

**UNIVERSIDADE DE BRASÍLIA
FACULDADE DE TECNOLOGIA
DEPARTAMENTO DE ENGENHARIA ELÉTRICA**

Ambiente Multimídia como Suporte
para o Ensino de Telecomunicações

VLADIMIR DAIGELE SIMÕES BARBOSA

ORIENTADOR: PROF. DOCTEUR HUMBERTO ABDALLA
JUNIOR

PROJETO FINAL DE GRADUAÇÃO
EM ENGENHARIA ELÉTRICA

BRASÍLIA / DF: ABRIL / 2002

**UNIVERSIDADE DE BRASÍLIA
FACULDADE DE TECNOLOGIA
DEPARTAMENTO DE ENGENHARIA ELÉTRICA**

TÍTULO DA MONOGRAFIA, DISSERTAÇÃO OU TESE

NOME DO ALUNO AUTOR

PROJETO FINAL DE GRADUAÇÃO SUBMETIDO AO DEPARTAMENTO DE ENGENHARIA ELÉTRICA DA FACULDADE DE TECNOLOGIA DA UNIVERSIDADE DE BRASÍLIA, COMO PARTE DOS REQUISITOS NECESSÁRIOS PARA A OBTENÇÃO DO GRAU DE ENGENHEIRO.

APROVADO POR:

**HUMBERTO ABDALLA JÚNIOR, Docteur, UnB
(ORIENTADOR)**

**LEONARDO R. A. X. MENEZES, Doutor, UnB
(EXAMINADOR INTERNO)**

**ANTONIO J. M. SOARES, Doutor, UnB
(EXAMINADOR INTERNO)**

DATA: BRASÍLIA/DF, 20 DE ABRIL DE 2002.

FICHA CATALOGRÁFICA

BARBOSA, VLADIMIR DAIGELE. S.

Ambiente Multimídia como Suporte para o Ensino de Telecomunicações [Distrito Federal] 2002. ix , 58p., 297 mm ENE/FT/UnB, Engenheiro, Engenharia Elétrica, 2002).

Projeto Final de Graduação – Universidade de Brasília, Faculdade de Tecnologia. Departamento de Engenharia Elétrica.

1. Multimídia

2. Programas Autoria

3. Telecomunicações

I. ENE/FT/UnB.

II. Título (Série)

REFERÊNCIA BIBLIOGRÁFICA

BARBOSA, VLADIMIR DAIGELE S. (2002). Ambiente Multimídia como Suporte para o Ensino de Telecomunicações. Projeto Final de Graduação, Publicação XXX/02, Departamento de Engenharia Elétrica, Universidade de Brasília , Brasília , DF, 58p.

CESSÃO DE DIREITOS

NOME DO AUTOR: Vladimir Daigele Simões Barbosa

TÍTULO DA DISSERTAÇÃO: Ambiente Multimídia como Suporte para o Ensino de Telecomunicações.

GRAU/ANO: Engenheiro / 2002.

É concedida à Universidade de Brasília permissão para reproduzir cópias deste Projeto Final de Graduação e para emprestar ou vender tais cópias somente para propósitos acadêmicos e científicos. O autor reserva outros direitos de publicação e nenhuma parte desta dissertação deste Projeto Final de Graduação ser reproduzido sem a autorização por escrito do autor.

Vladimir Daigele Simões Barbosa

Dedicatória : aos meus pais, minhas avós e familiares.

RESUMO

Este projeto final de graduação objetiva mostrar algumas ferramentas utilizadas no desenvolvimento de ambientes multimídla para suporte ao ensino, trazendo casos práticos ligados à telecomunicação.

ABSTRACT

The work described in this undergraduate final project aims to show some tools used for multimedia environments development in learning support, bringing practical cases concerning telecommunication.

ÍNDICE

1 INTRODUÇÃO.....	1
2 LINGUAGENS DE PROGRAMAÇÃO UTILIZADAS NA PRODUÇÃO DE FERRAMENTAS EDUCACIONAIS.....	3
2.1 História da Interface Gráfica do Utilizador (GUI).....	3
2.2 Programação Orientada a Objeto (POO).....	5
2.2.1 Classes e Objetos.....	5
2.2.2 Herança de tipos Existentes.....	6
2.2.3 Ligação Tardia e Polimorfismo.....	7
2.3 JavaScript e Applet.....	7
2.3.1 URLs.....	8
2.3.2 Obtenção de arquivos multimídia.....	8
2.4 ActiveX.....	9
2.4.1 Controles ActiveX.....	10
2.5 Banco de Dados SQL.....	10
3 SOFTWARE DE AUTORIA.....	12
3.1 Aplicações: Ambiente de Suporte ao Ensino Presencial. Caso de Estudo CD Comunicações Ópticas.....	14
3.1.1 Objetivo.....	14
3.1.2 Seminários Através da Videoconferência.....	15
3.1.3 Estrutura do Programa de Autoria.....	15
3.1.4 Resultados.....	19
3.2 Aplicação: Software Interativo Análise e Síntese de Filtros Passivos	20
3.2.1 Introdução	20
3.2.2 Estrutura do Ambiente Multimídia.....	22
3.2.3 Utilização do Software.....	24
3.2.4 Resultados.....	27
4 STREAMING.....	28

4.1 Formas de Transmissão	29
4.1.1 Transmissão unicast.....	29
4.1.2 Transmissão Broadcast- Unicast.....	30
4.1.3 Transmissão Multicast.....	31
4.2 Videostreaming.....	32
4.3 Rastreamento de Dados.....	33
4.4 Métodos de Rastreamentos de Dados	35
4.4.1 Rastreamento Baseado em Arquivo.....	35
4.4.2 Rastreamento de Dados via E-mail.....	36
4.4.3 Rastreamento de Dados Através de Banco de Dados.....	37
4.5 Tecnologia de Streaming em softwares de autoria Macromedia Authorware.....	38
4.6 Considerações.....	39
4.7 Experiência: Utilizando a tecnologia de Streaming sobre o software de autoria “Análise e Síntese de Filtros Passivos”	40
4.7.1 Objetivo.....	41
4.7.2 Estrutura do Tutorial via Web.....	41
4.7.3 Preparação do Material.....	42
4.7.4 Funcionamento da Aplicação.....	46
4.7.5 Resultados.....	49
5 CONCLUSÃO.....	50
6 BIBLIOGRAFIA.....	52
7 ANEXO A.....	54
8 ANEXO B.....	56
.....	58

ÍNDICE DE TABELAS

TABELA 3-1. FASES DE PRODUÇÃO DE MATERIAL MULTIMÍDIA [13].....	13
TABELA 4-2. PRÓS E CONTRAS RASTREAMENTO BASEADO EM ARQUIVO.....	36
TABELA 4-3. PRÓS E CONTRAS O RASTREAMENTO DE DADOS POR E-MAIL.....	37
TABELA 4-4. PRÓS E CONTRAS O RASTREAMENTO DE DADOS ATRAVÉS DE BANCO DE DADOS.....	38
TABELA 4-5. TEMPO DE ACESSO EM DIFERENTES TAXAS.....	40
TABELA 4-6. EQUIPAMENTOS, SOFTWARES E ARQUIVOS.....	45

ÍNDICE DE FIGURAS

FIGURA 2-1. AS SETAS INDICADAS SÃO CONTROLES ACTIVEX.....	10
FIGURA 3-2. ESTRUTURA DO CD-ROM COMUNICAÇÕES ÓPTICAS.....	14
FIGURA 3-3. TELA PRINCIPAL DO CD DE COMUNICAÇÕES ÓPTICAS.....	15
FIGURA 3-4. APRESENTAÇÃO DE SLIDES NO PROGRAMA DE AUTORIA.....	16
FIGURA 3-5. CONTEÚDO DO ÍNDICE CAD.....	17
FIGURA 3-6. PALESTRAS EM VIDEOCONFERÊNCIA.	18
FIGURA 3-7. LINKS E TELA DE ENCERRAMENTO.....	18
FIGURA 3-8. ESTRUTURA DO PROGRAMA INTERATIVO.....	21
FIGURA 3-9. TELA PRINCIPAL.....	22
FIGURA 3-10. ETAPAS DO AMBIENTE DA ANÁLISE E SÍNTESE DE FILTROS.....	23
FIGURA 3-11. TELA PRINCIPAL DE UM SOFTWARE DE PROJETO.....	25
FIGURA 3-12. RESPOSTA DE UMA SAÍDA TÍPICA.....	26
FIGURA 4-13. TRANSMISSÃO UNICAST.....	30
FIGURA 4-14. TRANSMISSÃO BROADCAST.....	31
FIGURA 4-15. TRANSMISSÃO MULTICAST.....	32
FIGURA 4-16. SAÍDA TÍPICA DE UM VIDEOSTREAMING.....	33

FIGURA 4-17. TELA COM A ATUALIZAÇÃO SOFRIDA PELO PROGRAMA DE AUTORIA.....	42
FIGURA 4-18. TOPOLOGIA DA INTRANET.....	43
FIGURA 4-19. DIRETÓRIO NO SERVIDOR DE STREAMING ONDE ESTÃO OS ARQUIVOS DA EXPERIÊNCIA.....	44
FIGURA 4-20. COMANDOS SQL SENDO EXECUTADOS PELO AUTHORWARE.....	46
FIGURA 4-21. TELA DE LOG-ON PARA ACESSO AO TUTORIAL.....	47
FIGURA 4-22. BANCO DE DADOS DO SISTEMA.....	48
FIGURA 4-23. PÁGINA DINÂMICA DE ADMINISTRAÇÃO.....	49

1 INTRODUÇÃO

A técnica de combinar, de um modo interativo, fundamentos teóricos, visualização de conceitos abstratos, aplicações práticas, e uso de softwares de simulação referidos a um tópico em particular, têm promovido vantagens significantes no incremento de um processo de aprendizado eficiente. Uma transmissão integrada do conhecimento é possível por causa da interatividade dos elementos citados anteriormente, onde mais responsabilidade pode ser requerida dos estudantes, aumentando assim suas responsabilidades e o encorajamento ao auto-aprendizado.

Os estudantes podem selecionar as informações que lhe forem pertinentes ao acessar, num determinado momento, apenas o conteúdo de suas vontades, e então, executar simulações nos tópicos de escolha.

Aplicativos para o aprendizado multimídia são bastante diversos, e sua construção abrange faixas que vão deste de tutorial constituídos de gráficos e textos até softwares sofisticados ou simuladores de hardwares. Nenhum software encontra essa faixa tão variada de aplicação, por essa razão que, desenvolvedores de aplicativos de aprendizagem utilizam combinações de vários outros aplicativos para construir seus cursos. Por exemplo, a inserção de uma animação produzida pelo Macromedia Flash sendo executada dentro de uma aplicação em Authorware, que por sua vez possui applets, slides, vídeos, imagens, sons, softwares de simulação, sendo executados internamente.

O ambiente educacional desenvolvido consiste da construção de aplicativos com função de reunir todo um material multimídia formatado para permitir a livre navegação do estudante e prover total interatividade, em uma seqüência que parte dos conceitos básicos sobre o assunto e progride em complexidade. Permite também ferramentas computacionais que ilustram por simulações ou animações conceitos que reforçam os conceitos.

Os recursos que a multimídia oferece são complementos para aulas expositivas. Este binômio mostra-se eficiente, estabelecendo uma nova relação de

ensino/aprendizagem que permite o estudante a desenvolver sua formação em correspondência com seus interesses, suas necessidades no seu próprio ritmo.

No capítulo 2 algumas das ferramentas e técnicas de programação as mais freqüentemente utilizadas neste trabalho são mostradas para a construção de softwares de simulação, e plataformas de controles e captura de dados.

O desenvolvedor de um aplicativo multimídia tem que conhecer as mídias com a qual ele terá que trabalhar, pois as diversas formas existentes de linguagens e construção das mídias podem acarretar na falta da junção das mesmas num mesmo programa de autoria. A interoperabilidade entre os elementos que compõem o programa é quesito fundamental, por exemplo, num mesmo programa de autoria podem estar presentes vídeos no formato Real Media e no formato Windows Media Player e que, ao mesmo tempo, apresenta documento de texto no formato Portable Document (pdf).

Para elucidar o conceito de programas de autoria o capítulo 3 traz informações e duas aplicações para apropriação de ferramentas sobre duas óticas diferentes de programas de ensino.

O capítulo 4 mostra a técnica de difusão do material multimídia via *Streaming* pela Web, e uma experiência é apresentada baseada no conceito de ambiente de aprendizado via Web.

Em um ambiente de aprendizado baseado em Web, o conteúdo está organizado na forma de vídeos, textos, animações, efeitos sonoros, applets, entre outros, disponíveis através de um web browser instalado no microcomputador com acesso para alunos e instrutores. O aprendizado via Web pode ser aplicado para fornecer o suporte a cursos presenciais ou via videoconferência. A utilização da Web como elemento de suporte pode oferecer ao aluno todo material didático, anotações de aula, exercícios, referências bibliográficas, facilidade de interação através de mural (guestbook), e-mail e grupos de discussão.

2 LINGUAGENS DE PROGRAMAÇÃO UTILIZADAS NA PRODUÇÃO DE FERRAMENTAS EDUCACIONAIS

São vários os tipos de linguagem de programação existente. Neste tópico, tenta-se ilustrar algumas das linguagens mais utilizadas na construção de ferramentas de simulação, análise, e que vêm sendo utilizadas desde longas datas para a criação de conteúdo educacional.

2.1 HISTÓRIA DA INTERFACE GRÁFICA DO UTILIZADOR (GUI)

Computadores e softwares de computadores operam sobre largos sistemas invisíveis que provêm poucas pistas físicas ou visuais do estado operacional ou organização do sistema (Norman 1993).

A proposta de desenvolvimento de uma interface gráfica é fornecer telas que criam a imagem de um ambiente operacional para o usuário, na formação de um contexto visual e funcional para as ações de usuários de computadores. A interface gráfica direciona, orchestra e foca a experiência dos usuários com base na estrutura organizacional dos sistemas de computadores ou documentos multimídia visíveis e acessíveis ao usuário.

Em meados dos anos 60 os sistemas computacionais eram terminais do tipo *typewriter-like teletype* (TTY) que usavam papel com forma de impressão visual, para imprimir ambas as instruções dos operadores e qualquer resposta resultante das atividades dos computadores. Desenvolvedores antecedentes de sistemas computacionais interativos usaram monitores de raio de tubos catódicos para criar gráficos e textos mostrados na tela . Esta modalidade é a base para o sistema operacional de comandos de linhas MS-DOS que ainda é utilizado amplamente na atualidade. Porém, mesmos os pesquisadores dos anos 60 tais como Ivan Sutherland (inventor da primeira tela de computador interativa por janelas) desenvolveram sistemas de impressão para monitores de tubo de raios catódicos que emulavam a complexidade gráfica de impressão de documentos e utilizavam os

caracteres dinâmicos na tela dos computadores para transcender as limitações gráficas da impressão em papel.

Durante os anos 70 o conceito básico para a maioria das interfaces gráficas para usuários foi desenvolvida pelo Centro de Pesquisa Palo Alto da Xerox (PARC). Estes conceitos incluíam transformações gráficas explícitas para objetos como documentos e programas de computadores, janelas de múltiplas sobreposições para subdividir atividades no monitor, e manipulação direta das janelas, ícones e outros objetos através da interface do mouse de Engelbart como um ponto ambulante (Smith 1982).

O trabalho do PARC na interface de computador para manipulação direta estava baseado nas observações cognitivas desenvolvidas pelos psicólogos Jean Piaget e Jerome Bruner no qual defendem que nossa compreensão do mundo é fundamentalmente ligada ao estímulo visual e experiências do tato em manipular objetos em nosso ambiente.

Em particular, o modelo de Bruner sobre o desenvolvimento humano era uma combinação de habilidades inatas (manipulação de objetos, conhecimentos da localização espacial), habilidades da memória (reconhecimento visual, comparação, contraste), e habilidades simbólicas (a habilidade de entender longas seqüências de estruturas abstratas) levaram os pesquisadores do PARC a tentar construir uma interface que explicitava a localização de todos estes três fundamentos de entendimento e manipulação do mundo que nos cerca. Os computadores (naquela época e agora) têm sempre exigido uma seqüência abstrata de tarefas; os trabalhos dos pesquisadores do PARC foi criar uma interface que também poderia explorar as habilidades visuais e de manipulação dos usuários.

Hoje em dia, a difusão e desenvolvimento destes e dos subseqüentes trabalhos na área são bastante visíveis para usuários de computadores e diferentes segmentos da sociedade, pois transformaram o modo de interagir com as máquinas, criaram para elas condições de se espalharem para o mundo. E, entre todas as aplicações de apelos visuais realizadas pelos computadores, a construção de ferramentas interativas para o ensino, não apenas de telecomunicações, vêm ganhando mais espaço a cada dia.

2.2 PROGRAMAÇÃO ORIENTADA A OBJETO (POO)

A maioria das linguagens de programação modernas oferece suporte à POO. Estas são baseadas em três conceitos fundamentais: encapsulamento (normalmente implementado com classes), heranças e polimorfismo (ou ligação tardia).

A POO tem a seguinte idéia: um programa é feito de objetos com certas propriedades e operações que os objetos podem realizar. O estado atual pode mudar com o passar do tempo, mas você vai depender sempre de que os objetos não interajam entre si de formas não documentadas. Se você vai elaborar ou comprar um objeto vai depender de orçamento ou de prazo. Porém, basicamente, enquanto os objetos atenderem às especificações esperadas, não é por que se preocupar em saber como a funcionalidade foi implementada. Na POO, você só se importa com o que o objeto expõe.

Este conceito auxilia na velocidade de construção e distribuição de aplicações, blocos separados, que podem ser desenvolvidos por pessoas diferentes, em épocas diferentes, e formam a estrutura completa.

A seguir segue uma breve introdução aos princípios fundamentais de programação orientada a objetos.

2.2.1 CLASSES E OBJETOS

Classe e objeto são dois termos normalmente usados em linguagens de POO. Uma classe é um tipo de dados definido pelo usuário, que tem um estado (sua representação) e algumas operações (seu comportamento). Uma classe possui alguns dados internos e alguns métodos, na forma de procedimentos ou funções, e normalmente descreve as características genéricas e o comportamento de vários objetos semelhantes.

Um objeto é uma instância de uma classe ou variável do tipo de dados definido pela classe. Os objetos são entidades *reais*. Quando o programa é executado, os objetos ocupam memória para sua representação interna. O

relacionamento entre objeto e classe é o mesmo que o existente entre variável e tipo.

2.2.2 HERANÇA DE TIPOS EXISTENTES

Freqüentemente, é preciso usar uma versão um pouco diferente de uma classe já existente que escrevemos ou que alguém forneceu. Por exemplo, pode ser preciso incluir um novo método ou mudar ligeiramente um método já existente. Com certa facilidade, pode-se modificar o código original. Mas, se a classe foi originalmente escrita por outra pessoa, talvez seja interessante manter as novas alterações em separado.

Uma alternativa típica é fazer uma cópia da definição de tipo original, mudar seu código para oferecer suporte aos novos recursos e dar um novo nome à classe resultante. Isso poderia funcionar, mas também poderia criar problemas: na duplicação do código, duplicam-se também os erros; e se a intenção é incluir um novo recurso, precisa-se incluí-lo duas ou mais vezes, dependendo no número de cópias do código original que será feito. Essa estratégia resulta em dois tipos de dados completamente diferentes; portanto, o compilador não pode ajudar a tirar proveito das semelhanças entre os dois tipos.

Para resolver esses tipos de problemas ao expressar semelhanças entre classes, o objeto pascal permite que se defina uma nova classe diretamente a partir de outra já existente. Essa técnica é conhecida como *herança* e é um dos elementos fundamentais das linguagens de programação orientadas a objetos.

2.2.3 LIGAÇÃO TARDIA E POLIMORFISMO

As funções e procedimentos de algumas linguagens são baseados em *ligação estática*. Isso significa que uma chamada de método é resolvida pelo compilador, que substitui a solicitação por uma chamada à posição de memória específica em que a função ou o procedimento reside. As linguagens de programação orientadas a objetos permitem o uso de outra forma de ligação, conhecida como *ligação dinâmica* ou *ligação tardia*. Nesse caso, o endereço real do método a ser chamada é determinado em tempo de execução, com base no tipo de instância usada para fazer a chamada.

A vantagem desta técnica é conhecida como *polimorfismo*. Polimorfismo significa que é possível escrever uma chamada para um método, aplicando-a em uma variável.

2.3 JAVASCRIPT E APPLETS

JavaScript é uma linguagem orientada a objeto desenvolvida para aplicações cliente-servidor para a Internet. Os web-browsers interpretam o código JavaScript vinculado em página HTML (Hyper Text Markup Language).

Applet é um tipo especial de programa Java no qual um navegador é capaz de executar descarregando-o a partir da Internet. Applets com muitos recursos dependem da habilidade do programador em dominar os recursos de rede da linguagem Java junto com sua habilidade de lidar com múltiplas linhas de execução.

As applets podem lidar com imagens e com áudio. As imagens podem ser, por exemplo, no formato GIF ou JPEG e os arquivos de áudio nos formatos AU, AIFF, WAV ou MIDI. Os GIFs animados também são aceitos e a animação é exibida normalmente. Geralmente, os arquivos contendo esse tipo de informação são especificados na forma de URLs.

Antes da linguagem Java, usava-se HTML para descrever o layout de uma página Web. O HTML é simplesmente um veículo para indicar os elementos de uma página de hipertexto. Por exemplo, <TITLE> indica o título da página e qualquer texto que vier depois da marca (tag), será o título da página. Indica-se o final do título com a marcação </TITLE>.

A idéia básica sobre como usar applets numa página Web é simples: a página HTML precisa informar ao navegador quais applets devem ser carregadas e onde colocar cada um na página Web. Como é de se esperar, a marca HTML Java necessária para usar um applet Java precisa informar aos navegadores capazes de executar Java o seguinte:

- O nome do arquivo de classe
- A localização dos arquivos de classe
- Como as applets devem ficar na página Web (tamanho, localização etc.)

Em seguida, o navegador obtém os arquivos das classes pela Internet (o de um diretório na máquina do usuário) e executa automaticamente a applet, usando sua máquina virtual Java.

Além da própria applet, a página Web pode conter todos os outros elementos HTML que se vêem em uso em muitas páginas Web: fontes múltiplas, listas com marcadores, figuras, link etc. As applets são apenas uma parte da página de hipertexto.

2.3.1 URLs

Um URL na verdade não é nada mais que a descrição de um recurso na Internet. Por exemplo, <http://java.sun.com/index.html> informa ao navegador para usar o protocolo de transferência de hipertexto no arquivo index HTML localizado em java.sun.com.

2.3.2 OBTENÇÃO DE ARQUIVOS MULTIMÍDIA

Podem-se obter arquivos de imagens e áudio usando os métodos *getImage* e *getAudioClip*. Por exemplo:

```
Image cat = getImage (getDocumentBase(), "images/cat.gif");
AudioClip meow = getAudio Clip (getDocumentBase(),
"audio/meow.au");
```

O método `getDocumentBase` retorna o URL a partir do qual a applet é carregado

É bom salientar que imagens e clipes de áudio precisam estar localizados no mesmo servidor que hospeda o código da applet. Por motivos de segurança, as applets não podem acessar nenhum arquivo de outro servidor.

2.4 ACTIVE X

ActiveX é um conjunto de novas tecnologias desenvolvida pela Microsoft desde meados de 1996. Em uma frase simples, ActiveX é um conjunto linguagem-neutra de tecnologias de integração que capacitam componentes de softwares distintos trabalharem juntos em um ambiente interconectado. Isso significa dizer que independente da linguagem adotada para o desenvolvimento de uma aplicação, os componentes ainda possuem interoperabilidade e comunicações válidas. ActiveX define o empacotamento de componentes de software e como estes componentes devem se comunicar. Como resultado, pode-se listar sobre os componentes ActiveX:

- Podem ser criados em qualquer linguagem
- Usados com qualquer outra aplicação
- Integrados com qualquer outro script de linguagem
- Executados em qualquer sistema operacional

Estas qualidades do ActiveX permitem que os desenvolvedores tenham a capacidade de criar facilmente aplicações interativas, utilizando para isto, uma estrutura de desenvolvimento aberta.

O conjunto de tecnologias ActiveX forma uma estrutura para a criação de conteúdo interativo Web através de componentes de softwares, scripts, e aplicações existentes. Especificamente, a tecnologia ActiveX torna os provedores de conteúdo e desenvolvedores de aplicação capazes de criarem facilmente conteúdo Web dinâmico e extensões de servidores Web pelo uso de controles ActiveX.

2.4.1 CONTROLES ACTIVE X

Controles ActiveX são componentes de softwares reutilizáveis criados por desenvolvedores de softwares, e têm fácil integração em páginas Web e outros aplicativos

Os controles activeX provêm uma maneira instantânea e dinâmica de se adicionar funções a aplicações Web. Existem controles que permitem integrar praticamente qualquer tipo de mídia (sons, vídeos, animações, aparelhos de entrada interativos, etc.) em outras aplicações, incluindo aplicações Web.

Existem vários tipos controles ActiveX, a Figura 2-1 mostra um controle ActiveX desenvolvido em VBScript sendo executado dentro de outra aplicação (Macromedia Authorware). Este controle tem a função de interagir com uma apresentação do PowerPoint. Resumidamente, no programa de autoria ilustrado pela figura estão uma apresentação de slides sendo executada por um controle ActiveX do navegador Internet Explorer, e outro externo, inserido também no programa de autoria, executando comandos sobre a apresentação.

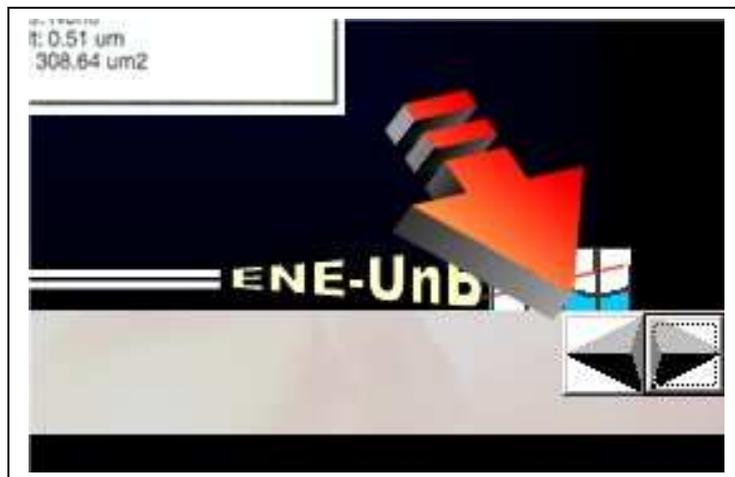


Figura 2-1. As setas indicadas são controles ActiveX.

2.5 BANCO DE DADOS SQL

Servidores SQL (Structured Query Language) usam um tipo de banco de dados chamado de banco de dados relacional.

Banco de Dados Relacionais são dados que são organizados em tabelas formando o banco de dados. Tabelas são organizadas por grupos de dados sobre o mesmo tipo de objeto. Cada tabela é, então, relatada para as outras pelo centro do banco de dados quando solicitado.

Um banco de dados pode geralmente ser pensado como uma coleção de dados relacionados. Em estruturas de banco de dados anteriores, um banco de dados era comumente apenas um arquivo – algo tipo empregados.abd – que continha uma simples tabela de dados. Colunas relacionando dados aos empregados tais como salário, data de contratação, nome, identidade, e outros dados, poderiam estar dentro do arquivo empregados.abd. Poderia existir um linha para cada pessoa na companhia, com os valores correspondentes nas colunas apropriadas. Índices, utilizados para aumentar a velocidade de acesso aos dados, eram mantidos em arquivos diferentes, bem como não existia nenhuma segurança.

Em servidores SQL, um banco de dados não é necessariamente amarrado a um arquivo, os dados são mais um conceito lógico baseado em uma coleção de objetos relacionados. Por exemplo, em SQL não existe apenas dados crus, ele também possui segurança, estrutura, índices, e pode conter outros objetos, tais como amostras ou processos relacionados a outro banco de dados em particular.

3 SOFTWARE DE AUTORIA

Os softwares de autoria são ferramentas que integram textos, imagens, áudio e vídeo. O material completo, então, possui um enorme volume de dados, e, para a sua disponibilização emprega-se tanto para a forma de produção do material para se entregue ou pela Web ou através de CD-ROM.

A produção de toda a informação que integra um software de autoria passa primeiro pela coleta do conteúdo multimídia. Estes, então, devem ser disponibilizados na forma digital. Ou seja, obtenção e editoração de imagem, gravação de vídeo, composição da trilha sonora, inserção efeitos, etc.

Tendo-se em mãos os arquivos previamente formatados. Estes podem ser finalmente combinados para que sejam apresentados de acordo com a programação de projeto, estabelecendo, inclusive, as formas de interação com o usuário através dos programas de autoria.

Tendo-se o programa que utiliza os recursos da multimídia interativa, de acordo com as especificações iniciais, teve-se cuidar da forma como os usuários terão acesso a esse material, ou seja, sua distribuição.

Podem-se definir duas formas de distribuição de conteúdos interativos: as “redes” (Internet / Intranet) e os “meios locais” (CDs, DVDs, etc.). devido à limitação e variação na velocidade da Internet, deve-se produzir programas de autorias com tamanhos para a transmissão da informação, implicando numa subtilização de recursos multimídia. Para as Intranets os arquivos podem, tomar tamanhos maiores, já que se tem garantido uma transferência de informação maior pela rede. Este tipo preocupações, quanto às condições do meio de transmissão, não existem quando a distribuição é feita através dos chamados meios locais, a Tabela 3-1 mostra os conteúdos e as fases de produção de material multimídia.

Tabela 3-1. Fases de produção de material multimídia [13].

Meio Externo	Computador						Distribuição
	Digitalização	Arquivos Brutos	Edição	Arquivos Finais	Autoria	Multiplicação	
Fita k7 (locuções)	Placa de som (kit Multimídia)	.WAV .VOC	Sound Forge Grav. de Som	.WAV .VOC	Multimídia: Auhtorware Director Internet: Dreamwaver FrontPage Apresentações: PowerPoint Textuais: Acrobat Folio FrameMaker	Gravadora de CD	CD-ROM
Fita VHS (vídeo)	Placa digitalizadora de vídeo	.AVI .MOV	Adobe Premiere	.AVI .MOV			
Papel (fotos)	Scanner	.TGA .TIF	Fireworks Photoshop	.TGA .TIF		Gravadora de disquetes	Disquetes
Cromos (fotos)	Scanner com adaptador	.TGA .TIF	Fireworks, Photoshop	.TGA .TIF			
CDs (trilhas sonoras)	Placa de som (kit multimídia)	.WAV .VOC	Sound Forge Grav. de Som	.WAV .VOC		Redes	Internet
Textos	Scanner / OCR	.DOC	Word	.DOC			Intranet
Animações	Criadas no computador		Flash / 3D Studio	.SWF .FLC		Disco rígido	quiosques
Textos	Digitados no computador			.DOC			
Trilha sonora	Criadas no computador			.MID			

Os softwares de autoria capacitam seus usuários a interagirem com a informação trazida em seu conteúdo e a progredir num determinado assunto no tempo estipulado pela própria pessoa.

Este controle passa a ser apresentado com um simples índice, selecionando o assunto específico de interesse

Existe uma grande variedade de ferramentas de autoria no mercado. O que se precisa entender é que toda ferramenta tem seu desenvolvimento para uma aplicação específica, dependendo do tipo de usuário, e podem ser classificadas como especializadas e de propósito geral.

O software de autoria utilizado neste trabalho para a construção das ferramentas de treinamento é o Authorware da empresa norte-americana Macromedia, e é classificado como um produto de um caráter especializado. Desenvolvido sobre uma plataforma de não-programadores que intencionam criar conteúdo de aprendizado interativo.

3.1 APLICAÇÕES: AMBIENTE DE SUPORTE AO ENSINO PRESENCIAL. CASO DE ESTUDO CD COMUNICAÇÕES ÓPTICAS

3.1.1 OBJETIVO

Através de um ambiente de aprendizagem composto por aulas presenciais, seminários em videoconferência, apresentações em slides, programas de simulação de conceitos e dispositivos ligados à disciplina Comunicações Ópticas ministrada no departamento de Engenharia Elétrica no segundo semestre 2000 em linguagens orientadas a objetos, e apresentação de páginas da Internet, criou-se um tutorial eletrônico em CD-ROM contendo as informações ilustradas pela Figura 3-2. O CD-ROM serve como uma memória eletrônica aos alunos de todo o curso, sendo possível acessá-la a qualquer momento.

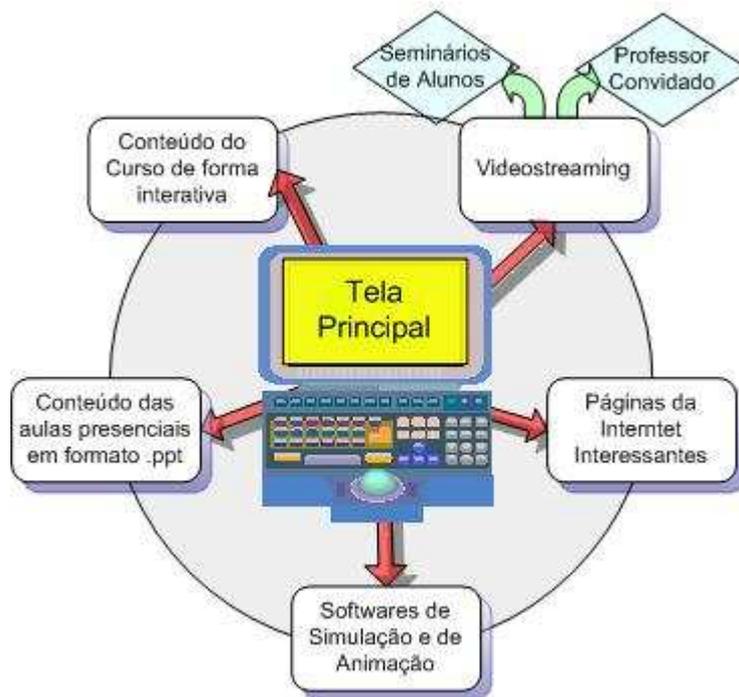


Figura 3-2. Estrutura do CD-ROM comunicações Ópticas

3.1.2 SEMINÁRIOS ATRAVÉS DA VIDEOCONFERÊNCIA

Os alunos do Curso de Comunicações Ópticas foram convidados a apresentarem seus trabalhos da disciplina em videoconferência para o resto da turma, e através dessa atitude, executaram-se as gravações dos seminários realizados pelos alunos de graduação do curso de Comunicações Ópticas em formato digital, a plataforma de compressão de vídeo utilizada foi a do fabricante Real Media. O fornecimento do conteúdo da disciplina, além de através da ferramenta de autoria Authorware, e construção do CD-ROM foi realizada obedecendo à estrutura proposta.

3.1.3 ESTRUTURA DO PROGRAMA DE AUTORIA

O CD – Comunicações Ópticas em sua estrutura pode ser apresentado a seguir:

Sobre a tela inicial do CD o usuário pode ter acesso a todo o conteúdo fornecido através dos menus mostrados pela Figura 3-3 .



Figura 3-3. Tela principal do CD de Comunicações Ópticas.

O índice Aulas do CD trás as transparências utilizadas pelos professores durante o curso. As transparências seguem uma ordem cronológica. Os alunos têm dessa forma acesso ao recurso na sua forma original com todas as animações utilizadas para ilustrar e promover a melhor compreensão dos novos conceitos. A

técnica utilizada para capacitar a visualização de um arquivo PowerPoint do Windows no programa de autoria, baseia-se na idéia de se utilizar o componente ActiveX do navegador Internet Explorer carregando a apresentação de slides. E, associado aos slides, para aumentar a interação dos usuários, foi criada uma ferramenta de controle na forma de ActiveX, em linguagem VBScript, e incorporada à estrutura do Authorware (Figura 3-4).

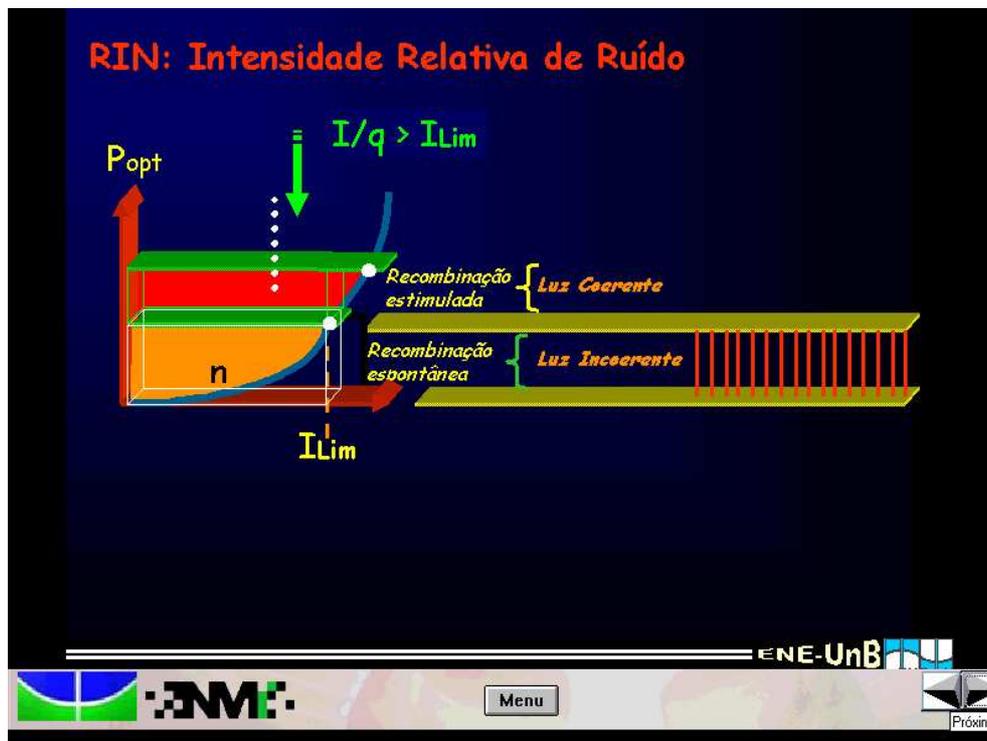


Figura 3-4. Apresentação de slides no programa de autoria.

No índice CAD são apresentados três softwares que consistem de: applets que ilustram conceitos básicos de comunicações ópticas. Um software de animação fornecido pela empresa Corning, do ramo de comunicações ópticas, que mostra diversos conceitos e processos na fabricação de componentes e matéria prima. Um programa de autoria com o software da Universidade de Brasília (Figura 3-5) no qual simula um link de fibra óptica sobre diferentes condições determinadas pelo interesse do usuário do programa.

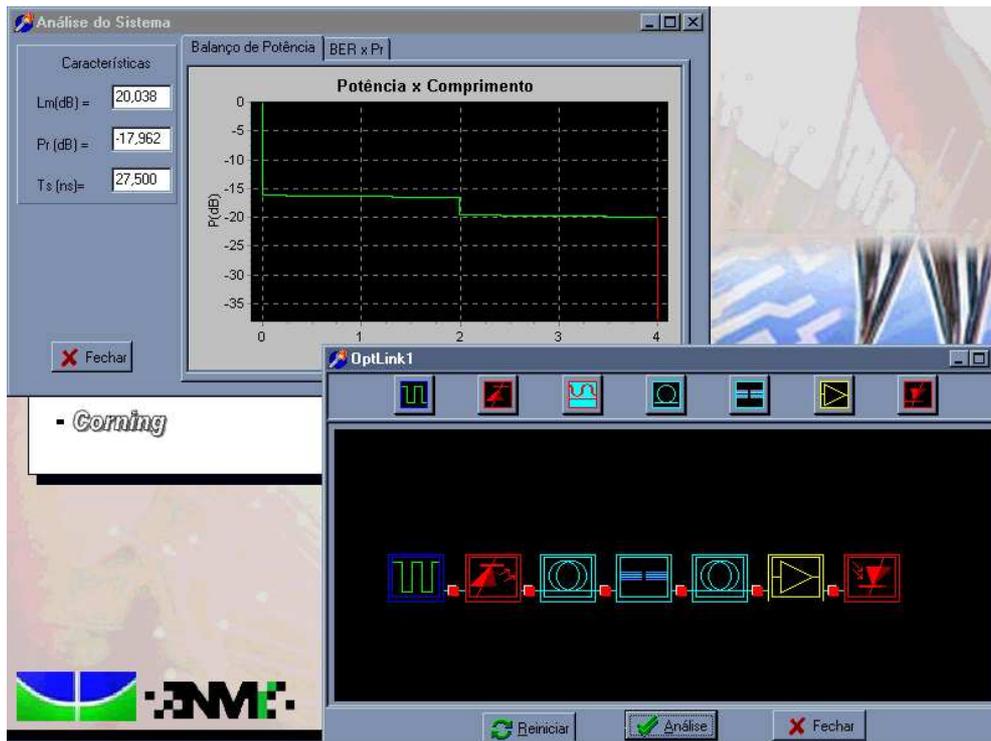


Figura 3-5. Conteúdo do índice CAD

Os seminários dos alunos foram realizados através da videoconferência, e, é mostrada a condição de interatividade nesta modalidade de transmissão de conhecimento que vence a barreira de longas distâncias. A forma como são apresentados os seminários é apresentada pela Figura 3-6. Os alunos palestrantes estavam no prédio da Faculdade de Tecnologia, e, o resto da turma no prédio SG-11, ambos, localizados no campus da Universidade de Brasília. As condições de tráfego e velocidade de transmissão da rede utilizada foram bastante positivas para as apresentações, podendo considerá-la, para este caso, como uma rede local. A taxa média da conexão estabelecida entre as duas estações de videoconferência era de aproximadamente 768 Kbps, taxa esta, considerada excelente para uma videoconferência no qual será apresentada uma palestra. A interatividade é garantida, como ilustrado na Figura 3-6:



Figura 3-6. Palestras em Videoconferência.

Como fonte bibliográfica e de pesquisa é apresentada uma página Web com links de acesso a informação de referência indicada pelo professor. A Figura 3-7 mostra a tela de encerramento do CD com o nome de todos os alunos que contribuíram para o conteúdo do CD.

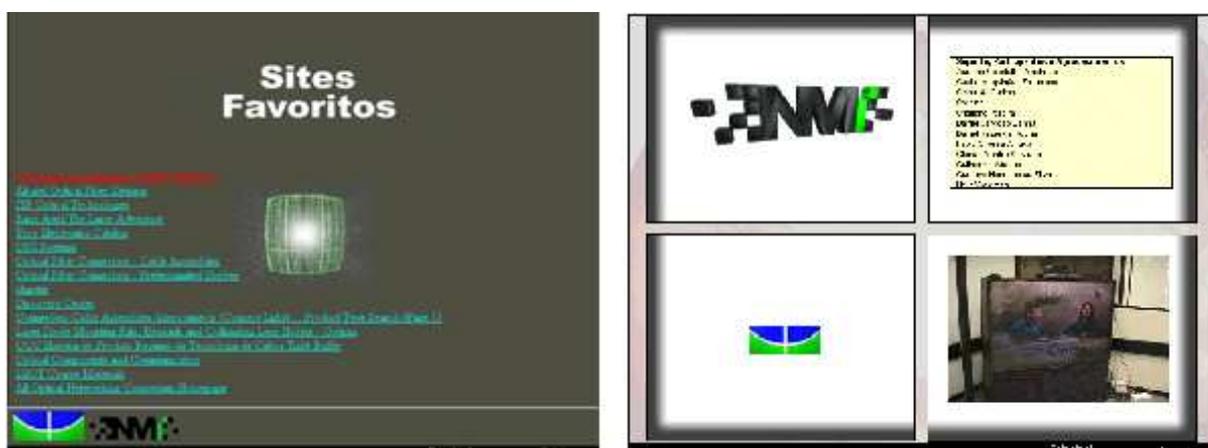


Figura 3-7. Links e tela de encerramento.

3.1.4 RESULTADOS

A ferramenta, além de ter a característica de memória eletrônica para os agentes que contribuíram para a constituição do CD-ROM – alunos e professores - apresenta-se como uma multimídia que fornece elementos para uma concepção integrada de transmissão do conhecimento no processo de auto-aprendizagem para os que se interessarem a ingressar no assunto de comunicações ópticas.

3.2 APLICAÇÃO: SOFTWARE INTERATIVO ANÁLISE E SÍNTESE DE FILTROS PASSIVOS

A aplicação a seguir descreve um ambiente educacional para a análise e síntese de filtros passivos que combina as vantagens de tutoriais multimídia com softwares de simulação.

3.2.1 INTRODUÇÃO

O ambiente educacional desenvolvido para o software interativo de Análise e Síntese de Filtros Passivos, consistiu-se de um tutorial que aborda os conceitos básicos de circuitos elétricos e suas aplicações com a teoria de filtros, e contém em sua estrutura softwares sobre análise e síntese de filtros passivos (Figura 3-8). A integração dos aplicativos de simulação é de grande importância na caracterização do ambiente proposto. O software torna possível para os estudantes testarem e compararem os vários tipos de filtros e os induz a proporem novas topologias que os forcem a usarem o novo conhecimento adquirido. Este tutorial multimídia, além de conter todas as bases teóricas necessárias para o desenvolvimento de filtros passivos, mostra ao estudante através de animações e simulações, como usar de forma adequada os softwares. A estrutura adotada contém animações freqüentes que são usadas para explicar e ajudar os estudantes a visualizarem os vários conceitos associados com a teoria de análise de circuitos temporal e espectral.

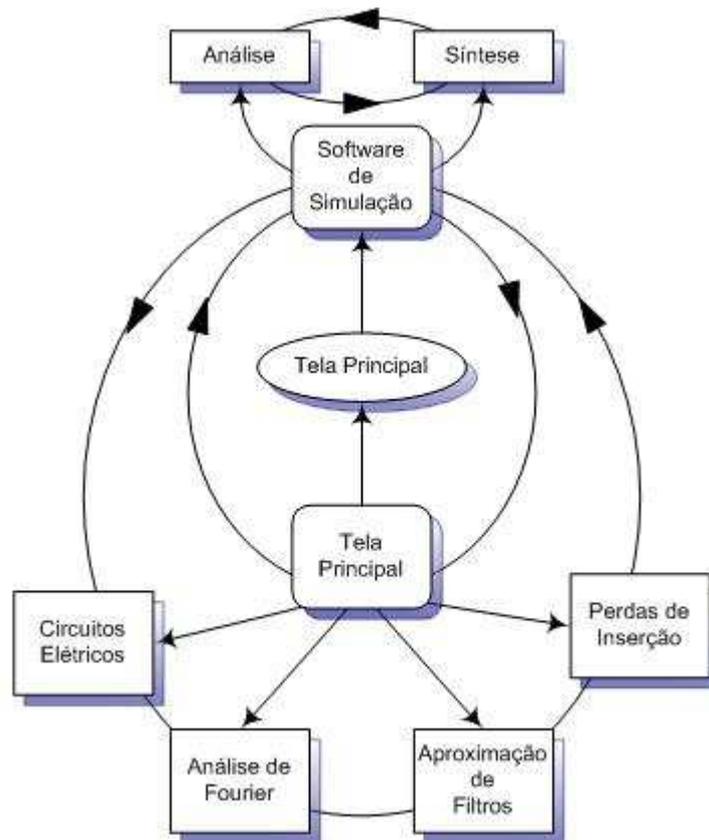


Figura 3-8. Estrutura do programa interativo.

O tutorial descreve, de uma maneira esquemática e detalhada, a teoria envolvida no desenvolvimento de filtros. O acesso à informação é obtido em quatro módulos onde os tópicos a seguir são relacionados:

- Filtros Elétricos – Conceitos Básicos, Análise de Resposta, Características e Ressonância, Aproximação de Butterworth, Aproximação de Chebyshev, Curva Ressonante Universal.
- Análise de Fourier – Séries de Fourier, Análise de Formas de Ondas Periódicas, Espectro de Frequência Discreto, Integral de Fourier, e Espectro Contínuo
- Aproximação de Filtro – O caso geral Filtro-Passa-Baixa, Efeito de Parâmetros na Resposta de Chebyshev, Comparação entre os filtros de Butterworth e Chebyshev, e Resposta ao Impulso e ao Degrau.
- Método de Perda de Inserção – Funções de Rede, Funções de Transformação, Algumas Características Úteis, Relações Interessantes, Expansão de Frações Continuadas, Procedimento de Síntese.

Além de tutoriais, cada módulo também inclui aplicativos de simulação, gráficos de duas e três dimensões, e animações, que são utilizadas para ajudar os estudantes a adquirir a compreensão adequada dos tópicos relacionados. A Figura 3-9 mostra a tela Principal do Ambiente, da qual é possível acessar os módulos multimídia.

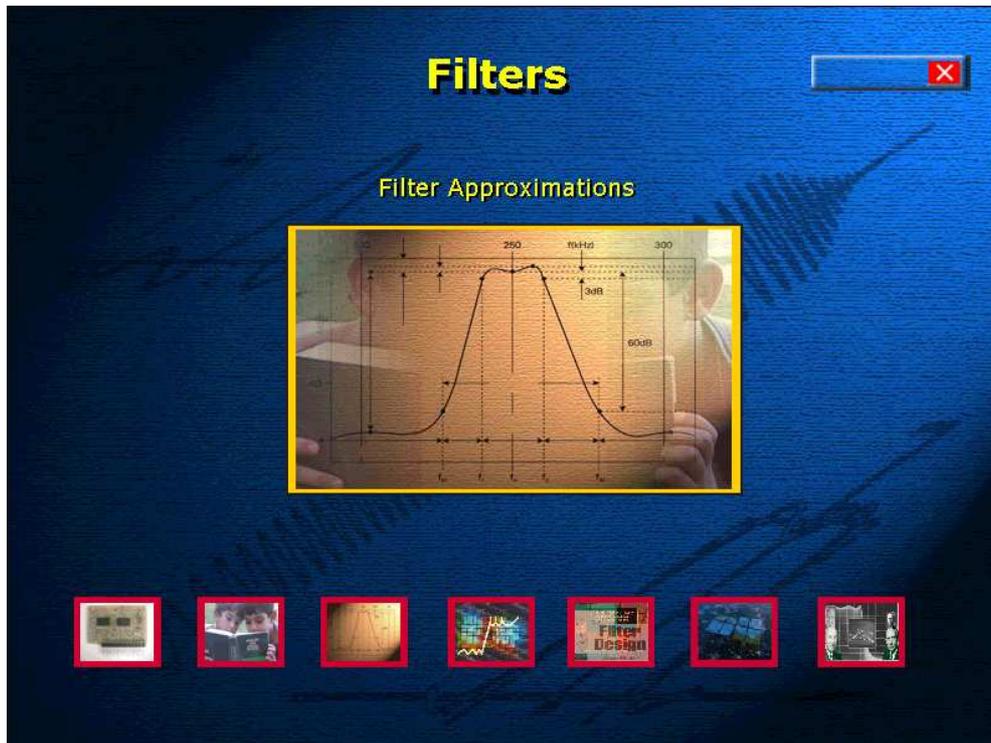


Figura 3-9. Tela principal.

3.2.2 ESTRUTURA DO AMBIENTE MULTIMÍDIA

Apesar de todo o tutorial e softwares serem formatados para permitirem a livre navegação do estudante pelo mesmo, o ambiente de aprendizado foi desenvolvido hierarquicamente em quatro níveis diferentes. O primeiro nível introduz o conceito de filtros elétricos, exemplos de filtros elétricos e suas características de ressonância. No segundo nível o estudante passa a se familiarizar com as ferramentas matemáticas que contribuem para a associação e interpretação dos sinais nos domínios do tempo e da frequência. No terceiro nível, com a ajuda da Análise de Fourier, a teoria de filtros é estudada e os parâmetros principais são definidos. Os filtros são caracterizados pelas suas funções de transferência e pela localização de seus pólos e zeros no plano de frequência complexo "s". No quarto

nível a Síntese de Filtros é invocada. Imagine que o problema da aproximação foi resolvido e a função de transferência foi obtida. A realização dos filtros baseados em uma dada função de transferência é acompanhada pelo método das perdas por inserção. O procedimento da transformação da frequência é explicado e aplicado para obter a resposta na frequência desejada. Neste ponto os conceitos apresentados no módulo anterior são utilizados exhaustivamente. O processo interativo permite o treinamento para o acesso aos módulos anteriores na hora em que o aluno achar necessário. A etapa final é dedicada para o uso dos softwares de simulação onde o estudante aplica os métodos de análise e síntese os quais ele estudou. A Figura 3-10 ilustra o princípio das atividades.

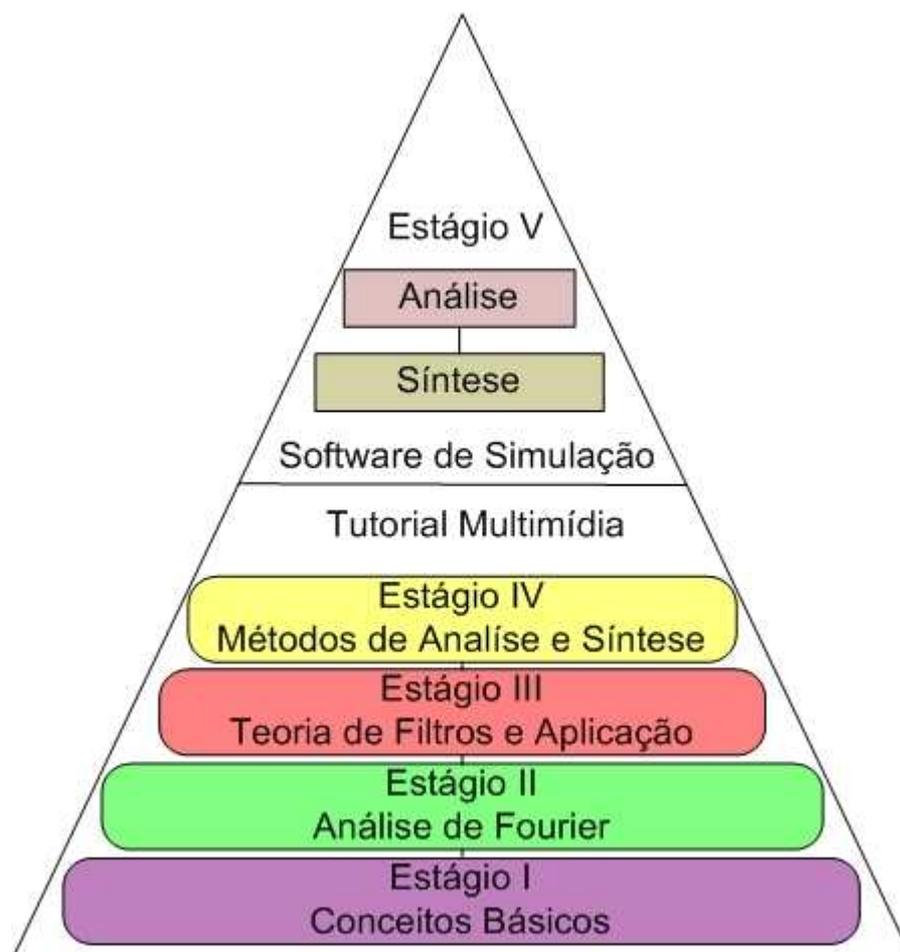


Figura 3-10. Etapas do ambiente da análise e síntese de filtros.

3.2.3 UTILIZAÇÃO DO SOFTWARE

Durante o processo de aprendizagem, o tutorial ilustra alguns aspectos relevantes da teoria de filtros por exemplos de simulação. Essas simulações em muitos casos utilizam a mesma estrutura que o software principal. Assim o estudante aprende passo a passo, a maneira correta de utilização do software, e o que ele é capaz de simular. A estrutura interativa do software permite que os usuários acessem todos os conceitos que foram previamente estudados, no atual ambiente de simulação. O programa de autoria é composto de quatro módulos independentes que executam as seguintes funções:

- Módulo de introdução que apresenta os conceitos básicos de filtros elétricos, suas formas primárias, análise de resposta impulsional e ao degrau, característica e ressonância e os softwares: Parâmetros de Onda, Fasores, Passa-baixa RLC Ressonante, Curva Ressonante Universal, Butterworth e Chebychev e o software Filtros de Segunda Ordem e suas características.
- Módulo de Construção da função de transferência. Série de Fourier, Transformada de Fourier, applet sobre o tema e links que encaminham o aluno a pesquisar sobre o tema.
- Visualização de gráficos e módulos de análise. Este módulo permite a visualização de pólos e zeros de filtros no plano s (plano complexo), respostas no tempo e simulação de canal.
- Módulo de Síntese dos filtros por perda de inserção. Investigação das funções de redes, expansão de frações continuadas e processos de síntese.

Todas as funções dos softwares podem ser acessadas por uma tela simples composta de quatro ambientes: entrada de dados do projeto, escolha de um tipo desejado de resposta, e visualização do resultado solicitado. A estrutura da tela do software e o protótipo de um passa baixo normalizado é mostrado pela Figura 3-11.

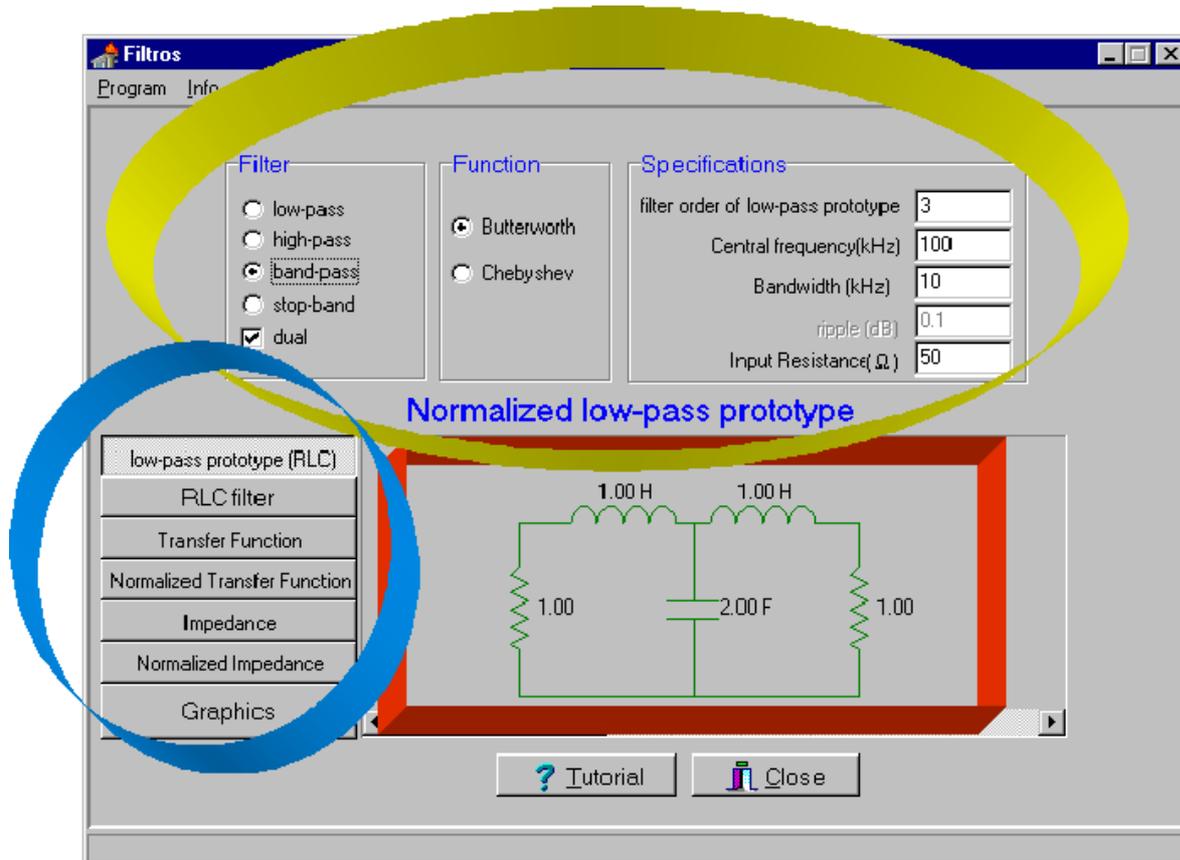


Figura 3-11. Tela principal de um software de projeto.

Um desenvolvimento passo a passo usando o software interativo é descrito abaixo, tomando como exemplo software desenvolvido pelo aluno do departamento de Engenharia Elétrica Eudes de Assunção [15]:

1. Seleção do tipo de filtro e do tipo da resposta em frequência. Completar as especificações do filtro
2. Com as especificações determinadas, escolhe-se uma das seguintes funções:
 - Função de Transferência Normalizada, Impedância Normalizada, e Protótipo Passa Baixo Normalizado
 - Função de Transferência, Impedância, e Filtro RLC.

A Resposta típica de saída é mostrada pela Figura 3-12, na qual é possível ver o Filtro RLC, a Função de Transferência Normalizada, e a Impedância Normalizada.

3. Quando as funções Gráficas são acessadas, é possível observar a característica da função de transferência e as impedâncias de entrada como funções da frequência. A interface gráfica permite a visualização da resposta no domínio do

tempo pelas respostas do impulso e do degrau. A eficiência do filtro pode ser analisada pelo retorno das curvas de perdas e pelas taxas de onda estacionárias.

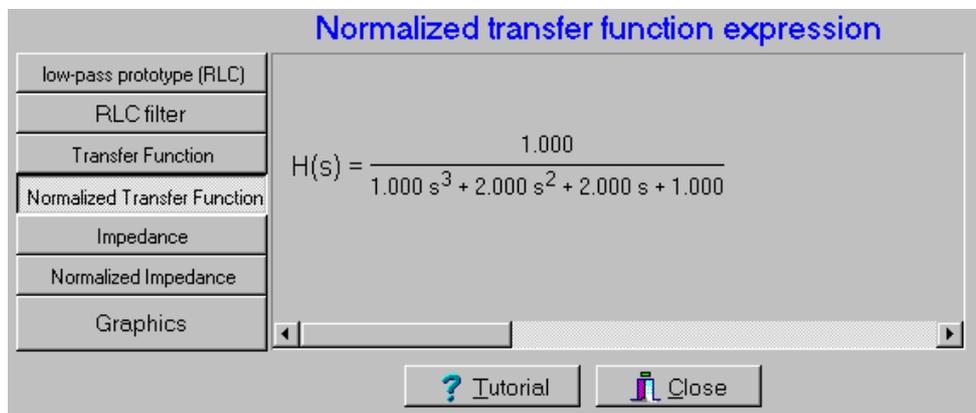
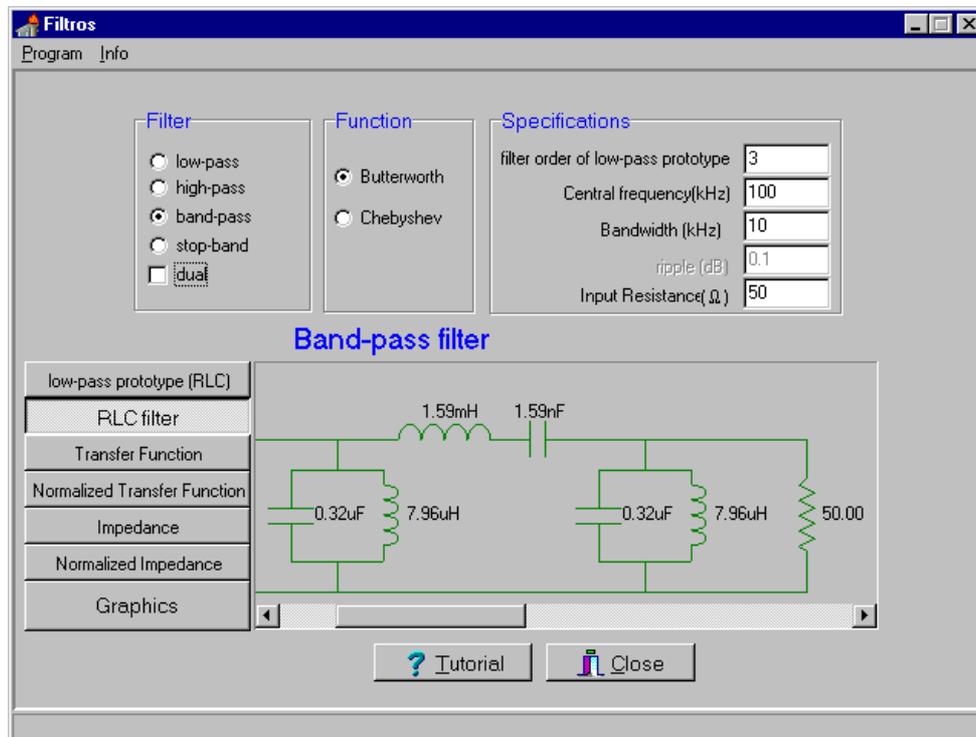


Figura 3-12. Resposta de uma saída típica.

3.2.4 RESULTADOS

O produto final deste trabalho é caracterizado por um ambiente com várias mídias integradas para o ensino e aplicação dos filtros elétricos. A tecnologia de multimídia com sua capacidade de interação oferece ao estudante um espaço de aprendizagem harmônica de conceitos teóricos, softwares simuladores e de análise, experimentos de laboratórios e aplicações práticas. A ferramenta desenvolvida pode ser utilizada na disciplina de Circuitos Elétricos do curso de graduação em Engenharia Elétrica das universidades brasileiras. As aulas expositivas são complementadas com recursos multimídia com alto grau de interatividade, onde o objeto de cada lição e suas aplicações são disponíveis com os comandos dos alunos. O Este binômio mostra-se eficiente, estabelecendo uma nova relação de ensino/aprendizagem que permite o estudante desenvolver sua formação em correspondência com seus interesses, suas necessidades e, o principal, no seu próprio ritmo.

4 STREAMING

O *Streaming* permite a transmissão de áudio e vídeo em tempo real através de uma rede local ou Internet. A tecnologia de *Streaming* fundamenta-se na mistura de técnicas de compressão e armazenamento em memória temporária nas máquinas clientes (*buffering*).

Para que o *Streaming* possa ser executado na máquina local é necessária a presença de um plug-in (ou *player*), programa especial que normalmente trabalha em conjunto com o navegador associado à tecnologia proprietária que estiver sendo utilizada.

O método de *Streaming* é implementado fornecendo-se o conteúdo em formatos específicos (*.aam – Authorware ou *.rm – Real Media) em um servidor. Quando um usuário solicita o acesso a um destes arquivos pelo navegador de Internet, através do formato do arquivo solicitado, pode-se identificar o arquivo correspondente de *Streaming*. Este arquivo é transmitido em um fluxo unidirecional e reproduzido imediatamente no computador local após a máquina solicitante ter armazenado uma quantidade reserva (*buffering*) de dados. A vantagem é que o fluxo depois de reproduzido é descartado, não ocupando espaço no disco local. Não há a necessidade de espera da transferência de todo arquivo antes que o mesmo possa ser reproduzido, como acontece na solicitação de um *download*, o usuário, então, espera menos tempo para obter acesso à informação, caso a sua conexão com o servidor seja satisfatória, pois a principal desvantagem da utilização da tecnologia de *Streaming* é que a reprodução no computador local é bastante dependente da largura de banda e do tráfego da rede pela qual o fluxo de dados esta sendo transmitido.

4.1 FORMAS DE TRANSMISSÃO

A transmissão de multimídia pode ser viabilizada através de três formas de transmissão. Essas formas são implementadas sobre o protocolo IP versão 4 (Ipv4): unicast, broadcast e multicast

4.1.1 TRANSMISSÃO UNICAST

A transmissão unicast é um protocolo ponto-a-ponto entre a máquina cliente e o servidor. Tal como mostrado pela Figura 4-13, a conexão existe entre cada estação de trabalho e o servidor. Para calcular a largura de banda necessária, deve-se adicionar o total solicitado por cada estação. No exemplo da figura abaixo se supormos que existam 5 conexões a 28 Kbps, assim serão o total de $28 \text{ Kbps} \times 5 = 140 \text{ Kbps}$ de largura de banda necessária. A difusão Web na modalidade unicast é o começo do que comumente se chama “programação sobre solicitação”, o qual se baseia na prerrogativa de que cada indivíduo do planeta pode ouvir a qualquer programa de áudio e vídeo remoto.

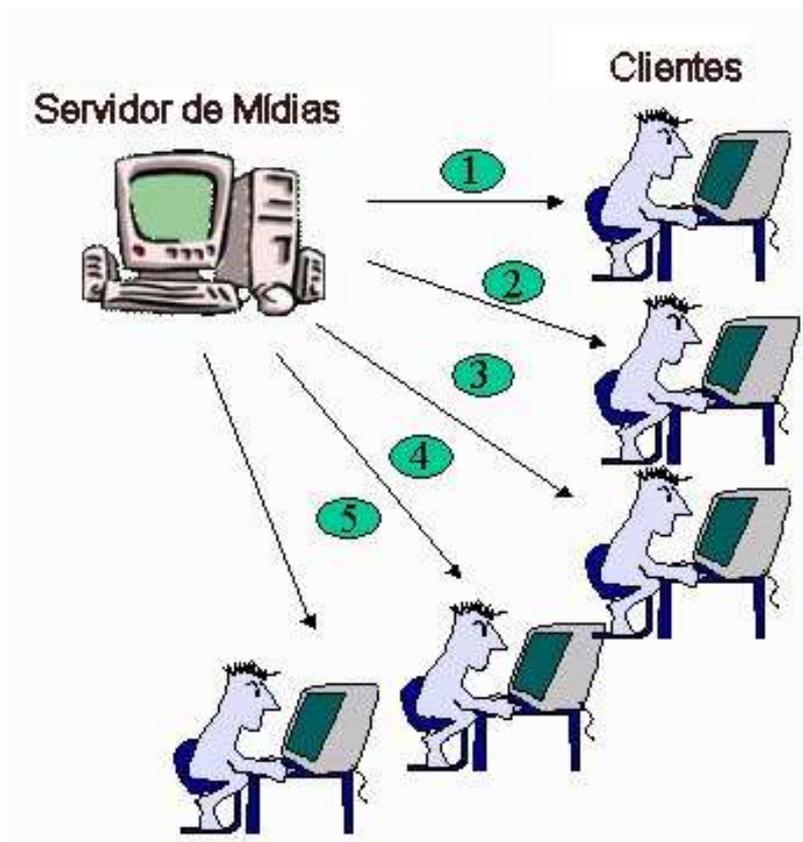


Figura 4-13. Transmissão Unicast.

4.1.2 TRANSMISSÃO BROADCAST- UNICAST

A transmissão broadcast é usualmente concebida como a de um ouvinte de uma estação de rádio, um telespectador de uma estação de televisão ou algum outro evento on line. A gravação do vídeo e do áudio são transmitidas imediatamente no servidor sem ocorrer gravação em disco (Figura 4-14). Os usuários usam uma conexão ponto-a-ponto entre suas máquinas e o servidor, o que implica, mais uma vez, numa conexão entre cada estação e o servidor. Para obter a largura de banda total utilizada, deve-se adicionar o total solicitado por cada conexão.

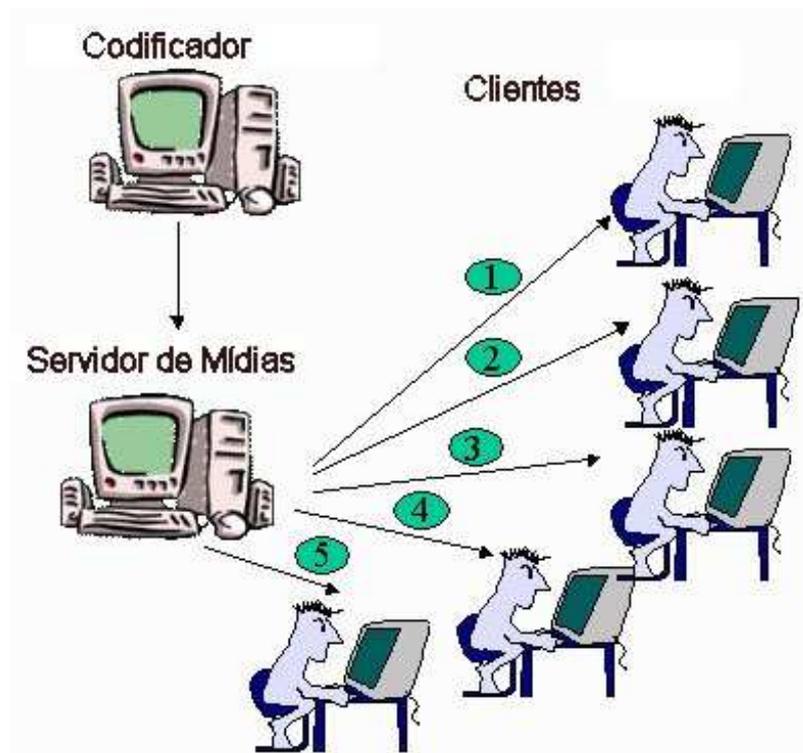


Figura 4-14. Transmissão Broadcast.

4.1.3 TRANSMISSÃO MULTICAST

A transmissão multicast é um processo de roteamento de dados que consiste em envios simultâneos do mesmo pacote de dados para muitos destinos, reduzindo em muito o carregamento da rede, e conseqüentemente, gerando uma economia de banda. Esta técnica torna possível enviar os pacotes sem necessariamente saber os endereços de destino; os dados são difundidos para quem quiser recebê-los (Figura 4-15).

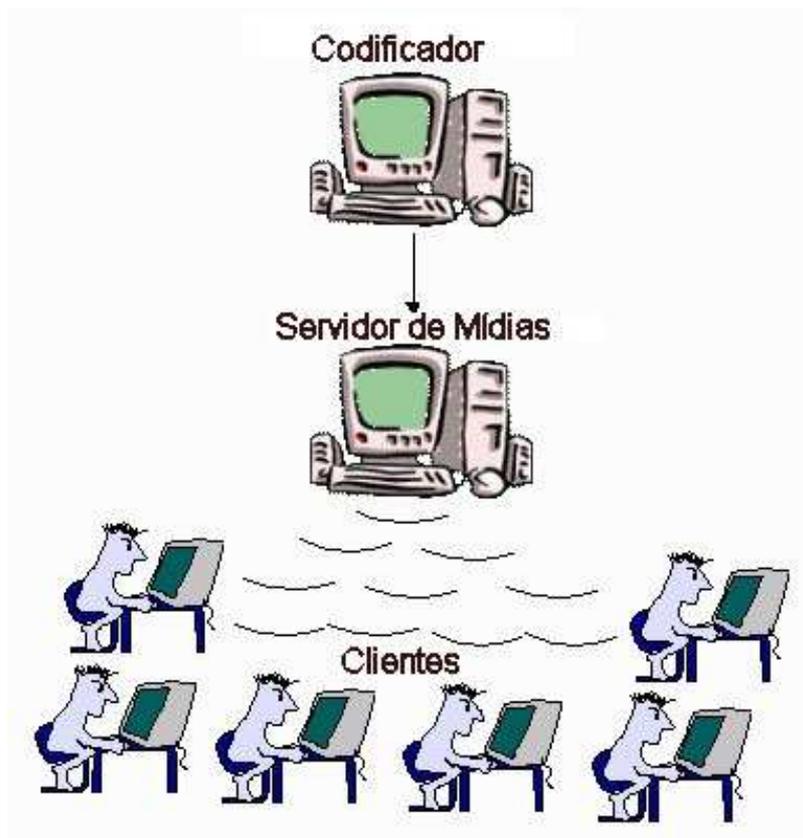


Figura 4-15. Transmissão Multicast.

4.2 VIDEOSTREAMING

O videostreaming é um método de transmissão de áudio/vídeo de forma unidirecional pela Internet. Esta tecnologia pressupõe o uso de um web browser acompanhado de um plug-in adequado, como por exemplo o Windows Media Player ou o Real Player para acesso desse serviço

Existem dois tipos de uso para tecnologia de videostreaming:

- *On-demand*: que permite a visualização de um conteúdo previamente armazenado;
- *Live*: que permite a transmissão de conteúdo ao vivo.

Normalmente as aulas ou reuniões que são realizadas utilizando videoconferência são simultaneamente transmitidas ao vivo (*live*) e gravadas. As apresentações armazenadas servem para pesquisa, consultas e referência. O aluno acessa o servidor de aulas utilizando seu web browser e pode executar o conteúdo de forma *on-demand* ou ainda fazer o seu *download* e salvá-lo em seu próprio

computador. Este material fica disponível o tempo todo, assim como numa biblioteca tradicional.

A informação em videostreaming via Internet é formatada para o usuário em duas janelas dispostas lado a lado (Figura 4-16) numa janela tem-se a apresentação com em áudio e vídeo do instrutor sincronizada com outra janela onde slides do assunto apresentado são mostrados.

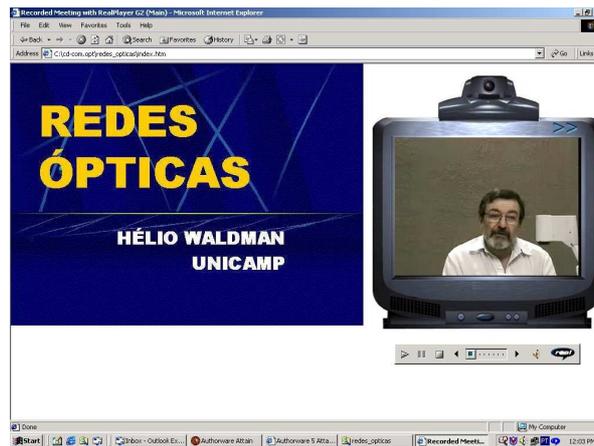


Figura 4-16. Saída Típica de um Videostreaming

4.3 RASTREAMENTO DE DADOS

Considere o caso de um curso no qual os alunos tenham a sua disposição um tutorial eletrônico construídos sobre ferramentas de autoria, e que o conteúdo do tutorial fosse sendo incrementado na medida em que se progride no curso. Se a forma de distribuição escolhida for através de CD-ROM o custo empregado para cada modificação no conteúdo do tutorial, fatalmente acabaria por torná-lo inviável, já que a atualização periódica acarretaria em uma novas gravações de CDs para cada aluno. Através do acesso ao tutorial a partir de *Streaming*, as modificações impostas ao longo do se tornariam economicamente viáveis.

Usar a tecnologia de *Streaming* empregado ao conteúdo de softwares de autoria significa também um número maior de alunos podendo acessá-lo de qualquer lugar e a qualquer hora.

Dessa forma também, cria-se a oportunidade de rastreamento de dados sobre os alunos, seus progressos, e resultados. É possível implementar sistemas para coletar e armazenar dados em um servidor Web e prover acesso a esses dados para os próprios alunos, administradores e gerentes.

A complexidade da solução do rastreamento de dados atinge faixas que vão de apenas capturar a pontuação e testes em um arquivo até controlar um complicado sistema de gerenciamento de conhecimento de uma grande empresa.

Existem várias razões para o rastreamento de dados para um curso on-line. Se o curso requer exigências para o oferecimento de certificados, você necessitará rastrear e monitorar o progresso do aluno e avaliar sua performance. Ou, outro exemplo, a sua organização pode estar interessada em implementar algum relatório de gerência do conhecimento. Alternativamente, se o curso fornecer uma instrução voluntária, sem prazo definido, ou opcional, você pode não ter nenhuma razão para rastrear os dados. Contudo, antes da idéia de não implementar o rastreamento de dados, devemos considerar que a produção do curso provavelmente tenha tomado uma quantidade significativa de esforço, tempo, e dinheiro. Como saber se o que foi investido valeu a pena. O rastreamento de dados pode ajudar a determinar o real retorno de investimento.

Informações típicas que são interessantes descobrir sobre um curso dessa natureza podem responder as seguintes questões:

- O Curso atingiu os objetivos pelo qual foi desenvolvido?
- Os alunos foram beneficiados com o curso?
- O curso está economizando dinheiro da sua organização?
- Quantas pessoas completaram o curso?
- As pessoas estão tendo problemas ao utilizarem o curso?
- O curso é popular?

4.4 MÉTODOS DE RASTREAMENTOS DE DADOS

Entre as maneiras mais comuns de rastreamento de dados estão o rastreamento baseado em arquivo, e-mail e banco de dados

4.4.1 RASTREAMENTO BASEADO EM ARQUIVO

Se a intenção é guardar uma pequena quantidade de dados e precisa-se de um rápido método a um custo inexpressivo de armazenamento de dados, a utilização de arquivos de textos satisfaz esses requerimentos.

Devemos considerar se o arquivo de dados armazenados para cada aluno será no rio disco rígido do local do computador do aluno ou no servidor de páginas. O armazenamento de arquivos nos discos locais não é uma boa opção para cursos do tipo e-learning. Se o aluno resolve acessar o curso de um computador diferente, ele não poderá ler nem atualizar o arquivo com seus dados pessoais. Outra inconveniência é que os arquivos estão armazenados em discos rígidos locais e, por isso, não muito seguros. A menos que eles sejam criptografados ou protegidos por senha, a informação que eles contêm podem ser facilmente acessadas e editadas localmente e até mesmo apagadas.

Arquivos armazenados em servidores Web podem ser protegidos para que não sejam vistos ou “baixados” pela rede. Se a intenção é ser capaz de atualizar o ler um dado de um arquivo de aluno, é necessário implementar algum tipo de procedimento de log-on para garantir que o estudante encontre o seu arquivo no começo de cada sessão. Outra vantagem de arquivos armazenados em servidores Web é que seu conteúdo pode ser disponibilizado como dados de entrada de outra aplicação ou sistemas de banco de dados.

Escrever dados para arquivos textos e a partir deles requer alguma experiência na linguagem da ferramenta de autoria de escolha. Existem várias formas de formatar os dados que serão armazenados em um arquivo de texto. Contudo, dados armazenados e que apresentam formatação com dados separados por vírgula ou delimitados por *tab* têm a vantagem de serem disponíveis para importação em uma planilha eletrônica ou um banco de dados.

A Tabela 4-2 destaca os prós e contras de se utilizar arquivos para o armazenamento de dados.

Tabela 4-2. Prós e Contras Rastreamento Baseado em Arquivo.

Prós	Contras
Para o armazenamento de Dados não requer nenhum software adicional.	Usar arquivos de texto para armazenamento de dados não é bom pelo seu grande volume de dados.
Os dados podem ser importados ou inseridos em outras aplicações.	Este método não prove capacidade de administração sobre os dados.
Utilizar arquivos de texto para a armazenagem de dados é barato.	Rastreamento Baseado em Arquivo não é tão seguro com relação aos outros métodos.

4.4.2 RASTREAMENTO DE DADOS VIA E-MAIL

Se o projeto é simplesmente coletar alguns itens básicos de dados sobre os alunos e suas performances ao longo do curso, uma mensagem de e-mail contendo os dados que são enviados para o gerente de treinamento ou administrador deve satisfazer totalmente os requerimentos de rastreamento de dados. A informação do e-mail pode ser manualmente encaminhada para um sistema administrador de treinamento, ou simplesmente preenchendo-se o registro pessoal do aluno.

Tem-se que considerar o caso em que o e-mail falha em atingir o seu destino. O que pode ser feito é programar o curso para checar por erros no envio da mensagem e tomar a devida ação. Por exemplo, esta ação poderia ser escrever um dado em um arquivo no servidor Web e disparar uma mensagem alertando ao aluno sobre o que aconteceu, isto caso o e-mail atinja seu destino mas apresente falhas em seu conteúdo. Contudo, se o e-mail ao chega a caixa de entrada da administração de treinamento, o aluno não vai poder saber que a mensagem não chegou. Por isto, é imperativo que se tenham métodos de segurança para a coleta e armazenamento de dados para dados que são considerados importantes e não permitidos de serem perdidos.

O método de envio de e-mail varia entre as ferramentas de autoria. A Tabela 4-3 destaca os prós e contras de se utilizar notificação por e-mail.

Tabela 4-3. Prós e Contras o Rastreamento de Dados por e-mail.

Prós	Contras
Configuração simples.	A entrega não pode ser garantida.
Não necessita de um software especial.	Precisa de métodos de backup para dados importantes.
Baixo custo.	E-mails não são eficientes para um volume grande de dados.
O dados podem ser importados ou inseridos em outras aplicações.	Não prove capacidades de administração reais sobre os dados.

4.4.3 RASTREAMENTO DE DADOS ATRAVÉS DE BANCO DE DADOS

Se a intenção é manter grandes quantidades de dados e se houver grande necessidade de geração de relatórios, o rastreamento de dados por banco de dados oferece vantagens significativas sobre a utilização de arquivos de textos ou e-mail. A segurança é característica da maioria dos bancos de dados e utilitários de recuperação de dados.

Depois que um dado é escrito em um banco de dados, este pode seguir para outros cursos e-learning ou lições, e por outros sistemas de informação.

O uso de banco de dados para o armazenamento de dados requer algum conhecimento especializado e experiência em programação. Além disso, a configuração e administração de um banco de dados requerem conhecimento na escolha do software de banco de dados ideal.

Middleware é o software passa os dados por entre o curso e-learning o banco de dados localizado no servidor Web. Exemplos de *middleware* incluem o Microsoft Active Server Pages (ASP), ColdFusion, e Common Gateway Interface (CGI).

É necessário conhecimento para escrever o código que implementa o rastreamento de dados através da solução de *middleware*.

ODBC (Open Database Connectivity) é um padrão amplamente aceito para acesso a banco de dados que usam uma estrutura SQL como linguagem. ODBC trabalha bem em aplicações locais a partir de CD-ROM ou LANs, mas sozinho não é suficiente para transmissão de dados sobre a rede. Pode-se, contudo, usar ODBC em conjunto com *middlewares*, permitindo que sua aplicação em banco de dados trabalhe sobre a rede. Dois exemplos de *middleware* são: Hot Sockets para o

Authorware da empresa Ensodex ou um arquivo Xtra de terceiros chamado Datagrip para o Authorware também. Se a escolha do banco de dados e do *middleware* suportam ODBC, a mudança do banco de dados num futuro é possível. Tem-se isto sem a necessidade de mudar a aplicação.

Os bancos de dados precisam de administração e manutenção de backup freqüente. Para projeto de um curso e-learning, é necessário determinar como o banco de dados terá sua cópia de segurança (backup), e além disso, quantas vezes será atualizado, e por quem.

Se a solução empregada for o uso do banco de dados e *middleware* é necessário obter a licença de utilização para o software específico. Alguns softwares

A Tabela 4-4 listas os prós e contras de se utilizar um banco de dados pra o armazenamento e rastreamento de dados:

Tabela 4-4. Prós e Contras o Rastreamento de Dados através de banco de dados

Prós	Contras
Podem armazenar grandes quantidades de dados	Conhecimento especializado e programação são requeridos
São seguros	A Licença de utilização incrementa o custo.
Utilitários de Recuperação de Dados	Necessita de manutenção e Administração permanentes.
Os dados podem ser por outras aplicações e sistemas de relatórios	

O melhor método para o rastreamento de dados em um determinado curso depende de muitos critérios: escalabilidade, volume de dados, segurança dos dados, experiência em programação e custo.

4.5 TECNOLOGIA DE **STREAMING** EM SOFTWARES DE AUTORIA **MACROMEDIA AUTHORWARE**

A tecnologia de *Streaming* permite “empacotar” em um arquivo *.a5r do Authorware especialmente para a Web, fazer o seu download sobre a Internet ou uma Internet, e executá-lo em um *browser* compatível tais como o Netscape Navigator ou o Internet Explorer.

O *Streaming* envolve o uso de dois componentes:

- Authorware Web Packager, utilizado para o preparo da ferramenta de autoria para a sua segmentação. Authorware Web Packager também cria um arquivo de mapeamento, que diz ao Authorware Web Player o que e quando fazer o download, e onde gravar os segmentos que chegam da rede.
- Authorware Web Player, responsável pela manipulação do download de acordo com o arquivo de mapeamento e executa o programa de autoria.

Uma das grandes vantagens de se empacotar a ferramenta de autoria é que este é dividido em segmentos. O Authorware Web Player pode começar a executar o programa tão logo chegue o primeiro segmento da rede – ele não tem que esperar pelo programa de autoria inteiro. Além disso, o player executa os segmentos apenas quando eles necessitam. Com essa propriedade é possível produzir largas aplicações – 20 MB , 3 MB ou maior – que pode ser executado eficientemente, com uma pequena espera para o início do programa.

Utilizando-se a tecnologia de *Streaming*, pode-se entregar com completa fidelidade ferramentas de autoria interativas, treinamento multimídia em qualquer lugar de uma companhia, rastreando informações dos usuários, oferecendo ajuda on-line, e trocando informação entre usuários e arquivos do servidor.

É possível se criar uma central com documentos interativos para serem usados pelos empregados de uma companhia – tais como lista de empregos, informação de benefícios, telefones de diretorias, e performances – com capacidade de atualização disponível a qualquer momento.

É possível incluir textos, gráficos, áudio, animações, e filmes, tanto quanto rastreamento e gerenciamento de dados a partir de banco de dados. O aplicativo pode obter informações da rede e enviar informação de volta ao servidor. É possível ir a outras URLs e fazer chamadas para JavaScripts a partir do aplicativo de autoria.

4.6 CONSIDERAÇÕES

A principal limitação sobre praticamente tudo o que pode ser feito é a velocidade de transmissão de dados da rede que carrega o aplicativo. Quanto menor a velocidade de transmissão, menos ambicioso deve ser o conjunto multimídia a ser empregado.

A Tabela 4-5 mostra tempos de acesso teóricos sobre diferentes taxas de bits ao acesso a conteúdos distintos. As taxas até 28.8 kbps são comuns em modems, as de 64 e 128 em linhas ADSL, e as de 1,5 Mbps encontradas em conexões padrões de acesso rápido à Internet (T1).

Tabela 4-5. Tempo de acesso em diferentes taxas.

Conteúdo	Tamanho	14.4 Kbps	28.8 Kbps	128 kbps	1.5 Mbps
Pequenos Gráficos e Animações	30K	17 segs	8,3 segs	1,9 segs	0,16 seg
Filmes pequenos completos	200K	110 segs	55 segs	12,5 segs	1 seg
Vídeo clipes médios	500K	900 segs	138 segs	32 segs	2,7 segs
Filmes grandes completos	1 MB	n/a	n/a	62,5 segs	5,3 segs
Stream de vídeo MPEG	N/a	n/a	n/a	n/a	Contínuo

4.7 EXPERIÊNCIA: UTILIZANDO A TECNOLOGIA DE *STREAMING* SOBRE O SOFTWARE DE AUTORIA “ANÁLISE E SÍNTESE DE FILTROS PASSIVOS”

4.7.1 OBJETIVO

A aplicação tem por objetivo mostrar o procedimento de difusão via Web de um software de autoria e analisar o resultado sobre o material multimídia “Síntese e Análise de Filtros Passivos” construído a partir do software de autoria Authorware, uma arquivo que no seu total possui aproximadamente 9 MB, e implementar um sistema de Rastreamento de Dados através de banco de dados que apresente tela de log-on e perfil de cada usuário cadastrado no sistema.

4.7.2 ESTRUTURA DO TUTORIAL VIA WEB

Depois de modificações no programa original, o novo programa de autoria ganhou dois ícones do tipo “Map” representados pela cor azul na Figura 4-17. Um início e o outro em cada saída do programa. Eles serão responsáveis pelo rastreamento de dados . A Figura 4-17 mostra a tela do Authorware com seus dois novos conjuntos de dados. Note que um é inserido no início, logo após a animação de abertura, e o outro no botão de saída do aplicativo.

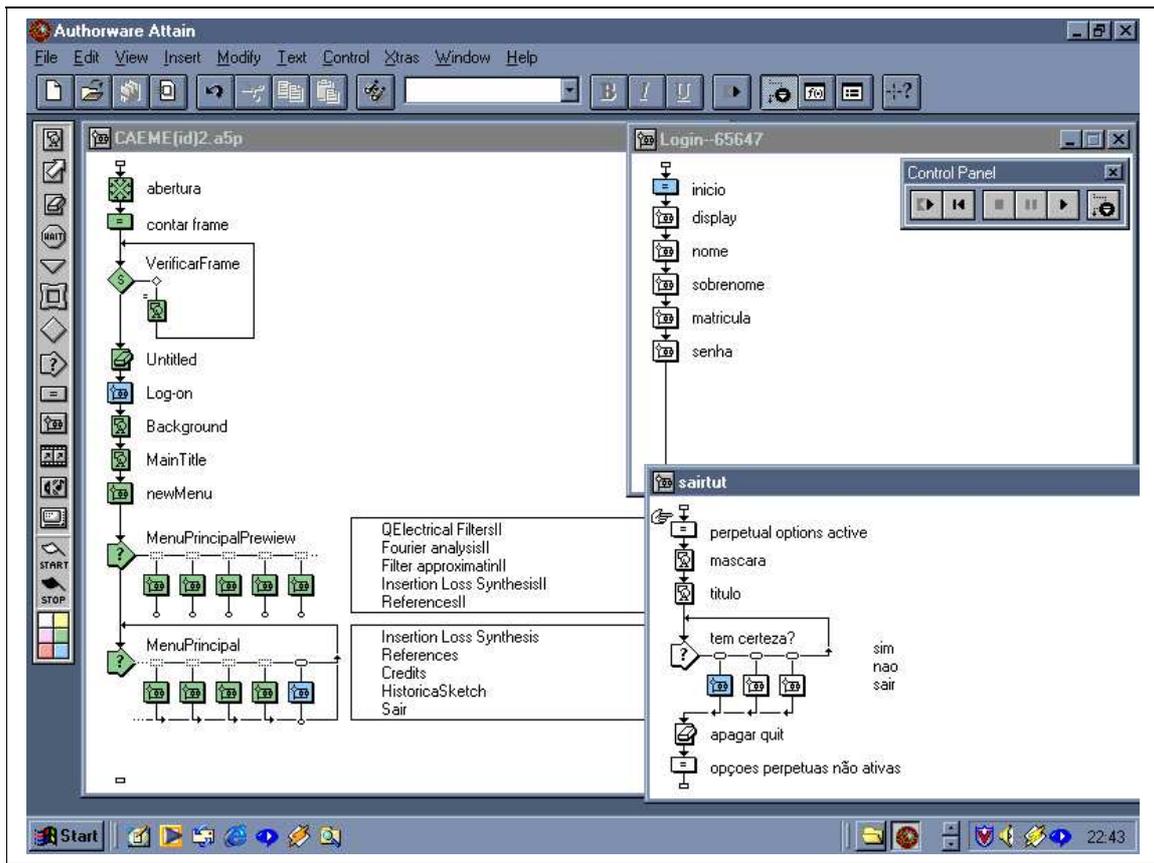


Figura 4-17. Tela com a atualização sofrida pelo programa de autoria.

4.7.3 PREPARAÇÃO DO MATERIAL

O conteúdo do tutorial, que antes era apresentada através do CD-ROM, tem sua entrega através da intranet. O arquivo original (filtros.a5p) é empacotado no formato (filtros.a5r) para poder ser particionado em segmentos de 16 kbytes. (filtros.aam).

Sobre a configuração da rede tem-se:

- A configuração de uma rede ethernet (10 Mbps) possuindo um servidor Web com suporte a tecnologia ASP e quatro máquinas clientes. Figura 4-18
- Instalação do player Authorware 5 nas máquinas clientes com sistema operacional Windows 98 / 2000.

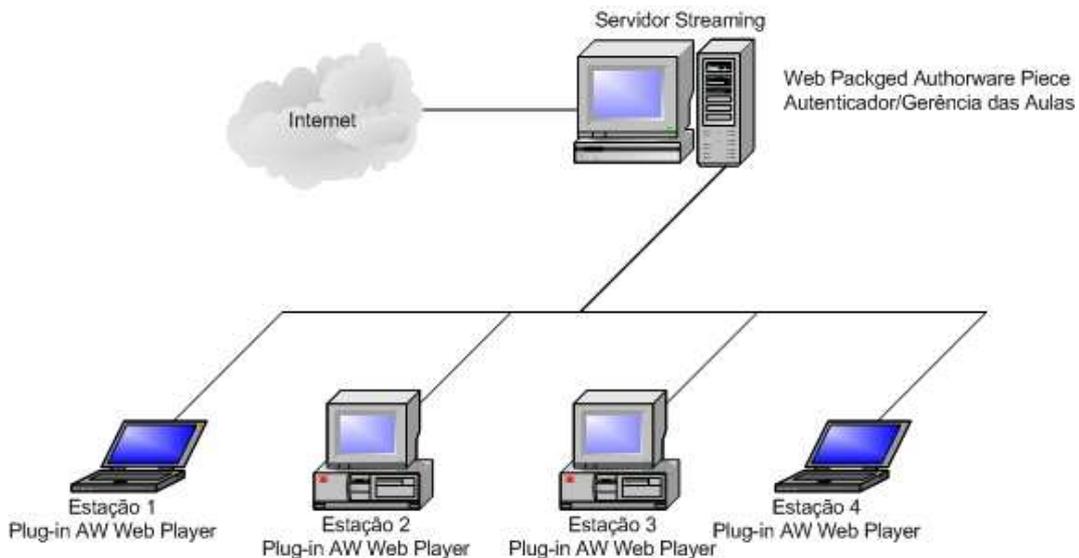


Figura 4-18. Topologia da Intranet

No servidor encontram-se:

- O programa de autoria no formato de *Streaming*; extensão *.aam que controla o tempo e o pacote - arquivos de extensão *.aas - a ser transmitido pela rede.
- Os programas, vídeos, animações, dll's, transições a serem enviados para as máquinas clientes.
- Banco de dados Microsoft Access, para armazenamento e administração dos dados rastreados.
- Página dinâmica conectada com o banco de dados para acesso remoto aos dados.

A Figura 4-19 mostra o diretório no Servidor onde se encontram estes dados. Alguns segmentos, extensão .aas, do CD foram retirados para a impressão da figura. Nota-se a presença de arquivos .swf (animações em Flash), .asp, segmentos .aas, .jpg (figuras), .aam, .htm, .mdb, .u32 (biblioteca Authorware); diretórios: SLIDES, contendo as apresentações em .ppt, Programas, applets e softwares de simulação, lib, mais bibliotecas do Authorware.

Nome	Tamanho	Tipo	Modificado
lib		Pasta de arquivos	13/04/02 13:32
Programas		Pasta de arquivos	14/04/02 00:07
SLIDES		Pasta de arquivos	14/04/02 00:04
Abertura.swf	13KB	Shockwave Flash O...	12/01/02 23:46
admin.asp	6KB	Documento do Activ...	15/04/02 18:22
caem0000.aas	17KB	Arquivo AAS	15/04/02 13:35
caem0001.aas	15KB	Arquivo AAS	15/04/02 13:34
caem0002.aas	11KB	Arquivo AAS	15/04/02 13:34
caem0003.aas	210KB	Arquivo AAS	15/04/02 13:34
caem0004.aas	18KB	Arquivo AAS	15/04/02 13:34
caem0005.aas	121KB	Arquivo AAS	15/04/02 13:34
caem0006.aas	99KB	Arquivo AAS	15/04/02 13:34
caem0007.aas	9KB	Arquivo AAS	15/04/02 13:34
caem0008.aas	388KB	Arquivo AAS	15/04/02 13:34
caem0009.aas	310KB	Arquivo AAS	15/04/02 13:34
caemeodbc.aam	5KB	Arquivo AAM	15/04/02 13:35
default.htm	1KB	Netscape Hypertext ...	13/04/02 15:15
filtroMenu3.swf	7KB	Shockwave Flash O...	07/03/02 16:12
logounb1.jpg	7KB	Mozilla JPEG Image ...	06/04/02 17:30
mmtmp.mdb	212KB	Aplicativo do Micros...	19/04/02 19:34
odbc.aam	1KB	Arquivo AAM	13/04/02 00:05
odbc.u32	30KB	Arquivo U32	15/04/02 13:35
persona.swf	15KB	Shockwave Flash O...	01/04/02 22:38
Thumbs.db	4KB	Arquivo DB	13/04/02 11:01
tmsdsn.u32	28KB	Arquivo U32	15/04/02 13:35

5,70KB Meu computador

Figura 4-19. Diretório no servidor de streaming onde estão os arquivos da experiência

Na máquina cliente:

- Authorware Web Player instalados nos Navegadores que o suportam. Tais como Internet Explore e Netscape. A máquina deve obedecer as condições mínimas do fabricante, que são:
 - PC compatível IBM 486 ou maior.
 - Windows 3.1, 95, 98, Windows NT 3.51 ou NT 4.
 - Monitor VGA de 8-bit, 256-bit ou maior
 - 8 MB de RAM e 6 MB de disco livre
 - Navegador Netscape 4.0 (ou compatível) ou Microsoft Internet Explorer 5.0 ou compatível

- A princípio não há necessidade de armazenamento de dados em disco nas máquinas clientes, mas pode ocorrer reserva temporária para execução de um programa, por exemplo.

Tabela 4-6. Equipamentos, softwares e arquivos.

Máquina	Softwares requeridos	Arquivos em disco
Servidor	-Servidor de páginas com suporte à tecnologia ASP - Microsoft Access. -Padrão Open Databased Connectivity	-Arquivo de controle de streaming (*.aam) -Segmentos do tutorial (*.aas) -Banco de Dados Microsoft Access (*.mdb) -Bibliotecas, transições, vídeos, animações, softwares de simulação e arquivos externos que acompanham o software de autoria (*.dll, *.xtra, *.exe, *.mpeg, etc.)
Cliente	-Navegador de Internet que suportam o Authorware Web Player (ex: Internet Explorer) -Authorware Web Player	-Previamente não é necessário nenhum arquivo armazenado em disco

O arquivo empacotado possui em sua estrutura linhas de comandos SQL que através do padrão ODBC se conecta com o banco de dados Access. A Figura 4-20 mostra o código fonte para o registro de novos usuários no banco de dados do sistema localizado no servido Web (para melhor visualização veja o Anexo I):

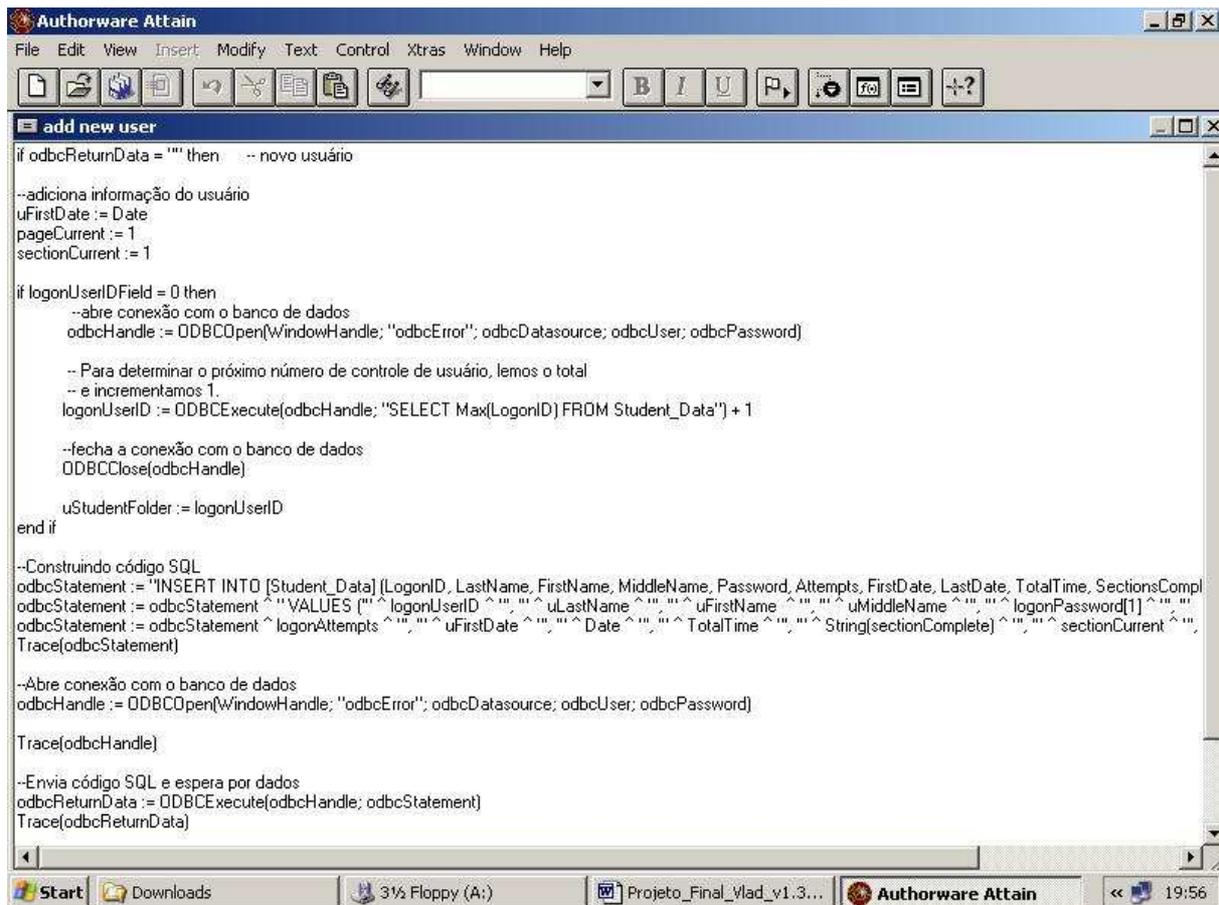


Figura 4-20. Comandos SQL sendo executados pelo Authorware.

4.7.4 FUNCIONAMENTO DA APLICAÇÃO

No primeiro acesso, a pessoa é convidada a se cadastra no sistema. Nome, sobrenome, matrícula e senha são exigidos (Figura 4-21). Se o cadastro é efetuado com sucesso, o aluno tem seus dados de log-on inseridos no banco de dados do sistema, e segue com o conteúdo do tutorial multimídia Análise e Síntese de Filtros Passivos.

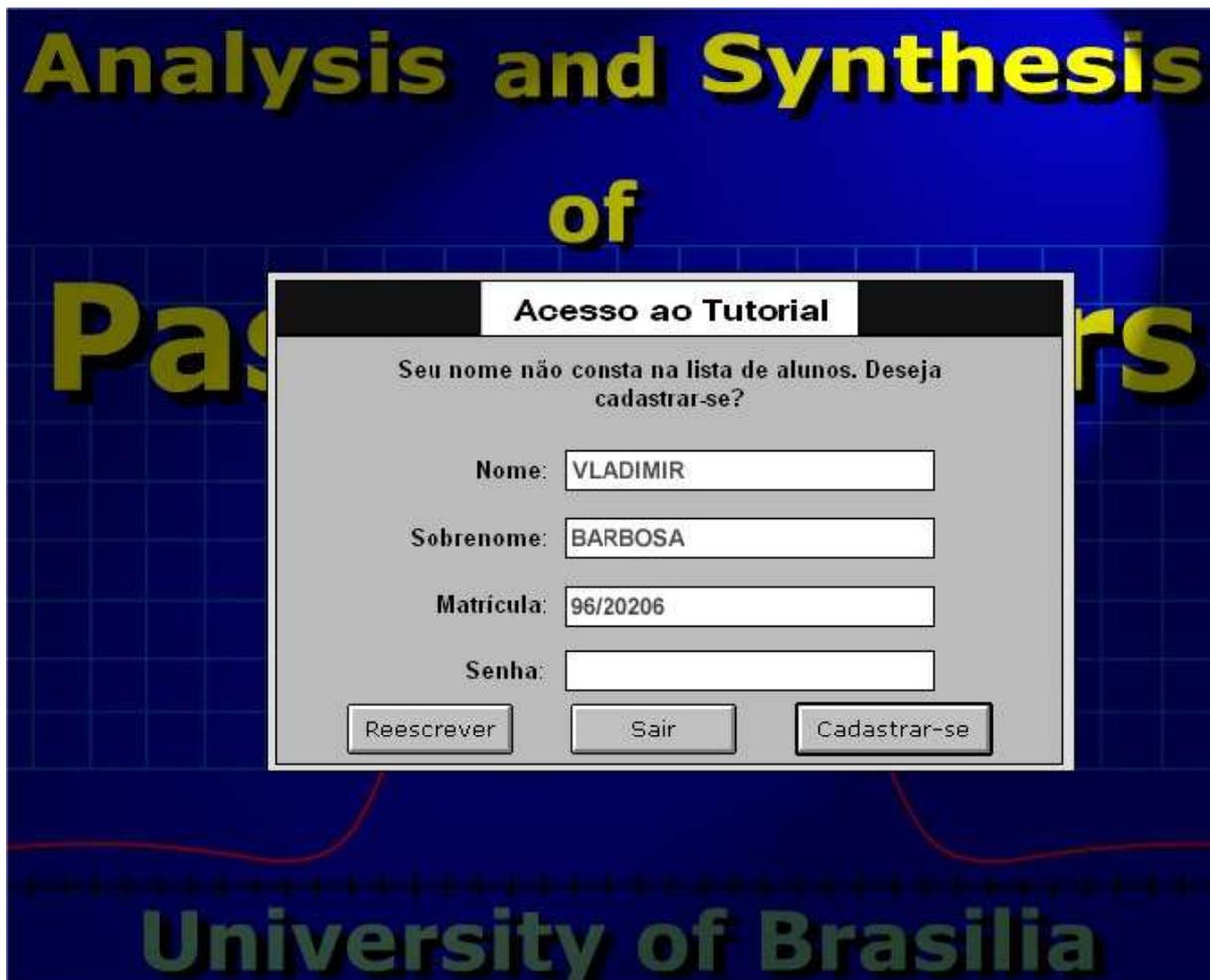


Figura 4-21. Tela de Log-on para acesso ao tutorial.

Os dados rastreados neste experimento foram:

- Primeiro acesso ao tutorial
- Último acesso
- Número de entradas no tutorial
- Tempo total de utilização do tutorial

Na saída do programa dados de rastreamento são atualizados e lançados no arquivo *.mdb localizado no servidor (Figura 4-22),

LogonID	Sobrenome	Nome	senha	N. acessos	Primeiro	Último	TempoTotal
123	NEY	IVAN	123	9	14/04/02	19/04/02	00:00:00
1234	JORGE	JORGE	1234	1	19/04/02	19/04/02	00:01:00
456	PASTOR	EDUARDO	456	1	14/04/02	14/04/02	01:29:00
93321	LIMA	FLAVIO	amor1	2	15/04/02	15/04/02	00:02:00
9619836	AMARAL	ADRIANO	9619836a	2	15/04/02	15/04/02	00:04:00
9620206	DAIGELE	VLADIMIR	atc	12	14/04/02	15/04/02	00:10:00
9707905	POVOA	LILIAN	ayla	3	15/04/02	15/04/02	00:00:00
9822909	IRYODA	FLAVIO	iryoda	1	14/04/02	14/04/02	01:17:00
9902546	EVANGELISTA	CARLOS	4321	2	19/04/02	19/04/02	00:01:00

Figura 4-22. Banco de dados do sistema.

O monitoramento de dados é passo importante para a coordenação do curso saber como a utilização do tutorial vem sendo aproveitada pelos alunos, se o acesso e o tempo despendido têm sido suficientes.

A página de gerência foi criada com o intuito da coordenação do curso ter em mãos um acesso rápido e remoto aos dados rastreados dos alunos. Uma visualização da página pode ser vista pela Figura 4-23.

Universidade de Brasília
Departamento de Engenharia Elétrica
2º Semestre de 2001

Analysis and Synthesis of Passive Filters
University of Brasília

Análise e Síntese de Filtros Passivos
Administração e Gerência

Matrícula:	Nome do Aluno:	Senha:	Data de Entrada	Última Data de Acesso	Tempo Total	Número de Acessos
9620206	VLADIMIR DAIGELE	atc	14/04/02	15/04/02	00:10:00	12
9822909	FLAVIO IRYODA	iryoda	14/04/02	14/04/02	01:17:00	1
123	IVAN NEY	123	14/04/02	19/04/02	00:00:00	9
456	EDUARDO PASTOR	456	14/04/02	14/04/02	01:29:00	1

Figura 4-23. Página dinâmica de administração.

O código fonte em HTML com as funções ASP é mostrado pelo Anexo II
Os dados são organizados, ordenados e ilustrados da maneira que se desejar.

4.7.5 RESULTADOS

Quanto à distribuição do material multimídia ficou marcante o comportamento das applets sobre os outros programas, por terem suas estruturas desenvolvidas para Web, comportaram-se de forma mais eficiente, não é preciso gravá-las no disco dos usuários e tão pouco mostram problemas de falta de dll's.

Levando-se em conta o fluxo de informação do resto do tutorial através da intranet, o desempenho obtido foi satisfatório.

5 CONCLUSÃO

A integração das diversas tecnologias modernas permite o aumento do nível da velocidade de assimilação, transformando os processos mais efetivos e o aprendizado mais divertido. No departamento de Engenharia Elétrica, algumas das várias ferramentas já construídas sobre o tema telecomunicações em formato de slides, softwares de simulação, videostreaming, etc., como no caso de comunicações óticas e filtros passivos descritos por este trabalho, encontram-se reunidos sobre forma de tutorias eletrônico bastante ricos em conteúdo multimídia.

A forma como o tutorial multimídia é disponibilizado para as diversas turmas de alunos que vierem a passar pelas matérias que as possuem pode variar. Se o plano de ensino não apresenta grade variação em sua estrutura no decorrer do curso, o CD-ROM pode ser a solução ideal; o preço unitário de um CD virgem fica em conta para o estudante. Como apresentado no trabalho, se a emenda do curso for propenso a sofrer mudanças até sua conclusão, a distribuição pela intranet do próprio departamento, com os alunos podendo acessar laboratórios de informática, é outra saída apresentada que possui diversas formas de controle e formatos sobre o acesso ao tutorial multimídia.

A questão fica sobre o desenvolvimento, montagem e manutenção dos tutoriais multimídias. À cerca do desenvolvimento do conteúdo integrador do tutorial, os próprios professores – estruturando suas aulas em formato digital, mas claro, sem precisarem esquecer do velho e bom método giz e quadro negro – e dos alunos desenvolvendo softwares de simulação e animação através de atividades complementares da sala de aula ou como projetos de pesquisa. Quanto à montagem e manutenção dos tutoriais, evidentemente estas ações podem representar custos para a instituição de ensino, visto que são necessárias algumas habilidades de manipulação dos programas de autoria, se ela procurar meios externos. Se a instituição mantiver um grupo ativo e reciclado, podendo ser formado por alunos do próprio departamento e/ou de outros, com transferência permanente das técnicas operacionais e funcionamento das diversas ferramentas envolvidas no processo de

concretização de todo o material, os custos seriam levados a uma redução significativa.

A abordagem deste trabalho representa apenas uma possibilidade de complementar o ensino tradicional através de fontes de informações modernas e acessíveis que merece destaque para discussões.

6 BIBLIOGRAFIA

- [1] Cantú, Marco. *Mastering Delphi 5*, São Paulo: MAKRON Books, 200
- [2] Horstman, Cay; Cornell, Gane. *Core Java – Volume 1*, São Paulo: MAKRON Books, 2001
- [3] [http:// www.macromedia.com](http://www.macromedia.com), abril, 2002
- [4] [http:// www.macromedia.com/support/authorware](http://www.macromedia.com/support/authorware), abril, 2002
- [5] <http://www.aspforums.com>
- [6] <http://forumasp.cjb.com.br>
- [7] <http://www.microsoft.com.br>
- [8] E-learning Guide. Marcromedia, janeiro 2002
- [9] <http://ltsc.ieee.org/index.htm>
- [10] <http://www.sun.com/java> John Wiley & Sons, Inc, Computer Applications in Engineering Education.
- [11] Bamberger, Roberto H..*Interactive Tools for Signal Processing Education: The Signal Processing Instructional Facility (SPIF Lab) at Washing State University*. John Wiley & Sons, Inc, Computer Applications in Engineering Education. Vol I(6) ,1993
- [12] Universidade de Brasília, Departamento de Engenharia Elétrica, *CD-ROM – Comunicações Ópticas*, Abril 2001.
- [13] Deus, Flavio E.Gomes. *Apropriação de Ferramentas Tecnológicas em Aplicações no Ensino*. Universidade de Brasília, Março / 2002

- [14] Molinaro, L. F.; Abdalla Humberto Jr; Carvalho, Paulo de; Deus, Flávio E.; Barbosa, Vladimir D. S.; Sobral, Rogério: *Mestrado Profissionalizante em Telecomunicações suportado por Mídias Integradas*. Conferencia Internacional sobre Educación, Formación y Nuevas Tecnologías, Virtual EDUCA Madrid, Espana. Junio, 2001
- [15] Assunção, Eudes Pimentel de; Cardoso, Renato Barreto; Afonso, Guilherme; Barbosa, Vladimir D. S.; Abdalla, Humberto Junior. *Interactive Software for Analysis and Synthesis of Passive Filters and its Educacional Applications*. John Wiley & Sons, Inc, Computer Applications in Engineering Education. Dezembro / 2001
- [16] Abdalla, Humberto Jr, Antonio J. M. Soares, Paulo O. Guimarães, Flávio E. G. de Deus, L. F. Molinaro, Vladimir D. S. Barbosa, Rogério T. do Prado Sobral. *Integração de Novas Tecnologias no Processo Educacional Tradicional*. XIX Simpósio Brasileiro de Telecomunicações, Fortaleza – CE, Setembro, 2001.

7 ANEXO A

Código fonte do comando baseado em estrutura SQL efetuado a partir do Authorware para o registro de novos usuários no banco de dados do sistema localizado no servido Web:

```
if odbcReturnData = "" then      -- novo usuário

--adiciona informação do usuário
uFirstDate := Date
pageCurrent := 1
sectionCurrent := 1

if logonUserIDField = 0 then
    --abre conexão com o banco de dados
    odbcHandle := ODBCOpen(WindowHandle; "odbcError";
odbcDatasource; odbcUser; odbcPassword)

    -- Para determinar o próximo número de controle de usuário,
    lemos o total
    -- e incrementamos 1.
    logonUserID := ODBCExecute(odbcHandle; "SELECT Max(LogonID) FROM
Student_Data") + 1

    --fecha a conexão com o banco de dados
    ODBCclose(odbcHandle)

    uStudentFolder := logonUserID
end if

--Construindo código SQL
odbcStatement := "INSERT INTO [Student_Data] (LogonID, Sobrenome, Nome,
Senha, N_acessos, Primeiro, Último, TempoTotal)"
odbcStatement := odbcStatement ^ " VALUES ('" ^ logonUserID ^ "', '" ^
uLastName ^ "', '" ^ uFirstName ^ "', '" ^ uMiddleName ^ "', '" ^
logonPassword[1] ^ "', '"
odbcStatement := odbcStatement ^ logonAttempts ^ "', '" ^ uFirstDate ^ "',
'" ^ Date ^ "', '" ^ TotalTime ^ "', '" ^ String(sectionComplete) ^ "', '"
^ sectionCurrent ^ "', '" ^ pageCurrent ^ "'"
Trace(odbcStatement)

--Abre conexão com o banco de dados
odbcHandle := ODBCOpen(WindowHandle; "odbcError"; odbcDatasource; odbcUser;
odbcPassword)

Trace(odbcHandle)

--Envia código SQL e espera por dados
odbcReturnData := ODBCExecute(odbcHandle; odbcStatement)
Trace(odbcReturnData)
```

```
--Fecha conexão com o banco  
ODBCclose(odbcHandle)  
  
end if
```

8 ANEXO B

Código fonte em HTML da página dinâmica de administração:

```
<%@LANGUAGE="VBSCRIPT"%>
<!--#include file="Connections/author.asp" -->
<%
set Recordset1 = Server.CreateObject("ADODB.Recordset")
Recordset1.ActiveConnection = MM_author_STRING
Recordset1.Source = "SELECT * FROM Student_Data"
Recordset1.CursorType = 0
Recordset1.CursorLocation = 2
Recordset1.LockType = 3
Recordset1.Open()
Recordset1_numRows = 0
%>
<%
Dim Repeat1__numRows
Repeat1__numRows = -1
Dim Repeat1__index
Repeat1__index = 0
Recordset1_numRows = Recordset1_numRows + Repeat1__numRows
%>
<html>
<head>
<title>.::An&aaacute;lise e S&iacute;ntese de Filros Passivos ::
Administra&ccedil;&atilde;o do Curso</title>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1">
</head>
<body bgcolor="#FFFFFF" text="#000000">
<table width="800" border="0" height="63" cellpadding="0" cellspacing="0">
<tr>
<td width="94" height="87">
<div align="center"><b><font face="Verdana, Arial, Helvetica, sans-serif"></font></b></div>
</td>
<td width="345" valign="middle" height="87">
<p align="left"><b><font face="Verdana, Arial, Helvetica, sans-serif"
size="2">Universidade
de Bras&iacute;lia<br>
Departamento de Engenharia El&eacute;trica<br>
<font size="1">2&deg; Semetre de 2001</font></font></b></p>
</td>
<td width="150" valign="middle" height="87" align="center"><object
classid="clsid:D27CDB6E-AE6D-11cf-96B8-444553540000"
codebase="http://download.macromedia.com/pub/shockwave/cabs/flash/swflash.c
ab#version=5,0,0,0" width="120" height="100">
<param name=movie value="Abertura.swf">
<param name=quality value=high>
<embed src="Abertura.swf" quality=high
pluginspage="http://www.macromedia.com/shockwave/download/index.cgi?>

```

```

Pl_Prod_Version=ShockwaveFlash" type="application/x-shockwave-flash"
width="120" height="100">
  </embed>
  </object></td>
</tr>
</table>
<p align="center"><font face="Verdana, Arial, Helvetica, sans-
serif"><b><font color="#006600">An&acute;lise
e Sintese de Filtros Passivos</b></font></p>
<p align="center"><b><font face="Verdana, Arial, Helvetica, sans-serif"
color="#0000FF">Administra&ccedil;&atilde;o
e Ger&ecirc;ncia</b> </p>
<%
While ((Repeat1__numRows <> 0) AND (NOT Recordset1.EOF))
%>

<table width="700" border="0" height="65" cellpadding="0" cellspacing="0">
  <tr>
    <td align="left" valign="top" height="56" width="221">
      <table border="0" cellpadding="0" cellspacing="0" width="220"
height="60">
        <tr>
          <td width="232" height="20"><b><font face="Verdana, Arial,
Helvetica, sans-serif" size="1">Matr&iacute;cula:
          <font color="#0099FF"><
%=(Recordset1.Fields.Item("LogonID").Value)%></font></font></b></td>
        </tr>
        <tr>
          <td height="20" width="232"><b><font face="Verdana, Arial,
Helvetica, sans-serif" size="1">Nome
          do Aluno: <font color="#0099FF"><
%=(Recordset1.Fields.Item("FirstName").Value)%> &nbsp;&nbsp;&nbsp;<
%=(Recordset1.Fields.Item("LastName").Value)%></font> </font></b></td>
        </tr>
        <tr>
          <td height="20" width="232"><b><font face="Verdana, Arial,
Helvetica, sans-serif" size="1">Senha:
          <font color="#0099FF"><
%=(Recordset1.Fields.Item("Password").Value)%></font></font></b></td>
        </tr>
      </table>
      <font face="Verdana, Arial, Helvetica, sans-serif" size="1"></font>
      <p>&nbsp;&nbsp;&nbsp;</p>
    </td>
    <td align="left" valign="top" height="56" width="520">
      <table width="500" border="1" cellpadding="0" cellspacing="0"
bordercolor="#CCCCCC">
        <tr>
          <td height="25" width="50">
            <div align="center"><b><font face="Verdana, Arial, Helvetica,
sans-serif" size="1">Data
            de Entrada</b></div>
          </td>
          <td height="18" width="50">
            <div align="center"><b><font face="Verdana, Arial, Helvetica,
sans-serif" size="1">&Uacute;ltima
            Data de Acesso</b></div>
          </td>
          <td height="18" width="50">
            <div align="center"><b><font face="Verdana, Arial, Helvetica,
sans-serif" size="1">Tempo

```

```

        Total</font></b></div>
    </td>
    <td height="18" width="50">
        <div align="center"><font face="Verdana, Arial, Helvetica,
sans-serif" size="1"><b>N&uacute;mero
de Acessos </b></font> </div>
    </td>
</tr>
<tr>
    <td height="25">
        <div align="center"><font face="Verdana, Arial, Helvetica,
sans-serif" color="#0099FF"><font size="1"><font size="1"><b><b><b><b><
%=(Recordset1.Fields.Item("FirstDate").Value)
%></b></b></b></b></font></font></font></div>
    </td>
    <td height="17">
        <div align="center"><font face="Verdana, Arial, Helvetica,
sans-serif" color="#0099FF"><font size="1"><font size="1"><b><b><b><b><
%=(Recordset1.Fields.Item("LastDate").Value)
%></b></b></b></b></font></font></font></div>
    </td>
    <td height="17">
        <div align="center"><font face="Verdana, Arial, Helvetica,
sans-serif" color="#0099FF"><font size="1"><font size="1"><b><b><b><b><
%=(Recordset1.Fields.Item("TotalTime").Value)
%></b></b></b></b></font></font></font></div>
    </td>
    <td height="17">
        <div align="center"><font face="Verdana, Arial, Helvetica,
sans-serif" size="1" color="#0099FF"><b><b><b><b><
%=(Recordset1.Fields.Item("Attempts").Value)%></b></b></b></b></font></div>
    </td>
</tr>
</table>
</td>
</tr>
</table>
<%
Repeat1__index=Repeat1__index+1
Repeat1__numRows=Repeat1__numRows-1
Recordset1.MoveNext()
Wend
%>
</body>
</html>
<%
Recordset1.Close()
%>

```