

**MARCELO CAETANO MARTINS MUNIZ**

**PROPOSTA DE UM FRAMEWORK DE GERENCIAMENTO  
ELETRÔNICO DE DOCUMENTOS VIA WEB**

Monografia de Graduação apresentada ao Departamento de Ciência da Computação da Universidade Federal de Lavras como parte das exigências da disciplina Projeto Orientado, para a obtenção do título de Bacharel em Ciência da Computação.

Orientador

Prof. André Luiz Zambalde

LAVRAS  
MINAS GERAIS - BRASIL  
2002

**MARCELO CAETANO MARTINS MUNIZ**

**PROPOSTA DE UM FRAMEWORK DE GERENCIAMENTO  
ELETRÔNICO DE DOCUMENTOS VIA WEB**

Monografia de Graduação apresentada ao Departamento de Ciência da Computação da Universidade Federal de Lavras como parte das exigências da disciplina Projeto Orientado, para a obtenção do título de Bacharel em Ciência da Computação.

APROVADA em 04 de fevereiro de 2002

---

Profa. Olinda Nogueira Paes Cardoso

---

Prof. André Luiz Zambalde  
UFLA  
(Orientador)

LAVRAS  
MINAS GERAIS – BRASIL

## **DEDICATÓRIA**

Dedico este trabalho aos meus pais João Acássio e Célia por me darem condições e principalmente apoio em todos os momentos para que eu realizasse este sonho de me formar em computação. Peço a Deus que os ilumine e retribua toda dedicação que sempre tiveram por mim.

## **AGRADECIMENTOS**

Deixo registrado aqui os meus sinceros agradecimentos ao Prof. André Luiz Zambalde pelo auxílio neste projeto como também pela confiança depositada em mim em trabalhos realizados nestes quatro anos de faculdade.

Agradeço também aos professores Joaquim Quintero Uchôa e Olinda Nogueira Paes Cardoso pelas ajudas nas áreas de segurança e recuperação de informação, que foram de grande importância para que esse projeto se tornasse realidade.

## RESUMO

MUNIZ, Marcelo Caetano Martins. **Proposta de um Framework de Gerenciamento Eletrônico de Documentos via Web.** Lavras: UFLA, 2002. 34p. (Monografia - Graduação em Ciência da Computação) \*

Nas organizações, o gerenciamento de documentos é um problema interessante que pode ser solucionado com o uso de tecnologias da informação. O presente trabalho apresenta um *framework* (modelo) de gerenciamento eletrônico de documentos, associado com a idéia de criação e compartilhamento de documentos de forma cooperativa. O modelo proposto, chamado eDMS, procura resolver os problemas de armazenamento e criação colaborativa de documentos através de controle de acesso a usuários e controle de versão dos documentos. Ele foi desenvolvido para *internet*, observando critérios como facilidade e flexibilidade. Seu conceito de controle de revisão aceita criação concorrente de documentos e possibilita que os usuários possam acessar fases intermediárias da criação de seus documentos de forma simultânea. Ao final, é proposta a implementação preliminar da aplicação eDMS, que utiliza somente *softwares* gratuitos e independentes de plataforma.

---

Orientador: André Luiz Zambalde - DCC - UFLA.

## SUMÁRIO

RESUMO .....	v
LISTA DE FIGURAS .....	vii
1. Introdução .....	1
2. Referencial Teórico .....	4
2.1 Trabalho Colaborativo Apoiado por Computador .....	4
2.2 Gerenciamento Eletrônico de Documentos (GED).....	5
2.3 Sistemas de GED.....	6
3. Metodologia .....	8
3.1 Ambiente de trabalho .....	8
3.2 Tecnologias utilizadas.....	8
4. eDMS - Um Modelo de Gerenciamento Eletrônico de Documentos... 11	
4.1 Considerações Básicas .....	11
4.2 Módulo Administrador.....	12
4.3 Módulo Usuário.....	14
4.4 Documentos e Revisões .....	15
4.5 Controle de Versão.....	16
4.6 Histórico de Mudanças .....	18
4.7 Controle de Acesso .....	19
4.8 Busca de Documentos.....	21
4.9 Segurança.....	22
5. Conclusões .....	24
6. Bibliografia .....	25

## LISTA DE FIGURAS

Figura 1 - Descrição da arquitetura de 3 camadas em uma utilização de programa CGI .....	9
Figura 2 - Arquitetura do Sistema eDMS .....	12
Figura 3 - Interface do Módulo Administrador .....	13
Figura 4 - Interface do Módulo Usuário .....	14
Figura 5 - Árvore de Revisões.....	17
Figura 6 - Interface do Histórico do Documento.....	19

## 1. Introdução

Nas organizações, são raras as atividades que não geram informações que devam ser arquivadas sob a forma de papel, montados em pastas específicas, ocupando muito espaço, o que dificulta sua recuperação.

Estudos [SKAPINKER91] mostram que há dez anos, as empresas nos Estados Unidos gastavam mais de seis bilhões de dólares por ano em reimpressão de arquivos e aproximadamente 120 bilhões de dólares em distribuição, armazenamento e processamento de documentos. Outro grande problema para essas empresas era o espaço ocupado pelos arquivos.

Porém, com as transformações tecnológicas ocorridas nesta última década, as estruturas organizacionais foram alteradas radicalmente. A alta taxa de crescimento e popularidade da *Internet*, além da redução dos custos dos computadores e das conexões para coloca-los em rede, possibilitou o uso de tecnologias (ou condição financeira de investir em tecnologias) capazes de criar, armazenar e distribuir documentos de forma digital. A tecnologia da informação passou a ser não somente uma ferramenta de apoio ao trabalho dos usuários, mas principalmente um meio de comunicação e cooperação entre estes.

A possibilidade de trabalhar utilizando redes de computadores ou mesmo a *internet*, de forma colaborativa, tem levantado muito o interesse de cientistas que vêm pesquisando formas de criar ferramentas que gerenciem esta colaboração. Muitas vezes, documentos dentro de uma organização passam por alterações e geralmente essas alterações são realizadas por equipes, ou grupos de pessoas. Com a evolução da computação, esses documentos agora em forma digital, criados de forma colaborativa, podem ter seus ciclos de vida inteiramente gerenciados por computadores.

O documento digital possibilita uma riqueza de recursos muito superior à simples utilização de papel. Atualmente, é possível armazenar na forma digital, documentos com textos, imagens coloridas, sons, animações e vídeos.

Segundo [MEIER96], os documentos dentro de uma empresa representam funções de vital importância, principalmente como mecanismo de comunicação, veículo de processo para negócios e memória da organização. Por esses motivos, eles devem ser organizados adequadamente de forma a permitir acesso fácil e rápido. O termo utilizado para programas que tentam gerenciar o ciclo de vida dos documentos dentro de uma organização é Gerenciamento Eletrônico de Documentos (GED).

O presente trabalho busca a solução para os seguintes problemas relacionados ao GED: a) Como criar, armazenar, distribuir e organizar documentos numa organização de forma colaborativa, utilizando a *internet*? b) É possível criar uma ferramenta que realize esta tarefa utilizando somente tecnologias gratuitas?

Apresenta-se, portanto, um estudo sobre GED e propõe-se um *framework* (modelo) cooperativo baseado em ferramentas computacionais para auxiliar o gerenciamento de documentos. Este *framework* recebe a denominação eDMS – *electronic Document Management System*. O eDMS possui capacidade de gerenciamento de qualquer tipo de arquivo eletrônico, controle de acesso aos documentos com o mínimo de segurança, controle de versão dos arquivos, ferramenta de busca, interface de fácil interatividade e acesso via *internet*. Foi implementado utilizando tecnologias com licenças públicas, tais como Banco de Dados MySQL, Servidor Apache e Linguagem de Programação PHP, todos em ambiente multiplataforma.

A ferramenta deverá ter licença de utilização livre, baseada em GPL (*GNU General Public Licence*) [GNU01] e necessitará ser aperfeiçoada pela comunidade da *internet*.

Esta monografia esta estruturada em 6 capítulos. No primeiro capítulo encontra-se a introdução apresentando o problema de gerenciamento eletrônico de documentos e são definidos os objetivos deste trabalho. No segundo capítulo é apresentada uma cuidadosa pesquisa bibliográfica sobre trabalho colaborativo apoiado por computador e gerenciamento eletrônico de documentos. No terceiro capítulo é apresentada a metodologia utilizada para o desenvolvimento e solução do problema. O quarto capítulo apresenta o modelo eDMS, que é uma proposta para gerenciamento de documentos via *web*. O quinto capítulo apresenta as conclusões deste trabalho e as referências utilizadas na monografia estão apresentadas no sexto capítulo.

## **2. Referencial Teórico**

### **2.1 Trabalho Colaborativo Apoiado por Computador**

Trabalho colaborativo (ou cooperativo) é aquele em que várias pessoas, locais ou remotamente distribuídas, cooperam para a realização de uma mesma tarefa de forma síncrona ou assíncrona [PINHO99]. Sendo este tipo de atividade cada vez mais importante, diversos trabalhos têm sido realizados na área que hoje recebe o nome de Trabalho Cooperativo Apoiado por Computador, do inglês *Computer Supported Cooperative Work – CSCW*.

Segundo [NIELSEN96], CSCW define o que acontece com pessoas que se colaboram em alguma tarefa com ajuda de computadores (ligados em rede). Podem ser definidos como: sistemas baseados em computador que suportam grupos de pessoas engajado em uma tarefa ou objetivo comum e que fornecem uma interface para um ambiente compartilhado.

Em seu trabalho [KYNG91] divide sistemas CSCW em quatro classes: Sistemas de Mensagens, Sistemas de Conferência, Sistema de Coordenação e Sistemas de Autoria Colaborativa e Argumentação. A presente proposta (eDMS) se enquadra na classe de Sistemas de Autoria Colaborativa e Argumentação, que objetiva auxiliar a cooperação necessária entre autores na produção de documentos.

No mundo real o trabalho cooperativo envolve uma grande quantidade de troca de informações, nas mais variadas formas, seja entre indivíduos, entre grupos ou entre indivíduos e grupos. Assim, a possibilidade e, principalmente a facilidade no compartilhamento e na troca de informações é crucial para o sucesso de um ambiente CSCW [PINHO99].

Estas informações podem ter relação direta com os dados da aplicação ou podem ser informações de controle, que não carregam ou possuem um significado direto para a tarefa a ser realizada, mas atuam principalmente como forma de sincronizar os vários colaboradores para a execução correta da tarefa.

A troca de informações se faz necessária por motivos como: a) para que a informação em si possa ser compartilhada entre os usuários e assim de fato o trabalho ser colaborativo; b) para que as mudanças de estado, de interesse ou de forma de interação de um usuário, a partir de um dado instante, possam ser percebidas e assimiladas pelos demais usuários; c) para que a definição de papéis, decomposição de tarefas (quem faz o que e quando) possa ser realizada; d) para que os colaboradores saibam o que está sendo e o que já foi realizado no contexto de suas tarefas; e) para que o usuário ou um grupo ou parte deste possam passar da realização de tarefas individuais a colaborativas e vice-versa, sem que isto atrapalhe os demais usuários ou impeça seu trabalho [PINHO99].

## **2.2 Gerenciamento Eletrônico de Documentos (GED)**

Segundo [KOCH98] um sistema de GED utiliza a tecnologia da informática para captar, armazenar, localizar e gerenciar versões digitais das informações.

Enquanto editores de textos comuns já são capazes de gerenciar a criação de documentos em pequenos grupos, a edição compartilhada de grandes publicações, em geral corporativas, precisam de ferramentas mais poderosas, sistemas de GED capazes de auxiliar desde a criação até o controle de acesso à versão mais atualizada do documento.

O chamado “Ciclo de Vida do Documento” consiste das seguintes etapas [CHAFFEY98]: a) criação; b) revisão; c) editoração; d) aprovação; e) disponibilização; f) distribuição e controle de acesso.

Por essas etapas serem dinâmicas, organizar e manter este trabalho torna-se ainda mais difícil quando o mesmo está inserido em um trabalho de uma equipe.

O gerenciamento manual de documentos torna-se complicado à medida que a complexidade dos parâmetros que o norteiam aumentam. Por isso, busca-se um sistema que automatize esse gerenciamento, principalmente as atividades de segurança e controle de acesso, armazenamento e recuperação destes documentos.

### **2.3 Sistemas de GED**

Atualmente existem algumas ferramentas que se propõem a resolver o problema do GED, quando os arquivos são organizados em projetos técnicos, como WebRCS (*Web Revision Control System*) e SCCS (*Source Code Control System*). Estes tipos de sistemas funcionam perfeitamente para arquivos do tipo texto, como códigos de programas e são utilizados para projetos de engenharia.

O SCCS foi desenvolvido em 1972 por Marc Rochkind nos laboratórios da Bell. Por ser um dos primeiros sistemas para controle de código, não atende a algumas necessidades ou facilidades disponíveis hoje. No entanto, continua sendo bastante utilizado principalmente pelos usuários de BSD (Sistema Operacional UNIX da Bell) [BOLINGER95].

O WebRCS, desenvolvido em 2001 pelo Laboratório de Técnicas Inteligentes da Universidade de São Paulo, utiliza tecnologias de ponta, como

utilização de servidor Web e interface via navegador. Porém, este sistema utiliza o RCS (*Revision Control System* – Sistema de Controle de Revisão) como gerenciador de arquivos, funcionando como uma interface para *internet* deste sistema. Ele é uma ótima opção somente para quem quer trabalhar com arquivos textos [LACERDA01].

Nossa proposta é de um sistema mais genérico, que seja viável para qualquer documento eletrônico dentro de uma organização, desde arquivos textos a documentos multimídia.

Estudos para o desenvolvimento de sistemas de gerenciamento de qualquer tipo de documentos como em [FAN99], [GINSBURG00] e [BACKER96] já foram realizados, mas geralmente estes trabalhos enfocam ou formas eficientes de indexar e recuperar documentos, ou ainda estão desatualizados em relação às novas tecnologias da informação.

### **3. Metodologia**

#### **3.1 Ambiente de trabalho**

O presente trabalho foi desenvolvido no Departamento de Ciência da Computação da Universidade Federal de Lavras/MG – UFLA. Para o seu desenvolvimento, inicialmente foram realizados alguns estudos bibliográficos sobre trabalho cooperativo apoiado por computador e gerenciamento eletrônico de documentos. Posteriormente, buscou-se o aprendizado e estudo de ferramentas de licenças gratuitas, tais como PHP, MySQL, o servidor web Apache e tecnologias de segurança como o SSL (*Secure Sockets Layer*).

Após preparo teórico e técnico, decidiu-se pela implementação do sistema de gerenciamento eletrônico de documentos eDMS. Para sua implementação foram utilizados computadores do DCC/UFLA, rodando o Sistema Operacional Linux.

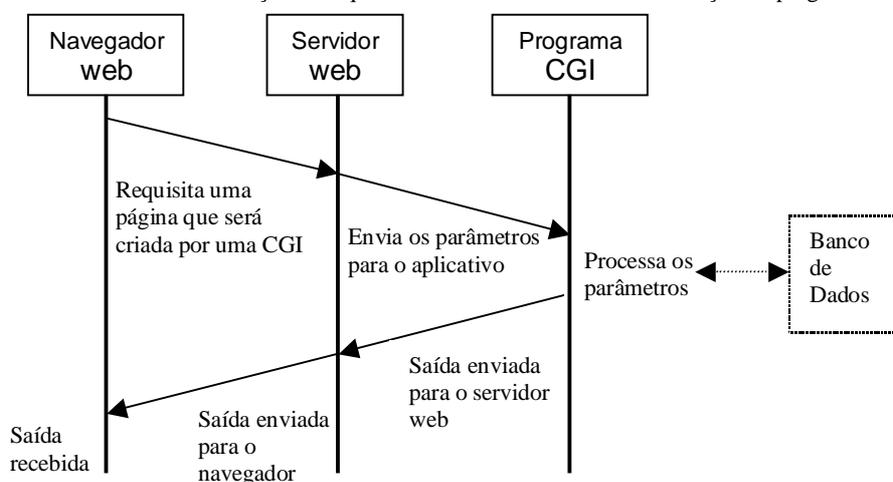
#### **3.2 Tecnologias utilizadas**

Por se tratar de um sistema cliente/servidor que utiliza *internet*, a ferramenta eDMS esta baseada em: servidor *web* para páginas HTML, banco de dados e software básico de interface entre o servidor *web* e o banco de dados.

Para servidor *web* foi escolhido o Apache. A escolha foi baseada no fato deste servidor ser de código fonte aberto e ter licença livre. De acordo com o Netcraft [NETCRAFT01] o Apache é o mais popular servidor *web* da *internet*, sendo utilizado em novembro de 2001 por 61.88 % dos *sites* ativos no mundo.

Como solução de *software* para interface entre servidor *web* e banco de dados foi escolhida a utilização de CGI (*Common Gateway Interface*). Os CGIs são programas que podem ser escritos em diversas linguagens de programação. Eles são chamados pelo servidor *web* para executar processos que devolvem como saída páginas HTML. Este procedimento, caracterizado por 3 camadas, encontra-se esquematizado na FIGURA 1.

FIGURA 1 - Descrição da arquitetura de 3 camadas em uma utilização de programa CGI.



Segundo [ROWE98] o processo de execução de um CGI é caracterizado pelos seguintes passos: a) o usuário chama um programa CGI dando um clique em um *link* ou pressionando um botão; b) o navegador *web* entra em contato com o servidor, solicitando permissão para rodar o programa CGI; c) o servidor *web* verifica a configuração e acessa os arquivos para garantir que o solicitante possa ter acesso ao programa CGI; d) o servidor *web* verifica se o CGI existe; e) se ele existe, o programa CGI é executado; f) a saída produzida pelo programa CGI é retornada para o navegador *web*; g) o navegador *web* apresenta a saída produzida.

Existem muitas linguagens disponíveis para escrever programas CGI, entre elas C, Shell Script, Perl e PHP. A linguagem escolhida para ser utilizada em nossa implementação foi a PHP (PHP: *Hipertext Processor*), pois além de ser gratuita e multiplataforma, ela possui conectividades com vários gerenciadores de banco de dados e dá suporte a funções para trabalhar com arquivos.

Para banco de dados, a solução escolhida foi o MySQL, um banco de dados relacional que utiliza a linguagem SQL (*Structured Query Language*) para consulta. Os fatores que levaram a essa escolha foram: a) é uma tecnologia gratuita que utiliza a licença GPL [GNU01]; b) o PHP possui um módulo que aceita comunicação com o MySQL; c) é uma tecnologia multiplataforma; d) possui um bom desempenho em relação a outros bancos de dados [MYSQL02].

Como o eDMS foi proposto para ser visualizado em navegadores *web*, sua interface foi desenvolvida na linguagem HTML versão 4.01 (*HiperText Markup Language*) [HTML99] e posteriormente portada para PHP.

Por ser um sistema que será distribuído gratuitamente, com licença GPL [GNU01], optou-se desenvolver a interface do sistema em língua inglesa, para que o mesmo seja aceito por toda a comunidade da *internet*.

## **4. eDMS - Um Modelo de Gerenciamento Eletrônico de Documentos**

### **4.1 Considerações Básicas**

O modelo eDMS foi proposto para dar suporte ao gerenciamento eletrônico de documentos em organizações, associado com a idéia de criação e compartilhamento de documentos de forma cooperativa. Em particular, ele foi desenvolvido visando trabalhar com mobilidade em organizações onde seus funcionários não têm locais fixos de trabalho, utilizando uma arquitetura cliente/servidor. O eDMS deverá ser usado para suportar grupos de trabalhos pequenos e médios, e mesmo para trabalhos individuais como um simples gerenciador de arquivos.

Este modelo trabalha o problema de colaboração e criação de documentos realizando o controle de acesso dos usuários e gerenciando suas revisões. Em seu ciclo de vida, o documento é criado e sofrerá alterações, cada vez arquivando uma nova revisão. O histórico das revisões e as próprias revisões serão disponibilizados a todos usuários que tiverem acesso a esse documento. Isso dará aos usuários a chance de acessar qualquer estágio intermediário de criação do documento, como a oportunidade de preparar novas revisões simultaneamente a outros usuários.

Este sistema é dividido em dois módulos: módulo administrador e módulo usuário. Sua arquitetura pode ser vista na FIGURA 2.

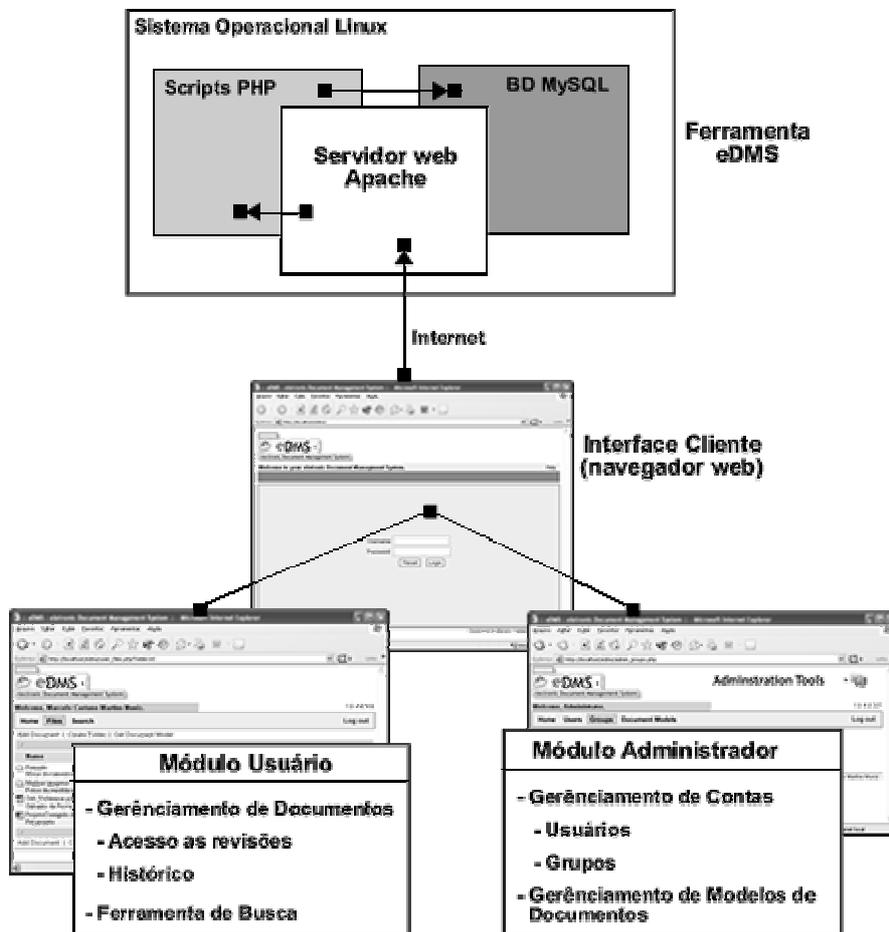


FIGURA 2 – Arquitetura do Sistema eDMS.

#### 4.2 Módulo Administrador

O módulo administrador (FIGURA 3) tem como principais funções administrar as contas e grupos de usuários e gerenciar os modelos de documentos.

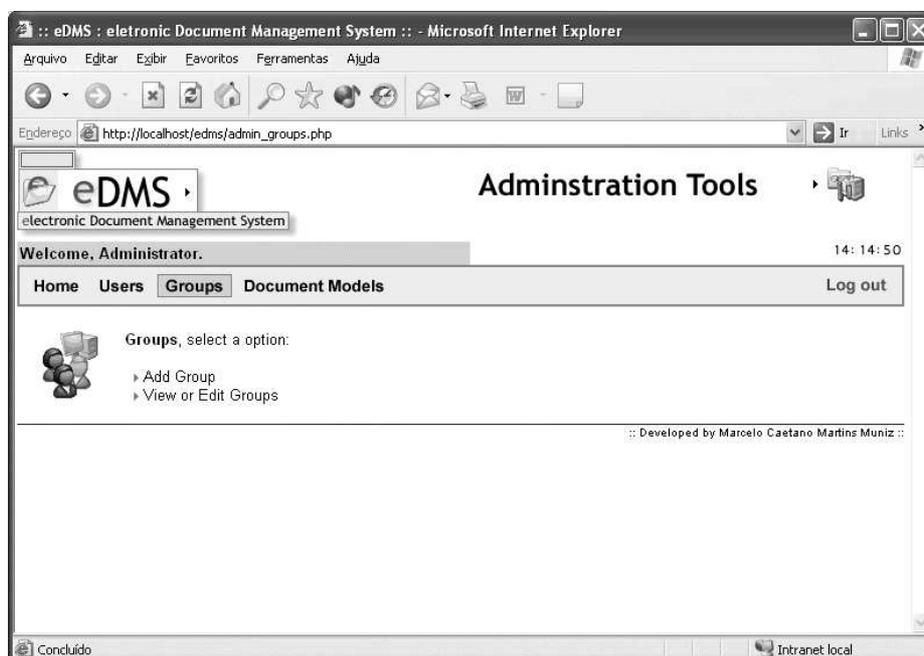


FIGURA 3 - Interface do Módulo Administrador.

O administrador do sistema poderá criar grupos de usuários e adicionar usuários a esses grupos, como também editar e apagar as informações destes usuários e grupos.

O usuário administrador sempre pertencerá ao grupo *Admin*, grupo o qual nunca poderá ser apagado do sistema. Todo usuário pertencente ao grupo *Admin* será visto pelo sistema como um administrador e terá acesso exclusivo ao módulo administrador, mas não terá acesso ao módulo usuário.

Este *framework* utiliza o conceito de uso de modelos de documentos (exemplos), o qual será gerenciado por usuários administradores. O uso de modelos de documentos visa padronizar o *layout* e a estrutura dos documentos dentro de uma organização. Estes documentos podem ser desde modelos de ofícios a formulários frequentemente utilizados pelos usuários da organização.

Cabe ao administrador adicionar e remover estes modelos de documentos. Os modelos adicionado ao sistema serão visíveis a todos usuários. Os usuários poderão escolher se vão ou não utilizar estes modelos.

### 4.3 Módulo Usuário

Este módulo (FIGURA 4) tem como finalidade disponibilizar a todos usuários não pertencentes ao grupo *Admin* o gerenciamento de seus documentos e aqueles os quais ele possa acessar, dando suporte a colaboração entre usuários através de controles de acesso.

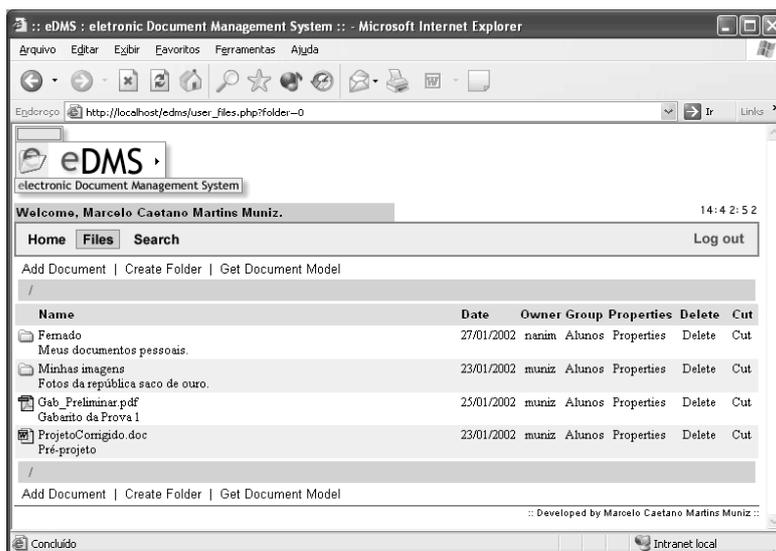


FIGURA 4 - Interface do Módulo Usuário.

O módulo usuário possui também uma ferramenta de consulta, onde o usuário pode procurar, através de palavras chaves que são cadastradas com o

documento, por documentos que ele tenha acesso ou que sejam de acesso público.

#### **4.4 Documentos e Revisões**

No eDMS, os termos documento e arquivo não são vistos com o mesmo significado. No sistema, cada documento cadastrado terá associado a ele um conjunto de arquivos contendo suas revisões.

O termo documento, na realidade, representará um arquivo cadastrado pelo usuário, mas fisicamente só serão gravadas as revisões. O processo é caracterizado da seguinte forma: o usuário adicionará um novo documento ao sistema informando suas características e enviando um arquivo. Este será gravado no servidor de arquivos como a primeira revisão do documento cadastrado. Os arquivos contendo suas revisões permanecerão disponíveis a todos usuários que tiverem acesso ao documento, que será visto como uma pasta que contém as revisões.

Cabe ressaltar que o eDMS aceita qualquer tipo de arquivo digital.

O sistema funcionará num servidor de *internet* e os usuários terão acesso a ele por navegadores *web*. Todos arquivos serão enviados e recebidos pela *internet*.

As revisões existentes sempre permanecerão com o acesso de leitura, isto é, poderão ser copiadas para o disco local, mas nunca poderão ser alteradas no servidor. Modificações para um documento somente poderão ser feitas com o envio de novas revisões. Este esquema evita o problema de perda de informação causado por trabalho concorrente numa mesma revisão de um documento,

impedindo que arquivos sejam sobrescritos. Uma outra vantagem é que isso permite que vários usuários leiam simultaneamente uma mesma revisão.

O eDMS organiza os documentos de maneira similar aos sistemas operacionais Windows e Linux, utilizando a idéia de árvore de diretórios. Deste modo, o usuário pode criar pastas e subpastas e adicionar arquivos a essas pastas.

#### **4.5 Controle de Versão**

Quando se trabalha com aplicações onde o processo de desenvolvimento é dinâmico, necessita-se de um registro que possa identificar univocamente a descrição de um documento que evolui com o tempo. Essa identificação é chamada versão. Modificações dessa versão são denominadas revisões [LACERDA01]. O eDMS utiliza o conceito de versão, onde o número da revisão aumenta durante o processo de colaboração nas atualizações do documento.

A numeração das revisões utilizada no eDMS foi inspirada na numeração utilizada no sistema de controle de versão CVS (*Concurrent Versions System* – Sistema de Versões Concomitantes) [CVS93], pois a mesma suporta revisões concorrentes.

Ela funciona da seguinte maneira. Cada versão de um documento tem um único número de revisão. Os números de revisões podem ser ‘1.1’, ‘1.2’, ‘1.3.2.2’ ou até ‘1.3.2.2.4.5’. Por padrão, a revisão 1.1 é a primeira revisão do documento. Para cada nova revisão será dado um novo número de revisão, aumentando o número mais à direita da revisão anterior em um.

Este sistema suporta revisões concorrentes, isto é, mais de uma nova revisão a partir de uma mesma revisão velha (arquivo pai). Utilizando o sistema de ramificação de revisões.

Geralmente, os números de revisão são uma série de incrementos, como '1.1', '1.2', '1.3', '1.4' e '1.5'. O número de revisão é sempre formado por um número par de números inteiros. Por exemplo, o número '1.3' significa que esta revisão pertence ao ramo principal (ramo 1) e corresponde a terceira revisão.

Porém, como o modelo suporta revisões concorrentes, ele não estará limitado a um desenvolvimento linear. A árvore de revisões (FIGURA 5) pode ser dividida em ramificações, tendo cada ramo uma linha própria de desenvolvimento.

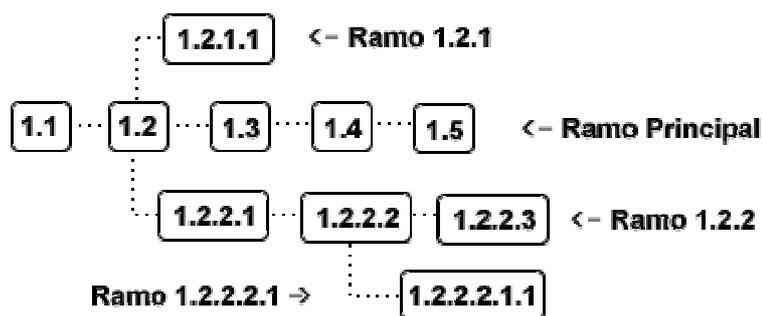


FIGURA 5 – Árvore de Revisões

Ao enviar uma nova revisão o usuário deverá informar, qual revisão anterior (arquivo pai) gerou esta nova revisão. Caso não exista um arquivo de revisão gerado pelo mesmo arquivo pai, somente somamos em um ao número mais à direita da revisão do arquivo pai. Caso já exista um arquivo de revisão gerado pelo mesmo arquivo pai, um novo ramo será criado para esta nova revisão. Este número será o número de revisão do arquivo pai mais um par de números, gerando o número do novo ramo e o número da revisão. Por exemplo,

caso o usuário crie uma nova revisão a partir da revisão '1.3'. Se já existe no sistema uma revisão com número '1.4', o número desta nova revisão será o número do arquivo pai, isto é, '1.3' mais o número do novo ramo e o número da revisão. Como não existe nenhuma revisão com ramo '1.3.1', será criado este ramo e o número desta nova revisão será '1.3.1.1'. Se já existisse um ramo '1.3.1', seria criado um novo ramo com o número '1.3.2' e o número da revisão seria '1.3.2.1'.

Existirão dois tipos de revisão: a preliminar e a definitiva. Na revisão preliminar serão inseridas novas revisões de uma versão (ramo principal). Ao realizar uma revisão definitiva, o usuário está criando uma nova versão para o documento. O número de uma nova versão é gerado somando em um ao número mais à esquerda do número de revisão do arquivo pai. Se o número de revisão do arquivo pai for '1.5.2.3' e a revisão enviada for uma revisão definitiva, o número de revisão deste arquivo será '2.1', indicando que o ramo principal agora é 2 e este arquivo é a primeira revisão. O usuário somente poderá criar uma nova versão a partir de arquivos pais que pertençam à versão mais atual. Por exemplo, se a versão mais atual é a do ramo 3. Não é possível criar uma versão 4, a partir de um arquivo pai que pertença a versão 1.

#### **4.6 Histórico de Mudanças**

Existem casos onde os usuários querem obter mais detalhes sobre o que foi modificado em um documento. Para estes casos, foi modelado o histórico de mudanças (FIGURA 6) para todas revisões do documento.

Ao enviar uma nova revisão, o usuário preencherá um formulário informando quais foram as modificações realizadas nesta nova revisão. Com isto, será possível ter um histórico que mostra a partir de quais revisões a revisão

atual foi criada, o que foi modificado em cada revisão e quem realizou estas modificações, permitindo de forma clara identificar a evolução do documento.

Document History: ProjetoOrientado.doc			
Revision 1.1	Date: 01/02/2002 13:12:30	Revisor: muniz	Size: 235.1 Kb
Primeira versão do Projeto Orientado.			
Revision 1.2	Date: 03/02/2002 08:00:01	Revisor: zambalde	Size: 245.2 Kb
Foi corrigido a introdução e adicionado o capítulo de referencial teórico.			
Revision 1.3	Date: 03/02/2002 16:21:10	Revisor: muniz	Size: 248.0 Kb
Foi corrigido o capítulo de referencial teórico e todo texto foi formatado no padrão de monografias da UFLA.			

:: Developed by Marcelo Caetano Martins Muniz ::

FIGURA 6 – Interface do Histórico do Documento

#### 4.7 Controle de Acesso

O controle de acesso do modelo eDMS é baseado em políticas de acessos a grupos e usuários. Existem quatro tipos de acessos e os acessos são aplicados a pastas e a documentos.

Quando um novo usuário é adicionado ao sistema por um administrador, é definido a qual grupo o usuário pertence. Sempre um usuário pertencerá a um grupo.

Quando o usuário cria uma pasta, ele poderá escolher qual o grupo de usuários terá acesso a esta pasta o tipo de acesso. Os demais grupos não terão acesso nenhum a esta pasta. Somente um grupo poderá ser definido para ter acesso a uma pasta. Os tipos de acesso possíveis para uma pasta são:

- a) **Nenhum acesso**: esta pasta não será vista por usuários do grupo selecionado;
- b) **Leitura**: os usuários deste grupo enxergarão esta pasta e poderão enxergar os documentos dentro dela;
- c) **Escrita**: os usuários deste grupo enxergarão esta pasta e poderão criar subpastas e documentos, mas não poderão remove-los;
- d) **Super**: Os usuários deste grupo enxergarão esta pasta, poderão criar subpastas

e documentos, bem como removê-los, além de ter acesso total a edição das propriedades da pasta.

Sempre o dono da pasta será o mesmo do diretório superior. Se um usuário tiver acesso para criar uma subpasta em uma pasta, ele pode criar, mas não será o dono da subpasta, a menos que ele seja o dono do pasta. Caso ele não seja o dono da pasta, ele terá somente acesso de *Escrita* na subpasta criada. Todo dono de uma pasta tem acesso *Super* sobre ela.

Além de dar acesso a um grupo em uma pasta, poderá ser adicionado acesso a qualquer usuário. Cada pasta terá uma lista de acesso formada por um grupo e usuários adicionados separadamente. Os tipos de acesso que os usuários podem ter são os mesmos que um grupo pode ter. Somente usuários com tipo de acesso *Super* poderão adicionar outros usuário a lista de acesso de uma pasta.

O controle de acesso aos documentos é muito parecido com o controle de acesso das pastas. Somente poderão adicionar novos documentos ao sistema os usuários que tenham pelo menos acesso de escrita na pasta em que estão.

Ao adicionar um novo documento ao sistema, o usuário informará qual grupo de usuários terá acesso ao documento e que tipo de acesso ele terá, da mesma forma que o controle de acesso às pastas. O que muda é o significado dos tipos de acesso possíveis. Da mesma forma que nas pastas eles são quatro: a) *Nenhum acesso*: o grupo não enxergará o documento; b) *Leitura*: o grupo enxergará o documento e poderá copiar as revisões deste documento, mas não poderão enviar novas revisões; c) *Escrita*: o grupo enxergará o documento, poderá copiar e enviar novas revisões a este documento, porém somente revisões preliminares; d) *Super*: este grupo terá acesso total ao documento, podendo até enviar revisões definitivas além de poder remover este documento.

Da mesma forma que nas pastas, usuários poderão ser adicionados separadamente a lista de acesso a um documento. Os tipos de acesso que os

usuários podem ter aos documentos são os mesmos que um grupo pode ter. Todo dono de um documento tem acesso *Super* sobre este documento.

No caso de um usuário estiver numa lista de acesso a uma pasta ou documento e ele pertencer ao grupo de acesso desta pasta ou documento, o sistema levará em consideração o maior acesso. Exemplo: o usuário **Marcelo** pertence ao grupo Alunos e o acesso do grupo Alunos a uma determinada pasta é de *Escrita*. Se o usuário Marcelo for adicionado a lista de acesso desta pasta com acesso de *Leitura*, o sistema não levará em consideração o acesso do usuário e sim o do grupo. Se o usuário **Marcelo** for adicionado a lista de acesso desta pasta com acesso de *Super*, o sistema levará em consideração o acesso do usuário e não o do grupo.

#### **4.8 Busca de Documentos**

Um elemento importante dentro de um sistema de GED é a forma de recuperação dos documentos. Para isto, o modelo eDMS sugere um sistema de consulta baseado em catalogação dos documentos por palavras chaves e utiliza o conceito de documento público.

Ao se criar um documento, o usuário poderá optar por tornar este documento público, isto é, qualquer pessoa que utilizar o sistema de busca poderá ter acesso de leitura a esse documento. Isto é útil para tornar o servidor de arquivos em biblioteca virtual.

A consulta realizada por um usuário sempre abrangerá os documentos que ele tiver acesso e os documentos públicos.

## 4.9 Segurança

A *internet* é um meio de transmissão de dados público, por isso, alguns cuidados foram tomados para garantir segurança ao sistema eDMS. Foi implementado no sistema o controle de seção, a utilização do protocolo SSL (*Secure Socket Layer*) e o uso de criptografia de senha pelo MySQL.

O administrador do sistema é o responsável por cadastrar os usuários na ferramenta. Ao cadastrar um usuário, o administrador deve adicionar um nome de usuário (*username*) e uma senha (*password*) ao usuário. Este *password* será gravado no banco de dados de forma criptografada utilizando a função `PASSWORD()` do MySQL. Esta função criptografa o *password* do usuário de maneira não reversiva.

Para o usuário ter acesso a sua interface no sistema, ele deverá entrar no sistema fornecendo seu *username* e *password*. Porém, para garantir que este *password* saia de seu navegador e chegue com segurança ao servidor *web*, o Apache foi configurado para utilizar o módulo `mod_SSL`.

O `mod_SSL` fornece ao Apache a utilização do protocolo SSL. Este protocolo age entre a camada de rede (TCP/IP) e a camada de aplicação (HTTP), fornecendo comunicação segura entre o cliente e o servidor. O SSL fornece autenticação mútua, uso de assinaturas digitais para integridade e encriptação para privacidade [ENGELSCHALL01]. Utilizando esta configuração, o usuário somente conseguirá acessar o sistema via HTTPS no navegador *web* e garantirá que todos dados transmitidos entre o navegador e o servidor *web* estarão criptografados [HTTPS96].

O eDMS utiliza também, o controle de seção, isto é, ao enviar o *username* e *password* ao servidor, será verificado no banco de dados se o usuário está cadastrado e se o *password* está correto. Se isto ocorrer, o usuário é

autenticado, e o sistema inicia uma sessão para o usuário, gravando um *cookie* no cliente com um identificador de sessão e horário da sessão. Os mesmos dados são gravados no banco de dados do servidor numa tabela de autenticação. Este sistema garante que o usuário passou pela interface de autenticação do eDMS.

Toda vez que um usuário tentar entrar em qualquer página do sistema, serão verificados se os dados do *cookie* gravados em sua máquina são os mesmos da tabela de autenticação no servidor. Se os dados forem os mesmos, o usuário terá acesso a página e o horário da sua sessão será atualizado, caso contrário o usuário terá que passar pela interface de autenticação do sistema.

Este sistema de sessão foi configurado para garantir a desconexão do usuário caso ele fique por mais de 10 minutos sem realizar nenhuma operação.

## 5. Conclusões

O presente trabalho apresentou um modelo de GED, denominado eDMS, para criar, armazenar, distribuir e organizar documento, de forma colaborativa, via *internet*, utilizando *software* livre.

O uso desta ferramenta trás benefícios como: a) centralização de arquivos em um único servidor; b) facilidade, confiabilidade e segurança no trabalho em grupo e criação de documentos; c) economia de tempo na busca e no processo de criação de documentos; d) mobilidade para trabalho em grupo com a utilização da *internet*.

Com a utilização do eDMS tem-se todo o histórico de um documento – de sua criação à versão final. É possível saber o que foi modificado, quando foi modificado e por quem foi modificado o documento.

O conceito do modelo eDMS pode ser facilmente incorporado a ferramentas de *internet* que auxiliem trabalho cooperativo como WebAgendas executivas, Ferramentas de ensino a distância e/ou sites de WebMail que trabalhem com grupos de usuários.

Em futuras implementações, pretende-se adicionar ferramentas de comunicação entre os usuários para melhorar o trabalho cooperativo. Outro problema encontrado foi o espaço ocupado pelos documentos no servidor, já que todas revisões dos documentos permanecem gravadas. Espera-se em novas implementações a adição de técnicas de compactação para estes arquivos.

## 6. Bibliografia

- [BACKER96] Backer, A.; Busbach, U. **DocMan: A Document Management System for Cooperation Support**. 29<sup>th</sup> Annual Hawaii International Conference on System Sciences. IEEE. p. 82-91. 1996.
- [BOLINGER95] Bolinger, D. & Bronson T. **Applying RCS and SCCS**. O'Reilly, 1995.
- [CHAFFEY98] Chaffey, D. **Groupware, Workflow and Intranets**. Ed. Digital Press. 1998.
- [CVS93] Cederqvist. **Version Management with CVS**. Signum Support AB. 176 p. 1993. Disponível em WWW em <http://www.cvshome.org/>
- [ENGELSCHALL01] Engelschall, R. S. **mod\_SSL 2.8, User Manual**. Disponível em WWW em <http://www.modssl.org/>
- [FAN99] Fan, X.; Sheng, F.; Ng, P. A. **DOCPROS: A Knowledge-Based Personal Document Management System**. 1999.
- [GINSBURG00] Ginsburg, Mark. **Intranet Document Management System as Knowledge Ecologies**. 33<sup>rd</sup> Annual Hawaii International Conference on System Sciences. 2000.

[GNU01] Projeto GNU. **GNU's NOT UNIX**. 2001. Disponível em WWW em <http://www.gnu.org>

[HTML99] WC3 Recommendations. **HTML 4.01 Specification**. 1999. Disponível em WWW em <http://www.w3.org/TR/1999/REC-html401-19991224>

[HTTPS96] Internet Draft. **The Secure HyperText Transfer Protocol**. Disponível em WWW em <http://www.terisa.com/shhttp/1.2.1.txt>

[KYNG91] Kyng, M. **Design for Cooperation: Cooperation in Design**. Communication of the ACM. Vol. 7, n. 4, 1991.

[KOCH98] Koch, W. **Gerenciamento Eletrônico de Documentos – Conceitos, Tecnologias e Considerações**. Cenadem. 146 p. 1998.

[LACERDA01] Lacerda, D. S.; Gomi, E. S. **WebRCS – Um Sistema de Gerenciamento de Documentos via Web**. 2001.

[MEIER96] Meier, J., Sprague, R. **Towards a Better Understanding of Electronic Document Management**. 29<sup>th</sup> Annual Hawaii International Conference on System Sciences. IEEE. p. 53-61. 1996.

[MYSQL02] MySQL AB. **MySQL Benchmarks**. 2002. Disponível por WWW em <http://www.mysql.com/information/benchmarks.html>

[NETCRAFT01] Netcraft. **Netcraft Web Server Survey**. 2001. Disponível em WWW em <http://www.netcraft.com/survey/>

[NIELSEN96] Nielsen, J. **Multimedia e Hypermedia - The internet and Beyond**, Academic Press Inc., 1996.

[PINHO99] Pinho, M. S. **Interação em Ambientes de Trabalho Cooperativo**. Trabalho Individual – TI 540, CPGCC, UFRGS, Ago 1999.

[ROWE98] Rowe, Jeff. **Construindo Servidores de Banco de Dados Internet com CGI**. Makron Books. São Paulo. 360p. 1998.

[SKAPINKER91] Skapinker, M. **Warm for Forms**. BYTE, 166 p. 1991.