetros de entrada visando a descrição das seqüências de páginas.

### **4.3 – Validação do Ergo-Monitor**

A definição conceitual da primeira versão do Ergo-Monitor foi validada por meio de uma aplicação piloto, para o monitoramento da usabilidade do *web site* da empresa Ruth Fraldas da cidade de Rio Claro, SP. Trata-se de uma empresa de artigos infantis que iniciava a implementação de seu *web site* na ocasião em que a validação do Ergo-Monitor estava sendo planejada. Assim foram mantidos contatos visando principalmente dotar o site de tarefas fechadas e objetivas com os devidos “marcadores” (mensagens de sucesso, de erro, de ajuda, entre outras). Além desta atividade, muito da implementação deste ambiente foi feita pelo autor desta tese, e muitos dos problemas de usabilidade existentes no *site* foram mantidos de propósito.

As seções a seguir apresentam maiores detalhes sobre a implementação do *web site* da empresa Ruth Fraldas e sobre a forma como foram conduzidos alguns ensaios de interação que visaram avaliar a usabilidade deste *web site* empregando-se também o Ergo-Monitor.

#### **4.3.1 – O Web Site de Comércio Eletrônico da Ruth Fraldas**

Trata-se de um *web site* relativamente simples, mas que contém as principais tarefas que podem ser monitoradas em um *web site* de comércio eletrônico, tais como: cadastros, buscas (simples e avançadas), compras e troca de senhas.

A implementação deste *web site* requisitou o emprego de técnicas e ferramentas recentes e específicas para o desenvolvimento de aplicativos voltados para a *web*, que fossem aliadas ao conceito de orientação a objetos. Este emprego proporcionou diversas facilidades para a implementação deste *web site*, tais como os acessos ao Banco de Dados SQL, o relacionamento entre os códigos implementados em Javascript e PHP, o uso de uma máquina local como servidor Apache e o possível acesso ao *web site* através do *localhost*, entre outras. Entre essas técnicas podemos citar as abaixo descritas, destacando de forma bastante sucinta o porquê de suas utilizações:

- A Linguagem PHP: que é uma linguagem de programação de páginas web que traz um dinamismo maior ao *web site*. O servidor interpreta o código PHP e gera um arquivo final ao usuário, ocultando os detalhes de conexão com banco de dados e gerenciamento de variáveis. Os dados são passados de uma página para outra através de parâmetros fornecidos por formulários através da ação GET/POST. Daí em diante o interpretador analisa estes parâmetros e passa ao *browser* somente as informações necessárias.
- Servidor Apache: que é um servidor de páginas *web* robusto e estável. Seu emprego foi escolhido devido à sua popularidade e fácil administração. Além disso, trata-se de um servidor que facilmente se comunica com o interpretador de páginas PHP, o que facilitou bastante o trabalho de implementação deste *web site*. O Servidor Apache foi uma das principais ferramentas empregadas no desenvolvimento deste *web site*, uma vez que foi permitido o acesso ao *web site* através de um *link* a

<http://localhost/ruthfraldas>. Com isso o web site pode ser testado e validado, bem como os programas que compõem o Log Control e até mesmo a realização dos ensaios de interação que foram filmados;

- MySQL: é um Banco de dados de fácil manipulação. Possui ferramentas próprias para gerenciamento dos dados, possibilitando consultas SQL sobre a base de dados. Outro fato importante que gerou sua escolha, é que ele é facilmente acessado através de códigos PHP.
- JavaScript: Diferentemente da linguagem PHP, o código da linguagem JavaScript se mantém no código fonte do web site e é processado do lado do cliente, e não do lado do servidor. Logo, é uma linguagem muito útil para criação de funções que serão processadas pelo próprio browser, como, por exemplo, uma abertura de determinada tela ou o uso de “loops” e comandos condicionais tradicionais em linguagens de programação.

Este *site* foi construído, incluindo-se algumas particularidades que se fizeram necessárias ao funcionamento do ambiente Ergo-Monitor. A principal delas foi o fato de que as mensagens de erro ou ajuda foram implementadas de forma a serem armazenadas nos *log files*. Para tanto, sempre que houver a necessidade de se apresentar uma mensagem ao usuário, o processamento do *web site* direciona o acesso para a página *erro.php* que, dependendo do parâmetro passado, apresenta a mensagem apropriada.

As Figuras 4.9 a 4.14 a seguir apresentam algumas telas (páginas *web*) que fazem parte das tarefas de Cadastro, Busca Avançada e de Compras do *web site* da Ruth Fraldas.

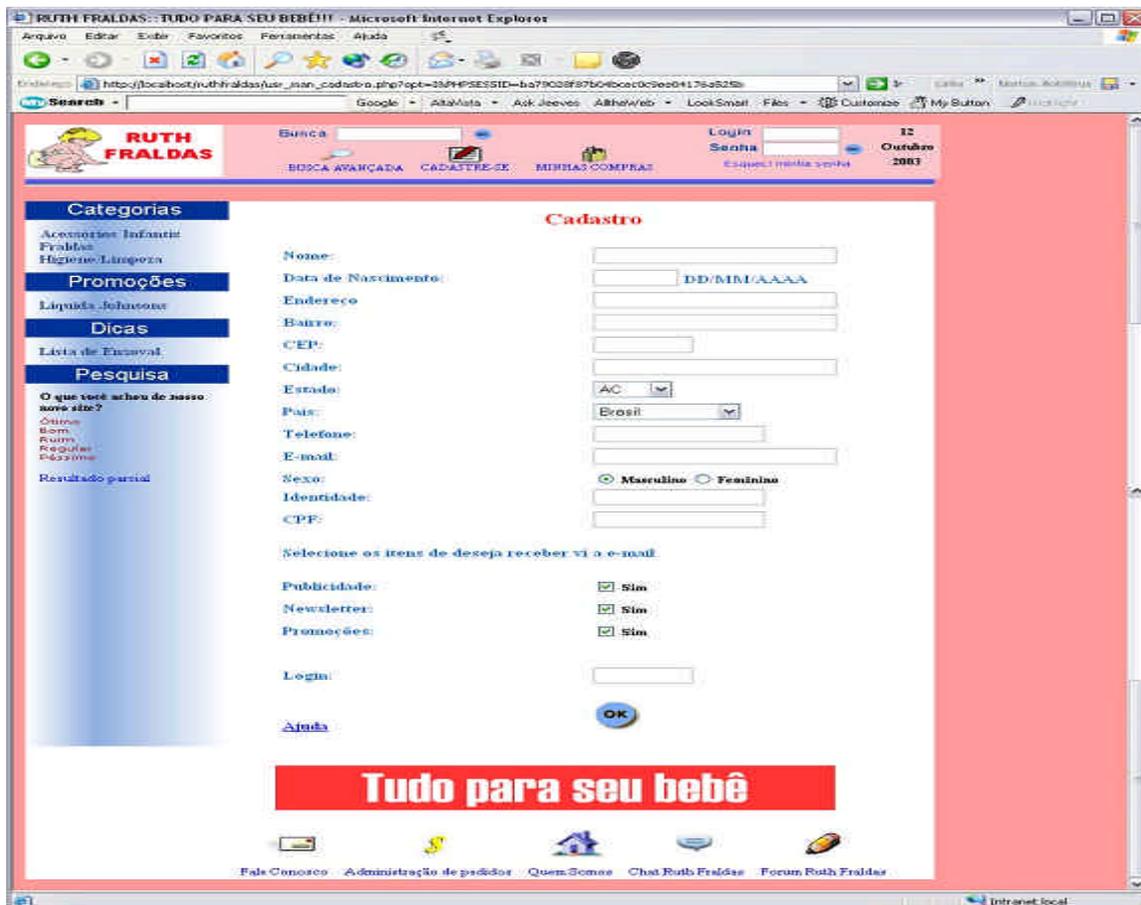


Figura 4.9 – Tela Inicial do Cadastro de Usuários



Figura 4.10 – Tela de Mensagem de Erro no Cadastro

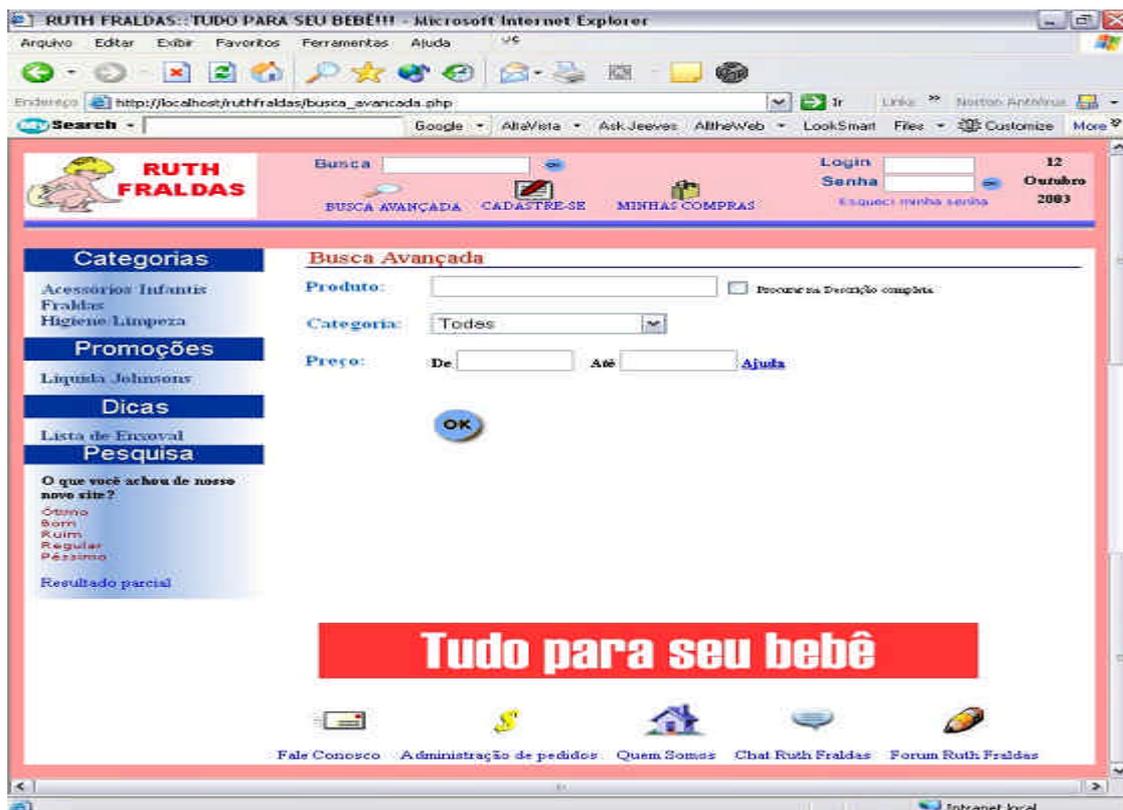


Figura 4.11 – Tela Inicial da Busca Avançada

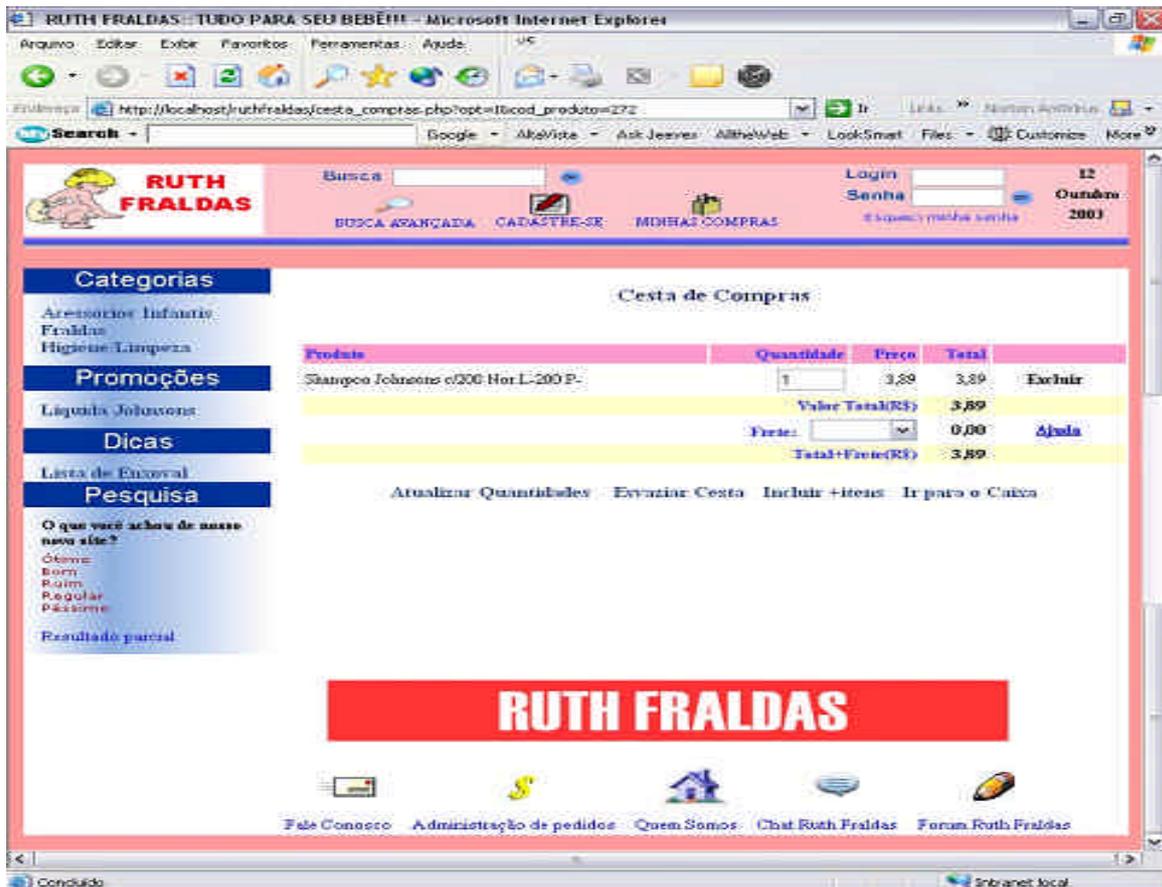


Figura 4.12 – Tela Inicial das Compras

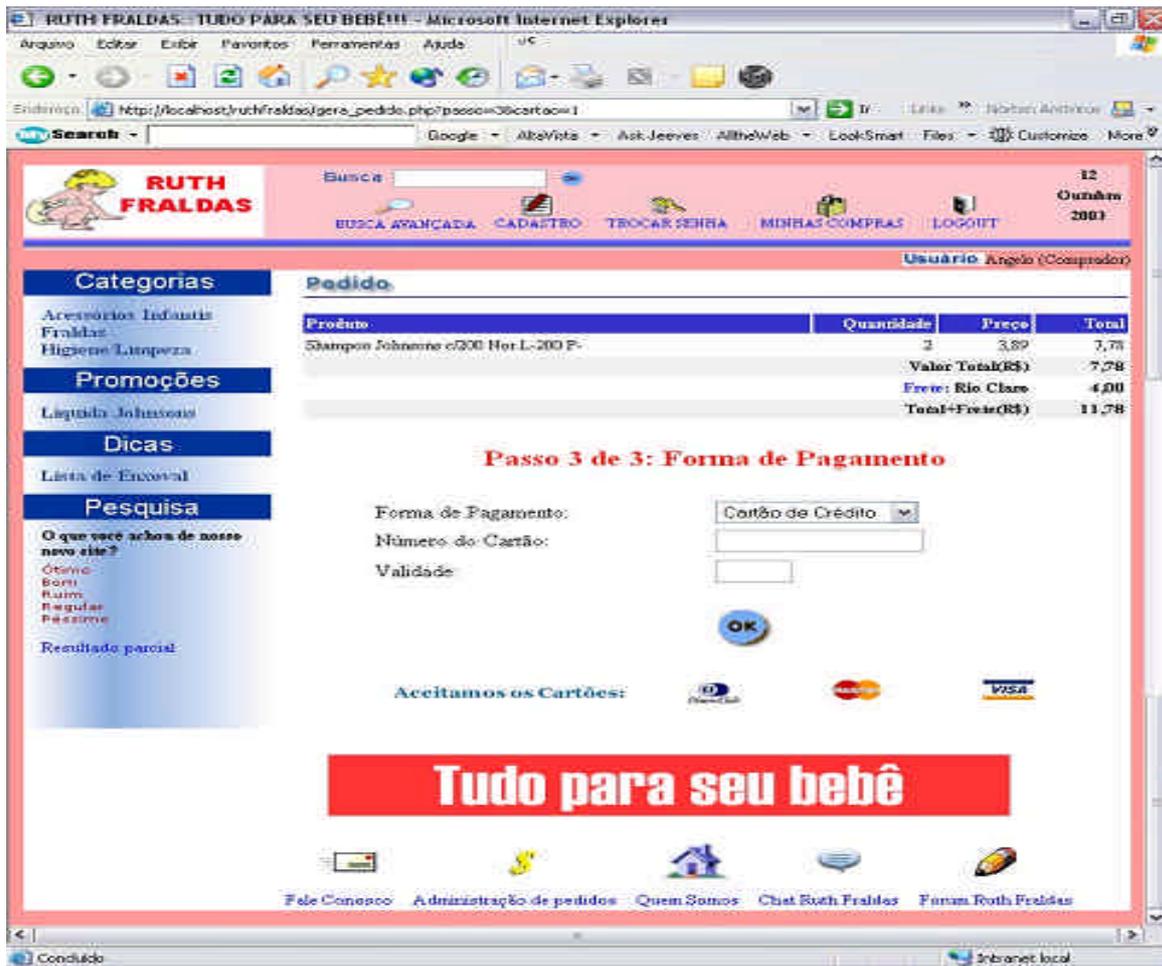
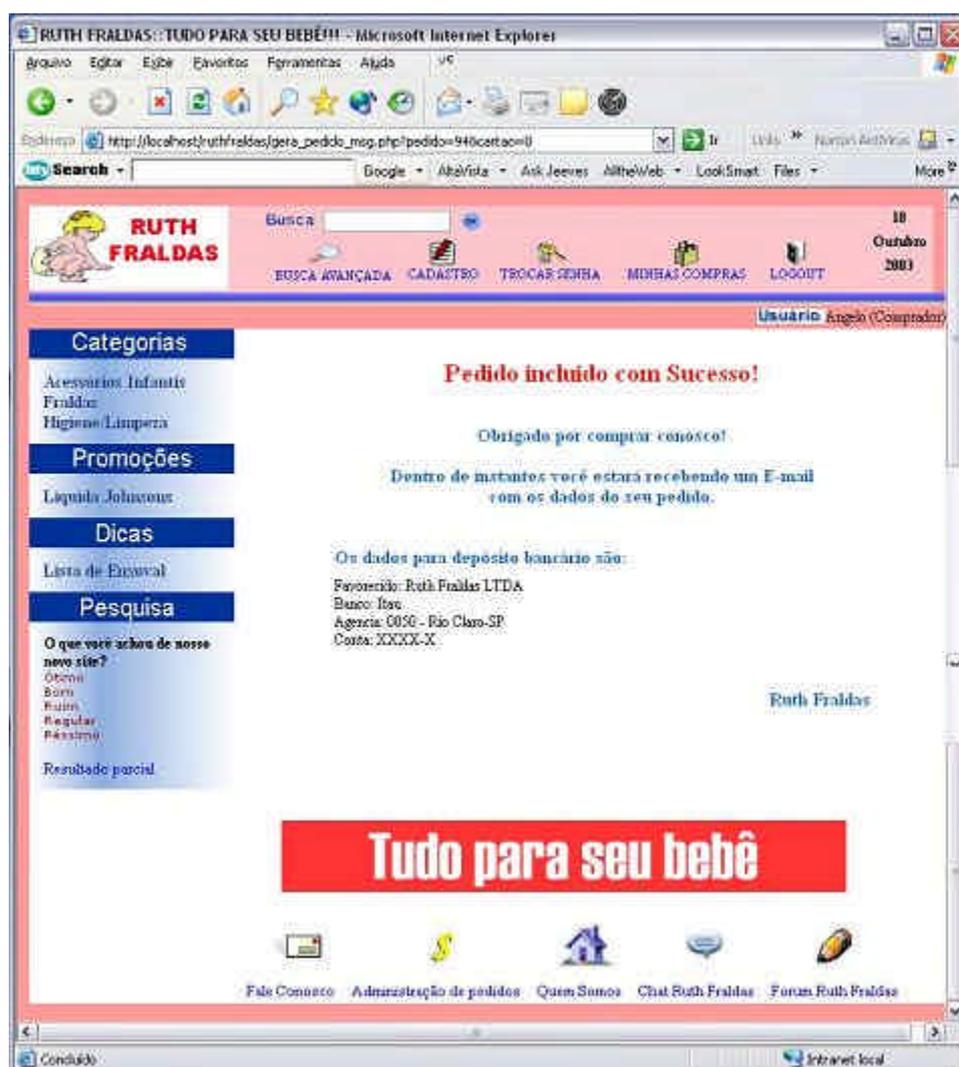


Figura 4.13 – Tela de Requisição da Forma de Pagamento



**Figura 4.14** – Tela de Confirmação das Compras

A Seção a seguir apresenta as particularidades que as tarefas de avaliação da usabilidade do *web site* da Ruth Fraldas apresentou.

### **4.3.2 – Avaliação da Usabilidade do Web Site da Ruth Fraldas**

Com a finalidade de validar os resultados obtidos em termos de taxas e métricas de usabilidade com o emprego dos programas desenvolvidos no ambiente Ergo-Monitor, foram realizados ensaios de interação com três usuários do *web site* da Ruth Fraldas e obtidos os dados acerca dessas interações especificamente para a tarefa de *cadastro*. Estes usuários possuem características bastante uniformes quanto à utilização de *web sites* de

comércio eletrônico (todos são experientes), mas fundamentalmente diferentes no contexto de utilização de um *web site* de comércio eletrônico específico para produtos infantis.

Um desses usuários (usuário A) é pai de dois filhos pequenos, conhece bastante bem produtos infantis, mas não faz compras de tais produtos regularmente (ele foi considerado, portanto, como um usuário intermitente); um outro usuário é estudante de graduação, não tem filhos e mora sozinho, logo, nunca havia feito compras de produtos infantis, nem conhece as reais necessidades de crianças e recém-nascidos (este foi considerado inexperiente). Finalmente, a terceira usuária é mãe de uma criança pequena e, portanto, conhece todas as necessidades de produtos infantis para esta criança e realiza tais compras freqüentemente (portanto, considerada experiente).

As interações foram gravadas em fita de vídeo, e os arquivos de *log* de cada sessão foram recuperados do servidor *web*. A comparação dos resultados se limita a comportamentos de Enfrentamento de Erros e de Busca de Ajuda. A Tabela 4.6 apresenta os resultados obtidos pelos programas que compõem o Ergo-Monitor atuando sobre os arquivos de *log* recuperados.

**Tabela 4.6** – Resultados Obtidos pelo Ergo-Monitor – Tarefa de Cadastro

<b>Cadastro</b>					
	<b>Tempo para a tarefa</b>	<b>Quantidade de Enfrentamentos de Erros</b>	<b>Quantidade de Busca de Ajuda</b>	<b>Taxa Individual de Enfrentamentos de Erros</b>	<b>Taxa Individual de Busca de Ajuda</b>
<b>Usuário 1</b>	330 segundos em 6 acessos	1	0	0,17	0
<b>Usuário 2</b>	245 segundos em 2 acessos	3	1	1,5	0,5
<b>Usuário 3</b>	176 segundos em 1 acesso	1	1	1	1
<b>Taxas Coletivas</b>				0,89	0,5

Já a Tabela 4.7 apresenta os dados observados dessas mesmas interações através de análises das gravações em vídeo. Observa-se que não existe uma diferença significativa entre tais dados, o que leva a crer que o Ergo-Monitor é capaz de monitorar tarefas de forma bastante adequada e apresentar eficientemente os dados obtidos por esses monitoramentos.

**Tabela 4.7** – Resultados Obtidos da Análise das Gravações -- Tarefa de Cadastro

<b>Cadastro</b>					
	<b>Tempo para a tarefa</b>	<b>Quantidade de Enfrentamentos de Erros</b>	<b>Quantidade de Busca de Ajuda</b>	<b>Taxa Individual de Enfrentamentos de Erros</b>	<b>Taxa Individual de Busca de Ajuda</b>
<b>Usuário 1</b>	332 segundos em 6 acessos	1	0	0,17	0
<b>Usuário 2</b>	245 segundos em 2 acessos	3	1	1,5	0,5
<b>Usuário 3</b>	181 segundos em 1 acesso	1	1	1	1
<b>Taxas Coletivas</b>				0,89	0,5

Uma comparação entre as Tabelas 4.6 e 4.7 mostra que não foram observadas discrepâncias significativas, uma vez que os tempos de acesso observados nas fitas foram comparados com os tempos obtidos pelo Ergo-Monitor e constatou-se que os resultados obtidos foram **praticamente os mesmos** (salvo algumas pequenas diferenças que podem muito bem ser provenientes de alguns erros de cronometragem ou arredondamentos dos valores determinados). Além disso, as quantidades de acesso a páginas de mensagens de erro e *help* foram **iguais**.

Sabe-se que com tais observações não se pode chegar a conclusão de que os programas do Ergo-Monitor estão 100% corretos, mas pode-se inferir que podem existir reduzidas chances de aparecerem erros. Entretanto, uma das atividades futuras deste trabalho que já está planejada é a realização de testes estruturais e funcionais [Pressman, 1994] para cada um desses programas.

Por outro lado, comparando os valores coletivos verificados para a usabilidade do site Ruth Fraldas (0,5 buscas da ajuda e 0,89 enfrentamentos de erros) com os valores admissíveis para a tarefa de cadastro (1 busca da ajuda e 2 enfrentamentos de erros – ver

Tabela 4.8), poderia-se chegar ao laudo final de que a usabilidade do *site* desta empresa está dentro do admissível.

Entretanto, tal afirmação pode não representar efetivamente a realidade, uma vez que a quantidade de ensaios de interação realizados foi relativamente baixa, tanto para a obtenção das taxas e métricas empregando-se o Ergo-Monitor, quanto para a determinação dos parâmetros admissíveis. Para que os pareceres sobre as características de usabilidade sejam mais fidedignos são necessárias a obtenção e análises de mais dados, tanto em uma direção quanto em outra. Assim, as peculiaridades dessas pequenas amostras de interações (e usuários) podem ser diluídas em um conjunto de dados maior. No futuro, empregando valores obtidos a partir de amostras significativas, o resultado de uma comparação, como a esboçada acima, deve ser mais fidedigno.

Na próxima seção são apresentadas algumas considerações a respeito do Ergo-Monitor, destacando as limitações que estão sendo impostas a esta primeira versão.

#### **4.4 – Considerações sobre o Ergo-Monitor**

Através da demarcação clara e objetiva de tarefas e comportamentos e de uma sucessão de processamentos adequados é possível obter-se de maneira rápida e barata indicadores de usabilidade que podem ser custosos de serem obtidos através de ensaios de interação. A atividade descrita acima mostra que os dados obtidos pelo emprego dessas duas abordagens apresentam semelhança quase que completa, mas com custos completamente diferentes. Isso pode ser facilmente observado, uma vez que os custos envolvidos no emprego da versão atual do Ergo-Monitor são basicamente resumidos ao tempo gasto pelo avaliador para a formatação dos arquivos que venham a descrever os comportamentos das tarefas a serem monitoradas, enquanto o emprego de metodologias “tradicionais” de avaliação têm sido apontadas como significativamente caras [Schneidermann, 2000].

Por um outro lado, as principais restrições que a aplicação do Ergo-Monitor pode apresentar, bem como as atividades que devem ser implementadas para que as próximas versões do Ergo-Monitor não sejam mais dependentes delas, são descritas abaixo:

1. Necessidade de se ter um endereço IP exclusivo e único para cada usuário. A forma de se corrigir tal problema é a definição do usuário que interagiu com o *web site* de alguma outra forma, como por exemplo a implantação de um cadastro de usuários ou então de um *login* inicial próprio para cada usuário do *web site*;
2. Necessidade de todas as páginas (incluindo as de mensagens de erro e *help*) serem registradas no *log file*. Para se minimizar tal problema deve-se procurar viabilizar as análises específicas do lado do cliente na conexão de modo a apresentar padrões de interação através de possíveis monitoramentos a teclados, mouses, processadores, etc;
3. Dificuldade para modelar tarefas e comportamentos a monitorar, devido à necessidade de prever todas as páginas intermediárias, (como as de mensagens de erro e de ajuda) que podem aparecer na execução de uma tarefa com sucesso. Minimizar esta restrição em versões futuras basicamente requer a implementação de um novo filtro capaz de desconsiderar páginas intermediárias quando necessário ou de interesse dos avaliadores;
4. Até o presente momento, apenas foram determinados alguns poucos dados que auxiliam na definição dos parâmetros admissíveis. Isso se deve ao fato de haver sido realizados poucos ensaios de interação e os que foram realizados estavam focados exclusivamente em tarefas passíveis de monitoramento automatizado pelo Ergo-Monitor. Com a realização de novas avaliações, haverá uma atualização no Banco de Dados de parâmetros admissíveis o que proporcionará maior precisão na análise das taxas e melhor caracterização dos laudos a serem apresentados;
6. Atualmente, os cálculos das taxas coletivas são feitos “manualmente” pelo avaliador, ainda sem o apoio de nenhuma ferramenta. Espera-se que a construção de ferramentas para dar suporte ao cálculo das taxas e às análises com relação aos parâmetros admissíveis seja capaz de auxiliar bastante esta fase do Ergo-Monitor.

O próximo Capítulo apresenta as conclusões que podem ser obtidas através do emprego do Ergo-Monitor como um ambiente de apoio à avaliação da usabilidade de *web sites* de comércio eletrônico, bem como as atividades futuras que estão sendo planejadas para dar continuidade a estas pesquisas e aprimorar a aplicação do Ergo-Monitor. Além disso, também no próximo Capítulo encontram-se as referências bibliográficas empregadas tanto na concepção e implementação do Ergo-Monitor, mas também na escrita desta tese.

## 5 – Conclusões e Trabalhos Futuros

Nesta Seção são apresentadas algumas conclusões que podem ser tiradas a partir das atividades envolvidas na elaboração e aplicação do ambiente Ergo-Monitor. Além disso, são apresentadas algumas atividades que podem ser realizadas para proporcionar melhores resultados na aplicação deste ambiente de apoio à avaliação da usabilidade de *web sites* comerciais.

### 5.1 – Conclusões

O emprego do Ergo-Monitor pode permitir que resultados importantes sejam obtidos independentemente da realização de testes de usabilidade, o que pode significar uma importante redução nos custos e tempos das avaliações. Por outro lado, também é esperada a obtenção de dados isentos da subjetividade natural de intervenções humanas.

O ambiente Ergo-Monitor induz a abstração das características do usuário e de seu contexto de operação, avaliando unicamente o que foi produzido como resultados da interação. Logo, trata-se de análise basicamente quantitativa, mas que para poder *reforçar* ou *derrubar* diagnósticos qualitativos previamente analisados, precisa envolver uma grande quantidade de dados de interação. Parâmetros que identifiquem os níveis de chamadas de ajuda, tempos de permanência na ajuda, cancelamentos, incidentes, sucessos, insucessos, repetições de tarefas, entre outros estão em constante atualização na medida em que os usuários interagindo com sistemas monitorados cresçam em número e variedade. Assim, quanto maior for a “massa” de usuários que interaja com o *web site* monitorado por este ambiente, mais dados podem ser interpretados e, portanto, mais fidedignos podem ser os resultados apresentados. O mesmo é válido para o cálculo de valores admissíveis para uma boa usabilidade. Em particular, os resultados obtidos em termos de valores admissíveis, embora limitados em abrangência, representam um importante diferencial deste trabalho, na medida em que não se tem notícia de levantamentos deste tipo na bibliografia. As análises baseadas nos dados coletados pelo Ergo-Monitor também podem apontar para uma possível falta de compatibilidade entre o usuário “previsto”, tal como esperado pelo *designer* do *web site*, e os “reais”, que são os que efetivamente interagem com o *web site*.

Por outro lado, os laudos produzidos pelo ambiente Ergo-Monitor podem proporcionar resultados mais expressivos quando analisados em conjunto com os resultados obtidos através do emprego de análises qualitativas. Elabora-se que a análise quantitativa de dados de *logs* venha a ser uma técnica mais eficiente quando estiver sendo empregada com base nos resultados obtidos a partir de uma técnica diagnóstica que já tenha sido realizada, e cujos resultados orientem a modelagem de tarefas e comportamentos a monitorar.

Espera-se que, com a utilização conjunta do Ergo-Monitor com técnicas diagnósticas qualitativas (tal como pretende suportar o ambiente Ergo-CoIn), possam ser produzidos outros resultados importantes relativos à usabilidade de *web sites*, tais como:

- indicar valores mais reais para as taxas de efetividade em diversas tarefas (ou interações) tais como acessos a links, mensagens de erro, entre outras;
- apoiar a implantação de atitudes gerenciais que possam proporcionar revisões, treinamentos para usuários, melhorias corretivas, perfectivas ou de manutenções gerais; e
- possibilitar novas versões (projetos) para o *web site* apontando questões pontuais que possam ser apresentadas como problemáticas, tais como formulários, campos, páginas, figuras, ícones, etc.

A produção de laudos quantitativos através da aplicação do Ergo-Monitor também pode ser bastante interessante quando o monitoramento das interações com o *web site* for realizado em um intervalo de tempo bastante grande. Com o aumento do período de monitoramento, espera-se constatar a evolução das taxas e métricas calculadas para a usabilidade do *web site* a medida em que ele evolua em estilos e tecnologias. A comparação entre taxas e métricas obtidas para versões diferentes do mesmo *web site* pode ser feita rapidamente e seus resultados podem ser utilizados para a verificação da usabilidade das novas características inseridas. Fornecem assim, subsídios para que os projetistas possam analisar o impacto das alterações implementadas.

O ambiente Ergo-Monitor se apresenta como uma abordagem para apoio às avaliações de usabilidade, mas NÃO tem a pretensão de excluir totalmente a necessidade de realização de avaliações de usabilidade de *web sites* com o emprego de avaliadores e da aplicação de técnicas e metodologias já conceituadas pela academia e amplamente utilizadas tanto em contextos acadêmicos quanto industriais. A idéia básica deste ambiente é servir como ferramenta de apoio ou de apresentação de um conjunto limitado de taxas e métricas de usabilidade dos *web sites*. As avaliações “tradicionais” devem continuar sendo empregadas sempre que os projetistas, usuários ou avaliadores encontrarem necessidades de indicativos mais completos sobre a usabilidade de *web sites*.

## **5.2 – Atividades Futuras**

O ambiente Ergo-Monitor ainda precisa de alguma implementação para poder ser melhor analisado e empregado em larga escala. Algumas delas são descritas nas seções abaixo.

### **5.2.1 – Manutenção da base de valores admissíveis para uma boa usabilidade**

A determinação em larga escala, de taxas e métricas que caracterizam as boas (e as más) interações com os *web sites* de comércio eletrônico é uma atividade necessária para o funcionamento do ambiente Ergo-Monitor. Como continuidade deste trabalho pretende-se empregar este ambiente para coletar e determinar valores reais para taxas e métricas de usabilidade para todos os tipos de usuários e contextos de operação. Assim, será possível não apenas aprimorar os valores admissíveis para as taxas e métricas de usabilidade para estilos e componentes de interface já existentes, mas também determinar valores referentes às evoluções de estilo e de tecnologia que venham a ser inseridas no contexto da *web*.

### **5.2.2 – Definição e Implementação do Monitoramento no Lado do Cliente**

Com a finalidade de se aprimorar o ambiente Ergo-Monitor, pode-se procurar também um foco interessante no lado do cliente que não está sendo explorado até o presente momento: os *cookies*.

Com isso, a proposição da elaboração do monitoramento das interações com o *web site*, instalado no lado do cliente em uma conexão, pode ser uma abordagem bastante

interessante. Para novas versões do Ergo-Monitor, espera-se ter um sistema parecido com o Log Control, mas que baseie suas coletas de dados no lado do cliente, buscando informações como:

1. eventos disparados pelo mouse ou teclado; e
2. informações apresentadas apenas pelo *browser* e que não ficam armazenadas nos *log files*, tal como ocorre na implementação de algumas mensagens de erro ou ajuda. Isso iria fazer com que pelo menos uma das restrições que atualmente são impostas ao Ergo-Monitor seja eliminada. Esta restrição refere-se ao fato de que a implementação do *web site* deve ser feita de forma que todas as mensagens de erro venham a ser apresentadas como um evento disparado pelo servidor, e, portanto, passível de ser registrado nos *log files*.

Ainda, a apresentação de resultados que possam ser analisados em conjunto com os resultados que foram apresentados para os *log files* pode ser mais precisa e, portanto, aumentar a qualidade dos resultados a serem apresentados após as avaliações realizadas.

Espera-se que essa atividade possa aprimorar o Ergo-Monitor e apontar maior praticidade no auxílio à tarefa de se avaliar a usabilidade de *web sites*.

### **5.2.3 – Caracterização dos Usuários Independentemente de Seu Endereço IP**

A proposição atual do ambiente Ergo-Monitor está sendo feita baseando-se no fato de que um usuário (cliente) do *web site* tem um endereço IP único e específico. Logo, as informações a respeito desse endereço IP são as que devem ser analisadas. Inclusive, para que sejam obtidas informações exclusivas a um endereço IP único de um arquivo de *log*, foi apresentado o programa *Filtro.java*, descartando-se todos os acessos realizados por outros endereços IPs que não sejam o de interesse.

Entretanto, sabe-se que nem sempre um mesmo usuário acessa um *web site* a partir do mesmo endereço IP em qualquer acesso. Logo, uma das atividades futuras deste trabalho é o estudo e implementação de algum mecanismo de caracterização dos usuários independentemente de seu endereço IP. Tal caracterização não é uma atividade simples,

uma vez que nem sempre o *web site* se preocupa em apresentar um cadastro de usuários, nem os projetistas do *web site* têm o interesse explícito de saber QUEM é o usuário que está fazendo um acesso inicial ao *web site*. A caracterização do usuário somente se faz necessária basicamente em casos específicos como realizações de compras, consultas, cadastros ou outras atividades similares.

Visando solucionar este problema, pode-se estudar a implementação de alguma rotina prévia ao primeiro acesso ao *web site* que se preocupe **apenas** com a caracterização do usuário, sendo que tal rotina pode ser única para qualquer *web site* que possa a vir ser avaliado empregando-se o Ergo-Monitor. Mas essa abordagem pode tornar o acesso ao *web site* ocioso e cansativo para o usuário que pode, então, desistir facilmente do acesso. Trata-se, portanto, de uma questão bastante interessante para trabalhos e estudos futuros.

#### **5.2.4 – Realizar Manutenções nos Programas do Log Control**

Atualmente, o Ergo-Monitor requer como entrada, arquivos que contenham as seqüências de páginas que descrevam as tarefas a serem monitoradas, e esses arquivos estão apresentando algumas limitações referentes ao fato de que uma tarefa possa ser monitorada a partir de um dentre vários conjuntos diferentes de seqüências de páginas.

As definições de seqüências de páginas tornam a etapa de definição do Modelo de Tarefas e Comportamentos Monitorados pelo Ergo-Monitor uma atividade cansativa, difícil e propensa a erros em sua condução. Logo, em trabalhos futuros devem ser estudadas possíveis modificações nos programas para que os arquivos de entrada possam ser editados empregando algumas estruturas de repetições e recursão das declarações de modo a facilitar a composição das seqüências de páginas que definem os possíveis comportamentos dos usuários quando da realização de uma tarefa.

Uma possível forma de se facilitar a definição dessas páginas é o emprego da Notação de Backus-Naur [Wirth, 1982]. Os programas do Ergo-Monitor poderiam ser capazes de entender esta nova notação e realizar os monitoramentos das seqüências de páginas acessadas durante as interações que ficam armazenadas nos *log files*.

### **5.2.5 – Desenvolvimento do Modelo de Tarefas Fechadas e Objetivas**

Uma outra atividade que deve ser realizada com o intuito de aprimorar o Ergo-Monitor, aumentando a abrangência de sua utilização é a continuidade na definição de novas tarefas e comportamentos. Com isso, novas taxas e métricas podem ser determinadas para outras tarefas que não apenas as descritas neste trabalho, o que pode aumentar a qualidade dos resultados que venham a ser produzidos.

Dessa forma, espera-se que com o apoio da definição de uma sintaxe mais abrangente para a atividade de modelar tarefas e comportamentos (tal como apresentada na seção anterior), possam ser produzidos laudos de monitoramentos com maior grau de confiabilidade.

O estudo dos *cookies*, que podem ser coletados durante as interações, pode proporcionar mecanismos para auxiliar nesta tarefa de aperfeiçoamento do Modelo de Tarefas de Comportamentos, uma vez que podem ser coletados novos dados a respeito das interações.

O Capítulo a seguir apresenta as referências bibliográficas empregadas nesta tese de doutoramento que serviram, tanto para a motivar a definição do Ergo-Monitor, quanto para dar suporte à proposição e ao desenvolvimento deste ambiente.

## 6 – Referências Bibliográficas

[Andres, 2000] Andres, D. P.; Elaboração de Listas de Verificação de Usabilidade para Sites de Comércio Eletrônico, Dissertação de Mestrado, Programa de Pós-Graduação em Engenharia de Produção, Universidade Federal de Santa Catarina, Agosto de 2000.

[Arlitt et alli, 1996] Arlitt, M. and Williamson, C.; Web Server Workload Characterization: The Search for Invariants; In *Proceedings of ACM Sigmetrics Conference*, pp. 126-137, Maio, 1996.

[Bastien, 1998] Bastien, J. M. C., Scapin, D. L.; A Validation of Ergonomic Criteria for The Evaluation of Human-Computer Interfaces; *International Journal of Human-Computer Interaction*, 4, 183-196, 1998.

[Bobby, 2002] Ferramenta Bobby, disponível em [www.cast.org](http://www.cast.org), acessado em maio, 2002.

[Brajnik, 2000] Brajnik, G.; Automatic Web Usability Evaluation: What Needs To Be Done, In *Proc. of 6th Human Factors and the Web Conference*, Austin, Texas - EUA, Junho, 2000.

[Chack, 2002] Chack, A.; *Usability Tools: A Useful Start*; disponível em <http://www.webtechniques.com/archives/2000/08/stratrevu/>; acessado em julho, 2002.

[Cybis, 2000] Cybis, W. A., Scapin, D. L. Andres, D. P.; Especificação de Método de Avaliação Ergonômica de Usabilidade para Sites/Web de Comércio Eletrônico; *IHC 2000 - Gramado*, Outubro de 2000.

[Cybis et alli, 2002] Cybis, W. A., Scapin, D. L., Morandini, M.; Conception and Definition of Ergo-CoIn, A Tool to Support Usability Evaluation of Web Sites; *Rapport Technique, March 2002*, INRIA – Rocquencourt, França.

[Downton, 1992] Downton, A.; *Engineering the Human-Computer Interface*, Mc-Graw Hill Book Company, 1992.

[Farenc et. Alli, 1996] Farenc, C.; Liberati, V., Barthet, M.; Automatic Ergonomic Evaluation: What Are the Limits?; *CADUI'96 – 2<sup>nd</sup> International Workshop on Computer-Aided Design of User-Interfaces*, Namur, Bélgica, Junho, 1996.

[Graham, 2000] Jeffrey Graham, Usability Testing Basics, [http://usability.gov/methods/usability\\_testing.html](http://usability.gov/methods/usability_testing.html), Julho de 2000.

[Gray, 1990] Using Protocol Analyses and Drawings to Study Mental Model Construction During Hypertext Navigation, *International Journal of Human-Computer Interaction*, 2 (4), 359-377, Abril, 1990.

[Hardman, 1986] Hardman, L.; Evaluating the Usability of the Glasgow Online Hypertext, *Hypermedia*, 1 (1), 34-63, Janeiro, 1986.

[Instone, 1997] Instone, K. Usability hueristics for the Web. available at: <http://webreview.com/97/10/10/usability/sidebar.htm>

[ISO 9126] ISO/DIS 9126; *International Standard ISO/IEC 9126. Information technology - Software product evaluation -- Quality characteristics and guidelines for their use*, International Organization for Standardization, International Electrotechnical Commission, Geneva.

[ISO 9241] ISO/DIS 9241; *Dialogue Principles in Guiding the Evaluation of User Interfaces*, Guidance on Usability, 1997.

[Ivory, et. al., 2001] Ivory, M., Hearst, M.; *State of Art in Automating Usability Evaluation of User Interfaces*; CAN Computing Surveys, 33(4), Dezembro, 2001, pp. 1-47.

[Java, 2003] Java Language Main Web Site; *The Source of Java Aplications*; disponível em <http://www.java.sun.com>; acessado em maio de 2003.

[Jones, 1997] Jones, S. and Hewitt, J. (1997). Heuristic evaluation of web site usability: experiences from two case studies. In P. Thomas and D. Withers (Eds.), *Human Computer Interaction '97: Adjunct Proceedings of the 12th British Computer Society Annual*,. Bristol, Inglaterra.

[Kaiser, 2001] Kaiser, J. F.; *Criteria for Web Site Evaluation: Points to consider when performing a site critique*; <http://webdesign.about.com/library/weekly/aa071801a.htm>; Julho, 2001.

[Kapoun, 1998] Kapoun, J.; <http://www.ala.org/acrl/undwebev.html>, at Southwest State University, Julho/Agosto, 522-523, 1998.

[LabiUtil, 2003] Laboratório de Utilizabilidade, Depto. de Informática, Universidade Federal de Santa Catarina, <http://www.labiutil.inf.ufsc>; acessado em abril de 2003.

[Lacerof and Paternó, 1998] Lacerof, Andreas, Paternó, F.; Automatic Support for Usability Evaluation, *IEEE Transactions on Software Engineering*, vol. 24, no. 10, outubro de 1998.

[Lea, 1988] Lea, M.; *Evaluating User Interface Designs*, John Wiley & Sons, 1988.

[Lift, 2002] WebSAT – Ferramenta de Apoio à Avaliação de *Web Sites*, desenvolvida pela UsableNet Company, disponível em [www.usablenet.com](http://www.usablenet.com), acessada em maio de 2002.

[Mach-5, 2002] Mach 5 Enterprises Corporation, *Fast Stats Analyzer: Lightning Web Site Statistics*; disponível em [www.mach5.com](http://www.mach5.com), acessada em abril de 2002.

[Macleod et alli, 1997] Macleod, M., Bowden, R., Bevan, N., Curson, I.; The MUSiC Performance Measurement Method, *Behaviour and Information Technology*, agosto, 1997.

[Matias, 1995] Matias, M. *Checklist: Uma Ferramenta de Suporte à Avaliação Ergonômica de Interfaces*; Dissertação de Mestrado submetida ao PPGEF, Universidade Federal de Santa Catarina, Florianópolis, 1995.

[Max, 2002] Ferramenta Max Web Criteria, disponível em [www.webcriteria.com](http://www.webcriteria.com), acessada em maio de 2002.

[Meira Jr, et. alli, 2002] Meira Jr., W.; Murta, C. D.; Campos, S. V. A.; Guedes Neto, D. O.; *Sistemas de Comércio Eletrônico – Projeto e Desenvolvimento*; Editora Campus, 2002.

[Netcraft, 2003] Netcraft Webserver Survey, disponível em [www.netcraft.com](http://www.netcraft.com), acessado em julho de 2003.

[Nielsen, 1995] Nielsen, J.; *Advances in HCI Vol. 5*. Ablex, Norwood, NJ, ISBN 1-56750-199-0 (hardcover), 1-56750-196-6 (softcover).

[Nielsen, 2000] Nielsen, J.; *Projetando Web Sites - Designing Web Usability*, Editora Campus, 2000.

[Nielsen, 2002] Nielsen, J. *Heuristic Evaluation*. available at:  
<http://www.useit.com/papers/heuristic/>

[Nielsen et. alli, 2002] Nielsen, J., Tahir, M.; *Homepage Usability – 50 Websites Deconstructed*; New Riders Publishing, 2002.

[PHP, 2002] PHP Language Official *Web Site*; <http://www.php.net>; acessado em junho de 2002.

[Preece et alli. 1994] Preece, J., Rogers, Y., Sharp, H., Benyon, D., Holland, S., Carey, T.; *Human Computer Interaction*, Addison-Wesley, 1994.

[Perlman, 2000] Perlman, G.; *Web-Based User Interface Evaluation with Questionnaires*; disponível em <http://www.acm.org/~perlman/question.html>, acessado em maio, 2000.

[Pressman, 1994] Pressman, R. S.; *Software Engineering - A Practitioner's Approach*, Makron Books, 1994.

[Rocha e Baranauskas, 2000] Rocha, H. V., Baranauskas, M. C.; *Design e Avaliação de Interfaces Humano-Computador*, Escola de Computação, USP, São Paulo, 2000.

[Scapin et. alli., 2001] Scapin, D., Leulier, C., Vanderbonckt, J., Mariage, C., Bastien, C, Palanque, P., Farenc, C., Bastilde, R.; *Towards Automated Testing of Web Usability Guidelines*, 2001, In. Tools for Working with Guidelines London Springer, pp. 293-304.

[Scapin, 1997] Scapin, D. L., Bastien, J. M. C.; Ergonomic Criteria for Evaluating the Ergonomic Quality of Interactive Systems, *Behaviour and Information Technology*, vol 16, no. 4/5, 1997.

[Scholtz, 2001] Scholtz, J.; Adaptation of Traditional Usability Testing Methods for Remote Testing, *Proceedings of the 34<sup>th</sup> International Conference on System Sciences*; Havai, EUA, 2001.

[Sears, 1993] Sears, A.; “Layout Appopriateness: A Metric For Evaluating User Interface Widget Layout”, *IEEE Transactions on Software Engineering*, Vol. 19, Num. 7, Jul 1993.

[Shneidermann, 1992] Shneidermann, B.; *Designing the User Interface: Strategies for Effective Human-Computer Interaction*, Addison-Wesley, 1992, 2<sup>a</sup>. Ed.

[Smith, 1997 et alli] Smith, P.A.; Newman, I.A., Parks, L.M.; *Virtual Hierarchies and Virtual Networks: Some Lessons Frm Hypermedia Usability Research Applied to the World Wide Web*; <http://www.hbuk.co.uk/ap/ojhcs/webusability/smith/smith.html>

[Treu, 1994] Treu, S.; *User Interface Evaluation: A Structured Approach*, Plenum Press, 1994.

[UsableWeb, 2002] *UsableWeb – Links About Usability*, disponível em <http://www.usableweb.com>, acessado em novembro de 2002.

[Usability, 2001] disponível em [http://usability.gov/methods/usability\\_testing.html](http://usability.gov/methods/usability_testing.html), de <http://www.usableweb.com> em Fevereiro de 2001.

[WebSAT, 2002] WebSAT – Ferramenta de Apoio à Avaliação de *Web Sites*, desenvolvida pela NIST – National Institute of Standards and Technology, disponível em [zing.ncsl.nist.gov/webmet/sat/websat-process.html](http://zing.ncsl.nist.gov/webmet/sat/websat-process.html), acessada em maio de 2002.

[W3C, 2002] *World Wide Web Consortium's (W3C) Web Accessibility Initiative*, disponível em <http://www.w3c.com>, acessado em maio de 2002.

[Wirth, 1982] Wirth, N., *Programming in Modula-2*, Berlin, Heidelberg: Springer, 1982.

## Apêndice A - Listagem dos Programas do Log Control

Nesse Apêndice serão apresentados os códigos-fonte de dois dos programas escritos em linguagem Java que compõem o Log Control. Esses programas são: Filtro.Java e Buscar.Java. Seus códigos-fonte são apresentados a seguir:

### A.1 - Filtro.Java

Filtra os *log files* de modo que, a partir do arquivo inicial de *log* (normalmente o *access.log*), seja criado um novo arquivo que contenha apenas as informações de interesse dos avaliadores, que normalmente são as páginas que compõem o *web site*.

<pre>import java.io.*; import java.awt.*; import java.awt.event.*; import javax.swing.*;  class arquivoprocura {     String str, texto, ocorrencias;     BufferedReader in;     BufferedWriter out;     int status,base;      public arquivoprocura(String origem,String destino) {         status = 1;         try {             in = new BufferedReader(new FileReader(origem));             out = new BufferedWriter(new FileWriter(destino));         }         catch(IOException e) {             status = 0;         }     }      void procuraString(boolean asp, boolean html, boolean     php,String ip) {         status=1;         try {             while((str = in.readLine()) != null) {                 base=str.indexOf("-");                 if(ip.indexOf(str.substring(0,base-1))&gt;=0) {                     if (asp) {                         if (str.indexOf("asp")&gt;0) {                             out.write(str);                             out.newLine();                         }                     }                     if (html) {                         if (str.indexOf("htm")&gt;0) {                             out.write(str);                             out.newLine();                         }                     }                 }             }         }     } }</pre>	<pre>asp = new JRadioButton("ASP"); asp.setSize(150,20); asp.setLocation(80,100); asp.setForeground(Color.WHITE); asp.setBackground(Color.BLACK); c.add(asp);  html = new JRadioButton("HTML"); html.setSize(150,20); html.setLocation(80,120); html.setForeground(Color.WHITE); html.setBackground(Color.BLACK); c.add(html);  php = new JRadioButton("PHP"); php.setSize(150,20); php.setLocation(80,140); php.setForeground(Color.WHITE); php.setBackground(Color.BLACK); c.add(php);  lip = new JLabel("IP"); lip.setSize(150,20); lip.setLocation(20,180); lip.setForeground(Color.WHITE); c.add(lip);  ip = new JTextField("127.0.0.1"); ip.setSize(150,20); ip.setLocation(20,200); c.add(ip);  lorig = new JLabel("Arquivo de origem"); lorig.setSize(150,20); lorig.setLocation(20,230); lorig.setForeground(Color.WHITE); c.add(lorig);  orig = new JTextField("access.log"); orig.setSize(150,20); orig.setLocation(20,250); c.add(orig);  ldest = new JLabel("Arquivo de Destino"); ldest.setSize(150,20);</pre>
---	--

<pre>         if (php) {             if (str.indexOf("php")&gt;0) {                 out.write(str);                 out.newLine();             }         }     } }  in.close(); out.close(); } catch(IOException e) {     status = 0; } } }  public class filtro extends JFrame implements ActionListener, WindowListener{     Container c;     String arquivo,arqorigem;     JButton btstart;     JTextField ip,dest,orig;     JTextArea apresenta;     JLabel lip,ldest,lorig;     JRadioButton asp,html,php;      // construtor     public filtro(){         setSize(300,430);         setTitle("Filtro");         c = new Container();          c = getContentPane();         c.setLayout(null);         c.setBackground(Color.black);          apresenta = new JTextArea(" Escolha, dentre as opções abaixo, \n as extensões de páginas"+ " a serem \n filtradas, tendo-se como base o log file \n do servidor Apache.");         apresenta.setSize(260,70);         apresenta.setLocation(20,20);         apresenta.setEditable(false);         apresenta.setBackground(Color.BLACK);         apresenta.setForeground(Color.CYAN);         c.add(apresenta); </pre>	<pre>         ldest.setLocation(20,280);         ldest.setForeground(Color.WHITE);         c.add(ldest);          dest = new JTextField("access.fil");         dest.setSize(150,20);         dest.setLocation(20,300);         c.add(dest);          btstart = new JButton("Filtrar");         btstart.setSize(250,40);         btstart.setLocation(20,330);         c.add(btstart);         btstart.addActionListener(             new ActionListener()             {                 public void actionPerformed(ActionEvent e)                 {                     arqorigem = orig.getText();                     arquivo = dest.getText();                     arquivoprocura arq;                     arq = new arquivoprocura(arqorigem,arquivo);                     arq.procuraString(asp.isSelected(),html.isSelected() ,php.isSelected() , ip.getText());                 }             }         );          show();     }      // método.      //Tratadores de eventos     public void actionPerformed(ActionEvent e){}     public void windowClosing(WindowEvent e){}     public void windowActivated(WindowEvent e) {}     public void windowClosed(WindowEvent e) {}     public void windowDeactivated(WindowEvent e) {}     public void windowDeiconified(WindowEvent e) {}     public void windowIconified(WindowEvent e) {}     public void windowOpened(WindowEvent e) {}      //Main     public static void main(String[] args) {         filtro app;         app = new filtro();          //Listener do evento fechar janela         app.addWindowListener(new WindowAdapter() {             public void windowClosing(WindowEvent e) {                 System.exit(0);             }         });     } } </pre>
---	--

**Figura A.1** – Código-Fonte do Programa *filtro.java*

## A.2 – Buscar.Java

Trata-se do programa de maior importância no Log Control. A partir de um arquivo texto de entrada com todas as seqüências de páginas que determinam os comportamentos a serem monitorados em uma tarefa, será gerado um arquivo texto já formatado de forma a poder ser importado pelo Excel. Essa importação irá, então, ser a responsável pela determinação numérica das taxas de usabilidade.

<pre> import java.io.*; import java.awt.*; import java.awt.event.*; import javax.swing.*;  class clock {     int segundo, minuto, hora;     public clock() {         segundo=0;         minuto=0;         hora=0;     }      int calculaseg() {         return segundo+minuto*60+hora*3600;     } }  class arquivobuscaativ {     String str, texto, ocorrencias,aux;     BufferedReader in;     BufferedWriter out;     int diftempo,totaltempo,somatempo;     clock datain, datafim;     int status,ocorre,i,base,linha; //linha de saida     boolean calcular;      public void nxtline() throws IOException{         out.newLine();         linha++;     }      public arquivobuscaativ(String destino, String tarefa) {         status = 1;         datain = new clock();         datafim = new clock();         linha=1;         try {             out = new BufferedWriter(new FileWriter(destino));             out.write("TAREFA: " + tarefa.substring(0,tarefa.length()-4));             nxtline();             nxtline();             out.write("TIPO TEMPO QUANTIDADE");             nxtline();         }         catch(IOException e) {             status = 0;         }     } } </pre>	<pre> } } //fecha  while          out.write(" "+somatempo+" "+(ocorre));         nxtline();         //soma          if (i&gt;=0 &amp;&amp; atividades[i+1] != null) {             if(                 (atividades[i].substring(0,atividades[i].length()- 1)).compareTo (                     atividades[i+1].substring(0,atividades[i+1].length()- 1)                 )==0             ) {                 if (at==0)                     at=linha-1;                 else                     bt=linha-1;                 if (aq==0)                     aq=linha-1;                 else                     bq=linha-1;                 cont=1;             }             else {                 if (cont == 1) { </pre>
---	---

<pre>     } }  void procuraString(String[] atividades,String[][] paginas, String fonte) { //totaltempo=tempo da ocorrencia //somatempo=tempo de todas as ocorrencias  String[][] taxas = new String[40][3]; int txi=0,tj=0; status=1; int cont=0; // tem mais do que uma atividade semelhante? 0 ou 1 int i=0; //atividade int j=0; //pagina na atividade int k=0; //contador p/chegar na linha da pagina1  int at=0,aq=0; //intervalos(a,b) na planilha de tempo e quantidade int bt=0,bq=0; //    int linhain=1; //linha de entrada int pagina1line=0; int achoupagina1=0; String pagina1 = new String();  try {  while(atividades[i] != null) {  ***** // ***** Atividade  *****  pagina1 = pagina1line=1; in = new BufferedReader(new FileReader(fonte)); ocorre=0;  out.write(atividades[i]);  calcular=false; totaltempo=0; somatempo=0; achoupagina1=0; linhain=0; while(str = in.readLine() != null) { //procura da atividade if (paginas[i][1] != null) {  if (str.indexOf(pagina1)&gt;0 &amp;&amp; achoupagina1==1 ) { //achou a pagina1 da atividade(2a)  achoupagina1=2;  pagina1line=linhain;  }  if (str.indexOf(pagina1)&gt;0 &amp;&amp; achoupagina1==0 ) { //achou a pagina1 da atividade(1a.)  achoupagina1=1;  }  if (achoupagina1==2 &amp;&amp; j==0) { // vai até a ocorrencia da pagina1 </pre>	<pre> out.write("Soma_"+atividades[i].substring(0,ativida des[i].length()-1)+ "+"=soma(B"+at+":B"+(bt+1)+)"+" "+"=soma(C"+aq+":C"+(bq+1)+)");  taxas[txi][0]=atividades[i].substring(0,atividades[i].l ength()-1);  taxas[txi][1]="(E"+linha+"/B4)"; taxas[txi][2]="(F"+linha+"/C4)";  txi++;  nxtline();  cont = 0 ;  at = bt =aq = bq =0;  }  else {  if(i &gt; 0) {  taxas[txi][0]=atividades[i];  taxas[txi][1]="(B"+(linha-1)+"/B4)";  taxas[txi][2]="(C"+(linha-1)+"/C4)";  txi++;  }  }  else {  if (cont == 1) {  out.write("Soma_"+atividades[i].substring(0,ativida des[i].length()-1)+ "+"=soma(B"+at+":B"+(bt+1)+)"+" "+"=soma(C"+aq+":C"+(bq+1)+)");  taxas[txi][0]=atividades[i].substring(0,atividades[i].l ength()-1);  taxas[txi][1]="(E"+linha+"/B4)"; taxas[txi][2]="(F"+linha+"/C4)";  txi++;  nxtline();  cont = 0 ;  at = bt =aq = bq =0;  }  else {  if(i &gt; 0) { </pre>
---	---

<pre> in = new BufferedReader(new FileReader(fonte));  k=0;  while(k&lt;=(pagina1line)) {  str=in.readLine();  k++;  }  linhain=pagina1line;  achoupagina1=1;  } else {  //proxima pagina  linhain++;  }  } else  linhain++;  base=str.indexOf("[");  if(calcular==true) {  datafim.hora=Integer.parseInt(str.substring(base+1  3,base+15));  datafim.minuto=Integer.parseInt(str.substring(base  +16,base+18));  datafim.segundo=Integer.parseInt(str.substring(bas  e+19,base+21));  diftempo=datafim.calculaseg()-datain.calculaseg();  totaltempo+=diftempo;  calcular=false;  }  if  (paginas[i][j]==null) {  (totaltempo&gt;=0) {  somatempo+=totaltempo;  }  else {  ocorre++;  totaltempo=0;  j=0;  }  if(j&lt;1) {  if  (paginas[i][j]!=null &amp;&amp; paginas[i][j].indexOf("!")&gt;=0) {  if  (str.indexOf(paginas[i][j].substring(1,paginas[i][j].length()))&gt;0) </pre>	<pre> taxas[txi][0]=atividades[i];  taxas[txi][1]="(B"+(linha-1)+"/B4)";  taxas[txi][2]="(C"+(linha-1)+"/C4)";  txi++;  }  }  }  //soma  // Fim Atividade  i++;  } //fecha while  out.newLine();  out.newLine();  out.newLine();  out.write("TAXAS  "+taxas[0][0]+  "+taxas[0][1]+  "+taxas[0][2];  nxtline();  txi=1;  while(taxas[txi][0] != null) {  out.write("  "+taxas[txi][0]+  "+taxas[txi][1]+  "+taxas[txi][2];  nxtline();  txi++;  }  in.close();  out.close();  } //fecha try  catch(IOException e) {  status = 0;  }  } //fecha função  } //fecha classe  if public class buscar extends JFrame implements  ActionListener, WindowListener{  Container c;  String arqfon,arqdest,aux;  JButton btstart;  JTextField tarefa,source,dest;  JTextArea apresenta;  JLabel ltarefa,lsource,ldest;  BufferedReader ativsite;  String[][] paginas;  String[] atividades;  // construtor  public buscar(){  setSize(330,400);  setTitle("Busca de Tarefas");  c = new Container();  c = getContentPane();  c.setLayout(null);  c.setBackground(Color.black); </pre>
--	---



<pre> } else {     datain.hora=Integer.parseInt(str.substring(base+13 ,base+15));      datain.minuto=Integer.parseInt(str.substring(base+ 16,base+18));      datain.segundo=Integer.parseInt(str.substring(base +19,base+21));          calcular=true;          j++;      }      if(paginas[i][j]!=null &amp;&amp; str.indexOf(paginas[i][j])&gt;0 ) {     datain.hora=Integer.parseInt(str.substring(base+13 ,base+15));      datain.minuto=Integer.parseInt(str.substring(base+ 16,base+18));      datain.segundo=Integer.parseInt(str.substring(base +19,base+21));      calcular=true;      j++;  else {  totaltempo=0;  j=0; } } </pre>	<pre> if (aux.indexOf("#")==0) {     i++;      atividades[i]=aux.substring(1,aux.length());     j=0;      } else {  paginas[i][j]=aux;     j++;      }  } catch(IOException ex) {  }      arqfon=source.getText();     arqdest=dest.getText();     arquivobuscaativ arq;     arq = new     arquivobuscaativ(arqdest,tarefa.getText());      arq.procuraString(atividades,paginas,arqfon);      System.exit(0);  } });  show(); }  // método.  //Tratadores de eventos public void actionPerformed(ActionEvent e){} public void windowClosing(WindowEvent e){} public void windowActivated(WindowEvent e) {} public void windowClosed(WindowEvent e) {} public void windowDeactivated(WindowEvent e) {} public void windowDeiconified(WindowEvent e) {} public void windowIconified(WindowEvent e) {} public void windowOpened(WindowEvent e) {}  //Main public static void main(String[] args) {     buscar app;     app = new buscar();      //Listener do evento fechar janela     app.addWindowListener(new WindowAdapter() {         public void windowClosing(WindowEvent e) {             System.exit(0);         }     }); } } </pre>
--	--

**Figura A.2** – Código-Fonte do Programa *buscar.java*

## Apêndice B - Códigos-Fonte das Páginas do Web Site da Ruth Fraldas

Neste Apêndice são apresentados alguns programas que compõem o código-fonte das páginas do *web site* da Ruth Fraldas. Ao todos são 258 programas e mais de 35000 linhas de código. Basicamente, a linguagem empregada foi o PHP e o Banco de Dados utilizado é o MySQL. Tanto a linguagem PHP quanto as particularidades do Banco de Dados MySQL já foram apresentadas anteriormente no Capítulo 2.

As Figuras B.1 e B.2 apresentam, respectivamente, os códigos-fonte escritos na linguagem PHP para a implementação dos programas *index.php* e *login.php*.

<pre> &lt;?php     \$area = 0; // 0 - NL, 1 - ADM, 2 - COMP     require("header.inc");     require_once("classcategorias.inc");     require_once("classmaterias.inc");     require_once("classmateria_vinculos.inc");     require_once("classmateria_links.inc");     require_once("classpromos.inc");     require_once("classpromo_vinculos.inc");     require_once("classprodutos.inc");     require_once("classdicas.inc");     require("body.inc");     \$cat = new categorias(\$con);     \$promo = new promo_vinculos(\$con);     \$dica = new dica(\$con);     \$mat = new materias(\$con);     \$prod = new produtos(\$con); ?&gt;  &lt;table border=0 cellpadding=0 cellspacing=0 width=550 valign="top"&gt; &lt;tr&gt;     &lt;td align="left" valign="top" width="550"&gt;         &lt;span align="left"             class="titulo"&gt;                 &lt;?php                     if(!empty(\$cod_cat))                     {                         \$cat=                         &gt;cod_categoria = \$cod_cat;                         echo                         implode("::",array_reverse(explode("",\$cat-                         &gt;arvore(\$cod_cat))));                     }                     elseif                     (!empty(\$cod_dica)) {                         \$dica=                         &gt;cod_dica = \$cod_dica;                         &gt;busca();                         echo                         \$dica-&gt;titulo."&lt;br&gt;&lt;br&gt;";                         echo                         \$dica-&gt;texto;                     }                 }             }         }     } </pre>	<pre> &lt;table border=0 cellpadding=0 cellspacing=0 width=550 valign="top"&gt;     &lt;tr&gt;         &lt;td align="left" width="550"             class="promo_titulo"&gt;&lt;img             border=0&gt;&lt;/td&gt;     &lt;/tr&gt;     &lt;tr&gt;         &lt;td align="left" height="1" width="550"             class="promo_titulo" bgcolor="#000099"&gt;&lt;/td&gt;     &lt;/tr&gt;     &lt;tr&gt;         &lt;td align="center" height="50"             valign="middle" width="550"&gt;             &lt;?=\$rel_mat?&gt;         &lt;/td&gt;     &lt;/tr&gt; &lt;/table&gt; &lt;?php     }     \$rel_promo     =     \$promo-     &gt;exibir(\$cod_promo,2,"", "link2",6,\$cod_cat);     if(!empty(\$rel_promo) and empty(\$cod_dica)) { ?&gt; &lt;table border=0 cellpadding=0 cellspacing=0 width=550 valign="top"&gt;     &lt;tr&gt;         &lt;td align="left" width="550"             class="promo_titulo"&gt;&lt;img             border=0&gt;&lt;/td&gt;     &lt;/tr&gt;     &lt;tr&gt;         &lt;td align="left" height="1" width="550"             class="promo_titulo" bgcolor="#000099"&gt;&lt;/td&gt;     &lt;/tr&gt;     &lt;tr&gt;         &lt;td align="center" valign="middle"             width="550"&gt;             &lt;?=\$rel_promo?&gt;         &lt;/td&gt;     &lt;/tr&gt;     &lt;tr&gt;         &lt;td height="20"&gt;&lt;/td&gt;     &lt;/tr&gt; </pre>
--	---

```

else
Fraldas : Tudo p/ seu bebê<br><br>";
?></span>
</td>
</tr>
<?php
?>
if(!empty($cod_cat)) {
<tr>
<td align="center" height="50"
valign="middle" width="550">
<?php
$cat=
>exibir($cod_cat,"",4,"link2");?>
</td>
</tr>
<?php
?>
}
</table>
<?php
if(empty($cod_promo) and empty($cod_dica)) {
$tipo = ($cod_cat!=0) ? "C" : "H";
$chave = ($tipo=="H") ? 0 : $cod_cat;
$rel_mat = $mat->exibir($tipo,$chave);
if(!empty($rel_mat)) {
?>
echo "Ruth
</table>
<?php
}
$rel_produtos = $rprod->rel_produtos($cod_cat);
if(!empty($rel_produtos)) {
?>
<table border=0 cellpadding=0 cellspacing=0 width=550
valign="top">
<tr>
<td align="left" width="550"
class="promo_titulo"><img
border=0></td>
</tr>
<tr>
<td align="left" height="1" width="550"
class="promo_titulo" bgcolor="#000099"></td>
</tr>
<tr>
<td align="center" height="50"
valign="middle" width="550">
<?=$rel_produtos?>
</td>
</tr>
</table>
<?php
}
if(!empty($cod_cat))
echo
" <p><p><a
href=javascript:window.history.go(-1)><img
src=/ruthfraldas/imagens/voltar.jpg border=0></a>";
require("footer.shtml");
?>
</body>
</html>

```

**Figura B.1** – Código-Fonte do Programa index.php

```

<?php
    $area = 0; // 0 - NL, 1 - ADM, 2 - COMP

    ob_start();
    require("header.inc");
    require("body.inc");

    if($usr_h->user_id!=0) {
        header("Location: gera_pedido.php");
        ob_end_clean();
        exit;
    }
    ob_end_flush();
?>

<table border=0 cellpadding=0 cellspacing=0 width=550
valign="top">
    <tr>
        <td align="left" valign="top" height="60"
width="550">
            <span align="left"
class="titulo_form">Passo 1 de 3: Identificação do
Comprador</span>
        </td>
    </tr>
</table>

<script>
    var campos = new Array(0,1);
    var nomes = new Array("Login","Senha");
    var tipos = new Array(9,9);
    var status = new Array(1,1);
    var formid = 1;
</script>
<form name="usr" method="post"
action="<?=$PHP_SELF?>" onsubmit="return
valida_form(formid,campos,nomes,tipos,status);">
<table border="0" cellpadding="0" cellspacing="5"
width="450" align="center" class="label_form">
<tr>
    <td width="60%" valign="top">
        <table border=0 class="label_form">
            <tr>
                <td width="30%"
height="30">Login:</td>
                <td width="70%"
height="30"><input type="text" name="usuario"
size="10"></td>
            </tr>
            <tr>
                <td width="30%"
height="30">Senha:</td>
                <td width="70%"
height="30"><input type="password" name="senha"
size="10" maxlength="10"></td>
            </tr>
        </table>
    </td>
    <td align="center" colspan="2" height="30"><a
class="link2" href="usr_man_cadastro.php?opt=E"
class="link2">Esqueceu sua Senha?</a></td>
    </tr>
</table>
</form>

<?php
    require("footer.shtml");
?>
</body>
</html>

```

**Figura B.2** – Código-Fonte do Programa login.php

## **Apêndice C - Arquivos-Texto de Entrada dos Programas do Log Control**

Neste Apêndice são apresentados os arquivos que devem servir como entrada para o Log Control. Deve existir um arquivo específico para cada uma das tarefas a serem monitoradas, e, em alguns casos, como por exemplo, *compras* para o *web site* da Ruth Fraldas, pode existir mais de um arquivo de entrada e a execução do Log Control deverá ser feita por passos.

Esses arquivos definem, dentro de uma notação específico e detalhada no Capítulo 4, as possibilidades de acessos às páginas do *web site* para que a tarefa a ser monitorada seja realizada. E, é claro, esses arquivos referem-se às páginas do *web site* da empresa Ruth Fraldas. Ao todo são 6 tarefas a serem monitoradas, sendo que a tarefa de compras precisou ser dividida em outras “sub-tarefas” e requisitou mais arquivos. As Figuras C.1 a C9 apresentam esses arquivos.

#Acesso busca_avancada.php #Help busca_avancada.php ajuda.php?assunto=buscaavancada #Mensagem de Erro busca_avancada.php erro.php #Abandono0 ?? #Abandono1 busca_avancada.php busca_avancada.php?acao=B !cesta_compras.php #Abandono2 busca_avancada.php busca_avancada.php?acao=B !ver_produto.php #Abandono3 ?? #Abandono4 ?? #CancErro0 ?? #CancErro1 ?? #CancErro2 ?? #CancErro3 ?? #CancErro4 ?? #Desistência0 busca_avancada.php !busca_avancada.php?acao=B #Desistência1 ?? #Desistência2 ?? #Desistência3 ?? #Desistência4 ??	#InSucesso0 ?? #InSucesso1 ?? #InSucesso2 ?? #InSucesso3 ?? #InSucesso4 ?? #Repetição0 ?? #Repetição1 busca_avancada.php erro.php busca_avancada.php #Repetição2 busca_avancada.php busca_avancada.php?acao=N busca_avancada.php #Repetição3 ?? #Repetição4 ?? #Sucesso0 ?? #Sucesso1 busca_avancada.php busca_avancada.php?acao=B cesta_compras.php #Sucesso2 busca_avancada.php busca_avancada.php?acao=B ver_produto.php cesta_compras.php #Sucesso3 ?? #Sucesso4 ??
---	---

**Figura C.1** – Definição das Páginas Acessadas para a Tarefa *Busca Avançada*

#Acesso rel_produtos.php #Help ?? #Mensagem de Erro ?? #Abandono0 ?? #Abandono1 ?? #Abandono2 ?? #Abandono3 ?? #Abandono4 ?? #Abandono0 ?? #CancErro0 ?? #CancErro1 ?? #CancErro2 ?? #CancErro3 ?? #CancErro4 ?? #Desistencia0 ?? #Desistência1 index.php rel_produtos.php !cesta_compras.php #Desistência2 index.php rel_produtos.php !ver_produto.php #Desistencia3 ?? #Desistencia4 ??	#Insucesso0 ?? #Insucesso1 index.php !rel_produtos.php #Insucesso2 ?? #Insucesso3 ?? #Insucesso4 ?? #Repeticao0 ?? #Repeticao1 ?? #Repeticao2 ?? #Repeticao3 ?? #Repeticao4 ?? #Sucesso1 index.php rel_produtos.php cesta_compras.php #Sucesso2 index.php rel_produtos.php ver_produto.php #Sucesso3 ?? #Sucesso4 ??
--	--

**Figura C.2** – Definição das Páginas Acessadas para a Tarefa *Busca Simples*

#Acesso usr_man_cadastro.php #Help Help_cadastro.php #Mensagem Erro Msg_erro_cadastro.php #Abandono0 ?? #Abandono1 ?? #Abandono2 ?? #Abandono3 ?? #Abandono4 ?? #CancErro0 ?? #CancErro1 usr_man_cadastro.php erro.php !usr_man_cadastro.php?opt=IG #CancErro2 usr_man_cadastro.php erro.php erro.php !usr_man_cadastro.php?opt=IG #CancErro3 usr_man_cadastro.php erro.php erro.php erro.php !usr_man_cadastro.php?opt=IG #Canc4 usr_man_cadastro.php erro.php erro.php erro.php erro.php !usr_man_cadastro.php?opt=IG #Desistência0 usr_man_cadastro.php !usr_man_cadastro.php?opt=IG #Desistência1 ?? #Desistência2 ?? #Desistência3 ?? #Desistência4 ??	#Insucesso0 ?? #Insucesso1 ?? #Insucesso2 ?? #Insucesso3 ?? #Insucesso4 ?? #Repetição0 ?? #Repetição1 usr_man_cadastro.php erro.php usr_man_cadastro.php #Repetição2 usr_man_cadastro.php erro.php erro.php usr_man_cadastro.php #Repetição3 usr_man_cadastro.php erro.php erro.php erro.php usr_man_cadastro.php #Repetição4 usr_man_cadastro.php erro.php erro.php erro.php erro.php usr_man_cadastro.php #Sucesso0 usr_man_cadastro.php usr_man_cadastro.php?opt=IG #Sucesso1 usr_man_cadastro.php erro.php usr_man_cadastro.php?opt=IG #Sucesso2 usr_man_cadastro.php erro.php erro.php usr_man_cadastro.php?opt=IG #Sucesso3 usr_man_cadastro.php erro.php erro.php erro.php usr_man_cadastro.php?opt=IG #Sucesso4 usr_man_cadastro.php erro.php erro.php erro.php usr_man_cadastro.php?opt=IG
---	--

**Figura C.3** – Definição das Páginas Acessadas para a Tarefa *Cadastro*

#Acessos	#Desistencia0
fale_conosco.php	fale_conosco.php
#Help	!fale_conosco.php?opt=G
fale_conosco.php	#Desistencia1
help.php	??
#MensagemErro	#Desistencia2
fale_conosco.php	??
msg_erro.php	#Desistencia3
#Abandono0	??
??	#Desistencia4
#Abandono1	??
??	#Insucesso0
#Abandono2	??
??	#Insucesso1
#Abandono3	??
??	#Insucesso2
#Abandono4	??
??	#Insucesso3
#CancErro0	??
??	#Insucesso4
#CancErro1	??
fale_conosco.php	#Repeticao0
erro.php	??
!fale_conosco.php?opt=G	#Repeticao1
#CancErro2	??
fale_conosco.php	#Repeticao2
erro.php	??
erro.php	#Repeticao3
!fale_conosco.php?opt=G	??
#CancErro3	#Repeticao4
fale_conosco.php	??
erro.php	#Sucesso0
erro.php	fale_conosco.php
erro.php	fale_conosco.php?opt=G
!fale_conosco.php?opt=G	#Sucesso1
#CancErro4	fale_conosco.php
fale_conosco.php	erro.php
erro.php	fale_conosco.php?opt=G
erro.php	#Sucesso2
erro.php	fale_conosco.php
!fale_conosco.php?opt=G	erro.php
	erro.php
	fale_conosco.php?opt=G
	#Sucesso3
	fale_conosco.php
	erro.php
	erro.php
	erro.php
	fale_conosco.php?opt=G
	#Sucesso4
	fale_conosco.php
	erro.php
	erro.php
	erro.php
	fale_conosco.php?opt=G

**Figura C.4** – Definição das Páginas Acessadas para a Tarefa *Fale Conosco*

#Acessos usr_man_cadastro.php?opt=S #Help ?? #MensagemErro ?? #Abandono0 usr_man_cadastro.php?opt=S !usr_man_cadastro.php?opt=SG #Abandono1 ?? #Abandono2 ?? #Abandono3 ?? #Abandono4 ?? #CancErro0 ?? #CancErro1 usr_man_cadastro.php?opt=S erro.php !usr_man_cadastro.php?opt=SG #CancErro2 usr_man_cadastro.php?opt=S erro.php erro.php !usr_man_cadastro.php?opt=SG #CancErro3 usr_man_cadastro.php?opt=S erro.php erro.php erro.php !usr_man_cadastro.php?opt=SG #CancErro4 usr_man_cadastro.php?opt=S erro.php erro.php erro.php erro.php !usr_man_cadastro.php?opt=SG #Desistencia0 ?? #Desistencia1 ?? #Desistencia2 ?? #Desistencia3 ?? #Desistencia4 ??	#Insucesso0 ?? #Insucesso1 ?? #Insucesso2 ?? #Insucesso3 ?? #Insucesso4 ?? #Repeticao0 ?? #Repeticao1 ?? #Repeticao2 ?? #Repeticao3 ?? #Repeticao4 ?? #Sucesso0 usr_man_cadastro.php?opt=S usr_man_cadastro.php?opt=SG #Sucesso1 usr_man_cadastro.php?opt=S erro.php usr_man_cadastro.php?opt=SG #Sucesso2 usr_man_cadastro.php?opt=S erro.php erro.php usr_man_cadastro.php?opt=SG #Sucesso3 usr_man_cadastro.php?opt=S erro.php erro.php usr_man_cadastro.php?opt=SG #Sucesso4 usr_man_cadastro.php?opt=S erro.php erro.php erro.php usr_man_cadastro.php?opt=SG
---	--

**Figura C.5** – Definição das Páginas Acessadas para a Tarefa *Trocar Senha*

#Acesso cesta_compras.php #Help cesta_compras.php ajuda.php #Mensagem Erro cesta_compras.php erro.php #Abandono0 cesta_compras.php?opt=l !login.php #Abandono1 cesta_compras.php?opt=l erro.php !login.php #Abandono2 cesta_compras.php?opt=l erro.php erro.php !login.php #Abandono3 cesta_compras.php?opt=l erro.php erro.php erro.php !login.php #Abandono4 cesta_compras.php?opt=l erro.php erro.php erro.php erro.php !login.php #CancErro0 ?? #CancErro1 ?? #CancErro2 ?? #CancErro3 ?? #CancErro4 ?? #Desistencia0 ?? #Desistencia1 ?? #Desistencia2 ?? #Desistencia3 ?? #Desistencia4 ?? #Insucesso0 ?? #Insucesso1 ?? #Insucesso2 ??	#Insucesso3 ?? #Insucesso4 ?? #Repeticao0 ?? #Repeticao1 ?? #Repeticao2 ?? #Repeticao3 ?? #Repeticao4 ?? #Sucesso0 ?? #Sucesso1 cesta_compras.php?opt=l index.php ?cod_cat cesta_compras.php?opt=l&cod_produto= login.php #Sucesso2 cesta_compras.php?opt=l index.php ?cod_cat cesta_compras.php?opt=l&cod_produto= index.php ?cod_cat cesta_compras.php?opt=l&cod_produto= login.php #Sucesso3 cesta_compras.php?opt=l index.php ?cod_cat cesta_compras.php?opt=l&cod_produto= index.php ?cod_cat cesta_compras.php?opt=l&cod_produto= login.php #Sucesso4 cesta_compras.php?opt=l index.php ?cod_cat cesta_compras.php?opt=l&cod_produto= index.php ?cod_cat cesta_compras.php?opt=l&cod_produto= index.php ?cod_cat cesta_compras.php?opt=l&cod_produto= index.php ?cod_cat cesta_compras.php?opt=l&cod_produto= login.php
---	---

**Figura C.6** – Definição das Páginas Acessadas para a Tarefa *Compras (escolha dos produtos)*

#Acesso login.php #Help login.php help.php #MensagemErro login.php erro.php #Abandono0 ?? #Abandono1 ?? #Abandono2 ?? #Abandono3 ?? #Abandono4 ?? #CancErro0 ?? #CancErro1 login.php erro.php !gera_pedido.php #CancErro2 login.php erro.php erro.php !gera_pedido.php #CancErro3 login.php erro.php erro.php !gera_pedido.php #CancErro4 login.php erro.php erro.php erro.php !gera_pedido.php #Desistencia0 ?? #Desistencia1 ?? #Desistencia2 ?? #Desistencia3 ?? #Desistencia4 ??	#Repeticao0 #Insucesso0 ?? #Insucesso1 ?? #Insucesso2 ?? #Insucesso3 ?? #Insucesso4 ?? #Repeticao0 ?? #Repeticao1 ?? #Repeticao2 ?? #Repeticao3 ?? #Repeticao4 ?? #Sucesso0 login.php gera_pedido.php #Sucesso1 login.php erro.php gera_pedido.php #Sucesso2 login.php erro.php erro.php gera_pedido.php #Sucesso3 login.php erro.php erro.php gera_pedido.php #Sucesso4 login.php erro.php erro.php erro.php gera_pedido.php #Sucesso4 login.php erro.php erro.php gera_pedido.php
---	---

**Figura C.7** – Definição das Páginas Acessadas para a Tarefa *Compra (Ir para o Caixa)*

#Acesso gera_pedido.php?passo=3&cartao=1 #Help ?? #MensagemErro gera_pedido.php?passo=3&cartao=1 erro.php #Abandono0 ?? #Abandono1 ?? #Abandono2 ?? #Abandono3 ?? #Abandono4 ?? #CancErro0 ?? #CancErro1 gera_pedido.php?passo=3&cartao=1 erro.php !gera_pedido_msg.php?pedido= #CancErro2 gera_pedido.php?passo=3&cartao=1 erro.php erro.php !gera_pedido_msg.php?pedido= #CancErro3 gera_pedido.php?passo=3&cartao=1 erro.php erro.php erro.php !gera_pedido_msg.php?pedido= #CancErro4 gera_pedido.php?passo=3&cartao=1 erro.php erro.php erro.php erro.php !gera_pedido_msg.php?pedido= #Desistencia0 ?? #Desistencia1 ?? #Desistencia2 ?? #Desistencia3 ?? #Desistencia4 ??	#Insucesso0 ?? #Insucesso1 ?? #Insucesso2 ?? #Insucesso3 ?? #Insucesso4 ?? #Repeticao0 ?? #Repeticao1 ?? #Repeticao2 ?? #Repeticao3 ?? #Repeticao4 ?? #Sucesso0 gera_pedido.php?passo=3&cartao=1 gera_pedido_msg.php?pedido= #Sucesso1 gera_pedido.php?passo=3&cartao=1 erro.php gera_pedido_msg.php?pedido= #Sucesso2 gera_pedido.php?passo=3&cartao=1 erro.php erro.php gera_pedido_msg.php?pedido= #Sucesso3 gera_pedido.php?passo=3&cartao=1 erro.php erro.php erro.php gera_pedido_msg.php?pedido= #Sucesso4 gera_pedido.php?passo=3&cartao=1 erro.php erro.php erro.php gera_pedido_msg.php?pedido=
--	--

**Figura C.8** – Definição das Páginas Acessadas para a Tarefa *Compra (forma pagamento – Cartão)*

#Acesso gera_pedido.php?passo=3&cartao=0 #Help ?? #MensagemErro gera_pedido.php?passo=3&cartao=0 erro.php #Abandono0 gera_pedido.php?passo=3&cartao=0 !gera_pedido_msg.php?pedido= #Abandono1 ?? #Abandono2 ?? #Abandono3 ?? #Abandono4 ?? #CancErro0 ?? #CancErro1 gera_pedido.php?passo=3&cartao=0 erro.php !gera_pedido_msg.php?pedido= #CancErro2 gera_pedido.php?passo=3&cartao=0 erro.php erro.php !gera_pedido_msg.php?pedido= #CancErro3 gera_pedido.php?passo=3&cartao=0 erro.php erro.php erro.php !gera_pedido_msg.php?pedido= #CancErro4 gera_pedido.php?passo=3&cartao=0 erro.php erro.php erro.php erro.php !gera_pedido_msg.php?pedido= #Desistencia0 ?? #Desistencia1 ?? #Desistencia2 ?? #Desistencia3 ?? #Desistencia4 ??	#Insucesso0 ?? #Insucesso1 ?? #Insucesso2 ?? #Insucesso3 ?? #Insucesso4 ?? #Repeticao0 ?? #Repeticao1 ?? #Repeticao2 ?? #Repeticao3 ?? #Repeticao4 ?? #Sucesso0 gera_pedido.php?passo=3&cartao=0 gera_pedido_msg.php?pedido= #Sucesso1 ?? #Sucesso2 ?? #Sucesso3 ?? #Sucesso4 ??
--	--

**Figura C.9** – Definição das Páginas Acessadas para a Tarefa *Compra (forma pagamento – depósito bancário)*

## **Apêndice D - Tabelas (*Excel*) Elaboradas Com Dados Observados Em Interações Para Apoiar a Determinação dos Parâmetros Admissíveis**

Este Apêndice apresenta os resultados obtidos após diversas análises de *web sites* específicos para comércio eletrônico e que podem ser acessados por qualquer usuário, visto que já estão implementados e passíveis de utilização. A finalidade dessas análises é a determinação de alguns parâmetros admissíveis de interações com *web sites* de comércio eletrônico para uma posterior comparação com os resultados que podem ser obtidos através da aplicação do Ergo-Monitor.

São apresentadas as tabelas determinadas para as atividades que atualmente são passíveis de monitoramento pelo Ergo-Monitor, sendo que para cada uma dessas tarefas são apresentadas 3 tabelas distintas. A primeira refere-se aos tempos de acesso requisitados para a realização da tarefa, a segunda refere-se à quantidade de acessos à página de *help* requisitadas durante as interações que visaram cumprir a tarefa em questão, enquanto que, analogamente, a terceira refere-se às quantidades de vezes que são apresentadas páginas de mensagens de erro durante as interações.

São apresentados valores estatísticos que permitem uma análise a respeito dos dados apresentados. Esses valores são: mediana, que é o número no centro de um conjunto de números e o desvio padrão, que é uma medida do grau de dispersão dos valores em relação ao valor médio.

As atividades passíveis de monitoramento são: cadastro, *login*, busca simples, busca avançada, troca de senhas e compras.

Ainda, foram analisados os *web sites* das seguintes empresas: Fnac, Alô Bebê, Lojas Americanas, Ponto Frio, Saraiva, Tempo Real e Shop Fácil. A caracterização dos usuários é expressada da seguinte forma: (1) experiente, (2) experiente, (3) mediano, (4) mediano, (5) iniciante, (6) experiente, (7) iniciante, (8) mediano, (9) iniciante

Ressalta-se que as colunas hachuradas em tom cinza representam que, a tarefa associada não está disponível para ser realizada pelo *web site* em questão.

**Tabela D.1 – Tempo (em segundos) para Realizar a Tarefa Busca Avançada**

	Saraiva	Tempo Real	Ponto Frio	Fnac	Shop Fácil	Americanas	Alô Bebê	Média	Mediana	Desvio Padrão
usuário 1	58	74	83	73		130	72	81,67	73,50	25,00
usuário 2	75	88	78	55		105	81	80,33	79,50	16,39
usuário 3	150	62	105	88		155	84	107,33	96,50	37,61
usuário 4	115	153	72	78		140	79	106,17	97,00	34,97
usuário 5	194	175	154	127		184	91	154,17	164,50	39,09
usuário 6	68	71	147	59		112	71	88,00	71,00	34,28
usuário 7	142	127	136	104		128	87	120,67	127,50	20,95
usuário 8	179	102	104	110		137	85	119,50	107,00	33,69
usuário 9	183	123	136	125		141	107	135,83	130,50	25,94

**Tabela D.2 – Quantidade de Acessos ao Help para Realizar a Tarefa de Busca Avançada**

	Saraiva	Tempo Real	Ponto Frio	Fnac	Shop Fácil	Americanas	Alô Bebê	Média	Mediana	Desvio Padrão
usuário 1	00	01	00	00		00	00	0,17	0,00	0,41
usuário 2	01	00	00	00		00	00	0,17	0,00	0,41
usuário 3	01	00	00	01		00	00	0,33	0,00	0,52
usuário 4	00	01	00	00		00	00	0,17	0,00	0,41
usuário 5	00	01	00	01		00	00	0,33	0,00	0,52
usuário 6	00	00	00	00		00	00	0,00	0,00	0,00
usuário 7	00	00	00	00		00	00	0,00	0,00	0,00
usuário 8	00	01	00	00		00	00	0,17	0,00	0,41
usuário 9	01	00	00	01		00	00	0,33	0,00	0,52

**Tabela D.3 – Quantidade de Mensagens de Erro Apresentadas na Tarefa de Busca Avançada**

	Saraiva	Tempo Real	Ponto Frio	Fnac	Shop Fácil	Americanas	Alô Bebê	Média	Mediana	Desvio Padrão
usuário 1	00	01	00	01		00	00	0,33	0,00	0,52
usuário 2	00	01	00	00		00	00	0,17	0,00	0,41
usuário 3	01	02	00	00		00	00	0,50	0,00	0,84
usuário 4	00	02	00	00		00	00	0,33	0,00	0,82
usuário 5	00	02	00	00		00	00	0,33	0,00	0,82
usuário 6	00	00	00	00		00	00	0,00	0,00	0,00
usuário 7	00	01	00	01		00	00	0,33	0,00	0,52
usuário 8	00	01	00	00		00	00	0,17	0,00	0,41
usuário 9	01	01	00	01		00	00	0,50	0,50	0,55

**Tabela D.4 – Tempo (em segundos) para Realizar a Tarefa *Busca Simples***

	Saraiva	Tempo Real	Ponto Frio	Fnac	Shop Fácil	Americanas	Alô Bebê	Média	Mediana	Desvio Padrão
usuário 1	03	05	03	04	32	50	11	15,43	5,00	18,43
usuário 2	04	08	04	03	28	60	09	16,57	8,00	21,01
usuário 3	04	03	08	03	37	45	14	16,29	8,00	17,47
usuário 4	04	04	02	05	21	70	10	16,57	5,00	24,43
usuário 5	04	05	04	04	50	91	15	24,71	5,00	33,67
usuário 6	03	03	04	03	31	57	12	16,14	4,00	20,71
usuário 7	03	04	05	06	24	80	18	20,00	6,00	27,65
usuário 8	03	04	04	04	26	56	10	15,29	4,00	19,72
usuário 9	03	04	04	04	33	72	13	19,00	4,00	25,73

**Tabela D.5 – Quantidade de Acessos ao *Help* para Realizar a Tarefa *Busca Simples***

	Saraiva	Tempo Real	Ponto Frio	Fnac	Shop Fácil	Americanas	Alô Bebê	Média	Mediana	Desvio Padrão
usuário 1	00	00	00	00	00	00	00	0,00	0,00	0,00
usuário 2	00	00	00	00	00	00	00	0,00	0,00	0,00
usuário 3	00	00	00	00	00	00	00	0,00	0,00	0,00
usuário 4	00	00	00	00	00	00	00	0,00	0,00	0,00
usuário 5	00	00	00	00	00	00	00	0,00	0,00	0,00
usuário 6	00	00	00	00	00	00	00	0,00	0,00	0,00
usuário 7	00	00	00	00	00	00	00	0,00	0,00	0,00
usuário 8	00	00	00	00	00	00	00	0,00	0,00	0,00
usuário 9	00	00	00	00	00	00	00	0,00	0,00	0,00

**Tabela D.6 – Quantidade de Mensagens de Erro Apresentadas na Tarefa *Busca Simples***

	Saraiva	Tempo Real	Ponto Frio	Fnac	Shop Fácil	Americanas	Alô Bebê	Média	Mediana	Desvio Padrão
usuário 1	00	00	00	00	00	00	00	0,00	0,00	0,00
usuário 2	00	00	00	00	00	00	00	0,00	0,00	0,00
usuário 3	00	00	00	00	00	00	00	0,00	0,00	0,00
usuário 4	00	00	00	00	00	00	00	0,00	0,00	0,00
usuário 5	00	00	00	00	00	00	00	0,00	0,00	0,00
usuário 6	00	00	00	00	00	00	00	0,00	0,00	0,00
usuário 7	00	00	00	00	00	00	00	0,00	0,00	0,00
usuário 8	00	00	00	00	00	00	00	0,00	0,00	0,00
usuário 9	00	00	00	00	00	00	00	0,00	0,00	0,00

**Tabela D.7 – Tempo (em segundos) para Realizar a Tarefa Cadastro**

	Saraiva	Tempo Real	Ponto Frio	Fnac	Shop Fácil	Americanas	Alô Bebê	Média	Mediana	Desvio Padrão
usuário 1	138	194	224	137	192	192	200	182,43	192,00	32,64
usuário 2	190	252	238	148	175	155	190	192,57	190,00	39,40
usuário 3	176	176	252	221	212	255	250	220,29	221,00	34,36
usuário 4	199	222	181	172	181	225	290	210,00	199,00	40,86
usuário 5	272	273	261	245	202	307	338	271,14	272,00	43,45
usuário 6	123	165	204	153	162	226	251	183,43	165,00	45,05
usuário 7	233	207	245	205	215	290	324	245,57	233,00	45,37
usuário 8	167	173	202	135	175	163	354	195,57	173,00	72,59
usuário 9	248	199	238	189	194	295	268	233,00	238,00	40,69

**Tabela D.8 – Quantidade de Acessos ao Help para Realizar a Tarefa Cadastro**

	Saraiva	Tempo Real	Ponto Frio	Fnac	Shop Fácil	Americanas	Alô Bebê	Média	Mediana	Desvio Padrão
usuário 1	00	01	00	00	00	02	00	0,43	0,00	0,79
usuário 2	01	00	00	00	00	01	00	0,29	0,00	0,49
usuário 3	01	01	00	00	00	01	00	0,43	0,00	0,53
usuário 4	02	01	00	00	00	00	00	0,43	0,00	0,79
usuário 5	01	01	00	00	00	00	00	0,29	0,00	0,49
usuário 6	00	00	00	00	00	01	00	0,14	0,00	0,38
usuário 7	01	00	00	00	00	01	00	0,29	0,00	0,49
usuário 8	01	01	00	00	00	01	00	0,43	0,00	0,53
usuário 9	01	01	00	00	00	02	00	0,57	0,00	0,79

**Tabela D.9 – Quantidade de Mensagens de Erro Apresentadas na Tarefa Cadastro**

	Saraiva	Tempo Real	Ponto Frio	Fnac	Shop Fácil	Americanas	Alô Bebê	Média	Mediana	Desvio Padrão
usuário 1	02	01	02	02	02	02	01	1,71	2,00	0,49
usuário 2	03	01	01	02	01	02	01	1,57	1,00	0,79
usuário 3	02	01	03	03	03	04	04	2,86	3,00	1,07
usuário 4	02	01	02	01	01	01	02	1,43	1,00	0,53
usuário 5	01	02	02	03	03	03	03	2,43	3,00	0,79
usuário 6	01	01	01	01	01	01	01	1,00	1,00	0,00
usuário 7	02	02	02	02	02	02	02	2,00	2,00	0,00
usuário 8	01	01	01	01	01	01	01	1,00	1,00	0,00
usuário 9	02	01	02	02	01	01	01	1,43	1,00	0,53

**Tabela D.10 – Tempo (em segundos) para Realizar a Tarefa *Compra***

	Saraiva	Tempo Real	Ponto Frio	Fnac	Shop Fácil	Americanas	Alô Bebê	Média	Mediana	Desvio Padrão
usuário 1	190	195	255	70	390	70	83	179,00	190,00	118,08
usuário 2	173	188	218	60	370	55	76	162,86	173,00	112,87
usuário 3	224	138	198	90	435	105	93	183,29	138,00	122,66
usuário 4	208	164	264	110	405	120	146	202,43	164,00	103,97
usuário 5	257	238	277	129	491	187	166	249,29	238,00	118,79
usuário 6	168	171	199	68	378	63	97	163,43	168,00	108,77
usuário 7	273	255	255	109	390	124	175	225,86	255,00	97,88
usuário 8	208	204	235	81	420	98	80	189,43	204,00	121,03
usuário 9	251	189	271	107	486	171	102	225,29	189,00	131,77

**Tabela D.11 – Quantidade de Acessos ao *Help* para Realizar a Tarefa *Compra***

	Saraiva	Tempo Real	Ponto Frio	Fnac	Shop Fácil	Americanas	Alô Bebê	Média	Mediana	Desvio Padrão
usuário 1	01	00	00	00	00	00	00	0,14	0,00	0,38
usuário 2	01	00	00	00	00	00	00	0,14	0,00	0,38
usuário 3	01	00	00	00	00	00	00	0,14	0,00	0,38
usuário 4	01	00	00	00	00	00	00	0,14	0,00	0,38
usuário 5	02	00	00	00	00	00	00	0,29	0,00	0,76
usuário 6	01	00	00	00	00	00	00	0,14	0,00	0,38
usuário 7	01	00	00	00	00	00	00	0,14	0,00	0,38
usuário 8	01	00	00	00	00	00	00	0,14	0,00	0,38
usuário 9	01	00	00	00	00	00	00	0,14	0,00	0,38

**Tabela D.12 – Quantidade de Mensagens de Erro Apresentadas na Tarefa *Compra***

	Saraiva	Tempo Real	Ponto Frio	Fnac	Shop Fácil	Americanas	Alô Bebê	Média	Mediana	Desvio Padrão
usuário 1	01	03	03	01	02	01	00	1,57	1,00	1,13
usuário 2	02	02	03	00	01	00	00	1,14	1,00	1,21
usuário 3	02	01	02	01	04	01	00	1,57	1,00	1,27
usuário 4	01	03	02	01	01	02	01	1,57	1,00	0,79
usuário 5	03	03	04	01	03	01	01	2,29	3,00	1,25
usuário 6	01	01	01	00	01	00	00	0,57	1,00	0,53
usuário 7	02	02	02	01	02	01	01	1,57	2,00	0,53
usuário 8	01	01	01	00	01	00	01	0,71	1,00	0,49
usuário 9	02	01	02	00	02	01	00	1,14	1,00	0,90

**Tabela D.13 – Tempo (em segundos) para Realizar a Tarefa *Fale Conosco***

	Saraiva	Tempo Real	Ponto Frio	Fnac	Shop Fácil	Americanas	Alô Bebê	Média	Mediana	Desvio Padrão
usuário 1	54				50	135	60	42,71	57,00	40,38
usuário 2	43				30	125	40	34,00	41,50	44,02
usuário 3	47				72	105	50	39,14	61,00	26,76
usuário 4	62				64	140	70	48,00	67,00	37,49
usuário 5	97				101	179	109	69,43	105,00	38,66
usuário 6	48				36	92	64	34,29	56,00	24,22
usuário 7	73				80	130	58	48,71	76,50	31,21
usuário 8	82				62	122	52	45,43	72,00	30,96
usuário 9	91				73	120	69	50,43	82,00	23,23

**Tabela D.14 – Quantidade de Acessos ao *Help* para Realizar a Tarefa *Fale Conosco***

	Saraiva	Tempo Real	Ponto Frio	Fnac	Shop Fácil	Americanas	Alô Bebê	Média	Mediana	Desvio Padrão
usuário 1	00				00	00	00	0,00	0,00	0,00
usuário 2	00				00	00	00	0,00	0,00	0,00
usuário 3	00				00	00	00	0,00	0,00	0,00
usuário 4	00				00	00	00	0,00	0,00	0,00
usuário 5	00				00	00	00	0,00	0,00	0,00
usuário 6	00				00	00	00	0,00	0,00	0,00
usuário 7	00				00	00	00	0,00	0,00	0,00
usuário 8	00				00	00	00	0,00	0,00	0,00
usuário 9	00				00	00	00	0,00	0,00	0,00

**Tabela D.15 – Quantidade de Mensagens de Erro Apresentadas na Tarefa *Fale Conosco***

	Saraiva	Tempo Real	Ponto Frio	Fnac	Shop Fácil	Americanas	Alô Bebê	Média	Mediana	Desvio Padrão
usuário 1	00				00	02	00	0,29	0,00	1,00
usuário 2	00				00	01	00	0,14	0,00	0,50
usuário 3	01				00	00	00	0,14	0,00	0,50
usuário 4	01				00	02	01	0,57	1,00	0,82
usuário 5	02				00	02	00	0,57	1,00	1,15
usuário 6	00				00	00	00	0,00	0,00	0,00
usuário 7	00				00	01	00	0,14	0,00	0,50
usuário 8	00				00	00	00	0,00	0,00	0,00
usuário 9	01				00	00	01	0,29	0,50	0,58

**Tabela D.16 – Tempo (em segundos) para Realizar a Tarefa Login**

	Saraiva	Tempo Real	Ponto Frio	Fnac	Shop Fácil	Americanas	Alô Bebê	Média	Mediana	Desvio Padrão
usuário 1	03				04	11	05	5,75	4,50	3,59
usuário 2	03				03	13	05	6,00	4,00	4,76
usuário 3	03				04	15	25	11,75	9,50	10,37
usuário 4	16				03	20	05	11,00	10,50	8,29
usuário 5	05				04	17	07	8,25	6,00	5,97
usuário 6	03				03	13	05	6,00	4,00	4,76
usuário 7	05				03	09	05	5,50	5,00	2,52
usuário 8	03				04	10	06	5,75	5,00	3,10
usuário 9	03				03	08	05	4,75	4,00	2,36

**Tabela D.17 – Quantidade de Acessos ao Help para Realizar a Tarefa Login**

	Saraiva	Tempo Real	Ponto Frio	Fnac	Shop Fácil	Americanas	Alô Bebê	Média	Mediana	Desvio Padrão
usuário 1	00				00	00	00	0,00	0,00	0,00
usuário 2	00				00	00	00	0,00	0,00	0,00
usuário 3	00				00	00	00	0,00	0,00	0,00
usuário 4	00				00	00	00	0,00	0,00	0,00
usuário 5	00				00	00	00	0,00	0,00	0,00
usuário 6	00				00	00	00	0,00	0,00	0,00
usuário 7	00				00	00	00	0,00	0,00	0,00
usuário 8	00				00	00	00	0,00	0,00	0,00
usuário 9	00				00	00	00	0,00	0,00	0,00

**Tabela D.18 – Quantidade de Mensagens de Erro Apresentadas na Tarefa Login**

	Saraiva	Tempo Real	Ponto Frio	Fnac	Shop Fácil	Americanas	Alô Bebê	Média	Mediana	Desvio Padrão
usuário 1	00				00	00	00	0,00	0,00	0,00
usuário 2	00				00	00	00	0,00	0,00	0,00
usuário 3	00				00	00	01	0,25	0,00	0,50
usuário 4	01				00	00	00	0,25	0,00	0,50
usuário 5	00				00	00	00	0,00	0,00	0,00
usuário 6	00				00	00	00	0,00	0,00	0,00
usuário 7	00				00	00	00	0,00	0,00	0,00
usuário 8	00				00	00	00	0,00	0,00	0,00
usuário 9	00				00	00	00	0,00	0,00	0,00

**Tabela D.19 – Tempo (em segundos) para Realizar a Tarefa Troca Senha**

	Saraiva	Tempo Real	Ponto Frio	Fnac	Shop Fácil	Americanas	Alô Bebê	Média	Mediana	Desvio Padrão
usuário 1			85					85,00	85,00	
usuário 2			94					94,00	94,00	
usuário 3			72					72,00	72,00	
usuário 4			102					102,00	102,00	
usuário 5			121					121,00	121,00	
usuário 6			152					152,00	152,00	
usuário 7			132					132,00	132,00	
usuário 8			162					162,00	162,00	
usuário 9			253					253,00	253,00	

**Tabela D.20 – Quantidade de Acessos ao Help para Realizar a Tarefa Troca Senha**

	Saraiva	Tempo Real	Ponto Frio	Fnac	Shop Fácil	Americanas	Alô Bebê	Média	Mediana	Desvio Padrão
usuário 1			00					0,00	0,00	
usuário 2			00					0,00	0,00	
usuário 3			00					0,00	0,00	
usuário 4			00					0,00	0,00	
usuário 5			00					0,00	0,00	
usuário 6			00					0,00	0,00	
usuário 7			00					0,00	0,00	
usuário 8			00					0,00	0,00	
usuário 9			00					0,00	0,00	

**Tabela D.21 – Quantidade de Mensagens de Erro Apresentadas na Tarefa Troca Senha**

	Saraiva	Tempo Real	Ponto Frio	Fnac	Shop Fácil	Americanas	Alô Bebê	Média	Mediana	Desvio Padrão
usuário 1			00					0,00	0,00	
usuário 2			00					0,00	0,00	
usuário 3			00					0,00	0,00	
usuário 4			01					1,00	1,00	
usuário 5			01					1,00	1,00	
usuário 6			00					0,00	0,00	
usuário 7			00					0,00	0,00	
usuário 8			00					0,00	0,00	
usuário 9			01					1,00	1,00	

## **Apêndice E - Resultados Obtidos para o Monitoramento de Tarefas Empregando-se o Log Control**

Neste Apêndice são listadas os arquivos que foram gerados durante o monitoramento automatizado de tarefas específicas a *web sites* de comércio eletrônico. Essas tarefas são:

- busca simples;
- busca avançada;
- compras - dividida em outras tarefas como ir para o caixa, definição do usuário comprador (login) e formas de pagamento: cartão ou depósito);
- fale conosco; e
- troca de senha;

As Figuras E.1 a E.8 apresentam os arquivos gerados para cada uma dessas tarefas monitoradas. Vale ressaltar que os resultados obtidos e apresentados nessas tabelas foram bastante próximos dos resultados obtidos através da análise das fitas com as gravações dos ensaios de interação que foram realizados para a determinação dos arquivos de *log* que foram utilizados como entrada para tais monitoramentos. Tais comparações visaram a validação dos programas do Log Control como um dos componentes do ambiente Ergo-Monitor. Sabe-se, porém, que novos testes devem ser conduzidos para que a eficiência desses programas seja melhor analisada.

Especificamente, a análise da tarefa de compras requer atenção especial pois a mesma foi dividida em sub-tarefas. Tal divisão se deve ao fato de que não foi possível a definição de todas as seqüências de páginas a serem monitoradas analisando-se toda a atividade de compra como uma atividade única, uma vez que o rol de possibilidades seria muito extenso e, portanto, muito difícil de ser gerado. Optou-se pela divisão dessa tarefa em sub-tarefas, uma vez que pode-se, assim, monitor tais sub-tarefas de forma individual e os resultados obtidos devem ser analisados segundo as heurísticas dos avaliadores (que são os usuários do Ergo-Monitor).

TAREFA: cadastro			
TIPO	TEMPO	QUANTIDADE	
Acesso 330	6		
Help 0	0		
Mensagem Erro	36	1	
Abandono0	0	0	
Abandono1	0	0	
Abandono2	0	0	
Abandono3	0	0	
Abandono4	0	0	
Soma_Abandono			=soma(B7:B11) =soma(C7:C11)
CancErro0	0	0	
CancErro1	50	1	
CancErro2	0	0	
CancErro3	0	0	
CancErro4	0	0	
Soma_CancErro			=soma(B13:B17) =soma(C13:C17)
Desistência0	30	2	
Desistência1	0	0	
Desistência2	0	0	
Desistência3	0	0	
Desistência4	0	0	
Soma_Desistência			=soma(B19:B23) =soma(C19:C23)
Insucesso0	0	0	
Insucesso1	0	0	
Insucesso2	0	0	
Insucesso3	0	0	
Insucesso4	0	0	
Soma_Insucesso			=soma(B25:B29) =soma(C25:C29)
Repetição0	0	0	
Repetição1	150	1	
Repetição2	0	0	
Repetição3	0	0	
Repetição4	0	0	
Soma_Repetição			=soma(B31:B35) =soma(C31:C35)
Sucesso0	0	0	
Sucesso1	160	1	
Sucesso2	0	0	
Sucesso3	0	0	
Sucesso4	0	0	
Soma_Sucesso			=soma(B37:B41) =soma(C37:C41)
TAXAS	Help		=(B5/B4) =(C5/C4)
	Mensagem Erro		=(B6/B4) =(C6/C4)
	Abandono		=(E12/B4)=(F12/C4)
	CancErro		=(E17/B4)=(F17/C4)
	Canc4		=(B18/B4)=(C18/C4)
	Desistência		=(E24/B4)=(F24/C4)
	Insucesso		=(E30/B4)=(F30/C4)
	Repetição		=(E36/B4)=(F36/C4)
	Sucesso		=(E42/B4)=(F42/C4)

**Figura E.1** – Planilha Excel que Apóia a Determinação de Taxas para a Atividade de Cadastro

TAREFA: buscasimples			
TIPO	TEMPO	QUANTIDADE	
Acesso	126	5	
Help	0		
Mensagem de Erro		0	0
Abandono0	0	0	
Abandono1	0	0	
Abandono2	0	0	
Abandono3	0	0	
Abandono4	0	0	
Soma_Abandono			=soma(B7:B11) =soma(C7:C11)
CancErro0	0	0	
CancErro1	0	0	
CancErro2	0	0	
CancErro3	0	0	
CancErro4	0	0	
Soma_CancErro			=soma(B13:B17) =soma(C13:C17)
Desistencia0	0	0	
Desistência1	0	0	
Desistência2	0	0	
Desistencia3	0	0	
Desistencia4	0	0	
Soma_Desistencia			=soma(B15:B19) =soma(C15:C19)
Insucesso0	0	0	
Insucesso1	40	1	
Insucesso2	0	0	
Insucesso3	0	0	
Insucesso4	0	0	
Soma_Insucesso			=soma(B17:B21) =soma(C17:C21)
Repeticao0	0	0	
Repeticao1	0	0	
Repeticao2	0	0	
Repeticao3	0	0	
Repeticao4	0	0	
Soma_Repeticao			=soma(B19:23) =soma(C19:C23)
Sucesso0	0	0	
Sucesso1	76	2	
Sucesso2	30	1	
Sucesso3	0	0	
Sucesso4	0	0	
Soma_Sucesso			=soma(B21:B25) =soma(C21:C25)
TAXAS	Help		=soma(B5/B4) =soma(C5/C4)
	Mensagem de Erro		=soma(B6/B4) =soma(C6/C4)
	Abandono		=soma(E12/B4) =soma(F12/C4)
	CancErro	=soma(E18/B4)	=soma(F18/C4)
	Desistencia		=soma(E24/B4) =soma(F24/C4)
	Insucesso		=soma(E30/B4) =soma(F30/C4)
	Repeticao		=soma(E36/B4) =soma(F36/C4)
	Sucesso	=soma(E43/B4)	=soma(F43/C4)

**Figura E.2** – Planilha Excel que Apóia a Determinação de Taxas para a Atividade de Busca Simples

TAREFA: log-compras			
TIPO	TEMPO	QUANTIDADE	
Acesso	290	8	
Help	90	2	
Mensagem Erro		43	2
Abandono0	0	0	
Abandono1	0	0	
Abandono2	0	0	
Abandono3	0	0	
Abandono4	0	0	
Soma_Abandono			=soma(B7:B11) =soma(C7:C11)
CancErro0	0	0	
CancErro1	0	0	
CancErro2	0	0	
CancErro3	0	0	
CancErro4	0	0	
Soma_CancErro			=soma(B13:B17) =soma(C13:C17)
Desistencia0	0	0	
Desistencia1	0	0	
Desistencia2	0	0	
Desistencia3	0	0	
Desistencia4	0	0	
Soma_Desistencia			=soma(B19:B23) =soma(C19:C23)
Insucesso0	0	0	
Insucesso1	0	0	
Insucesso2	0	0	
Insucesso3	0	0	
Insucesso4	0	0	
Soma_Insucesso			=soma(B25:B29) =soma(C25:C29)
Repeticao0	0	0	
Repeticao1	0	0	
Repeticao2	0	0	
Repeticao3	0	0	
Repeticao4	0	0	
Soma_Repeticao			=soma(B31:B35) =soma(C31:C35)
Sucesso0	48	2	
Sucesso1	57	1	
Sucesso2	52	1	
Sucesso3	0	0	
Sucesso4	0	0	
Soma_Sucesso			=soma(B37:B41) =soma(C37:C41)
TAXAS	Help		=soma(B5:B4) =soma(C5/C4)
	Mensagem Erro		=soma(B6/B4) =soma(C6/C4)
	Abandono		=soma(E12/B4) =soma(F12/C4)
	CancErro	=soma(E18/B4)	=soma(F18/C4)
	Desistencia	=soma(E24/B4)	=soma(F24/C4)
	Insucesso	=soma(E30/B4)	=soma(F30/C4)
	Repeticao	=soma(E36/B4)	=soma(F36/C4)
	Sucesso	=soma(E42/B4)	=soma(F42/C4)

**Figura E.3** – Planilha Excel que Apóia a Determinação de Taxas para a Atividade de Compras – Definição do Usuário Comprador (*login*)



TAREFA: deposito			
TIPO	TEMPO	QUANTIDADE	
Acesso	157	2	
Help	0		
MensagemErro	0	0	
Abandono0	51	1	
Abandono1	0	0	
Abandono2	0	0	
Abandono3	0	0	
Abandono4	0	0	
Soma_Abandono			=soma(B7:B11) =soma(C7:C11)
CancErro0	0	0	
CancErro1	0	0	
CancErro2	0	0	
CancErro3	0	0	
CancErro4	0	0	
Soma_CancErro			=soma(B13:B17) =soma(C13:C17)
Desistencia0	0	0	
Desistencia1	0	0	
Desistencia2	0	0	
Desistencia3	0	0	
Desistencia4	0	0	
Soma_Desistencia			=soma(B19:B23) =soma(C19:C23)
Insucesso0	0	0	
Insucesso1	0	0	
Insucesso2	0	0	
Insucesso3	0	0	
Insucesso4	0	0	
Soma_Insucesso			=soma(B25:B29) =soma(C25:C29)
Repeticao0	0	0	
Repeticao1	0	0	
Repeticao2	0	0	
Repeticao3	0	0	
Repeticao4	0	0	
Soma_Repeticao			=soma(B31:B35) =soma(C31:C35)
Sucesso0	106	1	
Sucesso1	0	0	
Sucesso2	0	0	
Sucesso3	0	0	
Sucesso4	0	0	
Soma_Sucesso			=soma(B37:B41) =soma(C37:C41)
TAXAS	Help		=soma(B5/B4) =soma(C5/C4)
	MensagemErro		=soma(B6/B4) =soma(C6/C4)
	Abandono		=soma(E12/B4) =soma(F12/C4)
	CancErro	=soma(E18/B4)	=soma(F18/C4)
	Desistencia	=soma(E24/B4)	=soma(F24/C4)
	Insucesso	=soma(E30/B4)	=soma(F30/C4)
	Repeticao	=soma(E36/B4)	=soma(F36/C4)
	Sucesso	=soma(E42/B4)	=soma(F42/C4)

**Figura E.5** – Planilha Excel que Apóia a Determinação de Taxas para a Atividade de Compras – Forma de Pagamento: Depósito

TAREFA: cartao			
TIPO	TEMPO	QUANTIDADE	
Acesso	264	5	
Help	0		
MensagemErro	74	2	
Abandono0	0	0	
Abandono1	0	0	
Abandono2	0	0	
Abandono3	0	0	
Abandono4	0	0	
Soma_Abandono			=soma(B7:B11) =soma(C7:C11)
CancErro0	0	0	
CancErro1	110	1	
CancErro2	0	0	
CancErro3	0	0	
CancErro4	0	0	
Soma_CancErro			=soma(B13:B17) =soma(C13:C17)
Desistencia0	43	1	
Desistencia1	0	0	
Desistencia2	0	0	
Desistencia3	0	0	
Desistencia4	0	0	
Soma_Desistencia			=soma(B19:B23) =soma(C19:C23)
Insucesso0	0	0	
Insucesso1	0	0	
Insucesso2	0	0	
Insucesso3	0	0	
Insucesso4	0	0	
Soma_Insucesso			=soma(B25:B29) =soma(C25:C29)
Repeticao0	0	0	
Repeticao1	0	0	
Repeticao2	0	0	
Repeticao3	0	0	
Repeticao4	0	0	
Soma_Repeticao			=soma(B31:B35) =soma(C31:C35)
Sucesso0	67	1	
Sucesso1	0	0	
Sucesso2	0	0	
Sucesso3	0	0	
Sucesso4	0	0	
Soma_Sucesso			=soma(B37:B41) =soma(C37:C41)
TAXAS	Help		=(B5/B4) =(C5/C4)
	MensagemErro		=(B6/B4) =(C6/C4)
	Abandono		=(E12/B4) =(F12/C4)
	CancErro	=(E18/B4)	=(F18/C4)
	Desistencia		=(E24/B4) =(F24/C4)
	Insucesso		=(E30/B4) =(F30/C4)
	Repeticao		=(E36/B4) =(F36/C4)
	Sucesso	=(E42/B4)	=(F42/C4)

**Figura E.6** – Planilha Excel que Apóia a Determinação de Taxas para a Atividade de Compras – Forma de Pagamento: Cartão de Crédito

TAREFA: faleconosco			
TIPO	TEMPO	QUANTIDADE	
Acessos	34	1	
Help	0		
MensagemErro	0	0	
Abandono0	0	0	
Abandono1	0	0	
Abandono2	0	0	
Abandono3	0	0	
Abandono4	0	0	
Soma_Abandono			=soma(B7:B11) =soma(C7:C11)
CancErro0	0	0	
CancErro1	0	0	
CancErro2	0	0	
CancErro3	0	0	
CancErro4	0	0	
Soma_CancErro			=soma(B13:B17) =soma(C13:C17)
Desistencia0	0	0	
Desistencia1	0	0	
Desistencia2	0	0	
Desistencia3	0	0	
Desistencia4	0	0	
Soma_Desistencia			=soma(B19:B23) =soma(C19:C23)
Insucesso0	0	0	
Insucesso1	0	0	
Insucesso2	0	0	
Insucesso3	0	0	
Insucesso4	0	0	
Soma_Insucesso			=soma(B25:B29) =soma(C25:C29)
Repeticao0	0	0	
Repeticao1	0	0	
Repeticao2	0	0	
Repeticao3	0	0	
Repeticao4	0	0	
Soma_Repeticao			=soma(B31:B35) =soma(C31:C35)
Sucesso0	34	1	
Sucesso1	0	0	
Sucesso2	0	0	
Sucesso3	0	0	
Sucesso4	0	0	
Soma_Sucesso			=soma(B37:B41) =soma(C37:C41)
TAXAS	Help		=(B5/B4) =(C5/C4)
	MensagemErro		=(B6/B4) =(C6/C4)
	Abandono		=(E12/B4) =(F12/C4)
	CancErro	=(E18/B4)	=(F18/C4)
	Desistencia		=(E24/B4) =(F24/C4)
	Insucesso		=(E30/B4) =(F30/C4)
	Repeticao		=(E36/B4) =(F36/C4)
	Sucesso	=(E42/B4)	=(F42/C4)

**Figura E.7** – Planilha Excel que Apóia a Determinação de Taxas para a Atividade de Compras – Fale Conosco

TAREFA: trocarsenha			
TIPO	TEMPO	QUANTIDADE	
Acessos	29	1	
Help	0		
MensagemErro	0	0	
Abandono0	0	0	
Abandono1	0	0	
Abandono2	0	0	
Abandono3	0	0	
Abandono4	0	0	
Soma_Abandono			=soma(B7:B11) =soma(C7:C11)
CancErro0	0	0	
CancErro1	0	0	
CancErro2	0	0	
CancErro3	0	0	
CancErro4	0	0	
Soma_CancErro			=soma(B13:B17) =soma(C13:C17)
Desistencia0	0	0	
Desistencia1	0	0	
Desistencia2	0	0	
Desistencia3	0	0	
Desistencia4	0	0	
Soma_Desistencia			=soma(B19:B23) =soma(C19:C23)
Insucesso0	0	0	
Insucesso1	0	0	
Insucesso2	0	0	
Insucesso3	0	0	
Insucesso4	0	0	
Soma_Insucesso			=soma(B25:B29) =soma(C25:C29)
Repeticao0	0	0	
Repeticao1	0	0	
Repeticao2	0	0	
Repeticao3	0	0	
Repeticao4	0	0	
Soma_Repeticao			=soma(B31:B35) =soma(C31:C35)
Sucesso0	29	1	
Sucesso1	0	0	
Sucesso2	0	0	
Sucesso3	0	0	
Sucesso4	0	0	
Soma_Sucesso			=soma(B37:B41) =soma(C37:C41)
TAXAS	Help		=(B5/B4) =(C5/C4)
	MensagemErro		=(B6/B4) =(C6/C4)
	Abandono		=(E12/B4) =(F12/C4)
	CancErro	=(E18/B4)	=(F18/C4)
	Desistencia		=(E24/B4) =(F24/C4)
	Insucesso		=(E30/B4) =(F30/C4)
	Repeticao		=(E36/B4) =(F36/C4)
	Sucesso	=(E42/B4)	=(F42/C4)

**Figura E.8** – Planilha Excel que Apóia a Determinação de Taxas para a Atividade de Compras – Troca de Senha