UNIVERSIDADE DE SÃO PAULO
INSTITUTO DE FÍSICA DE SÃO CARLOS

Software para Construção de Tabelas Interativas para Educação à Distância

SÉRGIO LUIS ANTONELLO

Dissertação apresentada ao Instituto de Física de São Carlos, da Universidade de São Paulo, para a obtenção do título de Mestre em Física – Área de Física Computacional.

Orientador: Prof. Dr. Dietrich Schiel

SÃO CARLOS
2003

USP/IFSC/SBI

8-2-001487

Antonello, Sérgio Luis

Software para construção de tabelas interativas para Educação à distância / Sérgio Luis Antonello.-- São Carlos, 2003.

128 p.

Dissertação (Mestrado) -- Instituto de Física de São Carlos, Universidade de São Paulo, 2003.

Orientador: Prof. Dr. Dietrich Schiel

1. Tabelas interativas. 2. Educação à distância. 3. EAD.
4. Experimentação à distância.
I. Título.

UNIVERSIDADE
DE SÃO PAULO
Instituto de Física de São Carlos

Caixa Postal 369
13560-970 São Carlos, SP
Av. Trabalhador São-carlense, 400,
13566-590 São Carlos, SP

Fone/Fax 16 273 9777
www.if.sc.usp.br
wladerez@if.sc.usp.br

MEMBROS DA COMISSÃO JULGADORA DA DISSERTAÇÃO DE MESTRADO DE SERGIO LUIS ANTONELLO, APRESENTADA AO INSTITUTO DE FÍSICA DE SÃO CARLOS, DA UNIVERSIDADE DE SÃO PAULO, EM 19/02/2003.

COMISSÃO JULGADORA:

Prof. Dr. Dietrich Schiel (Orientador) – IFSC/USP

Prof. Dr. Dilvan de Abreu Moreira – ICMC/USP

Prof. Dr. Gonzalo Travieso – IFSC/USP

Dedico este trabalho a toda a minha família,
em especial a minha esposa Raquel e ao meu filho Vinícius
pelo respeito e compreensão demonstrados neste período.

Agradecimentos

A Deus, pela força e coragem que me foi concedida para enfrentar e vencer mais uma etapa da minha vida.

À esposa Raquel e filho Vinícius, pelo amor e carinho que recebo e pela compreensão dos momentos distantes.

Aos meus pais, Augusta e Sérgio, por acreditarem em mim e a meus irmãos por estarem sempre comigo.

Ao Professor Schiel, meu orientador, pela extrema confiança em mim depositada, pela compreensão e paciência com a minha situação particular.

Ao Professor Norberto Morales, supervisor do CEAPLA, pela forma que me deixou conduzir este trabalho.

À Iria pela orientação e ajuda no decorrer do trabalho.

À Mônica, minha maior incentivadora, pelo apoio e colaboração nas horas mais difíceis.

Ao Godoy e Peter Krauss pelo apoio técnico sempre que precisei.

À Regina, Alice e Joãozinho pela colaboração especial.

Aos Professores Gilberto Garcia, Ailton Luchiari e Maria Juraci pelo valor creditado em mim.

Aos demais professores do IFSC e ICMC, que colaboraram diretamente para minha formação acadêmica no mestrado.

Aos casais amigos do ECC, pelas palavras de incentivo.

Aos amigos do Ceapla, em especial Juliene e Cláudio pela colaboração.

Aos demais funcionários do CDCC pelo excelente atendimento em todas as minhas necessidades, em especial Edna e Miriam.

Aos funcionários da Pós-Graduação e da Biblioteca pelo ótimo atendimento.

Aos demais familiares e amigos.

SUMÁRIO

LISTA DE FIGURAS vii

LISTA DE TABELAS ix

RESUMO x

ABSTRACT x

1. INTRODUÇÃO 1
   1.1. Considerações 2
   1.2. Objetivos 6

2. EDUCAÇÃO À DISTÂNCIA (EAD) 7
   2.1. Caracterização da EAD 9
   2.2. História da Educação À Distância 12
   2.3. A Estrutura de um Sistema EAD 14
   2.4. Fases de um Sistema EAD 15
      2.4.1. Planejamento, Projeto, Desenvolvimento, Implementação e Avaliação 18

3. A INTERNET 20
   3.1. A World-Wide Web (Web) 21
   3.2. Arquitetura Web 23
      3.2.1. O Módulo Cliente 24
      3.2.2. O Módulo Servidor 26
      3.2.3. Padrões para Comunicação 27

4. A INTERATIVIDADE 31
   4.1. Comunicação e Interatividade 31
   4.2. Interatividade na EAD 32
   4.3. Considerações Finais 35

5. EDUCAÇÃO À DISTÂNCIA NA INTERNET 36

6. INTEGRANDO WEB COM BANCO DE DADOS 41
   6.1. Arquitetura da Aplicação de Banco de Dados 41
   6.2. Scripts Web e a Interface CGI 42
   6.3. HTML, HTTP e os Programas CGI 45
   6.4. Programas Executáveis CGI 45

7. OS RECURSOS UTILIZADOS 47
   7.1. Scripts 47
      7.1.1. Client-Side Scripts 47
      7.1.2. Server-Side Scripts 47
   7.2. A Linguagem de Programação PHP 48
   7.3. Banco de Dados 49
      7.3.1. MySQL 50
   7.4. Considerações Finais 52

8. PROGRAMA EDUC@AR 53
8.1. Projetos do Programa Educ@r 54
8.2. Histórico 55
8.3. O Módulo de Mecânica Gráfica 56
8.3.1. As Tabelas do Mecânica Gráfica 59

9. DESCRIÇÃO DO PROBLEMA 61

10. TABELAS INTERATIVAS PROGRAMÁVEIS 63
10.1. A Interatividade das Tabelas Programáveis 64
10.2. O Módulo Mentor 65
10.3. O Módulo Aluno 68
10.4. A Estrutura do TABIPRO 70
10.5. Etapas para Criar um Protótipo de Tabela 71
10.6. Como Montar um Protótipo de Tabela – Um Exemplo Detalhado 72
10.6.1.O Exemplo Desloc 74
10.7. Formatação de Células, Operadores e Funções Matemáticas 83
10.7.1.Formatação de Células 83
10.7.2.Operadores 85
10.7.3.Funções Matemáticas e Trigonométricas 88
10.8. Utilização de uma Tabela no Módulo Aluno 94
10.8.1.Um Aluno Utilizando a Tabela Desloc 98

11. APLICAÇÃO DAS TABELAS INTERATIVAS PROGRAMÁVEIS NO MÓDULO DE MECÂNICA GRÁFICA 101

12. O AMBIENTE OPERACIONAL DO TABIPRO 103
12.1. A Estrutura dos Protótipos 104
12.1.1.Tabela header 105
12.1.2.Tabela _col 106
12.1.3.Tabela _lin 107
12.2. Considerações Finais 108

13. CONCLUSÕES 109
13.1. Considerações Iniciais 109
13.2. Contribuições 109
13.3. Sugestões para Trabalhos Futuros 111

ANEXO 112

REFERÊNCIAS BIBLIOGRÁFICAS 122

Lista de Figuras

Figura 1 – Processo de desenvolvimento instrucional 17
Figura 2 – Fases de um sistema de educação à distância 18
Figura 3 – As partes do modelo da Web 22
Figura 4 – O modelo cliente/servidor 23
Figura 5 – Configuração Cliente-Servidor da WWW 24
Figura 6 – Representação de links entre documentos Web 25
Figura 7 – Aplicação Web com uso de banco de dados 41
Figura 8 – A interface CGI 43
Figura 9 – Arquitetura de Programas Executáveis CGI 46
Figura 10 – Puck de Hessel 56
Figura 11 – Os módulos do TABIPRO 64
Figura 12 – Fluxo de comunicação no Módulo Aluno do TABIPRO 65
Figura 13 – Fluxo de tarefas do módulo Mentor 67
Figura 14 – Fluxo de tarefas do módulo Aluno 69
Figura 15 – Módulo Mentor, tela inicial para a criação de um protótipo de tabela 73
Figura 16 – Tela inicial para criação de um protótipo de tabela, com parâmetros preenchidos 77
Figura 17 – Tela com as células que comporão a tabela 78
Figura 18 – Conteúdo das células que comporão a tabela 81
Figura 19 – Destaque das células que comporão a tabela 81
Figura 20 – Execução da tabela Desloc no módulo Aluno 82
Figura 21 – Exemplo de uma página Web que aciona o módulo Aluno do TABIPRO 96
Figura 22 – Código HTML de uma página exemplo que aciona o módulo Aluno do TABIPRO 96
Figura 23 – Código HTML para criar campos de entrada de dados 97
Figura 24 – Módulo Aluno monta a tabela que deve ser preenchida 98
Figura 25 – Módulo Aluno recebe digitação de dados 99
Figura 26 – Módulo Aluno destaca a ocorrência de erro nos dados digitados 100

Figura 27 – Módulo Aluno exibe mensagem de parabéns se não ocorrer erro nos dados digitados 100
Figura 28 – O ambiente operacional do TABIPRO 104

Lista de Tabelas

Tabela 1 – Relação de operadores aritméticos 85
Tabela 2 – Operadores de string 85
Tabela 3 – Relação de operadores lógicos 86
Tabela 4 – Relação de operadores de comparação 86
Tabela 5 - Precedência de operadores 87
Tabela 6 - Estrutura da tabela header no MySQL 106
Tabela 7 - Estrutura da tabela _col no MySQL 106
Tabela 8 - Estrutura da tabela _lin no MySQL 107

Resumo

O objetivo deste trabalho é o desenvolvimento de um software, o Tabelas Interativas Programáveis (TABIPRO), que permita a professores e mentores de Educação à Distância (EAD) com pouca ou nenhuma experiência em programação, incluir em páginas HTML, tabelas dinâmicas que interpretem dados que nelas forem preenchidos, interagindo com o aluno. Protótipos destas tabelas, com nome, descrição, e-mail, número de linhas e colunas, componentes, fórmulas e mensagens, são criados e armazenados em um banco de dados, que é acessado em tempo real toda vez que uma página HTML com chamadas ao TABIPRO, for carregada em um *browser*. De ampla abrangência, a princípio este trabalho foi concebido para ser aplicado no módulo de Mecânica Gráfica do Programa Educ@r, em curso complementar de ensino à distância de Física para o nível médio, com uso da Internet como meio de comunicação e distribuição de informações. Através do TABIPRO, o aluno que participa de um treinamento ou curso à distância, poderá no processo de avaliação ter o conteúdo de suas respostas corrigidas imediatamente por este software, que ainda permite ao professor acompanhar o desenvolvimento do aluno, comentar suas respostas e direcioná-lo para um melhor aproveitamento do curso. Uma breve explanação do conceito de EAD e a apresentação das ferramentas e métodos utilizados para o desenvolvimento do TABIPRO são apresentados nessa dissertação.

Abstract

The objective of this paper is the development of a software, the Programmable Interactive Tables (TABIPRO), which allows professors and teachers of Distance Education with little or no experience in programming, to include in HTML pages, dynamic tables that interpret data filled in them, interacting with the student. Prototypes of these tables, with name, description, e-mail, number of lines and columns, components, formulas and messages, are created and stored in a database, which is accessed on line everytime a HTML page with calls to TABIPRO, is loaded in a browser. At first this software was created to be applied to a module of Graphic Mechanics of the Educ@r Program, in an extra distance course of Physics, for High School, using the Internet as a communication and information distributor means. Through the TABIPRO, the students who join a distance training or course, could have their answers immediately corrected by this software during the evaluation process, allowing the teacher to follow the students' development, comment the answers and direct them to the best result. A brief explanation of the concept of Distance Education and a presentation of the tools and methods used for the TABIPRO development are present in this paper.

1. Introdução

Nos dias de hoje a educação é sem dúvida alguma, um dos principais pilares de sustentação das relações interpessoais, sociais e profissionais, influindo diretamente na entrada ao mercado de trabalho e principalmente na permanência com sucesso no mesmo. A sociedade da era da globalização testemunha inovações tecnológicas que surgem com muita rapidez. Atentas a estas mudanças, pessoas dos mais variados níveis, como alunos do ensino fundamental, do ensino médio, graduandos, pós-graduandos e profissionais no mercado de trabalho, têm gerado expressiva demanda por uma educação continuada, buscando no dinamismo da área de informática um meio de aperfeiçoamento, reciclagem e qualificação em resposta à competitividade do mercado.

A revolução tecnológica contribui com uma geração de alunos que crescem em ambientes ricos de multimídia, com expectativas de mundo diferente de gerações anteriores (Santos et al, 2001). Neste contexto, o computador passa a ser utilizado como uma ferramenta educacional poderosa que pode ser usada para facilitar, estimular e consolidar o aprendizado (Oliveira et al, 2000). No ambiente acadêmico, a informática tem sido aplicada como uma importante ferramenta de apoio na educação, inúmeros hardwares e softwares são desenvolvidos com o propósito de melhorar os ambientes educacionais. O computador passa a ser administrado como

um instrumento de acesso às informações e é muito útil nas tarefas realizadas por alunos e professores. Considere ainda a Internet que invade nossos lares, empresas, órgãos governamentais e entidades educacionais, com todo o potencial de distribuir e disseminar informações, o que a torna um ambiente interessante para a área educacional.

Tudo isso tem gerado interesse no desenvolvimento de ferramentas, cursos e ambientes de produção, publicação e disponibilização de materiais didáticos utilizando o ambiente Web.

Diante dessa realidade, a educação à distância (EAD) principalmente utilizando-se da Internet como meio de comunicação, tem se apresentado como uma alternativa de aprendizagem adequada ao momento atual, onde as pessoas possuem vida agitada e com necessidades de atualizações que se contradizem com a disponibilidade de tempo e de deslocamento para as instituições de ensino.

1.1. Considerações

A figura do professor hoje é diferente de alguns anos atrás. Ele não é mais o detentor do conhecimento, aquele que sabe tudo, sendo seus alunos meros receptores do conhecimento. Com milhares de informações que estão ao alcance de todos, principalmente na Internet, o trabalho isolado do professor não se complementa com a necessidade dos alunos na busca pela construção do conhecimento. Trabalhar em conjunto e em cooperação, passa a ser uma necessidade dos alunos e o professor entra como mediador, orientador, aquele que mostra os

caminhos para os alunos em conjunto buscarem de forma interativa o saber e a construção de novos saberes (Moura et al, 2001).

Ensino e aprendizagem hoje são processos dinâmicos que se alternam entre seus participantes, principalmente quando a tecnologia eletrônica é utilizada como recurso de ensino (Leite, 2002).

A informática tem sido gradativamente inserida no processo ensino-aprendizagem, constituindo um novo paradigma através da utilização de recursos computacionais como agentes que colaboram para a construção do conhecimento (Valente, 1999; Valente, 1999a; Valente, 2000).

Valente (1993) destaca algumas formas de aplicação computacional na educação.

"

- **softwares tutoriais:** Constituem uma versão computacional da instrução programada. A vantagem dos tutoriais é o fato de o computador poder apresentar o material com outras características que não são permitidas no papel como: animação, som e a manutenção do controle da performance do aluno, facilitando o processo de administração das lições.
- **softwares do tipo exercício-e-prática:** Tipicamente são utilizados para revisar material visto em classe, principalmente os que envolvem memorização e repetição como aritmética e vocabulário. Estes programas requerem a resposta freqüente do aluno, propiciam *feedback* imediato, exploram as características gráficas e sonoras do computador e geralmente são apresentados na forma de jogos.
- **softwares de jogos educacionais e simulação:** Nesse caso, a pedagogia utilizada é a exploração auto dirigida ao invés da instrução explícita e direta.

Os proponentes desta filosofia de ensino defendem a idéia de que o aluno aprende melhor quando é livre para descobrir relações por ele mesmo, ao invés de ser explicitamente ensinado. A simulação envolve a criação de modelos dinâmicos e simplificados do mundo real. Estes modelos permitem a exploração de situações fictícias, de situações com risco, como manipulação de substância química ou objetos perigosos; de experimentos que são muito complicados, caros ou que levam muito tempo para se processarem, como crescimento de plantas; e de situações impossíveis de serem obtidas, como um desastre ecológico.

- **O computador como ferramenta educacional:** O computador não é mais o instrumento que ensina o aluno, mas a ferramenta com a qual ele desenvolve algo, e portanto, o aprendizado ocorre pelo fato de estar executando uma tarefa por intermédio do computador. Estas tarefas podem ser a elaboração de textos, usando os processadores de texto; pesquisa de banco de dados já existentes ou criação de um novo banco de dados; resolução de problemas de diversos domínios do conhecimento e representação desta resolução segundo uma linguagem de programação; controle de processos em tempo real, como objetos que se movem no espaço ou experimentos de um laboratório de física ou química; produção de música; comunicação e uso de rede de computadores. "

A utilização de computadores em ambientes educacionais tem aumentado com a evolução da Internet, que tem sido explorada para disponibilização e utilização de materiais didáticos no ambiente Web, além de mecanismos de comunicação, tais como teleconferência e videoconferência (Lucena, 1997).

Neste contexto a busca de mecanismos capazes de acelerar o processo de comunicação e transferência de informação entre as pessoas não é um acontecimento atual, entretanto as possibilidades tecnológicas atuais estão ampliando a capacidade de comunicação e alterando seus mecanismos, em um processo que certamente irá modificar muitos hábitos que desenvolvemos até hoje (Naves, 1998).

Os cursos de EAD proporcionam oportunidade às pessoas que se encontram distantes dos centros de difusão de informação de se capacitarem, transpondo barreiras do tempo, do espaço e da falta de recursos financeiros.

Com o avanço das tecnologias de comunicação e com a disseminação da Internet, muitos projetos de EAD têm sido desenvolvidos, utilizando o potencial desta rede em benefício da educação.

O oferecimento de um curso à distância utilizando o ambiente Web pode ser dividido em duas etapas distintas como descreve Silva (2000) a seguir:

A primeira etapa é a de criação do curso onde o professor/autor seleciona os conteúdos e disponibiliza-os.

A segunda etapa é a aplicação/gerenciamento do curso, visualizada em dois contextos. No primeiro contexto fica em evidência o gerenciamento de materiais didáticos e conteúdos do curso, através de mecanismos de disponibilização e monitoramento de materiais. No outro contexto fica o gerenciamento das atividades didáticas, que deve controlar da definição a realização de todas atividades previstas para o curso, como indicações de leitura, realização de trabalhos, exercícios e avaliações.

1.2. Objetivos

Este projeto insere-se na etapa de gerenciamento das atividades didáticas de um curso à distância, tendo como objetivo principal criar uma metodologia que resulte em um software, o TABIPRO. É interessante que este software seja de fácil utilização e permita criar tabelas dinâmicas que possam ser programadas pelo professor, inseridas em páginas de cursos na Web e utilizadas pelos alunos para verificar seus conhecimentos de forma rápida e interativa. É importante ainda que o software através das tabela interativas, possibilite ao professor avaliar e manipular as informações produzidas pelos alunos.

2. Educação à Distância (EAD)

As atuais possibilidades das novas tecnologias da informação e de comunicação incentivam o rápido crescimento da EAD como uma modalidade de ensino que permite aproximar o saber do aluno, levando em conta os limites individuais e as distâncias: espacial, temporal, tecnológica, psicossocial e sócio-econômica. O aluno, então, pode aprender no seu contexto imediato, planejar no tempo e no espaço suas atividades de estudo e seguir o seu ritmo de aprendizagem (Vieira, 2001).

Magalhães (1997) diz que a EAD se apresenta como uma das maneiras para resolver o problema da concentração das potencialidades de centros de excelência, sendo um meio eficaz de levar estas potencialidades a locais que não possuem tanta tecnologia e estão geograficamente distantes de centros mais desenvolvidos.

Para Spennemann (1995) a EAD não deve apenas resolver as questões das grandes distâncias. Sua composição pedagógica deve atender as necessidades de interatividade do aluno com o tema de estudo, utilizando-se das tecnologias disponíveis, como forma de aperfeiçoar os aspectos pedagógicos do ensino, atingindo os principais fatores de uma educação centrada no aprendizado interativo, dinâmico e contextualizado.

"A Educação à Distância é uma aprendizagem planejada que normalmente ocorre em um local diferente de uma sala de aula tradicional e como resultado requer projeto de curso e técnicas instrucionais especiais, métodos especiais de comunicação eletrônica ou outra tecnologia, bem como sistemas organizacionais e administrativos especiais" (Moore & Kearsley, 1996).

É cada vez mais comum nos ambientes de EAD atuais que o processo de aprendizagem valorize a interação entre os intervenientes. Neste contexto, passa a ser uma atividade fundamental do professor analisar estas interações de modo a incentivar a participação dos alunos, estimular a resolução cooperativa de problemas e a busca pela solução conjunta (Jaques et al, 2000).

Tanto na educação convencional quanto em EAD, o papel do professor no ambiente colaborativo deve ser o de orientador e desafiador. Ele deixa de ser o provedor de informações para ser gerenciador de entendimento. Caberá ao docente, motivar o grupo e monitorar a participação dos alunos, levando em conta os objetivos e interesse do grupo (Andrade & Beiler, 1999).

Em um curso à distância o aluno é o maior responsável pelo aprendizado. Ele deve estar ciente e ser orientado sobre a importância desta responsabilidade. A não proximidade do instrutor, apesar de todos os mecanismos de interação providos no curso, força o aluno a caminhar sozinho e tentar resolver os problemas através de seu próprio raciocínio. Entretanto, o professor deverá orientar e monitorar as atividades do aluno, fazendo com que ele sinta-se seguro e motivado.

Se motivado, no decorrer do curso o aluno percebe o potencial do seu aprendizado e espontaneamente torna-se responsável em planejar as metas a serem atingidas, deixando de ser passivo e passando a atuar de forma ativa no processo ensino-aprendizagem.

Um sistema EAD deve enfatizar a colaboração e a comunicação entre os envolvidos nas atividades do mesmo, havendo interatividade entre professor e aluno; entre os alunos; e entre aluno e o ambiente de aprendizado.

Segundo Matos (2002) abaixo estão as principais vantagens da EAD:

- Pode ser alcançado um número maior de alunos.
- O aluno não precisa se deslocar aos locais tradicionais de ensino, exceto quando necessário utilizar laboratórios ou oficinas.
- O aluno determina os horários e ritmo de estudo. Esta flexibilidade possibilita melhor aproveitamento do tempo.
- É possível receber contribuições de pessoas que, por razões de disponibilidade, não o poderiam fazer em um sistema tradicional.

Estas vantagens podem ser potencializadas se houver uma grande interatividade entre todos os envolvidos neste processo através de um ambiente dinâmico.

2.1. Caracterização da EAD

A EAD oferece ao aluno a opção de estudar e aprender de uma forma mais autônoma e em locais mais convenientes. O aluno tem escolhido este método de estudo devido à flexibilidade que este sistema de ensino oferece. Porém, encontrar o nicho adequado e satisfazer as reais necessidades instrucionais que os alunos necessitam é que torna um programa EAD eficaz.

A EAD é caracterizada por:

- Pela separação física e/ou temporal entre o professor e o aluno (Keegan, 1986; McIsaac & Gunawardena, 1996);
- O controle da aprendizagem é realizado mais intensamente pelo aluno do que pelo professor instrutor (Jonassen, 1992);
- A interação entre alunos e professores é mediada por alguma forma de tecnologia de comunicação (Keegan, 1986; Kearsley, 1995).

No ensino presencial, a instrução é mais centralizada no professor, sendo que em EAD esta centralização ocorre no aluno. Este então, passando por novas e diferentes situações, assume mais responsabilidades no processo ensino-aprendizagem, devendo estar atento à formação do conhecimento pessoal.

Nesta nova situação os alunos estão mais propensos a pensar, raciocinar e criticar, coletar e analisar informações, refletir os entendimentos anteriores, argumentar e defender suas idéias, construindo assim seus próprios pensamentos e o professor deve alterar sua forma de atuar, passando de um transmissor de informação para um orientador da aprendizagem.

De um modo geral, o professor põe-se à disposição do aluno, auxiliando-o na construção do seu próprio caminho do conhecimento, orientando-o na aprendizagem, ajudando-o a esclarecer dúvidas, identificando suas dificuldades, sugerindo novas leituras e atividades, organizando suas atividades de estudo em grupo, supervisionando a prática de laboratório, entre outros (Magalhães, 1997).

A interação entre professor e aluno deve ser possibilitada por algum tipo de tecnologia de fornecimento de informação, propiciando assim *feedback* para o aluno à distância (McIsaac & Guanawardena, 1996).

Um processo de comunicação ocorre em pelo menos duas vias. Todas as pessoas ao se comunicarem esperam um retorno ao que foi colocado. Em EAD isto não é diferente. Se os participantes do processo ensino-aprendizagem não recebem retorno sobre suas interações, eles gradativamente se sentirão desmotivados e poderão interromper a via de comunicação. Prover retorno é responsabilidade primária do professor. Toda etapa vencida pelo aluno deve gerar algum tipo de retorno por parte do professor, mesmo que seja um breve comentário.

O processo de comunicação na EAD é muito importante como canal propiciador de ambientes que gerem a interação, a cooperação, a produção do conhecimento e a possibilidade de customização ou personalização da educação (Gomes et al, 2000).

A comunicação é um ponto essencial em EAD, não só os meios com os quais se realiza a comunicação, mas também o aspecto da mensagem, que deve ser a mais completa, objetiva e clara possível, não permitindo interpretações dúbias, facilitando o entendimento do outro (Martins et al, 2001).

Segundo o "Manual de Criação e Elaboração de Materiais para Educação à Distância" comentado por Franco (2001), escrever para EAD é diferente de escrever em geral, pois "a redação para EAD é essencialmente didática, com uma forte obrigação no sentido de comunicar-se com os leitores, e com uma missão social muito clara". São aspectos importantes que o texto seja dividido em seções auto-suficientes, com parágrafos que apresentem apenas uma ou duas idéias relacionadas, que se faça uso de subtítulos para apresentar um novo assunto, utilize elementos de transição entre seções ou parágrafos e tenha recapitulação das idéias principais no fim de cada seção.

2.2. História da Educação à Distância

Magalhães (1997) destaca a evolução histórica da EAD em quatro estágios: cursos por correspondência; *Open Universities*; transmissão via rádio, via TV e teleconferência; multimídia e redes de computadores.

Por volta de 1840, na Inglaterra nascia o primeiro curso por correspondência. Issac Pitman ensinando taquigrafia foi pioneiro na transmissão de conhecimentos utilizando-se deste método. Em 1856, o francês *Charles Toussaint* e o alemão *Gustav Langenscheidt* começaram a ensinar línguas por correspondência. No estado de New York – EUA, o Chautauqua Institute foi autorizado a fornecer diploma pelo método de ensino à distância, sendo um marco na respeitabilidade acadêmica do curso por correspondência (Moore & Kearsley, 1996).

Estes cursos e muitos outros, foram criados com objetivo de dar formação a um grupo de pessoas que, por motivos geográficos, econômicos ou sociais, não podiam deslocar-se aos tradicionais locais de ensino (Matos, 2002).

A partir de 1950, as instituições de ensino primário, secundário e superior à distância, começaram a fornecer diplomas reconhecidos, dando um avanço para o desenvolvimento das *Open Universities*. A primeira Instituição a usar esta forma de EAD foi a *British Open University* estabelecida em 1969 (Magalhães, 1997).

No rádio tem-se registro de transmissão educacional desde a década de 20 e de uso da TV no final dos anos 70, introduzindo-se séries educativas de televisão projetadas para os alunos, formato este ainda utilizado nos dias atuais, sendo consideradas fontes complementares de informação. A teleconferência adicionou duas importantes dimensões para a EAD: grupos de alunos juntos em cada local e interação em tempo-real entre alunos e instrutor (Moore & Kearsley, 1996).

Com a disseminação dos microcomputadores ocorrida a partir da década de 1980, duas tecnologias têm sido muito utilizadas em EAD, a multimídia e as redes de computadores.

A multimídia revolucionou as técnicas de apresentação de material didático no formato digital. Texto, som e imagens estáticas e em movimento são utilizadas paralelamente otimizando e melhorando a qualidade do material didático. Por sua vez, as redes de computadores, principalmente a Internet, permitem com facilidade, o trânsito de todo este material didático das instituições de ensino aos alunos, dos alunos para as instituições e entre os próprios alunos.

A utilização de recursos tecnológicos de informática em sistemas de EAD fomenta a qualidade de ensino em larga escala, tornando esta modalidade de educação mais confiável e mais acessível. Estes recursos permitem o desenvolvimento de ambientes interativos de ensino, potencializando o processo ensino-aprendizagem. A hipermídia, a qual integra textos, imagens, vídeos, fotos, som e animação, pode proporcionar ambientes de ensino atraentes.

Recentemente, a partir da disseminação das tecnologias de EAD e suas possibilidades de apoio pedagógico, cada vez mais tem se procurado o ensino colaborativo, a interação, o diálogo, a troca de habilidades e conhecimento, em detrimento ao ensino voltado a auto-instrução e ao individualismo. A aprendizagem colaborativa pode ser propiciada através de ambientes que possibilitam a comunicação, a troca de idéias e a tomada de decisões. Ao utilizar estes ambientes, os alunos desenvolvem modelos mentais, realizam atividades que motivam a aprendizagem através de interações efetivas (Andrade & Beiler, 1999).

É importante ressaltar que com a Internet, principalmente através da Web, a EAD deu um salto de dinamismo, de qualidade nos materiais didáticos e na velocidade de comunicação entre as partes envolvidas no processo.

2.3. A Estrutura de um Sistema EAD

Para o desenvolvimento de um sistema de EAD vários subsistemas devem ser analisados, implementados e estarem interagindo de forma coesa entre si. São eles os subsistemas organizacionais, administrativos, instrucionais e tecnológicos.

A EAD faz parte do sistema educacional como um todo, o qual inclui escolas, departamento de treinamento etc. Portanto, quando um programa de EAD é implementado afeta de alguma maneira todo o sistema educacional da organização que o implementa (Moore & Kearsley, 1996).

O ensino através de EAD pode obter resultados tão eficazes quanto o ensino em sala de aula tradicional, desde que sejam utilizados apropriadamente tecnologia e métodos adequados para a aprendizagem neste sistema, fortalecendo a comunicação e interatividade entre os envolvidos no processo ensino-aprendizagem, propiciando assim, *feedback* em tempo hábil.

O êxito da EAD depende, além de sistemas e programas bem definidos, de recursos humanos capacitados, de material didático adequado e principalmente, de meios apropriados para fornecer instrução aos alunos (Magalhães,1997).

2.4. Fases de um Sistema de EAD

Seligman (1992) apud Magalhães (1997) diz que uma experiência educacional de qualidade deve incluir os seguintes elementos:

a) qualidade dos materiais de aprendizagem;

b) disponibilidade dos materiais;

c) suporte para alunos através de uma equipe bem treinada;

d) mecanismos de monitoramento;

e) *feedback* para a melhoria do sistema.

Preparar e estruturar adequadamente um curso para EAD é essencial para o sucesso do mesmo. O método de aprendizagem não pode dificultar ao aluno a aquisição dos conhecimentos. Um curso bem estruturado torna-se de fácil implementação e permite alterações e correções quando necessárias, sem grandes dispêndios de tempo e recursos humanos.

Cursos à distância precisam ser cuidadosamente planejados e buscar caminhos para estimular a participação do aluno. O desenho do curso deve incluir estratégias de ensino-aprendizagem que estimulem a participação do aluno e o alcance dos objetivos (Leite, 2002).

Segundo Willis (1992) o desenvolvimento instrucional (figura 1) é primordial para um sistema EAD. Fornece um processo e ambiente de planejamento, desenvolvimento e adaptação da instrução, baseada nas necessidades do aluno e nos requisitos do conteúdo. Nele, são definidas as etapas de projeto, desenvolvimento, avaliação e revisão, que se implementam num ciclo.

Na fase de projeto, definem-se as prioridades, os objetivos e qual será o público alvo. No desenvolvimento, ocorre a definição do perfil e estruturação dos conteúdos a ensinar, revisão e aproveitamento de material já publicado e métodos de oferecimento. Na avaliação, através de coleta e análise de dados, deve ser verificado se os objetivos estão sendo alcançados. E por último na revisão, retificam-se os possíveis erros, reiniciando todo o processo. As revisões geralmente são baseadas em *feedback* dos instrutores, especialistas e alunos.

Para Willis (1992) o processo de desenvolvimento instrucional é composto de quatro etapas:

- Projeto: Dividido em determinar as necessidades, definir o público-alvo e estabelecer metas;
- Desenvolvimento: Dividido em estruturar publicações dos conteúdos, revisar material já existente, organizar e desenvolver o conteúdo, e desenvolver materiais e métodos de fornecimento do curso;
- Avaliação: Dividido em rever metas e objetivos, desenvolver estratégias de avaliação, e coletar e analisar dados;
- Revisão: Dividido em desenvolver e implementar plano de revisão.

Seguir os princípios do desenvolvimento instrucional, não significa superar todos os obstáculos encontrados no desenvolvimento de programas eficazes de EAD, entretanto, eles subsidiarão procedimentos para nortear mudanças instrucionais quando necessárias (Willis, 1992).

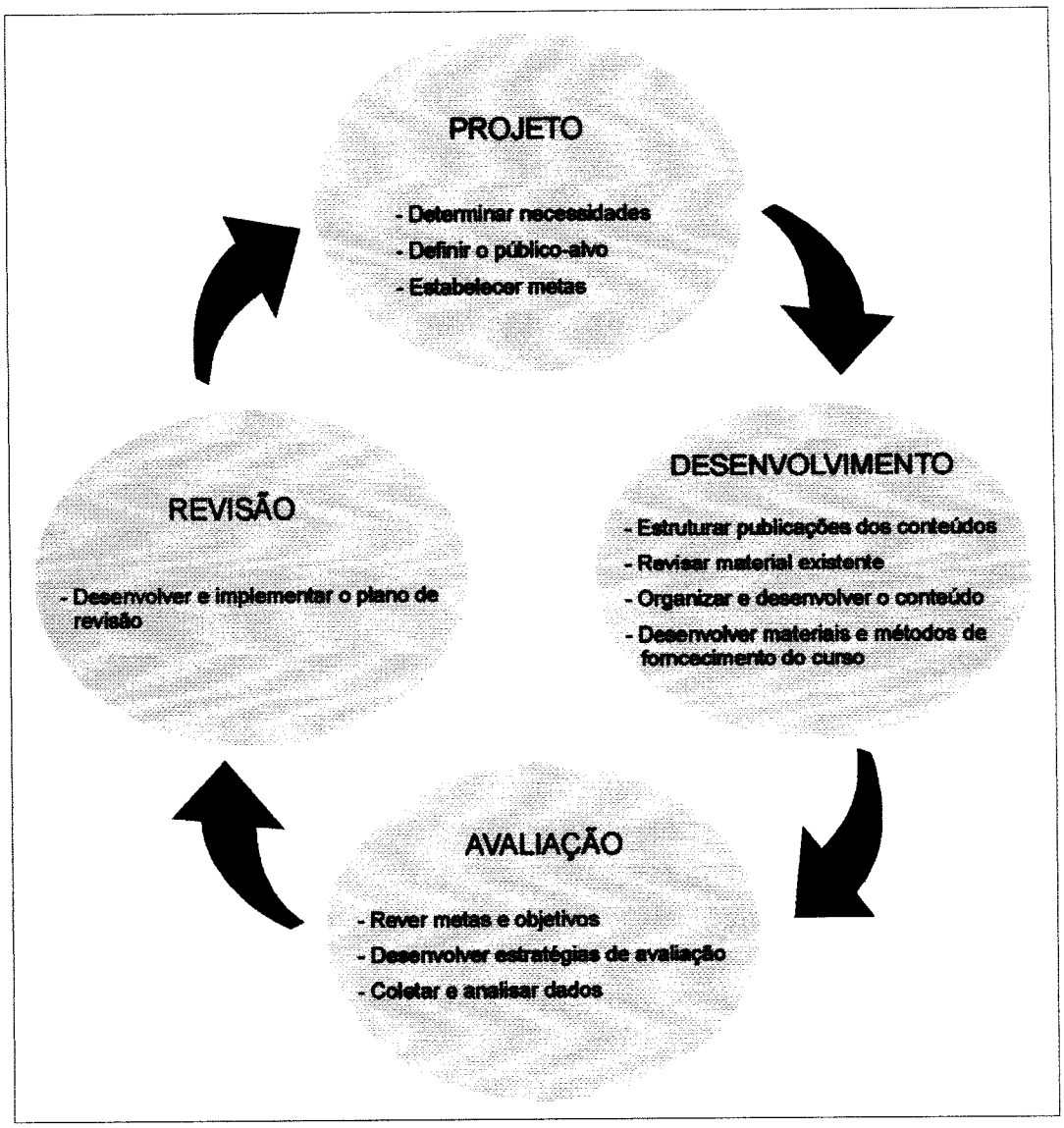


Figura 1 – Processo de desenvolvimento instrucional. Fonte: Willis. 1992.

Magalhães (1997) sugere a reorganização das fases de um sistema de EAD (figura 2), baseado no desenvolvimento do Programa Educ@r, apresentado mais à frente. Esta reorganização foi adaptada do Projeto de Esther Giraudo, da Universidade Nacional de Córdoba - Argentina, citado por Alves (1996).

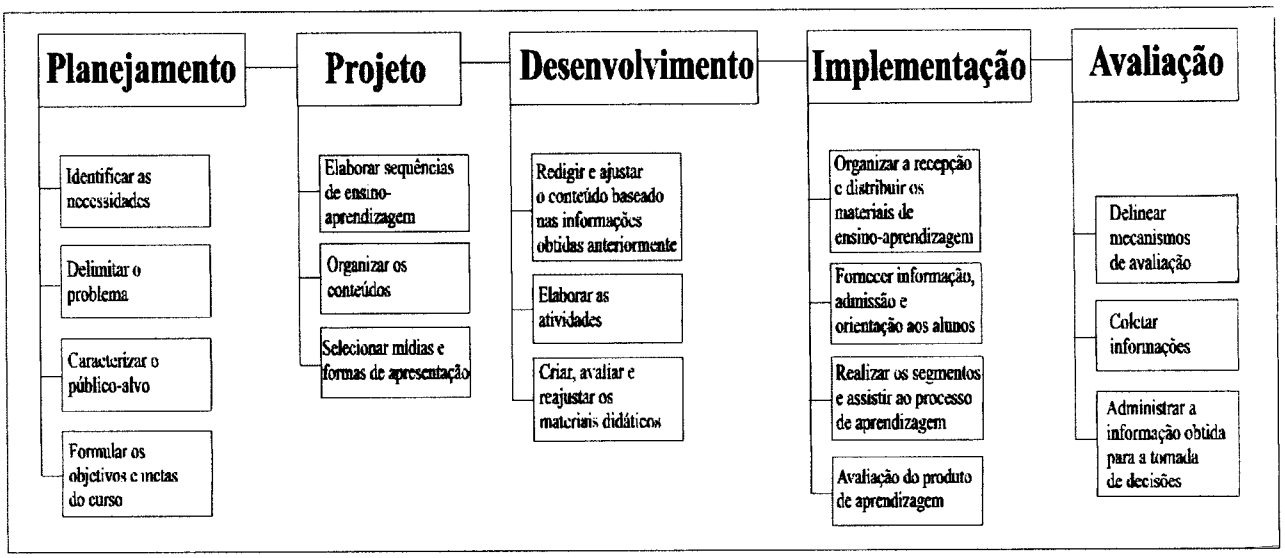


Figura 2 – Fases de um sistema de educação à distância. Fonte: Magalhães (1997).

2.4.1. Planejamento, Projeto, Desenvolvimento, Implementação e Avaliação

Durante o planejamento devem ser executados procedimentos para definir o problema e as necessidades; caracterizar o público-alvo; formular os objetivos e metas instrucionais, sendo que estes são estabelecidos com base na natureza do problema, bem como nas necessidades e características dos alunos.

Para o projeto deve-se elaborar seqüências de ensino-aprendizagem; organizar o conteúdo dos materiais; selecionar mídias e formas de apresentação dos materiais.

Na etapa do desenvolvimento é necessário redigir e ajustar o conteúdo baseado nas informações obtidas na etapa anterior; elaborar as atividades do curso; criar, avaliar e reajustar os materiais didáticos.

Os procedimentos para a implementação são organizar a recepção e distribuir os materiais de ensino-aprendizagem; fornecer informação, admissão e orientação aos alunos; realizar os segmentos e assistir ao processo de aprendizagem; avaliar o produto de aprendizagem.

A avaliação deve ser praticada continuamente nos ciclos de projeto, desenvolvimento e implementação, para garantir que tudo saia de acordo com o projetado e planejado (Moore & Kearsley, 1996).

O processo de avaliação é fundamental em EAD para manter o controle da qualidade do curso, facilitando a identificação das falhas no sistema instrucional e orientando quanto às mudanças necessárias para melhorar os resultados obtidos com o programa de EAD (Magalhães, 1997).

Um sistema de avaliação ajuda na identificação dos pontos fortes e críticos de um programa de EAD. Além de possibilitar tomar decisões para melhorar a qualidade do curso, o processo de avaliação indica a necessidade de novos desenvolvimentos.

O objetivo da avaliação é fornecer informações para a equipe do programa tomar decisões necessárias a fim de que os propósitos do projeto sejam atingidos. Os passos são delinear mecanismos de avaliação; obter informações; administrar a informação obtida para a tomada de decisões.

3. A Internet

Uma revolução no conceito de comunicação ocorreu com o surgimento da Internet. Arquivos de textos, sons, imagens estáticas e em movimento trafegam pela rede mundial, diminuindo ou extinguindo distâncias que antes dificultavam a comunicação entre pessoas no mundo todo.

Milhões de bytes estão trafegando pela Web neste instante, distribuindo informações em tempo real, do tipo texto, vídeo, sons, gráficos, dentre outros, de um lado para outro, sempre que solicitadas por pessoas em qualquer parte do mundo.

O aumento sistemático do número de usuários da Web, com características e finalidades diversificadas, gerando intensa troca de informações entre eles, tem impulsionado o crescimento da Internet e do ambiente Web, o que tem gerado em conseqüência, um interesse cada vez maior em disponibilização de material neste ambiente.

Segundo Silva et al (2000) a Internet é apresentada como um meio de distribuição de informação e de comunicação que oferece um ambiente propício para o desenvolvimento da EAD.

Conforme as tecnologias de comunicação tornam-se mais familiares, alunos e educadores têm encontrado maneiras úteis de incorporar os recursos da Internet na experiência educacional.

Um sistema de EAD mediado pela Internet envolve diversos componentes como: aprendizagem, ensino, comunicação, desenho e gerenciamento, além de toda parte editorial. Assim, cursos inseridos nestes sistemas devem ser caracterizados por um planejamento adequado e pelo estabelecimento claro da forma de operacionalizar o processo através da mídia (Vieira, 2001).

3.1. A World-Wide Web (Web)

A Web é a estrutura arquitetônica que permite acesso a documentos vinculados espalhados por milhares de máquinas na Internet. Em poucos anos, ela deixou de ser um meio para distribuir dados extremamente técnicos (que era sua finalidade inicial) para se tornar a aplicação que milhões de pessoas consideram ser "A Internet". A enorme popularidade deve-se à sua interface gráfica colorida, que é de fácil utilização mesmo para principiantes. Além disso, ela oferece uma imensa variedade de informações sobre quase todos os assuntos imagináveis (Tanenbaum, 1997).

Segundo Dillenbourg & Schneider (1995) a Web é um sistema hipermídia executado na Internet, onde documentos hipertexto são utilizados como veículo de informação padrão. Estes documentos são elaborados em HTML (HyperText Markup Language) e podem ser interativos contendo textos, sons, figuras e vídeos. A partir de um documento HTML e utilizando-se de *links* é possível recuperar outros documentos disponíveis em qualquer localidade na Internet, escritos por autores diferentes em datas diferentes. A utilização de *links* possibilita disponibilizar informações de forma não linear, o que é um fator de sucesso da Web.

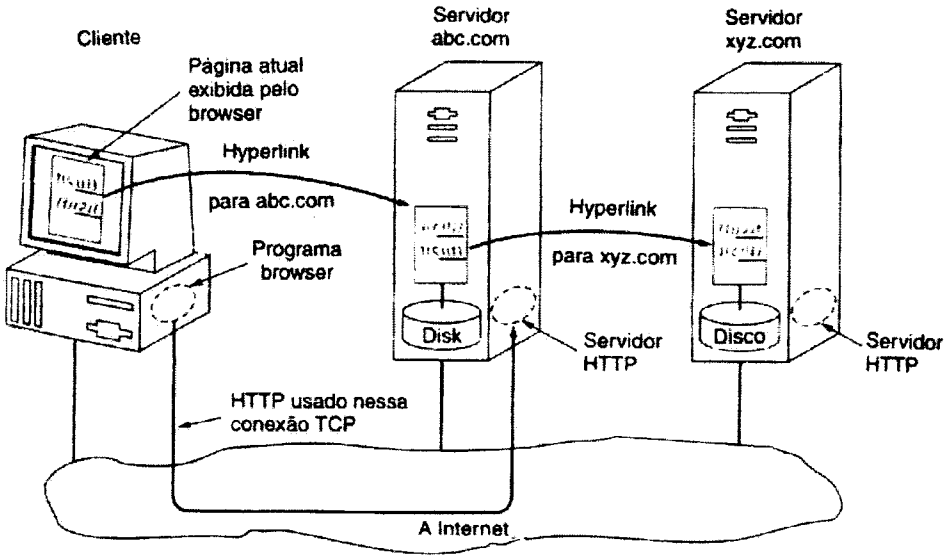


Figura 3 – As partes do modelo da Web. Fonte: Tanenbaum (1997).

Para Oliveira et al (2000), a Web como tecnologia tem despertado grande interesse nos meios educacionais, devido à sua capacidade para circulação de informações nas mais diferentes formas (áudio, vídeo, texto), interatividade e pela independência de plataforma. A Web é uma ferramenta que pode fazer parte de um ambiente de aprendizagem, combinando aspectos colaborativos de aprendizagem baseada em sala de aula com a flexibilidade proporcionada pela aprendizagem baseada em computador, permitindo que o aluno estude no seu próprio ritmo.

De modo especial, o ambiente Web tem sido largamente explorado na construção de aplicações para ambientes de ensino-aprendizagem onde professores e alunos interagem durante a realização de um curso. O uso do ambiente Web na educação torna-se ainda mais atraente frente à possibilidade de inserção de recursos multimídia que incrementam o processo de ensino-aprendizagem (Silva et al, 2000).

3.2. Arquitetura Web

A Web pode ser vista como um serviço de aplicação da arquitetura TCP/IP (Transmission Control Protocol/Internet Protocol) e como tal, a arquitetura da Web define um esquema de endereçamento (URL - Uniform Resource Locator) no nível da aplicação, estabelece protocolos de sessão (soquetes TCP/IP) e apresentação (HTML e auxiliares), define um protocolo (HTTP - HyperText Transfer Protocol) no nível da aplicação e segue o modelo cliente/servidor (Ricarte, 2000).

Seguindo um modelo de sistema cliente/servidor, de um lado está o módulo cliente (ou seja, usuário), que através de um *browser* solicita documentos à rede. Do outro lado está o módulo servidor, que atende aos pedidos solicitados pelo cliente. Então, a Web compreende múltiplos servidores, que podem enviar documentos solicitados para os usuários da Internet que através de um *browser*, "navegam" de servidor para servidor.

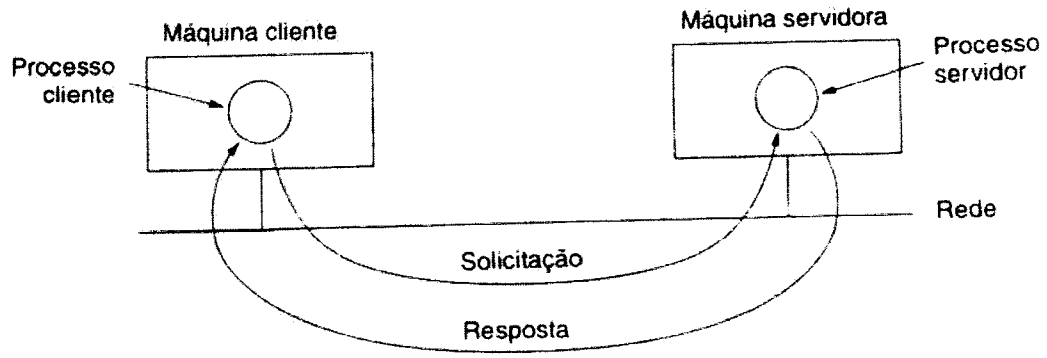


Figura 4 – O modelo cliente/servidor. Fonte: Tanenbaum (1997).

Um soquete TCP estabelece uma conexão bidirecional entre uma aplicação cliente e outra aplicação servidora. A aplicação cliente envia solicitações de serviços e recebe retornos das suas solicitações. A aplicação servidora monitora

constantemente a chegada de solicitações de serviço. Quando alguma solicitação é recebida, a aplicação servidora executa o serviço e utiliza a conexão para enviar o retorno com os resultados do serviço (Ricarte, 2000).

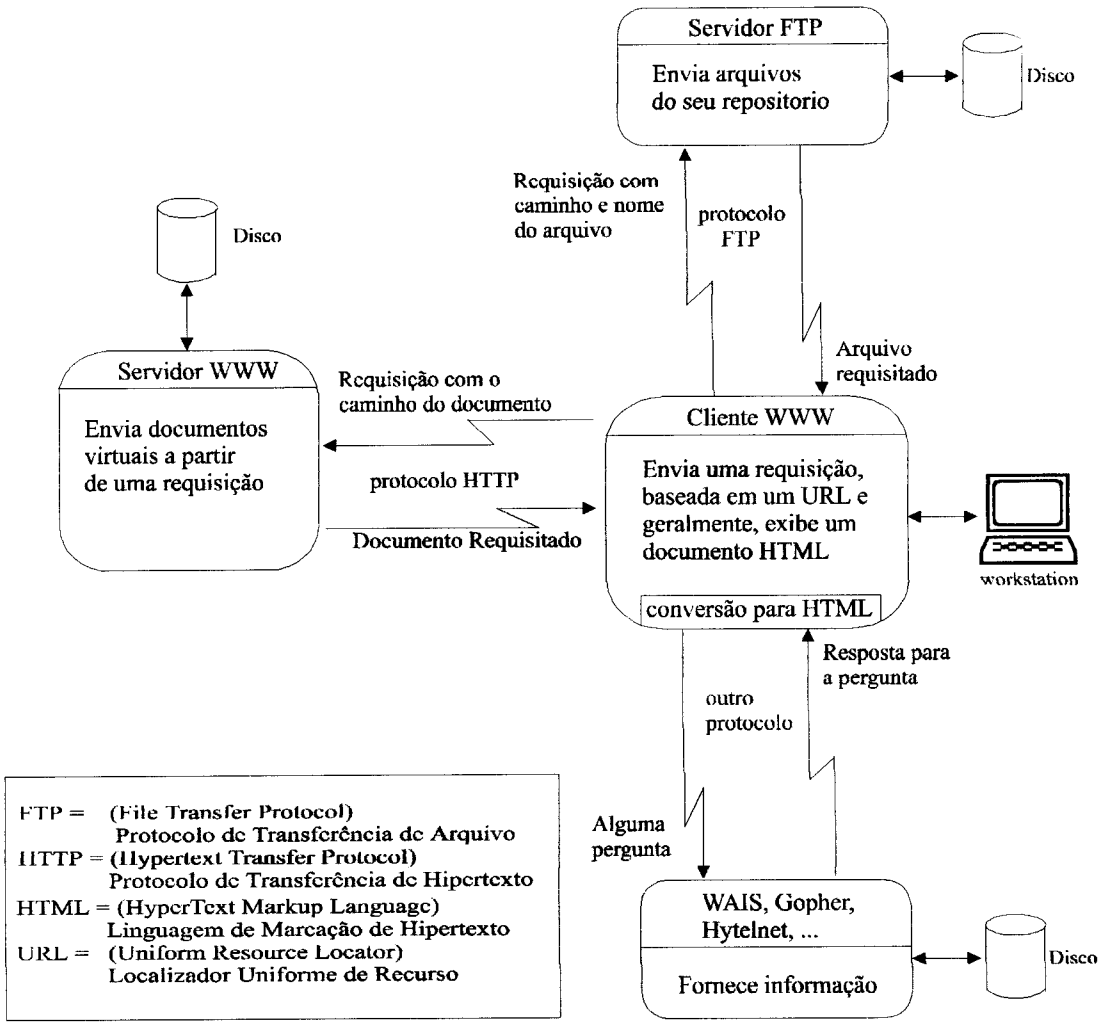


Figura 5 - Configuração Cliente-Servidor da WWW. Fonte: Ibrahim (1994).

3.2.1. O Módulo Cliente

Do ponto de vista do usuário, a Web é uma vasta coleção mundial de documentos, geralmente chamados de páginas. Cada página pode conter *links* para

outras páginas relacionadas em qualquer lugar do mundo. Os usuários podem seguir por um *link*, que os leva até a página apontada. Esse processo pode ser repetido indefinidamente, possibilitando aos usuários percorrer centenas de páginas. As páginas que apontam para outras páginas utilizam o hipertexto.

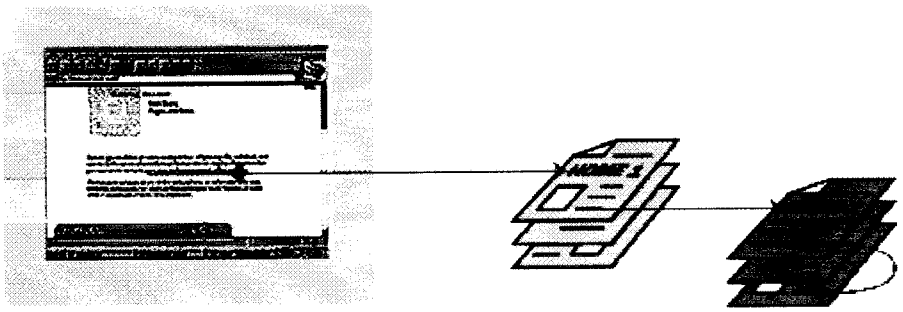


Figura 6 - Representação de *links* entre documentos Web.

As páginas são visualizadas com o auxílio de um programa denominado *browser*, dentre os quais o Netscape e o InternetExplorer são muito conhecidos. O *browser* busca a página solicitada, interpreta seu texto e seus comandos de formatação e a apresenta, corretamente formatada, na tela do computador.

As aplicações clientes na Web são usualmente navegadores *(browsers)* responsáveis pela apresentação de documentos HTML e pela solicitação de um recurso a servidores Web. O recurso pode ser um documento HTML ou um arquivo contendo informação texto ou binária (imagem, áudio, vídeo, applet Java, etc), conforme estabelecido pelo elemento que originou a solicitação.

Nem todas as páginas podem ser vistas da maneira convencional. Por exemplo, algumas delas consistem em sons, imagens ou os dois. Quando páginas de hipertexto são misturadas a outros meios de comunicação, o resultado é o que chamamos de hipermídia.

Não necessariamente o software navegador (*browser*) precisa saber manipular todos os tipos de recursos, podendo ele ativar rotinas auxiliares de exibição para tipos não reconhecidos. As rotinas auxiliares podem ser ativadas sob a forma de programas executáveis externos ao navegador ou através de *plug-ins* (Ricarte, 2000).

3.2.2. O Módulo Servidor

Todos os sites da Web têm processos servidores que aguardam conexões dos clientes (geralmente *browsers*). Depois de estabelecida a conexão, o cliente envia uma solicitação e o servidor envia uma resposta. A conexão é, então, liberada. O protocolo que define as solicitações e respostas válidas é chamado de HTTP (Tanenbaum, 1997).

Servidores Web são responsáveis por atender a solicitações de clientes, tendo como funcionalidades básicas organizar e gerenciar os recursos HTTP, prover acesso seguro a esses recursos e processar *scripts*.

Alguns dos protocolos de comunicação (Tanenbaum, 1997):

- HTTP (HyperText Transfer Protocol) – protocolo usado nas conexões às páginas disponíveis em sites na Web.
- FTP – (File Transfer Protocol) – protocolo utilizado para a transferência de arquivos de uma máquina para outra na Internet.
- Wais (Wide Area Information Service) – É um serviço que permite que o usuário tenha acesso à base de dados na Internet e efetue buscas por documentos através de palavras-chave.
- Gopher – Facilita a localização de arquivos através de menus.

3.2.3. Padrões para Comunicação

A Word Wide Web é baseada em três padrões abertos que permitem transferir a informação, descrever a formatação da informação e localizar a informação distribuída pela rede (Sá, 2000).

- **Transferir a informação**

  O protocolo de transferência padrão da Web é o HTTP. Cada interação consiste em uma solicitação seguida de uma resposta (Tanenbaum, 1997).

  O HTTP é o protocolo de comunicação para o transporte de informações entre o servidor e o cliente Web. É um protocolo aberto e especializado na transmissão de documentos Web na Internet. O protocolo HTTP é um protocolo *stateless*, ou seja, o cliente Web envia o pedido de uma operação ao servidor Web que atende ao pedido. Logo em seguida a conexão é encerrada e nenhuma informação sobre a solicitação do usuário é mantida no servidor Web.

- **Descrever a formatação da informação**

  Atualmente, os documentos disponibilizados na Web são estruturados de acordo com o padrão HTML, que é suportado pelos *browsers* existentes, e que define uma estrutura simples dos hiperdocumentos.

  A HTML é uma linguagem que descreve como os documentos devem ser formatados. É uma linguagem padrão para produzir documentos hipertexto na Web. Possui formato baseado no SGML (Standard Generalized Markup Language) e pode ser utilizada através de várias ferramentas, desde simples editores de textos a sofisticados softwares de autoria. Usa *tags* como <h1> e

</h1> para estruturar o texto em cabeçalhos, parágrafos, listas e *links* de hipertexto (W3C, 2002a).

Por ser uma linguagem baseada no padrão SGML é independe da plataforma em que é exibida, cabendo ao cliente Web interpretá-la de acordo com sua configuração. Ela é composta de um conjunto de elementos, denominados *tags*, os quais permitem ao desenvolvedor especificar a estrutura do documento Web (W3C, 2002).

Um documento HTML pode ser estruturado com títulos, parágrafos, cabeçalhos, citações e vínculos hipertexto para outros documentos. Documentos codificados em HTML podem ser interpretados pelos *browsers* e formatados segundo as características de cada plataforma em que são exibidos.

- **Localizar a informação**

Para identificar e localizar documentos Web na Internet é proposta a utilização de um formato para endereçamento de documentos denominado URL (Uniform Resource Locator).

A cada página é atribuído um URL que funciona como o nome universal da página. Os URLs possuem três partes: o protocolo, o nome da máquina em que a página está e um nome local que indica a página específica (normalmente um nome de arquivo na máquina onde está armazenado).

Por exemplo, o URL para o programa Educ@r é:

http://educar.sc.usp.br/index.html

Esse URL está dividido em três partes: o protocolo (HTTP), o nome DNS do host (educar.sc.usp.br) e o nome do arquivo (index.html).

A sintaxe de um URL é :

<Protocolo>://<Host>/<Path>/<Doc>[<#Location>], onde:

- Protocolo indica o tipo de recurso Internet que deve ser usado para a conexão com os servidores, que pode ser HTTP, FTP, NEWS, WAIS e FILE;
- Host é o nome de um computador na Internet que tem acesso bidirecional completo a outros computadores. Ele armazena, centraliza e distribui arquivos, serviços de correio eletrônico, redes de impressão etc (Devices, 2002);
- Path é uma lista de diretórios separado por barras;
- Doc é o nome do documento ou de um programa a ser executado pelo servidor Web; e
- Location é uma marca textual opcional de posição no documento.

A comunicação entre cliente e servidor Web é sempre feita na forma de requisição/resposta e é sempre iniciada pelo cliente. O cliente envia uma mensagem de sincronização para o servidor pedindo abertura de conexão. O servidor responde aceitando ou não este pedido. Se aceitar, a conexão é estabelecida. Uma recusa pode indicar que o servidor está sobrecarregado e não consegue tratar novos pedidos de conexão que chegam.

Uma vez estabelecida a conexão, o cliente envia uma mensagem com a requisição feita pelo usuário, representada por um URL. Esta mensagem é enviada através da rede para o servidor, que busca a informação no disco local ou dispara a execução de uma aplicação externa. Após o processamento do pedido, uma mensagem de resposta é enviada ao cliente que, ao recebê-la, retorna um pedido de fechamento de conexão.

Quando o cliente recebe a informação solicitada, é necessário verificar se os dados são válidos, através de informações contidas em campos do cabeçalho da mensagem. Somente após esta verificação o cliente pode mostrar o documento para o usuário. Portanto, ele precisa saber qual o tipo e o formato do arquivo. Esta informação também é obtida no cabeçalho da mensagem. Todo este processamento impõe um atraso no cliente, que é refletido no tempo de resposta observado pelo usuário (Lima, 1997).

4. A Interatividade

Embora EAD possua grandes vantagens como as apresentadas no capítulo 2, muito se tem discutido sobre a ausência ou deficiência neste sistema de ensino de uma das principais características do ensino convencional, que é o contato direto entre o aluno e o professor e entre os próprios alunos. Deste contato resulta mesmo que involuntariamente uma interatividade sobre os tópicos abordados na aula que estimula o desenvolvimento do assunto, gerando análises críticas e ricas conclusões.

Educar requer comunicação e diálogo, implica compartilhar conhecimentos e experiências pessoais, pensar e criar individual e coletivamente, é construção do saber. É neste ambiente social que a interatividade acontece (Schechtman, 2001).

4.1. Comunicação e Interatividade

A comunicação e a interação entre pessoas fazem parte de nossos comportamentos como seres humanos que vivem em sociedade, onde desde muito cedo aprendemos formas diferentes de nos comunicar e interagir para apontar e resolver as necessidades. Romiszowski & Mason (1996) afirmam o aspecto de que a comunicação tenha essencialmente mais de uma via, no mínimo que possua duas vias como no caso de duas pessoas estarem trocando mensagens através de e-mail.

O processo de comunicação pode ser síncrono onde duas ou mais pessoas se comunicam ao mesmo tempo, como em uma sala de aula tradicional ou em teleconferência, ou assíncrono onde os participantes do processo não precisam estar se comunicando um com o outro ao mesmo tempo (Romiszowski & Mason, 1996).

Das várias formas de comunicação com recursos computacionais, a Web tem grande potencial para promover a interação entre os usuários (Hill, 1997). Sherry & Wilson (1997) mostram a importância de utilizar algum meio de comunicação eletrônica como caminho para prover interação, permitindo existir *feedback* nas execuções de tarefas à distância. Quando um grupo de alunos se reúne num ambiente interativo para prover suporte mútuo numa aprendizagem comunitária, resolvem seus problemas com maior facilidade e ajudam a resolver problemas de outros alunos, sendo possível rapidamente identificar e resolver problemas comuns ao grupo.

4.2. Interatividade na EAD

Na EAD, tal como o propomos, o aluno assume uma postura ativa, interage, dialoga e sobretudo vê-se diante do desafio de selecionar informações e atribuir-lhes significado. Assim se por um lado o ensino à distância promove um conceito de autonomia por parte do aluno, por outro lado aparece a necessidade de interação e de contado entre alunos e entre aluno/professor, resultando em demanda por novas maneiras pelas quais os alunos possam estar interagindo para receber apoio e realimentação, que é essencial para os manterem motivados (Tarouco, 2002). Uma

referência comum associada com EAD é a necessidade de interação do aluno com o professor e com outros alunos (Hill, 1997).

Segundo Ferreira (1993), interação é ação que se exerce mutuamente entre duas ou mais coisas ou duas ou mais pessoas. A maioria dos cursos à distância atuais mediados por computador enfatiza a interação homem/máquina, dando destaque às interfaces gráficas dos softwares e aos componentes de comunicação do hardware. Não que estes itens devam ser desconsiderados, pois têm grande importância neste sistema de ensino, mas a comunicação e interatividade interpessoal não deve ficar a parte deste processo.

Já o termo interatividade é recente e emerge juntamente com a ampliação da informática e das tecnologias da informação e comunicação (Schechtman, 2001). Segundo Outing (1998), softwares e ambientes interativos permitem que o usuário interaja com o conteúdo, já os que permitem interatividade, proporcionam também a comunicação e a troca de informações entre os usuários. A interatividade é a capacidade dos participantes receberem *feedback* específico como contribuição de algum outro membro do processo de comunicação (Romiszowski & Mason 1996).

Fator determinante para o alcance dos objetivos a que se propõe a modalidade EAD, a interatividade entre professor e aluno deve ser exaustivamente estudada, não só quanto aos aspectos relativos à engenharia de software e hardware, mas principalmente quanto às concepções educacionais, filosóficas e cognitivas que definem este relacionamento (Schechtman, 2001).

Assim como na comunicação interpessoal, a interatividade em EAD ocorre em diferentes níveis determinados tanto pela capacidade dos meios técnicos, quanto pelo ambiente pedagógico em que se situa e pelas necessidades dos usuários. A

interatividade em EAD ocorre quando as diferenças forem respeitadas, os meios forem adequados, o aluno se sinta seguro e consiga compartilhar informações.

Martin & Bramble (1996) apud Cornell & Martin (1997) afirmam que normalmente os alunos preferem a interação com o professor ao software, como secções de perguntas mais freqüentes e respostas, apresentações ou lista de discussão.

Keller & Burkman (1993) apontam como estratégia para manter o aluno motivado, prover oportunidades para o aluno interagir com o professor, com outros alunos e com o material didático. Outra estratégia importante é o *feedback*, que o aluno deve receber em tempo oportuno com confirmação de suas intervenções corretas e correção das incorretas.

O e-mail é apontado como simples e eficiente método de interação e tem sido a principal ferramenta para manutenção de comunicação entre professor e aluno em muitos ambientes (Brooks, 1997). Sendo do tipo de comunicação assíncrona, o e-mail oferece como benefício maior tempo para pensar, sendo possível assim produzir respostas mais estruturadas e completas (Romiszowski & Mason, 1996).

Trabalhando com uma classe tradicional, Brooks (1997) exercitou a comparação do assédio que os alunos faziam através de e-mail e pessoalmente ao professor. Concluiu que poucos alunos visitam o escritório do professor na escola fora do horário de aula, porém, um grande número de contatos acontece através de e-mail.

4.3. Considerações Finais

Este trabalho propõe a interatividade síncrona do aluno com o software, tendo a participação assíncrona do mentor. A interatividade aluno/software é aceitável até certo ponto, porque este último por melhor que seja, tem limitações quanto às formas de transmitir o conteúdo e de reconhecer as dúvidas do aluno. É neste ponto que a participação do mentor mostra-se importante como moderador do processo, já que a perceptividade humana está muito longe de ser alcançada por qualquer ferramenta lógica ou mecânica.

5. Educação à Distância na Internet

Com a atual associação entre a capacidade de processamento de informações dos computadores e a facilidade de transmissão de dados através da Internet, está sendo criado um novo paradigma no processo educacional à distância.

Alguns fatores têm contribuído para que a Internet esteja sendo considerada o principal ambiente de aplicação para EAD. A utilização de microcomputadores mais poderosos e mais baratos, largura de banda disponível e ferramentas modernas principalmente para Web, têm permitido o uso de recursos da Internet em EAD.

Nos últimos anos tem ocorrido um aumento constante no volume de informações mantida e acessada eletronicamente. O número de periódicos eletrônicos tem aumentado, permitindo acessos *on line* às suas bases de dados, disponibilizando uma fartura de informações através da Web. Esses recursos atendem as necessidades de alunos, especialmente aqueles em áreas remotas que não possuem acesso às coleções em bibliotecas (Orr & Apletton, 1997).

A Web é apontada como principal responsável pelo crescimento do ensino via Internet. É um ambiente agradável, interativo, de fácil navegação, e que permite a utilização de som e imagem, que são considerados fatores importantes no processo ensino-aprendizagem.

A Web permite criar salas de aula virtuais, com alunos acessando-as como e quando melhor lhes convier. Isto é muito útil para os que estão impossibilitados de ir a uma classe física ou freqüentar as aulas em horários específicos e regulares.

Nas Web sites, o professor tem condições de expor claramente os conteúdos, propor e avaliar exercícios, e montar ligações (*links*) com tópicos afins no mesmo site ou em outros sites. O aluno então, tem à sua disposição, uma grande quantidade de informações para pesquisar e estudar. Não deve ser desconsiderada a utilização de listas de discussão, que são utilizadas para debates e interação entre os alunos. É aconselhável também, uma seção de perguntas e respostas, para esclarecer as dúvidas mais freqüentes dos alunos.

Em resumo, a Web apresenta algumas características educacionais como o fácil acesso a grande volume de informações, a possibilidade de manter informações e documentos sempre atualizados e a elaboração de cursos com avançados recursos de multimídia, hipertextos, sistemas de conferências, listas de perguntas e respostas mais freqüentes, dentre outras.

É interessante porém, que a tecnologia utilizada em EAD seja o mais transparente possível para o aluno, proporcionando a ele concentrar-se apenas no aprendizado, evitando assim discussão de pontos relativos à tecnologia empregada.

Um ambiente para cursos à distância via Internet não deve apenas funcionar, deve também ser didático, agradável e convidativo, despertando no aluno interesse pelo curso. Esse deve ser um dos objetivos do educador que projeta programas de EAD via Internet, já que esta questão está diretamente ligada a motivação, aspecto muito difícil de ser trabalhado mesmo na educação presencial (Leite, 2002).

Outros fatores importantes a serem considerados são: a interatividade, que é palavra-chave para qualquer processo educativo, não só com professor e alunos entre

si, mas também com o conteúdo do curso e com a tecnologia e recursos de ensino; as preocupações com as interfaces no aspecto da comunicação visual; integração e conhecimento com o meio e a tecnologia a ser utilizada no processo educativo.

É importante que seja desenvolvido um sistema de monitoração de acessos e movimentos dos alunos pelas páginas. Estas informações irão trazer subsídios para a melhoria do curso e para o auxílio aos alunos. Observa-se que este sistema deve ser transparente ao aluno proporcionando que o mesmo trabalhe com desenvoltura e sem a preocupação de estar sendo vigiado, tornando assim espontâneos todos os dados obtidos (Naves, 1998).

A possibilidade de uma maior interação entre professor e aluno criou condições para a realização de ensino-aprendizagem em EAD com níveis próximos aos obtidos através das experiências presenciais (Naves, 1998).

Segundo Matos (2002) a internet proporciona grandes vantagens para EAD, entre as quais é possível destacar:

a) A centralização do ensino no aluno que instiga o inter-relacionamento e a colaboração entre os alunos, proporcionando métodos menos rígidos de aprendizagem, tornando-a mais interativa e atrativa.
b) A flexibilidade no horário e local de ensino que permite o desenvolvimento de outras atividades paralelas pelo aluno, como trabalhar e nas horas vagas fazer o curso sem problemas de incompatibilidades de horário ou local.
c) Ferramentas para o aluno bastante conhecidas, onde computadores, softwares para comunicação via internet e outras normalmente utilizadas, também chamadas de ferramentas de apoio, são de fáceis manuseio e bastante conhecidas, ou quando não, são de fáceis aprendizados.

d) Rapidez no desenvolvimento dos ambientes virtuais, já que atualmente ferramentas de autoria e composição permitem projetar, desenvolver e disponibilizar um site na Internet com rapidez e até certa facilidade.

e) Reaproveitamento dos recursos, possibilitado pela grande quantidade de informações disponíveis na Internet associadas a poderosas ferramentas de busca, permitindo que o estudo de um determinado assunto possa ser reduzido a uma série de *links* para materiais já existentes.

f) O formato eletrônico dos documentos facilita as alterações de seus conteúdos economizando tempo e recursos humanos.

g) Um bom funcionamento do ambiente virtual é possível em função da padronização de tecnologias na Internet, que minimizam as incompatibilidades de hardware, softwares básicos e sistemas operacionais.

Reafirmando estas vantagens, Naves (1998) destaca opiniões relevantes de alunos após participarem de um curso à distância através da Internet.

- A participação do aluno em um programa à distância pela Internet foi associada ao não afastamento do ambiente de trabalho e da família e à falta de oportunidades para quem vive em locais distantes.
- Facilidade de acesso, flexibilidade de horários, não deslocamento e fator econômico (redução de custos), foram citados como os grandes benefícios desta modalidade de treinamento.
- A velocidade da rede e a perda de comunicação foram citadas como os principais obstáculos encontrados pelos alunos durante o curso.
- Muitos participantes acreditam terem aprendido mais por não estarem próximos ao instrutor e terem se esforçado mais na obtenção das respostas.

- A maioria dos alunos sentiu-se mais exigida num treinamento desta natureza.

Sabariz & Barreto (2002) também destacam manifestações de alunos após participarem de um curso de Educação Virtual baseada na Internet.

- O ritmo de cada aluno é respeitado, alunos com mais facilidade podem prosseguir mais rapidamente e até ajudar os outros.
- Existe uma constante interação entre aluno/professor e aluno/aluno, havendo no site do curso várias formas desta interação, tais como e-mail, formulários, chat e videoconferência.
- Além do site do curso, há inúmeros outros locais para serem consultados.
- Qualquer nova descoberta é facilmente incluída nas aulas, ao contrário dos livros didáticos, que estão sempre dependendo de novas edições atualizadas.
- As aulas são mais dinâmicas, não existindo aquela monotonia do quadro negro, giz e transparências estáticas, aparecem então vídeos, gráficos e outros que são muito mais atraentes.
- O aluno pode ter acesso às aulas de modo assíncrono, ou seja, a qualquer hora, pois elas estão disponíveis 24 horas por dia através do servidor.
- Não existe aula perdida, pois todas elas estão sempre disponíveis.

6. Integrando Web com Banco de Dados

Uma das principais atividades na informática é a recuperação e processamento de informações armazenadas em banco de dados. Podemos citar como exemplo, transações bancárias, e consultas a acervos de bibliotecas.

No ambiente Web, também é possível proporcionar serviços dinâmicos como estes. Para tanto, utilizam-se *scripts* Web que são programas que ficam armazenados e são executados no servidor sempre que solicitados pelo cliente Web.

6.1. Arquitetura da Aplicação de Banco de Dados

A montagem de uma aplicação com banco de dados em ambiente Web envolve a integração de vários elementos, conforme o exemplo demostrado na figura abaixo.

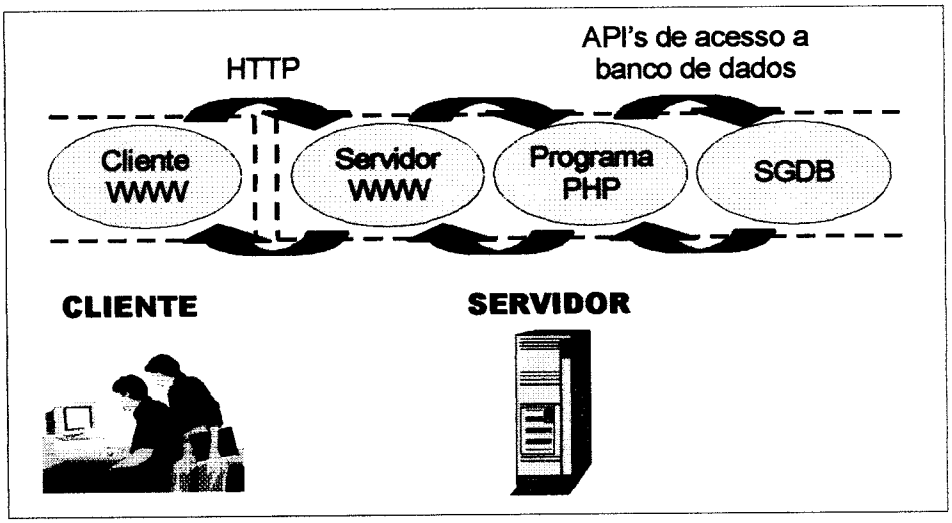


Figura 7 – Aplicação Web com uso de banco de dados.

Neste exemplo, o cliente Web solicita o URL de um programa que realiza acesso a um banco de dados. O servidor Web que recebeu a solicitação inicia a execução do programa PHP, que através de rotinas específicas acessa o servidor de banco de dados. O servidor de banco de dados devolve as informações solicitadas ao programa PHP que por intermédio do servidor Web encaminha-as ao cliente.

6.2. Scripts Web e a Interface CGI

Atualmente a maior parte de documentos consultados através de um URL é estática, ou seja, documentos já prontos e armazenados no servidor Web, porém existem situações em que não é apropriada a utilização destes tipos de documentos. Assim é necessário utilizar documentos que sejam gerados dinamicamente em tempo de execução, ou seja, *on line*. O TABIPRO enquadra-se nesta segunda forma de geração de documentos Web.

Um *script* Web pode ser escrito a partir de linguagens de programação que geram códigos compilados como os escritos em C++ ou Java, ou códigos interpretados como os escritos em PHP ou Perl.

Um *script* Web é um programa que é executado pelo servidor Web em resposta a um pedido de um cliente Web. Seu objetivo principal é gerar informação dinâmica e em tempo real para o cliente Web que o solicitou. Ele deve retornar os dados num formato que o cliente consiga interpretar, como um documento HTML por exemplo. Através dele também é possível executar outros programas ou realizar conexões com outros servidores.

As operações descritas abaixo são realizadas pelo servidor Web sempre que um programa *script* é solicitado para ser executado (Sá, 2000):

a) Avaliar se o programa *script* pode ser executado.

b) Iniciar a execução do programa *script,* passando os dados de entrada do cliente Web para o programa *script.*

c) Receber os dados retornados pelo programa *script* e enviá-los ao cliente Web.

d) Enviar uma mensagem ao cliente Web se a execução do programa *script* não for bem sucedida.

e) Encerrar a conexão apropriadamente, assim que o programa *script* finalizar sua execução.

CGI *(Common Gateway Interface)* é a especificação de uma interface que permite que servidores Web tenham acesso a funcionalidades oferecidas por programas executando no ambiente da máquina servidora. Através de programas conectados a essa interface é possível por exemplo conectar uma base de dados a Web ou gerar dinamicamente o conteúdo de uma página HTML.

A interface CGI é um conjunto de regras criadas para padronizar a execução de programas *scripts* e é utilizada pela maioria dos servidores Web existentes. Esta interface descreve como os programas *scripts* serão inicializados e como os dados serão passados entre o *script* e o servidor Web.

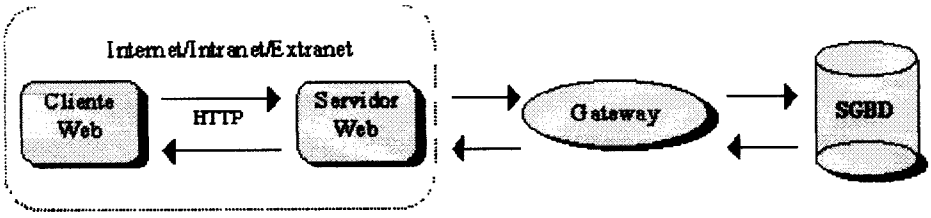


Figura 8 – A interface CGI.

O servidor Web reconhece uma requisição CGI quando o URL especificado na solicitação identifica um arquivo executável (programa ou *script*) localizado em um diretório específico dentro do espaço Web de recursos disponibilizados aos clientes.

A programação CGI representa a criação de programas que recebem e traduzem dados enviados via Internet para seu servidor na Web. A mesma envolve o uso desses dados traduzidos e também a compreensão de como se pode enviar cabeçalhos válidos de resposta de HTTP e tags da HTML ao seu cliente da Web. Ela é denominada *gateway* porque com uma freqüência bastante grande, os programas precisam atuar como interfaces ou elementos de transição (*gateways*) entre outras aplicações maiores (Hermann, 1997).

Por exemplo, podemos criar um programa que traduza e formate os dados enviados e recebidos de aplicações como catálogos *on line*. Em seguida, esses dados traduzidos serão passados para um servidor de banco de dados que executa as operações necessárias em seus dados e retoma os resultados ao programa CGI. A seguir o programa CGI pode reformatar os dados retornados de acordo com a necessidade da Internet e retorná-los ao cliente que consulta o catálogo *on line*. Neste caso o programa CGI atua como um *gateway* entre o catálogo HTML, os cabeçalhos de solicitação/resposta de HTTP e o servidor de banco de dados.

6.3. HTML, HTTP e os Programas CGI

O programa CGI, a HTML e o HTTP precisam operar de forma coordenada a fim de que a sua aplicação *on line* para a Internet funcione de forma apropriada. O código em HTML define a forma como o usuário visualiza a interface do programa, sendo também responsável pela coleta de dados inseridos pelo usuário. O HTTP representa o mecanismo de transporte para o envio de dados entre o programa CGI e o usuário, ele traduz e envia as informações entre o cliente da Web e o programa CGI. O programa CGI é responsável por entender as instruções do HTTP e as solicitações do usuário. Ele recebe as solicitações feitas pelo usuário e retorna respostas válidas e úteis ao mesmo, que está dando cliques na sua página HTML da Web.

A programação CGI constitui a criação de aplicações que atuam como programas de interface ou de *gateway* entre o navegador cliente, o servidor da Web e uma aplicação tradicional de programação.

6.4. Programas Executáveis CGI

É uma arquitetura composta por um ou mais programas CGI, geralmente implementados usando-se alguma linguagem de programação hospedeira do DBMS. Para usar a interface CGI implementada pelos servidores Web é necessário uma linguagem que possa ler a entrada padrão, obter as variáveis de ambiente e escrever na saída padrão. Como a grande maioria dos DBMSs existente incorpora alguma linguagem de programação com estas características, praticamente todos podem se

comunicar com a Web por meio de programas executáveis CGI sem grandes custos adicionais (Lima, 1997).

Esta arquitetura funciona da seguinte forma: O cliente Web solicita ao servidor Web a execução de uma aplicação externa, que neste caso, é um programa executável na plataforma do servidor Web. O servidor Web dispara um processo para execução do programa através da interface padrão CGI, enviando os dados recebidos do cliente Web de acordo com a implementação do servidor Web para a interface CGI. O programa recebe os dados passados pelo servidor Web via interface CGI, decodifica-os, formula o comando SQL e abre uma conexão com o Banco de Dados para sua execução. O DBMS retorna os dados e a conexão é finalizada. O programa executável trata os dados recebidos e os repassa ao servidor Web via interface CGI, num formato que o cliente Web entenda, como por exemplo, no formato HTML. O processo iniciado pelo servidor Web para execução do programa é finalizado neste momento. O Servidor Web retorna os dados repassados pelo programa ao cliente Web (Sá, 2000).

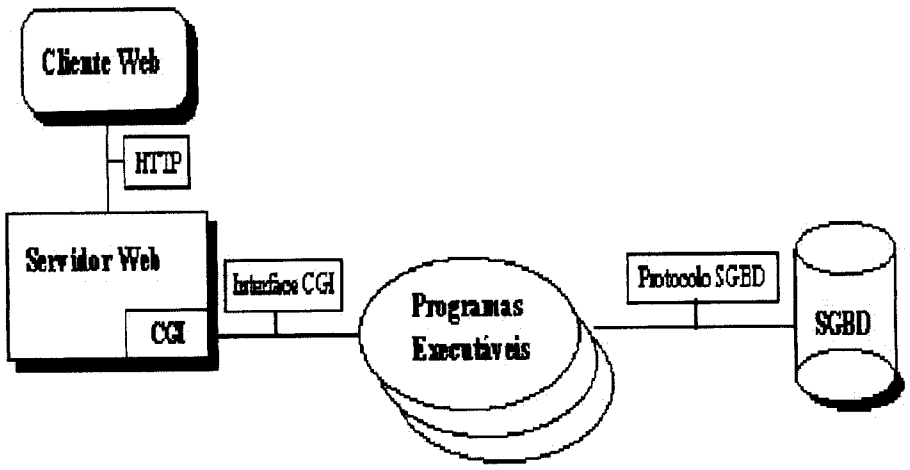


Figura 9 - Arquitetura de Programas Executáveis CGI. Fonte: Sá (2000).

7. Os Recursos Utilizados

7.1. Scripts

7.1.1. Client-Side Scripts

São responsáveis pelas ações executadas no *browser*, sem contato com o servidor. São exemplos, os *scripts* que fazem validações de formulários sem utilizar processamento do servidor e com isso, sem provocar tráfego na rede.

7.1.2. Server-Side Scripts

Os *scripts server-side* são responsáveis pela criação de páginas em tempo real. No momento em que o usuário solicita um URL, o servidor apresentará ao *browser* um código HTML dinâmico, isto é muito útil para construções de aplicações baseadas em informações *on line*.

Exemplos de *server-side scripts* são os sistemas de busca como www.google.com, www.yahoo.com, www.cade.com.br e outros. Neles, após digitar a palavra chave e solicitar a pesquisa, o resultado de um processamento *on line* é exibido ao usuário. Seria inviável manter um arquivo para cada consulta a ser

realizada. O que existe é um modelo da página de resposta, que é mesclado com os dados no momento em que a página é requisitada.

7.2. A Linguagem de Programação PHP

PHP é uma linguagem de programação interpretada que é especialmente interessante no desenvolvimento de aplicações e sites para a Web, podendo ser embutida a códigos HTML (Bakken et al, 2000).

É principalmente utilizada para produzir *server-side scripts* que cria sites dinâmicos, possibilitando interação com o usuário através de formulários, parâmetros de URL e *links*. Sites dinâmicos são aqueles que retornam para o cliente uma página criada em tempo real.

Uma vantagem do PHP é a execução de seu código no servidor, sendo enviado para o cliente apenas HTML puro. Desta maneira é possível interagir com bancos de dados e aplicações existentes no servidor, com a vantagem de não expor o código fonte para o cliente. Isso pode ser útil quando o programa está lidando com informações confidenciais (Medeiros, 2002).

O PHP pode ser utilizado na maioria dos sistemas operacionais, incluindo Linux, várias variantes Unix, Microsoft Windows, Mac OS X, RISC OS, além de outros. O PHP também é suportado pela maioria dos servidores Web atuais, incluindo Apache, Microsoft Internet Information Server, Personal Web Server, Netscape and iPlanet Servers, Oreilly Website Pro Server, Caudium, Xitami, OmniHTTPd, dentre outros (Bakken et al, 2002). Com o PHP, portanto, se tem a liberdade para escolher o sistema operacional e o servidor Web.

Também é possível escolher entre utilizar programação estrutural ou programação orientada a objetos, ou ainda uma mistura delas. Mesmo não desenvolvendo nenhum recurso padrão de OOP (Object Oriented Programming, Programação Orientada a Objetos) na versão atual do PHP, muitas bibliotecas de código e grandes aplicações foram escritos somente utilizando OOP (Bakken et al, 2002b).

Talvez a mais forte e mais significativa característica do PHP é seu suporte a uma ampla variedade de banco de dados, o que é de extrema importância para o desenvolvimento de aplicações onde é necessário ler e armazenar informações.

Atualmente os seguintes bancos de dados são suportados pelo PHP: Adabas, dBase, Direct MS-SQL, Empress, FilePro (read-only), FrontBase, Hyperwave, IBM DB2, Informix, Ingres, InterBase, mSQL, MySQL, ODBC, Oracle, Ovrimos, PostgreSQL, Solid, Sybase, Unix dbm e Velocis (Bakken et al, 2002).

Adicionalmente, o PHP suporta ODBC (Open Database Connection, ou Padrão Aberto de Conexão com Bancos de Dados), permitindo que você utilize qualquer outro banco de dados que suporte esse padrão.

7.3. Banco de Dados

Um Banco de Dados é um conjunto de dados estruturado e organizado com informações a respeito de um tópico específico (Forcier, 1999). Consiste de um ou mais arquivos, cada um contendo informações no formato de um conjunto de registros. Cada registro é dividido em campos que descrevem a classe ou tipo de

informação contida neles. Um mesmo tipo de informação é armazenado em cada campo de todos os registros (Jonassen & Reeves, 1996).

O Gerenciador de Banco de Dados (DBMS – Database Management System) deve permitir três atividades básicas: armazenar dados, reorganizá-los de várias formas, e recuperá-los quando e na forma que for necessário (Forcier, 1999). As mais importantes funções do DBMS fazem parte de um conjunto de ferramentas que possibilitam incluir, alterar, excluir e classificar informações, e a principal delas que é fornecer resposta às pesquisas realizadas nas informações contidas no banco de dados.

Neste projeto, o servidor de banco de dados tem o propósito de armazenar protótipos de tabelas criadas no módulo Mentor e acessadas pelo módulo Aluno que montará as tabelas em tempo real. Cada protótipo de tabela contém nome, título, número de linhas e colunas, percentual de tolerância a erro, e-mail do mentor, e células que podem ter mensagens e fórmulas matemáticas. O protótipo é utilizado como base para a montagem dinâmica e *on line* da tabela no *browser* do aluno.

O MySQL foi o DBMS escolhido para este projeto, já que possui características adequadas ao mesmo.

7.3.1. MySQL

O MySQL é um servidor de banco de dados relacional, ágil, multiusuário, multitarefa que trabalha com SQL, uma das linguagens de manipulação de dados mais popularizada do mundo (MySQL AB, 2002).

SQL é uma linguagem simples com rotinas para gravar, alterar e recuperar informações com segurança e rapidez. Ela foi desenvolvida pelo departamento de

pesquisas da IBM como forma de interface para o sistema de banco de dados relacionais System R, no início dos anos 70. Em 1996, o American National Institute (ANSI) publicou um padrão SQL. A SQL estabeleceu-se como linguagem padrão de banco de dados relacional (Cox Jr, 2000).

O MySQL foi originalmente desenvolvido pela empresa sueca TCX, que necessitava de um servidor de banco de dados que operasse rapidamente com grandes escalas de dados, sem exigir caríssimas plataformas de hardware. Servidor arrojado, sabe-se hoje de uma aplicação que usa o MySQL contendo 50 milhões de registros, e de outra com 60.000 tabelas e cerca de 5.000.000.000 de linhas (MySQL AB, 2002b).

Existe estimativa de haver aproximadamente 4.000.000 de cópias ativas do Mysql pelo mundo, dentre as quais algumas usadas em aplicações críticas por grandes corporações como Yahoo!, Cisco, NASA, Lucent Technologies, Motorola, Google, Silicon Graphics, HP, Xerox and Sony Pictures (MySQL AB, 2002a).

7.3.1.1. Características do MySQL

Segundo o Manual do MySQL (MySQL AB, 2002b) suas principais características são:

- Velocidade, robustez e fácil utilização.
- Suporta diferentes plataformas: Linux, FreeBSD, Unix, Win32, dentre outras.
- Suporta API's de várias linguagens: PHP, Perl, C++,Java, Pynthon, dentre outras.
- Suporte a múltiplos processadores.

- Um sofisticado sistema de senhas criptografadas flexível e seguro.
- Suporte a ODBC.
- Código fonte escrito em C e C++ e testado com uma variedade de diferentes compiladores.
- O cliente se conecta no MySQL através de conexões TCP/IP.

7.4. Considerações Finais

Podendo ser utilizado nos principais sistemas operacionais e servidores Web, a linguagem de programação PHP proporciona de forma rápida e simples o desenvolvimento de *scripts* que atendam o propósito do projeto, permitindo interação dentre outros, com o banco de dados MySQL, que foi escolhido pelas suas características técnicas. Estas duas ferramentas tornam o TABIPRO independente em relação às plataformas operacionais do mentor, do aluno e do próprio curso em que é inserido, potencializando sua utilização para diversos tipos de aplicações.

8. O Programa Educ@r

O Programa Educ@r (http://educar.sc.usp.br) do Centro de Divulgação Científica e Cultural da USP (CDCC), é compreendido por subprogramas de ensino interativo baseados na Web e tem como objetivo aplicar as tecnologias da informática na EAD, na tentativa de contribuir, através delas, com a formação de professores e a melhoria da aprendizagem dos alunos.

Também são objetivos do programa Educ@r:

- O uso do conhecimento em ciências como meio auxiliar no processo de ensino e aprendizagem;
- A interligação do sistema de ensino público em uma rede de apoio e interatividade;
- Geração de pesquisa e desenvolvimento em informática e em EAD;
- Avaliação dos impactos e das transformações decorrentes da presença do ensino de ciências nas escolas públicas e nos centros de divulgação;
- Produção e a avaliação de software para EAD.

8.1. Projetos do Programa Educ@r

A seguir serão apresentados segundo o CDCC (2002) subprogramas já desenvolvidos e em desenvolvimento com uma breve descrição de cada um:

- ✓ Ciências: Oferece subsídios para as aulas de ciências do ensino fundamental.
- ✓ Educação Ambiental: Através do estudo de uma bacia hidrográfica permite conhecer a cidade e relacionar os conhecimentos adquiridos com as possíveis mudanças de atitudes em relação ao ambiente.
- ✓ Matemática: Curso à distância que oferece ao professor de primeira à quarta série do ensino fundamental a oportunidade de atualizar os seus conhecimentos.
- ✓ Mecânica Gráfica: Permite realizar, experimentalmente, movimentos em Física e simular esses movimentos em uma linguagem de computador: a linguagem Logo. Os mesmos movimentos são filmados com uma câmera de vídeo e após fazer a captura das imagens para o computador, faz-se a análise quantitativa destes, utilizando o software SAM (Sistema para Análise de Movimentos).
- ✓ Melhoria do Ensino de Ciências na América Latina: Apresenta projetos desenvolvidos em parceria entre a Casa de Las Ciencias, da Universidad Nacional de Cordoba (UNC), Argentina, o Instituto de Estudos Avançados de São Carlos (IEASC) e o CDCC, ambos da USP São Carlos.
- ✓ Ótica: Curso completo com fundamentação teórica e experimentos para alunos e professores do ensino médio.
- ✓ Química: Oferece subsídios para aulas de química no ensino fundamental e médio.

8.2. Histórico

O Programa EDUC@R teve como origem o Programa Experimentoteca Pública Nacional, com o apoio da Vitae de 1992 a 1994, tendo como objetivos a produção e difusão de kits de material didático. Este trabalho resultou na constituição de centros encarregados de promover a circulação dos materiais e de realizar programas de treinamento de professores. Com o objetivo de capacitar esses centros para a constituição de pólos permanentes de divulgação científica e de apoio ao aperfeiçoamento do ensino de Ciências nos ensinos fundamental e médio, foi elaborado um novo projeto sob coordenação do CDCC, prevendo a expansão da rede de centros, a interligação eletrônica desses centros através da Internet, a produção e a realização de programas de EAD para professores e alunos vinculados aos centros, a produção de textos científicos de referência e a implantação de um processo de assessoria técnico-pedagógica direta (CDCC, 2002).

De 1996 a 1998 o Programa Educ@r fez parte do Programa Educação à Distância em Ciência e Tecnologia (EDUCADI), coordenado pela Universidade Federal do Rio Grande do Sul (www.ufrgs.br/ufrgs).

8.3. O Módulo de Mecânica Gráfica

As idéias básicas para a proposta de Mecânica Gráfica surgiram juntamente com as primeiras aplicações didáticas do computador, no começo da década de 70. Tinha sido criado a linguagem BASIC com o objetivo de tornar a programação de computadores acessível a leigos. Huggins (1979) desenvolveu uma proposta de analisar movimentos gravitacionais registrados de forma estroboscópica com auxílio de programas feitos na citada linguagem. Posteriormente Seymour Papert criou a linguagem LOGO com propósitos educacionais e inventou-se em São Carlos - SP o Puck de Hessel (figura 10) (Hessel & Schiel, 1998) que permite o registro de movimentos bidimensionais a um custo bem inferior à fotografia estroboscópica (CDCC, 2002a).

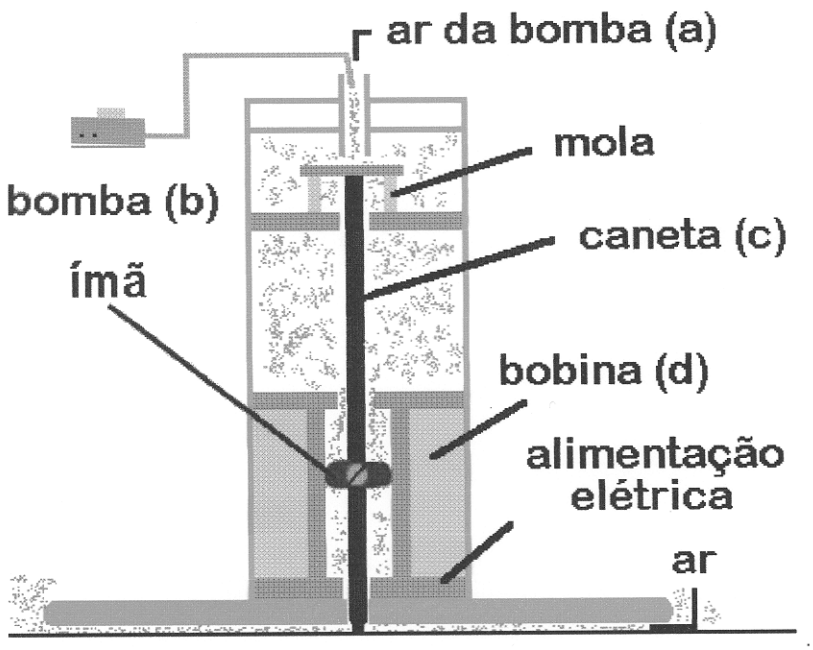


Figura 10 – Puck de Hessel.

Tão logo desenvolveu as primeiras idéias da mecânica, Newton percebeu que a análise do movimento não uniforme poderia levar a cálculos repetidos num número extremamente grande de vezes, relativos a intervalos de tempo muito pequenos. Como isto seria impossível, resolveu o problema de forma analítica, criando para isto um ramo novo da matemática, o cálculo diferencial e integral. Hoje esta impossibilidade está sendo removida com a velocidade de cálculo possível nos computadores e podemos por exemplo, prever a trajetória de uma astronave sem conhecer sua equação analítica. O aluno do ensino médio que participa deste programa, calcula órbitas gravitacionais sem que venha a saber o que é uma equação diferencial (CDCC, 2002a).

O Mecânica Gráfica, tem a proposta de complementar o ensinamento de Física para os cursos do ensino médio e foi elaborado de forma que o aluno execute experimentos quantitativos, tendo apoio de um mentor distante. É baseado em trabalho experimental conduzido por estudantes, com a assistência do professor presente na sala de aula ou no laboratório e do mentor distante alocado no CDCC em São Carlos.

O professor ministra aula com tópicos relacionados à Física e supervisiona o trabalho experimental dos alunos. Os alunos são livres para gerar seus próprios resultados experimentais, incorporando a eles correções e respostas recebidas da equipe de suporte do CDCC e de pesquisas a textos e materiais disponíveis na Web site do curso (Schiel et al, 2002).

No curso de Mecânica Gráfica, o aluno adquire familiaridade com a mecânica, programando seu computador para que este reobtenha resultados alcançados experimentalmente. A facilidade de comunicação via Web é usada tanto para que o aluno possa rapidamente conferir a correção de seus cálculos, quanto para

obter orientação do mentor. Em seu trabalho, o aluno realiza e analisa experiências e programas feitos no computador. Isto ocorre nas seguintes etapas (CDCC, 2002a):

1. Familiarização com a linguagem LOGO, aproveitando a motivação que o aspecto lúdico desta linguagem traz.

2. Realização de experimentos quantitativos com o Puck e obtenção de dados sobre a trajetória deste. De posse destes dados o aluno calcula velocidade e aceleração desse corpo. Valores obtidos nestes experimentos juntamente com os calculados são inseridos em tabelas interativas, que permitem ao aluno verificar automaticamente se seus cálculos estão corretos. Imprevistos, dúvidas e problemas conceituais são respondidos pelo mentor. Todos os cálculos feitos pelo aluno são registrados pela direção do programa.

3. No final de cada capítulo, o aluno faz um programa em LOGO, simulando na tela do computador o experimento realizado, tentando obter os mesmos resultados experimentais.

4. Na fase final, o aluno produz programas relativos a situações não realizáveis em laboratório: o movimento central gravitacional.

Na etapa 2 citada anteriormente, um programa realiza cálculos com os dados digitados e os envia de volta ao aluno mostrando no caso de erros onde estes ocorreram. Envia-os também ao mentor no CDCC, para que este analise os resultados e corrija os erros não só matemáticos, mas também os erros de medições realizadas pelo aluno durante a fase experimental do processo. Após rever todos os resultados, o mentor através de e-mail, retorna-os para o aluno juntamente com

comentários. Assim como as tabelas, os programas escritos em LOGO também são enviados ao mentor no CDCC, que os avalia e comenta (Schiel et al, 2002).

8.3.1. As Tabelas do Módulo Mecânica Gráfica

No módulo de Mecânica Gráfica, existem tabelas interativas, que devem ser preenchidas por alunos que estão fazendo o curso.

Este projeto está criando a terceira geração destas tabelas, sendo que na primeira geração elas foram desenvolvidas por Sérgio Gasques Rodrigues em linguagem C, que acarretava complexa manutenção no código fonte, necessitando para isso um programador experiente e com ótimos conhecimentos desta linguagem.

Na segunda geração Peter de Pádua Krauss utilizou a linguagem Perl no desenvolvimento dos códigos fontes. Houve evolução na forma de apresentação e comunicação entre o aluno e o professor, porém, um código fonte muito extenso e sem documentação adequada dificultava qualquer manutenção necessária, precisando também de um programador experiente para realizá-la.

A criação da terceira geração de tabelas através deste projeto, utiliza um método que permite ao mentor controlar a formatação do conteúdo das tabelas, sem no entanto conhecer a linguagem de programação que foi utilizada para seu desenvolvimento, o PHP. O código fonte do TABIPRO tem tamanho reduzido e quando necessário será de fácil manutenção, precisando para isso de um programador PHP.

A grande vantagem do TABIPRO em relação às gerações anteriores, é que quando uma tabela for criada ou alterada, não haverá mais a mesma necessidade de

manutenção no código fonte, já que os dados da tabela não ficam nele embutidos. Todas as informações necessárias para a montagem dinâmica e em tempo real da tabela ficam armazenadas no banco de dados e estão passíveis de alterações através do módulo Mentor.

9. Descrição do Problema

Nas páginas do Programa Educ@r, existem tabelas que são montadas em tempo real, sempre que o *browser* do aluno que está fazendo o curso solicita uma das páginas em questão. As tabelas são montadas dinamicamente por *server-side scripts*, que contém todos os parâmetros necessários. Para cada tabela do site, uma rotina diferente do *script* é acionada, sendo que cada rotina contém os parâmetros da respectiva tabela.

Dois pontos negativos foram levantados neste método:

1. O *script* é muito grande já que ele trabalha cada tabela de forma individual. Inúmeras rotinas montam as tabelas e ficam encarregadas de processamentos inerentes a elas. Pelo tamanho e forma como o *script* foi escrito, a sua manutenção é trabalhosa e muitas vezes dificultada, tendo que ser realizada em alguns casos, em todas as rotinas para que tenha efeito em todas as tabelas.

2. O professor mentor responsável pelo curso, tem dificuldade de realizar manutenções nos parâmetros das tabelas, uma vez que estes estão inseridos no código fonte do *script*, que deve sofrer manutenção por pessoa especializada na linguagem em que foi escrito, o Perl.

Em função dos problemas descritos, foi proposto o desenvolvimento de um projeto que permita ao mentor maior controle sobre a montagem e conteúdo das tabelas, sem que o mesmo possua conhecimentos em linguagem de programação necessária para a manutenção do código fonte dos *scripts*. Com este propósito foi desenvolvida a metodologia que resultou no software chamado Tabelas Interativas Programáveis (TABIPRO), que ataca o cerne dos dois problemas levantados.

10. Tabelas Interativas Programáveis

Após executar experimentos quantitativos, o aluno tem a possibilidade de avaliar seus conhecimentos utilizando os dados neles obtidos.

Ter uma ferramenta que permita ao aluno verificar rapidamente se os dados obtidos nos experimentos são consistentes é um incentivo ao mesmo para realizar novos experimentos, já que, as maiores queixas do aluno são em relação ao tempo despendido na realização de cálculos e produção de relatórios, em comparação com o tempo gasto para a realização do experimento e da compreensão do mesmo.

O TABIPRO é um software que analisa quantitativamente os dados obtidos nos experimentos realizados pelo aluno, permite uma verificação rápida dos valores numéricos informados e ainda abre um canal de discussão entre o aluno e o mentor.

O TABIPRO é dividido em dois módulos. O primeiro é o módulo Mentor, onde é possível criar os protótipos de tabelas que serão utilizados pelo módulo Aluno. No módulo Aluno é que ocorre a montagem dinâmica e em tempo real da tabela que está sendo acessada pelo aluno através de seu *browser*.

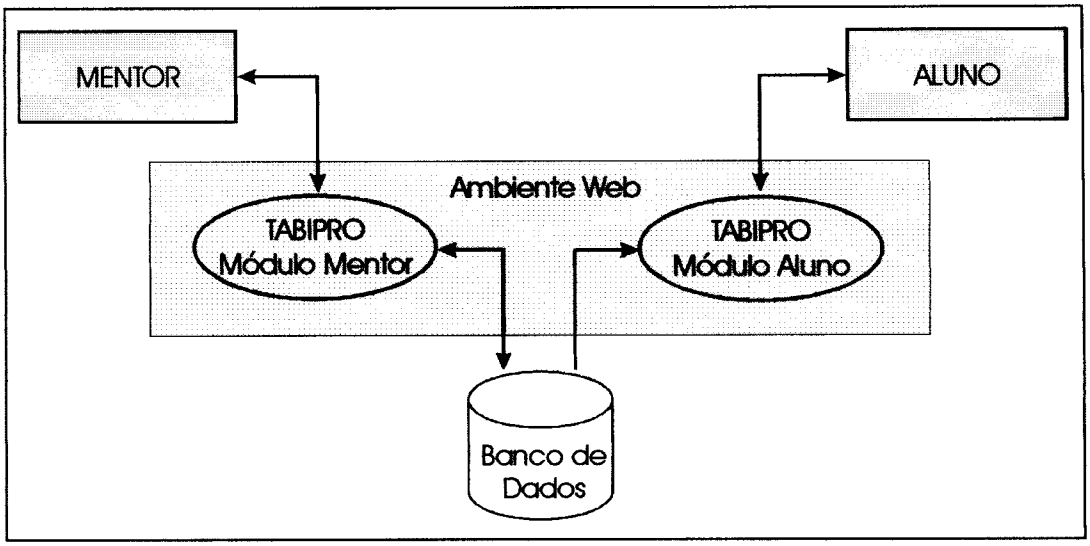


Figura 11 – Os módulos do TABIPRO.

10.1. A Interatividade das Tabelas Programáveis

A tecnologia deve ser utilizada pelo professor para que encoraje o aluno a buscar soluções aos problemas, ter pensamento crítico, realizar análises e fazer questionamentos (Forcier, 1999). Desta forma a interatividade proporcionada pelas tabelas do TABIPRO entre o aluno e o mentor, é fator importante como colaboração e suporte para que o aluno alcance os objetivos a ele propostos, induzindo-o a análises críticas e questionamentos, fornecendo *feedback* necessário através de comentários e orientações sobre suas observações e respostas informadas nas tabelas, contribuindo para seu crescimento no curso, como propõem Johnson & Johnson (1996).

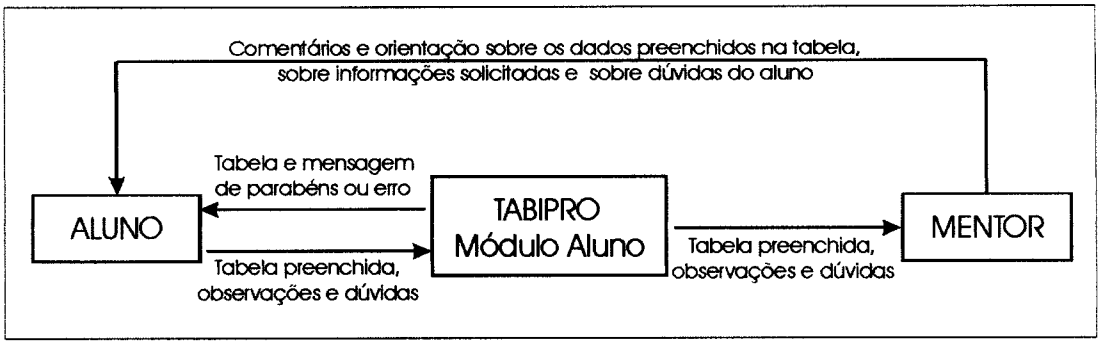


Figura 12 – Fluxo de comunicação no Módulo Aluno do TABIPRO.

Durante o preenchimento das tabelas propostas no TABIPRO, ocorre a geração de mensagens orientando o aluno, se este respondeu corretamente ou não os cálculos esperados em função da experiência executada no laboratório. A geração de mensagens dinâmicas explorada pelo TABIPRO, é um fator importante no processo de interação, como comenta Hill (1997).

A interatividade da tabela e a possibilidade de utilizar fórmulas matemáticas para a realização de cálculos com os dados obtidos nos experimentos, comparando o resultado destes cálculos com os valores preenchidos na tabela pelo aluno, informando a ele se está ou não no caminho certo, são pontos fortes do TABIPRO.

10.2. O Módulo Mentor

A utilização do TABIPRO no módulo de Mecânica Gráfica traz grande benefício no processo de elaboração das tabelas pelo mentor, já que possibilita a ele a manipulação do formato e conteúdo da tabela sem a necessidade de conhecer linguagens de programação. Aliado a isso, é possível ao mentor acompanhar o andamento das atividades dos alunos através de cópias das tabelas preenchidas que chegam a ele por e-mail emitido automaticamente pelo TABIPRO, abrindo um canal

de comunicação aluno/mentor, que pode ser acionado sempre que uma das partes desejar.

O mais importante na metodologia empregada pelo TABIPRO, é que o mentor não necessita possuir conhecimentos em linguagem de programação para montar o script e controlar a operação da tabela.

O módulo Mentor é um *server-side script* desenvolvido na linguagem de programação PHP e é encarregado de permitir ao mentor criar protótipos de tabelas, com todos os parâmetros necessários para que as mesmas sejam montadas dinamicamente em tempo real, sempre que o *browser* do aluno solicitar.

O protótipo da tabela fica armazenado em um banco de dados MySQL, sendo reconhecido por um nome definido pelo mentor. No protótipo estão todos os parâmetros da tabela, como o título da mesma, número de linhas e colunas, percentual de tolerância a erro, e-mail do mentor, títulos das colunas, mensagens e fórmulas matemáticas definidas pelo mentor.

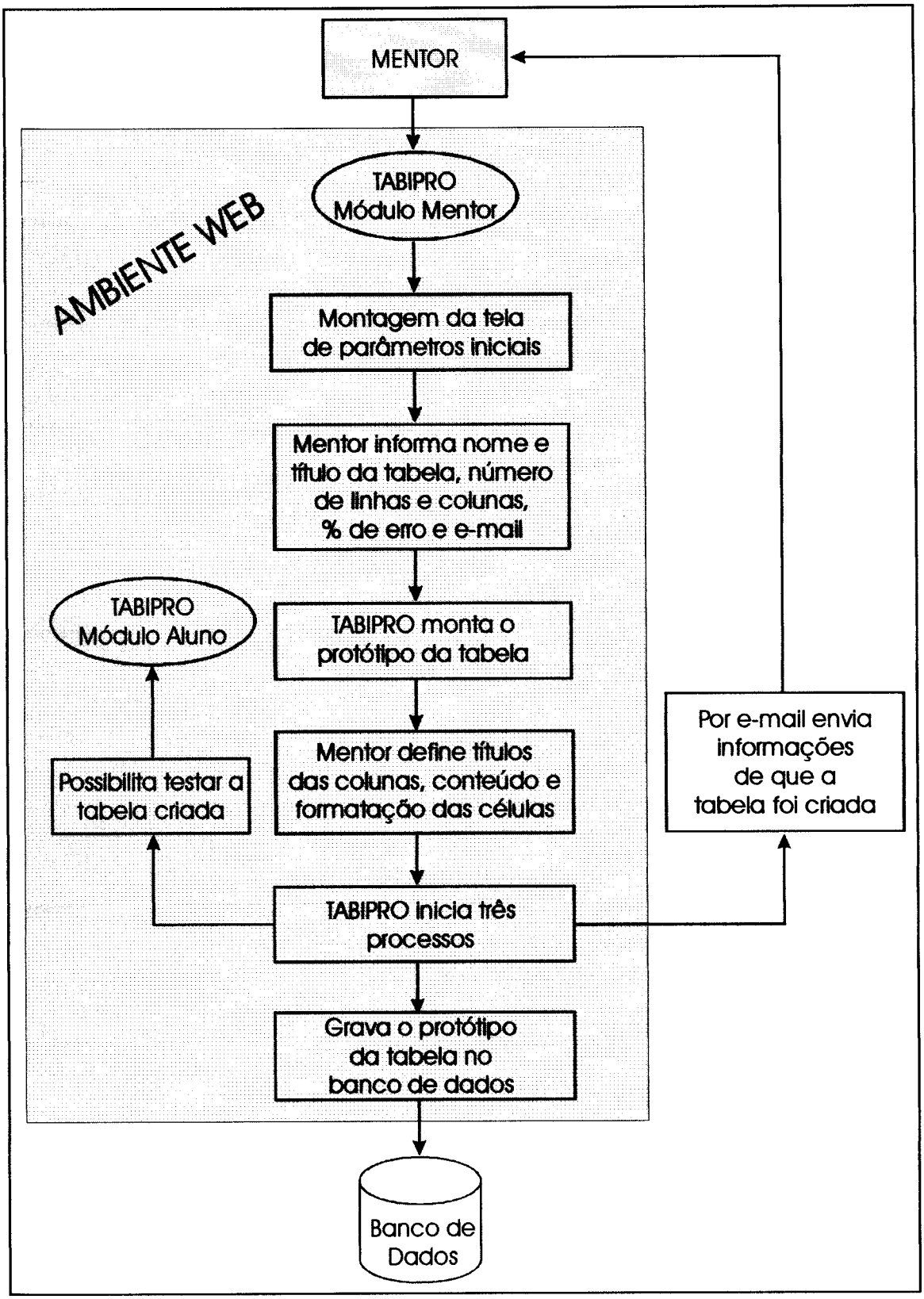


Figura 13 – Fluxo de tarefas do módulo Mentor.

10.3. O Módulo Aluno

Um outro *server-side script* completa o TABIPRO. Este é o módulo Aluno que também foi desenvolvido em PHP, e possui uma única rotina para montar dinamicamente e em tempo real, tantas tabelas quantas forem necessárias, além de realizar o processamento destas tabelas. Desta forma este *script* tem tamanho reduzido e é de fácil manutenção opondo-se aos problemas ocorridos nas gerações anteriores das tabelas do módulo de Mecânica Gráfica do Programa Educ@r.

Uma tabela é composta de um conjunto de células. Células com fórmulas matemáticas e operações lógicas são definidas de forma a permitir comparar resultados preenchidos pelo aluno nas mesmas e mostrar em tempo real se estes resultados estão corretos ou não.

A tabela preenchida pelo aluno é reencaminhada a ele tendo em destaque células que foram preenchidas com valores incorretos, quando comparados aos resultados esperados. A tabela também é enviada ao mentor para que ele acompanhe o aluno, analisando suas respostas e fornecendo ao mesmo *feedback* necessário.

O TABIPRO implementa um histórico de todas as tabelas preenchidas pelos alunos, permitindo ao mentor ou outro pesquisador realizar análises estatísticas com todos os alunos, por grupos ou individualmente, propiciando estudos para encaminhar os alunos para obter os melhores resultados, ou ainda, realinhar atividades do curso se necessário.

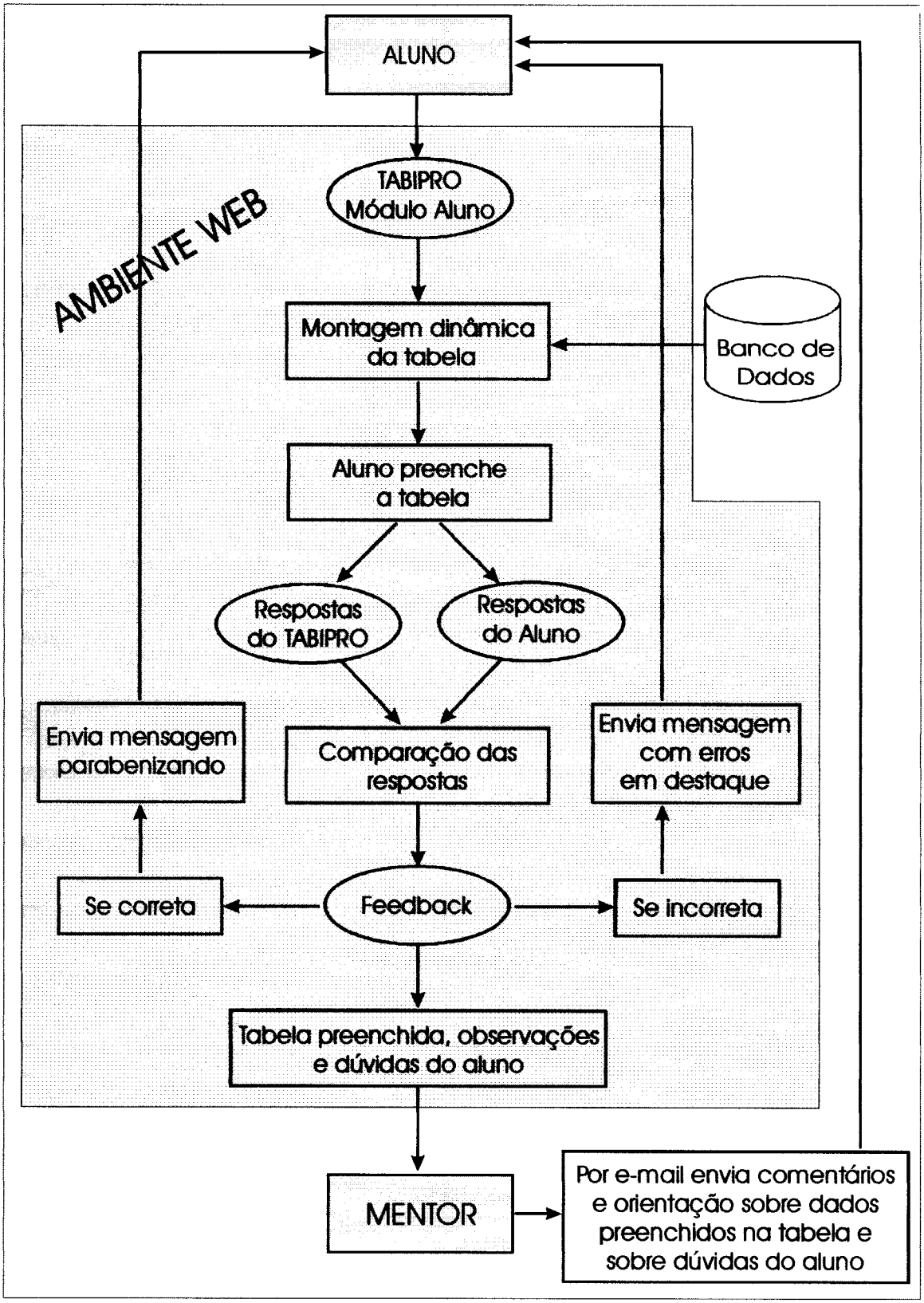


Figura 14 – Fluxo de tarefas do módulo Aluno.

10.4. A Estrutura do TABIPRO

O método de trabalho do TABIPRO pode ser apresentado em cinco etapas distintas, em sua aplicação no módulo de Mecânica Gráfica do Programa Educ@r.

a) O mentor através do TABIPRO cria o protótipo da tabela desejada, fornecendo informações que alimentam um banco de dados. As informações necessárias são:
   - ✓ Nome e título da tabela;
   - ✓ Número de linhas e de colunas;
   - ✓ Percentual de tolerância a erro para cálculos matemáticos;
   - ✓ E-mail do mentor;
   - ✓ Títulos das colunas;
   - ✓ Conteúdo de cada célula (incluindo fórmulas, comentários etc).

b) Por intermédio de *links*, as páginas do módulo de Mecânica Gráfica do Programa Educ@r acionam o TABIPRO, que busca as informações armazenadas no banco de dados e monta a tabela para o aluno preencher com os dados coletados e calculados nos experimentos realizados em laboratório.

c) Funções do TABIPRO avaliam se os cálculos dos alunos estão corretos em relação ao esperado pelo protótipo da tabela, levando em consideração a margem de erro definida pelo mentor.

d) O aluno recebe mensagem dizendo se os cálculos estão corretos ou não. Em caso de erro, o TABIPRO mostra onde está ocorrendo o erro.

e) O mentor recebe por e-mail a identificação e comentário do aluno, além de uma réplica da tabela preenchida, apontando os erros se existirem.

10.5.Etapas para Criar um Protótipo de Tabela

A seguir serão apresentadas resumidamente no formato de itens, as etapas necessárias para criar um protótipo de tabela utilizando o módulo Mentor:

a) Utilizando um *browser* acessar o endereço:
http://educar.sc.usp.br/tabipro/modmentor.php

b) Digitar o nome da tabela, que deve ser uma única palavra e ter no máximo dez caracteres.

c) Digitar o título da tabela com no máximo cem caracteres.

d) Digitar o número de linhas.

e) Digitar o número de colunas.

f) Digitar o percentual de tolerância de erro para usar nos cálculos das células com fórmula matemática.

g) Digitar o e-mail do mentor, para que cópias da tabela sejam a ele remetidas.

h) Pressionar o botão <Montar Tabela>.

i) Colocar o título em cada coluna da tabela, que deverá conter no máximo cem caracteres.

j) Montar o formato de exibição das células no módulo Aluno, determinando seus respectivos conteúdos que podem ser fórmulas,

comentários, imagens, solicitação de transparência e permissão para receber qualquer tipo de digitação.

k) Pressionar o botão <Montar Tabela>.

A seção 10.6 mostra através de exemplo como criar um protótipo de tabela detalhando cada um dos itens acima.

As seções 10.7.2 e 10.7.3 mostram os operadores e funções matemáticas que podem ser utilizados nas fórmulas.

10.6. Montando um Protótipo de Tabela – Um Exemplo Detalhado

Este capítulo deve tornar clara a criação de um protótipo de tabela e para isso, serão detalhadas todas as etapas necessárias, utilizando como exemplo a criação de uma tabela que ao ser solicitada pelo módulo Aluno, permita que suas células sejam preenchidas com dados obtidos em uma simulação, para cálculo do deslocamento de um ônibus numa trajetória retilínea. O aluno deve fazer cálculos de deslocamento ou variação do espaço entre vários pontos levantados na simulação. O objetivo desta tabela exemplo é permitir ao aluno informar dados de pontos iniciais e pontos finais determinados experimentalmente, sempre aos pares, calcular e informar o deslocamento ocorrido em cada par de pontos e verificar se estes cálculos estão corretos.

Como já visto, o TABIPRO é dividido em dois módulos, sendo um deles o módulo Mentor, onde é possível criar os protótipos de tabelas. No módulo Aluno, ocorre a montagem dinâmica e em tempo real da tabela que está sendo acessado pelo aluno através de um *browser*.

Para criar um protótipo de tabela com o módulo Mentor, utilizando-se de um *browser* o mentor acessa o endereço http://educar.sc.usp.br/tabipro/modmentor.php. Através do endereço o *browser* solicita ao servidor Web a execução de um *script* que exibirá para o mentor a tela (figura 15) pela qual será executada a rotina para criação do protótipo de tabela.

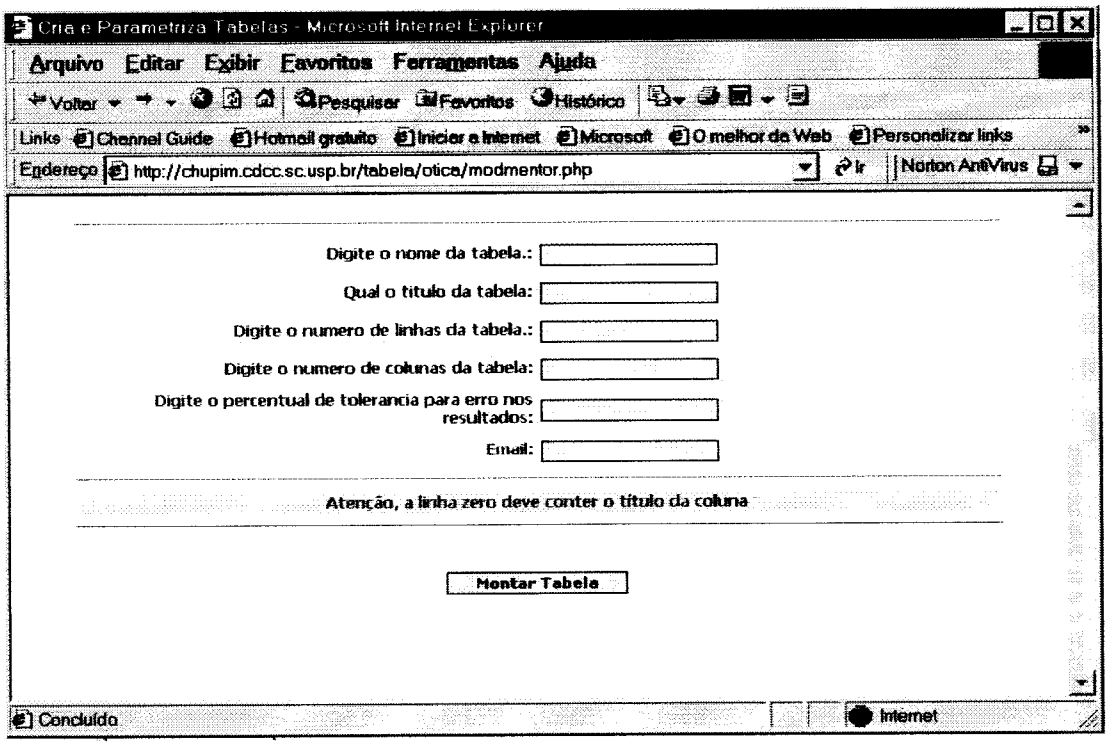


Figura 15 – Módulo Mentor, tela inicial para a criação de um protótipo de tabela.

10.6.1. O Exemplo Desloc

Para ilustrar o processo, será criado um protótipo de tabela chamado Desloc, que permita ao aluno analisar o deslocamento de um ônibus em linha reta, calculando o seu deslocamento a partir de posições ocupadas instantaneamente. Em seis instantes será coletado como informação o espaço percorrido pelo ônibus no marco quilométrico da estrada em que ele se encontra. Os seis marcos quilométricos serão chamados de D1, D2, D3, D4, D5 e D6. O Aluno deverá então, preencher a tabela com as informações de espaços coletados e calcular o deslocamento percorrido pelo ônibus entre cada par de dados informado. Por exemplo, se em D1 o ônibus estava no quilômetro 80 e em D2 o ônibus passou pelo quilômetro 105, então o deslocamento percorrido entre o par D1 e D2 foi de 35 quilômetros.

10.6.1.1. Definindo um Nome para a Tabela

Primeiramente o mentor define e digita um nome para a tabela, nome este que deve ter apenas uma palavra com até 10 caracteres e será utilizado pelo TABIPRO sempre que a tabela for chamada pelo módulo Aluno, ficando transparente para o aluno, ou seja, o aluno não precisa conhecer a existência deste nome como identificador da tabela.

Nome da tabela	**Desloc**

10.6.1.2. Um Título para a Tabela Desloc

A seguir o mentor digita um título para a tabela, que será exibido acima da mesma na tela do computador do aluno, quando esta for montada dinamicamente durante a execução do módulo Aluno do TABIPRO. É através deste título que o aluno deve conhecer e se referenciar à tabela quando estiver complementando seus experimentos laboratoriais ou interagindo com o mentor.

Título para a tabela	**Deslocamento em linha reta**

10.6.1.3. Número de Linhas e Colunas da Tabela Desloc

Uma tabela existe na forma de uma matriz de linhas e colunas. Sendo assim, é necessário definir o número de linhas e o número de colunas que formará a tabela, tendo ambos que ser maior que zero.

O próximo passo é digitar o número de linhas e o número de colunas da tabela.

Número de linhas	**5**

Número de colunas	**3**

10.6.1.4. Percentual de Erro para Fórmulas Matemáticas

No TABIPRO, o cruzamento de uma linha e uma coluna recebe o nome de célula. A célula só existe se houver um cruzamento de linha x coluna. A primeira célula da tabela estará localizada na primeira linha com a primeira coluna.

Durante a criação de um protótipo de tabela, a célula pode receber diversos formatos de conteúdo, que serão detalhados na seção 10.7.1. Um dos tipos possíveis de conteúdo é a utilização de fórmulas matemáticas. Se ao criar o protótipo uma célula recebe uma fórmula, quando esta tabela for exibida no módulo Aluno o conteúdo preenchido nesta célula será confrontado pela execução da fórmula definida no módulo Mentor. Com isso, é possível verificar se o aluno realizou corretamente os cálculos previstos.

Em alguns casos é possível considerar corretos os resultados que estiverem dentro de uma margem de erro definida por um valor percentual. Por exemplo, o aluno informa como resposta em uma célula o valor 51, sendo 50, o valor correto de acordo com a fórmula definida na mesma. Se a margem de erro for de 0% (zero porcento), a resposta do aluno está errada, porém, se a margem de erro for de 2% (dois porcento) a resposta está correta.

Continuando a criação do protótipo, será informado neste instante o valor percentual (sem o símbolo %) de tolerância a erro para ser utilizado nos cálculos de todas as fórmulas propostas para a tabela.

Percentual de tolerância para erro	**1**

10.6.1.5. E-mail para Envio da Tabela

Toda tabela que for preenchida no módulo Aluno terá uma réplica enviada ao mentor através de e-mail. Isto é importante para que o mentor acompanhe o desenvolvimento do aluno através dos exercícios práticos representados na tabela. Para tanto, é necessário informar um endereço eletrônico (e-mail) para que o módulo Aluno faça a remessa da tabela preenchida.

E-mail para remessa de tabelas preenchidas	sergio@rc.unesp.br

A figura 16 mostra a tela do módulo Mentor com os parâmetros iniciais necessários para a criação do protótipo da tabela.

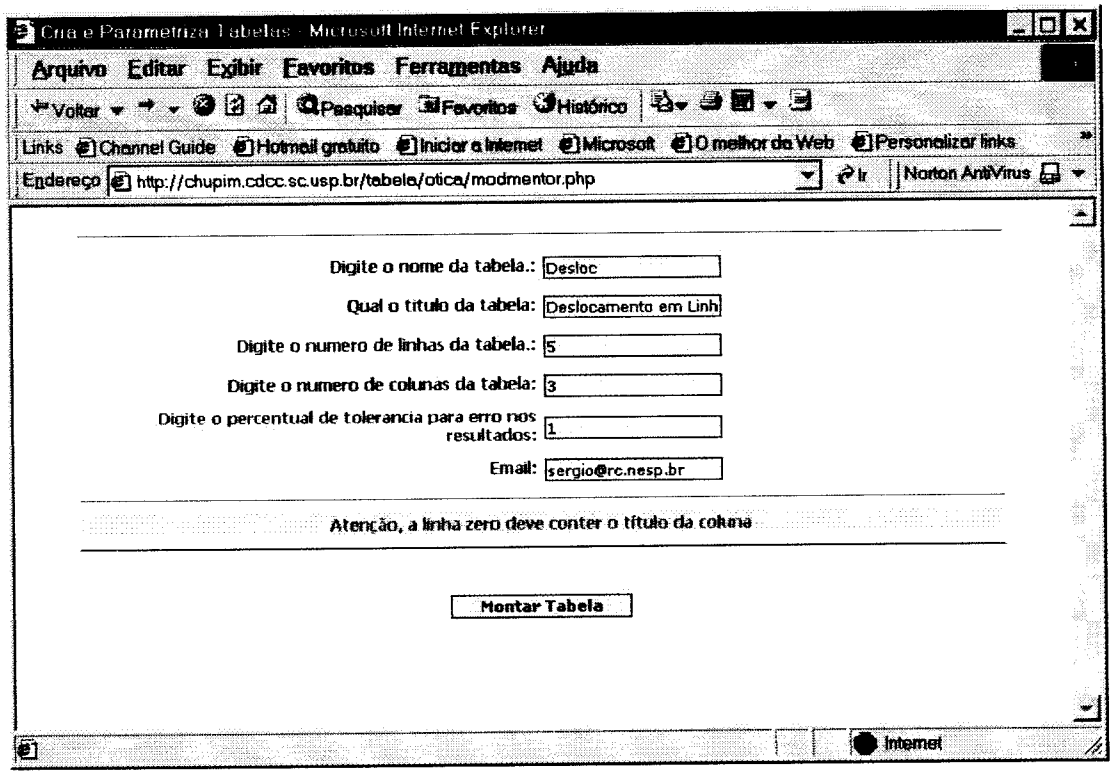


Figura 16 – Tela inicial para criação de um protótipo de tabela, com parâmetros preenchidos.

Até o momento, as seguintes informações foram fornecidas ao módulo Mentor:

- ✓ Nome da tabela
- ✓ Título para a tabela
- ✓ Número de linhas
- ✓ Número de colunas
- ✓ Percentual de tolerância para erro
- ✓ E-mail para remessa de tabelas preenchidas

Após o preenchimento dos parâmetros iniciais, o mentor pressiona o botão <Montar Tabela>, que resultará na tela mostrada na figura 17.

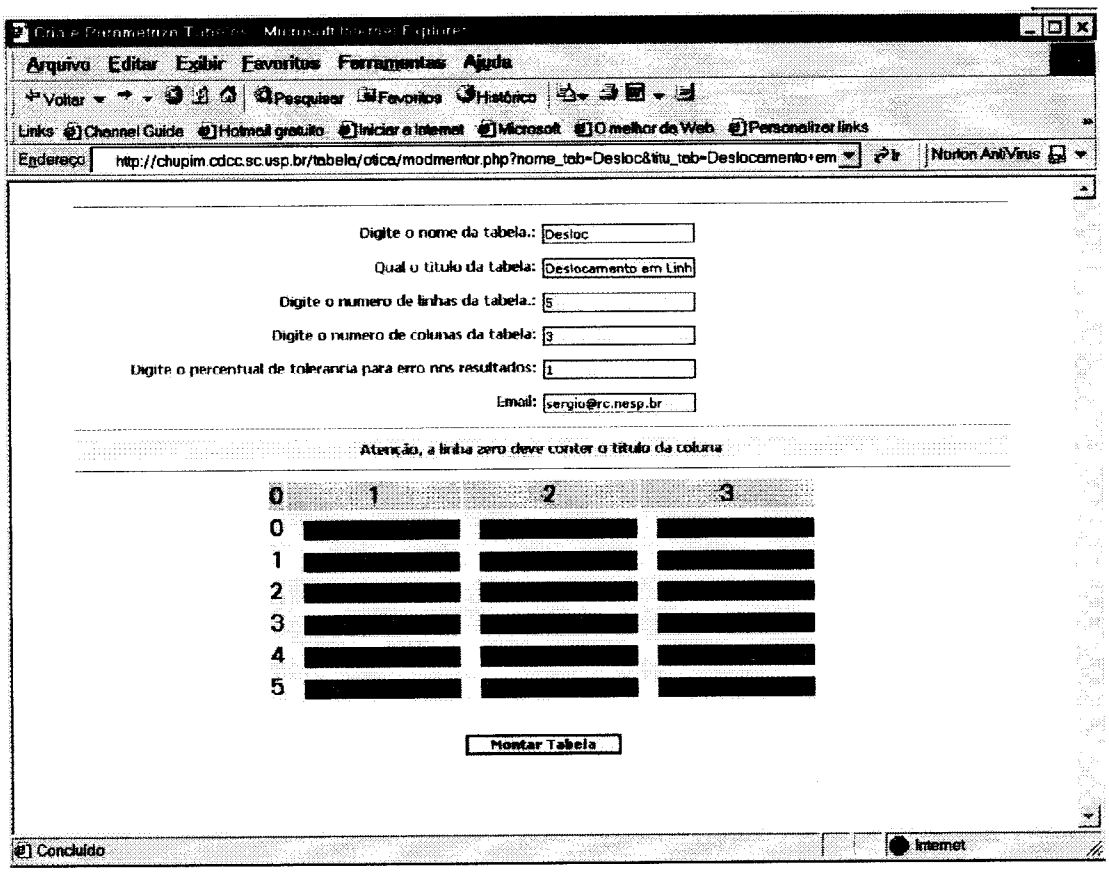


Figura 17 – Tela com as células que comporão a tabela.

10.6.1.6. A Formatação das Células da Tabela Desloc

Agora chegou o momento de formatar o conteúdo das células da tabela, que implicará diretamente na forma como a mesma será apresentada no módulo Aluno, além de como ela deverá ser preenchida.

Na seção 10.7.1 deste trabalho, são detalhados os mais diversos tipos de formatações de conteúdos de célula.

10.6.1.7. Títulos para as Colunas

Toda tabela é composta de no mínimo uma coluna, sendo importante definir um título para cada coluna existente. O título será mostrado no topo da respectiva coluna, sempre que a tabela for exibida no módulo Aluno do TABIPRO. No processo de montagem do protótipo é que ocorre a definição do título das colunas.

Observe que este protótipo possui três colunas e cinco linhas. No módulo Mentor aparece a linha 0 (zero) que é utilizada para a inserção do título da respectiva coluna. É aconselhável que o título da coluna represente os valores que em suas células serão preenchidos ou exibidos.

Continuando o exemplo do protótipo Desloc, a primeira coluna receberá o título 'Dini', uma vez que representa o espaço inicial de cada par coletado. A segunda coluna terá como título 'Dfin', representando o espaço final do par coletado. A terceira e última coluna receberá o título 'Desloc', já que conterá o resultado do deslocamento realizado entre os dois valores do par, ou seja, deverá conter o valor de Dini subtraído de Dfin.

Título da primeira coluna	**Dini**

Título da segunda coluna	**Dfin**

Título da terceira coluna	**Desloc**

10.6.1.8. Utilizando Fórmulas Matemáticas

Toda célula em uma tabela do TABIPRO é referenciada pela linha e coluna em que se encontra, sendo obrigatório utilizar o símbolo $ na identificação da mesma. Por exemplo, a célula da terceira linha e segunda coluna deve ser referenciada como $L3C2.

Na terceira coluna do protótipo da tabela Desloc, ocorre o tipo de conteúdo de célula que é a fórmula. Todas as células desta coluna conterão uma fórmula, que demonstra como deve ser realizada a subtração 'Dfin' menos 'Dini'.

O conteúdo da célula da linha 1 coluna 3, representa a fórmula do deslocamento entre a posição inicial e a posição final do ônibus (Dfin – Dini) em dois momentos distintos, e será $L1C2 - $L1C1, lembrando que na primeira coluna da tabela encontra-se o espaço inicial e na segunda coluna o espaço final do par coletado. Para as linhas subsequentes, as fórmulas da terceira coluna serão: $L2C2 - $L2C1, na segunda linha; $L3C2 - $L3C1, na terceira linha; $L4C2 - $L4C1, na quarta linha; e $L5C2 - $L5C1, na quinta e última linha deste protótipo.

Fórmula da primeira linha x terceira coluna	**$L1C2 - $L1C1**

As figuras 18 e 19 mostram como os conteúdos das células do protótipo da tabela Desloc deverão estar preenchidos.

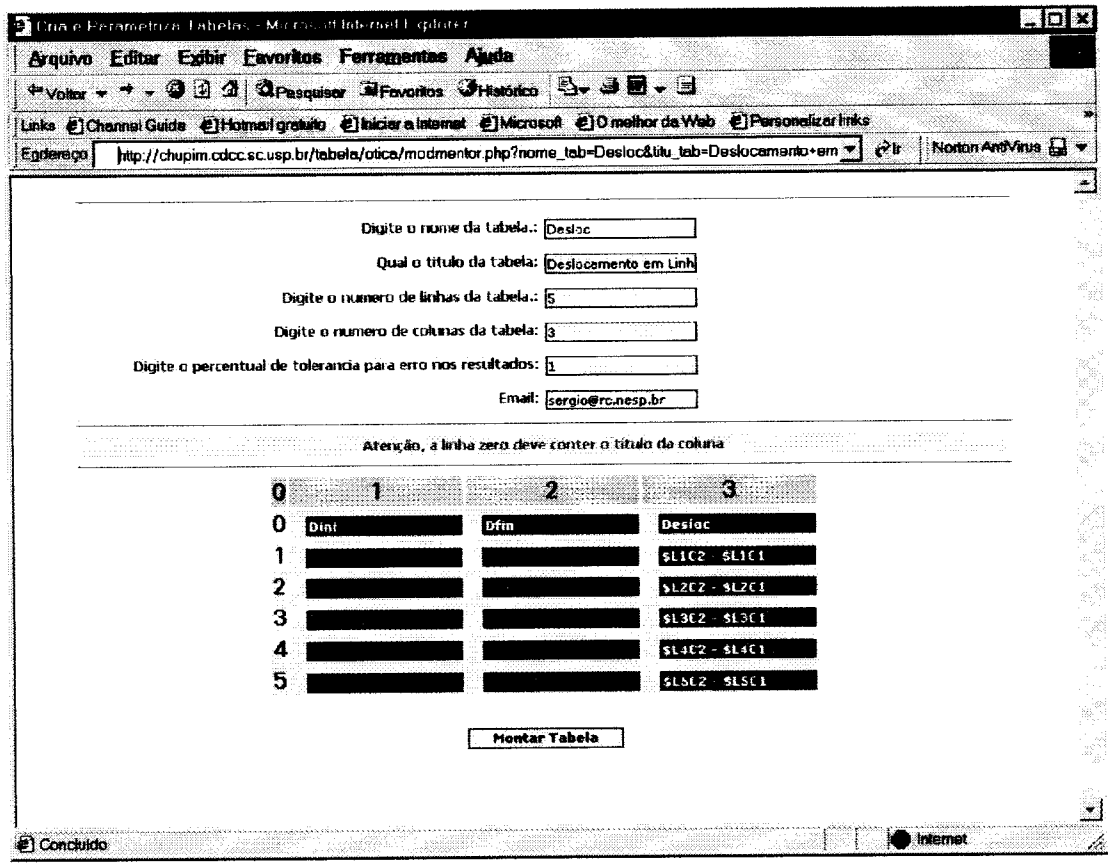


Figura 18 – Conteúdo das células que comporão a tabela.

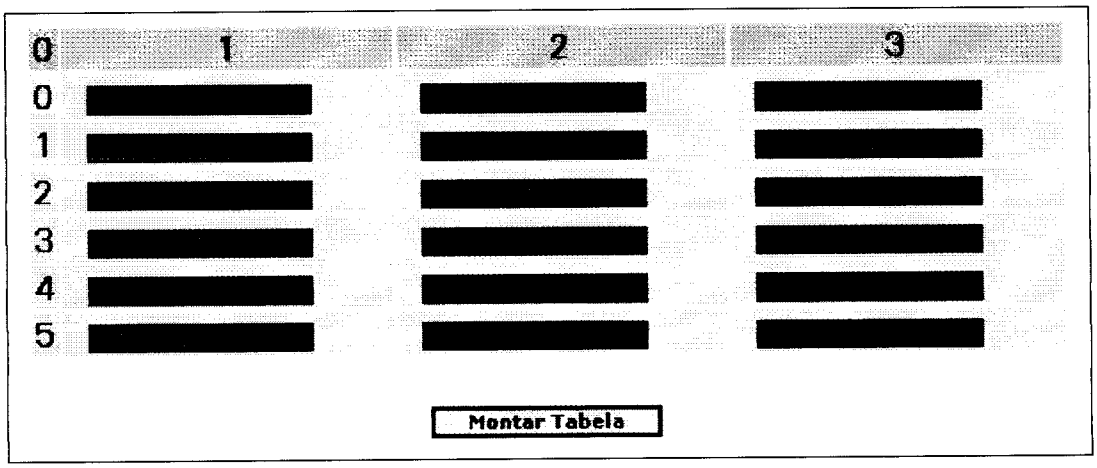


Figura 19 – Destaque das células que comporão a tabela.

10.6.1.9. Concluindo a Criação do Protótipo da Tabela Desloc

Após o preenchimento do conteúdo de todas as células, o mentor pressiona o botão <Montar Tabela> que concluirá o processo. Uma mensagem de processo finalizado será exibida na tela juntamente com um *link*, que o convida a testar a tabela que acaba de ser criada.

Se *link* citado for utilizado o módulo aluno é acionado, criando dinamicamente a tabela Desloc e apresentando-a em seguida ao mentor, conforme mostra a figura 20.

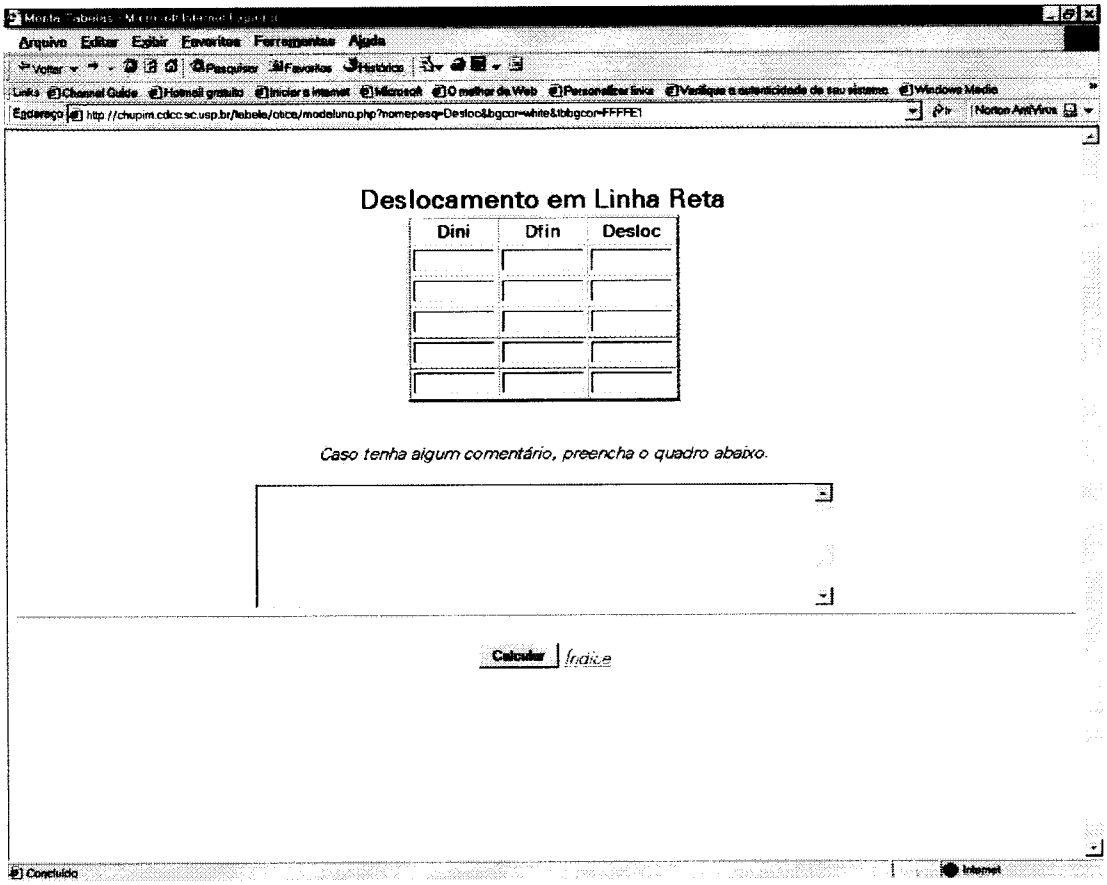


Figura 20 – Execução da tabela Desloc no módulo Aluno.

10.7. Formatação de Células, Operadores e Funções Matemáticas

10.7.1. Formatação de Células

No TABIPRO, uma célula é por definição o elemento resultante do cruzamento de uma linha com uma coluna.

Durante a criação de um protótipo de tabela, a célula pode receber diversos formatos de conteúdo como: comentários, fórmulas matemáticas, operações lógicas e até mesmo solicitação de que a célula não seja exibida durante a montagem da tabela no módulo Aluno.

A seguir serão apresentadas as formatações possíveis para células do TABIPRO:

ne – Não exibe

Quando o módulo Aluno encontra uma célula com conteúdo 'ne', omite a exibição da mesma durante a montagem da tabela para o aluno.

ex:Texto – Comentário

Ao encontrar uma célula com 'ex:', o módulo Aluno apresentará 'Texto' como comentário nesta célula, não permitindo ao aluno alterar seu conteúdo, ou seja, esta célula estará protegida contra digitação.

fórmulas – Utilização de fórmulas

É possível criar fórmulas com operações matemáticas e lógicas utilizando operadores e funções. Se uma célula estiver envolvida em uma operação matemática ou lógica, deverá estar representada por $LxCy, onde Lx indica a linha e Cy a coluna. As variáveis x e y devem ser substituídas por números que representam a linha e a coluna da referida célula. Como exemplo, a fórmula $L1C1 / $L1C2, mostra a definição onde o conteúdo da célula da linha 1 coluna 1 será dividido pelo conteúdo da célula da linha 1 coluna 2.

Imagens - Utilização de imagens

É possível a inserção de figuras no formato raster nos títulos das colunas e no conteúdo de células, utilizando o recurso IMG SRC, descrito a seguir:

<IMG SRC=host/diretório/nome_da_imagem.gif>

" " – Digitação livre, sem checagem de conteúdo

Para definir as células que possuam como característica, a possibilidade do usuário do módulo Aluno realizar digitação sem que seu conteúdo seja verificado, basta que na definição do protótipo esta célula fique com conteúdo vazio.

10.7.2. Operadores

Durante o processo de criar tabelas, o módulo Mentor permite utilizar fórmulas com operadores aritméticos, lógicos, de string e de comparação. Estes operadores e a forma de utilizá-los estão descritos segundo Bakken et al (2003) a seguir.

10.7.2.1. Operadores Aritméticos

Devem ser utilizados para tratamento de células com conteúdo numérico. Se forem de outro tipo, terão seus valores convertidos antes da realização da operação.

Operador	Nome	Exemplo	Resultado
+	Adição	$a + $b	Soma de $a e $b.
-	Subtração	$a - $b	Diferença entre $a e $b.
*	Multiplicação	$a * $b	Produto de $a e $b.
/	Divisão	$a / $b	Quociente de $a por $b.
%	Módulo	$a % $b	Resto de $a dividido por $b.

Tabela 1 – Relação de operadores aritméticos. Fonte: Bakken, et al (2003).

O operador de divisão ("/") sempre retorna um valor com ponto flutuante, mesmo que os dois operandos sejam inteiros (ou strings que sejam convertidas para inteiros).

10.7.2.2. Operador de String

Deve ser utilizado para tratamento de conteúdo alfanumérico. A concatenação ('.') retorna a junção dos seus argumentos direito e esquerdo.

Operador	Nome	Exemplo	Resultado
.	concatenação	"Olá " . "Mundo"	"Olá Mundo"

Tabela 2 – Operadores de string. Fonte: Bakken, et al (2003).

10.7.2.3. Operadores Lógicos

Utilizados para inteiros representando valores booleanos. Existem dois operadores para "e" e para "ou", porque eles têm diferentes posições na ordem de precedência (ver tabela 5).

Operador	Nome	Exemplo	Resultado
And	"e" lógico	$a and $b	verdadeiro se $a e $b são verdadeiros
Or	"ou" lógico	$a or $b	Verdadeiro se $a ou $b são verdadeiros
Xor	Ou exclusivo		Verdadeiro se $a ou $b são verdadeiros, mas não ambos.
!	Não (inversão)	!$a	Verdadeiro se $a for falso
&&	"e" lógico	$a && $b	Verdadeiro se tanto $a quanto $b são verdadeiros.
||	"ou" lógico	$a || $b	Verdadeiro se $a ou $b são verdadeiros.

Tabela 3 – Relação de operadores lógicos. Fonte: Bakken, et al (2003).

10.7.2.4. Operadores de Comparação

Operadores de comparação, como os seus nomes implicam, permitem que você compare dois valores. As comparações são feitas entre os valores contidos nas células. Sempre retornam um valor booleano (TRUE para verdadeiro e FALSE para falso).

Operador	Nome	Exemplo	Resultado
= =	Igual a	$a = = $b	Verdadeiro se $a for igual a $b
!= ou < >	Diferente de	$a != $b	Verdadeiro se $a for diferente de $b
<	Menor que	$a < $b	Verdadeiro se $a for menor que $b
>	Maior que	$a > $b	Verdadeiro se $a for maior que $b
<=	Menor ou igual a	$a >= $b	Verdadeiro se $a for maior ou igual a $b
>=	Maior ou igual a	$a <= $b	Verdadeiro se $a for menor ou igual a $b

Tabela 4 – Relação de operadores de comparação. Fonte: Bakken, et al (2003)

Existe também um operador de seleção, o operador "?:" , que é ternário e funciona assim: (expressao1)?(expressao2):(expressao3)

Neste caso é avaliada a primeira expressão. Se ela for verdadeira, o operador retorna o valor de expressão2. Senão, retorna o valor de expressão3.

10.7.2.5. Precedência de Operadores

A precedência de um operador especifica quem tem mais prioridade quando há dois deles juntos. Por exemplo, na expressão, 1 + 5 * 3, o resultado é 16, porque o operador de multiplicação ("*") tem prioridade de precedência em relação ao operador de adição ("+"). Parênteses podem ser utilizados para forçar a precedência, se necessário. Assim, o resultado da operação (1 + 5) * 3 é 18.

A tabela 5 mostra a precedência dos operadores, indo da maior para a menor.

Associação	Operador
esquerda	* / %
esquerda	+ -
não associativo	< <= > >=
não associativo	= = !=
esquerda	&
esquerda	^
esquerda	|
esquerda	&&
esquerda	||
esquerda	? :
esquerda	and
esquerda	xor
esquerda	or

Tabela 5 – Precedência de operadores. Fonte: Bakken, et al (2003).

10.7.3. Funções Matemáticas e Trigonométricas

A seguir estão as funções matemáticas e trigonométricas (Bakken et al, 2002a) que podem ser utilizadas no módulo Mentor durante a criação de um protótipo de tabela.

abs - Valor absoluto

Retorna o valor absoluto de um número.

Formato: abs (número)

acos - Cosseno Inverso (arco cosseno)

Retorna o inverso do coseno de arg em radianos. acos() é a função complementar de cos().

Formato: acos (arg)

acosh - Coseno Hiperbólico Inverso

Retorna o coseno hiperbólico de arg, isto é, o valor cujo coseno hiperbólico é arg.

Formato: acosh (arg)

asin - Seno Inverso (arco seno)

Retorna o inverso do seno de arg em radianos. asin() é a função complementar de sin().

Formato: asin (arg)

asinh - Seno Hiperbólico Inverso

Retorna o inverso do seno hiperbólico de arg, isto é, o valor cujo seno hiperbólico é arg.

Formato: asinh (arg)

atan2 - Tangente inversa de duas variáveis

Esta função calcula a tangente inversa de duas variáveis x e y. Esta função é o equivalente a calcular a tangente inversa de y / x, exceto que os sinais de ambos os argumentos são usados para determinar o quadrante do resultado.

Esta função retorna o resultado em radianos, estando entre -PI e PI (inclusive).

Formato: atan2 (y, x)

atan - Tangente Inversa (arco tangente)

Retorna o inverso da tangente de arg em radianos. atan() é a função complementar de tan().

Formato: atan (arg)

atanh - Tangente hiperbólica inversa

Retorna a tangente hiperbólica inversa de arg, isto é, o valor cuja tangente hiperbólica é arg.

Formato: atanh (arg)

bindec - Binário para decimal

Retorna o equivalente decimal do número binário representado pelo argumento binary. bindec() converte um número binário em um inteiro

Formato: bindec (binary)

ceil - Arredonda frações para cima

Retorna o próximo maior valor inteiro arredondando para cima do valor, se fracionário. O valor retornado de ceil() é do tipo float porque a dimensão dos valores suportados por float é normalmente maior do que o do tipo int.

Formato: ceil (valor)

cos - Coseno

Retorna o coseno de arg. O parâmetro arg deve estar em radianos.

Formato: cos (arg)

cosh - Coseno hiperbólico

Retorna o coseno hiperbólico de arg, definido por (exp(arg) + exp(-arg))/2.

Formato: cosh (arg)

decbin - Decimal para binário

Retorna uma string contendo a representação binária do parâmetro número.

Formato: decbin (número)

dechex - Decimal para hexadecimal

Retorna uma string contendo a representação hexadecimal do argumento número.

Formato: dechex (número)

decoct - Decimal para octal

Retorna uma string contendo a representação octal do parâmetro número.

Formato: decoct (número)

deg2rad - Converte o número em graus ao equivalente em radianos

Esta função converte número de graus ao equivalente em radianos.

Formato: deg2rad (número)

exp - Potência

Retorna e a potência de arg.

Formato: exp (arg)

floor - Arredonda frações para baixo

Retorna o próximo menor valor inteiro ao se arredondar para baixo o número, se necessário. O valor retornado de floor() é do tipo float.

Formato: floor (número)

hexdec - Hexadecimal para decimal

Retorna o decimal equivalente do número hexadecimal representado pelo argumento hex_string. hexdec() converte uma string hexadecimal para um número decimal. hexdec() substituirá qualquer caracter não hexadecimal que encontrar por 0. Desta forma, todos os zeros a esquerda são ignorados, mas zeros a direita serão avaliados.

Formato: hexdec (hex_string)

log10 - Logaritmo Base 10

Retorna o logaritmo na base 10 de arg.

Formato: log10 (arg)

log - Logaritmo natural

Retorna o logaritmo natural de arg.

Formato: log (arg)

max - Localiza o maior número

Retorna o maior número dos parâmetros.

Formato: max (arg1, arg2, ..., argn)

min - Localiza o menor número

Retorna o menor número dos parâmetros.

Formato: min (arg1, arg2, ..., argn)

mdc – Média em coluna

Retorna o cálculo de média entre células de uma mesma coluna. A média é calculada entre os conteúdos das células definidas pelos parâmetros X (linha inicial) e Y (linha final), na coluna em que a fórmula se encontra.

Formato: mdcX:Y

mt_rand - Gerador melhorado de números aleatórios

Esta função utiliza um gerador de números aleatórios com características conhecidas, o Mersenne Twister. A homepage do Mersenne Twister pode ser encontrada em http://www.math.keio.ac.jp/~matumoto/emt.html. Se os parâmetros min e max forem definidos, será gerado um número aleatório entre min e max.

Formato: mt_rand (min, max)

octdec - Octal para decimal

Retorna o decimal equivalente do número octal representado pelo argumento octal_string.

Formato: octdec (octal_string)

pow - Potência

Retorna a base elevada ao expoente exp.

Formato: pow (base, exp)

rad2deg - Converte o número em radianos para o equivalente em graus

Esta função converte arg de radianos para graus.

Formato: rad2deg (arg)

rand - Gera um número aleatório

Gera um valor randômico. Se utilizar os parâmetros opcionais min e max, retorna um número aleatório entre min e max (inclusive).

Formato: rand ([min, max])

round - Arredonda um número

Retorna um valor arredondado de val em precision casas decimais.

Formato: round (val [, int precision])

sin - Seno

Retorna o seno do parâmetro arg. O argumento arg deve estar em radianos.

Formato: sin (arg)

sinh - Seno hiperbólico

Retorna o seno hiperbólico de arg, definido como (exp(arg) - exp(-arg))/2.

Formato: sinh (arg)

sqrt - Raiz quadrada

Retorna a raiz quadrada de arg.

Formato: sqrt (arg)

tan - Tangente

Retorna a tangente do parâmetro arg. O argumento arg deve estar em radianos.

Formato: tan (arg)

tanh - Tangente hiperbólica

Retorna a tangente hiperbólica de arg, definida como sinh(arg)/cosh(arg).

Formato: tanh (arg)

10.8. Utilização de uma Tabela no Módulo Aluno

Como já explicado, o módulo Aluno do TABIPRO é um *server-side script*, e foi desenvolvido para ser executado a partir de uma página Web hospedada em um servidor, que não é necessariamente o mesmo servidor do TABIPRO. Como o

TABIPRO pode atender inúmeros projetos de EAD, é interessante que esta página esteja no servidor do curso para o qual o módulo Aluno estará montando as tabelas interativas. Para isso, deve existir no código fonte da referida página uma chamada ao módulo Aluno. Nesta chamada é necessário passar como informações, o nome da tabela, a sua cor e cor de fundo da página em que a mesma será montada dinamicamente.

A utilização de um protótipo de tabela será demostrada através do exemplo da tabela Desloc, criado e discutido na seção 10.6 com a utilização do módulo Mentor. A partir de uma página Web, o módulo Aluno do TABIPRO é acionado e a tabela Desloc carregada e montada dinamicamente na tela do computador.

A figura 21 é um exemplo de página Web que aciona o módulo Aluno. Esta página tem seu código fonte visualizado na figura 22, com destaque onde ocorre a chamada ao módulo Aluno. Como já visto, são três os parâmetros que devem ser enviados para o módulo aluno: nome da tabela, que será enviada através da variável nomepesq; cor de fundo da página, que será enviada através da variável bgcor; e cor da tabela, enviada pela variável tbbgcor.

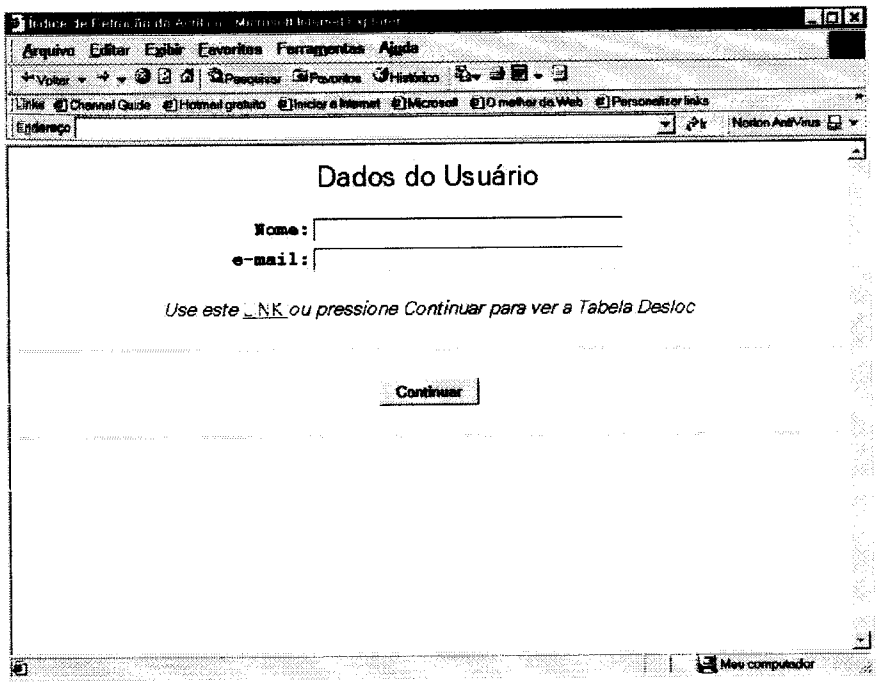


Figura 21 – Exemplo de uma página Web que aciona o módulo Aluno do TABIPRO.

```
<html>

<head> <title> Tabela Desloc </title> </head><body bgcolor="white">
<FORM name="f1" id="f1" method="POST"
action="http://chupim.cdcc.sc.usp.br/tabela/otica/modaluno.php?nomepesq=
Desloc&bgcor=white&tbbgcor=FFFFE1">

<center>
 <font size=+2> Dados do Usuário</font>
</center><p>
<center>
<table border=0>
<tr>
<td><TT><b><font size=+1><P ALIGN="right">Nome: </font></b></TT></td>
<td><input type="text" name="aluno" value="" size="40">
</td></tr>
<tr><td><TT><b><font size=+1><P ALIGN="right">e-
mail:</font></b></TT></td>
<td><input type="text" name="mailaluno" value="" size="40">
</td>
</tr>
```

Figura 22a – Código HTML de uma página exemplo que aciona o módulo Aluno do TABIPRO.

```
<center>  <i> Pressione Continuar para ver a Tabela Desloc </i></center>
<BR><HR ALIGN="center" SIZE=3>

<p> <center>
<input type="submit" name="submit" value="Continuar">
 </form> </center> </basefont>
 <HR ALIGN="center" SIZE=3> <br> <font size= 1>
</font> <br>
</body> </html>
```

Figura 22b – Continuação do código HTML de uma página exemplo que aciona o módulo Aluno do TABIPRO.

Durante a execução do módulo aluno, será aberto um canal de interatividade entre aluno e mentor via e-mail. Para que isso ocorra, é necessário existir na página Web que chamou o módulo Aluno, dois campos de entrada de dados nomeados aluno e mailaluno. O mailaluno deverá receber como conteúdo, o e-mail do aluno que irá preencher os dados na tabela.

O código HTML a seguir, mostra como podem ser criados os campos aluno e mailaluno, para entrada de dados em uma página Web.

```
<input type="text" name="aluno" value="" size="40">

<input type="text" name="mailaluno" value="" size="40">
```

Figura 23 – Código HTML para criar campos de entrada de dados.

10.8.1. Um Aluno Utilizando a Tabela Desloc

Após coletar dados do deslocamento do ônibus proposto como exemplo na seção 10.6, o aluno deve preencher a tabela que lhe foi apresentada na tela do computador (figura 24) pelo módulo Aluno.

Os dados coletados foram a posição inicial no quilômetro 5, e as posições nos quilômetros 8, 11.5, 15, 18 e 20.

O aluno deve calcular o deslocamento ocorrido entre cada par de pontos coletados a saber: entre 5 e 8, deslocamento de três quilômetros; entre 8 e 11.5, três quilômetros e meio; ente 11.5 e 15, três quilômetros e meio; entre 15 e 18, três quilômetros; e entre 18 e 20, dois quilômetros.

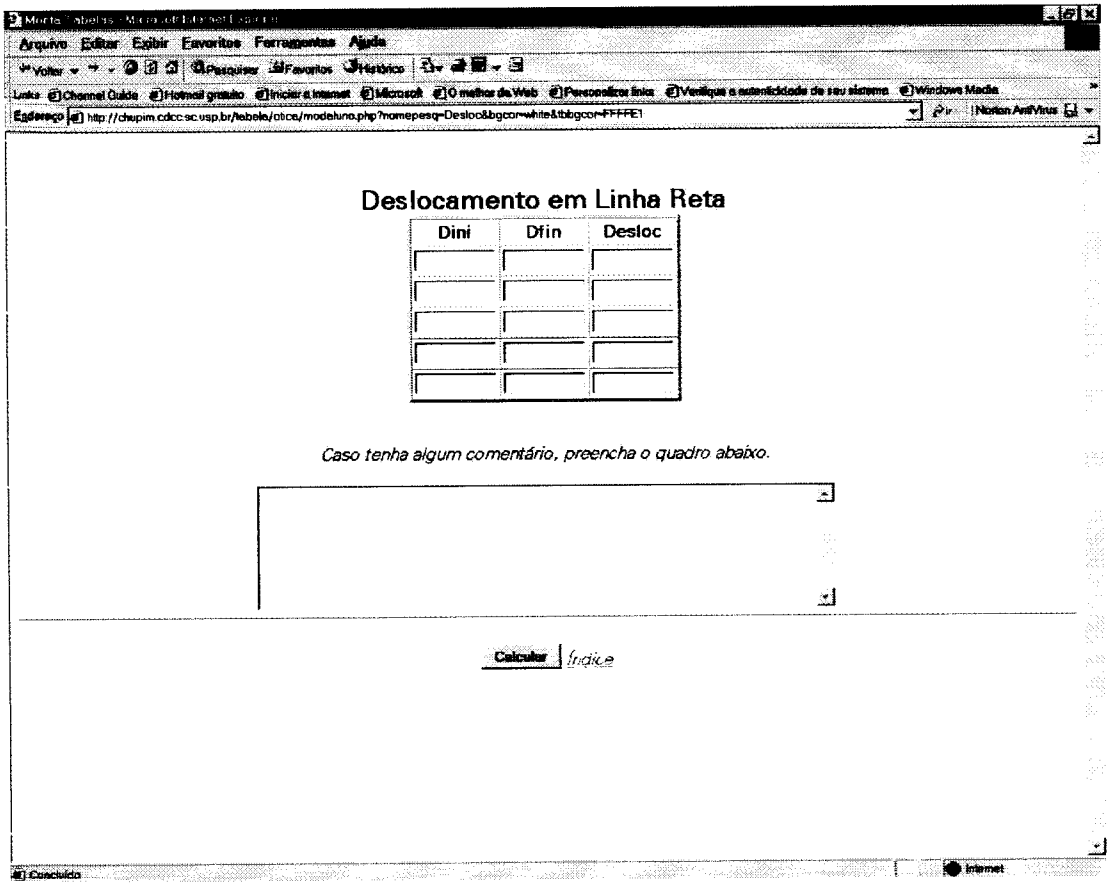


Figura 24 – Módulo Aluno monta a tabela que deve ser preenchida.

Ao preencher a tabela com os dados coletados, o aluno comete um erro no cálculo do deslocamento no último par de dados informado. Nas posições 18 e 20 quilômetros, o aluno preenche a tabela informando deslocamento de três quilômetros, sendo dois o valor correto. O módulo Aluno do TABIPRO verifica o erro, exibe a tabela com destaque sobre o mesmo e emite uma mensagem de erro, como demonstra a figura 26. Após refazer o cálculo, o aluno digita o valor correto na tabela. Então, o módulo Aluno compara os dados digitados e verifica a não existência de erros, emitindo assim mensagem de parabéns para o aluno, mostrada na figura 27.

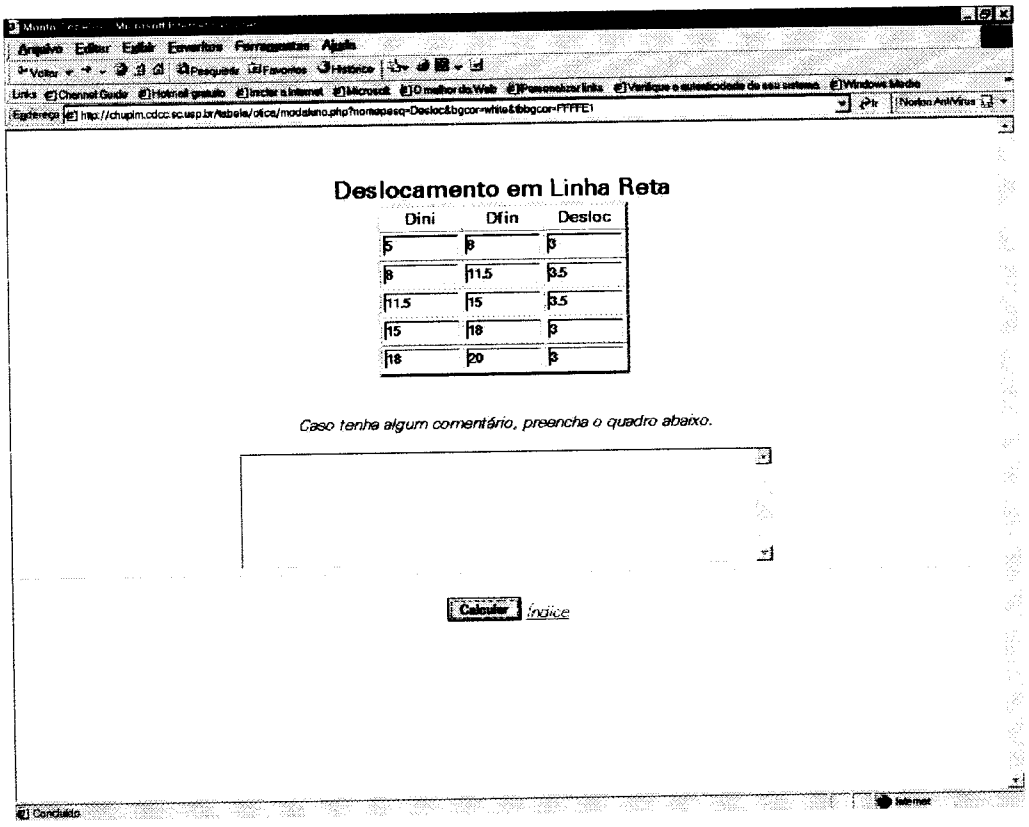


Figura 25 – Módulo Aluno recebe digitação de dados.

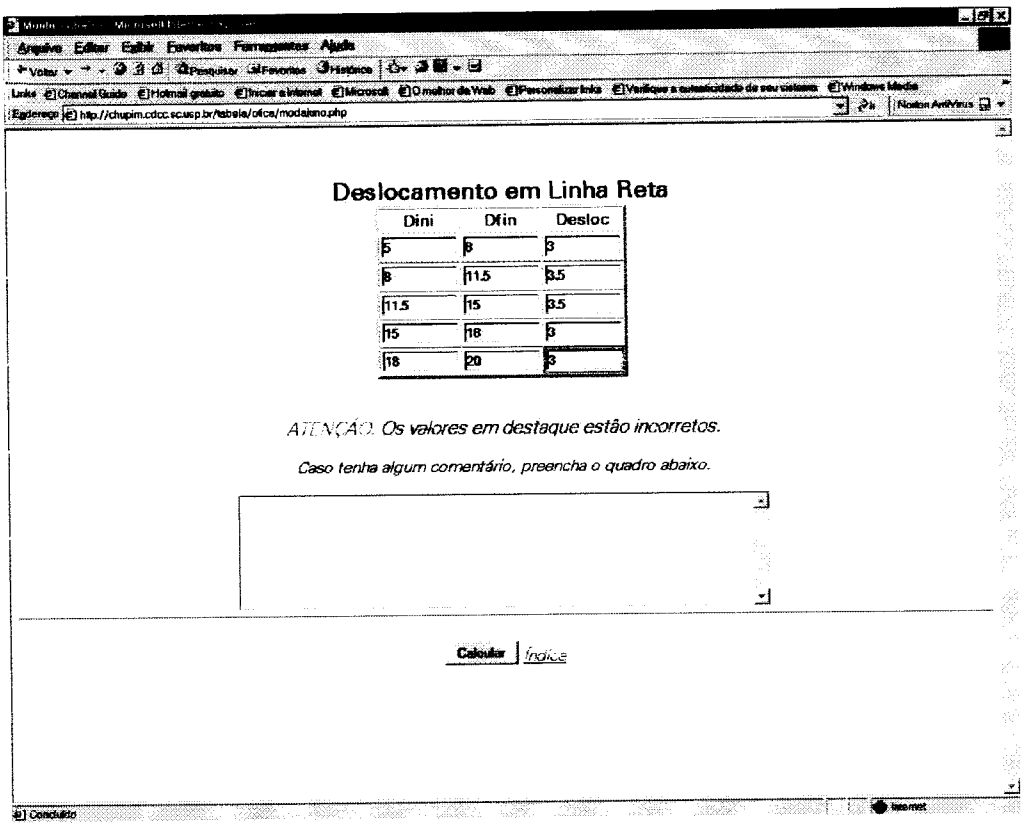


Figura 26 – Módulo Aluno destaca a ocorrência de erro nos dados digitados.

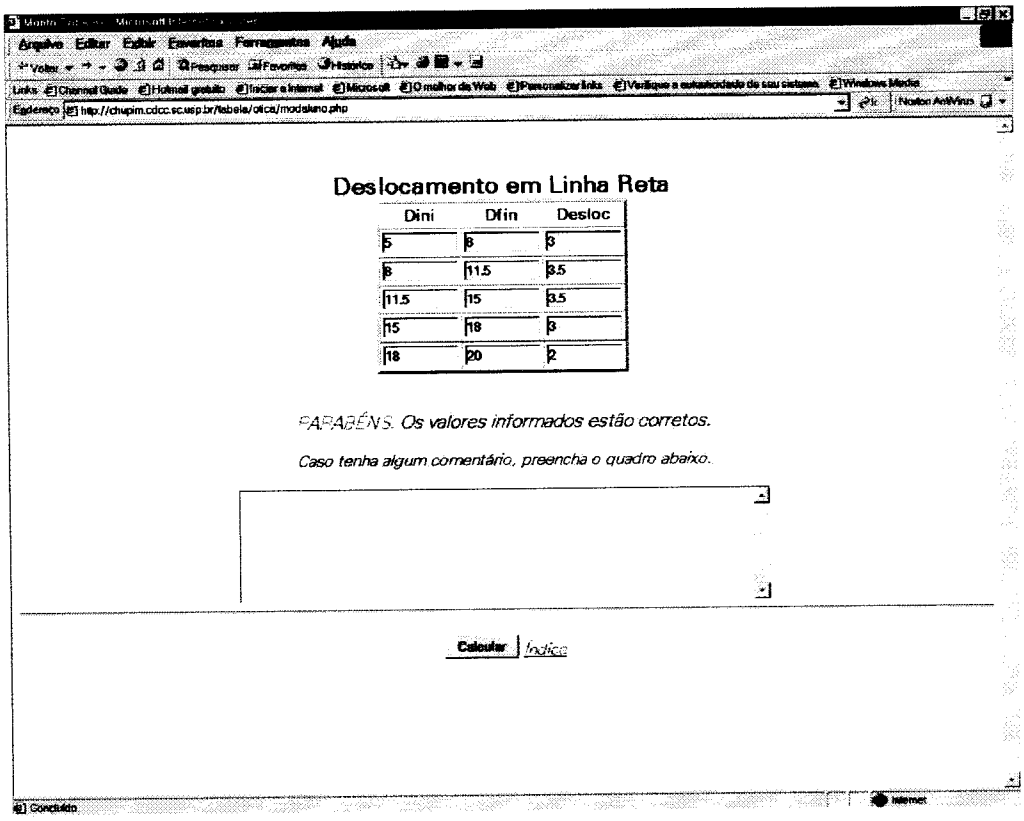


Figura 27 – Módulo Aluno exibe mensagem de parabéns se não ocorrer erro nos dados digitados.

11. Aplicações das Tabelas Interativas Programáveis no Módulo de Mecânica Gráfica do Programa Educ@r

O módulo de Mecânica Gráfica do Programa Educ@ar possui dezessete tabelas, que são utilizadas pelos alunos após as aulas presenciais e experimentos laboratoriais. As tabelas possuem formatos diferentes e para serem utilizadas no TABIPRO, precisam ter seus protótipos criados pelo mentor.

Abaixo está a relação das tabelas do módulo de Mecânica Gráfica que podem ser visualizadas no anexo deste trabalho.

- ✓ Tabela 1.1 - Movimento unidimensional
- ✓ Tabela 1.2 - Movimento bidimensional
- ✓ Tabela 2.1 - Movimento uniforme 1
- ✓ Tabela 2.2 - Movimento uniforme 2
- ✓ Tabela 3.1 - Movimento uniformemente variado 1
- ✓ Tabela 3.2 - Movimento uniformemente variado 2

✓ Tabela 4.1 - Movimento de projéteis 1

✓ Tabela 4.2 - Movimento de projéteis 2

✓ Tabela 4.3 - Movimento de projéteis 3

✓ Tabela 4.4 - Movimento de projéteis 4

✓ Tabela 5.1 - Movimento circular 1

✓ Tabela 5.2 - Movimento circular 2

✓ Tabela 5.3 - Movimento circular 3

✓ Tabela 6.1 - Introdução ao estudo da dinâmica 1

✓ Tabela 6.2 - Introdução ao estudo da dinâmica 2

✓ Tabela 7.1 - Movimento central

✓ Tabela 8.1 - Conservação de energia mecânica

12. O Ambiente Operacional do TABIPRO

O ambiente operacional do Tabelas Interativas Programáveis é composto por sistema operacional e softwares *freeware*, para que possa ser reproduzido e utilizado em outras instituições sem ônus financeiro para sua operacionalização.

Embora a estrutura original do TABIPRO suporte a criação e utilização de protótipos de tabela de qualquer lugar na Internet, um fator que deve ser levando em consideração é o tráfego na rede e o tempo de resposta necessário para o módulo Aluno trabalhar. Este módulo é acionado a partir de uma página Web hospedada em um servidor na Internet. Se o *link* desta página ao servidor do TABIPRO (que está localizado em São Carlos) for lento, é recomendado reproduzir o ambiente operacional do TABIPRO no servidor da página em questão, ou em outro onde o tráfego da rede seja menor, a fim de melhorar o tempo de resposta da comunicação entre o software e o aluno.

O sistema operacional onde o TABIPRO está operando é o FreeBSD 4.5 que pode ser obtido no endereço http://www.freebsd.org, tendo o Apache 1.3 como servidor Web (http://www.apache.org/).

Como já visto, o TABIPRO é composto por dois módulos desenvolvidos na linguagem de programação PHP 4, que pode ser obtida por download no site http://www.php.net. Para escrever o código fonte destes dois módulos, pode ser utilizado qualquer editor de texto.

Tanto o módulo Mentor quanto o módulo Aluno interagem com o MySQL 3.23.36, que é o gerenciador de banco de dados onde ficam armazenados os protótipos das tabelas, podendo também ser obtido por download através do site http://www.mysql.com.

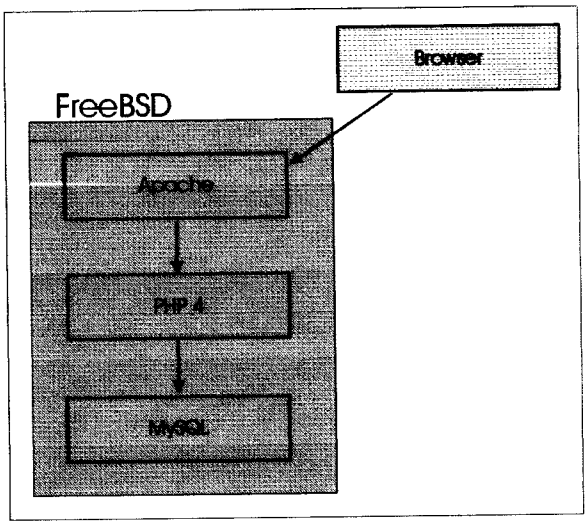


Figura 28 - O ambiente operacional do TABIPRO.

12.1. A Estrutura dos Protótipos

O protótipo de uma tabela criada pelo mentor, se desdobra em três tabelas na estrutura do banco de dados do TABIPRO que são: header, _col e _lin. A header é uma tabela comum a todos os protótipos criados, tendo em cada um dos seus registros informações de um protótipo. Já as tabelas _col e _lin têm como conteúdo

dados exclusivos de um protótipo. Se houver cinco protótipos criados, haverá uma tabela header com cinco registros, cinco tabelas _col e cinco tabelas _lin. Na realidade o nome das tabelas _col e _lin são precedidos pelo nome do protótipo definido pelo mentor ao criá-lo.

No MySQL foi criado um banco de dados de nome tabelas, onde ficam a tabela header e todas as tabelas _col e _lin. A estrutura destas tabelas e a descrição de cada um dos campos estão mostradas a seguir:

12.1.1. Tabela header

Cada registro da tabela header é composto pelos campos:

- nome_tab, identifica o protótipo da tabela dentro do TABIPRO, deve ser de uma única palavra com tamanho máximo de 10 caracteres.
- titu_tab, onde é armazenado um título para a tabela a ser utilizada no módulo Aluno, com tamanho máximo de 100 caracteres.
- nro_linh, aponta o número de linhas da tabela.
- nro_colu, aponta o número de colunas da tabela.
- perc_tol, indica o valor percentual de tolerância para erro, a ser utilizado sempre que no conteúdo de uma célula houver fórmulas matemáticas.
- respmail, para armazenar o endereço eletrônico (e-mail) do mentor ou responsável por receber as réplicas das tabelas preenchidas pelos alunos, tendo tamanho máximo de 100 caracteres.

Field	Type	Null	Key	Default	Extra
nome_tab	varchar(10)		PRI		
titu_tab	varchar(100)	YES		NULL	
nro_linh	smallint(6)	YES		NULL	
nro_colu	smallint(6)	YES		NULL	
perc_tol	float	YES		NULL	
respmail	varchar(100)	YES		NULL	

Tabela 6 - Estrutura da tabela header no MySQL.

12.1.2. Tabela _col

Todo protótipo criado gera uma tabela _col, que conterá em seus registros informações de cada coluna da tabela. Se o protótipo a ser criado possuir 3 colunas, então haverá três registros na tabela _col, com os campos:

- nume_col, identifica o número da coluna na tabela.
- titu_col, é alimentado automaticamente pelo módulo Mentor.
- desc_col, contém o título da coluna da tabela, com tamanho máximo de 100 caracteres.

Field	Type	Null	Key	Default	Extra
nume_col	smallint(6)		PRI		
titu_col	varchar(10)				
desc_col	varchar(100)	YES		NULL	

Tabela 7 - Estrutura da tabela _col no MySQL.

12.1.3. Tabela _lin

Assim como na _col, cada protótipo criado também gera uma tabela _lin, que conterá em seus registros informações de todas as linhas da tabela detalhadas célula por célula. Se o protótipo a ser criado possuir 5 linhas, então haverá 5 registros na tabela _lin, com os campos:

- nume_lin, identifica o número da linha na tabela.
- L0C1, identifica a primeira coluna dentro da linha e armazenará o conteúdo da célula representada pela linha nume_lin com a coluna 1.
- L0C2, identifica a segunda coluna dentro da linha e armazenará o conteúdo da célula representada pela linha nume_lin com a coluna 2.
- L0Cn, identifica a enésima coluna dentro da linha e armazenará o conteúdo da célula representada pela linha nume_lin com a coluna n.

Field	Type	Null	Key	Default	Extra
nume_lin	smallint(6)		PRI		
L0C1	varchar(100)	YES		NULL	
L0C2	varchar(100)	YES		NULL	
. . .					
L0C n	varchar(100)	YES		NULL	

Tabela 8 - Estrutura da tabela _lin no MySQL.

Abaixo está a representação de como fica armazenado no MySQL, o conteúdo dos registros das três tabelas em que um protótipo é desmembrado. Os dados exibidos são do protótipo da tabela Desloc criada na seção 10.6.

Tabela header

nome_tab	titu_tab	nro_linh	nro_colu	perc_tol	respmail
....	....	....	....	....	....
Desloc	Deslocamento em Linha Reta	6	3	0	sergio@rc.unesp.br
....	....	....	....	....	....

Tabela Desloc_col

nume_col	titu_col	desc_col
1	L0C1	Dini
2	L0C2	Dfin
3	L0C3	Desloc

Tabela Desloc_lin

nume_lin	L0C1	L0C2	L0C3
1			$L1C2 - $L1C1
2			$L2C2 - $L2C1
3			$L3C2 - $L3C1
4			$L4C2 - $L4C1
5			$L5C2 - $L5C1

12.2. Considerações Finais

Ao final deste capítulo, é possível que uma pessoa experiente em administrar o sistema operacional FreeBSD, instale os softwares necessários, monte a estrutura do banco de dados, adeque o ambiente operacional e coloque o TABIPRO em operação.

13. Conclusões

13.1. Considerações iniciais

A educação à distância se operacionaliza através de uma tecnologia que serve como o canal de comunicação entre o professor e o aluno. A Internet é tida como bom ambiente para aplicação desta modalidade de ensino, porque possui potencial de comunicação e interação.

Mostrando a importância da educação para o progresso dos Estados Unidos, Greenspan diz que capital humano – os professores – e tecnologia se complementam na produção de resultados educacionais, da mesma forma que em outras atividades (Greenspan, 2000 apud Leite, 2002).

13.2. Contribuições

Evolução na qualidade e praticidade em relação às tabelas anteriores do Módulo de Mecânica Gráfica do Programa Educ@r era esperado pela equipe do CDCC, quando foi feita a proposta de desenvolver este projeto.

Com o desenvolvimento do software TABIPRO, de um modo geral o mentor tem maior controle sobre o formato e conteúdo das tabelas que compõem o módulo

de Mecânica Gráfica, sem que conheça a linguagem de programação em que foi desenvolvido.

Além das condições técnicas favoráveis aos propósitos deste projeto que a linguagem de programação PHP oferece, outro fator foi decisivo para que esta se tornasse a ferramenta escolhida para o desenvolvimento do TABIPRO, a facilidade para programar. Pensando na herança a ser deixada aos técnicos de informática do CDCC, no que diz respeito a futuras manutenções do código fonte, foi interessante escolher uma linguagem simples para programar, porém, com características que atenda as necessidades do projeto.

Usando o potencial da linguagem de programação PHP 4 para desenvolvimento de aplicações para a Web, o TABIPRO de forma bastante simples, permite ao mentor criar tabelas customizadas, utilizando fórmulas matemáticas que possam checar em tempo real os cálculos realizados pelo aluno no preenchimento das mesmas. As montagens das tabelas que nas versões anteriores levaram meses, passam a ser realizadas em poucas horas de trabalho.

Uma característica importante do TABIPRO é a sua independência em relação às plataformas operacionais do mentor, do aluno e do próprio curso em que é inserido, tendo assim potencial para ser explorado em inúmeras outras aplicações que envolvam o formato de tabela.

A inexistência de manutenção do código fonte para inserção, exclusão e alteração de tabelas, corrigiu o principal problema das versões anteriores das tabelas utilizadas no Mecânica Gráfica.

A fácil interação aluno/software e a interatividade proporcionada pelo TABIPRO entre aluno e mentor são premissas que permitem a este software ser

utilizado em ambiente de ensino à distância, propiciando *feedback* oportuno que estimula o desenvolvimento do aluno dentro do curso que está fazendo.

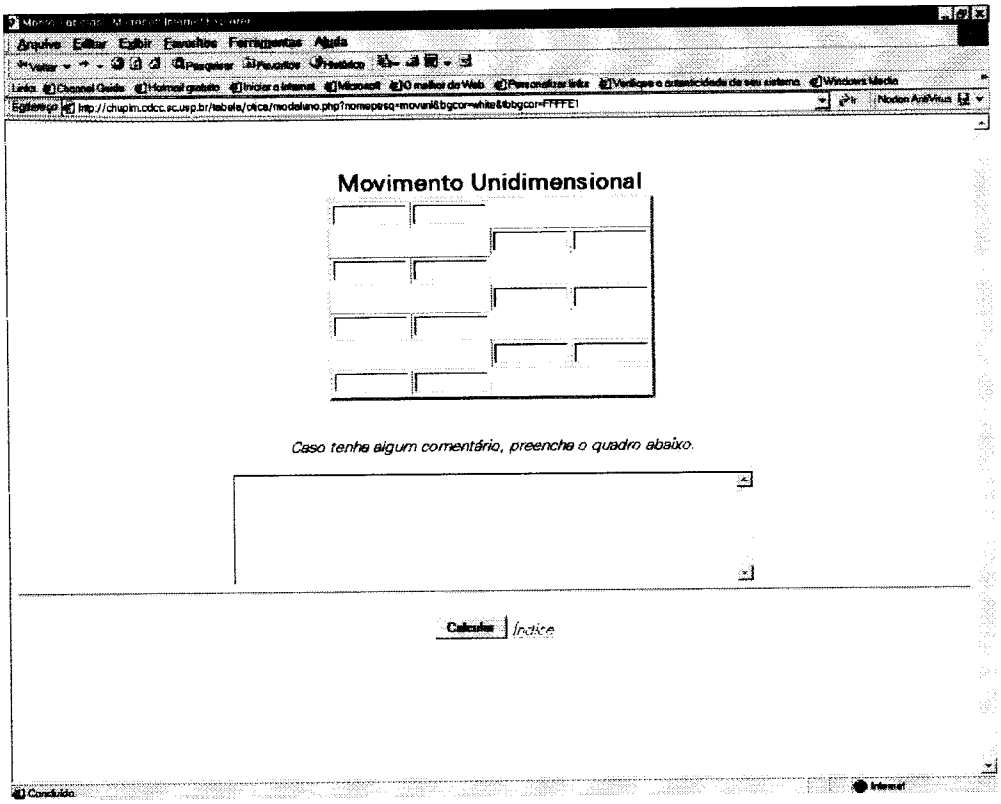
Espera-se com este trabalho, que o TABIPRO venha a ser utilizado em muitas outras aplicações, que sua metodologia possa ser largamente utilizada em softwares similares que venham a ser desenvolvidos, primando sempre pelos quesitos interatividade e customização, e que os resultados aqui alcançados possam significar contribuições efetivas para utilização dos recursos computacionais na educação à distância.

Acredito assim, que a expectativa inicial da equipe do CDCC tenha sido alcançada pelo produto desenvolvido neste trabalho.

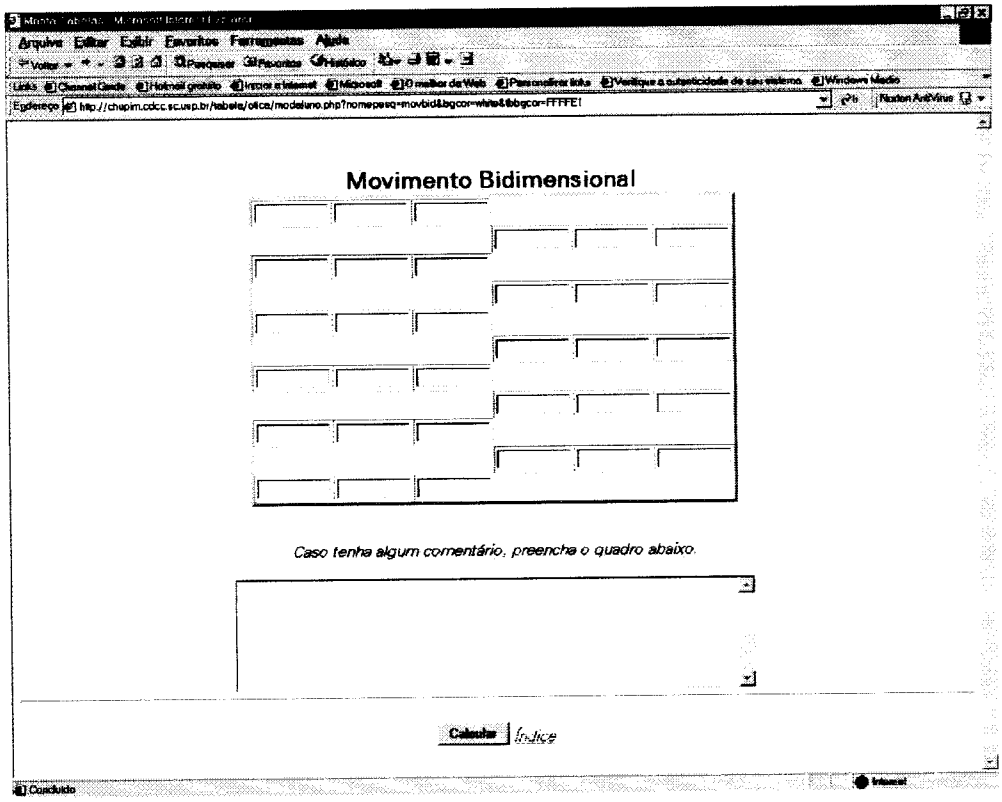
13.3. Sugestões para Trabalhos Futuros

Como continuidade para o desenvolvimento do TABIPRO, é interessante a criação de um módulo administrativo que permita armazenar informações sobre o mentor, sobre o aluno, sobre os cursos e projetos em que está sendo utilizado, possibilitando maior controle da utilização deste software no ambiente em que está operando.

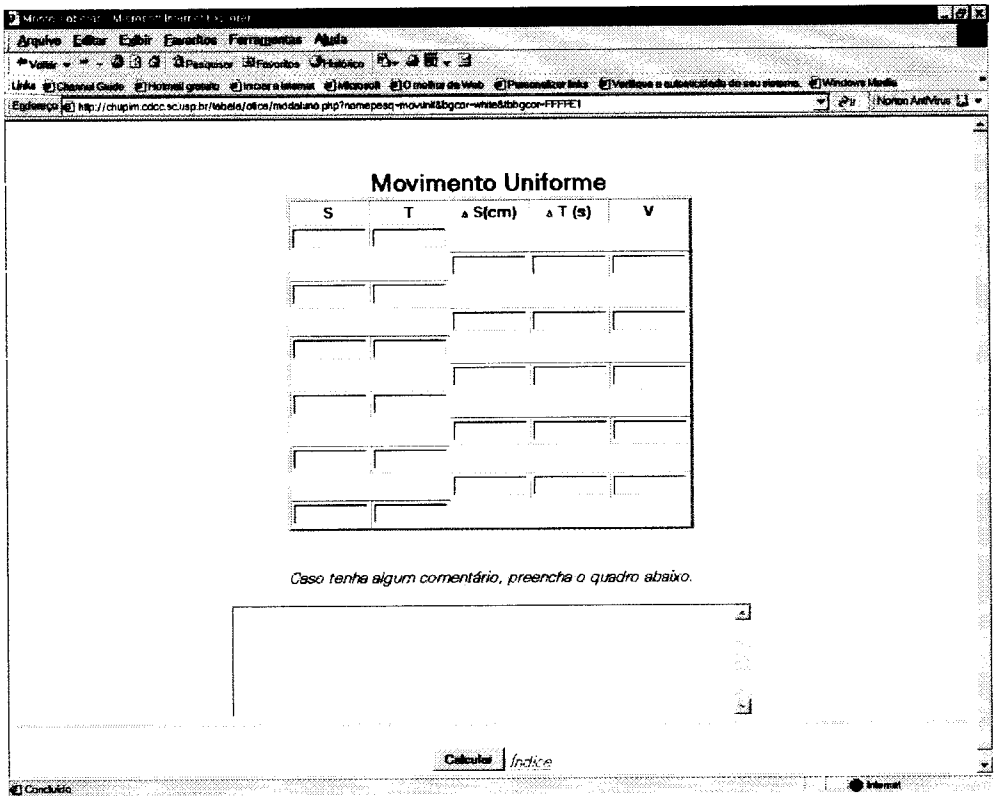
Anexo



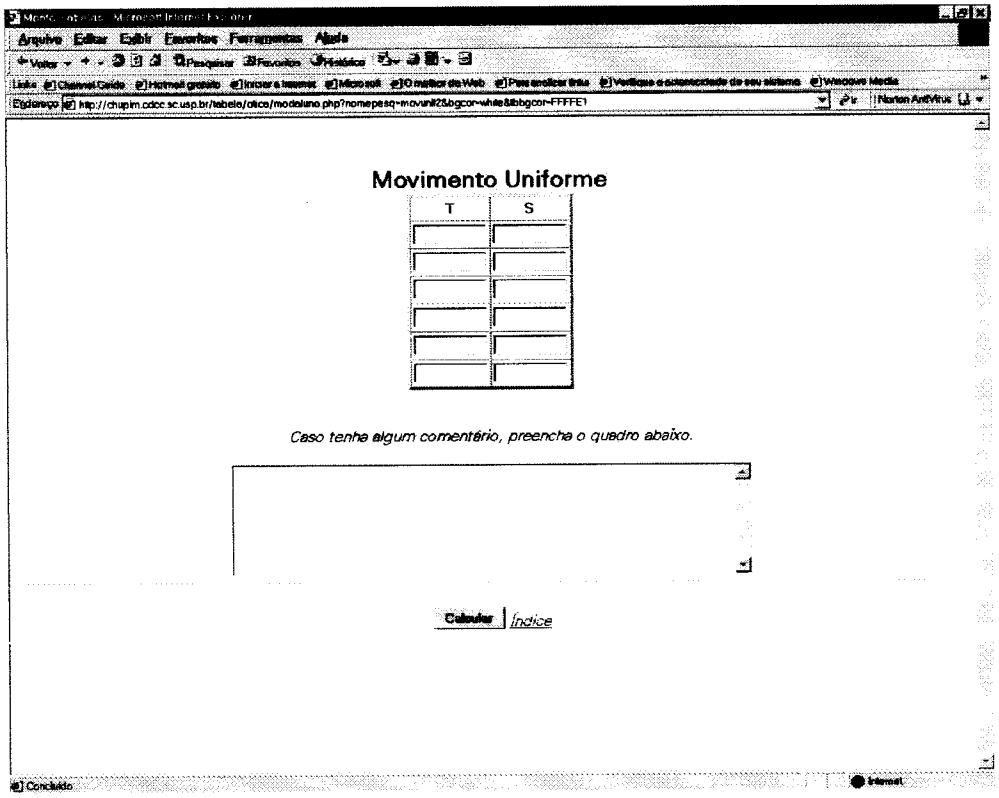
Mecânica Gráfica. Tabela 1.1 - Movimento unidimensional



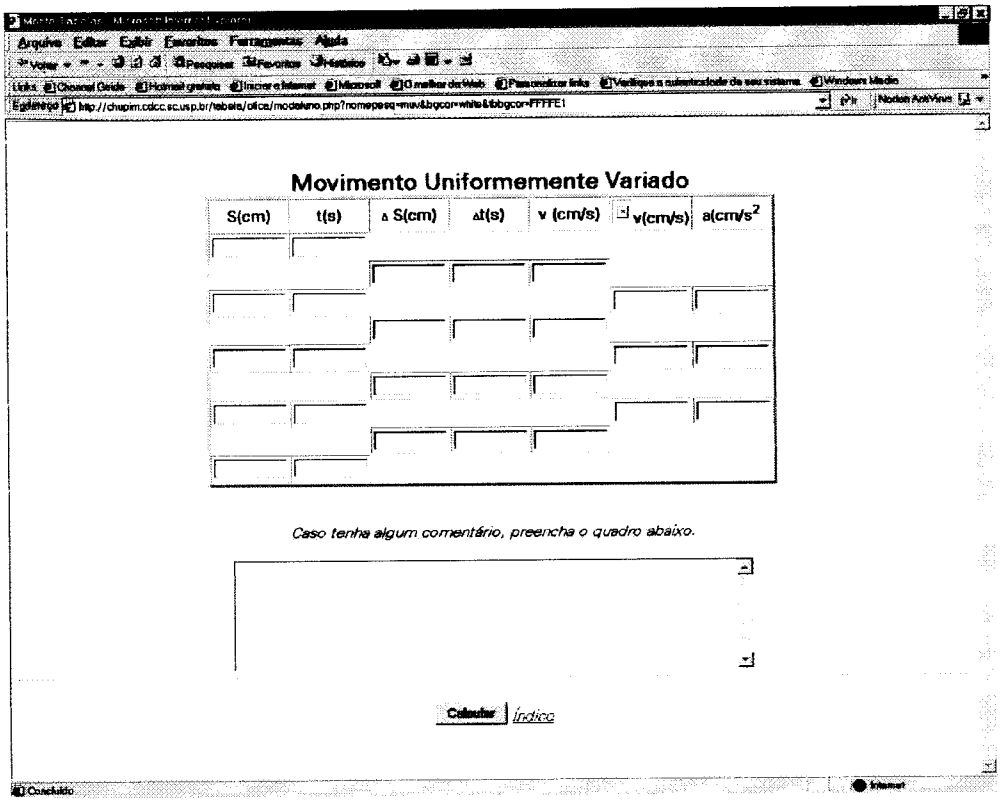
Mecânica Gráfica. Tabela 1.2 - Movimento bidimensional



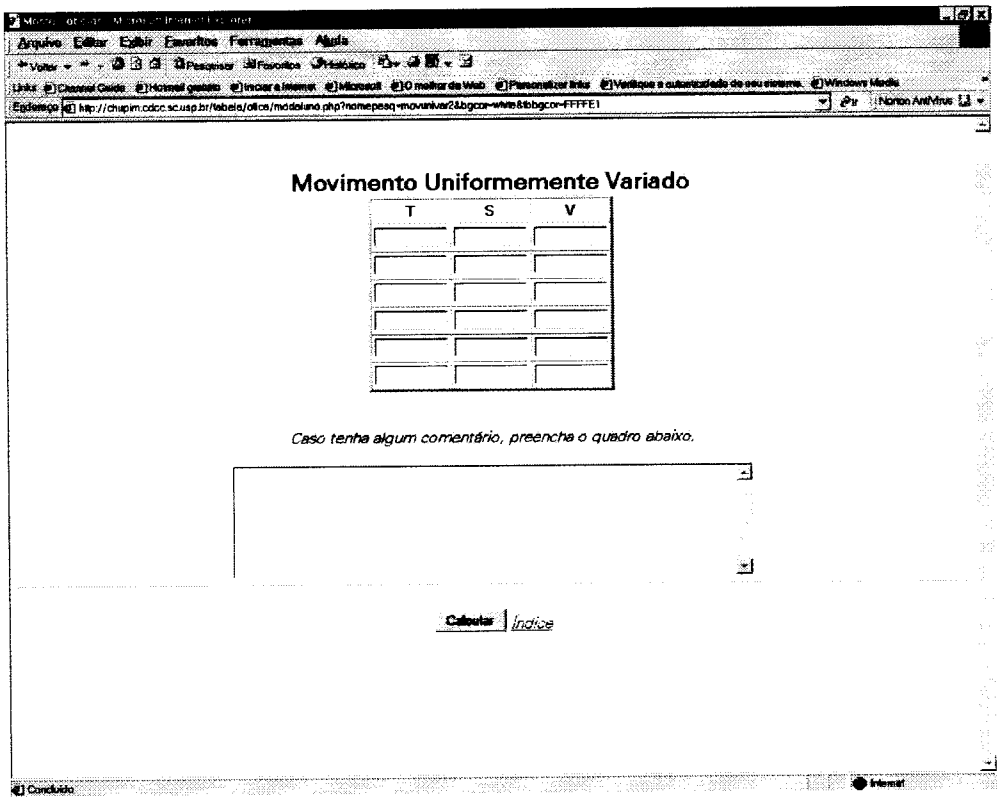
Mecânica Gráfica. Tabela 2.1 - Movimento uniforme



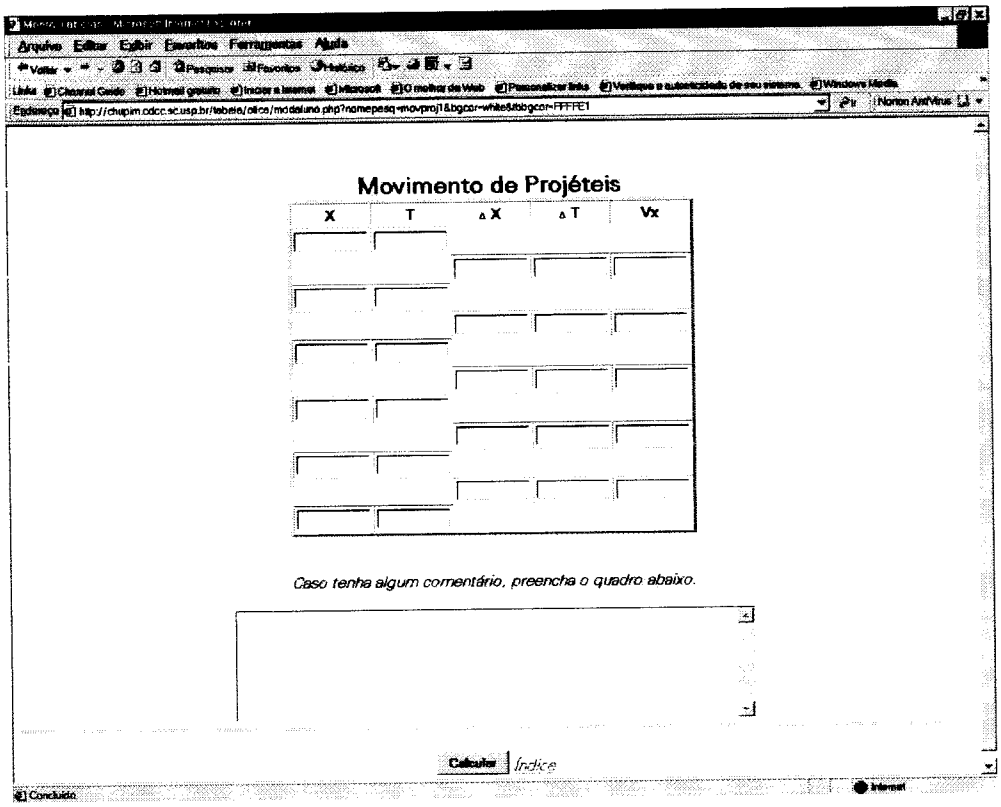
Mecânica Gráfica, Tabela 2.2 - Movimento uniforme



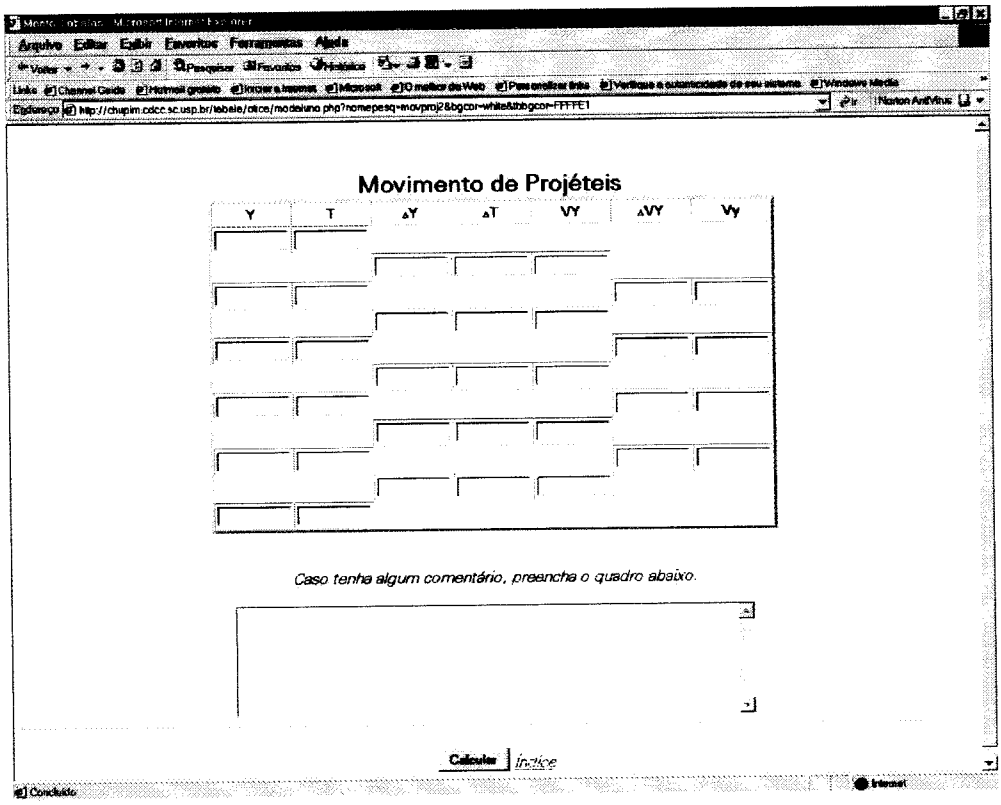
Mecânica Gráfica. Tabela 3.1 – Movimento uniformemente variado



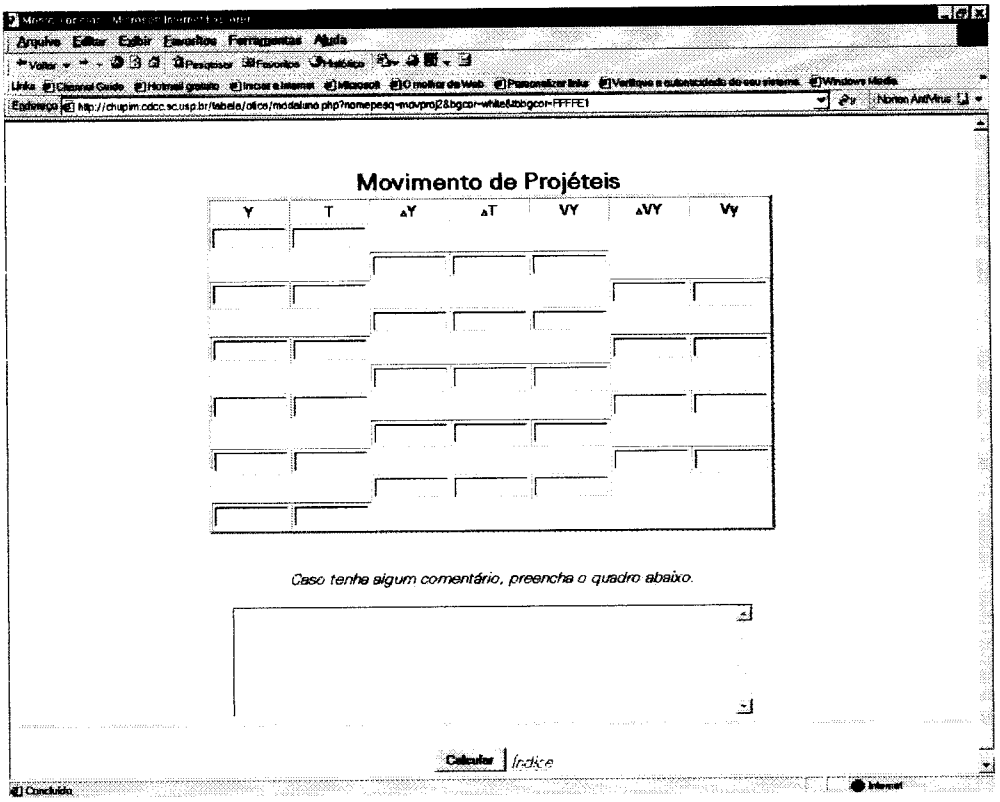
Mecânica Gráfica. Tabela 3.2 – Movimento uniformemente variado



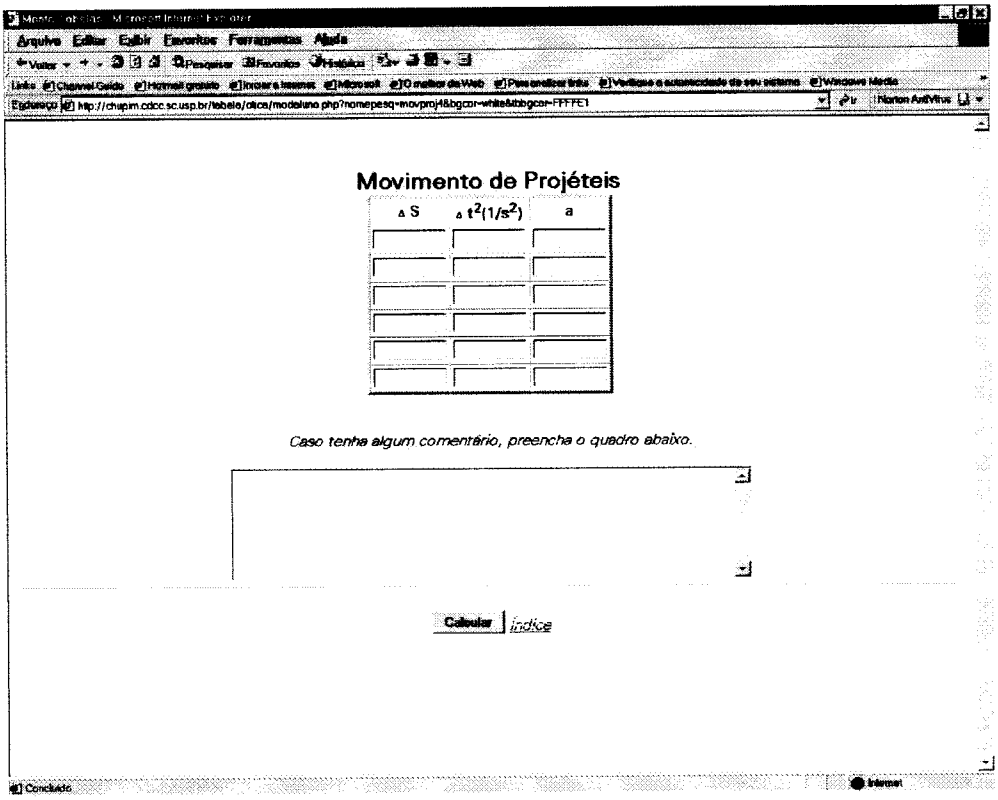
Mecânica Gráfica, Tabela 4.1 – Movimento de projéteis



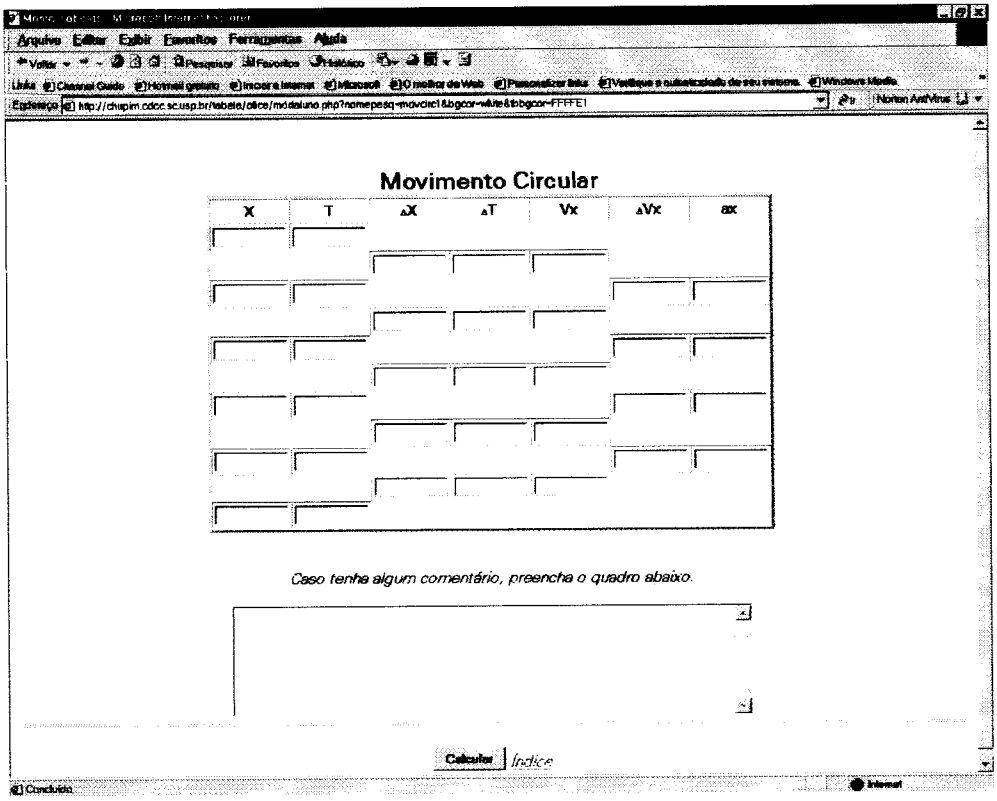
Mecânica Gráfica, Tabela 4.2 – Movimento de projéteis



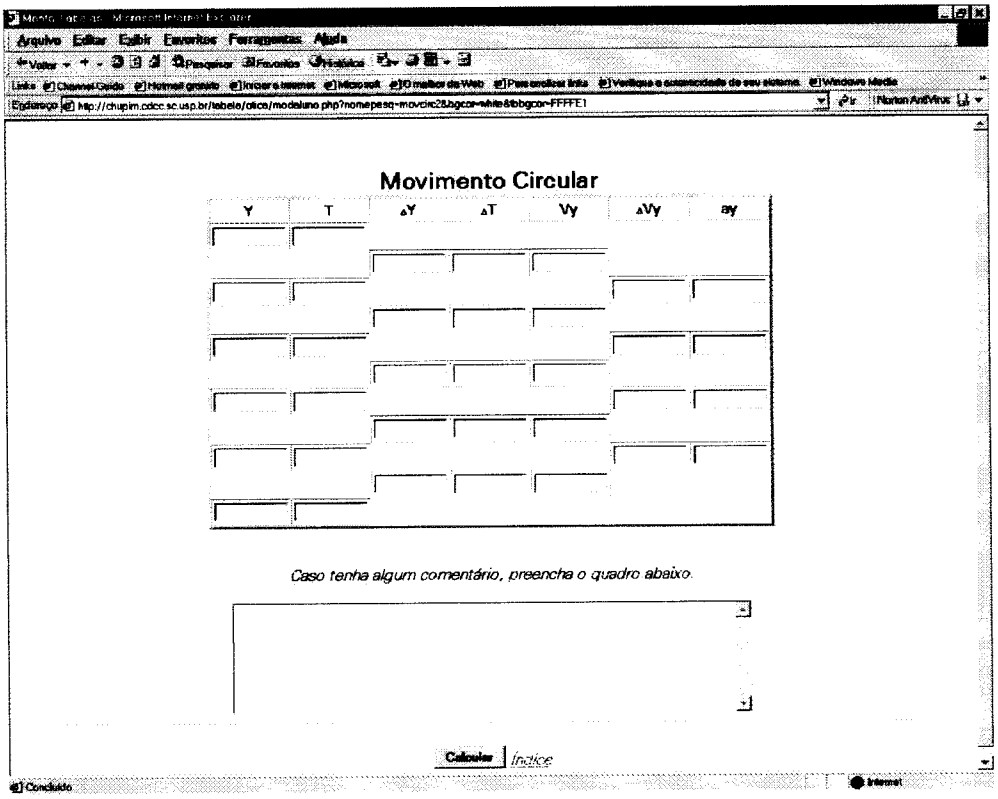
Mecânica Gráfica. Tabela 4.3 – Movimento de projéteis



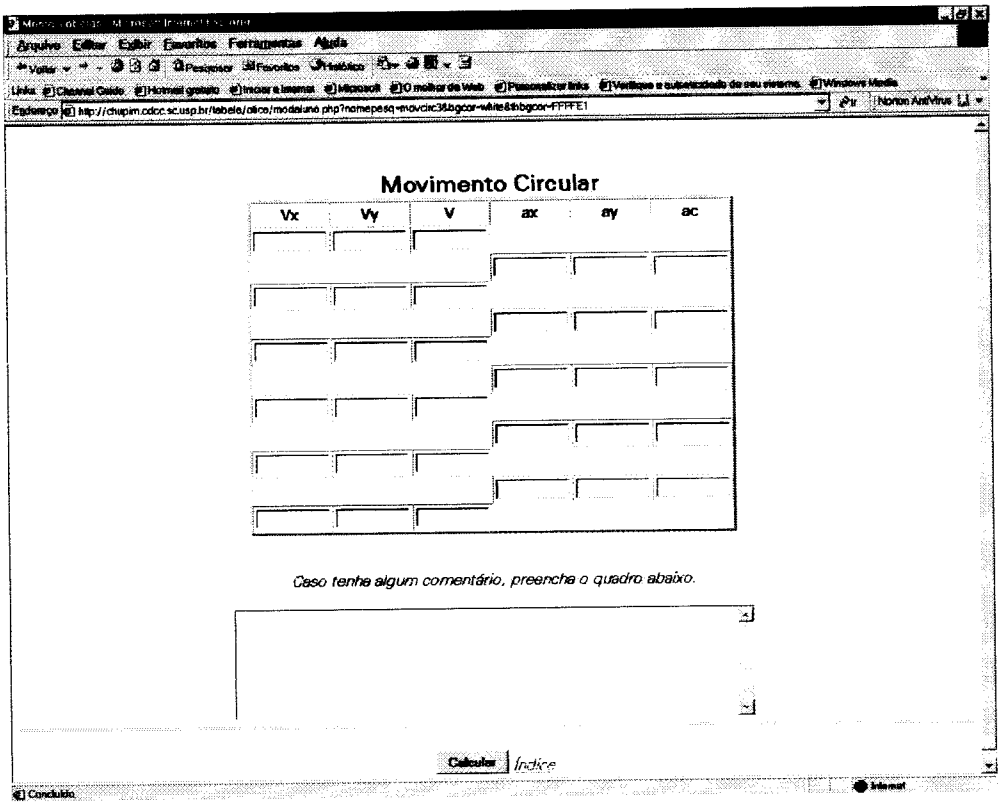
Mecânica Gráfica, Tabela 4.4 – Movimento de projéteis



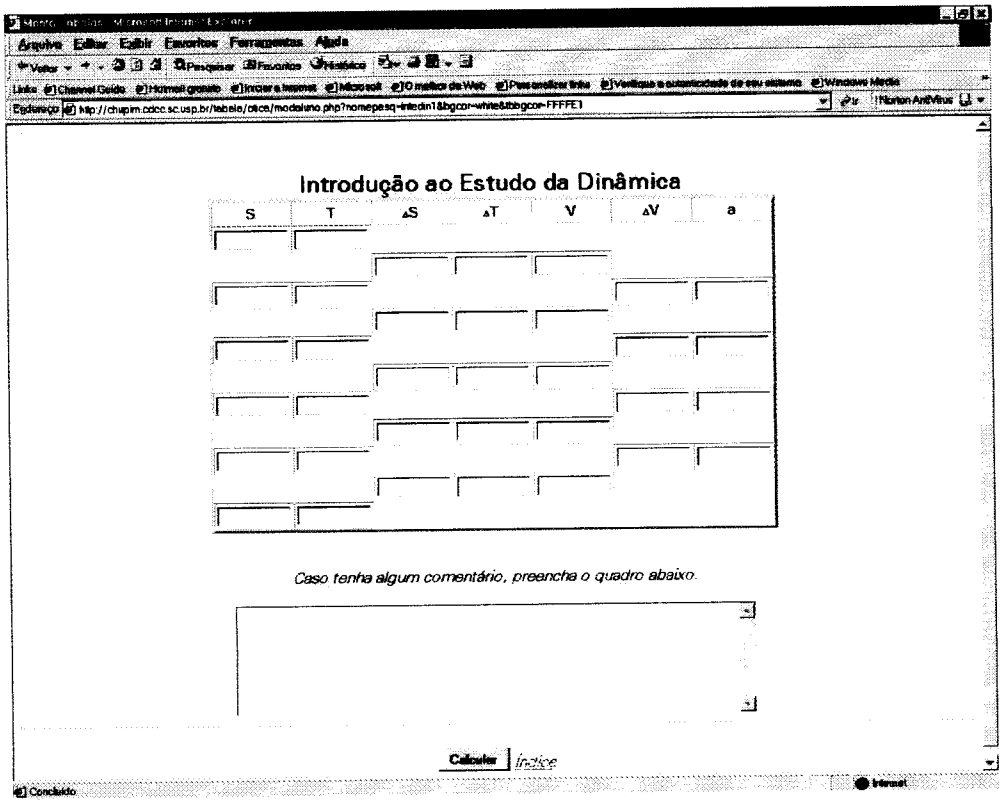
Mecânica Gráfica. Tabela 5.1 – Movimento circular



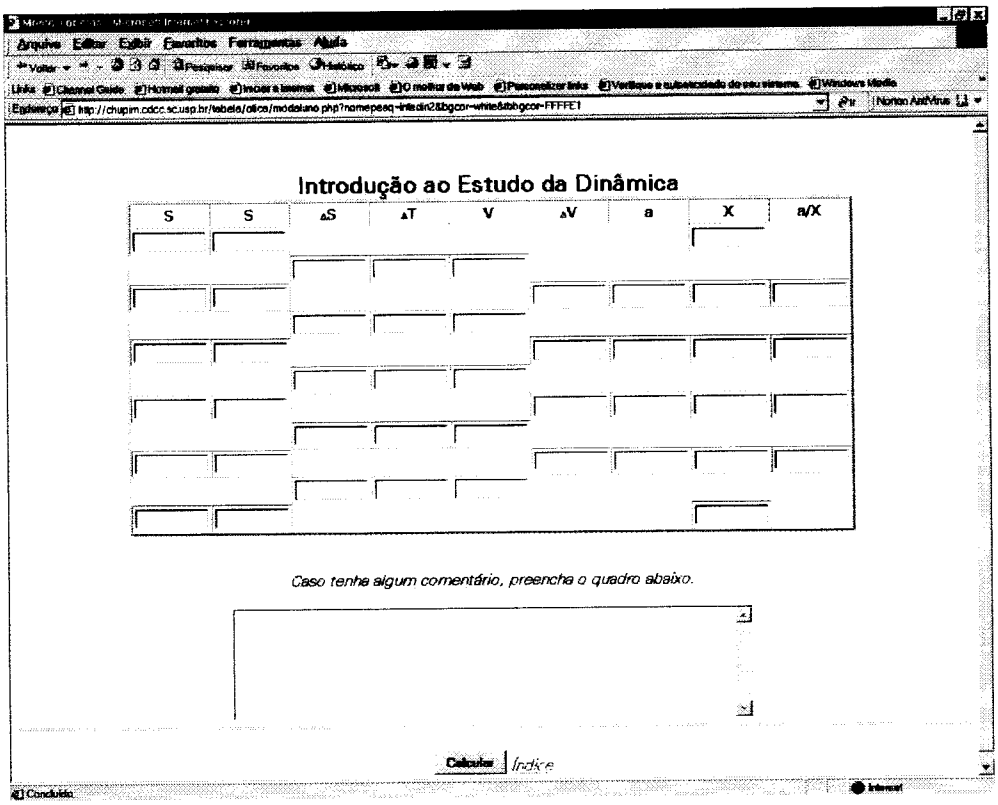
Mecânica Gráfica. Tabela 5.2 – Movimento circular



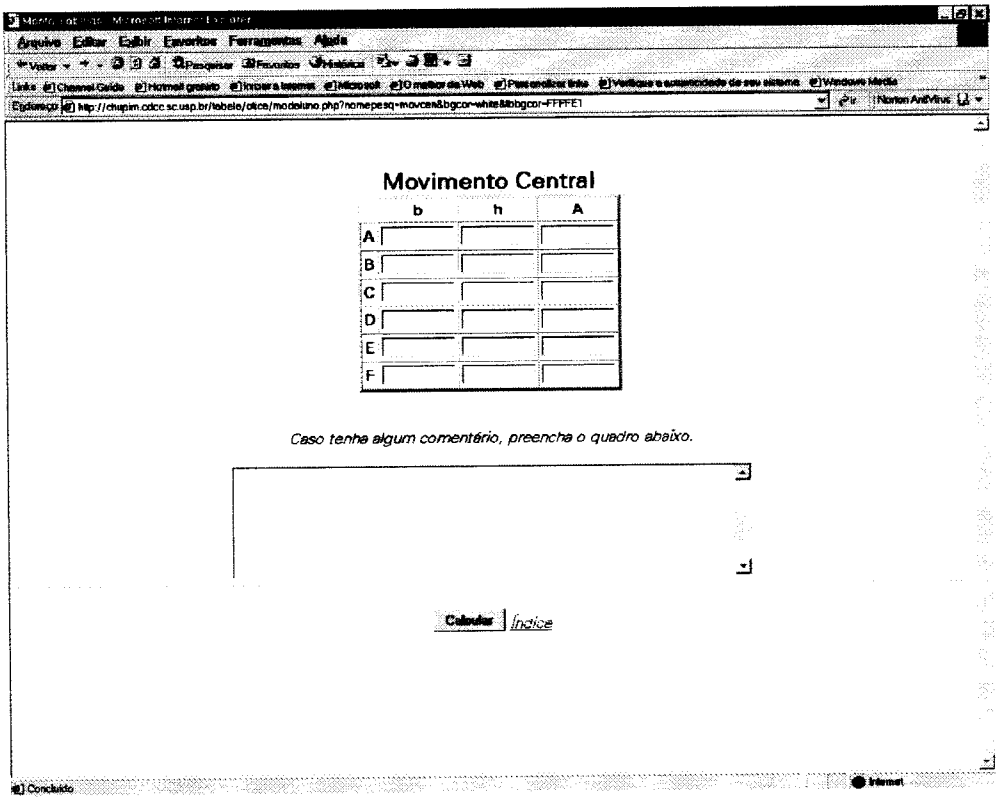
Mecânica Gráfica. Tabela 5.3 – Movimento circular



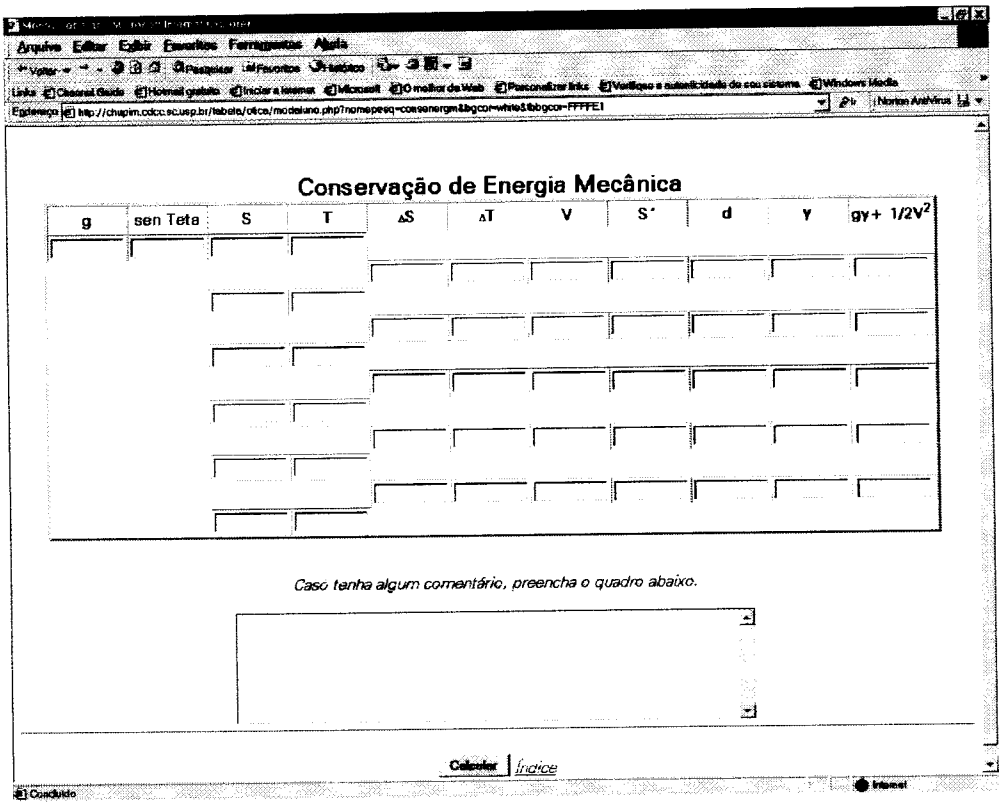
Mecânica Gráfica, Tabela 6.1 – Introdução ao estudo da mecânica



Mecânica Gráfica. Tabela 6.2 – Introdução ao estudo da mecânica



Mecânica Gráfica. Tabela 7.1 – Movimento central



Mecânica Gráfica, Tabela 8.1 – Conservação de energia mecânica

Referências Bibliográficas

(Allaire, 2002) ALLAIRE, J. *Macromedia MX*: Componentes e serviços Web. Acessado em 2002.
http://www.macromedia.com/br/desdev/documents/component_pt.pdf.

(Alves, 1996) ALVES, J.R.M. Educação à distância e as novas tecnologias de informação na aprendizagem. In: TELECONFERÊNCIA ENGENHEIRO 2001, 7., São Paulo, 8 outubro, 1996. p.5-25.

(Andrade & Beiler, 1999) ANDRADE, A.F.; BEILER,A. Análise de Ferramentas Computacionais Colaborativas Visando Aprendizagem à Distância. In: TALLER INTERNACIONAL DE SOFTWARE EDUCATIVO, Santiago, Chile, 1999.

(Bakken et al, 2000) BAKKEN, S.S.; AULBACH, A.; SCHMID, E., WINSTEAD, J.; WILSON, L.T.; LERDORF, R.; SURASKI, Z. *PHP Manual*. 2000.
http://www.phpbuilder.com/manual.

(Bakken et al, 2002) BAKKEN, S.S.; AULBACH, A.; SCHMID, E.; WINSTEAD, J.; WILSON, L.T.; LERDORF, R.; ZMIEVSKI A.; AHTO, J. *PHP Manual*: Introduction. Acessado em 2002. http://www.php.net/manual/en/ref.math.php.

(Bakken et al, 2002a) BAKKEN, S.S.; AULBACH, A.; SCHMID, E.; WINSTEAD, J.; WILSON, L.T.; LERDORF, R.; ZMIEVSKI A.; AHTO, J. *PHP Manual*: Mathematical Functions. Acessado em 2002.
http://www.php.net/manual/en/introduction.php.

(Bakken et al, 2002b) BAKKEN, S.S.; AULBACH, A.; SCHMID, E.; WINSTEAD, J.; WILSON, L.T.; LERDORF, R.; ZMIEVSKI A.; AHTO, J. *PHP Manual*: What can PHP do?. Acessado em 2002. http://www.php.net/manual/en/intro-whatcando.php.

(Bakken et al, 2003) BAKKEN, S.S.; AULBACH, A.; SCHMID, E.; WINSTEAD, J.; WILSON, L.T.; LERDORF, R.; ZMIEVSKI A.; AHTO, J. *PHP Manual*: Operators. Acessado em 2003. http://www.php.net/manual/en/language.operators.php.

(Brooks, 1997) BROOKS, D.W. *Web-Teaching*: a guide to designing interactive teathing for the World Wide Web. New York, Plenum Press, 1997.

(Cammy & Botelho, 2002) CAMY, A.R.; BOTELHO, W.T. *Sistemas Multiagentes Cognitivos*: Modelando um Protótipo de Comércio Eletrônico B2C. Universidade Católica Dom Bosco, junho, 2002 (Relatório Técnico).

(Castro, 1997) CASTRO, C. de M. *Education in the Information Age:* what works and what doesn't. New York, Felipe Herrera Library, 1997.

(CDCC, 2002) CENTRO DE DIVULGAÇÃO CIENTÍFICA E CULTURAL. *Programa Educ@r*. Universidade de São Paulo. São Carlos. Acessado em 2002.
http://educar.sc.usp.br/historico.html.

(CDCC, 2002a) CENTRO DE DIVULGAÇÃO CIENTÍFICA E CULTURAL. *Programa Educ@r*. Universidade de São Paulo. São Carlos. Acessado em 2002. http://educar.sc.usp.br/historico.html#mecanica.

(CDCC, 2002b) CENTRO DE DIVULGAÇÃO CIENTÍFICA E CULTURAL. *Programa Educ@r*. Universidade de São Paulo. São Carlos. Acessado em 2002. http://educar.sc.usp.br/historico.html#centro.

(Cornell & Martin, 1997) CORNELL, R.; MARTIN, B. L. The Role of motivation in Web-Based Instruction. In: KHAN, B. H., ed. *Web-Based instruction*. Englewood Cliffs, Educational Technology Publications Inc., 1997. p. 93-100.

(Cox Jr, 2000) COX Jr. F. *Programando para Web com PHP/MySQL* Escola Politécnica de Pernambuco, Pernambuco, 2000. http://www.superphp.com.br/apostilas/index.php?secao=1.

(Devices, 2002) DEVICES ELETRONIC MEASUREMENT. *Telecomunicações e Informática*. Acessado em 2002. http://www.deviceselectronic.com.br/abctelecom.htm.

(Dillenbourg & Schneider, 1995) DILLENBOURG, P.; SCHNEIDER, D. Collaborative learning and the Internet. In: INTERNATIONAL CONFERENCE ON COMPUTER ASSISTED INSTRUCTION. Hsinchu, Taiwan, 1995. http://tecfa.unige.ch/tecfa/research/CMC/colla/iccai95_1.html.

(Ferreira, 1993) FERREIRA, A.B. de H. *Dicionário Aurélio da Língua Portuguesa*. Rio de Janeiro, Nova Fronteira, 1993.

(Forcier, 1999) Forcier, R. C. *The Computer as an Educational Tool*: productivity and problem solving. 2nd ed. Upper Saddle River, New Jersey, Prentice-Hall, Inc., 1999.

(Franco, 2001) FRANCO, M.A. A escrita para EAD. *Boletim EAD*, Universidade de Campinas, 31 de Outubro de 2001. http://www.ead.unicamp.br.

(Gomes et al, 2000) GOMES, R. de C.G.; RODRIGUES, R.S.; DAMEZ, L.; BARCIA, R.M.. Comunicação Multidirecional – Um Ambiente de Aprendizagem na Educação a Distância. In: VII CONGRESSO INTERNACIONAL DE EDUCAÇÃO A DISTÂNCIA. São Paulo, 2000. http://www.abed.org.br/congresso2000/texto09.htm

(Gonçalves, 1996) GONÇALVES, C.T.F. Quem tem medo do ensino a distância *Educação a Distância*. INED/IBASE, n.7-8, 1996. http://www.intelecto.net/ead_textos/consuelo.html.

(Greenspan, 2000) GREENSPAN, A. The Back Page. *Technology & Learning*. V.21(3), p.78, Oct 2000.

(Hahn, 1995) HAHN, H.; STOUT, R. *Dominando a Internet*. Trad. de Antonio Augusto Orselli. São Paulo, Makron Books, 1995.

(Hermann, 1997) HERMAN, E. *Aprenda em 1 semana CGI com PERL 5*. Trad. de Geraldo Costa Filho. Rio de Janeiro, Campus, 1997.

(Hessel & Schiel, 1998) HESSEL, R.; SCHIEL, D. Discos Sustentados por Colchão de Ar: Uma Nova Proposta. In: NARDI, R. org. *Pesquisa em ensino de Física*. São Paulo, Escrituras Editora, 1998. p.95-103.

(Hill, 1997) Hill, J.R. Distance Learning Environments Via the World Wide Web. In: KHAN B. H., ed. *Web-Based Instruction*. Englewood Cliffs, Educational Technology Publications, Inc., 1997. p.75-80.

(Huggins, 1979) HUGGINS, E. *Graphical Mechanics*. Wentworth COMPress, Inc., Dartmouth College. 1979.

(Ibrahim, 1994) IBRAHIM, B. *Distance learning with the World-Wide Web*. In: INTERNATIONAL CONFERENCE ON "OPEN AND DISTANCE LEARNING - CRITICAL SUCCESS FACTORS", Geneva, 10-12 October, 1994.

(Jaques et al, 2000) JAQUES, P.A.; ANDRADE, A.; MORAES, M.C..; MORA, M. C. Uma Arquitetura de Agente para a Análise Qualitativa da Interação em Ambientes de Educação à Distância. In: XI SIMPÓSIO BRASILEIRO DE INFORMÁTICA NA EDUCAÇÃO – SBIE. Maceió, Alagoas, novembro de 2000.

(Jonhson & Jonhson, 1996) JOHNSON, D.W.; JONHSON, R.T. Cooperation and the Use of Technology. In: JONASSEN, D. H., ed. *Handbook of research for educational communications and technology*: a project of the Association for Educational Communications and Technology. New York, Simon & Schuster Macmillan, 1996. p.1017-1044.

(Jonassen, 1992) JONASSEN, D.H. Applications and limitations of hypertext technology for distance learning. In: DISTANCE LEARNING WORKSHOP, Armstrong Laboratory, San Antonio, Texas, 1992.

(Jonassen & Reeves, 1996) JONASSEN, D.H.; REEVES, T.C. Learning with Technology: using computers as cognitive tools. In: JONASSEN, D. H., ed. *Handbook of research for educational communications and technology*: a project of the Association for Educational Communications and Technology. New York, Simon & Schuster Macmillan, 1996. p.693-719.

(Kearsley, 1995) KEARSLEY, G. The nature and value of interaction in distance learning. In: DISTANCE EDUCATION RESEARCH SYMPOSIUM, 18-21 Maio, 1995. http://www.gwu.edu/~etl/interact.html.

(Keegan, 1986) KEEGAN, D. *The foundations of distance education*. London, Croom Helm, 1986.

(Keeler & Burkman, 1993) KEELER, J.; BURKMAN, E. Motivation principles. In: FLEMING, M. & LEVIE, W. H. eds. *Instructional message design*: Principles from the behavioral and cognitive sciences. Englewood Cliffs, New Jersey, Educational Technology Publications Inc., 1993. p. 3-53.

(Lasser, 1997) LASSER, W. *Manual de criação e elaboração de materiais para educação a distância*. Brasília, CEAD e Editora Universidade de Brasília, 1997.

(Leite, 2002) LEITE, L.S. *Temas e Questões Estratégicas para o Desenvolvimento da EAD*. Intertexto - Gestão da Informação, Estudos e Projetos. Acessado em 2002. http://www.intelecto.net/desenvolvimento_ead.htm.

(Lifschitz, 1998) LIFSCHITZ, S.; LIMA, I.N. Arquiteturas de Integração Web SGBD: um Estudo do Ponto de Vista de Sistemas de Banco de Dados. IN XXV SEMINÁRIO INTEGRADO DE SOFTWARE E HARDWARE, Belo Horizonte, MG, Agosto 1998. http://sites.uol.com.br/iremar.prof/publicacoes/semish98html/semish98.htm.

(Lima, 1997) LIMA, I.N. de. *O Ambiente Web Banco de Dados*: Funcionalidades e Arquiteturas de Integração. Rio de Janeiro, 1997. Dissertação (Mestrado) - Departamento de Informática, PUC.

(Lucena, 1997) LUCENA, J.C..P. de: *Curso sobre Sociedade da Informação*. Acessado em 2002. http://www.les.inf.puc-rio.br/socinfo.

(Magalhães, 1997) MAGALHÃES, M.G. de M. de. *Estudo e avaliação de Educação à Distância utilizando a tecnologia WWW*. São Carlos, 1997. Dissertação (Mestrado) - Instituto de Física de São Carlos, Universidade de São Paulo.

(Magalhães et al, 2002) MAGALHÃES, M.G. M.; SCHIEL, D; GUERRINI, I.M.; MAREGA Jr, E. Utilizando Tecnologia Computacional na Análise Quantitativa de Movimentos: Uma Atividade para Alunos do Ensino Médio. *Rev. Bras. de Ensino de Física*, 24, 2, p.97-102, 2002.

(Martin & Bramble, 1996) Martin, B. M.; Bramble, W. J. Designing Effective Video Teletraining Instruction: The Florida Teletraining Project. *Educational Technology Research and Development*, 44(1), p.85-89. 1996.

(Martins et al, 2001) MARTINS, A. da R.; CARNEIRO, M.L.F.; FABRE, M.J.M.; KELLER, R.dos S. O Suporte em Educação a Distância. In: VIII CONGRESSO INTERNACIONAL DE EDUCAÇÃO A DISTÂNCIA, Brasília, 6 a 8 de agosto, 2001. http://www.abed.org.br/congresso2001/index.html.

(Matos, 2002) MATOS, H.A.B. *Ensino a Distância Suportado pela internet*. Universidade de Coimbra. Coimbra, Portugal. Acessado em 2002. http://eden.dei.uc.pt/~kikas/DL/.

(McIsaac & Gunawardena, 1996) MCISAAC, M.S.; GUNAWARDENA, C.N. Distance Education. In: JONASSEN, D. H., ed. *Handbook of research for educational communications and technology*: a project of the Association for Educational Communications and Technology. New York, Simon & Schuster Macmillan, 1996. p. 403-437.

(Medeiros et al, 2001) MEDEIROS, M. F. de.; MEDEIROS, G.M. de.; COLLA, A.L.; HERRLEIN, M.B.P. A Produção de um Ambiente de Aprendizagem em Educação a Distância com o Uso de Mídias Integradas: A Pucrs Virtual. In: VIII CONGRESSO

INTERNACIONAL DE EDUCAÇÃO À DISTÂNCIA. Brasília, 6 a 8 de agosto, 2001. http://www.abed.org.br/congresso2001/index.html.

(Medeiros, 2002) MEDEIROS E. L. de. *O que é PHP?*. SuperPHP. Acessado em 2002. http://superphp.alfa.8x.com.br/artigos/index.php?id=2.

(Moore & Kearsley, 1996) MOORE, M.G.; KEARSLEY, G. *Distance education*: a systems view. Belmont, California, Wadsworth Publishing, 1996.

(Moura et al, 2001) MOURA, A.M.M. de; AZEVEDO, A.M. P. de; MEHLECKE, Q. As Teorias de Aprendizagem e os Recursos da Internet Auxiliando o Professor na Construção do Conhecimento. In: VIII CONGRESSO INTERNACIONAL DE EDUCAÇÃO A DISTÂNCIA. Brasília, 6 a 8 de agosto, 2001. http://www.abed.org.br/texto10c2001.htm.

(MySQL AB, 2002) MySQL AB COMPANY. *MySQL Database Server*. 2002. http://www.mysql.com/products/mysql.

(MySQL AB, 2002a) MySQL AB COMPANY. *Fact Sheet*. 2002. http://www.mysql.com/company/factsheet.html.

(MySQL AB, 2002b) MySQL AB COMPANY. *The Main Features of MySQL*. 2002. http://www.mysql.com/doc/en/Features.html.

(Naves, 1998) NAVES, C.H.T. *Educação Continuada e a Distância de Profissionais da Ciência da Informação no Brasil via Internet*. Brasília, 1998. Dissertação (Mestrado) - Departamento de Ciência da Informação e Documentação, Universidade de Brasília.

(Oliveira et al, 2000) OLIVEIRA, E.H.T. de; SCAPIN, R.H.; VICENTINI, W.B. Proposta de um Ambiente Hipermídia Integrado para o Desenvolvimento de Cursos a Distância. In: VII CONGRESSO INTERNACIONAL DE EDUCAÇÃO À DISTÂNCIA. São Paulo, 13 a 18 de agosto, 2000. http://www.abed.org.br/texto30.doc.

(Orr & Appleton, 1997) ORR, D.; APPLETON, M. New opportunities for remote students. *Journal of Library Services for Distance Education*, v.1, n.1, agosto de 1997, State University of West Georgia, Carrollton. http://www.westga.edu/~library/jlsde/vol1/1/DOrr_MAppleton.html.

(Outing, 1998) Outing, S. What Exactly is Interactivity?. *Editor & Publisher Interactive*, Friday, December 4, 1998. http://www.mediainfo.com/ephome/news/newsshtm/stop/st120498.htm.

(Ricarte, 2000) RICARTE, I.L.M. *World-Wide Web*. DCA/FEEC/UNICAMP, 2000. http://www.dca.fee.unicamp.br/courses/PooJava/network/web.html.

(Romiszowski & Mason, 1996) ROMISZOWSKI, A.J.; MASON, R. Computer-Mediated Communication. In: JONASSEN, D. H., ed. *Handbook of research for educational communications and technology*: a project of the Association for Educational Communications and Technology. New York, Simon & Schuster Macmillan, 1996. p.438-456.

(Sá, 2000) SÁ, P.S.S. *Gerador Automático de Arquivos HTML de Ajuda para Aplicação em Educação a Distância (GAAHA).* São Carlos, 2000. Dissertação (Mestrado) - Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo.

(Sabariz & Barreto, 2002) Sabariz, A.; Barreto, P. *Migrando de Aulas Presenciais em um Curso de Engenharia para Educação Virtual Baseada na Internet.* Acessado em 2002. http://www.abed.org.br/texto25.doc.

(Santos et al, 2001) SANTOS, E. de F.G. dos; CRUZ, D.M.; PAZZETTO, V.T. Ambiente Educacional Rico em Tecnologia: A Busca Do Sentido. In: VIII CONGRESSO INTERNACIONAL DE EDUCAÇÃO A DISTÂNCIA. Brasília, 6 a 8 de agosto, 2001. http://www.abed.org.br/texto07c2001.htm.

(Schechtman, 2001) Schechtman, S. *Interatividade entre Orientador e Aprendiz em um Curso Virtual. Prática ou Teoria?* Rio de Janeiro, 2001. Monografia (Curso de Especialização) - Universidade Católica de Brasília.

(Schiel et al, 2002) SCHIEL, D.; DASSIN, J.; MAGALHÃES, M.G. de M. de., GUERRINI, I. M. High School Physics Instruction by Way of the World Wide Web: A Brazilian Case Study. *Journal of Interactive Learning Research (JILR)*, v.13, n. 4, 2002. p.293-309. Association for the Advancement of Computing in Education. Norfolk, VA, USA.

(Seligman, 1992) SELIGMAN, D. The comparative nature of quality: Distance education in the developing world. In: WORLD CONFERENCE OF THE INTERNATIONAL COUNCIL FOR DISTANCE EDUCATION. Bangkok, Thailand, Nov, 1992.

(Sherry, 1996) SHERRY, L. Issues in distance learning. *International Journal of Distance Education*, v.1, n.4, 1996. p.337-365.

(Sherry & Wilson, 1997) Sherry, L.; Wilson, B. Transformative Communication as a Stimulus to Web Innovastions. In: KHAN, B. H., ed. *Web-Based instruction.* Englewood Cliffs, Educational Technology Publications, Inc., 1997. p.67-73.

(Silva, 2000) Silva, E.Q. da. *Agente Gerenciador de Cursos a Distância via Internet.* São Carlos, 2000. Dissertação (Mestrado) - Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo.

(Silva et al, 2000) SILVA, E.Q.; MOREIRA, D.A.; DOS SANTOS Jr, J.B. Computadores no Ensino - Uma Abordagem Voltada para o Suporte aos Professores no Desenvolvimento de Atividades Didáticas. In: XI SIMPÓSIO BRASILEIRO DE INFORMÁTICA NA EDUCAÇÃO. Maceió, novembro, 2000. pag. 81-88.

(Soares, 2001) SOARES, W. *Crie um site B2C – Busines to Consumer*: Com PHP 4 e MySQL. São Paulo, Érica, 2001.

(Spennemann, 1995) Spennemann, D.H.R. *On-line study packages for distance education: Some considerations of conceptual parameters*. School of Environmental and Information Sciences. Charles Sturt University, Albury, Australia. http://life.csu.edu.au/~dspennem/Publications/WWW_Publishing/On-line_Packages.html.

(Tanenbaum, 1997) TANENBAUM, A. S. *Rede de Computadores*. Tradução da Terceira edição de Computer Networks. Rio de janeiro, Campus, 1997.

(Tarouco, 1999) TAROUCO, L.M.R. Ambiente de suporte para educação à distância. In: WORKSHOP INFORMÁTICA NA EDUCAÇÃO. Porto Alegre, 4 a 8 de Janeiro. UFRGS. 1999. http://penta.ufrgs.br/pgie/workshop/ambiente.htm.

(Tarouco, 2002) TAROUCO, L.M.R. Tecnologia para aprender/comunicar a distância. In: I CONGRESSO BRASILEIRO DE EDUCAÇÃO SUPERIOR A DISTÂNCIA. Petrópolis, RJ, 26 a 28 de março, 2002. http://www.sead.ufrj.br/esud/material/LianeTarouco.doc.

(Valente, 1993) VALENTE, J.A. Diferentes usos do Computador na Educação. In: *Computadores e Conhecimento: repensando a educação*. Gráfica Central da Unicamp, 1993. p.1-23. http://www.geocities.com/cadej_99/textos/texto1.htm.

(Valente, 1999) VALENTE, J.A. Informática na Educação no Brasil: Análise e Contextualização Histórica. In: ARMANDO J., org. *O computador na sociedade do conhecimento*. Campinas, UNICAMP/NIED, 1999. 156p.

(Valente, 1999a) VALENTE, J.A. Análise dos Diferentes Tipos de Software Usados na Educação. . In: ARMANDO J., org. *O computador na sociedade do conhecimento*. Campinas, UNICAMP/NIED, 1999. 156p.

(Valente, 2000) VALENTE, J.A. Educação a distância: uma oportunidade para mudança no ensino. In: Maia, C., org. *Educação a distância no Brasil na era da Internet*. São Paulo, Anhembi Morumbi Editora, 2000, p. 97-122.

(Vieira, 2001) VIEIRA, F.M.S. Considerações Teórico-metodológicas para Elaboração e Realização de Cursos Virtuais. In: VIII CONGRESSO INTERNACIONAL DE EDUCAÇÃO A DISTÂNCIA. Brasília, 6 a 8 de agosto, 2001. http://www.abed.org.br/congresso2001/index.html.

(Vivas, 2000) VIVAS, M. *Curso de Aplicaçoes WEB em PHP*. SuperPHP. Acessado em 2002. http://www.superphp.com.br/apostilas/index.php?secao=1.

(Willis, 1992) WILLIS, B. Instructional Development for Distance Education. *ERIC Document Reproduction Service*. /Resumo/ Acessado em 2002. http://ericae.net/edo/ED351007.htm.

(W3C, 2002) WORLD WIDE WEB CONSORTIUM (W3C*): Overview of SGML and XML Resources*. Acessado em 2002. http://www.w3.org/MarkUp/SGML/.

(W3C, 2002a) WORLD WIDE WEB CONSORTIUM (W3C): *What is HTML?*. Acessado em 2002. http://www.w3.org/MarkUp.