

PONTIFÍCIA UNIVERSIDADE CATÓLICA DE MINAS GERAIS
Bacharelado em Sistemas de Informação

**Construção de um sistema de *Blackboard* para
gestão de documentos usando XML**

Lucas Gonçalves de Oliveira

Arcos
2004

Lucas Gonçalves de Oliveira

Construção de um sistema de *Blackboard* para gestão de documentos usando XML

Trabalho de Conclusão de Curso
apresentado ao Curso de Sistemas
de Informação da Pontifícia
Universidade Católica de Minas
Gerais

Orientadora: Elza Marisa Paiva de
Figueiredo
Orientador de Conteúdo: Hermes
Pimenta de Moraes Júnior

Arcos
2004

AGRADECIMENTOS

Agradeço a DEUS e a todas as pessoas que, direta ou indiretamente, me ajudaram a vencer todas as batalhas que foram impostas à minha formação. Agradeço em especial, a minha família e amigos. Agradeço ainda, a todos os professores que passaram por minha vida nesta instituição, agradecendo especialmente a Professora Elza Marisa e o Professor Hermes Moraes, os quais tanto me ajudaram no desenvolvimento deste trabalho.

RESUMO

Com o aumento da utilização de recursos tecnológicos, há uma crescente busca por mecanismos que garantam maior proveito das redes de computadores. Um destes mecanismos é o trabalho em grupo, que permite maior comunicação entre pessoas. Como exemplo, é possível citar os fóruns digitais, onde várias pessoas se comunicam, independentemente de sua localização geográfica.

Além disso, procura-se cada vez mais a padronização, como princípio básico para o crescimento de qualquer organização, seja em processos de produção, ou administrativos trazendo para a empresa melhor desempenho e maior organização.

Neste trabalho é proposto realizar a padronização e a evolução na comunicação, já que os documentos padronizados serão apresentados virtualmente para os alunos, podendo esta apresentação ser considerada como comunicação entre os gestores dos documentos e os alunos. Esta padronização só poderá ser feita por um determinado grupo de pessoas e a comunicação se estenderá por todos os alunos da instituição que estejam interessados nos documentos.

Desta forma, esta pesquisa estabelece um ponto aceitável de padronização e aumenta a evolução na apresentação da informação, gerando assim uma melhoria na comunicação.

LISTA DE FIGURAS

FIGURA 2.1 – Documento XML válido.....	36
FIGURA 2.2 – Documento XML inválido.....	36
FIGURA 2.3 – Processamento de estilos	37
FIGURA 2.4 – XSL com diferentes finalidades	38
FIGURA 3.1 – Quadro de avisos	47
FIGURA 3.2 – Fórum do <i>Learnloop</i>	49
FIGURA 3.3 – Arquitetura <i>blackboard</i>	50
FIGURA 4.1 – Arquitetura do sistema	52
FIGURA 5.1 – Estrutura do documento XML usado.....	65
FIGURA 5.2 – <i>Blackboard</i> do sistema	68
FIGURA 5.3 – Sistema gerente do <i>blackboard</i>	69
FIGURA 5.4 – Autenticação dos usuários.....	70
FIGURA 5.5 – Criação de novos documentos de palestra.....	71
FIGURA 5.6 – Documento de palestra.....	72
FIGURA 5.7 – Remoção de avisos do <i>blackboard</i>	73
FIGURA 5.8 – Inclusão de novos usuários.....	74
FIGURA 5.9 – Exclusão de usuários	74
FIGURA 5.10 – Alteração de usuários	74

LISTA DE ABREVIATURAS

ANSI – *American National Standards Institute* (Instituto Nacional Americano de Padrões)
ASCII – *American Standard Code* (Código Americano Padrão)
CSS – *Cascading Style Sheet* (Folhas de Estilos em Cascata)
DSSSL – *Document Style Semantics and Specification Language* (Linguagem de Especificação e Semântica de Estilos de Documentos)
DTD – *Document Type Definitions* (Definição de Tipo de Documento)
EAD – Ensino a Distância
GML – *Generalized Markup Language* (Linguagem de Marcação Generalizada)
HTML – *HyperText Markup Language* (Linguagem de Marcação de Hipertexto)
IAD – Inteligência Artificial Distribuída
IBM – International Business Machines
ISO – *Internacional Organization for Standardization* (Organização Internacional de Padronização)
PHP – *PHP Hypertext Preprocessor* (Préprocessador de Hipertexto PHP)
RTF – *Rich Text Format* (Formato Rico de Texto)
SGBD – Sistema Gerenciador de Banco de Dados
SGML – *Standard Generalized Markup Language* (Linguagem de Marcação Generalizada Padrão)
TEI – *Text Encoding Initiative* (Iniciativa de codificação de Texto)
TeX – *Technical* (Linguagem para Preparação de Documentos)
URL – *Universal Resource Locator* (Localização Universal do Recurso)
W3C – *World Wide Web Consortium*
WAP – *Wireless Application Protocol* (Protocolo de Aplicação sem Fio)
XHTML – *eXtensible HyperText Markup Language* (Linguagem de Marcação de Hipertexto Extensível)
XML – *eXtensible Markup Language* (Linguagem de Marcação Extensível)
XSL – *eXtensible Style Language* (Linguagem de Estilo Extensível)
Xlink – *eXtensible Link* (Link Extensível)

SUMÁRIO

1. INTRODUÇÃO.....	8
1. 1. Justificativa	9
1. 2. Contextualização de temas pertinentes	11
1. 3. Objetivos	13
1. 3.1. <u>Objetivo geral</u>	13
1. 3.2. <u>Objetivos específicos</u>	13
1. 4. Metodologia Utilizada	14
1. 5. Resultados esperados	15
1. 6. Estrutura do trabalho	15
2. XML	17
2.1. Introdução	17
2.2. Linguagem de marcação	18
2.3. Linguagens de marcação utilizadas na Internet	20
2. 3.1. <u>HTML</u>	20
2. 3.2. <u>O surgimento da XML</u>	22
2. 3.3. <u>XML</u>	25
2. 3.4. <u>Relações entre HTML e XML</u>	28
2. 3.5. <u>SGML</u>	31
2. 3.6. <u>XHTML</u>	33
2. 3.7. <u>A XML nos navegadores</u>	35
2.4. Folhas de estilo	36
3. BLACKBOARD.....	40
3.1. Introdução	40
3.2. Trabalho cooperativo	42
3. 2.1. <u>Comunicação</u>	43
3. 2.2. <u>Percepção</u>	43
3. 2.3. <u>Coordenação</u>	44
3. 2.4. <u>Negociação</u>	44
3. 2.5. <u>Co-realização e Compartilhamento</u>	45
3.3. <u>Utilização do blackboard</u>	46
3. 3.1. <u>Blackboard como quadro de avisos</u>	46
3. 3.2. <u>Blackboard auxiliando o ensino a distância</u>	47
3. 3.3. <u>Blackboard na inteligência artificial distribuída</u>	49
3.4. <u>Blackboard e este trabalho</u>	51
4. ARQUITETURA DO SISTEMA.....	52
4.1. Usuários	52
4.2. <i>Blackboard</i>	53
4.3. Geradores dos documentos	55
4.4. Segurança	56
4.5. Sistema de geração dos documentos	58
4.6. Base de dados	59

5. SISTEMA DE PADRONIZAÇÃO DOS DOCUMENTOS	62
5.1. Linguagens utilizadas.....	62
5. 1.1. <u>HTML</u>	62
5. 1.2. <u>PHP</u>	63
5. 1.3. <u>XML</u>	65
5. 1.4. <u>XSL</u>	66
5.2 Ferramentas utilizadas.....	67
5. 2.1. <u>MySQL</u>	67
5.3. Funcionalidades do sistema	68
5. 3.1. <u>Blackboard</u>	68
5. 3.2. <u>Sistema gerente do blackboard</u>	69
5. 3.3. <u>Criação de novos documentos</u>	70
5. 3.4. <u>Remoção de avisos do blackboard</u>	72
5. 3.5. <u>Controle de usuários</u>	73
6. CONCLUSÕES E TRABALHOS FUTUROS	75
REFERÊNCIAS.....	78

1. INTRODUÇÃO

No Curso de Sistemas de Informação da Pontifícia Universidade Católica de Minas Gerais, situada na cidade de Arcos, existem várias coordenações diferentes, as quais realizam trabalhos pertinentes ao funcionamento do Curso. Dentre elas, é possível citar a coordenação do curso, a coordenação de estágios, a coordenação de atividades complementares e a coordenação de extensão.

As coordenações citadas acima produzem documentos como declarações, editais, resultados de seleções e convocações. Os editais geralmente são para divulgação de inscrições em algum evento novo. Os resultados anunciam os inscritos selecionados em alguma atividade. A convocação intima os alunos em geral para realização de alguma tarefa.

Os documentos criados pelos setores mencionados são feitos de forma padronizada. Contudo, a forma de padronização é constituída na hora em que o documento está sendo gerado. Tais documentos, hoje, são gerados em programas editores de texto, no caso descrito, o Microsoft Word. Porém, o mercado atual apresenta várias soluções para automatização de padronização. Hoje em dia, não faz sentido usar normas de padronização não automatizadas, já que a instituição referida possui ferramentas e conhecimentos para a padronização de forma automatizada.

Uma outra forma de se avaliar a praticidade e a validade desta padronização é a facilidade gerada na gerência de documentos. Na atual situação em que se encontra a instituição, a tendência é de aumentar a quantidade de documentos confeccionados e utilizados para algum fim. Com a padronização, um documento poderá ser criado com uma facilidade maior, o que economizará tempo das pessoas que os criam (CARUSO, 1999).

Para resolver os problemas citados acima, estão sendo propostas uma forma de padronização e representação diferente e mais evoluída tecnologicamente. Trata-se de um *blackboard* digital, ou seja, um mural eletrônico onde todos os documentos serão apresentados aos alunos e será tratado neste trabalho no Capítulo 3. Esta apresentação ocorre de forma virtual, de modo que os documentos fiquem disponibilizados em uma página da Internet e os alunos possam verificar os avisos, editais, seleções e outros documentos criados a todo instante. Para criar os documentos, a fim de poderem ser apresentados neste mural, haverá um sistema de interface com os professores responsáveis pelos setores especificados que construirá os documentos a partir de dados obtidos junto a estes professores.

Sendo assim, o trabalho a ser realizado será o de padronização de documentos da instituição referida, utilizando-se da linguagem XML (*eXtensible Markup Language* – Linguagem de Marcação Extensível) no processo de desenvolvimento. Este trabalho conduzirá-se de forma escrita e de forma implementada.

Os documentos criados estarão aptos a serem apresentados para os alunos do curso? O tipo de sistema a ser construído tem viabilidade de utilização pelos setores apresentados? Após o término deste, é esperado que todas essas respostas sejam obtidas de forma prática e positiva com a utilização do sistema.

1.1. Justificativa

A escolha deste tema para a pesquisa proposta baseia-se no fato de ainda não existir este tipo de ferramenta na instituição. Esta pesquisa é de extrema importância, porque, com a realização deste trabalho, haverá um acúmulo de conhecimentos em áreas consideradas

promissoras para um futuro bem próximo. Além disso, a coordenação poderá preparar documentos padronizados, independentemente das suas finalidades.

A pesquisa também assume um caráter de grande importância, tendo em vista que a maior parte das grandes empresas presentes no mundo realiza projetos de padronização desde a produção até os serviços mais técnicos. Não se pode esquecer dos trabalhos realizados por essas grandes empresas que garantem um processo de padronização reconhecido internacionalmente, lembrando que esses processos não avaliam apenas a produção da empresa e, sim, ela como um todo (BRANDÃO, 2004). Então, essa instituição não pode ficar distante das grandes intenções do mercado.

Existem várias formas de se representar os documentos criados. A forma de representação, que é utilizada hoje pelos criadores dos documentos, é de forma impressa. Estes documentos são impressos e são apresentados para as turmas por meio de um mural presente em cada sala do curso.

Contudo, esta forma que está sendo utilizada pode ser considerada não tão adequada, levando-se em consideração as formas encontradas hoje, em termos tecnológicos. O problema maior desse tipo de apresentação é que os documentos, que estão presentes no mural, geralmente excedem a data do evento. Além disso, muitas vezes, avisos que não teriam de ser obrigatoriamente lançados em sala de aula são impressos e disponibilizados no mural, pois esta é a única forma de apresentação dos documentos, gerando assim uma impressão sem necessidade, o que conseqüentemente gera custos desnecessários.

Um outro motivo para a realização deste trabalho é a necessidade de documentos padronizados, em todas as áreas atuais. Como todas as empresas da atualidade, todos têm de buscar um processo de produção de serviço que atinja uma melhor qualidade. Uma forma de se alcançar tal melhora é a padronização de processos realizados em uma empresa, ou, como neste caso, na Instituição. Por meio desta padronização será possível se entender melhor o

sentido dos documentos, já que todos eles terão uma estrutura semelhante (BRANDÃO, 2004).

Outra vantagem encontrada após a padronização dos documentos é o aumento da facilidade na hora da criação de tais documentos. Isso ocorre porque o processo de construção deverá seguir uma forma estruturada e padronizada de confecção, o que tornará mais efetivo a produção desses documentos (CARUSO, 1999).

Este trabalho ainda poderá, futuramente, ser agregado a um outro trabalho em desenvolvimento na mesma instituição e no mesmo período, por Diego Zuquim sob orientação do Professor Marco Rodrigo (ZUQUIM, COSTA, 2004). Esta agregação tem por finalidade chegar a um sistema completo de padronização de documentos administrativos e acadêmicos.

1.2. Contextualização de temas pertinentes

Há um trabalho desenvolvido por um aluno do Curso de Ciência da Computação da UNIVALI (Universidade do Vale do Itajaí – SC), coordenada pelo Centro de Ciências Tecnológicas da Terra e do Mar. Este aluno pretende desenvolver um editor de artigo, utilizando a tecnologia XML. Tal trabalho torna-se interessante e assume um caráter semelhante ao desta pesquisa, já que esses artigos que ele pretende padronizar, denominado por ele de periódicos, são de interesse da instituição da qual ele participa. O autor acredita que da forma que os periódicos são desenvolvidos, a padronização se encontra de forma distante daquela que pretende ser alcançada. Considera este problema de falta de padronização provindo da falta de material, interpretação errada dos fatos ou até mesmo falta de domínio

dos autores dos periódicos. Contudo esse trabalho está sendo desenvolvido com o intuito de facilitar a criação de documentos que já são criados padronizados, o que não é o caso deste.

Outro trabalho que também foi verificado é um fomentado pelo CNPq (Centro Nacional de Pesquisas) juntamente com as universidades UFSC, UNICAMP, UFRJ, USP, UFRGS, UFBA e UFRN, com o intuito de padronizar um currículo utilizado para verificações de pós-graduação, denominado Currículo Lattes. Tal trabalho é muito interessante, já que o âmbito da pesquisa, que é a padronização, não se limita em território nacional. O padrão criado será utilizado por mais países pan-americanos. Este trabalho gera grande facilidade, considerando a facilidade a qual este currículo padronizado traz para as instituições as quais avaliam-nos, já que, uma vez padronizados, será mais fácil e eficiente classificar e qualificar tais currículos.

Um outro trabalho analisado vem da Universidade Estadual de Campinas, onde os realizadores do trabalho pretendem padronizar o ambiente colaborativo de ensino e aprendizagem. Os alunos responsáveis por este trabalho acreditam que a utilização de XML para padronizar documentos de educação poderia permitir a identificação de campos dos tipos: "título", "palavras-chave" e "links", facilitando assim a busca de documentos por conteúdo e por estrutura.

Porém, o trabalho que mais interessa e estimula é um que está sendo desenvolvido por uma aluna da USP, no Instituto do Coração do Hospital das Clínicas da Faculdade de Medicina. Tal trabalho pretende transformar documentos criados em texto livre para documentos padronizados, usando a linguagem XML e folhas de rosto XSL (*eXtensible Style Language* – Linguagem de estilo extensível), assim como este trabalho.

Vale ressaltar que trabalhos que envolvem as mesmas ferramentas e linguagens que aqui estão sendo utilizadas, ainda não foram realizados. Além disso, a padronização e uma melhor apresentação de documentos se torna cada vez mais imperativo e realizar essa

padronização e melhorar a apresentação desses documentos de tão grande importância acadêmica é algo extremamente relevante. Por estes motivos, este trabalho se destaca de todos os demonstrados anteriormente, já que apesar de todos possuírem um objetivo semelhante, que é a padronização de documentos, este utiliza uma apresentação dos documentos mais evoluída.

1.3. Objetivos

1.3.1. Objetivo geral

O objetivo geral deste trabalho é a padronização automatizada dos documentos criados pelo Curso de Sistemas de Informação, da PUC Minas Arcos. Espera-se que estes documentos sejam construídos com alto índice de padronização e conseqüentemente, qualidade.

1.3.2. Objetivos específicos

Têm-se como objetivos específicos os seguintes itens:

- criar, de forma automatizada, a estruturação de cada documento;
- criar o sistema de automatização da padronização;
- estabelecer um maior grau de facilidade na criação destes documentos;

- oferecer um sistema de representação mais evoluído tecnologicamente.

1.4. Metodologia Utilizada

Questões como o que possuem os documentos e como são construídos, quais bancos de dados devem ser criados para o sistema, foram estudadas e discutidas com quem realmente utiliza o sistema, ou seja, com os responsáveis pelas coordenações citados nas seções anteriores.

Pesquisas em livros foram realizadas para buscar as informações necessárias para a realização deste sistema, ou seja, a cada ponto essencial do sistema, o qual não estivesse completamente apto a realizar, foi realizada uma pesquisa, seja em livros, artigos de fontes confiáveis ou até mesmo em grupos de trabalho da linguagem especificada a fim de sanar o problema encontrado.

Outra questão importante e relevante a este trabalho é o fato de este não ficar apenas em documentos impressos. Como foi dito anteriormente, este trabalho foi implementado de forma a se possuir um sistema que solucione todos os problemas citados e existentes na instituição.

1.5. Resultados esperados

Ao término desta pesquisa, espera-se a criação do sistema que resolverá o problema da padronização e apresentação dos documentos. Também é esperado que as coordenações para as quais o sistema está sendo desenvolvido, e os alunos do curso de Sistemas de Informação desta Instituição, utilizem o recurso apresentado.

Outro fator importante e relevante que também é esperado após o término deste, é a padronização de tais documentos, analisando o fato que, na atualidade, a busca por padrões e qualidade é cada vez maior (BRANDÃO, 2004). Sendo assim, não se pode “ficar para trás”, levando em consideração o fato de esta ser uma entidade de renome internacional.

Por fim, espera-se um acúmulo de conhecimento nas linguagens e nas necessidades do sistema estudado.

1.6. Estrutura do trabalho

O Capítulo 2 tratará a ferramenta escolhida como base da padronização, XML, como dito anteriormente. A origem, as utilizações, um breve histórico e uma rápida apresentação desta linguagem serão abordados. Outros recursos pertinentes a essa linguagem, os quais serão utilizados no trabalho, serão apresentados.

Logo após, no Capítulo 3, será tratado o sistema *blackboard*. Uma crescente revolução tecnológica é apresentada a partir deste recurso, abrangendo novas formas de representação da informação, sendo esta simples, valiosa e necessária para os sistemas de hoje em dia

(OLIVEIRA, 2001). Do que se trata, para que serve e como funciona, sistemas semelhantes e todas as informações relevantes a este assunto serão tratados neste capítulo.

No capítulo 4, será apresentada a arquitetura do sistema. Os agentes que participarão do sistema, como participarão e como o sistema reagirá com os participantes, mostrando formas de interface e criação do sistema, ou seja, neste capítulo será tratado o projeto do sistema a ser desenvolvido. No capítulo 5 será tratada a implementação do sistema, quando será mostrado como foi desenvolvido o sistema. Serão apresentados as ferramentas utilizadas, a forma que elas foram utilizadas e o porquê do uso de cada ferramenta separada e unida com as demais.

Por fim, no capítulo 6 são encontrados as conclusões e os trabalhos futuros a este trabalho. E desta forma se define estruturalmente este trabalho.

2. XML

2.1. Introdução

De acordo com Daum e Merter (2002), desde meados da última década, é testemunhado um crescimento espantoso da *Web* e, aliado a ele, observado as limitações da linguagem HTML (*HyperText Markup Language*).

Em resposta à crescente demanda de extensões, bem como da necessidade de interoperabilidade, houve um esforço inicial de estender a linguagem HTML, provendo-a com folhas de estilo em cascata, mais comumente conhecida como CSS (*Cascading Style Sheet*). Entretanto, essa solução constituiu-se apenas num paliativo. Como resultado, um esforço coordenado pelo W3C (*WWW Consortium*) encontrado em www.w3c.org, foi movido visando oferecer uma nova linguagem que pudesse satisfazer às necessidades de interoperabilidade, escalabilidade e flexibilidade, permitindo-se fácil extensão.

Surge, então, a linguagem XML (*eXtensible Markup Language*) que é caracterizada por prover independência de dados bem como separar conteúdo de apresentação. Um programa em XML compreende a descrição de dados, tornando-se possível seu processamento por uma aplicação. XML tem sido, cada vez mais, utilizada por desenvolvedores de aplicações devido ao suporte que ela oferece tanto a interoperabilidade quanto funcionalidade da *Web*. Trata-se de uma linguagem baseada em texto a qual permite qualquer pessoa escrever um código em XML, sendo ele, facilmente, tanto compreensível às pessoas quanto manipulável aos computadores (MOULTIS, KIRK, 2000).

A XML é linguagem de marcação de dados (*meta-markup language*) que provê um formato para descrever dados estruturados. Ela foi desenvolvida para ser utilizada em marcações de documentos. Isso facilita declarações mais precisas do conteúdo e resultados mais significativos de busca por meio de múltiplas plataformas. O XML também vai permitir o surgimento de uma nova geração de aplicações de manipulação e visualização de dados via internet (ANDERSON et al., 2001).

Contudo, antes de se aprofundar neste tema, deve-se entender o que é uma linguagem de marcação, a qual será apresentada agora.

2.2. Linguagem de marcação

Segundo Marchal (2000), desde a era primordial, quando os seres humanos começaram a se comunicar e a representar o conhecimento, já havia a preocupação com a formatação de documentos. Mesmo sem computadores, as pessoas se preocupavam com a forma de representar seus documentos, tendo, assim como hoje, a necessidade de definir como estes seriam demonstrados.

Mesmo sem possuir as tecnologias que existem hoje, as marcações eram realizadas de outras formas, sendo por meio de símbolos, de letras e de caracteres especiais, para que as pessoas responsáveis pela formatação destes documentos soubessem apresentá-los de forma mais legível, já que entendiam mais facilmente como estes eram formados e como se organizavam estruturalmente (MARCHAL, 2000).

A partir desta época, várias formas de marcação foram utilizadas. Quem não se lembra dos antigos editores de texto? Neles, as marcações eram feitas pelos próprios autores dos

textos, sendo necessário definir, explicitamente no documento, qual forma de representação alguma parte do texto iria receber. O grande problema deste tipo de marcação era que os usuários dos editores tinham de saber todos os comandos necessários para realizarem as formatações.

Contudo, as marcações continuaram evoluindo. Nos editores de texto atuais, tais marcações são criadas de forma implícita para o usuário que, com apenas um clique, defina como alguma parte de seu documento será apresentada. Porém, a parte complicada do processo de formatação é feita nos “bastidores” do programa. O usuário não sabe como a marcação é feita. Apenas sabe que, quando o seu documento é aberto, ele é carregado da forma em que foi criado (MARCHAL, 2000).

Uma outra maneira de marcação é aquela voltada para programas e textos dispostos na Internet. Essa marcação se faz por meio de delimitadores chamados de *tags* que são reconhecidos por processadores deste tipo de documento. Geralmente estes arquivos são lidos e interpretados por navegadores de Internet, os quais interpretam as marcações e definem a forma como um texto, uma imagem e qualquer objeto seriam representados (PROFFITT, ZUPAN, 2001).

Em HTML, trabalha-se com *tags* pré-definidas. Sendo assim, o documento que será criado tem de obedecer às marcações que já existam nesta linguagem. Já em XML, alvo deste capítulo, é oferecida uma forma mais ampla e simples de marcação, para que o usuário possa definir suas próprias marcações, ou seja, suas *tags* (MOULTIS, KIRK, 2000).

2.3. Linguagens de marcação utilizadas na Internet

2.3.1. HTML

Segundo Moulitis e Kirk (2000), de todas as linguagens de marcação existentes, a HTML (*Hypertext Markup Language*) ainda é a mais utilizada na Internet, embora seja considerada a mais simples e limitada. Sua utilização está voltada tanto para a estruturação de documentos quanto na apresentação visual destes documentos em um navegador (*browser*).

Apesar de simples, ela pode representar documentos na Internet da mesma forma que as linguagens mais complexas. Talvez se torne um pouco ultrapassada, já que hoje existem linguagens e formas de representação mais evoluídas, não se usando apenas *tags* pré-definidas. Contudo, este fato não impede que a HTML continue sendo utilizada em larga escala.

HTML é derivada da linguagem pioneira de marcação SGML (*Standard Generalized Markup Language*) e foi criada por Tim Berners Lee – o idealizador da WWW – especificamente para a composição e apresentação de documentos na *Web* (MOULTIS, KIRK, 2000).

Um documento estruturado pode ser apresentado ao usuário de diferentes formas, por exemplo, numa impressão em papel, na tela de um navegador com janelas gráficas (*Windows*, *XWindows*, etc), na tela simplificada de um *notebook* ou de um telefone celular ou até em algum equipamento multimídia. Pode também ser enviado para processamento por algum software especializado como um Gerenciador de Banco de Dados (ANDERSON et al., 2001).

Os requisitos para estruturação de um documento são ortogonais aos requisitos necessários para sua apresentação. Este é um dos principais problemas da linguagem HTML em que ambos estão misturados e que recentemente foram atacados em duas frentes:

- introdução de Folhas de Estilo, também conhecida como *cascading style sheets*, permitindo especificar a apresentação desejada para cada elemento de um documento, separadamente da sua marcação HTML,
- especificação da linguagem XHTML, sucessora de HTML 4.01, seguindo o padrão da nova metalinguagem de marcação XML da *www.org*¹. XML é voltada para o intercâmbio de documentos via *Web* de forma independente da sua apresentação final e é um subconjunto restrito da linguagem pioneira de marcação, SGML.

Segundo Moulitis e Kirk (2000), um fator que enriquece muito a HTML são os programas externos a ela, os quais conseguem realizar todas as tarefas tão quanto às demais linguagens. Esses programas são reconhecidos como *plug-ins*.

Plug-ins são sistemas criados pelos fabricantes de navegadores *Web*. Eles permitem a execução de recursos que não foram desenvolvidos em HTML, assim como áudio e vídeo em uma página *Web*. O grande problema da utilização de *plug-in* é que, para cada recurso utilizado em um documento HTML, deve-se possuir seu respectivo *plug-in* o que pode diminuir a eficiência do computador, já que mais programas deverão estar instalados em um navegador de Internet para acessar todos e quaisquer sites.

2.3.2. O Surgimento da XML

Em meados dos anos 60, os desenvolvedores necessitavam de uma maneira mais fácil de construir documentos e compartilhá-los da melhor forma possível, tornando-os mais compatíveis com todos os computadores e sistemas utilizados no mundo. Essa ideologia era para todos os tipos de documentos, independentemente da forma que representassem (ANDERSON et al., 2001).

Pensando nisso, a IBM começou a trabalhar com GML (*Generalized Markup Language* – Linguagem de Marcação Generalizada). Mesmo a aceitação desta linguagem não sendo ampla, já que ela não era aquilo que os desenvolvedores esperavam, a IBM continuou trabalhando neste projeto por longos anos (RAY, 2001).

Por volta de 1986, a ISO (*International Organization for Standardization* – Organização Internacional de Padronização) reconheceu o modelo desenvolvido pela IBM. Esse modelo ficou conhecido como SGML (*Standard Generalized Markup Language* - Linguagem de Marcação Generalizada Padrão). A SGML, a partir deste momento, tornou-se a linguagem de marcação padrão para muitos tipos documentos, distribuídos pelas organizações do mundo (DEITEL et al., 2003).

Segundo Anderson et al. (2001), quando a *Web* se tornou um pouco mais popular no início dos anos 90, a SGML era considerada perfeita para ser introduzida na *Web* como linguagem de marcação padrão. Contudo, esta era muito complexa e difícil de ser utilizada. Assim sendo, surgiu a HTML (*Hipertext Markup Language* - Linguagem de Marcação de Hipertexto), a qual era uma aplicação da SGML, ou seja, era apenas uma representação visual de documentos.

¹ É importante lembrar que HTML, XHTML, XML e SGML são padrões internacionais originados na www.org,

A HTML logo se tornou popular, já que era fácil e simples de ser utilizada. Além da facilidade, ela suportava algumas falhas dos criadores que, mesmo assim, continuavam representando seus documentos de forma desejável.

Com o aumento freqüente do uso da Internet e com a necessidade da troca de informações cada vez maior, tornou-se necessário criar um sistema que melhorasse a forma de representação na *Web*. Tal melhora era a necessidade de tornar os sistemas mais padronizados, para gerar representações de documentos, as quais fossem mais claras e mais precisas. Além disso, esperava-se da *Web* um lugar onde todos os tipos de informações pudessem ser transportados, sejam estes vídeos, áudios, recursos multimídia ou textos (ANDERSON et al., 2001).

Para tentar resolver este problema, as empresas Microsoft e Netscape começaram a criar seus próprios recursos para os seus sistemas navegadores. Tais recursos foram criados a fim de realizarem operações que não eram suportadas simplesmente pela linguagem HTML. Estes novos recursos apresentados foram chamados de *plug-ins*. Isso foi uma revolução em termos de representação de conteúdo, já que, com esses recursos, os navegadores começariam a representar vídeo, áudio e todas as outras formas de documentos pela *Web*. Então, estes *plug-ins* apresentavam um problema: cada utilização deveria possuir o seu próprio *plug-in*, ou seja, a representação de vídeo tinha o seu *plug-in*, assim como a representação de áudio tinha o seu o que pode diminuir a eficiência do sistema, como visto anteriormente (MOULTIS, KIRK, 2000).

Dessa forma, com o aumento das necessidades dos desenvolvedores *Web*, chegou-se ao ponto em que as especificações estabelecidas pela HTML, que possuíam apenas recursos de apresentação, não estavam sendo suficientes para o desenvolvimento de documentos e sistemas. Necessitava-se de algo que traria maior flexibilidade; algo que fosse mais

extensível. Contudo, este novo sistema deveria possuir uma grande capacidade que se encontrava no SGML, que possuía recursos suficientes para solucionar qualquer problema em linguagens de marcação, o que será visto posteriormente neste trabalho. Apesar de possuir tal capacidade, considerada grande, este sistema também necessitava da aceitação do HTML, já que o HTML era compatível com programas de todo o mundo.

Mesmo necessitando da capacidade da SGML, esta nova linguagem deveria ser mais simples, já que a SGML tinha um alto grau de complexidade, tendo assim uma pequena aceitação por parte dos desenvolvedores.

Por esse motivo, nos meados dos anos 90, a *World Wide Web Consortium (W3C)* começou a desenvolver uma linguagem que sanasse todos esses problemas apresentados acima. Por volta de 1998, esta mesma organização lançou as especificações desta nova linguagem que ficou conhecida como XML (*eXtensible Markup Language* – Linguagem de Marcação Extensível) (ANDERSON et al., 2001).

Os objetivos iniciais desta linguagem eram torná-la uma ferramenta que possuísse a capacidade da SGML e a aceitação da HTML. Esta capacidade do SGML deveria ser utilizada de forma mais simples e objetiva. Nessa nova linguagem, desenvolvedores de todo o mundo poderiam criar seus programas em suas linguagens prediletas e, logo após, representariam as informações geradas por estes programas, por meio de documentos mais flexíveis e simples de se entender, ou seja, em documentos que utilizassem a linguagem XML (RAY, 2001).

Segundo Daum e Merter (2002), tal linguagem é considerada como ferramenta de padronização, já que esta utiliza marcações, isto é, *tags* que podem ser definidas pelo usuário. A partir destas *tags* definidas, o usuário pode delimitar seu texto da forma que bem entender, tornando-o assim mais legível e mais interpretável. Isso ocorre porque, geralmente, o usuário utiliza palavras-chave do texto que pretende se estruturar em XML. Para delimitá-lo, estas palavras-chave dão nome às novas *tags* que estão sendo criadas.

Outro fator que determinou a linguagem XML como uma linguagem de padronização é o fato de ela possuir compatibilidade com mais de um conjunto de caracteres. Este fato se torna uma vantagem em relação às outras linguagens que trabalham apenas com padrões americanos, assim como o ASCII (*American Standard Code* -Código Americano Padrão). Com o uso desses outros conjuntos, usuários de países que possuem caracteres especiais em sua língua, assim como os gregos, podem utilizar esta ferramenta, usando sua própria forma de linguagem. Mesmo assim, seu documento será reconhecido normalmente.

Outra utilização encontrada por usuários de todo o mundo, para a XML, é o armazenamento de informações. Muitos usam esta linguagem para armazenamento, já que esta ferramenta possui características para estruturação de dados utilizando uma arquitetura que segue o modelo de dados hierárquico. Tais características permitem que dados sejam organizados e trabalhados de forma simples (ANDERSON et al., 2001).

Um grande exemplo de utilização da linguagem XML, hoje em dia, é a construção de sistemas *e-business* (MARCHAL, 2000). Contudo, este uso não será tratado de forma abrangente, pois não é o objetivo deste trabalho.

É importante lembrar que a XML possui vários comandos e definições, mas esta parte da sintaxe não será apresentada neste trabalho.

2.3.3. XML

Para Anderson et al. (2001), XML é, conforme descrito, anteriormente, uma metalinguagem de marcação criada a partir de SGML. Na verdade, ela é uma forma restrita de SGML. Seus principais objetivos são:

- prover o intercâmbio de documentos por meio da *Web* de forma independente de sistemas operacionais ou formatos de arquivos;
- suportar uma grande gama de aplicações, permitindo a definição de elementos pelo usuário (ou aplicação) para estruturar o documento;
- facilitar a análise de documentos XML por programas;
- documentos XML devem ser legíveis por humanos;
- economia de *tags* de marcação não é importante;
- ter uma especificação formal para a marcação de documentos.

Como a XML não possui um conjunto pré-definido de *tags* ou elementos, ela é considerada uma meta-linguagem para descrição de linguagens de marcação. Desta forma, num documento XML, os elementos – aqui também chamados de *tags* – usados para definir o significado dos dados, podem ser definidos livremente de acordo com o domínio dos dados e da aplicação (DAUM, MERTEN, 2002).

Segundo Moulis e Kirk (2000), a XML não fornece mecanismos para apresentação de dados nem para ligação entre documentos. Tais recursos são disponibilizados por outras especificações de linguagens como, por exemplo, XSL (*Extensible Style Language*) e Xlink. A primeira é utilizada na descrição de estilos, já a segunda especifica *links* entre documentos.

O suporte que a XML oferece a separação entre conteúdo e apresentação é um aspecto de suma importância na linguagem.

Torna-se imperativo ainda acrescentar que XML é uma linguagem simples, possui um conjunto de estruturas de dados ricas, permite a troca e exibição de conteúdo de bases de dados e pode ser utilizada como formato para troca de mensagens na comunicação entre aplicações. Dentre essas, a troca de mensagens na comunicação entre aplicações de empresas oferece um meio de comunicação de baixo custo para aplicações B2B (*Business to Business*),

e essa é uma das áreas que também pode tirar proveito da tecnologia XML, contanto que protocolos seguros sejam usados para assegurar as comunicações (ANDERSON et al., 2001).

XML tornou-se um padrão internacional em 1998. Desde então várias linguagens de marcação seguindo o padrão XML foram criadas e estão sendo padronizadas no âmbito do *w3.org*. Recentemente, pequenas adaptações a HTML 4.01 para seguir os requisitos de XML, deram origem à linguagem XHTML (RAY, 2001).

De acordo com Moulis e Kirk (2000), ao contrário de SGML – apresentada no item 2.3.5. deste capítulo –, XML não requer um DTD (*Document Type Definitions* – Definição de Tipo de Documento) para cada documento, muito embora o DTD defina uma gramática para os elementos e atributos do documento, o que o tornaria muito desejável.

Segundo Deitel et al. (2003), a DTD é um dicionário que ajudará o sistema, que irá representar o documento XML, a interpretá-lo. Além de ajuda para a interpretação dos documentos XML, a DTD ajuda um analisador XML a verificar a validade de um documento XML, ou seja, se este foi escrito de forma sintaticamente correta.

Uma outra vantagem alcançada, usando esta definição de tipo de documento, é o fato de qualquer desenvolvedor conseguir entender um documento XML. Isso ocorre pelo fato de a DTD especificar a composição possível para um documento XML qualquer (RAY, 2001).

Segundo Daum e Merter (2002), esta definição de tipo pode ser interna, quando declarada no princípio do arquivo XML, ou externa, quando existe um arquivo separado que é vinculado ao arquivo XML. Outra classificação possível para DTDs é o fato de ela ser pública, quando se usa uma definição que já foi criada ou de sistema, quando o desenvolvedor cria sua própria definição de documento. Contudo, estas classificações entram no âmbito da sintaxe de desenvolvimento XML e como enfatizado anteriormente, não será apresentado neste trabalho.

De acordo com Anderson et al. (2001), se o documento XML possui um DTD associado e segue a gramática especificada pelo DTD, então ele é dito válido. A validação de um documento garante que os dados sejam completos, colocados na ordem correta no documento e com os valores apropriados dos atributos.

Além disso, documentos XML adotam e exigem uma forma mais simples e verbosa de aninhamento completo e explícito de todos os elementos de um documento XML, além de requisitos rígidos para definição de valores de atributos, etc. Nesse caso o documento é dito bem formado – *well formed* – e requisitos para isto estão detalhados na especificação XML 1.0² (MOULTIS, KIRK, 2000).

2.3.4. Relações entre HTML e XML

Embora as duas linguagens tenham princípios semelhantes, algumas características da XML tornam-na especialmente interessantes para certas aplicações. Tratar as diferenças entre o HTML e o XML não é tarefa fácil, uma vez que essas diferenças são extremas. A HTML é utilizada para a apresentação de conteúdo, enquanto a XML é usada na estruturação do conteúdo de um documento (ANDERSON et al., 2001).

De acordo com Marchal (2000), a principal diferença que se pode destacar está ligada à capacidade de apresentação. Enquanto a HTML possui um estilo único no documento, sendo este pré-definido na sua implementação, a XML possui uma forma de representação mais ampla, usando estilos diferentes para documentos diferentes. Ainda, em HTML, uma modificação no *layout* de um documento implicaria na modificação total deste. Isto não

² Disponível em: <http://www.w3.org/TR/2000/REC-xml-20001006> (acessado em 07 de outubro de 2004).

ocorre na linguagem XML. Caso haja necessidade de mudar o *layout* de um documento, basta alterar a folha de estilo relacionada a ele.

Além disso, pelo fato de a XML possuir suas marcações definidas pelo desenvolvedor, com ela é mais simples tratar de documentos grandes. Tal simplicidade provém do fato de se poder apresentar, a quem está acessando o documento, apenas as informações relevantes as suas necessidades. Este benefício é consequência do uso de folhas de estilo, as quais serão tratadas no decorrer deste capítulo (ANDERSON et al., 2001).

Segundo Marchal (2000), a flexibilidade da linguagem XML provém do armazenamento estruturado de dados. A partir deste armazenamento, o documento pode ser utilizado para fins diferentes dentro de organizações diferentes. Além disso, utilizando HTML não é possível extrair de um texto as partes as quais são consideradas convenientes, ou seja, esta linguagem não auxilia no intercâmbio de informações.

O XML possui outra vantagem sobre o HTML que se refere às consultas realizadas. Pelo fato de possuir marcações que se autodescrevem, o XML permite um maior nível de qualidade nas respostas às consultas realizadas pelo usuário, uma vez que com o uso do XML, pode-se distinguir os diferentes tipos de dados ali cadastrados. Desta forma é possível enviar os dados financeiros de uma empresa, para o seu respectivo departamento.

O XML ainda oferece outra vantagem: uma maior flexibilidade nos *links* encontrados em uma página *Web*. Para isso, a W3C desenvolveu para o XML, o *Xlink*. Usando o *Xlink* os desenvolvedores podem apontar seus *links* de forma dinâmica, utilizando apenas uma base de dados, o que não é possível em HTML (DEITEL et al., 2003).

Os *Xlinks* podem ser usados para “*definir relacionamentos entre documentos similares, para definir uma seqüência na qual os documentos devem ser navegados ou mesmo para embutir conteúdo não XML em um documento XML*” (ANDERSON et al, 2001, p. 343). Ao contrário dos *links* usados em HTML, o *Xlink* pode ser declarado fora do documento

fonte, permite declaração bi-dimensional, ou seja, é possível declarar um *link* que interliga duas páginas e não uma com a outra e permite a declaração de ações nas renderizações (MOULTIS, KIRK, 2000).

Segundo Ray (2001), o *Xlink* pode ser simples ou extensível. O *Xlink* simples é semelhante ao *link* utilizado em HTML, possuindo uma fonte e um alvo. Já com o *Xlink* extensível é possível definir *links* onde mais de um recurso é ligado a outro e definir *links* fora do código fonte da página *Web*. Assim sendo, utilizando deste, é possível ligar várias páginas entre si.

Com o recurso de *Xlink* extensível é possível criar um banco de *links*. Este banco armazena vários *links* que podem ser utilizados mais de uma vez. Sendo assim, quando se alterar o endereço especificado de uma página que está sendo *linkada*, não é necessário alterar todos os documentos que ligam a ela, bastando apenas alterar o endereço no banco de *links* (ANDERSON et al., 2001).

XML possui ainda um diferencial que permite uma aplicação bem mais interessante: a troca de informações na Internet (DAUM, MERTEN, 2002). Suponha por exemplo que uma fábrica de bolos de festa receba pelo correio encomendas de seus clientes. Para cada encomenda, uma carta diferente deve ser interpretada pelo funcionário responsável, de forma a extrair informações relevantes como modelo do bolo, tipo de cobertura, sabor do recheio, tamanho e formato do bolo. A tarefa não será nada fácil, já que cada encomenda pode ser feita de forma diferente, pois não há um formato padrão. É nesse momento que a linguagem XML pode contribuir para agilizar o processo.

A fábrica de bolos pode, por exemplo, usando a linguagem XML, determinar um formato padrão de encomenda de bolos que contenha as marcações que identificam certas informações específicas para essa atividade. Com isso, a fábrica pede que todos os seus

clientes criem suas solicitações na forma de arquivos XML utilizando o padrão especificado. A tarefa do funcionário se tornará bem mais simples.

Mas isso é só o início. Um padrão pode ser desenvolvido para que informações sejam trocadas entre a fábrica e os fornecedores. Outro para a troca de informações financeiras entre a fábrica e seu banco. Todos seguindo a mesma estrutura básica determinada pela linguagem XML. Por conta disso, documentos formatados em XML podem ser tratados automaticamente por computadores – o que aumentaria ainda mais a eficiência do uso dessa linguagem.

Como se pode perceber, por meio deste pequeno exemplo, a XML vem mudando a forma como se trocam informações na Internet. Por volta de quinhentos padrões diferentes já têm sido usados para tanto. De padrões simples como formatos para receitas de comida, calendários, jogos de xadrez e ofertas de emprego a padrões mais sofisticados para operações com cartões de crédito, comunicação em redes sem fio e troca de informações comerciais em geral. Cada um com suas marcações específicas, mas todos utilizando a mesma estrutura: XML.

2.3.5. SGML

De acordo com Moulis e Kirk (2000), a SGML (*Standard Generalized Markup Language* - Linguagem de Marcação Generalizada Padrão) foi a linguagem de marcação que originou tanto a HTML, quanto a XML, sendo ainda utilizada em algumas aplicações. Sistemas que utilizam esta linguagem são sistemas grandes e complexos, os quais precisam utilizar grandes especificações.

SGML foi criada no final da década de 60 pelos pesquisadores da IBM Goldfarb, Mosher e Lorie, com o objetivo de construir um sistema portátil, ou seja, um sistema independente de sistema operacional, formatos de arquivos, etc. O objetivo que os pesquisadores buscavam era o intercâmbio e manipulação de documentos. Eles optaram por um sistema de marcação generalizada, que deveria descrever a estrutura do documento e outros atributos, em vez de especificar o processamento a ser feito sob eles. Essa marcação também deveria ser rigorosamente definida, a ponto dos sistemas formais pudessem ser utilizados para processar o documento em questão (ANDERSON et al., 2001).

Em outras palavras, a marcação generalizada não restringe documentos a uma única aplicação, estilo de formatação ou sistema de processamento. Desde a sua concepção, portanto, SGML foi uma evolução sobre os sistemas vigentes de editoração eletrônica com funções de processamento complexas, embutidas na marcação do documento, denominadas por Goldfarb de sistemas de marcação *procedural*. Como exemplo destes sistemas podemos citar o clássico TeX (*Technical* - Linguagem para Preparação de Documentos) , desenvolvido por Knuth no final da década de 70 para preparação de textos matemáticos (www.cs-faculty.stanford.edu/~knuth/).

De acordo com Anderson et al. (2001), em 1986, SGML tornou-se um padrão internacional e foi adotada por várias empresas de porte como padrão para intercâmbio e armazenamento de documentos. Duas linguagens de marcação baseadas em SGML e largamente utilizadas são: *DocBook*, projetada para marcação de documentação técnica e TEI (*Text Encoding Initiative* – Iniciativa de codificação de Texto), projetada para marcação de textos literários.

Como SGML não possui um conjunto pré-definido de *tags* e de elementos, ela é na verdade uma *meta-linguagem* para especificar linguagens de marcação. Ela não possui uma semântica pré-definida.

A sintaxe de uma linguagem específica de marcação definida usando SGML é especificada por meio de construções coletadas num documento denominado DTD (*Document Type Definition*) ou incluídas diretamente no documento SGML, onde aparecem declarações de elementos, que especificam para cada elemento da linguagem todas as possíveis marcações válidas para o elemento, como por exemplo, a ordem e obrigatoriedade de elementos num documento, quantidade, etc, e declarações de atributos, que especificam, por exemplo, tipos de valores de atributos e valores *default*. O DTD define, portanto, uma gramática para a linguagem de marcação do documento (ANDERSON et al., 2001).

Segundo Moulis e Kirk (2000), o grande problema desta linguagem é o fato dela ser uma linguagem grande e complexa, já que possui centenas de páginas de especificação. Além disso, para desenvolver algo nesta linguagem é obrigatória a presença da definição de tipo de documento, assim como visto anteriormente, o que não é necessário em XML.

2.3.6. XHTML

XHTML (*eXtensible HyperText Markup Language* – Linguagem de Marcação de Texto Extensível) combina XML com HTML. Mais formalmente, XHTML é um XML reescrito do HTML. Isto se deve ao fato do XHTML seguir a sintaxe XML mas possuir funcionalidades semelhantes ao HTML, ou seja, possuir o mesmo funcionamento. Dessa forma, o “XHTML é um subconjunto do XML que foi desenvolvido para se comportar como HTML, contendo, porém, apenas as coisas mais interessantes” (PROFFITT, ZUPAN, 2001, P. xv).

Assim como podemos considerar o XML uma versão mais “enxuta” do SGML, o XHTML pode ser enquadrado com uma versão mais “enxuta”, porém do XML. Isso ocorre pelo fato do XHTML seguir as normas do XML, mas não utilizar todas as especificações encontradas neste. Apesar de prover recursos como definição de tipo de documento e folhas de estilo, a XHTML se comporta como o HTML, podendo assim ser interpretado pelos navegadores da atualidade (PROFFITT, ZUPAN, 2001).

Essa linguagem foi estabelecida como linguagem de marcação para *Web* no ano 2000 pela W3C. Ela surgiu pela necessidade de características de meta linguagem em representações HTML. Com isso, XHTML passou a ser utilizada por muitos desenvolvedores do mundo, no lugar da HTML 4 (RAY, 2001).

A busca de uma meta linguagem deve-se à crescente necessidade de dados que se auto descrevem. Utilizando este tipo de dados, os auto descritivos, é possível estabelecer um nível maior de padronização. Ainda, com essa linguagem, é possível utilizar somente o que é necessário dos documentos, ou seja, tem-se a liberdade de extrair dos documentos e representar para o usuário, somente os dados que são considerados relevantes.

Segundo Proffitt e Zupan (2001), essa nova linguagem se aproxima muito do HTML 4, o que facilita o trabalho dos desenvolvedores, os quais já possuem conhecimento sobre *tags* descritas em HTML. Contudo, ela oferece a flexibilidade e a portabilidade do XML. Dessa forma, têm-se uma linguagem de marcação de textos, com características e elementos da meta linguagem.

2.3.7. A XML nos navegadores

Segundo Moulitis e Kirk (2000), os mais conhecidos navegadores da atualidade - Netscape e o Internet Explorer - já oferecem recursos para validar documentos XML. Ainda há muito para se alcançar. Novas compatibilidades devem ser criadas, a fim de que documentos XML sejam interpretados diretamente pelos navegadores, pois, à medida que o tempo passa, essa linguagem se torna mais popular, sendo utilizada por todos os usuários de Internet e de Intranet de todo o mundo.

Ainda para Moulitis e Kirk (2000), um problema encontrado por estes navegadores é o fato de eles ainda não conseguirem interpretar os documentos feitos nesta linguagem, pois o que provém da maioria deles é um mecanismo que consegue interpretar as informações contidas nesses documentos, a partir da transformação de documentos XML em HTML. Essa compatibilidade com folhas de estilo, as quais serão apresentadas na próxima seção, já foram disponibilizadas na maioria dos navegadores utilizados atualmente.

As figuras seguintes representam, respectivamente, um documento XML válido (Figura 2.1) e um inválido (Figura 2.2), por meio de um navegador. Contudo, estas representações não utilizam uma folha de estilo. Sendo assim, o navegador só pode validar, ou não, o documento analisado.

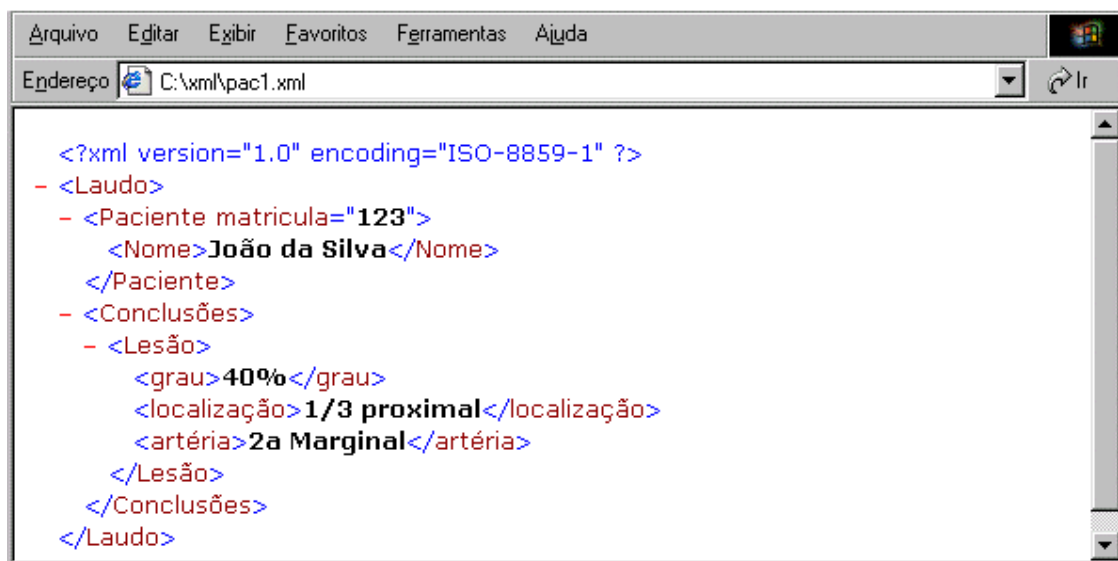


FIGURA 2.1 – Documento XML válido

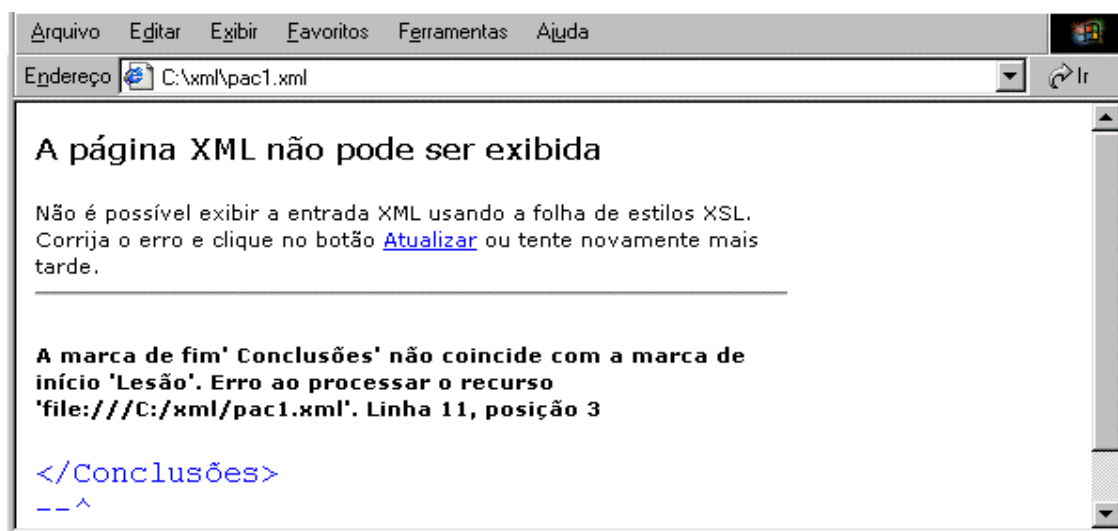


FIGURA 2.2 – Documento XML inválido

2.4. Folhas de estilo

Segundo Marchal (2000), o uso de uma folha de estilo pode ser comparado ao princípio básico da computação, assim como pode ser visualizado na figura 2.3. Isto ocorre pelo fato de possuir uma entrada, a qual é o documento XML, um processamento, que é a

transformação do XML a partir da folha de estilo e uma saída, que é o documento resultante deste processamento.

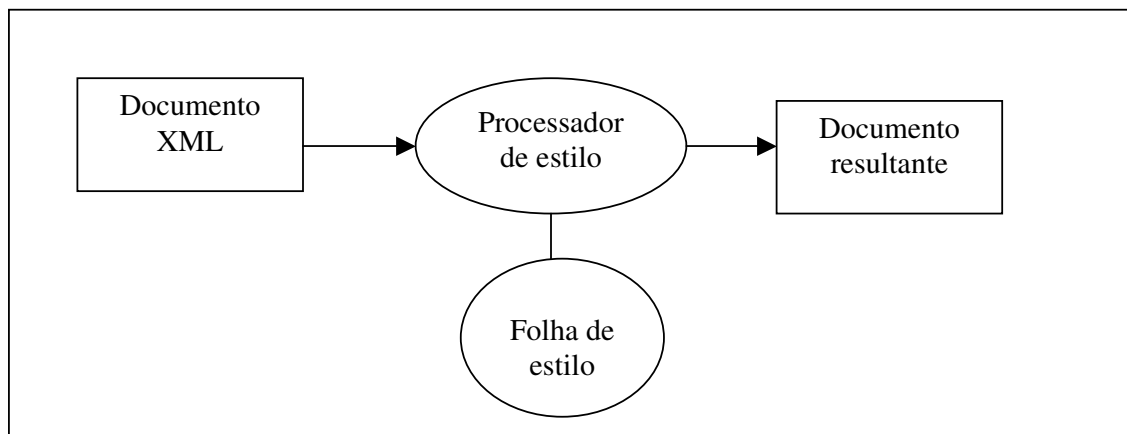


FIGURA 2.3 – Processamento de estilos

FONTE – MARCHAL (2000)

De acordo com Moulitis e Kirk (2001), ainda não é possível representar documentos XML nos navegadores. O máximo que eles realizam é a verificação da validade dos documentos XML.

Para resolver este problema, utilizam-se folhas de estilo que são usadas para a representação de documentos criados em XML. Essas folhas dão ao documento XML a interface, ou seja, a forma que o documento será apresentado para o usuário. Uma grande vantagem destas folhas é que elas são independentes da estrutura do documento. Elas apenas representam o *layout* que será usado em um determinado documento (MOULTIS, KIRK, 2000).

Segundo Anderson et al. (2001), folhas de estilos podem ser utilizadas para separar as informações, ou seja, cada documento será apresentado de acordo com a sua finalidade. A figura 2.4 demonstra este funcionamento.

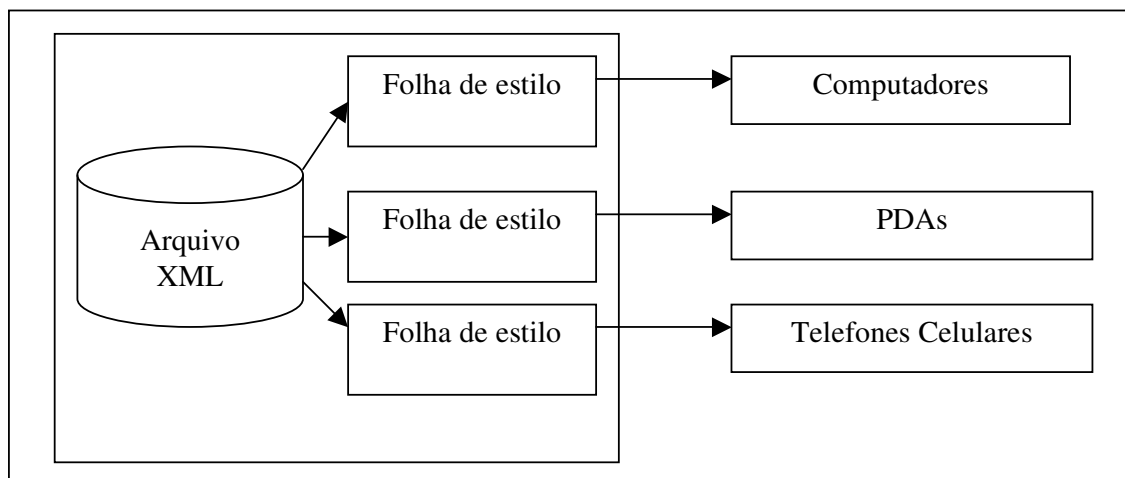


FIGURA 2.4 – XSL com diferentes finalidades

FONTE – ANDERSON et al (2001).

Uma folha de estilo usada atualmente é a CSS (*Cascading Style Sheets* – folhas de estilo em cascata) , ou seja, ela é uma forma de transformar um arquivo XML em HTML (DEITEL et al., 2003).

Segundo Ray (2001), esta forma de transformação possui seus problemas. Outras linguagens de estilo já apresentam características encontradas em linguagens de programação, utilizando condições, iterações e outros recursos. Esta, porém, não possui tais características, tornando assim inviável o seu uso em documentos mais complexos.

Para Moulitis e Kirk (2000), outra folha de estilo a se destacar é a DSSSL (*Document Style Semantics and Specification Language* – linguagem de especificação e semântica de estilos de documentos), poderosa e complexa linguagem de transformação de documentos XML. Ela transforma os documentos XML, para que estes possam ser representados pelos navegadores da atualidade. Essa linguagem tem o poder de transformar o XML em documentos HTML e formatá-los da melhor maneira possível.

A DSSSL foi desenvolvida para suportar documentos confeccionados em SGML. Desta forma, levando em consideração que a linguagem XML é um subconjunto da linguagem SGML, pode-se usar esta forma de transformação em documentos XML, sem causar nenhum transtorno.

Porém a folha de estilo que deve ser mais destacada é a XSL, já que está é a mais utilizada atualmente. Como foi criada após as demais, essa linguagem tem suas raízes nas outras linguagens apresentadas. A XSL é uma ferramenta de criação de folhas de estilo para XML. Esta linguagem possui uma grande vantagem sobre as outras apresentadas, já que foi desenvolvida exclusivamente para ser usada em XML. Este fator determinou a XSL como linguagem-padrão de folhas de estilo para XML (MOULTIS, KIRK, 2000).

Em relação a outras linguagens de estilo apresentadas, essa é mais simples, tornando, assim, o seu uso mais fácil. Esta linguagem é escrita em XML e tem o poder de transformar documentos XML em documentos HTML, RTF (*Rich Text Format*) e também em outros documentos XML (MARCHAL, 2000).

3. BLACKBOARD

3.1 Introdução

Atualmente, com a globalização, a informação tem uma influência visível nos processos de uma organização. Mas, tão importante quanto a informação, está a forma de apresentação desta, a qual pode ser por meio de textos, de sons e de imagens, dependendo do pretexto geral que essa forma possa ter (BORGES, 1995).

Quem nunca se deparou com cartazes, *folders* e vários outros tipos de apresentação de informação nas ruas? Segundo Borges (1995), a criatividade, a simplicidade e a disponibilidade se tornam cada vez mais importantes na demonstração do conhecimento. As apresentações mais simples, atrativas e que possuem uma maior disponibilidade, tornam a informação, que transportam, mais acessível ao público.

Tempos atrás, tal apresentação era realizada somente por meio de informações escritas ou desenhadas em paredes, muros e painéis. Logo após, as empresas publicitárias de todo o mundo iniciou uma forma mais dinâmica e atrativa, sendo murais mecânicos onde informações diferentes são mudadas mecanicamente de tempo em tempo. Este modo de apresentação ainda se encontra muito utilizado em propagandas e anúncios.

De acordo com Oliveira (2001), uns dos últimos estágios, encontrados atualmente, são os murais eletrônicos. Estes trabalham de forma semelhante à dos murais mecânicos, porém eles possuem uma melhor forma de utilização e apresentação das informações. Por possuir recursos altamente tecnológicos, uma maior quantidade de informação pode ser transportada

por este meio. Além disso, esta forma de apresentação contém meios mais criativos e atrativos para apresentar alguma informação.

Um dos tipos de mural eletrônico muito utilizado são os sistemas *blackboard*, ou seja, sistemas de quadro negro. Estes são usados de forma semelhante ao do quadro negro tradicional. Contudo, as informações são apresentadas utilizando recursos tecnológicos, tornando, assim, a apresentação da informação mais visível e legível (OLIVEIRA, 2001).

Em um sistema *blackboard*, as informações são apresentadas de forma dinâmica e precisa. Utilizando este recurso, um público alvo maior pode ser atingido, já que não é necessário estar no mesmo local para ter acesso às informações, como ocorreria em quadros negros tradicionais.

Utilizando *blackboard*, as informações necessárias podem estar dispostas virtualmente em algum servidor *Web*. Sendo assim, as pessoas podem acessá-las na hora oportuna. Por esse motivo, tal sistema se torna mais acessível, pois basta possuir um computador configurado para acesso à Internet.

Este sistema opera utilizando o ator produtor e o consumidor, sendo esses quem criam as informações para serem disponibilizadas e as pessoas que acessam essas informações, respectivamente (OLIVEIRA, 2001). A forma que o sistema tratará esses dois atores vai depender do propósito da utilização do quadro. Em alguns casos, é necessário que todos os usuários possam ler e escrever informações no sistema. Mas existem casos em que a escrita é restrita, ou seja, poucas pessoas devem possuir acesso para criar documentos disponibilizados. Em outros casos, até a leitura deve ser restrita, isto é, apenas pessoas autorizadas têm acesso ao conhecimento apresentado em um *blackboard*.

Cada dia, as pessoas se tornam usuárias de um mural eletrônico, seja em empresas, em instituições de ensino ou em sistemas especializados. O *blackboard* pode assumir várias características diferentes, possuindo desde recursos simples, como um quadro de avisos

utilizado em uma empresa, até recursos mais complexos, como sistemas que utilizam inteligência artificial distribuída. Uma outra utilização para o *blackboard*, muito usada atualmente, é o ensino a distância.

No entanto, o *blackboard* utiliza técnicas de trabalho cooperativo. Sendo assim, antes de entender melhor a utilização desses sistemas, deve-se conhecer esse tipo de trabalho que é o alvo da próxima seção.

3.2. Trabalho cooperativo

O trabalho cooperativo pode ser definido como todo e qualquer processo produtivo realizado por mais de uma pessoa, quando todos os envolvidos possuem um objetivo em comum. Barros, citado por Oliveira (2001), separa trabalho em grupo em duas dimensões: a dimensão cooperativa em que o processo é feito de forma vertical, ou melhor, as atividades são impostas aos membros do grupo, existindo uma certa hierarquia organizacional, e a dimensão colaborativa em que o processo é realizado de forma horizontal, ou seja, os membros pertencentes ao projeto possuem uma meta em comum, além de possuírem um poder de decisão igual ao dos demais participantes.

Barros, citado por Cavalcanti, Borges e Campos (1995, p. 04), afirma que “*o trabalho colaborativo envolve comunicação, percepção, coordenação, negociação, co-realização, compartilhamento*”. Estes processos de colaboração serão apresentados nas próximas seções.

3.2.1. Comunicação

Em se tratando de um trabalho no qual mais de uma pessoa irá trabalhar, esse processo se torna o mais importante do trabalho colaborativo. É por meio deste que as pessoas envolvidas num projeto expõem e esclarecem as suas dúvidas, apresentando idéias e informando aos demais participantes alguma alteração ocorrida no projeto.

De acordo com Fuks, Raposo e Gerosa (2002), a comunicação pode ser direta, sendo a informação passada por meio de mensagens, ou indireta, sendo a informação obtida a partir do conhecimento do grupo. Neste caso, é oriunda de experiências passadas, as quais o grupo em questão já vivenciou.

A comunicação também pode ser classificada de acordo com o tempo da interação. Nesta classificação, ela pode ser síncrona, onde os participantes interagem em tempo real, ou assíncrona, onde os integrantes se comunicam em tempos diferentes.

Uma outra classificação possível para este processo de trabalho colaborativo é o fato de ela ser estruturada, sendo utilizado um formato padrão para a realização da comunicação ou livre, quando esta é feita sem modelo algum, como é o caso de conversas ao telefone e *e-mails*.

3.2.2. Percepção

De acordo com Pinheiro (2001), a percepção de um trabalho colaborativo está relacionada à forma que uma pessoa visualiza um trabalho realizado por outra. Em alguns

sistemas, um usuário é informado sobre alguma alteração no projeto quando esta foi realizada, elevando assim, a percepção do usuário. Contudo, em outros sistemas, a mudança realizada em algum documento só é percebida quando o local da modificação é acessado.

3.2.3. Coordenação

Este processo existe devido à interdependência de atividades dentro de um mesmo projeto. É a partir deste que são planejadas as atividades e distribuídas as tarefas. Contudo, o projeto deve ser controlado e acompanhado de forma que não haja nenhuma interferência de uma tarefa em outra.

Assim, o processo de coordenação é uma tarefa árdua. Neste, são envolvidas questões sociais que possam influenciar no andamento do processo. As pessoas do grupo devem assumir os compromissos necessários a fim de que o projeto não seja prejudicado pelo cumprimento ou não de uma tarefa (FUKS, RAPOSO, GEROSA, 2002).

3.2.4. Negociação

Segundo Cavalcanti, Borges e Campos (1995), negociação é a forma que o projeto é gerido. Tende a decidir quais idéias serão aceitas e quais caminhos serão seguidos. Geralmente é realizada a partir de votações e argumentações sobre alguma idéia.

Dentro de um grupo com várias pessoas, é esperado que diversas idéias sejam apresentadas para uma só solução. O processo de negociação existe para que se possa encontrar um ponto ótimo a ser seguido por todos os membros da equipe. Para trabalhar com negociação, deve-se levar em consideração que a religião, a cultura e a política podem influenciar no modo como as pessoas pensam (CAVALCANTI, BORGES, CAMPOS, 1995).

3.2.5. Co-realização e Compartilhamento

Ainda em Cavalcanti, Borges e Campos (1995), co-realização é o processo de produção procedido por mais de uma pessoa. Para que este processo ocorra, é necessário especificar a forma que os produtos devem ser criados. Também é fundamental a geração de memória de grupo na qual ficarão armazenadas todas as alterações realizadas e os produtos finais obtidos.

Compartilhamento é o princípio básico para a realização de co-autoria. Para que pessoas possam construir, em conjunto, um produto ou objeto, é necessário que os membros do grupo compartilhem suas idéias e ações, a fim de que os demais participantes do projeto tenham acesso ao que foi alterado no processo e o porquê desta alteração (CAVALCANTI, BORGES, CAMPOS, 1995).

3.3. Utilização do *blackboard*

Um sistema de *blackboard*, como dito anteriormente, possui várias utilizações, dependendo apenas da finalidade que possui. Além de várias opções para o uso, este sistema também pode ser utilizado em diversos lugares, como empresas, escolas e universidades. Agora adiante serão mostradas algumas utilizações possíveis para o sistema de *blackboard*.

3.3.1. *Blackboard* como quadro de avisos

De acordo com Oliveira (2001), em diversas organizações do mundo, existe a necessidade de várias pessoas comunicarem entre si, utilizando avisos. Em tempos atrás, quando a tecnologia não havia se desenvolvido, estes recados eram feitos de forma manual, pois alguém utilizando um giz escrevia no quadro negro o aviso necessário, para que os companheiros pudessem ler.

Por meio da tecnologia atual, novos sistemas puderam ser criados. Estes realizaram o mesmo que o citado anteriormente, porém possuem recursos que tornam esta interação entre usuários mais simples e mais econômica. Uma pessoa pode entrar no sistema e deixar um determinado recado para outra que, por sua vez, acessará o quadro de avisos e terá capacidade de ler os recados deixados por outros integrantes de um determinado grupo (OLIVEIRA, 2001).

Esse fato possui um dos grandes processos de colaboração. Percebem-se, neste sistema, grandes características de comunicação, já que por meio deste quadro de avisos, um

integrante de um grupo deixa avisos para outro, a fim de que possa dar continuidade ou saber o que foi feito em um determinado projeto.

Essa interação ocorre de forma assíncrona. Os recados deixados no sistema são escritos de forma livre, ou seja, não possuem estrutura fixa. Assim sendo, eles são expostos da forma que foram criados. A outra classificação que este sistema apresenta é quanto ao modo que a comunicação é realizada, sendo esse de forma direta.

Um exemplo de *blackboard*, com essa finalidade, pode ser visualizado na figura 3.1.

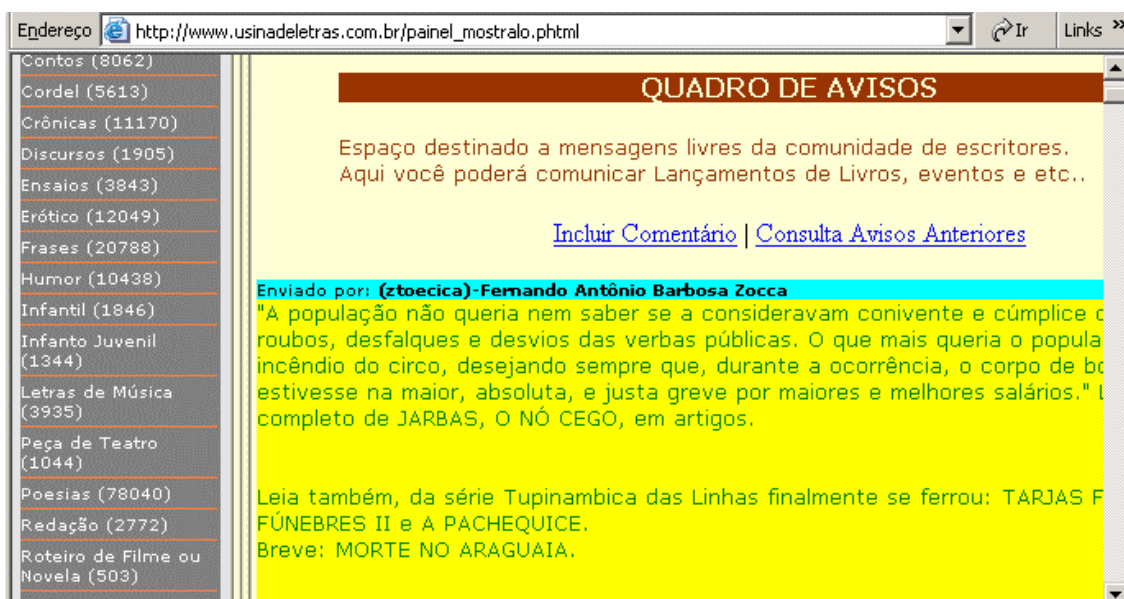


FIGURA 3.1 – Quadro de avisos

FONTE: (http://www.usinadeletras.com.br/painel_mostralo.phtml)

3.3.2. *Blackboard* auxiliando o ensino a distância

“No atual mundo dos negócios, o foco do aprendizado está se deslocando do ensino em sala de aula para o aprendizado no próprio local de trabalho, sendo este um dos aspectos

que tem se mostrado mais atraente para a iniciativa privada” (WICK, 1997 apud BOGO, 2003, p.42).

Costa, citado por BOGO (2003), afirma que a maior parte da utilização de sistemas de *blackboard* da atualidade possui características que auxiliam de forma esplêndida. EAD (ensino a distância). Por meio deste recurso, alunos e professores podem interagir de forma simples e completa. Esse sistema tende a completar o estudo presencial, já que esse ensino pode estender-se até as residências dos alunos.

Diversas instituições do mundo já utilizam esse recurso para ensino a distância. Alguns destes *blackboards* são utilizados para auxiliar o estudo presencial, enquanto outros foram criados exclusivamente para EAD. Esse sistema aproxima o professor do aluno e torna a comunicação entre eles mais simples e eficaz (BOGO, 2003).

Sistemas com essas características possuem recursos como troca de arquivos, imagens e textos, dentre outros. Eles também provêem recursos que integram provas e atividades extra classe com suas respectivas correções para os alunos. Mensagens entre os integrantes dos grupos também são suportadas por este sistema.

Além de prover o processo colaborativo de comunicação, este uso do *blackboard* possui recursos que auxiliam na construção cooperativa do conhecimento. A comunicação geralmente ocorre de forma livre, direta e assíncrona. Alguns destes sistemas possibilitam o uso de *chats*, o que torna a interação entre os usuários, síncrona.

Na PUC (Pontifícia Universidade Católica) de Minas Gerais, os cursos direcionados para a área de computação possuem um sistema que contém características de *blackboard*, o qual é voltado para EAD. O sistema chamado *learnloop* pode ser acessado a partir do endereço <http://inf.pucminas.br/learnloop>. Neste recurso, utilizado por esta instituição, é encontrado para cada disciplina o sistema de fóruns, o qual pode ser visualizado na figura 3.2.

Os alunos podem registrar as suas dúvidas, enquanto os professores disponibilizam informações suficientes para sanarem as dúvidas dos discentes.

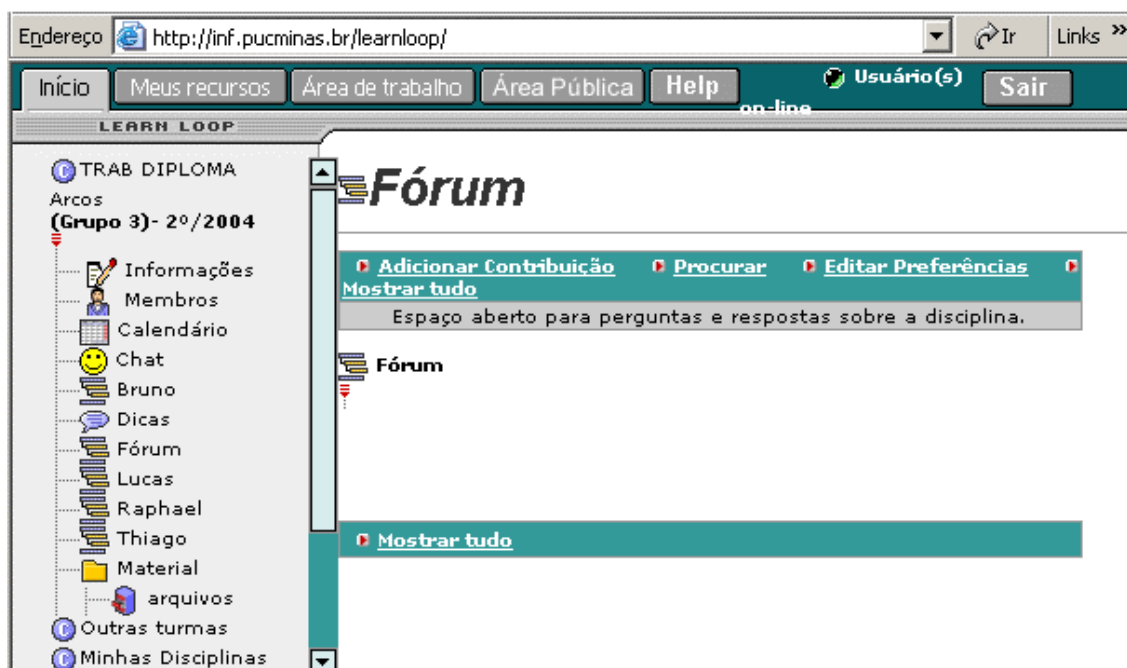


Figura 3.2 – Fórum do *Learnloop*

FONTE: (<http://www.inf.pucminas.br/learnloop/>)

Existem outros sistemas próprios para esta utilização. Dentre eles, podemos citar o *webct*, *e-college*, *learning space*, *aulanet*, *gestum* e o *docent*, além de outros existentes nas instituições do mundo.

3.3.3. *Blackboard* na inteligência artificial distribuída

Esta utilização talvez seja a mais complexa para sistemas de *blackboards*. Oliveira, citado por BOGO (2003, p. 13), comenta que nesses sistemas “*são encontrados algoritmos matemáticos que agem o tempo inteiro, gerando dados para o consumidor*”. Na IAD

(Inteligência Artificial Distribuída), encontramos o recurso de *blackboard* ativo, cujo papel é avisar aos integrantes alguma modificação realizada nos dados contidos na base de dados.

A principal utilização de *blackboard* em IAD está ligada aos algoritmos que geram dados o tempo todo a partir das informações contidas na área de armazenamento. Estes sistemas possuem uma área global, um conjunto de agentes, os quais formam a base de conhecimento e mecanismos de controle. A partir de informações dispostas no quadro negro, os agentes verificam seu conteúdo e, dependendo do contexto, criam dados que voltam para a área global. Tais dados são hipóteses para a solução de algum problema (NASSAR, 1996 apud BOGO, 2003).

Blackboard é uma estrutura única e compartilhada entre vários agentes, onde as informações serão escritas e lidas durante o desenvolvimento das tarefas. Como não há comunicação direta entre os agentes, eles devem consultar a estrutura de tempos em tempos para verificar se existe alguma informação destinada a eles (JAQUES, 1999 apud BOLZON, 2002, p.45).

A arquitetura encontrada neste tipo de sistema esta descrita na figura 3.3, segundo Bolzon (2002).

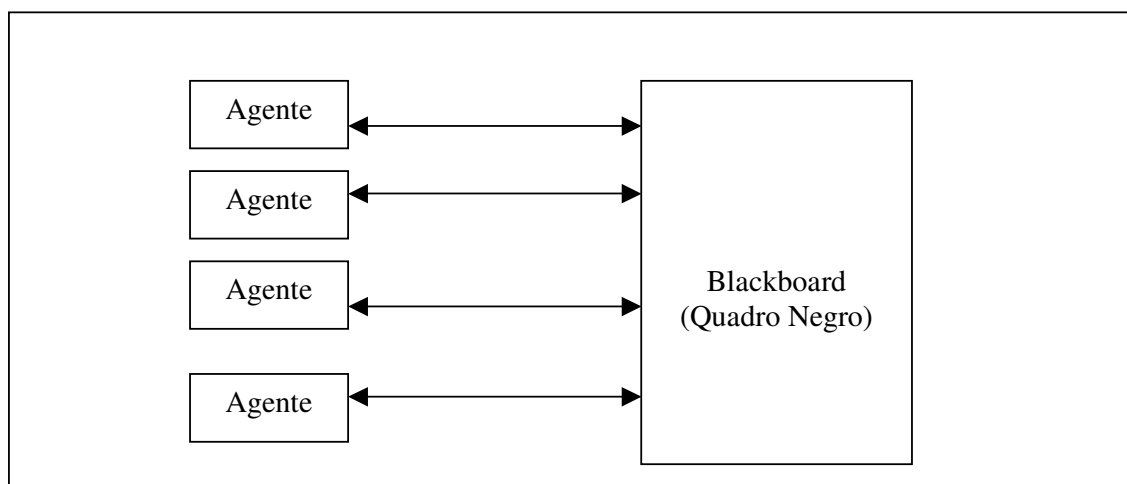


FIGURA 3.3 – Arquitetura *blackboard*

FONTE: Bolzon (2002, p.46)

3.4. *Blackboard* e este trabalho

A utilização do *blackboard* neste trabalho tem, como finalidade, a apresentação da informação. Neste sistema, o acesso às informações será disposto a todos os usuários. Contudo, a criação destes documentos será restrita, ou seja, só poderá ser realizada por professores autorizados a criarem estes documentos.

O sistema deste trabalho possui características de trabalho colaborativo. No entanto, essas características não passam de comunicação. Por meio do *blackboard*, haverá mais interação entre os alunos e a coordenação do curso. Essa comunicação será feita de forma estruturada e direta, porque os documentos gerados pelos professores possuirão estruturas pré-definidas, ou seja, padronizadas. Como os alunos não acessarão as informações, assim que estas estiverem dispostas no sistema, os docentes estarão se comunicando de forma assíncrona.

É importante lembrar que este sistema estará situado na *Web*, no *site* do Curso de Sistemas de Informação da PUC Minas, *campus* Arcos (www.arcos.pucminas.br/si). Os usuários só poderão acessá-lo se entrarem neste *site* da *Web* e acessar o devido *link*.

Mais detalhes sobre este *blackboard*, a ser utilizado, e o seu funcionamento serão apresentados no Capítulo 4 em que estará sendo detalhada a arquitetura do sistema.

4. ARQUITETURA DO SISTEMA

Este capítulo, apresenta a arquitetura do sistema proposto por este trabalho. Essa arquitetura está descrita na ilustração abaixo (Figura 4.1).

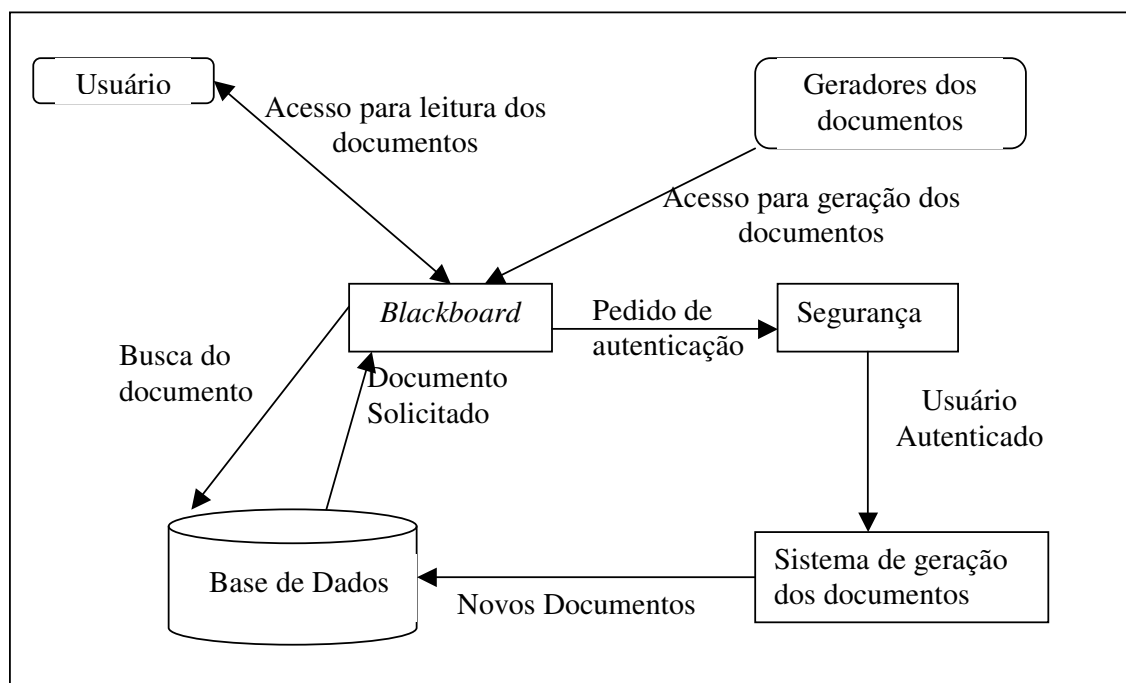


FIGURA 4.1 – Arquitetura do sistema

Mostrada a composição do sistema, serão especificadas as tarefas de cada item apresentado na figura 4.1.

4.1. Usuários

Os usuários do sistema são todas aquelas pessoas que o utilizam. Eles podem ser alunos, professores e funcionários da Instituição.

Contudo, o sistema pode ser acessado por qualquer pessoa que queira se tornar usuária deste. Para isso, no entanto, é necessário que ela se conecte à Internet e acesse o respectivo *link*. No entanto, pessoas que não possuem relação com a PUC Minas, não serão consideradas usuárias, já que essas não possuem interesse nas informações dispostas no sistema.

Para acessar o sistema, os alunos poderão utilizar qualquer navegador. As ações dos alunos são apenas leituras, já que eles acessam o *site* para obter informações oriundas dos geradores de documentos.

4.2. Blackboard

O sistema pode ser acessado a partir do *blackboard*, seja pelos usuários ou pelos geradores de documentos. Todos devem passar por esta página para acessar o restante do sistema. Sendo assim, pode-se considerar o *blackboard* como a página principal da aplicação.

Neste *blackboard* serão encontrados os avisos sobre as informações presentes na base de dados e o *link* necessário para redirecionar algum usuário para o sistema de geração de documentos. Se um usuário, que esteja lendo as informações presentes no *blackboard*, desejar visualizar o documento por inteiro, basta um clique, que este arquivo será buscado na base de dados.

Portanto, como foi mencionado anteriormente, este sistema é a interface da aplicação. Sendo assim, ele deve obedecer às regras de usabilidade.

Segundo Nielsen e Tahir (2002), a usabilidade de um sistema indica o esforço necessário do usuário para utilizá-lo. Geralmente, essa usabilidade é estudada em vários

objetos utilizáveis, como: telefones, microondas e outros. Em se tratando de usabilidade, podem ser encontradas as seguintes sub-características:

- inteligibilidade: que trata a facilidade no entendimento do sistema, ou seja, a facilidade para entender as funcionalidades do sistema;
- apreensibilidade: que trata a facilidade de um usuário se relembrar das tarefas necessárias para realizar alguma função, mesmo estando um tempo sem utilizar o sistema;
- operacionalidade: que trata a facilidade em termos de operação e controle do sistema;
- atratividade: que se refere à presença de recursos que tornam o sistema atrativo para os seus usuários.

Ainda para Nielsen e Tahir (2002), alguns fatores podem prejudicar a usabilidade de um sistema, como: irrelevância em que informações sem importância são encontradas em uma aplicação; repetições desnecessárias que ocorrem quando um conteúdo é exposto mais de uma vez; conflitos de informações que acontecem quando informações são estabelecidas em lugares indevidos; má estruturação da informação, o que mais ocorre na atualidade, em que informações são organizadas e apresentadas de forma incorreta e sem estrutura conveniente; inconsistência, quando existe falta de coesão nas informações e violação de convenções, que é o descaso com padrões estabelecidos por programas semelhantes.

A falta de usabilidade pode fazer com que o usuário pare de acessar o sistema, ou seja, se a usabilidade não agradá-lo na primeira visita, ele provavelmente não retornará a utilizar o sistema e fatalmente procurará outro que lhe atenda. Na maior parte das vezes, a preocupação dos usuários se relaciona à demora na carga das informações. Por esse motivo, o sistema deve possuir somente o que é necessário, a fim de gerar um melhor desempenho (SEARA, 2004).

Geralmente, em se tratando de usuários experientes, a usabilidade é mais facilmente obtida. No entanto, com um usuário novato na informática é mais difícil alcançar a usabilidade desejada. Este motivo faz com que a maior parte dos usuários novatos desista de utilizar o sistema, sem ao menos conhecer suas funcionalidades (SEARA, 2004).

4.3. Geradores dos documentos

Os geradores dos documentos são aquelas pessoas que possuem a finalidade de incluir novos documentos para serem acessados. Essas pessoas, inicialmente, são os professores responsáveis por áreas estabelecidas pela Coordenação do Curso de Sistemas de Informação da PUC Minas, como: atividades complementares, estágios, pesquisa e outros.

Para a utilização deste sistema, tais usuários acessarão a página principal, ou seja, o *blackboard*, e acessarão o *link* que os redirecionará para o sistema gerador de documentos. No entanto, antes de serem transferidos para o sistema, estes usuários deverão passar por uma autenticação.

Tal autenticação tem por objetivo, identificar as pessoas que estão querendo anexar informações ao sistema. A partir de um nome de *login* e de uma senha, inseridos pelo usuário, é possível liberar ou não a utilização do sistema para algum usuário. Essa interação poderá ser feita por meio de um programa para navegação na Internet.

Depois de autenticados, os usuários escolherão o tipo de documento que desejam gerar. Realizada a escolha, basta adicionar algumas informações, como data do evento ou inscrição. O sistema criará, então, o documento para ser apresentado.

Mais detalhes da criação dos documentos estão apresentados na seção que trata o sistema de geração de documentos. Vale lembrar que estes usuários poderão incluir novos documentos na base de dados, de forma remota, ou seja, não estando presentes dentro da Instituição.

4.4. Segurança

Antes de se pensar em segurança é preciso estabelecer o que vai ser protegido e de quem vai ser protegido. “*Para definirmos uma política de segurança, primeiramente devemos determinar quais os ativos em questão e qual o propósito do negócio da empresa*” (MELO, TRIGO, 2004, p.02).

Assim sendo, deve-se definir quais informações são elementares para a empresa, ou, como nesse caso, para a Instituição, antes de se definir uma política de segurança. Neste trabalho, os documentos criados devem ser protegidos das pessoas que têm apenas o acesso de leitura no sistema (MELO, TRIGO, 2004).

Se todas as pessoas que visitam o sistema pudessem criar documentos, vários documentos não-oficiais e até mesmo não-verdadeiros, seriam criados sem autorização de divulgação.

Contudo, como descrito anteriormente, a escrita no *blackboard* é restrita, ou seja, somente pessoas autorizadas podem inserir informações neste sistema. Para que isso ocorra, deve ser implantada uma política de segurança visando a garantia do controle de acesso ao sistema descrito.

Para garantir a segurança de uma empresa, vários critérios devem ser analisados. Segundo GIL (1998), a segurança deve se realizada visando a proteção dos recursos humanos, que são os profissionais que atuam na organização, a proteção dos recursos materiais, que envolve os *hardwares* e materiais de consumo da empresa, a proteção dos recursos tecnológicos, que englobam a parte lógica da organização, ou seja, *softwares* e dados e, por último, a proteção dos recursos financeiros, que são utilizados para a manutenibilidade dos recursos citados anteriormente.

Ainda para GIL (1998), a segurança é dividida em física, que tende a proteger o acesso físico à organização, lógica, que visa garantir a interoperabilidade dos serviços em termos de dados e programas, ocupacional, que se preocupa com a saúde ocupacional do profissional, ou seja, estabelece pontos ótimos em ergonomia e temperatura do ambiente e ambiental, que se relaciona com a infra-estrutura do ambiente em que estão empregados os recursos tecnológicos e confidencialidade, que visa a proteger as informações sigilosas de uma organização, seja na hora de aquisição ou de utilização.

Neste trabalho, o intuito é garantir a segurança na escrita dos documentos. Para isso será implementado um controle de acesso ao sistema de geração dos documentos. A partir dele, os usuários interessados deverão digitar o nome de *login* e senha a fim de realizar a autenticação. Por meio dessa autenticação, é estabelecido um objetivo de segurança que garante que somente pessoas autorizadas utilizem esse sistema.

Para estabelecer um nível ainda maior de segurança no controle de acesso, são utilizadas ferramentas de criptografia, as quais são usadas para mascarar as informações de senhas, passadas pelos usuários. Esta ferramenta é usada a fim de garantir que senhas não sejam “roubadas” durante o transporte pela rede.

A criptografia pode ser simétrica, onde a chave de encriptação é a mesma utilizada para decriptar. Uma chave de encriptação/decriptação é um código utilizado para cifrar ou decifrar um dado. A criptografia também pode ser classificada como assimétrica, a qual se baseia em chaves públicas para encriptar e privadas para decriptar dados (JUCÁ, 2003, p. 5-6).

No entanto, a criptografia não será implementada neste trabalho, devido às restrições de tempo, podendo esta ser explorada em trabalhos futuros.

4.5. Sistema de geração dos documentos

Esta parte da arquitetura é o lugar onde os documentos serão gerados. A partir desta ferramenta, os usuários geradores poderão criar novos documentos realizando poucos “cliques”.

Por intermédio de informações passadas por esses usuários, o sistema cria e monta um arquivo XML, contendo todas as informações necessárias dos documentos. Logo após, este arquivo é armazenado na base de dados, a fim de poder ser acessado pelos usuários de leitura.

Com este sistema, os geradores de documentos poderão criar os seus arquivos de forma fácil e rápida. Seguindo as instruções da tela, basta adicionar as informações necessárias em um formulário de preenchimento, que o novo documento será montado e enviado à base de dados.

Quando o usuário abre o sistema, é necessário que ele escolha o tipo de documento a ser gerado. Após esta escolha, o usuário é direcionado para outra página, a qual possui um formulário para que sejam inseridas informações relativas ao documento em questão. Logo após o preenchimento deste formulário, o sistema cria o documento, concatenando as informações passadas no formulário com um texto pré-definido. Cada tipo de documento possui o seu texto pré-definido. Após esta concatenação é gravado um arquivo XML, que guardará e demonstrará o conteúdo do documento utilizando uma folha de estilo XSL.

Como visto no capítulo 2, a linguagem XML foi estabelecida para prover uma maior padronização e por este motivo ela é utilizada neste trabalho. Com a folha de estilo relacionada, todos os documentos são apresentados de forma semelhante, ou seja, de forma padronizada (ANDERSON et al., 2001).

Sendo assim, a linguagem XML é utilizada no lugar de outras linguagens de marcação pelo motivo de trabalhar com o conceito de folhas de estilo, fato que lhe garante vantagens sobre a HTML, e por ser uma linguagem que garante a padronização de forma mais simples, o que não ocorre no SGML, já que esta é mais complexa de se utilizar (ANDERSON et al., 2001).

Tal aplicativo foi escrito e desenvolvido, utilizando as linguagens PHP (*PHP Hypertext Preprocessor* - Préprocessador de Hipertexto PHP) e HTML. Maiores detalhes da implementação serão mostrados no próximo capítulo.

4.6. Base de dados

A base de dados é o local onde serão armazenadas as informações oriundas do sistema gerador. Como tal sistema gera os arquivos em XML, a função desta base de dados é armazenar os arquivos XML recebidos.

Além de possuir esta interação com o sistema gerador de documentos, os arquivos que serão armazenados vêm desse sistema. A base de dados também possui uma interação com o *blackboard*. Como descrito nas seções anteriores, as informações que serão mostradas no *blackboard* estarão vindo deste armazenamento.

Tal base de dados deverá possuir uma tabela, contendo os nomes e as senhas dos usuários geradores, a fim de garantir o controle de acesso e outra tabela que contenha as informações sobre os avisos, encontrados no *blackboard*.

Ambas as tabelas serão gerenciadas pelo Sistema Gerenciador de Banco de Dados *MySQL*. Este SGBD foi escolhido, por ser o mais compatível com o PHP e ser o mais fácil de ser usado. Além disso, este SGBD possui a vantagem de ser livre, ou seja, qualquer pessoa pode usá-lo sem a necessidade de adquirir uma licença (CHOI et al., 2001).

A tabela de geradores dos documentos conterá o nome do professor, que é a chave primária da tabela, o seu nome de *login* e a sua senha. Essa tabela será utilizada na autenticação dos usuários.

A outra tabela relacionada é a tabela que será utilizada para mostrar, no *blackboard*, os avisos que contém as informações básicas dos documentos armazenados. Essa conterá um código, o título do aviso, a *URL* (*Universal Resource Locator* - Localização Universal do Recurso) para os documentos que estão sendo apresentados, e o nome e o telefone do responsável pelo aviso.

O *blackboard* será montado buscando as informações desta última tabela. As informações estarão todas dispostas na tela do *browser*. Dessa forma, quando um usuário necessitar ter acesso a algum documento de forma exclusiva, ou seja, de forma não resumida, basta dar um clique no devido *link*, para que esse documento seja mostrado em seu navegador. Tal redirecionamento será realizado a partir da *URL* cadastrada para cada documento.

Além dessas tabelas, é necessário possuir um local para armazenamento dos arquivos XML que serão utilizados. Esse local pode ser simplesmente um diretório, disposto no servidor *Web*.

O acesso a essa base de dados será feito somente pelo sistema. O sistema de geração dos documentos irá acessar essa base com a finalidade de autenticar os usuários e cadastrar

novas informações para serem representadas no *blackboard*, que por sua vez, utilizará a base de dados para montar os avisos , que serão mostrados.

Como dito anteriormente, o usuário de leitura poderá acessar a base de dados por meio do *blackboard*. Para isso, basta que ele “clique” no *link* que representa o documento por ele desejado. Após este “clique”, o usuário é redirecionado para esse documento.

5. SISTEMA DE PADRONIZAÇÃO DOS DOCUMENTOS

Neste capítulo será apresentado o sistema implementado. Além de apresentar as ferramentas utilizadas e como estas foram utilizadas, também serão mostradas as funcionalidades do sistema, ou seja, quais os recursos que ele oferece e como disponibiliza esses recursos.

5.1. Linguagens utilizadas

5.1.1. HTML

O sistema foi implementado a partir da linguagem HTML, apresentada no segundo capítulo deste trabalho.

HTML é uma linguagem poderosa e fácil de ser utilizada (PROFFIT, ZUPAN, 2001). Justifica-se aí o uso de tal linguagem para a implementação dos formulários de preenchimento, os quais são utilizados na criação de documentos, e nos processos de interface geral com o usuário. Sendo assim, todas as telas que o usuário possui acesso foram criadas utilizando HTML.

De acordo com Deitel et al. (2003), esta linguagem possui um grande poder de representação. Contudo, para a construção de sistemas mais complexos, como o sistema de padronização dos documentos, esta linguagem não pode ser utilizada sozinha. Pensando nisso,

nos módulos criados em HTML foram incluídos *scripts*, ou seja, códigos de programação que permitem um maior dinamismo.

Esses códigos utilizados foram criados usando a linguagem PHP, apresentada a seguir.

5.1.2. PHP

Choi et al. definem PHP como “*uma linguagem de programação para desenvolvimento do sites Web interativos e dinâmicos, originalmente projetada por Rasmus Lerdorf nos idos de 1994. Desde então, passou por várias alterações e foi adotada por vários programadores Web ao redor do globo*” (CHOI et al., 2001, p. xvii).

De acordo com Choi et al. (2001), PHP é uma poderosa linguagem de *script*, que é utilizada combinada com comandos de HTML. Além de atribuir a um *site Web*, criado em HTML um maior dinamismo, esta linguagem oferece aos usuários uma maior flexibilidade, já que por meio dessa pode-se exibir o mesmo *site* de formas diferentes adequando-se a cada usuário interessado.

PHP é uma linguagem baseada em cliente/servidor, portanto os *scripts* a serem rodados devem estar situados no computador servidor, que é responsável pela interpretação dos códigos dos *scripts* todas as vezes que um cliente solicitar uma comunicação. Desta forma, por estar sendo usado na máquina servidor, esta linguagem permite aos usuários a vantagem de usar sistemas multi-plataforma, ou seja, um usuário que possua um *browser* compatível pode estar utilizando qualquer sistema operacional, que os *scripts* continuarão sendo interpretados da mesma forma (CHOI et al., 2001).

Foram desenvolvidos vários *scripts* em PHP para a implementação desse sistema. Tais *scripts* foram criados para montar o *blackboard* junto à interface principal do sistema, criada em HTML, e também para realizar a construção dos documentos que são formados usando a linguagem de marcação XML.

Ao iniciar o sistema de *blackboard*, o *script* em PHP busca na base de dados as informações que serão utilizadas para a criação do mural de recados. A partir deste momento, essas informações recuperadas junto à base de dados são exibidas no *browser* cliente de acordo com uma estrutura estipulada usando a linguagem HTML.

Quando um usuário entra no módulo que garante a confecção dos documentos é exibido um formulário, criado usando a linguagem HTML. A partir do momento em que esse usuário completa o formulário, ou seja, preenche as informações necessárias, um *script* PHP “entra em ação”. Ele toma as informações oriundas do formulário HTML e concatena essas informações de forma a gerar a estrutura do documento. A partir do momento em que as informações estão organizadas da forma necessária, este *script* cria e armazena um arquivo, com extensão XML, no servidor *Web*.

Além dessas utilizações, o PHP também foi utilizado para construir um *script* que juntamente com um banco de dados MySQL implementarão uma maior segurança ao sistema. Tal segurança é oriunda da autenticação dos usuários, realizada a partir de um formulário HTML. Neste formulário, o usuário deve informar o *login* e a senha. Desta forma, o *script* PHP obtém as informações passadas pelo usuário e, junto ao banco de dados MySQL, verifica se este usuário possui acesso ao sistema.

Da mesma forma que o PHP auxilia na criação dos documentos e na montagem do *blackboard*, ele também auxilia na parte gerencial do sistema. Desta forma, o sistema possui *scripts* que garantem a criação, exclusão e alteração de usuários e também permite que

informações consideradas ultrapassadas e ainda presentes no *blackboard* sejam excluídas do sistema.

5.1.3. XML

Como dito no Capítulo 2 deste trabalho, XML é uma linguagem que garante a padronização de documentos. Isso ocorre pelo fato do XML ser uma meta-linguagem que possui *tags* autodescritivas, ou seja, *tags* que a partir de seus nomes já definem os seus conteúdos (ANDERSON et al., 2001).

Por este motivo, como o objeto principal deste trabalho é a padronização de documentos, é natural a utilização da linguagem XML para a realização desta padronização.

Sendo assim, a utilização de XML neste trabalho nada mais é do que a estrutura para os documentos a serem padronizados. Desta forma, tais documentos são criados de acordo com as normas de sintaxe XML. A estrutura destes documentos XML pode ser visualizada na figura 5.1.

```
<?xml version="1.0" encoding="ISO-8859-1" ?>
- <comunicado>
  <titulo>TÍTULO DO AVISO</titulo>
  <corpo>CORPO DO AVISO.</corpo>
  <data>DATA DO AVISO</data>
  <traço1>_____</traço1>
  <assinatura1>PRIMEIRA ASSINATURA DO AVISO</assinatura1>
  <cargo1>CARGO DA PRIMEIRA ASSINATURA</cargo1>
  <traço2>_____</traço2>
  <assinatura2>SEGUNDA ASSINATURA DO AVISO</assinatura2>
  <cargo2>CARGO DA SEGUNDA ASSINATURA</cargo2>
</comunicado>
```

FIGURA 5.1 – Estrutura do documento XML usado

5.1.4. XSL

Como visto no segundo capítulo deste trabalho, XML, sem uma folha de estilo, não é exibido pelos navegadores da atualidade. Ainda no segundo capítulo foram apresentadas as opções para folhas de estilo mais utilizadas (MOULTIS, KIRK, 2001).

De acordo com Anderson et al.(2001), a folha de estilo XSL é a mais utilizada atualmente, pois ela foi desenvolvida para uso exclusivo da linguagem XML e possui vantagens sobre as outras encontradas. Maiores informações destas vantagens podem ser encontradas no segundo capítulo deste trabalho.

Pelos motivos apresentados nos últimos parágrafos, a folha de estilo escolhida para este trabalho foi a XSL. Esta folha possui a função de exibir o arquivo XML. Desta forma, quando é solicitada a visualização de algum documento é realizada a transformação da XML para HTML. Tal transformação é de responsabilidade exclusiva da folha de estilo XSL. Após esta transformação, o documento XML pode ser visualizado em qualquer navegador que interprete a linguagem HTML.

5.2. Ferramentas utilizadas

5.2.1. MySQL

Para o funcionamento do sistema será necessária a utilização de uma base de dados. Vale lembrar que esta base de dados será usada para armazenar informações de *login* dos usuários e também informações sobre os documentos que foram criados e que serão utilizados na montagem do *blackboard*.

O MySQL é um SGBD (Sistema Gerenciador de Banco de Dados) que possui um relacionamento perfeito com aplicações *Web*. Este fato ocorre pelo motivo do MySQL possuir várias funções que tendem a facilitar o desenvolvimento *Web*. Desta forma, a maior parte das aplicações *Web*, da atualidade, utilizam este SGBD (CHOI et al., 2001).

De acordo com Choi et al. (2001), além de ter a vantagem de possuir funções que facilitam o desenvolvimento *Web*, esse SGBD também é um sistema eficiente e tolerante a falhas. O MySQL também possui a vantagem de ser facilmente utilizado e, como dito anteriormente, possui funções que facilita o seu uso.

Além de todas essas vantagens acima citadas, o MySQL é um sistema de código livre, ou seja, ao instalar este sistema, é instalado o código fonte do mesmo. Sendo assim, um usuário pode estudar e alterar seu código, afim de melhor adequar este SGBD ao seu uso (CHOI et al., 2001).

Sendo assim, a partir de todas essas vantagens apresentadas pelo MySQL, ele foi escolhido como o SGBD a ser utilizado neste sistema. Como dito anteriormente, ele terá a função de armazenar informações, que garantirão o bom funcionamento do sistema.

5.3. Funcionalidades do sistema

5.3.1. *Blackboard*

Os documentos que estão sendo padronizados serão exibidos eletronicamente e virtualmente. Essa apresentação se dará a partir de um sistema chamado *blackboard*, o qual foi visto no terceiro capítulo deste trabalho.

Tal *blackboard* é um mural eletrônico onde as informações estarão sendo dispostas para os interessados. O *blackboard* deste trabalho pode ser visualizado na figura 5.2.



FIGURA 5.2 – *Blackboard* do sistema

Este *blackboard* é a primeira tela que o usuário terá acesso. Por meio deste sistema, são exibidas as informações referentes aos documentos criados.

Neste *blackboard* é apresentado apenas o título da informação. Caso o usuário deseje obter mais informações sobre os documentos, basta clicar no *link* “Maiores Informações” ou procurar o professor responsável pelo aviso. O nome e o telefone deste professor é descrito logo após o aviso.

Desta forma, o *blackboard* é a interface principal com o usuário. Caso o usuário queira entrar no sistema, que gerencia usuários e os documentos, basta clicar no *link* “Entrar no Sistema” e ser autenticado (Figura 5.4).

5.3.2. Sistema gerente do *blackboard*

Nesta parte do sistema, serão encontradas as opções da aplicação. É aqui que o usuário define o que quer fazer, seja criar algum documento, excluir algum aviso do *blackboard* ou gerenciar os usuários. Todas essas funções são iniciadas a partir deste momento, o que pode ser visualizado na figura 5.3.



FIGURA 5.3 – Sistema gerente do *blackboard*

A partir deste sistema, o usuário escolhe a opção desejada e é direcionado para o respectivo local.

5.3.3. Criação de novos documentos

Assim que o usuário opta pelo documento que deseja criar, ele é direcionado para a tela de criação de novos documentos. Tal tela é diferente para cada tipo de documento. Contudo, antes de entrar em qualquer opção do sistema citada na seção anterior, o usuário deve ser, novamente, autenticado. Esse processo de autenticação é mostrado na figura 5.4.

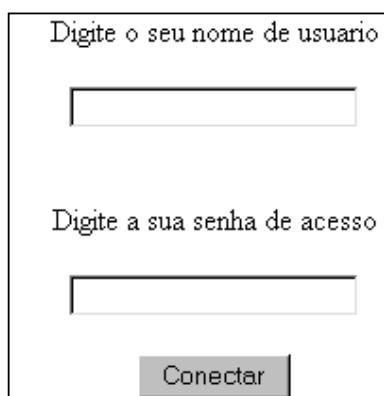
A imagem mostra uma interface de autenticação dentro de um retângulo com uma borda preta. No topo, há o texto "Digite o seu nome de usuario" em uma fonte serifada. Abaixo dele é um campo de entrada retangular. Mais abaixo, há o texto "Digite a sua senha de acesso" na mesma fonte. Abaixo dele é outro campo de entrada retangular. No fundo, centralizado, há um botão retangular com o texto "Conectar".

FIGURA 5.4 – Autenticação dos usuários

Para ser autenticado, o usuário deve digitar o seu nome de *login* e a sua senha e clicar em “Conectar”. Ao fazer isto, desde que sua senha e *login* sejam validadas, o usuário se conecta ao programa.

Suponha que o usuário tenha escolhido criar um novo documento de palestra. Se este foi autenticado, ele é direcionado para a página que cria novos documentos de palestra. Esta página, a de criação de novos documentos de palestra, está ilustrada na figura 5.5.

Endereço <http://localhost/palestrai.html>

Data desta palestra: Tema da palestra:

Horário inicial da palestra: Horário final da palestra:

Nome do palestrante:

Escolha a primeira assinatura

- ☒ Professor Marco Rodrigo Costa
- ☐ Professor Alisson Rabelo Arantes
- ☐ Professor Hermes P. de Moraes Junior
- ☐ Professora Laura Aguiar Ferreira
- ☐ Professor João Caram Santos de Oliveira
- ☐ Professor Ramon Sila Leite

Escolha a segunda assinatura

- ☐ Professor Marco Rodrigo Costa
- ☐ Professor Alisson Rabelo Arantes
- ☐ Professor Hermes P. de Moraes Junior
- ☐ Professora Laura Aguiar Ferreira
- ☐ Professor João Caram Santos de Oliveira
- ☐ Professor Ramon Sila Leite

[Voltar ao sistema principal](#)

FIGURA 5.5 – Criação de novos documentos de palestra

A partir desta página, é criado o novo documento de palestra. O usuário deve preencher todos os campos do formulário e clicar no botão “Criar Arquivo”. Caso o usuário não tenha preenchido todos os campos do formulário, o sistema lhe emite um aviso de que todos os campos devem ser preenchidos. Caso contrário, um aviso de operação bem sucedida é exibido na tela e é possível visualizar o novo documento, a partir de um botão “Visualizar Documento”, presente na tela. Clicando neste *link* o usuário terá acesso ao documento que criou, o qual é mostrado na figura 5.6.

Assim que um novo documento é criado, este é enviado para o *blackboard* contendo as suas informações, o *link* para o arquivo e o responsável pelas informações. Para retirar este novo documento do *blackboard* é necessário acessar a página de “Excluir Aviso” que será mostrada na próxima seção.



Comunicado

A Coordenação do Curso de Sistemas de Informação comunica que no dia 31 de Outubro, das 19:00 às 20:40, haverá a realização da palestra: "Distribuição de Software", ministrada por "Erick Santos". Assim, é solicitado aos professores que compareçam, juntamente com suas turmas, ao auditório da PUC Minas Arcos no horário e data indicados.

Arcos, 31 de outubro de 2004

Prof. Marco Rodrigo Costa
Coordenador

Prof. Hermes P. de Moraes Junior
Coordenador de Atividades Complementares

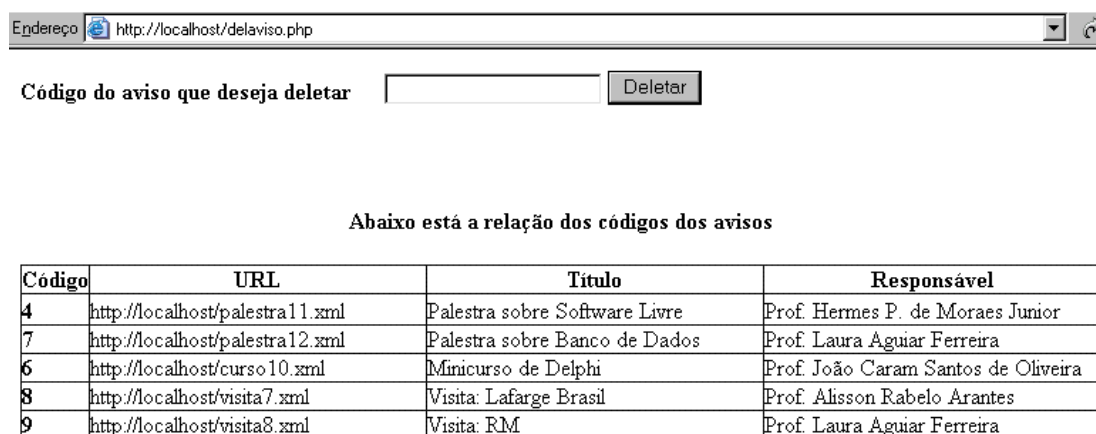
FIGURA 5.6 – Documento de Palestra

5.3.4. Remoção de avisos do *blackboard*

Para remover algum aviso do *blackboard*, basta clicar no botão “Excluir Aviso”, presente no sistema gerente do *blackboard*. Com este clique, o usuário é direcionado para a

tela de remoção de avisos. Contudo, antes disso, o usuário deve ser autenticado. Esse processo de autenticação é semelhante ao mostrado na figura 5.4.

Após a realização da autenticação, caso o usuário seja validado, ele terá acesso à página de exclusão de avisos do *blackboard*. Nesta página, basta que o usuário digite o código do aviso que deseja remover do *blackboard* e em seguida clicar no botão “Deletar”. Os códigos de todos os avisos podem ser visualizados na parte inferior da página. Assim que é terminada a operação é emitida, ao usuário, a informação de que o aviso foi excluído com sucesso. A página de remoção de avisos está ilustrada na figura 5.7.



Código do aviso que deseja deletar

Abaixo está a relação dos códigos dos avisos

Código	URL	Título	Responsável
4	http://localhost/palestra11.xml	Palestra sobre Software Livre	Prof. Hermes P. de Moraes Junior
7	http://localhost/palestra12.xml	Palestra sobre Banco de Dados	Prof. Laura Aguiar Ferreira
6	http://localhost/curso10.xml	Minicurso de Delphi	Prof. João Caram Santos de Oliveira
8	http://localhost/visita7.xml	Visita: Lafarge Brasil	Prof. Alisson Rabelo Arantes
9	http://localhost/visita8.xml	Visita: RM	Prof. Laura Aguiar Ferreira

FIGURA 5.7 – Remoção de avisos do *blackboard*

5.3.5. Controle de usuários

Nesta parte do sistema são permitidas a inserção, alteração e remoção de algum usuário. Para ter acesso a esta parte, mais uma vez é necessário passar por um processo de autenticação.

Após a autenticação, caso o usuário seja validado, ele pode incluir um novo usuário. Para isto basta informar o nome, o *login* e a senha do novo usuário e clicar no botão

“Cadastrar” (Figura 5.8). Uma mensagem de operação sucedida com sucesso será exibida para o usuário.

A screenshot of a web browser window showing a registration form. The address bar displays 'http://localhost/insuser.php'. The form contains three input fields: 'Nome do usuario:', 'Login do usuario:', and 'Senha do usuario:'. A 'Cadastrar' button is positioned to the right of the password field.

FIGURA 5.8 – Inclusão de novos usuários

Se a utilização for para remoção do usuário, basta informar o *login* do usuário que deseja excluir e clicar no botão “Deletar” (Figura 5.9). Uma mensagem de operação bem sucedida será emitida para o usuário.

A screenshot of a web browser window showing a deletion form. The address bar displays 'http://localhost/deluser.php'. The form has a single input field labeled 'Nome do usuario que deseja deletar' and a 'Deletar' button to its right.

FIGURA 5.9 – Exclusão de usuários

Por fim, é possível a alteração dos dados de algum usuário. Para isso, deve-se informar o nome do usuário, o seu novo *login*, caso queira alterá-lo, e a sua senha, se assim for sua vontade. Em seguida, o usuário deve clicar no botão “Alterar” (Figura 5.10). Logo após, é exibido para o usuário uma mensagem de operação bem sucedida.

A screenshot of a web browser window showing a modification form. The address bar displays 'http://localhost/altuser.php'. The form includes three input fields: 'Nome do usuario', 'Novo login do usuario:', and 'Nova senha do usuario:'. An 'Alterar' button is located to the right of the password field.

FIGURA 5.10 - Alteração de usuários

6. CONCLUSÕES E TRABALHOS FUTUROS

Para realizar este trabalho, partiu-se do pressuposto de que todos os documentos eram criados sem uma devida padronização. O presente trabalho teve, como objetivo principal, estabelecer normas para a criação de uma estrutura padrão para todos os documentos.

Sua realização seguiu uma abordagem em que as principais tecnologias empregadas foram apresentadas e detalhadas.

Como a linguagem utilizada no desenvolvimento desse trabalho foi a XML, antes de trabalhar com a mesma, foram apresentados os conceitos de linguagem de marcação. Algumas linguagens utilizadas com este intuito também foram apresentadas e comparadas com a XML.

Como parte da padronização, foram estabelecidos os conceitos relativos à linguagem de marcação XML, a qual é estabelecida como a linguagem que provê um padrão em desenvolvimento *Web*. Também, foram apresentadas folhas de estilo e a definição de tipo de documento, as quais foram utilizadas no desenvolvimento XML.

Além de realizar uma padronização nos documentos, este trabalho também se preocupou com a importância da apresentação dos documentos. Uma nova maneira de exibir documentos foi utilizada neste trabalho. Trata-se de um *blackboard* que se refere a um mural disposto de forma eletrônica, o qual é uma saída revolucionária para trabalhos que envolvem pesquisa em grupo.

O *blackboard* pode ser usado para auxiliar o ensino a distância e a inteligência artificial. Pode também ser usado, com o único intuito de estabelecer a comunicação entre os participantes de seu sistema.

A partir das definições de XML e de *blackboard*, foi possível estabelecer critérios para a implementação de um sistema que permita a comunicação entre os alunos do Curso e a Coordenação do mesmo, da melhor maneira possível. Além de uma maior comunicação, este sistema garante mais facilidade na criação e padronização dos documentos em questão.

As vantagens da utilização deste sistema são inúmeras, e todas essas vantagens foram apresentadas no decorrer deste trabalho. Assim sendo, foram definidas as soluções para o problema apresentado.

Um dos primeiros fatores a serem trabalhados futuramente é a *criptografia* de senhas. Como dito anteriormente, este processo garante uma maior segurança. Sendo assim, é pretendido estabelecer regras de criptografia para este sistema.

Com o atual crescimento da tecnologia, cada vez mais é crescente o uso da computação móvel. Para a realização desta computação móvel, podem ser utilizados telefones ou computadores portáteis (ANDERSON et al., 2001).

Ainda como trabalhos futuros, tem-se a execução com maior ênfase sobre o protocolo WAP, o qual transporta as informações presentes em um servidor, utilizando-se de redes sem fio. Pretende-se estudar WAP e fornecer ao sistema, implementado neste trabalho, a capacidade de utilizar este recurso.

Outro fator que se estende a este trabalho é o tratamento da busca das informações. Como explicitado durante o trabalho, as informações são buscadas na base de dados, pelo *blackboard*. Pretende-se estudar melhor esta busca e estabelecer um ponto ótimo de eficiência, tornando assim o sistema mais rápido e acessível.

Outra atividade a ser realizada como trabalho futuro é o aperfeiçoamento do sistema. Para isso algumas funcionalidades devem ser acrescentadas e outras alteradas. Uma funcionalidade a ser agregada é a padronização encontrada no trabalho realizado, neste mesmo período e local, por Diego Zuquim, sob orientação do Professor Marco Rodrigo Costa.

Tal trabalho, chamado “Desenvolvimento de uma aplicação para formatação de documentos de atividades acadêmicas”, realiza a padronização em documentos acadêmicos dos docentes, assim como provas e exercícios.

Esta aplicação pode ainda ser estendida para toda a Instituição. Este trabalho foi realizado para o Curso de Sistemas de Informação da PUC Minas, campus Arcos. Para isso, será necessária a inclusão de novos documentos, a fim de que tal aplicação passe a ser utilizada tanto pelas coordenações dos cursos quanto pela diretoria da Instituição.

REFERÊNCIAS

ANDERSON, Richard et al. **Professional XML**. 1. ed. Rio de Janeiro: Ciência Moderna, 2001. 1266p.

ANTÔNIO, Marco. **Ambiente colaborativo de ensino e aprendizagem**. [S.I] [199-] Disponível em <http://www.dca.fee.unicamp.br/courses/IA368F/1s1999/Tema6/rel6_1.html> Acesso em 1 jun. 2004.

ARROYO, Carlos. **Proposta de ambiente colaborativo de ensino e aprendizagem**. [S.I] [199-] Disponível em <http://www.dca.fee.unicamp.br/courses/IA368F/1s1999/Tema6/rel6_2.html> Acesso em 1 jun. 2004.

BOGO, Henrique Luis. **Criação de comunidades virtuais apartir de agentes inteligentes: uma aplicação em e-learning**. 2003. 116f. Dissertação (Mestrado em Engenharia de Produção) - Universidade Federal de Santa Catarina. Disponível em: <<http://teses.eps.ufsc.Br/defesa/pdf/9874.pdf>> Acesso em 26 out. 2004.

BOLZAN, Willian. **Estudo comparativo sobre sistemas tutores inteligentes multiagentes Web**. 2002. 54f. Dissertação (Mestrado em Educação) - Faculdade de Informática PUCRS. Disponível em: <<http://www.inf.pucrs.br/tr/tr024.pdf>> Acesso em 26 out. 2004.

BORGES, Mônica Erichsen Nassif. A informação como recurso gerencial das organizações na sociedade do conhecimento. **Revista Ciência da Informação**, Brasília, Vol 24, n. 2, p.181-188, mai/ago. 1995. Disponível em: <<http://www.ibict.br/cienciadainformacao/include/getdoc.php?id=961&article=599&mode=pdf>> Acesso em 26 out. 2004.

BRANDÃO, Marcelo Moll. Marketing de relacionamento: valor do cliente é diferente de qualidade total. **Marketing Place**, São Paulo, 2004. Disponível em: <http://gazetaonline.globo.com/marketingplace/artigo.php?id_artigo=26> Acesso em: 20 out. 2004.

CAVALCANTI, Maria Cláudia Reis; BORGES, Marcos Roberto da Silva; CAMPOS, Maria Luíza Machado. Suporte por Computador ao Trabalho Cooperativo. In: XV Congresso da Sociedade Brasileira de Computação, 1995, Canelas-RS. **Jornada de Atualização em Informática**. Disponível em: <<http://chord.nce.ufrj.br/docs/cursos/JAI1995-apostila.pdf>> Acesso em: 20 out. 2004.

CARUSO, Carlos Alberto Antonio. **Segurança em informática e de informações**. 2. ed. São Paulo: SENAC-SP, 1999. 367p.

CHOI, Wanky et al. **Beginning PHP4 Programando**. 1. ed. São Paulo: Makron Books, 2001. 719p.

DAUM, Berthold; MERTER, Udo. **Arquitetura de sistemas com XML**. 1. ed. Rio de Janeiro: Campus, 2002. 441p.

DEITEL, H. M. et al. **XML: como programar**. 1. ed. Porto Alegre: Bookman, 2003. 972p.

FUKS, Hugo; RAPOSO, Alberto B.; GEROSA, Marco A. Engenharia de groupware: desenvolvimento de aplicações colaborativas In: XXI JORNADA DE ATUALIZAÇÃO EM INFORMÁTICA. 2002. Florianópolis- SC. **Anais do XXII Congresso da Sociedade Brasileira de Computação, V2, Cap. 3, p. 89-128, 2002.** Disponível em: <<http://groupware.les.inf.puc-rio.br/groupware/publicacoes/JAI2002.pdf>> Acesso em 26 out. 2004.

GIL, Antonio Lourenço. **Segurança em informática**. 2. ed. São Paulo: Atlas, 1998. 193p.

JUCÁ, Humberto. **Implementação de firewall em linux**. 1. ed. Rio de Janeiro: Brasport, 2003. 122p.

KOKUBO, Eduardo. **Desenvolvimento de editor de artigos**. [S.I] [199-] Disponível em <http://www.cttmar.univali.br/pesquisa/projetos_det.php?id_projeto=12> Acesso em 1 jun. 2004.

MARCHAL, Benoît. **XML: conceitos e aplicações**. 1. ed. São Paulo: Berkeley Brasil, 2000. 548p.

MELO, Sandro; TRIGO, Clodonil Honorio. **Projeto de segurança em software livre**. 1. ed. Rio de Janeiro: Alta Books, 2004. 193p.

MOULTIS, Natanya Pitts; KIRK, Cheryl. **XML: black book**. 1. ed. São Paulo: MAKRON Books, 2000. 627p.

NACIONAL, Centro de Pesquisas. **Concepção do curriculum Lattes**. [S.I] [199-] Disponível em <http://www.uems.br/propp/pesquisa/lattes_info.htm> Acesso em 1 jun. 2004.

NARDON, Fabiane Bizinella. **Utilizando XML para representação de informação em saúde**. São Paulo, 2000. Disponível em: <<http://www.tridedalo.com.br/fabiane/publications/XML-SBISNews.pdf>> Acesso em: 1 jun. 2004.

NIELSEN, Jakob; TAHIR, Marie. **Homepage usabilidade: 50 websites desconstruídos**. 1. ed. Rio de Janeiro: Campus, 2002. 315p.

OLIVEIRA, Daniela Ferreira; BERNADI, Giliane. **Desenvolvimento de um mural eletrônico**. Santa Maria – RS, 2001. Disponível em <<http://www.inf.unifra.br/~giliane/Documentos/TFG/tfgdaniela.pdf>> Acesso em 1 ago. 2004.

PINHEIRO, Manuele Kirsch. **Mecanismo de suporte à percepção em ambientes cooperativos**. 2001. 167f. Dissertação (Mestrado em Ciência da Computação) - Universidade Federal do Rio Grande do Sul. Disponível em: <<http://www-lsr.imag.fr/Les.Personnes/Manuele.Kirsch-Pinheiro/MscKirschPinheiro.pdf>> Acesso em 26 out. 2004.

PROFFITT, Brian; ZUPAN, Ann. **XHTML: desenvolvimento Web**. 1. ed. São Paulo: MAKRON Books, 2001. 369p.

RAY, Erick. **Aprendendo XML**. 1. ed. Rio de Janeiro: Campus, 2001. 372p.

SEARA, Empresa. **Usabilidade e comunicação na Internet**. Portugal, 2004. Disponível em: <www.searasoft.com/downloads/ficheiros/usabilidade.pdf> Acesso em 06 out. 2004.

VELHO, Amir N. Ferreira; FIGUEIREDO, Júlio C. Bastos. **Conversão de exames laboratoriais armazenados em texto livre para XML**: uma aplicação para telefones celulares. São Paulo, [200-]. Disponível em: < http://www.avesta.com.br/so/so21_xml.pdf> Acesso em: 1 jun. 2004.

ZUQUIM, Diego G. Garcia; COSTA, Marco Rodrigo. **Desenvolvimento de uma aplicação para formatação de documentos de atividades acadêmicas**. 2004. Trabalho de Conclusão de Curso (Sistemas de Informação) – Pontifícia Universidade Católica de Minas Gerais.