

UNIVERSIDADE REGIONAL DE BLUMENAU
CENTRO DE CIÊNCIAS EXATAS E NATURAIS
CURSO DE SISTEMAS DE INFORMAÇÃO – BACHARELADO

DESENVOLVIMENTO DE UM SISTEMA DE
GERENCIAMENTO ELETRÔNICO DE DOCUMENTOS NA
SECRETARIA DA EDUCAÇÃO E CULTURA DE POMERODE

JEAN WILHELM KLEMMANN

BLUMENAU
2006

2006/2-07

JEAN WILHELM KLEMANN

**DESENVOLVIMENTO DE UM SISTEMA DE
GERENCIAMENTO ELETRÔNICO DE DOCUMENTOS NA
SECRETARIA DA EDUCAÇÃO E CULTURA DE POMERODE**

Trabalho de Conclusão de Curso submetido à
Universidade Regional de Blumenau para a
obtenção dos créditos na disciplina Trabalho
de Conclusão de Curso II do curso de Sistemas
de Informação - Bacharelado.

Prof. Wilson Pedro Carli, Mestre - Orientador

**BLUMENAU
2006**

2006/2-07

**DESENVOLVIMENTO DE UM SISTEMA DE
GERENCIAMENTO ELETRÔNICO DE DOCUMENTOS NA
SECRETARIA DA EDUCAÇÃO E CULTURA DE POMERODE**

Por

JEAN WILHELM KLEMANN

Trabalho aprovado para obtenção dos créditos na disciplina de Trabalho de Conclusão de Curso II, pela banca examinadora formada por:

Presidente: _____
Prof. Wilson Pedro Carli, Mestre – Orientador, FURB

Membro: _____
Prof. Evaristo Baptista, Mestre – FURB

Membro: _____
Prof. Paulo Roberto Dias, Mestre – FURB

Blumenau, 07 de Dezembro de 2006

Dedico este trabalho à minha família, à minha namorada e a todos os amigos, especialmente aqueles que me ajudaram diretamente na realização deste.

AGRADECIMENTOS

À Deus, pelo seu imenso amor e graça.

À minha família e à minha namorada, que mesmo longe, sempre estiveram presentes.

Aos meus amigos, pelos empurrões e cobranças.

Ao meu orientador, Wilson Pedro Carli, por ter acreditado na conclusão deste trabalho.

Os homens perdem a saúde para juntar dinheiro, depois perdem o dinheiro para recuperar a saúde. E por pensarem ansiosamente no futuro, esquecem do presente de tal forma que acabam por não viver nem o presente nem o futuro. E vivem como se nunca fossem morrer... e morrem como se nunca tivessem vivido.

Dalai Lama

RESUMO

Este trabalho teve como objetivo o desenvolvimento de um sistema de Gerenciamento Eletrônico de Documentos para a Secretaria da Educação e Cultura de Pomerode. O aplicativo desenvolvido em Active Server Pages permite cadastrar documentos e imagens escanizadas, que serão armazenadas em um banco de dados Access e que poderão ser recuperados remotamente pelos usuários, visando fornecer um mecanismo de disseminação de informações via web.

Palavras-chave: Gestão do conhecimento. GED. Sistemas de informação.

ABSTRACT

This work had a objective the development of a Electronic Document Management System for the Secretaria da Educação e Cultura from Pomerode. The application, developed in Active Server Pages, allows cadastre scanning documents and images and which will stored in an Access database and can be followed remotely by users, aiming at to supply a mechanism of dissemination of information via web.

Key-words: Management of the knowledge. EDMS. Information Systems

LISTA DE ILUSTRAÇÕES

Quadro 1: Comparação entre atividades de recuperação de documentos em papel e GED	15
Figura 1: Ciclo de vida de documentos	16
Figura 2: Ambiente típico.....	16
Quadro 2: Requisitos Funcionais.....	21
Quadro 3: Requisitos não funcionais.....	21
Figura 3: Diagrama de casos de uso	22
Figura 4: Diagrama de classes	23
Figura 5: Diagrama de atividades.....	24
Figura 6: Modelo entidade relacionamento (MER).....	25
Figura 7: Processamento de uma solicitação ASP	27
Quadro 4: Exemplo do código do login em ASP	28
Figura 8: Tela inicial do aplicativo.....	30
Figura 9: Tela principal do sistema	30
Figura 10: Tela de cadastro de usuários	31
Figura 11: Tipos de documentos	32
Figura 12: Cadastro dos documentos.....	33
Figura 13: Relatório de documentos cadastrados	34
Figura 14: Tela de pesquisa de documentos.....	34
Figura 15: Resultado da pesquisa	35

SUMÁRIO

1 INTRODUÇÃO	11
1.1 OBJETIVOS DO TRABALHO	12
1.2 ESTRUTURA DO TRABALHO	12
2 FUNDAMENTAÇÃO TEÓRICA	14
2.1 GERENCIAMENTO ELETRÔNICO DE DOCUMENTOS	14
2.2 A SECRETARIA DA EDUCAÇÃO E CULTURA DE POMERODE.....	17
2.3 TRABALHOS CORRELATOS	18
3 DESENVOLVIMENTO DO TRABALHO	20
3.1 REQUISITOS PRINCIPAIS DO PROBLEMA A SER TRABALHADO.....	20
3.1.1 Requisitos.....	20
3.2 ESPECIFICAÇÃO	22
3.2.1 Casos de uso.....	22
3.2.2 Diagrama de classes	23
3.2.3 Diagrama de atividades	24
3.2.4 Modelo entidade relacionamento	25
3.3 IMPLEMENTAÇÃO	26
3.3.1 Técnicas e ferramentas utilizadas.....	26
3.3.1.1 ASP	26
3.3.1.2 Banco de Dados	28
3.3.1.3 Modelo cliente-servidor.....	29
3.3.2 Operacionalidade da implementação	29
3.4 RESULTADOS E DISCUSSÃO	35
4 CONCLUSÕES	36
4.1 EXTENSÕES	36
REFERÊNCIAS BIBLIOGRÁFICAS	37
APÊNDICE A – Página inicial – index.html	39
APÊNDICE B – Página principal – menu.asp	40
APÊNDICE C – Página de cadastro de usuários – inserir_usuarios.asp	42
APÊNDICE D – Página de listagem de tipos – listar_tipos.asp	44
APÊNDICE E – Página de cadastro de documentos – cadastrar_documentos.asp	46
APÊNDICE F – Página de relatório – listar_documentos_paginado.asp	48

APÊNDICE G – Página de busca – buscar_documentos.asp.....	51
APÊNDICE H – Página do resultado da busca – buscando_documentos.asp.....	53

1 INTRODUÇÃO

“Com o passar dos anos a produção de informação, cresce de maneira cada vez mais acelerada. Nos últimos 50 anos a humanidade produziu uma quantidade de informação equivalente ao produzido nos mil anos anteriores.” (BACK, 2004, p. 13). Conforme Koch (1998 apud FANTINI, 2001, p. 6), “não vai parar por aí, pois estudos indicam que este número duplicará novamente entre os próximos 18 a 36 meses.” Segundo levantamento do Centro Nacional de Desenvolvimento de Gerenciamento da Informação (CENADEM), “em 2010 a informação duplicará a cada 11 horas.” (CENADEM, 2006)

A velocidade com que o conhecimento é gerado e compartilhado caracteriza a sociedade da informação. Gerenciar a produção e administrar a disponibilização do conhecimento produzido passou a ser fundamental para as empresas que vivem basicamente do conhecimento que possuem (BACK, 2004).

Um sistema de Gerenciamento Eletrônico de Documentos (GED) é um sistema de informação que armazena, recupera e mantém a integridade de documentos (MACEDO, 2003). O GED transforma documentos em formatos digitais, arquivando-os de maneira eficiente, com um alto grau de compactação e gerando índices eficientes para pesquisas (FANTINI, 2001). “O GED compreende todas as tecnologias empregadas para o controle digital de documentos, seja originalmente em papel, microfilme, som, imagem ou um arquivo digital.” (BACK, 2004, p. 31).

É normal imaginar que hoje em dia qualquer um que trabalhe de maneira interativa com o computador gere documentos que precisam ser gerenciados para que possam ser recuperados no futuro. Normalmente, para este tipo de atividade usamos o Explorer do Windows. Mas em vez de somente seus documentos, imagine milhares (ou milhões) de documentos nesta situação, que precisam ser gerenciados por uma forma comum de pesquisa, com segurança, restrição de acesso e com possibilidade de compartilhar essas informações com outros processos e sistemas. Não é possível imaginar isso pelo simples uso de um Explorer. (BALDAM; VALLE; CAVALCANTI, 2002, p. 33).

Pensar na quantidade de documentos utilizados há cinco anos atrás e a quantidade utilizada atualmente, passar mais tempo procurando um documento do que criá-lo novamente ou desistir de procurar um documento e encontrá-lo depois de ter feito todo o trabalho são fatos muito comuns em empresas (BALDAM; VALLE; CAVALCANTI, 2002).

Observando-se a comunicação entre a Secretaria da Educação e Cultura da Prefeitura Municipal de Pomerode e as Unidades Escolares do município, percebeu-se a necessidade de um sistema capaz de compartilhar arquivos de forma restrita e segura.

As informações eram produzidas e utilizadas de maneira isolada em computadores que geralmente eram acessados por um único setor. Para a distribuição e disseminação destas informações, utilizava-se grande quantidade de papel, pois os documentos eram reproduzidos e encaminhados posteriormente.

O correio eletrônico ainda é utilizado intensivamente nas Unidades de Ensino do município, mas esta prática representa um risco, pois é impossível garantir a identificação do remetente. As mensagens de correio eletrônico podem ser interceptadas sem grandes dificuldades em seu trajeto e podem ser uma porta para propagação de vírus.

O sistema de GED desenvolvido amenizou esta situação com a disponibilização rápida e eficiente de documentos, sem a necessidade de reproduzir tais formulários, pois nele os documentos são facilmente compartilhados e não se desgastam como o papel. O sistema desenvolvido destina-se a um grupo limitado de usuários, que tem acesso de forma restrita. Desta forma compartilham-se documentos de maneira segura.

1.1 OBJETIVOS DO TRABALHO

O objetivo deste trabalho foi desenvolver um Sistema de Gerenciamento Eletrônico de Documentos na Secretaria da Educação e Cultura da Prefeitura Municipal de Pomerode.

Os objetivos específicos do trabalho são:

- a) permitir controle de acesso;
- b) permitir armazenamento e recuperação de documentos;
- c) permitir controle de versões gerando históricos.

1.2 ESTRUTURA DO TRABALHO

Este trabalho está organizado em quatro capítulos.

No primeiro capítulo tem-se a introdução, a justificativa e os objetivos do trabalho desenvolvido.

O segundo capítulo apresenta a definição de GED, um breve resumo da Secretaria da Educação e Cultura de Pomerode e uma análise de sistemas de gerenciamento de documentos já desenvolvidos.

No terceiro capítulo são apresentados a análise e o desenvolvimento da ferramenta, incluindo a descrição dos requisitos funcionais, os casos de uso e diagramas referentes ao sistema implementado assim como o resultado de sua utilização. Também são apresentadas definições das tecnologias utilizadas.

No capítulo 4 são apresentadas as conclusões do trabalho e sugestões para trabalhos futuros.

2 FUNDAMENTAÇÃO TEÓRICA

Neste capítulo são apresentados conceitos sobre GED, informações referentes à Secretaria da Educação e Cultura de Pomerode e em seguida são apresentados alguns sistemas de GED e as principais vantagens e desvantagens destes aplicativos.

2.1 GERENCIAMENTO ELETRÔNICO DE DOCUMENTOS

A história do GED iniciou pela preocupação com o armazenamento de grandes volumes de informação em mídias seguras e que ocupassem pouco espaço, por meio de digitalização de documento de origem papel. O GED surgiu para gerenciar uma grande quantidade de arquivos, gerados de forma isolada, usados geralmente por uma pessoa em um computador de difícil acesso e compartilhamento. Funciona através do preenchimento de uma ficha para cada documento criado, gerando um perfil com informações significativas e customizadas. Para a pesquisa do documento é usado um ou mais campos definidos no perfil. Quando os dados estão armazenados de forma organizada, tem-se um ganho de tempo, além de não expor o original que gera desgastes e extravios (FANTINI, 2001).

Além da praticidade proporcionada pelo armazenamento digital, que exige menos espaço físico, a consulta à documentação é mais rápida e flexível, pois não é necessário o acesso ao documento físico, permitindo o acesso simultâneo a um documento, garantindo segurança ao original. (BACK, 2004, p. 45).

Informações, às vezes caras e extremamente valiosas para uma organização, podem desaparecer pela constante mudança no quadro de funcionários. Agregar valor à informação é um fator que leva as organizações a dedicar esforços ao tratamento das informações relevantes do seu ambiente, pois incentivar o compartilhamento do conhecimento garante a execução das tarefas de uma maneira mais eficiente, levando a organização a obter vantagem competitiva (BACK, 2004).

O quadro 1 mostra uma comparação entre atividades de recuperação de documentos em papel e GED.

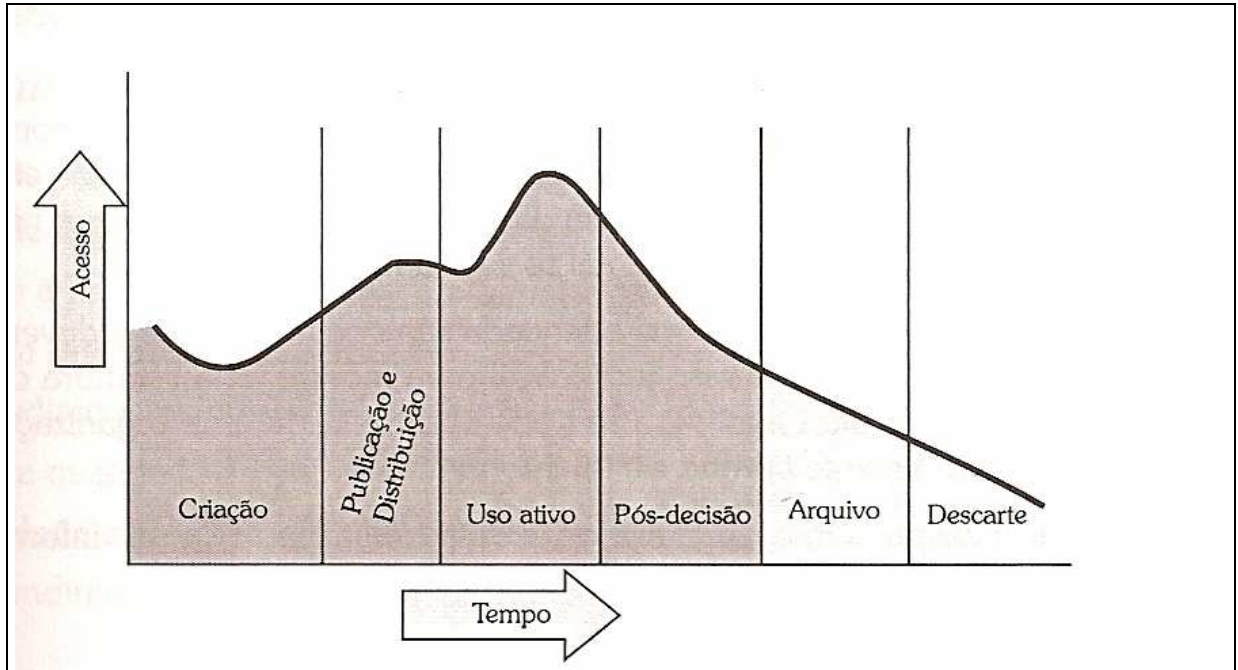
ATIVIDADE	PAPEL	GED
Capturar um documento	São armazenados em armários e pastas	Documentos são digitalizados para gerar imagens
Uso de mais de uma forma de armazenar documentos ou arquivos setoriais	Cópias são feitas e armazenadas em diversos arquivos	Busca por índices de diferentes maneiras para localizar o mesmo documento
Recuperação	Exemplo de fácil consulta: ir até a sala do arquivo, encontrar o documento, removê-lo, ir à copiadora, fazer a cópia, retornar o original ao local de origem	Ir ao computador, pesquisar pelo índice desejado, visualizar ou imprimir
Tempo de recuperação	Desde vários minutos até semanas	Segundos
Espaço exigido para armazenamento (documentos por m ³)	Alguns milhares	Milhões
Potencial de perda de documentos	Alta	Mínima
Impacto na infra-estrutura de computadores	Nenhum	Alto
Impacto no sistema atualmente em uso	Nenhum	Potencialmente alto – pode requerer revisão de processos.

Fonte: Adaptado de Baldam, Valle, Cavalcanti (2002, p. 37)

Quadro 1: Comparação entre atividades de recuperação de documentos em papel e GED

Tem sido uma atitude muito comum pensar no gerenciamento de documentos concluídos, ou seja, começar a gerenciá-los somente após terem sido tomadas todas as decisões sobre eles. Isso gera um grande problema, pois grande parte da consulta ao documento é realizada antes do envio ao arquivo. O ganho e a produtividade da equipe serão maiores se começarem a fazer o gerenciamento desse documento o mais cedo possível.

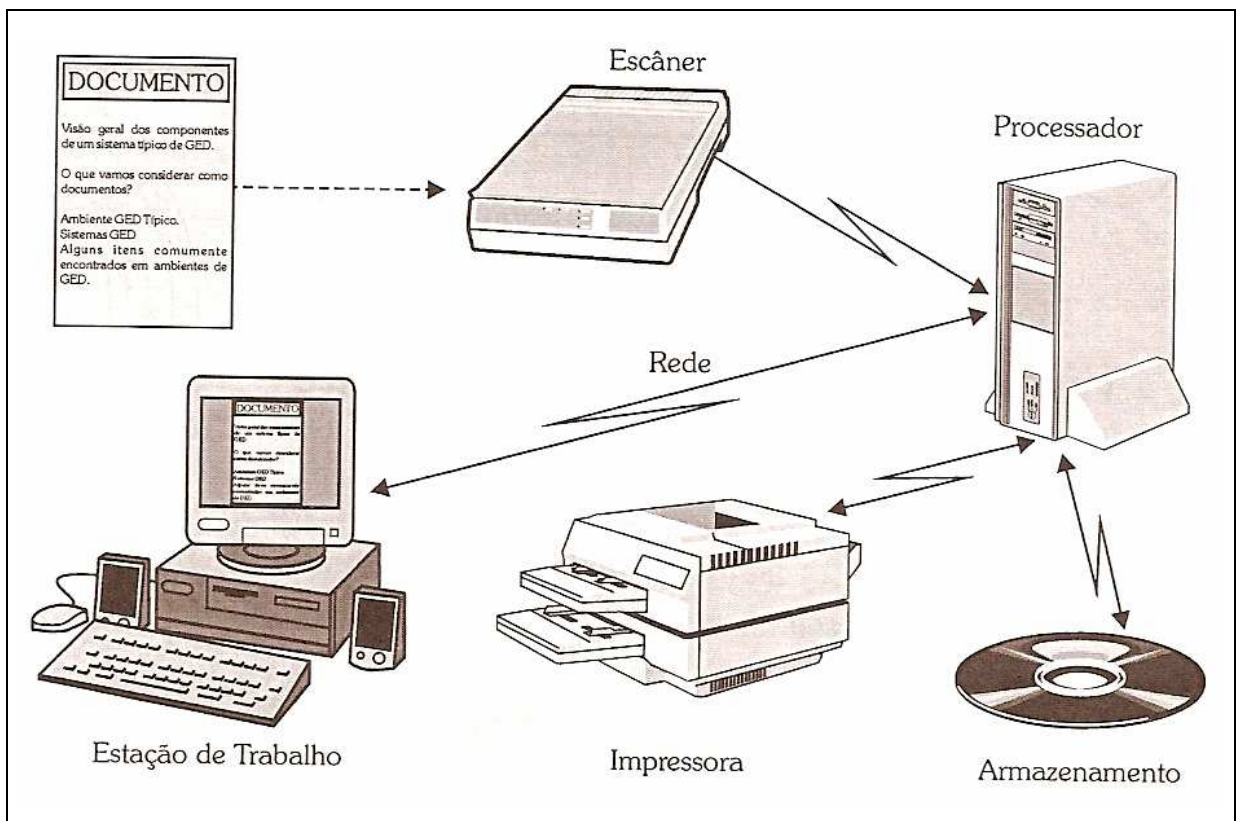
A Figura 1 mostra as fases do ciclo de vida dos documentos.



Fonte: Baldam, Valle, Cavalcanti (2002, p. 35)

Figura 1: Ciclo de vida de documentos

Embora um ambiente de GED possa ter configurações totalmente variadas, dependendo da aplicação desejada, os componentes típicos desse sistema podem ser visualizados na Figura 2.



Fonte: Baldam, Valle, Cavalcanti (2002, p. 40)

Figura 2: Ambiente típico

2.2 A SECRETARIA DA EDUCAÇÃO E CULTURA DE POMERODE

A Secretaria da Educação e Cultura de Pomerode é atualmente o setor do município com maior número de colaboradores, são cerca de 350 funcionários. Entre as suas responsabilidades está a instalação e conservação de estabelecimentos municipais de ensino, manutenção do serviço de alimentação escolar e do transporte escolar e a sustentação de cursos do ensino fundamental e da educação infantil. Da mesma forma, está incumbida de fiscalizar os convênios firmados com outros órgãos ou entidades das áreas de educação. É ainda encarregada da instalação e manutenção da biblioteca pública municipal e de outras unidades culturais. Trabalha pela difusão e estímulo à cultura em todos os seus aspectos, proteção ao patrimônio histórico e cultural do município e também cuida da promoção e execução de programas desportivos para maior desenvolvimento do esporte municipal em suas mais diversas modalidades (PMP, 2006).

A Prefeitura Municipal de Pomerode não possui um servidor para armazenar as informações produzidas pelas secretarias. De uma forma geral as informações eram produzidas e utilizadas de maneira isolada em computadores que geralmente eram acessados por um único setor. Utiliza-se uma grande quantidade de documentos prontos, que não serão mais alterados. São portarias, leis, decretos, processos, fichas, relatórios que são consultados e logo após retornam aos arquivos.

A grande maioria dos documentos emitidos pela Secretaria da Educação e Cultura de Pomerode, destinados às Unidades de Ensino do município eram em papel. A dificuldade em compartilhar informações e a redundância de conteúdos eram fatores que dificultavam o trabalho.

A dificuldade de se trabalhar com o papel e a importância de tratar adequadamente a documentação na Secretaria da Educação e Cultura de Pomerode geravam vários problemas. A preocupação em não apenas arquivar ou acumular documentos gerou uma análise da situação dos documentos da instituição.

2.3 TRABALHOS CORRELATOS

No ano de 1999 foi desenvolvido no Núcleo de Informática da Universidade Regional de Blumenau (FURB), um Sistema para Administração de Arquivos e Gerenciamento Eletrônico de Documentos.

Foram pesquisados sistemas e técnicas existentes para a administração de documentos e arquivos, e com base neste estudo foi apresentada uma especificação e implementação de um sistema desenvolvido na FURB. O sistema utiliza o banco de dados Oracle para armazenar os dados e o ambiente Oracle Developer para Windows 95 para executar a aplicação gerada na ferramenta Case Designer 2000. O sistema foi desenvolvido a partir da Metodologia de Desenvolvimento de Sistemas do Núcleo de Informática da FURB. Este sistema foi implantado em diversos setores na universidade possibilitando acesso às informações corretas de forma mais rápida e organizada. (NOVELLETTO, 1999, p. 5).

Em 2004 implantou-se um sistema de GED em uma empresa de base tecnológica. Essa categoria de empresa é caracterizada pela dependência da informação e a demanda por sistemas de informação é maior. O sistema foi desenvolvido em Delphi utilizando um banco de dados Oracle.

Durante o trabalho são apresentados as principais tecnologias e conceitos necessários ao entendimento do assunto e são expostos argumentos sustentando a importância do uso destas tecnologias para a gestão do conhecimento. Uma análise do ambiente organizacional permitiu identificar elementos que justificam o uso de sistemas de GED, bem como os benefícios proporcionados. Para uma aplicação efetiva é importante que a implantação seja criteriosa e respeite as particularidades de cada empresa. Desta forma, foi proposta uma metodologia de aplicação de sistemas de GED. O modelo foi validado por meio do estudo de caso em uma empresa de base tecnológica. O trabalho desenvolvido junto à empresa envolveu uma grande interação com todos os profissionais por meio de entrevistas, realização de eventos de sensibilização e *workshops*. Ao se utilizar a metodologia proposta garantiu-se objetividade e fluência das atividades. Como principal produto deste trabalho teve-se a implantação de um sistema de GED dedicado ao tratamento de documentos de origem digital, que oferece suporte a todas as atividades de criação e manipulação de documentos. A aplicação permitiu verificar os resultados proporcionados por uma tecnologia facilmente assimilável pelos usuários que constataram benefícios na gestão documental, nas facilidades proporcionadas pelo compartilhamento do conhecimento documentado e na melhoria geral das condições de trabalho. (BACK, 2004, p. 5).

O sistema implantado na FURB e o desenvolvido por Back (BACK, 2004) permitiram uma análise dos principais pontos a serem abordados no desenvolvimento e implantação de um sistema de Gerenciamento Eletrônico de Documentos. Os dois sistemas são semelhantes, porém nenhum deles permite o acesso via Internet. Com a expansão e diante da importância que a Internet adquiriu, é inoportuno um sistema incapaz de compartilhar informações em uma rede.

Foram pesquisados alguns softwares de gerenciamento eletrônico de documentos. Um deles é o EDock Server. Extraindo informações do *site* da empresa responsável, percebeu-se que o software só está disponível em inglês. Este fato dificulta o uso porque a interface com o usuário não é amigável (SOFTWARE602, 2006).

Os softwares AutoManager View e AutoManager TeamWork desenvolvidos pela Cyco, para o gerenciamento de imagens e documentos são sistemas eficientes e robustos. Estão disponíveis em português. Mas são extremamente caros. O preço das licenças e do treinamento os inviabilizam para a situação (CYCO, 2006).

3 DESENVOLVIMENTO DO TRABALHO

O presente trabalho consiste na criação de um sistema que permite aos usuários cadastrar e recuperar documentos. Na seqüência são apresentados os requisitos principais do problema a ser trabalhado.

3.1 REQUISITOS PRINCIPAIS DO PROBLEMA A SER TRABALHADO

Através de entrevista com a Diretora Administrativa e demais funcionários da Secretaria da Educação e Cultura de Pomerode, foi levantado a ausência de um sistema informatizado para compartilhar documentos. Constatou-se que a disseminação das informações demanda muito trabalho, pois grande parte é feita de forma manual.

O trabalho consistiu em desenvolver um aplicativo que permitirá aos usuários conectar-se ao sistema através de senha, possibilitando o cadastramento e a pesquisa de documentos remotamente via web.

O aplicativo proposto para automatizar estes processos foi desenvolvido em *Active Server Pages* (ASP), acessando um Banco de Dados Access que ficará hospedado em um servidor, sendo acessado pelos usuários através de um navegador.

As próximas seções descrevem a especificação e a implementação do aplicativo proposto.

3.1.1 Requisitos

No quadro 2 são apresentados os requisitos funcionais levantados para o sistema.

Requisitos Funcionais	Caso de Uso
RF01: O sistema deverá permitir ao administrador o cadastramento de usuários.	UC01
RF02: O sistema deverá permitir o cadastramento dos níveis de acesso dos usuários pelo administrador.	UC02
RF03: O sistema deverá permitir ao administrador o cadastramento de tipos de documentos.	UC03
RF04: O sistema deverá permitir o cadastramento e disponibilização de documentos e imagens.	UC04
RF05: O sistema deverá permitir o controle de versões de documentos, gerando históricos dos documentos cadastrados.	UC05
RF06: O sistema deverá emitir um relatório com todos os documentos cadastrados.	UC06
RF07: O sistema deverá permitir pesquisa através de filtros e impressão de documentos.	UC07

Quadro 2: Requisitos Funcionais

O Quadro 3 lista os requisitos não funcionais previstos para o sistema.

Requisitos Não Funcionais
RNF01: Os perfis de usuário para acesso ao sistema são: 1 – Administrador: Possui acesso total ao sistema. 2 – Operador: Pode efetuar consultas, inclusões e alterações em áreas específicas do sistema
RNF02: O sistema deve ser compatível com navegador Explorer 5.0.
RNF03: O sistema será desenvolvido em ASP.
RNF04: O sistema utilizará banco de dados Access.
RNF05: O sistema poderá ser acessado via Internet através de login e senha.
RNF06: Os documentos disponibilizados deverão ser de natureza pública.

Quadro 3: Requisitos não funcionais

3.2 ESPECIFICAÇÃO

Esta seção apresenta as especificações utilizadas para o desenvolvimento do sistema. Para a criação dos modelos e diagramas, foi utilizada a ferramenta Enterprise Architect.

3.2.1 Casos de uso

A figura 3 apresenta o diagrama de casos de uso do sistema.

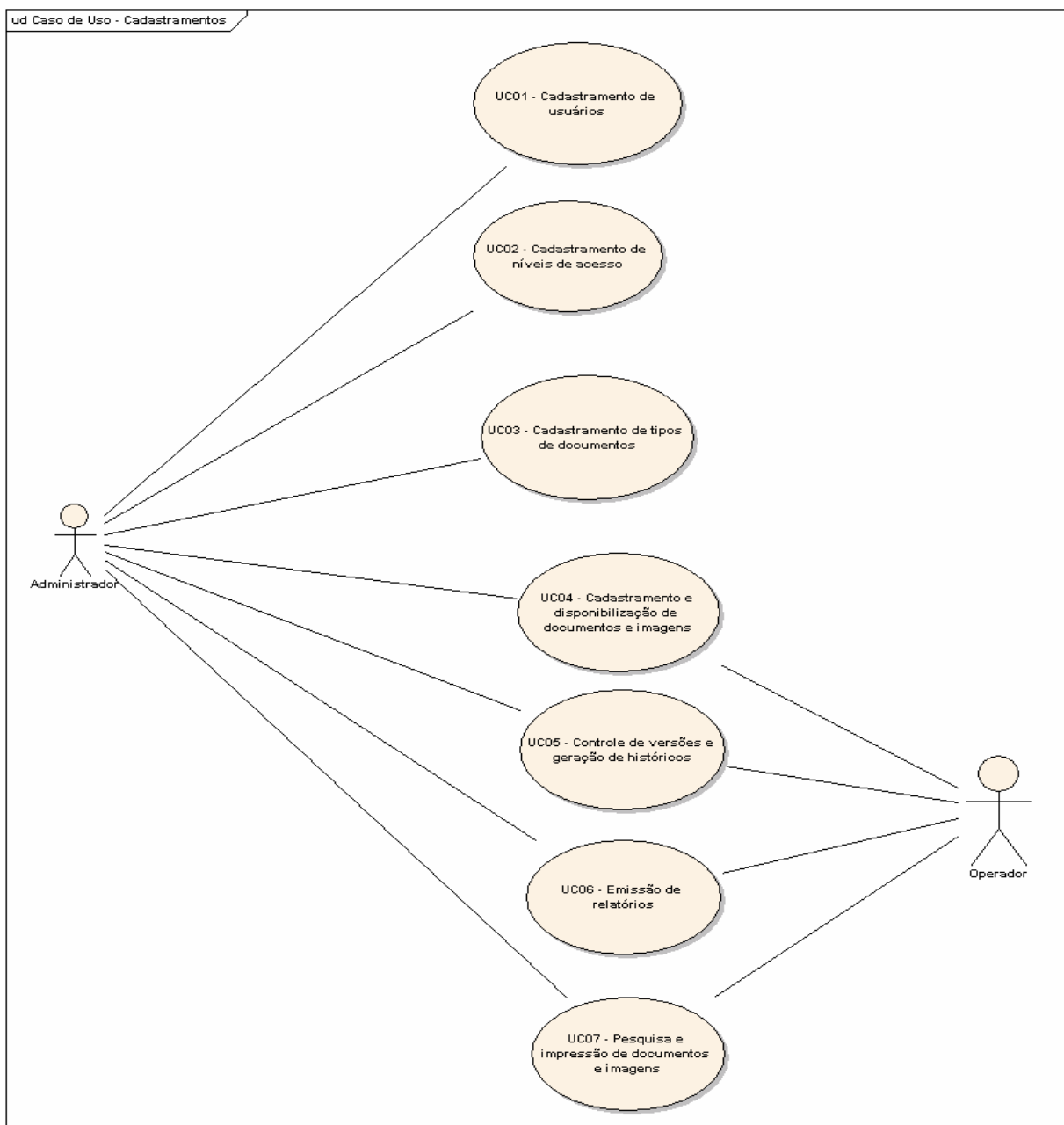


Figura 3: Diagrama de casos de uso

3.2.2 Diagrama de classes

A seguir está sendo apresentado o diagrama de classes do sistema e a maneira como as classes interagem entre si conforme a figura 4.

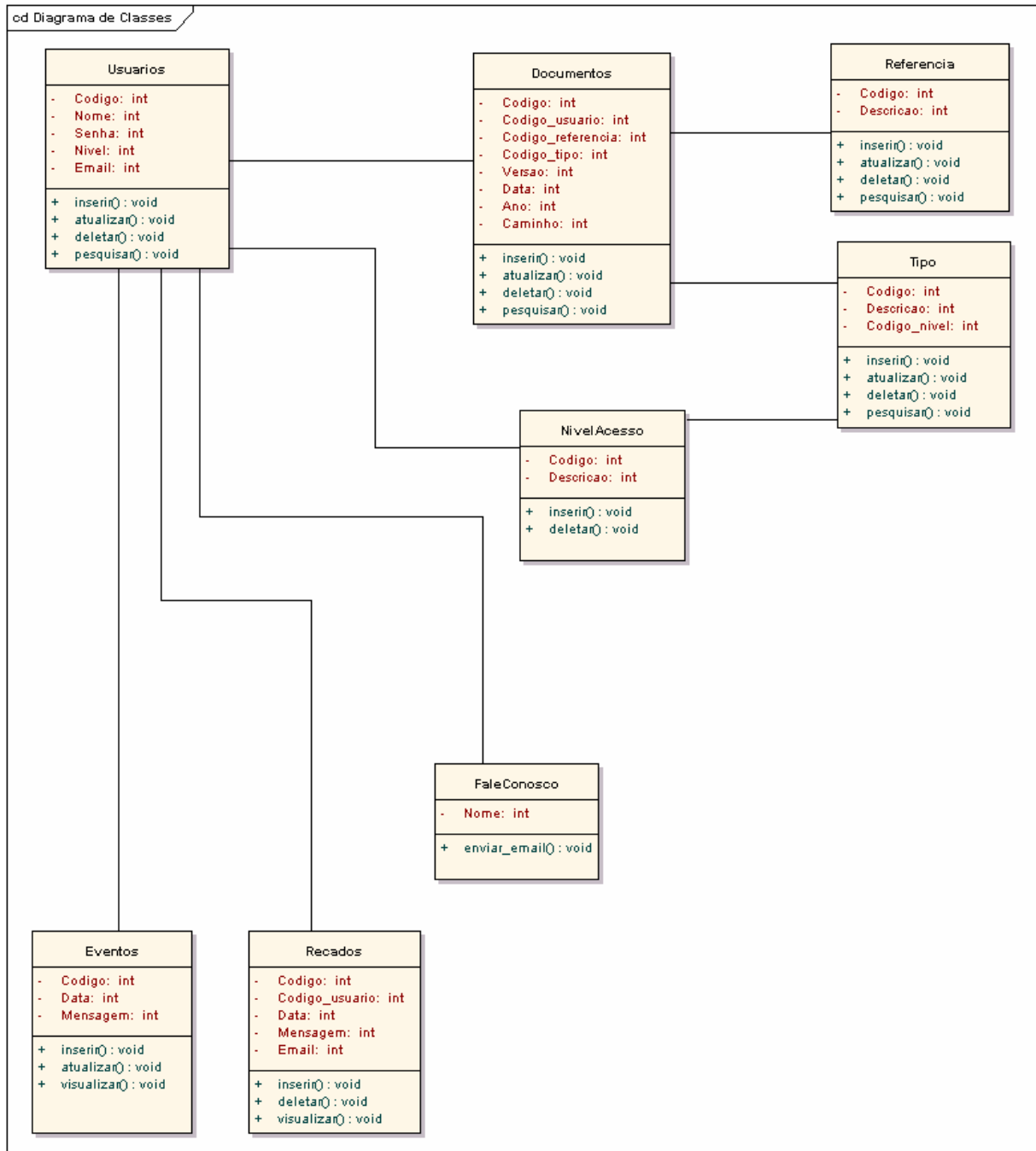


Figura 4: Diagrama de classes

3.2.3 Diagrama de atividades

A seguir, na figura 5 é apresentado o diagrama de atividades com as principais atividades do sistema.

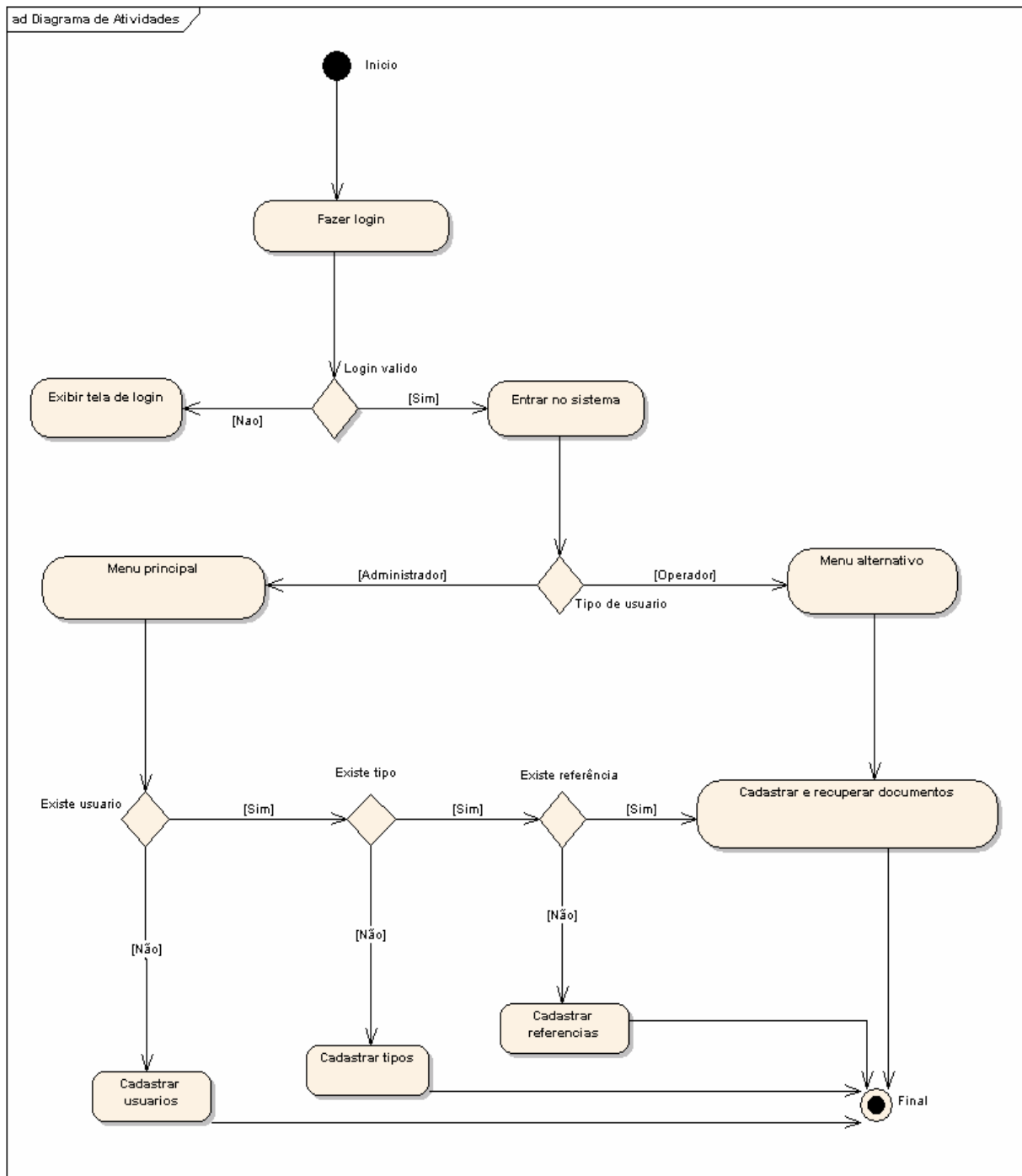


Figura 5: Diagrama de atividades

3.2.4 Modelo entidade relacionamento

O modelo entidade relacionamento (MER) pode ser visualizado na figura 6.

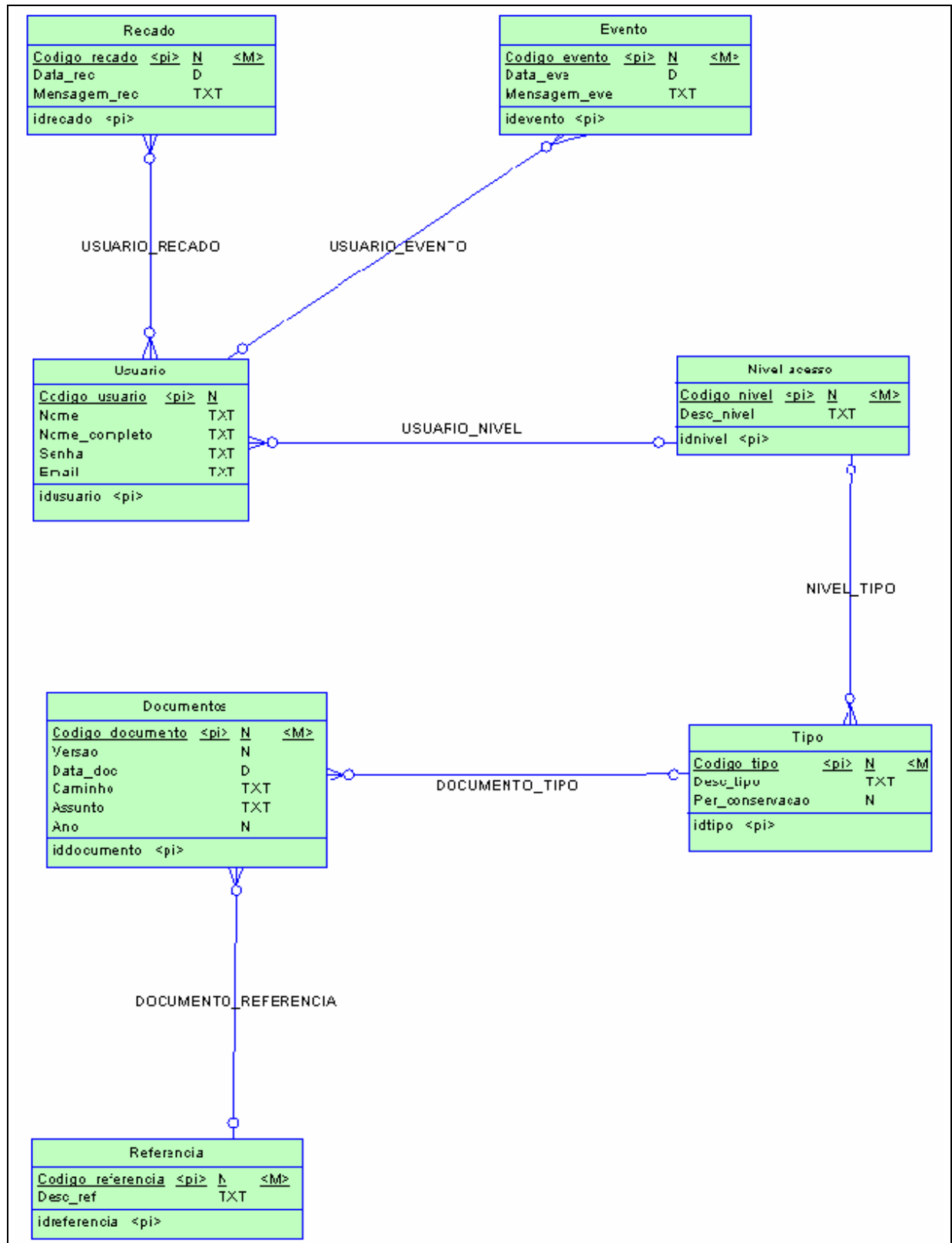


Figura 6: Modelo entidade relacionamento (MER)

3.3 IMPLEMENTAÇÃO

Neste capítulo são apresentadas informações sobre a implementação do trabalho expondo as técnicas e ferramentas utilizadas na implementação.

3.3.1 Técnicas e ferramentas utilizadas

Para a realização do trabalho utilizou-se ASP com o auxílio da ferramenta Dreamweaver e utilizou-se banco de dados Access. A seguir são apresentados os principais conceitos sobre os mesmos.

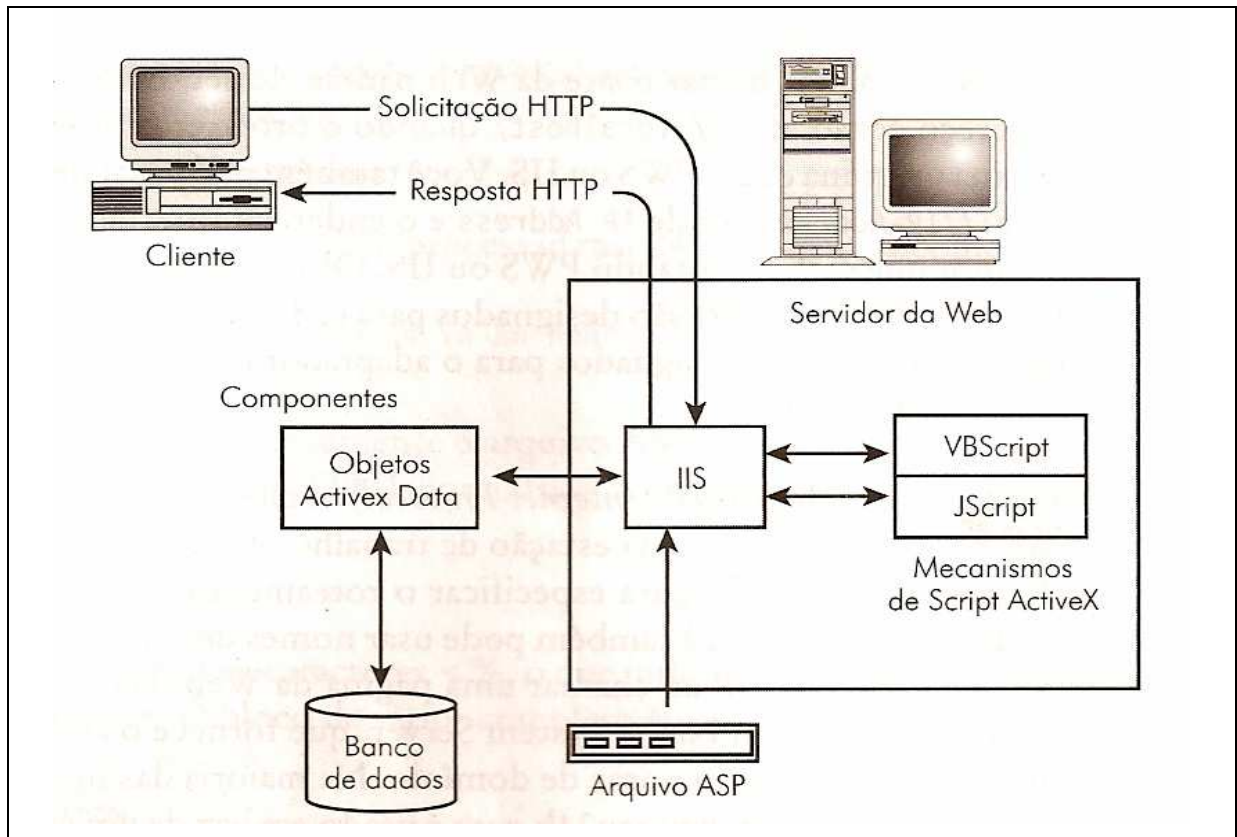
3.3.1.1 ASP

“As páginas ASP são scripts ou pequenos excertos de código interpretado pelo servidor da web para realizar uma tarefa particular.” (ROCHA, 2003, p. 1).

Esse ambiente de roteiros de servidores permite criar aplicações dinâmicas, rápidas e interativas na web em ambientes Microsoft, sem precisar preocupar-se com os recursos do navegador dos seus clientes. Um arquivo que contem código ASP é identificado pela extensão .asp. Ele contem duas partes: código *Hiper Text Markup Language* (HTML) e *scripts* que irão rodar no servidor, sendo VBScript a linguagem de roteiros mais utilizada. Uma linguagem de roteiros é uma sintaxe particular utilizada para executar comandos em um computador. As páginas ASP rodam no servidor e somente os resultados são enviados ao usuário (WILLIE; KOLLER, 1999).

Quando páginas ASP são recebidas pelo servidor é verificada a extensão do arquivo, se .html ou .asp. Encontrando a extensão .asp um programa especial no servidor lê o código contido no arquivo e executa-o. O resultado do processamento é devolvido ao navegador no formato HTML padrão (MARCORATTI, 1999).

A Figura 7 mostra o processamento de uma solicitação do ASP.



Fonte: Willie, Koller (1999, p. 16)

Figura 7: Processamento de uma solicitação ASP

Conforme Rocha (2003, p. 49), “basicamente, qualquer coisa pode ser feita com ASP, desde coletar dados de formulários até gerar páginas dinâmicas oriundas de bancos de dados, XML, arquivos texto.”

O quadro 4 mostra um exemplo do código de login em ASP.

```
<%@ Language=VBScript %>
<% Option Explicit %>
<!-- #include virtual="adovbs.inc" --> <%
Dim conexao
Dim strSql
Dim rs_login
Dim vnome
Dim vnivel
Dim vdescricao
Dim rs_nivel
Dim vcodigo
set conexao = Server.CreateObject("ADODB.Connection")
conexao.Open "DRIVER={Microsoft Access Driver (*.mdb)}; DBQ= "
&Server.MapPath ("data/Banco_de_dados.mdb")
strSql = "select * from Tabela_usuarios where Nome = '" &
Request.Form("Login") & "' and Senha = '" & Request.Form("Senha") &
"' "
set rs_login = conexao.execute(strSql)
if rs_login.eof then
'Usuario nao cadastrado
Response.Redirect("index.html")
```

```

else
vcodigo = rs_login("Codigo_usuario")
vnome = rs_login("Nome")
vnivel = rs_login("Codigo_nivel")
Response.Cookies("user")("nome") = vnome
Response.Cookies("user")("nivel") = vnivel
Response.Cookies("user")("codigo") = vcodigo
Response.Cookies("user")("ok") = "true"
end if
set conexao = Server.CreateObject("ADODB.Connection")
conexao.Open "DRIVER={Microsoft Access Driver (*.mdb)}; DBQ= "
&Server.MapPath ("data/Banco_de_dados.mdb")
strSql = "select * from Tabela_niveis where Codigo_nivel = " &vnivel&
""
set rs_nivel = conexao.execute(strSql)
if rs_nivel.eof then
'Nao encontrou
else
Do until rs_nivel.eof
vdescricao = rs_nivel("Descricao_acesso")
rs_nivel.moveNext
loop
end if
Response.Cookies("user")("descricao") = vdescricao
Response.Redirect("Menu.asp")
rs_login.close
conexao.close
set rs_login=nothing
set conexao=nothing %>

```

Quadro 4: Exemplo do código do login em ASP

3.3.1.2 Banco de Dados

“O mais útil recurso das Active Server Pages é a capacidade de uma página ASP interagir facilmente com um banco de dados. Um banco de dados é uma coleção de informações que pode ser facilmente consultadas e modificadas.” (MITCHELL; ATKINSON, 2000, p. 438).

Os bancos de dados fornecem o melhor desempenho para armazenar informações longas e heterogêneas, pois são projetados para simplificar o processo de armazenamento e recuperação das informações. Embora cada sistema de banco de dados possa ter suas pequenas diferenças, todos utilizam tabelas para armazenar informações. Uma tabela é uma matriz bidimensional que é utilizada para armazenar informações em um banco de dados. Um banco de dados relacional representa cada objeto, relacionado ou não, como sua própria tabela de banco de dados (MITCHELL; ATKINSON, 2000).

3.3.1.3 Modelo cliente-servidor

A Internet opera em um modelo cliente servidor, onde dois computadores trabalham juntos para realizar uma tarefa. Um computador cliente solicita algumas informações e o usuário interage com elas. Um servidor da web é um computador que contém todas as páginas da web para um site particular e tem um software especial instalado para enviar essas páginas para os navegadores que as solicitam (MITCHELL; ATKINSON, 2000).

Ao utilizar ASP, é fundamental entender que o código de ASP existe somente no servidor. O código ASP, que é o código cercado pelos delimitadores <% e %>, é processado completamente no servidor e o cliente não pode acessar esse código. Os roteiros são executados no lado do servidor e sua saída é enviada para o cliente (MITCHELL; ATKINSON, 2000).

3.3.2 Operacionalidade da implementação

A seguir são apresentadas as telas da aplicação visando mostrar as funcionalidades do sistema.

A figura 8 apresenta a tela inicial do aplicativo, sendo necessário a digitação do nome do usuário e senha para que o acesso seja liberado.

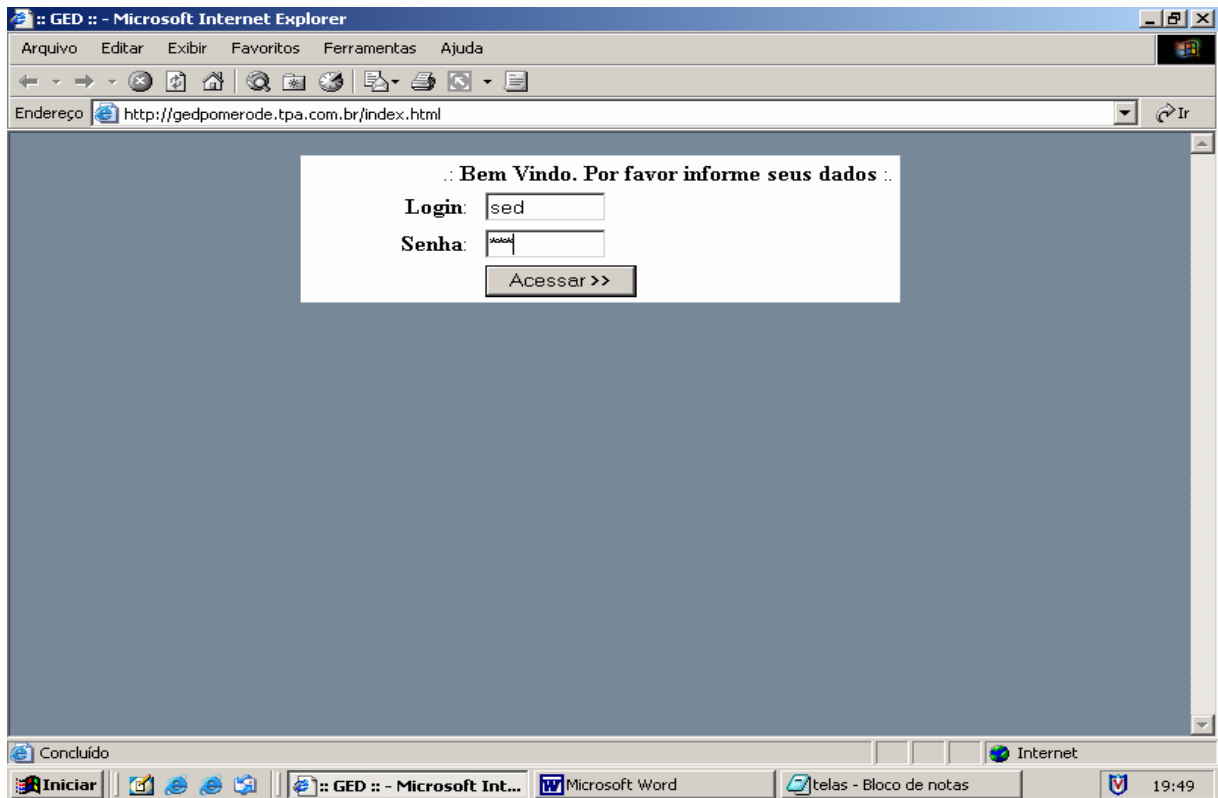


Figura 8: Tela inicial do aplicativo

A figura 9 apresenta a tela principal do sistema com as opções disponíveis.

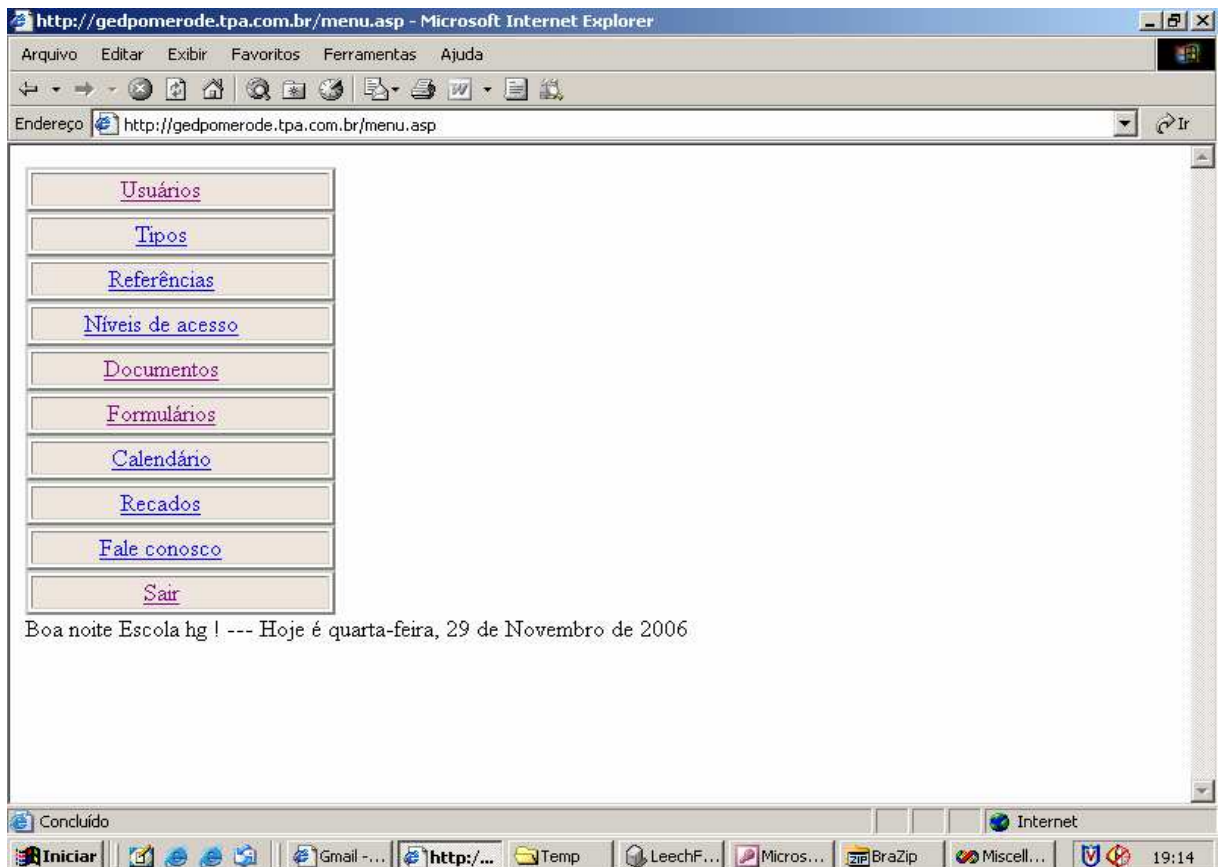
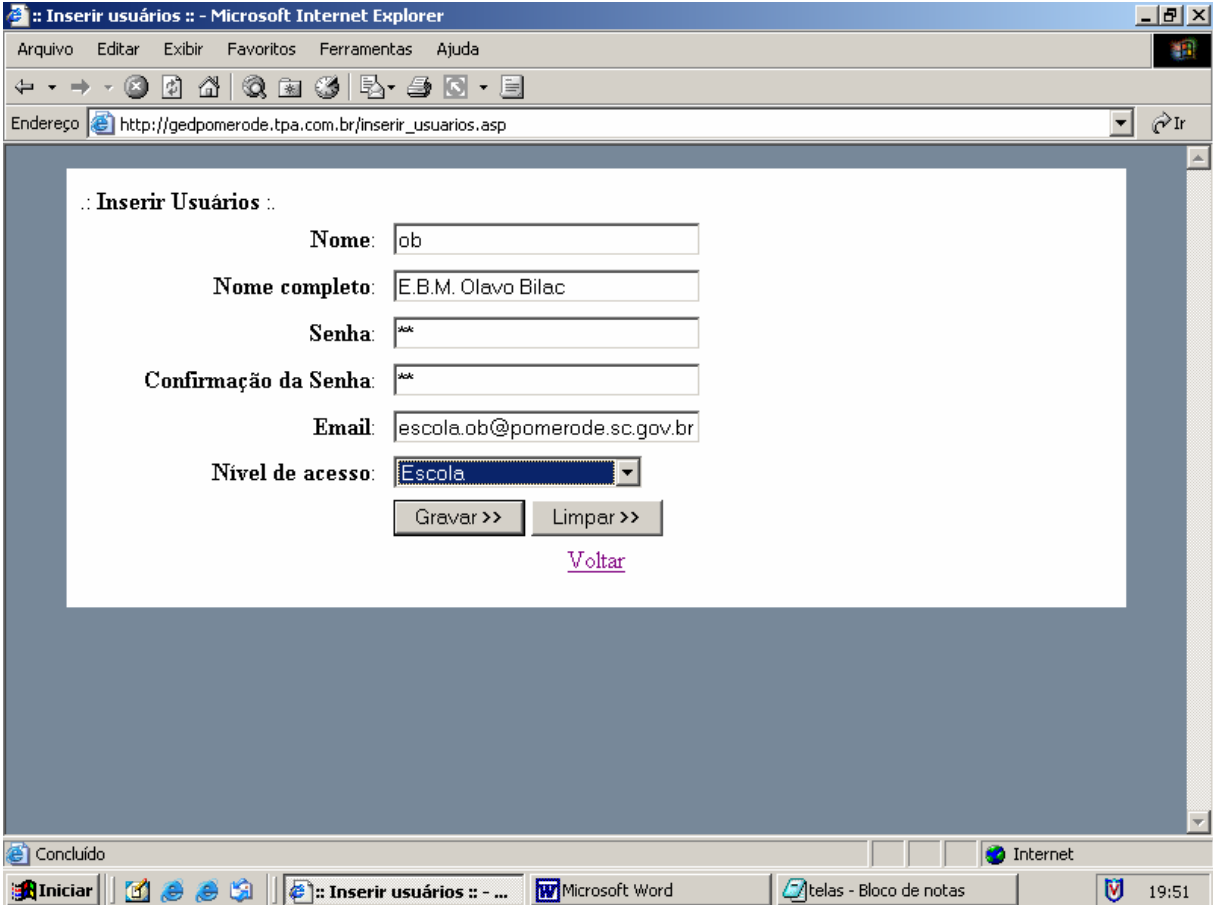


Figura 9: Tela principal do sistema

No sistema ora em questão, o cadastro de usuários foi realizado pela Diretora Administrativa da Secretaria da Educação e Cultura, para garantir que somente funcionários pré-estabelecidos pela Secretaria tenham acesso ao sistema. Cada usuário utilizará uma senha pessoal e intransferível e ficará responsável pelos documentos cadastrados ou postados com esta senha. Durante este cadastro será estabelecido o nível de acesso do usuário.

Apenas o usuário administrador contará com acesso total ao sistema. Os demais usuários terão acesso de forma restrita. A utilização de níveis serve para restringir o acesso a determinadas funções do sistema, como o cadastro de usuários, por exemplo.

Na figura 10 é apresentada a tela de cadastro de usuários quando o usuário conectado é o administrador.



The screenshot shows a Microsoft Internet Explorer browser window with the address bar displaying "http://gedpomerode.tpa.com.br/inserir_usuarios.asp". The page content is a registration form titled "Inserir Usuários" with the following fields and values:

- Nome: ob
- Nome completo: E.B.M. Olavo Bilac
- Senha: * (masked)
- Confirmação da Senha: * (masked)
- Email: escola.ob@pomerode.sc.gov.br
- Nível de acesso: Escola (selected in a dropdown menu)

At the bottom of the form, there are two buttons: "Gravar >>" and "Limpar >>". Below these buttons is a purple link labeled "Voltar". The browser's taskbar at the bottom shows the system tray with the time 19:51 and several open applications including "Iniciar", "Microsoft Word", and "telas - Bloco de notas".

Figura 10: Tela de cadastro de usuários

O cadastro de tipos de documentos padronizará as informações, facilitando a pesquisa e recuperação dos arquivos. Durante o cadastro de tipos será estabelecido qual o nível de acesso necessário para acessar estes documentos e o período de conservação dos documentos, em anos.

A figura 11 mostra a tela de tipos de documentos.

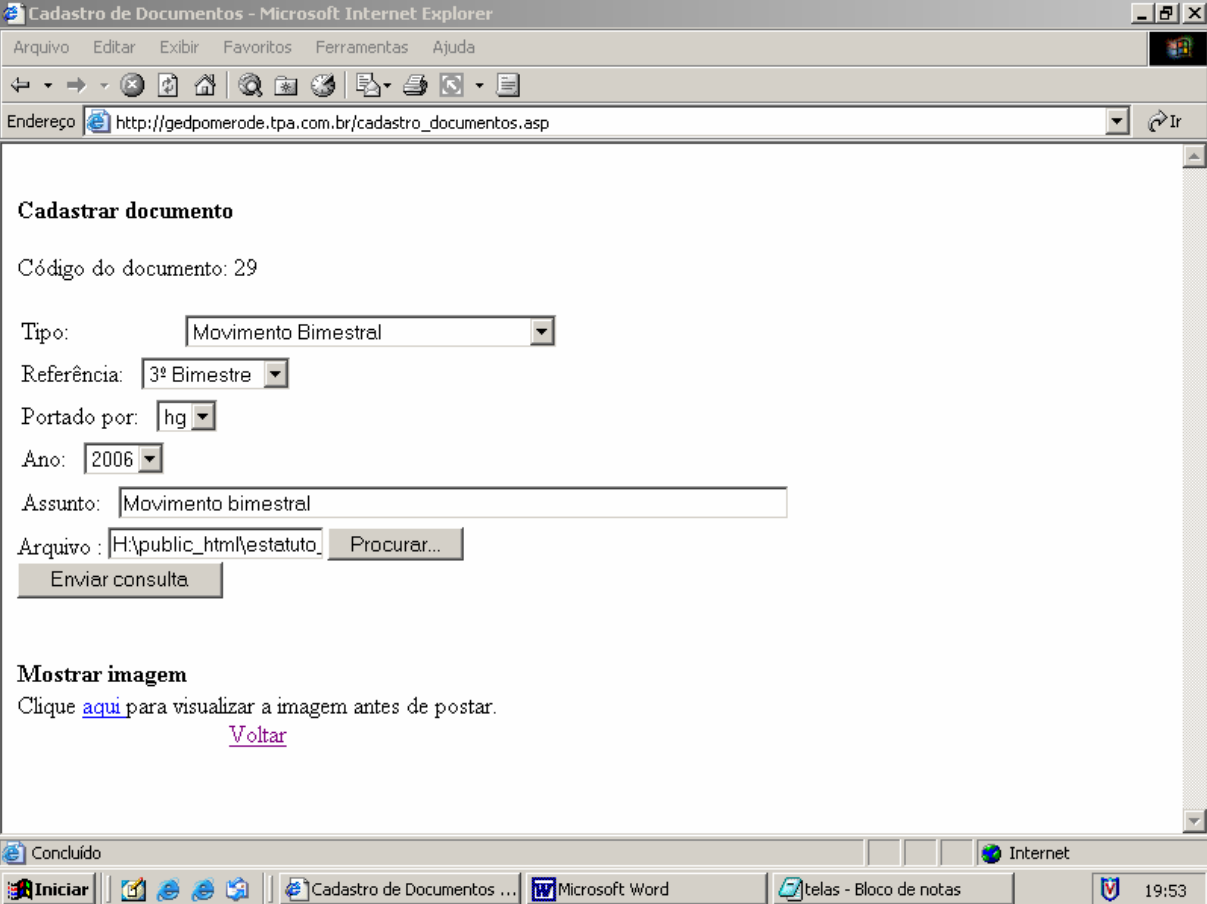
The screenshot shows a web browser window with the address bar displaying 'http://gedpomerode.tpa.com.br/listar_tipos.asp'. The page title is 'TIPOS'. Below the title are four buttons: 'Cadastrar', 'Alterar', 'Excluir', and 'Buscar'. A message reads 'Consulte documentos clicando sobre o tipo:'. Below this is a table with the following data:

Código	Descrição tipo	Tipo de acesso requerido	Guardar por
11	Portaria de Exoneração	Departamento Pessoal	Indeterminado
12	Portaria de Nomeação	Escola	Indeterminado
17	Calendário Administrativo	Escola	5
19	Resumo Ponto	Escola	30
20	Movimento Bimestral	Escola	30
21	Ficha SIS	Escola	30
24	FNDE	Escola	30
25	Documentação Escolar	Escola	30
27	Conselho Tutelar	Escola	10
28	ACTs	Visitante	5
29	Merenda Escolar	Escola	30
30	Relatórios Anuais	Escola	30
31	Históricos Escolares	Escola	Indeterminado

Figura 11: Tipos de documentos

O cadastramento de documentos e imagens permite aos usuários disponibilizar os arquivos que serão compartilhados. O usuário preenche uma ficha de cadastro com as informações referentes ao arquivo e faz o *upload* em seguida. Os arquivos cadastrados podem ter várias versões. Ao alterar o arquivo, a versão também será alterada. Esta rotina permite a visualização das alterações efetuadas nos documentos cadastrados. Existe também a opção de apagar um determinado arquivo. Ao escolher esta opção, o usuário é direcionado para uma página com a listagem dos documentos cadastrados por ele. Desta forma só poderá excluir documentos que ele mesmo cadastrou ou postou.

A figura 12 apresenta a tela de cadastro de documentos.



Cadastro de Documentos - Microsoft Internet Explorer

Arquivo Editar Exibir Favoritos Ferramentas Ajuda

Endereço http://gedpomerode.tpa.com.br/cadastro_documentos.asp

Cadastrar documento

Código do documento: 29

Tipo:

Referência:

Portado por:

Ano:

Assunto:

Arquivo: Procurar...

Mostrar imagem

Clique [aqui](#) para visualizar a imagem antes de postar.

[Voltar](#)

Concluído

Internet

Iniciar

Cadastro de Documentos ...

Microsoft Word

telas - Bloco de notas

19:53

Figura 12: Cadastro dos documentos

Poderão ser emitidos listagens ou relatórios de todos os arquivos cadastrados. Estes relatórios conterão informações da ficha de cadastro e facilitarão a pesquisa, assegurando maior controle as informações do sistema. Este relatório também exibe o número total de registros. Um exemplo do relatório pode ser observado na figura 13.

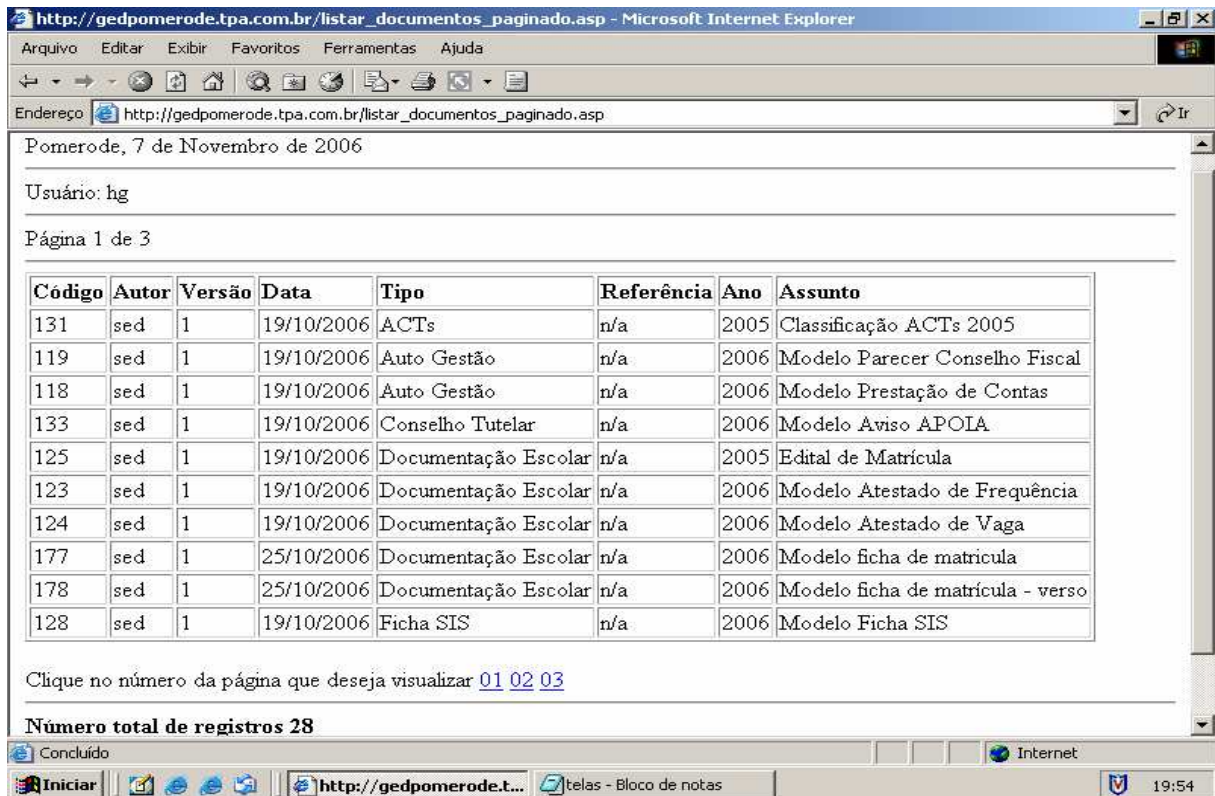


Figura 13: Relatório de documentos cadastrados

Na pesquisa, o usuário preenche pelo menos um campo-chave com as informações referentes ao arquivo que gostaria de localizar, como pode ser visto na figura 14.

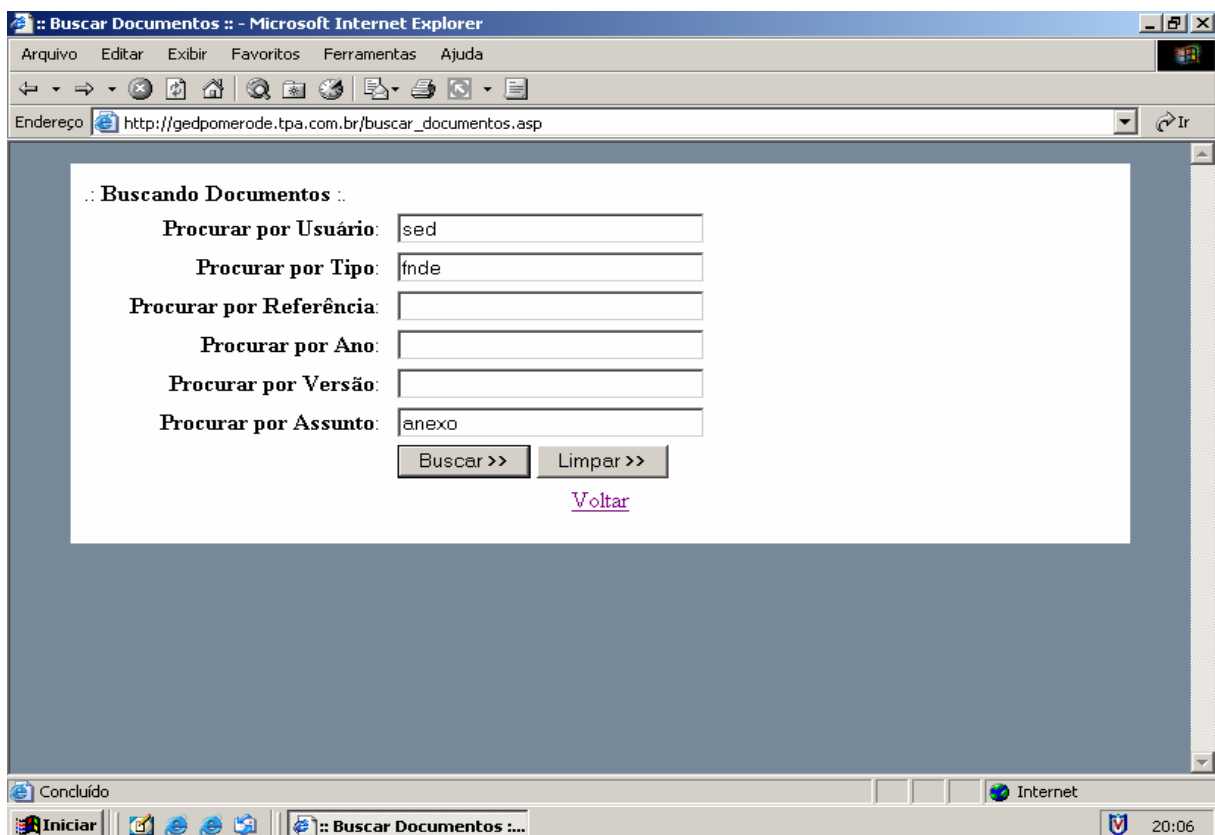


Figura 14: Tela de pesquisa de documentos

O resultado da pesquisa é semelhante à listagem de documentos cadastrados, pois o usuário poderá abrir e visualizar o documento como pode ser observado na figura 15.

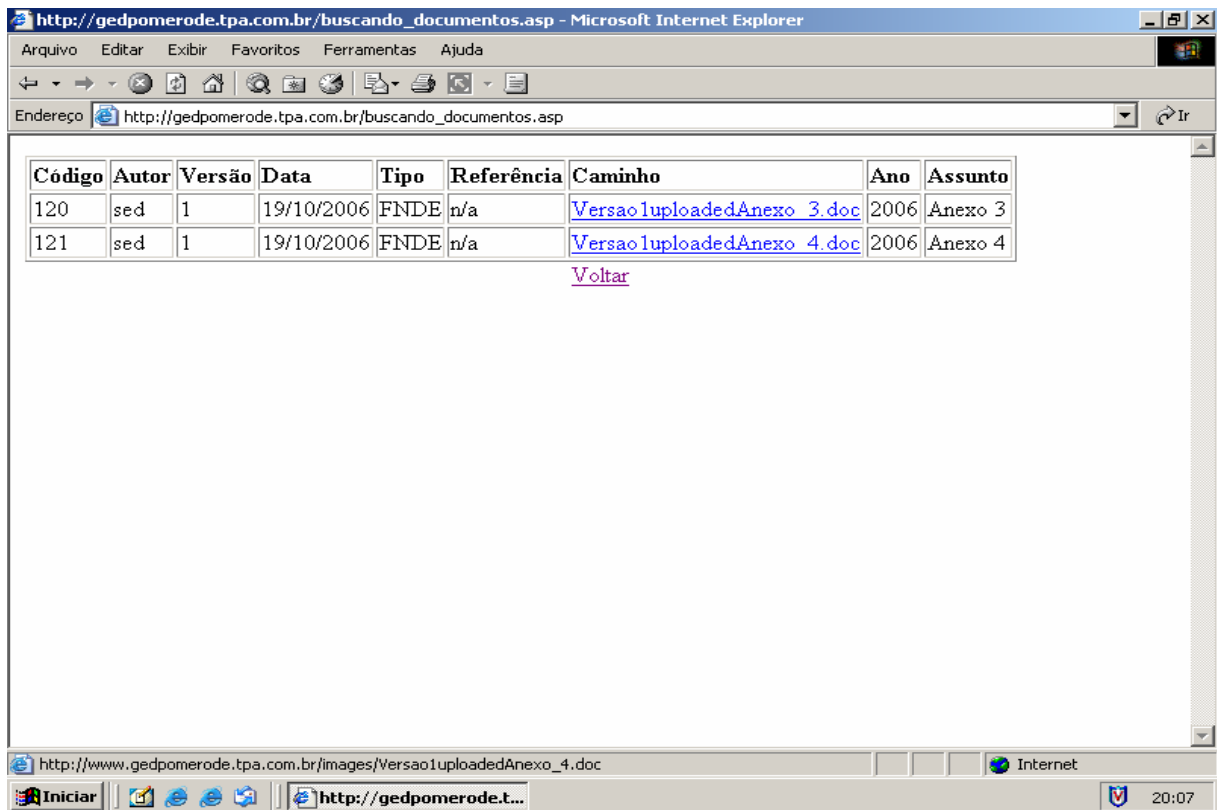


Figura 15: Resultado da pesquisa

3.4 RESULTADOS E DISCUSSÃO

O sistema desenvolvido enfoca a possibilidade do cadastro e recuperação de documentos via web, usando uma interface amigável. Conforme pesquisado, existem sistemas de GED que não permitem o acesso via Internet, são extremamente caros ou não possuem uma interface amigável.

Levando-se em consideração as limitações dos sistemas citados, optou-se pelo desenvolvimento de um sistema próprio para a Secretaria da Educação e Cultura Pomerode, com baixo custo, de fácil utilização e que pode adaptar-se facilmente.

4 CONCLUSÕES

A realização deste trabalho possibilitou o estudo e aprendizagem da linguagem de programação ASP, que se mostrou prática e adequada para o desenvolvimento do sistema. Uma característica importante do trabalho é a possibilidade do cadastramento e recuperação de documentos remotamente. Os arquivos inseridos no sistema podem ser acessados de qualquer lugar, via Internet, facilitando o compartilhamento de informações.

O sistema desenvolvido possui uma interface comum a todos os usuários, mas existe o controle de acesso a funções e documentos, conforme privilégios atribuídos ao usuário. O uso de uma *interface* única para todos os usuários facilita o uso e cria identidade. O controle de acesso garante a integridade dos documentos quanto ao uso indevido por usuários não autorizados.

O sistema permite a inserção de diferentes versões de um determinado documento, gerando um histórico com as alterações ocorridas. É imprescindível que o sistema disponibilize a versão atual do documento, mas também é fundamental que as alterações ocorridas fiquem registradas.

Ainda não foi utilizada uma métrica específica para avaliar o retorno proporcionado pela aplicação do sistema.

Comparado com outros sistemas de gerenciamento eletrônico de documentos existentes, o sistema desenvolvido apresenta como limitação a ausência de um procedimento de *backup*, tendo em vista que a criação de cópias de segurança dos documentos cadastrados e seus respectivos registros no banco de dados é feita de forma manual.

4.1 EXTENSÕES

Como sugestão para trabalhos futuros pode ser citado o desenvolvimento da pesquisa de documentos com *Full Text Retrieval* (FTR), que permite pesquisar uma base de dados de documentos não somente pelos seus índices, mas por qualquer palavra que o documento possua.

REFERÊNCIAS BIBLIOGRÁFICAS

BACK, Leonardo de Carlos. **Metodologia de implantação do gerenciamento eletrônico de documentos em empresa de base tecnológica.** 2004. 97 f. Dissertação (Mestrado em Engenharia de Produção) – Programa de Pós-graduação em Engenharia de Produção, Universidade Federal de Santa Catarina, Florianópolis.

BALDAM, Roquemar; VALLE, Rogério; CAVALCANTI, Marcos. **GED** : gerenciamento eletrônico de documentos. São Paulo: Érica, 2002.

CENADEM, **Centro Nacional de Desenvolvimento do Gerenciamento da Informação.** Apresenta informações sobre Gerenciamento Eletrônico de Documentos. 2006. Disponível em: <<http://www.cenadem.com.br/ged09.php>>. Acesso em 27 set. 2006.

CYCO. **Cyco Software.** Apresenta informações sobre o Cyco AutoManager View e Cyco AutoManager TeamWork. 2006. Disponível em: <<http://www.cyco.com/products/>>. Acesso em 12 abr. 2006.

FANTINI, Sérgio Rubens. **Aplicação do gerenciamento eletrônico de documentos:** estudo de caso de escolha de soluções. 2001. 118 f. Dissertação (Mestrado em Engenharia de Produção) – Programa de Pós-graduação em Engenharia de Produção, Universidade Federal de Santa Catarina, Florianópolis.

MACEDO, Geraldo Majela Ferreira de. **Bases para implantação de um sistema de gerenciamento eletrônico de documentos:** um estudo de caso. 2003. 154 f. Dissertação (Mestrado em Engenharia de Produção) – Programa de Pós-graduação em Engenharia de Produção, Universidade Federal de Santa Catarina, Florianópolis.

MARCORATTI, José Carlos. **ASP, ADO e banco de dados na Internet.** Florianópolis: Visual Books Editora, 1999.

MITCHELL, Scott; ATKINSON, James. **Aprenda em 21 dias Active Server Pages 3.0.** Rio de Janeiro: Campus, 2000.

NOVELLETTO, Douglas Guioroni Guaitorini. **Sistema para administração de arquivos e gerenciamento eletrônico de documentos para a FURB.** 1999. 41 f. Trabalho de Conclusão de Curso (Bacharelado em Ciências da Computação) – Centro de Ciências Exatas e Naturais, Universidade Regional de Blumenau, Blumenau.

PMP, Prefeitura Municipal de Pomerode. **Secretaria da Educação e Cultura.** 2006. Disponível em: <<http://www.pomerode.sc.gov.br>>. Acesso em: 11 abr. 2006.

ROCHA, Cerli Antonio da. **Desenvolvendo web sites dinâmicos:** PHP, ASP e JSP. Rio de Janeiro: Campus, 2003.

SOFTWARE602, Inc. **Edock Server.** 2006. Disponível em: <<http://www.software602.com/products/edock/>>. Acesso em: 12 abr. 2006.

WILLIE, Christoph; KOLLER, Christian. **Aprenda em 24 horas Active Server Pages**. Rio de Janeiro: Campus, 1999.

APÊNDICE A – Página inicial – index.html

“Código fonte da página inicial do aplicativo”

```

<%@ Language=VBScript %>
<% Option Explicit %>
<!-- #include virtual="adovbs.inc" -->
<% session.abandon %>

<html>
<head>
<title>:: GED ::</title>
</head>
<body bgcolor="#778899" >
<form name="form1" action="login.asp" method=post>
<table width="400" border="0" align="center">
<tr>
<td bgcolor=white>
<table width="100%" border="0" align="center">
<tr>
<td align=right colspan=2>.: <b>Bem Vindo. Por favor informe
seus dados</b> :.</td>
</tr>
<tr>
<td align=right width="30%"><b>Login</b>: &nbsp;</td>
<td><input
name="login" type=text id="login"
style="width:80px" value="" size=5></td>
</tr>
<td align=right width="30%"><b>Senha</b>: &nbsp;</td>
<td><input
name="senha" type=password id="senha"
style="width:80px" value="" size=5></td>
<tr>
<td>&nbsp;</td>
<td><input type=submit name=submit value="Acessar
&gt;&gt;"></td>
</tr>
</table>
</td>
</tr>
</table>
</form>
</body>
</html>

```


APÊNDICE B – Página principal – menu.asp

“Código fonte da página principal do aplicativo”

```

<%@ Language=VBScript %>
<% Option Explicit %>
<!-- #include virtual="adovbs.inc" --><%
Dim vsession
Dim vnivel
Dim vdescricao
Dim vok
vsession = Request.Cookies("user")("nome")
vnivel = Request.Cookies("user")("nivel")
vdescricao = Request.Cookies("user")("descricao")
vok = Request.Cookies("user")("ok")
if vok <> "true" then
Response.Redirect("index.html")
end if%>
<table width="27%" border=2 bgcolor=White>
<tr align="center">
<td width=168 bgcolor="#EEE5DE">
<a href="listar_usuarios.asp">Usuários</a>
</td>
<table width="27%" border=2 bgcolor=White>
<tr align="center">
<td width=168 bgcolor="#EEE5DE">
<a href="listar_tipos.asp">Tipos</a>
</td>
<table width="27%" border=2 bgcolor=White>
<tr align="center">
<td width=168 bgcolor="#EEE5DE">
<a href="listar_referencias.asp">Referências</a>
</td>
<table width="27%" border=2 bgcolor=White>
<tr align="center">
<td width=168 bgcolor="#EEE5DE">
<a href="listar_niveis.asp">Níveis de acesso</a>
</td>
<table width="27%" border=2 bgcolor=White>
<tr align="center">
<td width=168 bgcolor="#EEE5DE">
<a href="listar_documentos.asp">Documentos</a>
</td>
<table width="27%" border=2 bgcolor=White>
<tr align="center">
<td width=168 bgcolor="#EEE5DE">
<a href="calendario.asp">Calendário</a>
</td>
<table width="27%" border=2 bgcolor=White>

```

```

<tr align="center">
<td width=168 bgcolor="#EEE5DE">
<a href="recados.asp">Recados</a>
</td>
<table width="27%" border=2 bgcolor=White>
<tr align="center">
<td width=168 bgcolor="#EEE5DE">
<a href="fale.asp">Fale conosco</a>
</td>
<table width="27%" border=2 bgcolor=White>
<tr align="center">
<td width=168 bgcolor="#EEE5DE">
<a href="index.html">Sair</a>
</td>
</table>
<%
session.LCID = 1046
dim dia(7)
dia(1)="domingo"
dia(2)="segunda-feira"
dia(3)="terça-feira"
dia(4)="quarta-feira"
dia(5)="quinta-feira"
dia(6)="sexta-feira"
dia(7)="sábado"
dim mes(12)
mes(1)="Janeiro"
mes(2)="Fevereiro"
mes(3)="Março"
mes(4)="Abril"
mes(5)="Maio"
mes(6)="Junho"
mes(7)="Julho"
mes(8)="Agosto"
mes(9)="Setembro"
mes(10)="Outubro"
mes(11)="Novembro"
mes(12)="Dezembro"
Select Case Hour(now())
Case 0,1,2,3,4,5,6,7,8,9,10,11,12
Response.Write "Bom dia " &vdescricao& " "&vsession& " ! ---
Hoje é "
Case 13,14,15,16,17
Response.Write "Boa tarde " &vdescricao& " "&vsession& " ! ---
Hoje é "
Case else
Response.Write "Boa noite " &vdescricao& " "&vsession& " ! ---
Hoje é "
End Select
Response.write dia(weekday(now()))&" , "&day(now())&" de "&
mes(month(now()))&" de "& year(now()) %>

```

APÊNDICE C – Página de cadastro de usuários – inserir_usuarios.asp

“Código fonte da página de cadastro de usuários do aplicativo”

```

<%@ Language=VBScript %>
<% Option Explicit %>
<!-- #include virtual="adovbs.inc" --><%
Dim conexao
Dim rs_nivel
Dim strSql
Dim vsenha
Dim vsenha2
Dim vok
vok = Request.Cookies("user")("ok")
if vok <> "true" then
Response.Redirect("index.html")
end if
set conexao = Server.CreateObject("ADODB.Connection")
conexao.Open "DRIVER={Microsoft Access Driver (*.mdb)}; DBQ= "
&Server.MapPath ("data/Banco_de_dados.mdb")%>
<html>
<head>
<title>:: Inserir usuários ::</title>
</head>
<body bgcolor="#778899" >
<table border=0 align=center bgcolor=White width="700">
<tr>
<td valign=top width="100%" align=center>
</td></tr>
<td valign=top>
<form name="form1" action="inserindo_usuarios.asp"
method=post>
<table width="100%" border="0" align="center">
<tr>
<td bgcolor=white> <table width="100%" border="0"
align="center">
<tr>
<td align=left colspan=2>.: <b>Inserir Usuários</b> :.</td>
</tr>
<tr>
<td align=right width="30%"><b>Nome</b>: &nbsp;</td>
<td><input name="NOME" type=text id="NOME" value="" size=30
maxlength="50"></td>
</tr><tr> </tr><tr>
<td align=right width="30%"><b>Nome completo</b>: &nbsp;</td>
<td><input name="NOME_COMPLETO" type=text id="NOME_COMPLETO"
value="" size=30 maxlength="100"></td>
</tr>
<tr> </tr>

```

```

<tr>
<td align=right width="30%"><b>Senha</b>: &nbsp;</td>
<td><input name="SENHA" type=password id="SENHA" value=""
size=30 maxlength="50"></td>
</tr>
<tr> </tr>
<tr>
<td align=right width="30%"><b>Confirmação da Senha</b>:
&nbsp;</td>
<td><input name="SENHA2" type=password id="SENHA2" value=""
size=30 maxlength="50"></td>
</tr>
<tr> </tr>
<td align=right width="30%"><b>Email</b>: &nbsp;</td>
<td><input name="EMAIL" type=text id="EMAIL" value="" size=30
maxlength="50"></td></tr>
<tr> </tr>
<td align=right width="30%"><b>Nível de acesso</b>: &nbsp;</td>
<td>
<select name="listaop" size="1">
<%
strSql = "Select * from Tabela_niveis"
set rs_nivel = conexao.execute(strSql)
do while not rs_nivel.eof
%>
<option value="<%=rs_nivel("Codigo_nivel")%>">
<%=rs_nivel("Descricao_acesso")%></option>
<% rs_nivel.Movenext
Loop
%>
</tr>
<tr>
<td>&nbsp;</td>
<td><input type=submit name=submit value="Gravar &gt;&gt;">
<input type=reset name=submit2 value="Limpar &gt;&gt;"></td>
</tr>
</table></td>
</tr>
</table>
<center> <a href="listar_usuarios.asp">Voltar</a></P>
</center>
</form></td>
</tr>
</table>
</body>
</html>
<%
rs_nivel.close
conexao.close
set rs_nivel = nothing
set conexao=nothing %>

```

APÊNDICE D – Página de listagem de tipos – listar_tipos.asp

“Código fonte da página de listagem de tipos de documentos”

```

<%@ Language=VBScript %>
<% Option Explicit %>
<!-- #include virtual="adovbs.inc" -->
<%
Dim conexao
Dim rs_tipos
Dim strSql
Dim item
Dim vdesc
Dim vnivel_permitido
Dim vcodigo
Dim vdescricao
Dim vok
Dim vconservar
vok = Request.Cookies("user")("ok")
if vok <> "true" then
Response.Redirect("index.html")
end if
vdescricao = Request.Cookies("user")("descricao")
if vdescricao = "Administrador" then
%><center> <b> TIPOS </B> </center>
<table width="100%" border=2 bgcolor=White>
<tr align="center">
<td width=160 bgcolor="#EEE5DE">
<a href="inserir_tipo.asp">Cadastrar</a>
</td>
<td width=155 bgcolor="#EEE5DE">
<a href="listar_alterar_tipo.asp">Alterar</a>
</td>
<td width=145 bgcolor="#EEE5DE">
<a href="listar_excluir_tipo.asp">Excluir</a>
<td width=135 bgcolor="#EEE5DE">
<a href="buscar_tipos.asp">Buscar</a>
</td>
</tr>
<%else%>
<center> <b> TIPOS </B> </center>
<table width="100%" border=2 bgcolor=White>
<tr align="center">
<td width=135 bgcolor="#EEE5DE">
<a href="buscar_tipos.asp">Buscar</a>
</td>
</tr>
<% end if

```

```

set conexao = Server.CreateObject("ADODB.Connection")
conexao.Open "DRIVER={Microsoft Access Driver (*.mdb)}; DBQ= "
&Server.MapPath ("data/Banco_de_dados.mdb")
strSql      =      "select      Tabela_tipos.Codigo_tipo,
Tabela_tipos.Descricao_tipo,      Tabela_tipos.Conservar,
Tabela_niveis.Descricao_acesso from Tabela_niveis INNER JOIN
Tabela_tipos      ON      Tabela_niveis.Codigo_nivel      =
Tabela_tipos.Nivel_permitido;"
set rs_tipos = conexao.execute(strSql)
'Determina se ha registros
If rs_tipos.EOF then
%>
<center> <b> Tipo pesquisado não está cadastrado!!! </b>
</center>
<%
Else
%>
<table border = 1>
<b> Consulte documentos clicando sobre o tipo: </b> </p>
<% Response.Write "<tr>"
Response.Write      " <td><B>Código</B></td><td><B>Descrição
tipo</B></td><td><B>Tipo
de      acesso
requerido</B></td><td><B>Guardar por</B></td>"
Response.Write"</tr>"
Do until rs_tipos.EOF
vcodigo = rs_tipos("Codigo_tipo")
vdesc = rs_tipos("Descricao_tipo")
vnivel_permitido = rs_tipos("Descricao_acesso")
vconservar = rs_tipos("Conservar")
Response.Write "<tr>"
Response.Write "<td>" & vcodigo &"</td>"
Response.Write      " <td><A
href="&"listar_documentos_tipo.asp?id="& vcodigo & ">" & vdesc
&" </a></td>"
Response.Write "<td>" & vnivel_permitido &"</td>"
Response.Write "<td>" & vconservar &"</td>"
rs_tipos.moveNext
loop
end if
%>
</table>
<hr>
<center> <a href="menu.asp">Voltar</a></P> </center>
<%
rs_tipos.Close
Set rs_tipos = Nothing
conexao.Close
Set conexao = Nothing
%>
</body>
</html>

```

APÊNDICE E – Página de cadastro de documentos – cadastrar_documentos.asp

“Código fonte da página de cadastramento de documentos do sistema”

```
<%@ Language=VBScript %>
<% Option Explicit %>
<!-- #include virtual="adovbs.inc" -->
<%
Dim UploadRequest
Dim conexao
Dim rs_up
Dim byteCount
Dim RequestBin
Dim filepathName
Dim ScriptObject
Dim pathEnd
Dim MyFile
Dim idt
Dim inserir
Dim usuario
Dim referencia
Dim tipo
Dim versao
Dim ano
Dim assunto
Dim vdescricao
Dim vok
vok = Request.Cookies("user")("ok")
if vok <> "true" then
Response.Redirect("index.html")
end if
vdescricao = Request.Cookies("user")("descricao")
Response.Expires=0
Response.Buffer = TRUE
Response.Clear
byteCount = Request.TotalBytes
RequestBin = Request.BinaryRead(byteCount)
Set UploadRequest = CreateObject("Scripting.Dictionary")
BuildUploadRequest RequestBin
usuario = UploadRequest.Item("listaPostado").Item("Value")
tipo = UploadRequest.Item("listaTipo").Item("Value")
referencia =
UploadRequest.Item("listaReferencia").Item("Value")
ano = UploadRequest.Item("listaAno").Item("Value")
assunto = UploadRequest.Item("Assunto").Item("Value")
contentType =
UploadRequest.Item("blob").Item("ContentType")'tipo
filepathname = UploadRequest.Item("blob").Item("FileName")
```

```

'caminho original
filename = Right(filepathname,Len(filepathname)-
InstrRev(filepathname,"\\")) 'nome
value = UploadRequest.Item("blob").Item("Value")
'Cria componente
Set ScriptObject =
Server.CreateObject("Scripting.FileSystemObject")
set conexao = Server.CreateObject("ADODB.Connection")
conexao.Open "DRIVER={Microsoft Access Driver (*.mdb)}; DBQ=
" &Server.MapPath ("data/Banco_de_dados.mdb")
'Cria e grava arquivo
pathEnd =
Len(Server.mappath(Request.ServerVariables("PATH_INFO")))-24
'Set MyFile =
ScriptObject.CreateTextFile(Left(Server.mappath(Request.Server
Variables("path_info")),
pathEnd)&"Versaoluploaded"&filename)
Set MyFile =
ScriptObject.CreateTextFile(Server.MapPath("\images\Versaolup
loaded"&filename))
For i = 1 to LenB(value)
    MyFile.Write chr(AscB(MidB(value,i,1)))
Next
MyFile.Close
idt="Versaoluploaded"&filename
versao = 1
inserir = "insert into
Tabela_documentos(Codigo_usuario,Versao,Tipo,Referencia,Ano,C
aminho,Assunto) VALUES
('"&usuario&"','"&versao&"','"&tipo&"','"&referencia&"','"&an
o&"','"&idt&"',
' "&assunto&"')
rs_up = conexao.execute(inserir)
%>
<b><% Response.write "Nome do arquivo no servidor: " &idt
%></b><BR>
<b><% Response.write "Caminho:
http://www.gedpomerode.tpa.com.br/images" %></b><BR>
<b><% Response.write "Versao: " &versao %></b><BR>
<b><% Response.write "Assunto: " &assunto %></b><BR>
<b><% Response.write "Caminho original: " &filepathname
%></b><BR>
<b><% Response.write "Tipo de arquivo: " &contentType
%></b><BR>
<!--#include file="upload_teste.asp"-->
<center>    <a    href="listar_documentos.asp">Voltar</a></P>
</center>

```


APÊNDICE F – Página de relatório – listar_documentos_paginado.asp

“Código fonte da página de relatório de documentos do sistema”

```
<%@ Language=VBScript %>
<!-- #include virtual="adovbs.inc" --><%
Dim conexao
Dim rs_documentos
Dim pagina
Dim tamanhopagina
Dim SQL
Dim SQLtemp
Dim contador
Dim maximopagina
Dim maximoregistros
Dim registros
Dim campos
Dim i
Dim fieldvalue
Dim pad
Dim scriptname
Dim counter
Dim vnome
Dim ref
Dim vok
Dim vnivel
session.LCID = 1046
dim dia(7)
dia(1)="domingo"
dia(2)="segunda-feira"
dia(3)="terça-feira"
dia(4)="quarta-feira"
dia(5)="quinta-feira"
dia(6)="sexta-feira"
dia(7)="sábado"
dim mes(12)
mes(1)="Janeiro"
mes(2)="Fevereiro"
mes(3)="Março"
mes(4)="Abril"
mes(5)="Maio"
mes(6)="Junho"
mes(7)="Julho"
mes(8)="Agosto"
mes(9)="Setembro"
mes(10)="Outubro"
mes(11)="Novembro"
mes(12)="Dezembro"
```

```

vok = Request.Cookies("user")("ok")
if vok <> "true" then
Response.Redirect("index.html")end if
vnivel = Request.Cookies("user")("nivel")
vnome = Request.Cookies("user")("nome")
set conexao = Server.CreateObject("ADODB.Connection")
conexao.Open "DRIVER={Microsoft Access Driver (*.mdb)}; DBQ=
" &Server.MapPath ("data/Banco_de_dados.mdb")
set rs_documentos=Server.createObject("ADODB.Recordset")
rs_documentos.cursorlocation=aduseclient
rs_documentos.cachesize=5
SQLtemp = "select Tabela_documentos.Codigo_documento,
Tabela_usuarios.Nome, Tabela_documentos.Versao,
Tabela_documentos.Data, Tabela_tipos.Descricao_tipo,
Tabela_referencias.Descricao_referencia,
Tabela_documentos.Ano,
Tabela_documentos.Assunto from Tabela_referencias INNER JOIN
(Tabela_tipos
INNER JOIN (Tabela_usuarios INNER JOIN Tabela_documentos on
Tabela_usuarios.Codigo_usuario =
Tabela_documentos.Codigo_usuario) ON
Tabela_tipos.Codigo_tipo = Tabela_documentos.Tipo) ON
Tabela_referencias.Codigo_referencia =
Tabela_documentos.Referencia where
Tabela_tipos.Nivel_permitido >= "&vnivel&" order by
Tabela_tipos.Descricao_tipo,
Tabela_referencias.Descricao_Referencia,
Tabela_documentos.Assunto,
Tabela_documentos.Versao desc;"
pagina=request("qualpagina")
if pagina = "" then
    pagina=1
end if
tamanhopagina=request("pagesize")
if tamanhopagina="" then
    tamanhopagina = 10
end if
SQL=request("SQLquery")
if SQL="" then
SQL=SQLtemp
end if
rs_documentos.open SQLtemp, conexao, adOpenKeyset
contador = rs_documentos.recordCount
rs_documentos.Movefirst
rs_documentos.pagesize=tamanhopagina
maximopagina=cint(rs_documentos.PageCount)
maximoregistros=cint(rs_documentos.PageSize)
rs_documentos.AbsolutePage=pagina
registros=0
campos =rs_documentos.fields.count-1
response.write "Pomerode, "&day(now())&" de "&

```

```

mes(month(now()))&" de "& year(now())%><hr><%
response.write " Usuário: " &vnome%><hr><%
response.write " Página " &pagina &" de " &maximopagina &
"<br>"%><hr><%
response.write "<table border='1'><tr>"
Response.Write "<td><B>Código</B></td><td><B>Autor</B></td>
<td><B>Versão</B></td><td><B>Data</B></td><td><B>Tip</B></td>
<td><B>Referência</B></td><td><B>Ano</B></td><td><B>Assunto</
B></td>"
do until rs_documentos.eof or registros >=maximoregistros
  response.write "<tr>"
  for i=0 to campos
    fieldvalue=rs_documentos(i)
    if isnull(fieldvalue) then
      fieldvalue="n/a"
    end if
    if trim(fieldvalue)=" " then
      fieldvalue="&nbsp;"
    end if
    response.write "<td valign='top'>"
    response.write fieldvalue
    response.write "</td>"
  next
  response.write "</tr>"
  rs_documentos.movenext
  registros=registros+1
loop
response.write "</table><p>"
rs_documentos.Close
Set rs_documentos = Nothing
conexao.Close
Set conexao = Nothing%>
Clique no número da página que deseja visualizar
<%pad ="0"
scriptname=request.servervariables("scrip_name")
for counter=1 to maximopagina
  if counter >= 10 then
    pad="" end if
  ref="<a href=" &scriptname
  ref=ref & "?qualpagina="&counter
  ref=ref & "&pagesize="& tamanhopagina
  ref=ref & "&sqlQuery="&server.URLencode(SQL)
  ref=ref & ">" & pad & counter"</a>"
  response.write ref & " "
  if counter mod 10 = 0 then
    response.write "<br>" end if
next %><hr>
<b>Número total de registros <%=contador%></b>
<center><a href="listar_documentos.asp">Voltar</a></P>
</center></body></html>

```

APÊNDICE G – Página de busca – buscar_documentos.asp

“Código fonte da página de busca de documentos do sistema”

```

<%@ Language=VBScript %>
<% Option Explicit %>
<!-- #include virtual="adovbs.inc" -->
<%
Dim vok
vok = Request.Cookies("user")("ok")
if vok <> "true" then
Response.Redirect("index.html")
end if
%>
<html>
<head>
<title>:: Buscar Documentos ::</title>
</head>
<body bgcolor="#778899" >
<table border=0 align=center bgcolor=White width="700">
<tr>
<td valign=top width="100%" align=center>
</td></tr>
<td valign=top>
<form name="form1" action="buscando_documentos.asp"
method=post>
<table width="100%" border="0" align="center">
<tr>
<td bgcolor=white> <table width="100%" border="0"
align="center">
<tr>
<td align=left colspan=1>.: <b>Buscando Documentos</b> :.</td>
</tr>
<tr>
<td align=right width="30%"><b>Procurar por Usuário</b>:
&nbsp;</td>
<td><input name="key1" type=text id="key1" value="" size=30
maxlength="50"></td>
</tr>
<tr>
<td align=right width="30%"><b>Procurar por Tipo</b>:
&nbsp;</td>
<td><input name="key2" type=text id="key2" value="" size=30
maxlength="50"></td>
</tr>
<tr>
<td align=right width="30%"><b>Procurar por Referência</b>:
&nbsp;</td>

```

```

<td><input name="key3" type="text" id="key3" value="" size=30
maxlength="50"></td>
</tr>
<tr>
<td align="right" width="30%"><b>Procurar por Ano</b>:
&nbsp;   </td>
<td><input name="key4" type="text" id="key4" value="" size=30
maxlength="50"></td>
</tr>
<tr>
<td align="right" width="30%"><b>Procurar por Versão</b>:
&nbsp;   </td>
<td><input name="key5" type="text" id="key5" value="" size=30
maxlength="50"></td>
</tr>
<tr>
<td align="right" width="30%"><b>Procurar por Assunto</b>:
&nbsp;   </td>
<td><input name="key6" type="text" id="key6" value="" size=30
maxlength="50"></td>
</tr>
<tr>
<td>&nbsp;   </td>
<td><input type="submit" name="submit" value="Buscar &gt;&gt;">
<input type="reset" name="submit2" value="Limpar &gt;&gt;"></td>
</tr>
</table></td>
</tr>
</table>
<center> <a href="listar_documentos.asp">Voltar</a></P>
</center>
</form>
</td>
</tr>
</table>
</body>
</html>

```

APÊNDICE H – Página do resultado da busca – buscando_documentos.asp

“Código fonte da página com o resultado da busca de documentos”

```

<%@ Language=VBScript
Dim conexao
Dim rs_documentos
Dim strSql
Dim item
Dim vcodigo_documento
Dim vcodigo_usuario
Dim vtipo
Dim vreferencia
Dim vdata
Dim vano
Dim vversao
Dim vcaminho
Dim vassunto
Dim key1
Dim key2
Dim key3
Dim key4
Dim key5
Dim key6
Dim vnivel
Dim vok
vok = Request.Cookies("user")("ok")
if vok <> "true" then
Response.Redirect("index.html")
end if
vnivel = Request.Cookies("user")("nivel")
set conexao = Server.CreateObject("ADODB.Connection")
conexao.Open "DRIVER={Microsoft Access Driver (*.mdb)}; DBQ=
" &Server.MapPath ("data/Banco_de_dados.mdb")
strSql = "select Tabela_documentos.Codigo_documento,
Tabela_usuarios.Nome, Tabela_documentos.Versao,
Tabela_documentos.Data, Tabela_tipos.Descricao_tipo,
Tabela_referencias.Descricao_referencia,
Tabela_documentos.Caminho, Tabela_documentos.Ano,
Tabela_documentos.Assunto from Tabela_referencias INNER JOIN
(Tabela_tipos INNER JOIN (Tabela_usuarios INNER JOIN
Tabela_documentos on Tabela_usuarios.Codigo_usuario =
Tabela_documentos.Codigo_usuario) ON
Tabela_tipos.Codigo_tipo= Tabela_documentos.Tipo)ON
Tabela_referencias.Codigo_referencia
=Tabela_documentos.Referencia where Tabela_usuarios.Nome
like '%" &Request.Form("key1") & "%' and
Tabela_tipos.Descricao_tipo like '%" &Request.Form("key2") &

```

```

"%" and Tabela_referencias.Descricao_referencia like '%"
&Request.Form("key3") & "%" and Tabela_documentos.Ano like
"%" &Request.Form("key4") & "%" and Tabela_documentos.Versao
like '%" &Request.Form("key5") & "%" and
Tabela_documentos.Assunto like '%" &Request.Form("key6") &
"%" and Tabela_tipos.Nivel_permitido >= "&vnivel&" order by
Tabela_tipos.Descricao_tipo,
Tabela_referencias.Descricao_referencia,
Tabela_documentos.Assunto, Tabela_documentos.Versao desc"
set rs_documentos = conexao.execute(strSql)
If rs_documentos.EOF then      'Determina se ha registros
%> <center> <b> Documento pesquisado não está cadastrado!!!
</b> </center>
<hr> <center><a href="buscar_documentos.asp">Voltar</a>
</center> <% else 'Ha pelo menos um item que atende a
condição.%> <table border = 1>
<% Response.Write "<tr>"
Response.Write
"<td><B>Código</B></td><td><B>Autor</B></td><td><B>Versão</B>
</td><td><B>Data</B></td><td><B>Tipo</B></td><td><B>Referênci
a</B></td><td><B>Caminho</B></td><td><B>Ano</B></td><td><B>As
sunto</B></td>"
Do until rs_documentos.EOF
vcodigo_documento = rs_documentos("Codigo_documento")
vcodigo_usuario = rs_documentos("Nome")
vversao = rs_documentos("Versao")
vdata = rs_documentos("Data")
vtipo = rs_documentos("Descricao_tipo")
vreferencia = rs_documentos("Descricao_referencia")
vcaminho = rs_documentos("Caminho")
vano = rs_documentos("Ano")
vassunto = rs_documentos("Assunto")
Response.Write "<tr>"
Response.Write "<td>" & vcodigo_documento &"</td>"
Response.Write "<td>" & vcodigo_usuario &"</td>"
Response.Write "<td>" & vversao &"</td>"
Response.Write "<td>" & vdata &"</td>"
Response.Write "<td>" & vtipo &"</td>"
Response.Write "<td>" & vreferencia &"</td>"
Response.Write "<td><A
href="&"http://gedpomerode.tpa.com.br/images/"&vcaminho&">"
& vcaminho &" </a></td>"
Response.Write "<td>" & vano &"</td>"
Response.Write "<td>" & vassunto &"</td>"
rs_documentos.moveNext
loop%><center> <a href="buscar_documentos.asp">Voltar</a></P>
</center><% end if
rs_documentos.Close
conexao.Close
set conexao = Nothing%>

```