

defendida por Vanessa Lia Righetto  
e aprovada pela Comissão  
Julgada em 28 / 11 / 2001  
Pedro Luis Dias Peres

Orientador

**UNIVERSIDADE ESTADUAL DE CAMPINAS  
FACULDADE DE ENGENHARIA ELÉTRICA E DE  
COMPUTAÇÃO**

**DEPARTAMENTO DE TELEMÁTICA**

**PROVANET: UM SISTEMA DE AVALIAÇÃO DE  
APRENDIZADO NA INTERNET**

**VANESSA LIA RIGHETTO**

**Orientador**

Prof. Dr. Pedro Luis Dias Peres

Tese apresentada à Faculdade de Engenharia Elétrica e de Computação da Universidade Estadual de Campinas (UNICAMP) como parte dos requisitos exigidos para obtenção do título de Mestre em Engenharia Elétrica.

**Banca Examinadora**

Prof. Dr. Pedro Luis Dias Peres - DT/FEEC/UNICAMP

Prof. Dr. Ivanil Sebastião Bonatti - DT/FEEC/UNICAMP

Prof. Dr. Ivan Luiz Marques Ricarte - DCA/FEEC/UNICAMP

Profa. Dra. Sandra Maria Aluísio - ICMC/USP- São Carlos

**Campinas, 28 de novembro de 2001**

UNIDADE	80
Nº CHAMADA	
	UNICAMP
	R449p
V	
	48599
	16-837102
PREÇO	R\$ 11,00
DATA	
Nº CPD	

CM00165657-9

BIBID. 244040

FICHA CATALOGRÁFICA ELABORADA PELA  
BIBLIOTECA DA ÁREA DE ENGENHARIA - BAE - UNICAMP

R449p

Righetto, Vanessa Lia

ProvaNet: um sistema de avaliação de aprendizado na Internet / Vanessa Lia Righetto.--Campinas, SP: [s.n.], 2001.

Orientador: Pedro Luis Dias Peres.

Dissertação (mestrado) - Universidade Estadual de Campinas, Faculdade de Engenharia Elétrica e de Computação.

1. Redes de computação. 2. Internet (Redes de computação). 3. Ensino a distância. I. Peres, Pedro Luis Dias. II. Universidade Estadual de Campinas. Faculdade de Engenharia Elétrica e de Computação. III. Título.

## Resumo

Este trabalho apresenta um sistema de divulgação e avaliação de provas através da Internet denominado ProvaNet. Utilizando ferramentas computacionais de domínio público na rede WWW, o ProvaNet possui um conjunto de opções para criação e manipulação de provas e questões de múltipla escolha, para usuários professores; para alunos, fornece retorno imediato sobre o resultado de cada questão e do desempenho nas provas, e a possibilidade de realização de simulados. O ProvaNet não pressupõe habilidades específicas de informática, bastando o conhecimento básico dos comandos de navegação na Internet, contribuindo assim para disseminar e expandir o uso da Internet como veículo para o ensino e o aprendizado.

## **Abstract**

This work presents a package for test application and evaluation through the Internet named ProvaNet. Using public domain tools, ProvaNet offers a set of options which allows the teacher to create and manipulate multiple choice questions and tests; for the students, ProvaNet supplies immediate feedback about each answered question and about the overall performance, and the possibility of making tests not computed for evaluation. The use of ProvaNet does not require specific skills neither a deep knowledge on computer tools but only basic mouse commands; ProvaNet is aimed to be a contribution to spread and to disseminate the use of the Internet as a vehicle of learning and teaching.

À minha família.

## **Agradecimentos**

Agradeço ao meu orientador Prof. Dr. Pedro Luis Dias Peres pelo apoio, incentivo e amizade durante o desenvolvimento deste trabalho.

Agradeço aos professores Ivanil Sebastião Bonatti, Ivan Luiz Marques Ricarte, Rogério Drummond, Rosana Miskulin e Ana Regina Lanner pela apoio e sugestões que muito enriqueceram este trabalho.

Agradeço aos professores com os quais tive oportunidade de me aperfeiçoar através das disciplinas cursadas e a todos os funcionários de maneira geral.

Agradeço aos meus pais Alceu e Arlete, e aos meus irmãos Andréa, Janaina e Christian pelo apoio e amor.

Agradeço ao Marcelo, meu marido e grande companheiro, pelo amor, amizade e compreensão.

Agradeço à Fran, minha amiga querida, pela amizade, ajuda, imensa paciência e apoio.

Agradeço às amigas sempre presentes pela amizade e pelas boas risadas.

Agradeço aos amigos do DT com quem convivi durante o mestrado, pela amizade e companheirismo.

Agradeço à CAPES pelo apoio financeiro ao projeto.

Agradeço ao DT/FEEC/Unicamp por fornecer as condições e o material necessário para que este trabalho pudesse ser realizado.

Agradeço a todos que direta ou indiretamente contribuíram para a realização deste trabalho.

Agradeço a Deus por ter me dado a oportunidade e sabedoria para poder virar mais uma página no livro da minha vida.

# Índice

<b>Lista de Figuras</b> .....	<b>4</b>
<b>Introdução Geral</b> .....	<b>5</b>
<b>Capítulo 1. Documentos Hipertexto e Ferramentas de Avaliação na Web</b> .....	<b>8</b>
1.1. Introdução.....	8
1.2. A World Wide Web .....	8
1.3. Arquitetura .....	9
1.3.1. Cliente.....	10
1.3.2. Servidor.....	10
1.4. Ferramentas de Avaliação na Internet.....	11
1.4.1. QuestCALM.....	11
1.4.2. Hot Potatoes .....	12
1.4.3. QForm.....	12
1.4.4. Question Mark for Web .....	12
1.4.5. QuizTest.....	12
1.4.6. WebCourse.....	13
1.4.7. WebCT .....	13
1.4.8. AvalWeb.....	13
1.4.9. Castle Toolkit.....	14
1.4.10. Clyde Virtual University Assessment Engine .....	14
1.4.11. WebTest.....	14
1.4.12. CAPA.....	15
1.5. Comparação entre as Ferramentas Analisadas e o ProvaNet .....	15
1.5.1. Gerenciamento.....	15
1.5.2. Criação de Provas.....	15
1.5.3. Disponibilização de Material.....	16
1.5.4. Ambientes Distintos.....	16
1.5.5. Base de Dados .....	17
<b>Capítulo 2. Especificação da Ferramenta ProvaNet</b> .....	<b>18</b>
2.1. Introdução.....	18
2.2. Características do ProvaNet.....	18
2.3. Ambientes de Utilização .....	19
2.3.1. Ambiente de Criação de Provas .....	19
A. Criação e Alteração de Questões .....	21
B. Criação, Alteração e Disponibilização de Provas .....	22
C. Consulta do Desempenho dos Alunos .....	23
D. Simulação de Provas .....	23
2.3.2. Ambiente de Realização de Provas.....	24
A. Resolução de Provas .....	24
B. Verificação de Desempenho.....	25
<b>Capítulo 3. Modelagem do Sistema ProvaNet</b> .....	<b>26</b>
3.1. Introdução.....	26
3.2. Metodologia UML.....	26

3.3. Diagramas .....	27
3.3.1. Diagrama de Caso de Uso .....	28
3.3.2. Diagrama de Fluxo de Atividades .....	29
3.3.3. Diagrama de Seqüência .....	29
3.4. Modelagem do ProvaNet .....	29
3.4.1. Pacote Acesso e Ajuda .....	32
A. Login ProvaNet .....	33
B. Cadastrar Professor .....	33
C. Cadastrar Aluno .....	34
D. Consultar Manual de Ajuda .....	34
E. Habilitar ProvaNet - Ambiente de Criação .....	34
F. Habilitar ProvaNet - Ambiente de Realização .....	35
3.4.2. Pacote Questão .....	35
A. Criar Questão .....	36
B. Verificar Questão .....	36
C. Excluir Questão .....	36
D. Alterar Questão .....	37
3.4.3. Pacote Prova .....	40
A. Criar Avaliação .....	41
B. Criar Simulado .....	41
C. Verificar Prova .....	41
D. Alterar Dados da Prova .....	42
E. Inserir Questão na Prova .....	42
F. Excluir Prova .....	42
G. Substituir Questão da Prova .....	42
H. Excluir Questão da Prova .....	43
I. Alterar Disponibilização de Provas .....	43
J. Verificar Desempenho dos Alunos .....	43
K. Realizar Simulação de Provas .....	43
3.4.4. Pacote Resolução .....	47
A. Escolher Prova .....	48
B. Resolver Avaliação .....	48
C. Realizar Simulado .....	49
D. Verificar Desempenho .....	49
<b>Capítulo 4. Desenvolvimento do Sistema ProvaNet .....</b>	<b>53</b>
4.1. Introdução .....	53
4.2. Arquitetura .....	53
4.3. Ferramentas de Construção .....	57
4.3.1. JavaScript .....	57
4.3.2. PHP .....	58
4.3.3. MySQL .....	58
4.4. Implementação .....	59
4.4.1. Camada de Interface .....	59
4.4.2. Camada de Negócio .....	62
4.4.3. Camada de Dados .....	63



<b>Capítulo 5. Exemplo de Utilização da Ferramenta ProvaNet.....</b>	<b>65</b>
5.1. Introdução.....	65
5.2. Utilização.....	66
5.2.1. Acesso e Ajuda.....	66
5.2.2. Ambiente de Criação de Provas .....	68
A. Criar Questão .....	68
B. Verificar Questão .....	71
C. Alterar Questão .....	72
D. Excluir Questão.....	74
E. Criar Prova .....	75
F. Verificar Prova .....	77
G. Alterar Prova .....	77
H. Alterar Dados da Prova.....	79
I. Inserir Questão.....	79
J. Excluir Questão da Prova .....	80
K. Substituir Questão .....	80
L. Excluir Prova .....	81
M. Disponibilizar Prova.....	81
N. Verificar Desempenho .....	83
O. Simulação de Provas .....	84
5.2.2. Ambiente de Realização de Provas .....	86
A. Resolução de Avaliações.....	87
B. Resolução de Simulados .....	88
C. Verificação de Desempenho.....	88
<b>Conclusões e Projetos Futuros .....</b>	<b>89</b>
<b>Glossário e Lista de Acrônimos.....</b>	<b>91</b>
<b>Bibliografia .....</b>	<b>95</b>

## Lista de Figuras

Figura 1.1: Arquitetura Cliente-Servidor.....	9
Figura 3.1: Atores identificados no sistema ProvaNet.....	30
Figura 3.2: Diagrama de fluxo de atividades do ProvaNet.....	31
Figura 3.3: Diagrama de caso de uso - pacote acesso e ajuda.....	33
Figura 3.4: Diagrama de caso de uso - pacote questão.....	35
Figura 3.5: Diagrama de fluxo de atividades - criação de uma questão.....	38
Figura 3.6: Diagrama de seqüência - criação de uma questão.....	39
Figura 3.7: Diagrama de caso de uso - pacote prova.....	40
Figura 3.8: Diagrama de atividades - criação de uma prova.....	44
Figura 3.9: Diagrama de atividades - inserção de questões em uma prova.....	45
Figura 3.10: Diagrama de seqüência - criação de uma prova.....	47
Figura 3.11: Diagrama de caso de uso - pacote resolução.....	48
Figura 3.12: Diagrama de atividades - realização de uma prova.....	50
Figura 3.13: Diagrama de seqüência - realização de uma avaliação.....	51
Figura 3.14: Diagrama de seqüência - realização de um simulado.....	52
Figura 4.1: Arquitetura em três camadas.....	55
Figura 4.2: Arquitetura do ProvaNet.....	56
Figura 4.3: Divisão da tela de execução em frames.....	60
Figura 4.4: Identificação de uma classe de interface.....	61
Figura 4.5: Diagrama de classes da camada de interface.....	61
Figura 4.6: Diagrama de classes da camada de sistema.....	62
Figura 4.7: Tabelas do ProvaNet.....	63
Figura 5.1: Tela inicial do ProvaNet.....	66
Figura 5.2: Cadastramento de um aluno.....	67
Figura 5.3: Manual de ajuda.....	67
Figura 5.4: Entrada do ambiente de criação.....	68
Figura 5.5: Criar questão - enunciado e escrita das alternativas.....	69
Figura 5.6: Criar questão - enunciado e definição do valor das alternativas.....	69
Figura 5.7: Verificar questão.....	71
Figura 5.8: Alterar questão - enunciado.....	72
Figura 5.9: Alterar questão - alternativas.....	73
Figura 5.10: Excluir questão - confirmação da exclusão.....	74
Figura 5.11: Confirmação da criação de uma prova.....	75
Figura 5.12: Criar prova - definição da ordem de disponibilização.....	76
Figura 5.13: Alterar prova - funcionalidades.....	78
Figura 5.14: Alterar prova - alterar dados.....	79
Figura 5.15: Disponibilizar prova.....	82
Figura 5.16: Verificar desempenho - desempenho geral.....	83
Figura 5.17: Simulação de provas.....	84
Figura 5.18: Realização de provas - análise da resposta (simulado).....	85
Figura 5.19: Entrada do ambiente de realização.....	86
Figura 5.20: Resolução de avaliações - questão.....	87
Figura 5.21: Verificação de desempenho.....	88

---

## **Introdução Geral**

---

Atualmente, em todas as áreas do conhecimento, é marcante a necessidade de estudo e atualização progressiva dos indivíduos, em um aprendizado não mais restrito ao período escolar, mas que se estende por toda a vida, resultando em um aprimoramento do conhecimento na forma de educação continuada.

Paralelamente a isto, o mundo atual está cada vez mais exigente, em termos de tempo de dedicação, responsabilidades, quantidade e qualidade de trabalho, o que torna a educação continuada difícil de ser alcançada de acordo com as possibilidades tradicionais (presenciais) de ensino [Cruz et al., 2001].

Surge então a possibilidade do auto-aprendizado que, a bem da verdade, sempre existiu. Através da evolução da tecnologia, a popularização do uso de computadores facilitou o acesso ao material didático, permitindo que o processo de ensino fosse automatizado em todas as fases, desde a disponibilização do conteúdo até o processo de avaliação. E com a disseminação do uso da Internet, um enorme horizonte de informação se tornou disponível, permitindo que o usuário tenha acesso também ao material instalado em outras máquinas que não o seu próprio computador.

Desta forma, o que se tenta construir, no mundo da informática e da Internet, é um conjunto crescente de ferramentas poderosas e eficientes, capazes de orientar o homem nesta jornada de compreensão de conteúdos cada vez mais diversos e dispersos [Cruz et al., 2001].

Dentre as inúmeras vantagens do uso da Internet na educação, destaca-se a capacidade de utilização das diversas mídias no apoio ao ensino, como vídeo, imagens, sons e animação, permitindo a disponibilização de material didático de forma mais abrangente, aumentando a riqueza e o valor da informação apresentada. Devido a todos esses fatores, têm surgido inúmeros pacotes computacionais e instrumentos voltados para o ensino através do computador.

Com o avanço e refinamento dos ambientes de ensino à distância, novos sistemas eficientes e interativos que permitem o monitoramento do aprendizado e apresentam uma realimentação construtiva, tanto para o professor como para o aluno, têm sido disponibilizados.

Este trabalho apresenta o sistema ProvaNet, construído para facilitar a interação entre professores e alunos no processo de avaliação e aprendizado através da Internet.

O ProvaNet foi concebido a partir de ferramentas de domínio público, visando sobretudo a facilidade de utilização (não requer habilidades específicas) e a re-utilização de material já disponível em alguma mídia.

Nesta tese são abordados a construção do aplicativo e os meios disponíveis para viabilizar a proposta do sistema, sem a preocupação de discutir os aspectos educacionais, nem a questão de segurança e identificação dos usuários.

Maiores detalhes com relação ao aspecto educacional no ensino à distância podem ser encontrados em [Wolz et al., 1998], [Shute e Psotka, 1996], [McKnight et al., 1996], [Santoro et al., 1999], [Romiszowski e Manson, 1996], [Moran, 1997], [Moran, 1998], [Fróes, 1998], [Santos, 1998]. Questões relativas à segurança de um aplicativo disponibilizado em ambiente Web podem ser encontradas em [Soares et al., 1995], [Bernstein, 1999].

Este trabalho está estruturado em cinco capítulos, uma introdução geral e uma conclusão, da seguinte forma:

O Capítulo 1 aborda as principais características da Internet voltadas para apresentação de páginas HTML, além de relatar de forma sucinta algumas ferramentas de avaliação atualmente utilizadas no ensino à distância.

O Capítulo 2 contém a descrição da ferramenta ProvaNet, apresentando os usuários e as atividades suportadas pelo aplicativo nos ambientes de manipulação.

O Capítulo 3 descreve a modelagem concebida para o ProvaNet, apresentando as funcionalidades do sistema mapeadas através de diagramas construídos segundo a metodologia UML.

O Capítulo 4 apresenta o desenvolvimento do sistema, com a divisão em camadas e os *softwares* utilizados.

Um exemplo completo de utilização das ferramentas concebidas é mostrado no Capítulo 5, com a elaboração de questões e provas por parte do professor e a realização de provas por parte dos alunos.

Finalmente, é apresentada a conclusão deste trabalho, e algumas possibilidades de projetos futuros na área de ensino à distância são discutidas.

---

## Capítulo 1

### Documentos Hipertexto e Ferramentas de Avaliação na Web

---

#### 1.1. Introdução

Este capítulo tem por objetivo abordar as principais características da rede Internet na disponibilização de páginas HTML, apresentando de forma resumida a evolução e arquitetura de comunicação da rede atual.

Em seguida, é feito um breve relato de algumas ferramentas atualmente utilizadas para a avaliação do aprendizado através da Internet.

#### 1.2. A World Wide Web

A *World Wide Web* (WWW) é uma estrutura de rede que permite o acesso a documentos vinculados espalhados por milhares de máquinas na Internet. Sua enorme popularidade se deve principalmente às interfaces gráficas amigáveis, de fácil utilização para aqueles que não possuem conhecimento técnico em computadores, oferecendo uma imensa variedade de informações [Tanenbaum, 1997], [Laquey e Ryer, 1994].

Uma das principais características da Internet é a de prover mecanismos para que o usuário navegue pelos documentos disponibilizados na rede sem precisar conhecer os protocolos envolvidos, podendo inclusive ignorar a existência destes protocolos [Tanenbaum, 1997].

A disponibilização de programas construídos para suportar a visualização de páginas (*browsers*) motivou a construção de documentos através da utilização de uma linguagem de formatação chamada *HyperText Markup Language* (HTML).

O HTML é uma linguagem de marcação derivada do *Standard Generalized Markup Language* (SGML), que permite a estruturação de documentos. A linguagem HTML oferece um conjunto reduzido de instruções que, quando

dispostos em um arquivo texto (ASCII), possibilitam que o *browser* exiba o documento formatado.

Programas CGI (*Common Gateway Interface*) podem ser usados em conjunto com páginas HTML para implementar interfaces interativas, propiciando a troca de informações entre o usuário e os sistemas disponibilizados através da rede.

### 1.3. Arquitetura

A arquitetura Internet proporciona a interligação de diferentes tecnologias de redes [Soares et al., 1995]. A arquitetura de comunicação na WWW é do tipo cliente-servidor, como mostrado na Figura 1.1, que representa a interação entre o computador cliente, usando o *browser*, o computador servidor, com o servidor Web e outras aplicações (chamadas de aplicações CGI) e o computador servidor de base de dados.

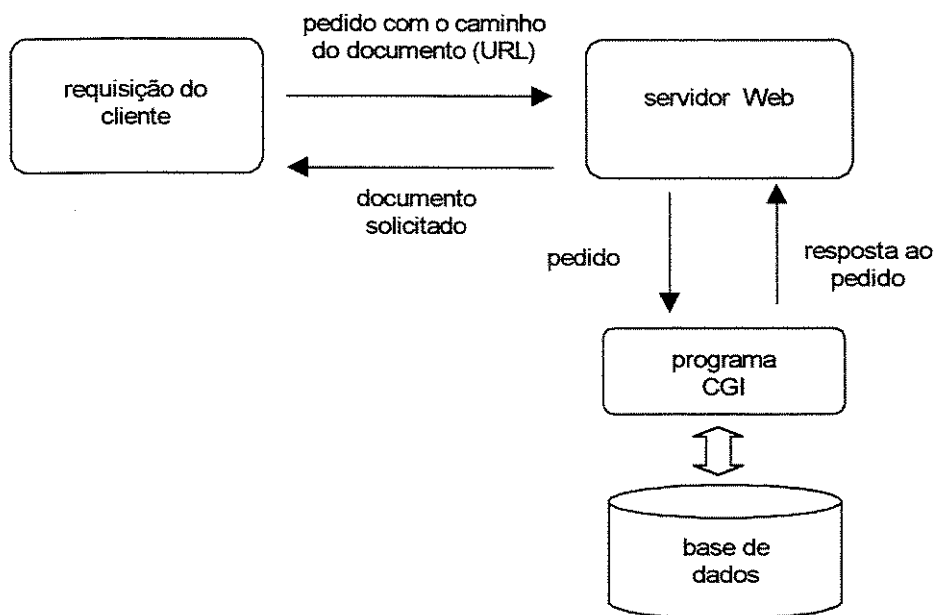


Figura 1.1: Arquitetura Cliente-Servidor.

### **1.3.1. Cliente**

Do ponto de vista do usuário [Tanenbaum, 1997], a WWW é uma vasta coleção de documentos. Cada página pode conter *links* para outras páginas, de forma que os usuários possam seguir este apontamento que leva até a informação referenciada. Este processo pode ser repetido indefinidamente, possibilitando aos usuários percorrer centenas de documentos.

Quando um usuário solicita a visualização de um documento na Internet, este processo ocorre através do *browser*, que recupera a página solicitada, interpreta o seu conteúdo (comandos de formatação) e apresenta a página ao usuário.

No processo de visualização do documento, é gerada uma mensagem que contém o pedido referente à página requisitada. O *HyperText Transfer Protocol* (HTTP) é o protocolo usado na WWW para a distribuição e recuperação de informação, em sua maioria documentos hipertexto. A troca de informações entre um *browser* e um servidor Web é toda feita através deste protocolo, que define um conjunto de regras para a efetiva comunicação.

Um pedido HTTP é composto do método, que é a ação a ser realizada, e da URL (*Universal Resource Locator*), que é a informação requisitada. O servidor Web, ao receber o pedido, processa-o de modo a determinar o que deverá ser feito.

### **1.3.2. Servidor**

Todos os *sites* na Web possuem processos servidores que aguardam conexões de clientes (geralmente *browsers*). Depois de estabelecida a conexão, o servidor recebe a requisição enviada, processa este pedido e envia a resposta. Após o envio da resposta a conexão é liberada. O protocolo HTTP define as solicitações e respostas válidas [Tanenbaum, 1997], [Soares et al., 1995].



Se a solicitação de um cliente requisitar uma consulta a uma base de dados, por exemplo, programas CGI são executados de forma a permitir que estes dados se tornem disponíveis. A execução do programa CGI se realiza através da criação de um *script* (ou um programa) que é utilizado como interface entre o banco de dados e a Web. A este *script* é atribuída uma URL, que os servidores HTTP processam como arquivos executáveis [Tanenbaum, 1997].

Através de programas que interagem via CGI com o servidor Web, é possível criar qualquer tipo de aplicação, desde códigos simples que enviam uma mensagem de correio eletrônico ou gravam uma informação em um arquivo texto, até aplicações que interagem com bancos de dados diversos. Pode-se dizer que a integração de bases de dados à Internet, através do serviço *Web*, tem sido um dos principais focos de crescimento da *World Wide Web*.

#### **1.4. Ferramentas de Avaliação na Internet**

A seguir, um breve relato de algumas ferramentas disponibilizadas para a construção de ambientes de avaliação é apresentado [Leiva, 1999], [Righetto et al., 2001]:

##### **1.4.1. QuestCALM**

O QuestCALM é uma ferramenta para autoria de questionários, que é incorporada ao ambiente CALM (*Computer-Aided Learning Material*). O CALM consiste em um sistema de aprendizado colaborativo que recomenda material didático com base no objetivo da aprendizagem de um estudante e no seu perfil evolutivo, apoiando a interação e a colaboração entre os seus participantes e materiais de aprendizagem envolvidos.

A ferramenta possibilita a autoria de questões nos formatos: múltipla escolha, dissertativa e verdadeiro ou falso [Leiva, 1999].

### **1.4.2. Hot Potatoes**

Consiste em um conjunto de cinco ferramentas para criar os seguintes tipos de exercícios: múltipla escolha, preenchimento de lacunas, questão dissertativa (ou aberta), sentença embaralhada e palavras cruzadas.

Os exercícios são criados através de interfaces simples e estruturadas, de forma que a tecnologia envolvida fica totalmente transparente para o usuário. Ou seja, não é necessário que o usuário conheça as linguagens envolvidas na construção das questões, como HTML e JavaScript, nem mesmo qualquer outra ferramenta utilizada [Martin e Stewart, 1998].

### **1.4.3. QForm**

Ferramenta utilizada em ambiente Internet composta de um pacote com dois programas: o QForm, utilizado para criação de testes em linguagem HTML e o QScore, utilizado para a correção dos testes criados pelo QForm.

Este conjunto de programas facilita a criação e a correção automática de questões de múltipla escolha e verdadeiro ou falsa [Cunningham, 1995], [Santos, 1998].

### **1.4.4. Question Mark for Web**

Ferramenta que possibilita a criação de questões tipo múltipla escolha, dissertativa, verdadeiro ou falso e relacionamento de colunas, que podem ser posteriormente disponibilizadas em ambiente Internet [Question, 1997].

### **1.4.5. QuizTest**

Permite a criação de testes de múltipla escolha e verdadeiro ou falso. Os autores criam os testes usando HTML, gerando um arquivo sobre o qual uma rotina CGI verifica a resposta dada pelo estudante ao teste [Pfaff-Harris, 1999], [Santos, 1998].

#### **1.4.6. WebCourse**

Ferramenta baseada em formulários HTML que são preenchidos pelos instrutores para a criação de provas com questões dos seguintes tipos: múltipla escolha, preenchimento de lacunas, resposta livre e verdadeiro ou falso [Scapin, 1997].

#### **1.4.7. WebCT**

Um dos pacotes mais completos para desenvolvimento de cursos na Internet, fornecendo ao instrutor diversos dados estatísticos sobre o andamento do curso e o progresso individual dos alunos. Esta ferramenta permite a inserção de questões abertas, de múltipla escolha e verdadeiro ou falso [WebCT, 2001].

#### **1.4.8. AvalWeb**

Sistema que gera provas automaticamente para disciplinas cadastradas por um professor a partir de questões definidas previamente. As provas são criadas de acordo com o nível das questões inseridas, porém o sistema não permite a manutenção de questões e provas criadas. O tempo de realização de uma prova é controlado.

No sistema AvalWeb, o professor cria um número mínimo de questões (no caso, cinco) e especifica a porcentagem de questões fáceis, médias e difíceis que deverão fazer parte da prova. A ferramenta escolhe aleatoriamente dentre as questões criadas e gera automaticamente uma prova no sistema, que é imediatamente disponibilizada para resolução do aluno e o professor então não mais atua nas provas e questões já criadas [Avalweb, 2001].

#### **1.4.9. Castle Toolkit**

A ferramenta Castle Toolkit (*Computer Assisted Teaching and Learning*) permite a criação de testes de múltipla escolha, nos quais as questões podem ser criadas usando arquivos PDF, imagens e sons, tanto na composição do enunciado como na das alternativas.

As questões podem ser criadas e manipuladas sem que o autor tenha qualquer conhecimento anterior em informática [CAA Centre, 2001].

#### **1.4.10. Clyde Virtual University Assessment Engine**

Ferramenta que permite aos usuários projetar testes e avaliações em ambiente Web, sendo que as questões podem ser no formato de escolha múltipla, resposta múltipla, texto livre e lista ordenada.

O ambiente primeiramente disponibiliza a questão ao usuário, para em seguida fornecer o retorno com a avaliação da resposta. Este retorno pode ser dado imediatamente ou também ser enviado através de *email* [CAA Centre, 2001].

#### **1.4.11. WebTest**

O WebTest permite a criação de questões através do uso de diversos elementos nas perguntas, como números, textos, imagens e diagramas. Elementos multimídia tais como gráficos e diagramas também podem ser incluídos. As respostas às perguntas elaboradas podem ser dadas através de números, expressões matemáticas, texto ou escolha múltipla [CAA Centre, 2001].

O retorno com o desempenho do aluno é fornecido através do envio de um *email*.

#### **1.4.12. CAPA**

A ferramenta CAPA (*Computer Assisted Personalized Approach*) permite que o usuário crie e manipule questões de múltipla escolha através da utilização de figuras, animações, gráficos, tabelas, *links*, etc. Aos estudantes é dado o retorno imediato do seu desempenho. Além disso, o sistema disponibiliza ferramentas para análise estatística [Capa, 2001].

### **1.5. Comparação entre as Ferramentas Analisadas e o ProvaNet**

As principais diferenças entre as ferramentas analisadas e o ProvaNet são as seguintes:

#### **1.5.1. Gerenciamento**

O ProvaNet realiza o gerenciamento das questões e provas criadas pelos professores, permitindo que o autor possa verificar, alterar e excluir questões e provas criadas no sistema.

Na maioria das ferramentas analisadas, a única interação entre um professor e o sistema ocorre durante a criação de uma questão, que posteriormente poderá ser inserida em uma prova. As questões e provas geradas nem sempre podem ser excluídas ou mesmo alteradas através do aplicativo.

#### **1.5.2. Criação de Provas**

No ProvaNet, uma prova é criada pelo professor, que determina quais questões deverão ser inseridas e qual a ordem de apresentação ao aluno. Depois de criada, uma prova pode ser totalmente manipulada pelo professor.

Além disso, uma prova pode ser criada com um número arbitrário de questões, sendo que sua disponibilização ao aluno fica a critério do professor.

Na maioria das ferramentas analisadas, uma vez criada, o sistema disponibiliza automaticamente a prova ao aluno, não permitindo que esta ação aconteça de maneira programada pelo professor.

### **1.5.3. Disponibilização de Material**

Nas ferramentas analisadas, as questões podem ser criadas em diversos formatos: múltipla escolha, preenchimento de lacunas, questão dissertativa (ou aberta), sentença embaralhada, palavras cruzadas, entre outras.

No ProvaNet, as questões obedecem ao padrão de múltipla escolha, porém com um número arbitrário de alternativas. Além disso, o professor pode inserir material previamente criado, sendo que o sistema (através do fornecimento do caminho para localização do material) recupera o arquivo fornecido e o disponibiliza ao aluno. Como o material informado está inserido no domínio do usuário, o sucesso da sua disponibilização está vinculada a possibilidade de acesso ao arquivo.

### **1.5.4. Ambientes Distintos**

Tanto o ProvaNet como as ferramentas analisadas disponibilizam ambientes distintos para professores e alunos.

O ProvaNet, porém, permite que o professor possa simular a realização de uma avaliação ou um simulado através do acesso ao ambiente de simulação de provas, verificando a forma como as questões são apresentadas aos alunos.

### **1.5.5. Base de Dados**

Tanto o ProvaNet como a maioria das ferramentas analisadas possuem um banco de dados utilizado para gravação das questões e provas inseridas no sistema. Quando uma prova ou questão é criada, estas informações são armazenadas na base de dados dos sistemas e disponibilizadas quando requisitadas.

Porém, no ProvaNet, se um professor preferir criar questões através da utilização de arquivos localizados em seu próprio domínio, a manutenção do material didático disponibilizado se torna independente do aplicativo. Ou seja, quando um professor atualiza o material que está localizado na seu domínio, não é necessário alterar a questão que foi criada no sistema ProvaNet.

---

## Capítulo 2

### Especificação da Ferramenta ProvaNet

---

#### 2.1. Introdução

Este capítulo aborda a especificação da ferramenta ProvaNet, descrevendo os ambientes de manipulação e as funcionalidades disponíveis ao usuário.

#### 2.2. Características do ProvaNet

O objetivo básico do ProvaNet é o apoio à disseminação do conhecimento, através da estruturação e disponibilização de temas de estudo na Internet. Nesse sentido, o ProvaNet apresenta um ambiente de avaliação educacional baseado em interface WWW voltado para usuários sem experiência técnica em computadores, provendo todos os mecanismos necessários para elaboração de questões e provas (avaliações e simulados) [Righetto et al., 2001].

Para que o ambiente se torne atrativo tanto para professores como para alunos, o sistema faz uso de interfaces amigáveis que oferecem ferramentas para criação, manipulação, disponibilização e resolução de questões e provas.

Duas premissas básicas foram adotadas na concepção do ProvaNet: utilizar ferramentas de domínio público para a construção do aplicativo e prover aos usuários a facilidade de incorporação de material já existente.

A opção por ferramentas não comerciais gratuitas e disponíveis na Internet torna o aplicativo atraente para o meio acadêmico, em função da facilidade de uso e reprodução em outros *sites*.

A possibilidade de incorporar arquivos já existentes permite ao professor reutilizar provas e questões previamente criadas, em qualquer formato que seja interpretado pelo navegador. Por exemplo, arquivos em formato *Portable Document Format* (PDF), amplamente utilizados no meio acadêmico; ou



qualquer outro tipo de arquivo interpretado pelos navegadores, como arquivos de imagens, arquivos textos ou ASCII e arquivos em formato HTML. Desta forma, facilita-se o processo de construção e visualização das questões que compõem as provas, permitindo que documentos já elaborados possam ser imediatamente utilizados, dispensando nova formulação e mantendo a fidelidade à formatação original.

A incorporação de arquivos previamente criados na construção de questões traz uma vantagem adicional: o arquivo utilizado permanece no domínio do usuário e apenas o endereço é armazenado pelo aplicativo. Se o usuário deseja criar uma questão com um enunciado extenso, poderá optar por fazê-lo em um arquivo independente, indicando apenas a localização do arquivo (URL), não sobrecarregando a base de dados do sistema.

A utilização do ProvaNet é detalhada a seguir.

### **2.3. Ambientes de Utilização**

O ProvaNet oferece funcionalidades específicas de acordo com os tipos de usuários, que podem ser professores ou alunos. Após a solicitação de um cadastramento inicial, o sistema disponibiliza o ambiente apropriado, ou seja, o ambiente de criação de provas, para professores, e o ambiente de realização de provas, para estudantes.

#### **2.3.1. Ambiente de Criação de Provas**

Para o professor, o ProvaNet disponibiliza módulos que atendem ao processo de avaliação como um todo, englobando a criação e manipulação de questões, a criação e manipulação de provas, a simulação de realização de provas, a análise do desempenho dos alunos e a criação de simulados.

No ProvaNet, as provas contemplam dois tipos de documentos: avaliação e simulado.

O primeiro tipo de prova concebido pelo sistema, a avaliação, permite que o aluno indique a resposta de uma questão apenas uma vez, sem a opção de

resolver a questão novamente. Uma vez respondida a alternativa de uma questão, o sistema disponibiliza a próxima cadastrada, inibindo a visualização da questão anterior. Neste caso, o aluno tem o desempenho computado pelo sistema de acordo com as questões certas resolvidas.

O segundo tipo de prova é o simulado, no qual o aluno pode resolver a questão quantas vezes achar necessário. Ao indicar a alternativa, o sistema apenas informa se a questão foi respondida corretamente ou não. Caso o aluno tenha errado a questão, o sistema permite a volta à questão anterior para uma nova tentativa. A resposta correta da questão não é dada ao aluno, o sistema apenas indica se a questão resolvida está certa ou errada.

Os professores criam questões para provas (avaliações ou simulados), que são armazenadas pelo aplicativo, de acordo com a disciplina, assunto e enunciado associados a cada questão. Todas as questões criadas são disponibilizadas para consulta, de acordo com as características definidas no momento da criação.

A construção de uma prova se faz através da inserção de questões previamente criadas para uma mesma disciplina. Como principal característica, o ProvaNet permite a utilização de documentos previamente editados. Não é necessária a introdução de enunciados através de caixas de diálogos, apesar desta opção estar disponível.

A ordem de inserção das questões é preservada na prova no momento da disponibilização; para que uma prova possa ser resolvida pelo aluno, deve ser previamente disponibilizada pelo professor.

O ProvaNet também permite a verificação do desempenho dos alunos em uma determinada prova, individual ou coletivamente.

O ambiente de criação é composto pelo conjunto de ferramentas para a criação e alteração de questões, conjunto de ferramentas para criação, alteração e disponibilização de provas e simulados, ferramentas de consulta de desempenho dos alunos, e pelo ambiente de simulação de provas, detalhados a seguir.

## **A. Criação e Alteração de Questões**

As questões concebidas no sistema estão no formato de múltipla escolha. Desta forma, cada questão criada possui um enunciado, que pode ser inserido através da digitação do texto ou através da associação de um arquivo previamente criado; e possui também um número arbitrário de alternativas, inserido no momento da definição das características da questão.

Após a definição do número de alternativas, estabelecem-se quais as certas e as erradas. O professor pode definir como corretas um número arbitrário de alternativas (deve existir pelo menos uma certa), porém na resolução da questão o aluno escolhe apenas uma.

Este módulo provê a manipulação de questões já criadas, permitindo também a alteração e eliminação de questões. O professor pode alterar o enunciado da questão, atualizando um texto inserido inicialmente; substituir um texto por um arquivo e vice-versa; substituir um arquivo por outro; alterar o número de alternativas, inserindo novas ou excluindo as inseridas inicialmente; alterar o enunciado das alternativas, alterar o valor das alternativas ou ainda remover a questão.

Quando um usuário solicita a eliminação de uma determinada questão, o sistema verifica se a questão está inserida em uma prova. Em caso afirmativo, o sistema pede a confirmação da exclusão, informando quais as provas afetadas. Se for confirmado, o sistema remove a questão, atualizando todas as provas envolvidas.

As questões podem ser criadas de forma independente, sem necessariamente fazer parte de alguma prova imediatamente. O sistema armazena as questões criadas em um banco de dados, que pode ser consultado sempre que necessário através da busca por disciplina e assunto.

Um professor tem acesso a todas as questões criadas no ambiente, mesmo às formuladas por outros professores. Esta regra foi definida com o objetivo de estimular o compartilhamento de informações entre docentes que utilizam o sistema.

## **B. Criação, Alteração e Disponibilização de Provas**

Uma prova (avaliação ou simulado) é concebida a partir da escolha de um conjunto de questões, dentre as criadas para uma determinada disciplina. Para isso, o professor deve definir qual a disciplina desejada, o nível da prova (básica, intermediária ou avançada) e o peso (de um a cinco). Após a definição destes dados, o sistema disponibiliza as questões existentes na disciplina para serem inseridas. A ordem de inserção das questões na prova será a ordem apresentada ao aluno.

O professor pode inserir na prova um número arbitrário de questões, fazendo esta escolha sempre de acordo com um assunto e estabelecendo em seguida qual o peso daquela questão na prova. Estes dados são utilizados posteriormente no cálculo da nota final do aluno.

Ao final do processo de definição de questões, o ProvaNet solicita a ordem de disponibilização da prova, já que diversas provas podem ser geradas para a mesma disciplina, apresentando para escolha o número que representa o total de provas criadas e já disponibilizadas na disciplina. O professor pode oferecer um número qualquer de provas aos alunos, estabelecendo a ordem de habilitação. No caso de diversas provas apresentarem a mesma ordem, será exibida ao aluno aquela com data de criação mais recente.

As provas podem ser criadas sem necessariamente serem disponibilizadas para resolução. O professor pode criar um conjunto de provas e utilizá-las convenientemente, fazendo a disponibilização no momento em que finalizar a criação de uma prova, ou deixar esta operação para um estágio posterior. O objetivo é disponibilizar diversos testes para tópicos específicos da matéria, permitindo a realização de avaliações progressivas.

Cada prova é identificada pela disciplina à qual pertence e pela data de criação. Desta forma, o ProvaNet disponibiliza ao usuário a opção de alteração dos dados de identificação de uma prova, através da atualização da data de elaboração, do nível e do peso.

É possível também a manipulação de questões da prova, permitindo-se a inserção, eliminação ou alteração de questões. Uma questão pode ser inserida ou retirada da prova, sempre mantendo a ordem de apresentação. Na alteração de uma prova, uma questão pode ser trocada por outra, mas não manipulada em seu conteúdo. A ordem de disponibilização de uma prova também pode ser alterada.

O sistema permite a consulta a uma prova, exibindo os dados de identificação e as questões inseridas, detalhando o enunciado, as alternativas, o valor e o peso das questões.

Finalmente, uma prova pode ser eliminada pelo usuário, porém as questões inseridas permanecem no banco de dados do ProvaNet e podem ser utilizadas em um outro momento.

### **C. Consulta do Desempenho dos Alunos**

O professor pode verificar o desempenho individual ou coletivo dos alunos que realizaram as avaliações. Escolhendo uma avaliação específica em uma determinada disciplina, o professor obtém a média geral e as notas dos alunos que realizaram a prova. Esta escolha é feita pela consulta à data de criação de uma avaliação.

Um professor pode consultar o desempenho dos alunos em qualquer prova criada no sistema, mesmo que não tenha sido o autor da avaliação. Para simulados, esta opção não está disponível, pois neste caso o sistema não calcula o desempenho dos alunos.

### **D. Simulação de Provas**

O professor pode simular a realização da prova que criou, mesmo que esta prova, avaliação ou simulado, ainda não tenha sido disponibilizada. O objetivo é permitir ao professor a visualização da prova da mesma forma que esta será apresentada a um aluno, podendo assim fazer uma pré-avaliação do material.

### **2.3.2. Ambiente de Realização de Provas**

O ambiente de realização de provas é destinado aos alunos, para resolução de provas disponibilizadas pelo professor.

Nesta área, os alunos têm acesso somente às provas que foram habilitadas pelo professor, sempre de acordo com a disciplina na qual estão cadastrados. Nenhuma funcionalidade de manipulação de provas é disponibilizada ao aluno, sendo que este ambiente é responsável apenas pela apresentação de questões e provas cadastradas por professores no ambiente de criação de provas.

#### **A. Resolução de Provas**

No momento da definição da disciplina, o sistema apresenta as avaliações e simulados habilitados, se houver, permitindo que aluno faça a escolha do tipo de prova que deseja resolver.

Definido qual o tipo, o ProvaNet apresenta a primeira prova habilitada pelo professor na disciplina e exibe as questões cadastradas uma a uma, obedecendo aos critérios estabelecidos para a resolução de cada tipo de prova.

No caso da escolha ter sido por uma avaliação, o ProvaNet apresenta as questões cadastradas, aguarda pela resposta do aluno baseada na definição da alternativa correta, calcula o erro ou acerto e exibe a próxima questão. Não é permitido voltar à questão anterior.

No caso da escolha ter sido por um simulado, o ProvaNet apresenta as questões cadastradas, aguarda pela resposta e analisa o erro ou acerto do aluno. Caso a alternativa escolhida esteja correta, o sistema exibe uma nova tela com o resultado da resposta definida pelo aluno e a próxima questão cadastrada. Caso a resposta esteja errada, o sistema informa o erro ao aluno, não fornece a alternativa correta e permite que o aluno volte à questão errada para uma nova tentativa. Se desejar, o aluno pode passar para a próxima questão. Em um simulado, o desempenho do aluno não é calculado.

## **B. Verificação de Desempenho**

Depois de realizar uma avaliação, o aluno pode verificar o seu desempenho e também ter acesso à média dos demais alunos que concluíram a prova.

Um exemplo da utilização do ProvaNet é representado no Capítulo 5.

---

## Capítulo 3

### Modelagem do Sistema ProvaNet

---

#### 3.1. Introdução

Este capítulo tem como objetivo apresentar a modelagem concebida para o desenvolvimento do sistema ProvaNet.

O mapeamento das funções do sistema está descrito através da metodologia de desenvolvimento *Unified Modeling Language* (UML) [Boggs e Boggs, 1999], que enfoca a definição de diagramas concebidos para o desenvolvimento do sistema e os relacionamentos identificados.

A UML é o resultado da fusão dos métodos *Booch*, desenvolvido por Grady Booch, *Object Modeling Technology* (OMT), desenvolvido por James Rumbaugh e *Object Oriented Software Engineering* (OOSE), desenvolvido por Ivar Jacobson. Esta unificação deu origem a uma linguagem padronizada para modelagem de sistemas orientados a objetos, adotada pela indústria de *software* como linguagem padrão e também por fornecedores de sistemas de engenharia assistida por computador, ou seja, fornecedores de ferramentas CASE (*Computer Aided System Engineering*). Consiste em uma linguagem gráfica para visualização, especificação, construção e documentação dos artefatos de um sistema fundamentado em *software*, permitindo o desenvolvimento de diferentes tipos de diagramas visuais que representam diversos aspectos do sistema [Boggs e Boggs, 1999].

#### 3.2. Metodologia UML

Os trabalhos para criação da UML iniciaram-se em 1994 com Grady Booch da *Rational Software* (Booch) e James Rumbaugh (OMT), que combinaram seus dois métodos mais populares. Em 1996, Ivar Jacobson (OOSE) passou a integrar a equipe de desenvolvimento, objetivando a expansão do escopo da



UML. Desta forma, ocorreu a formação de um consórcio de empresas para apoiar a definição da linguagem.

A UML é uma linguagem para especificação, documentação, visualização e desenvolvimento de sistemas orientados a objetos. Sintetiza os principais métodos existentes, sendo considerada uma das linguagens mais expressivas para a modelagem de aplicativos [Booch et al., 1997].

Por meio dos diagramas é possível representar sistemas de *software* sob diversas perspectivas de visualização, facilitando o entendimento da transição de uma fase de construção para outra, desde a fase de análise até a fase de testes e implementação do sistema concebido. Desta forma, a UML não é uma linguagem de programação visual e, sim uma linguagem para modelagem visual usada na construção de artefatos de *software* que fornecem a base necessária para a tomada de decisão quanto à definição de diversos aspectos do sistema. Como uma linguagem de modelagem, deve ser aplicada no processo de desenvolvimento, tratando, em diversos domínios, diferentes problemas com objetivos distintos.

### **3.3. Diagramas**

Na modelagem do aplicativo foram utilizados principalmente os diagramas referentes à fase de concepção do sistema, que contém a definição das funcionalidades desejadas e os usuários identificados. O objetivo deste esforço é o mapeamento da estrutura e do comportamento do ProvaNet e, para isso, foram utilizados diagramas de caso de uso, diagramas de fluxo de atividades e diagramas de seqüência [Booch et al., 1997], [Boggs e Boggs, 1999].

Casos de uso representam as funcionalidades do sistema, isto é, os requisitos que o sistema deve ter sob a perspectiva do usuário e que devem ser identificados de modo a agregar algum valor para quem os utiliza. Funcionalidades como verificação, consistência, etc., estão embutidas nos casos de uso, pois não apresentam um resultado de valor que seja percebido

diretamente pelo usuário, sendo funcionalidades transparentes descritas ao longo do fluxo de realização.

Atores são papéis desempenhados por qualquer usuário de um caso de uso, ou seja, o ator é quem solicita os serviços disponibilizados pelo sistema. Pode ser uma pessoa ou um *hardware* que interage com o aplicativo, ou então um outro sistema que tenha a necessidade de utilizar os casos de uso. Atores, portanto, representam pessoas ou sistemas que enviam ou recebem algum tipo de informação do aplicativo.

Durante o processo de modelagem, são identificados os atores que fazem parte do sistema, com os papéis que devem desempenhar. Também são identificados os cenários do aplicativo, considerando qual o comportamento desejado representado por meio dos casos de uso. Em seguida, é efetuada a organização dos diagramas de casos de uso em pacotes, agrupando-os segundo o tipo de cenário empregado e finalmente é detalhada a realização dos casos de uso.

Na modelagem UML, itens como atores, casos de uso e classes de objetos podem ser agrupados em pacotes, como forma de facilitar o entendimento das funcionalidades identificadas no sistema, cada qual tratando um aspecto ou um conjunto de funcionalidades da concepção do sistema.

Para a modelagem do sistema ProvaNet, foi utilizada a ferramenta *Rational Rose* [Boggs e Boggs, 1999], que suporta a construção de todos os artefatos contemplados pela modelagem UML.

### **3.3.1. Diagrama de Caso de Uso**

Os diagramas de caso de uso representam um conjunto de cenários identificados que captam o comportamento desejado do sistema por meio da análise das condições necessárias para sua construção. Permitem a melhor compreensão da abrangência dos requisitos, pois utilizam recursos gráficos para representar a interação entre os casos de uso identificados e os atores que utilizam o sistema.

A estrutura de um diagrama de caso de uso é bastante simples; os cenários são formados por textos, descritos em linguagem informal, possibilitando o entendimento de todos que participam do processo de desenvolvimento [Jacobson et al., 1995], [Boggs e Boggs, 1999].

### **3.3.2. Diagrama de Fluxo de Atividades**

Os diagramas de fluxo de atividades descrevem o fluxo de controle das funcionalidades do sistema mapeado através dos casos de uso e oferecem suporte para análise de comportamentos condicionais e paralelos [Boggs e Boggs, 1999], [Booch et al., 1997], [Pressman, 1995].

### **3.3.3. Diagrama de Seqüência**

Os diagramas de seqüência dão ênfase à ordenação seqüencial em que o comportamento de um caso de uso acontece, representando a dinâmica dos eventos disparados entre ator e os objetos do sistema contemplados pelo caso de uso.

A partir dos eventos gerados pelos atores, operações de resposta são disparadas e descritas em função deste estímulo, até que todo o comportamento do caso de uso esteja mapeado.

As barras verticais nos diagramas de seqüência representam a vida do ator ou objeto durante a interação e uma mensagem transmitida é representada por uma flecha na posição horizontal [Boggs e Boggs, 1999], [Booch et al., 1997].

## **3.4. Modelagem do ProvaNet**

O sistema ProvaNet é composto por dois ambientes distintos de manipulação, sendo um ambiente para criação de provas, utilizado por professores e um ambiente para realização de provas, utilizado por alunos cadastrados.

Foram identificados quatro atores que interagem com o sistema, descritos a seguir:

- **Usuário:** representa qualquer usuário que utiliza o sistema.
- **Administrador:** responsável pelo gerenciamento do cadastro de professores.
- **Professor:** cadastrado no sistema pelo administrador, usuário do ambiente de criação de provas. Pode criar provas e questões e disponibilizá-las aos alunos.
- **Aluno:** usuário do ambiente de realização de provas. Pode cadastrar-se no sistema a qualquer momento, porém somente poderá realizar provas das disciplinas nas quais houver provas disponibilizadas.

O diagrama mostrado na Figura 3.1 identifica o mapeamento dos atores.

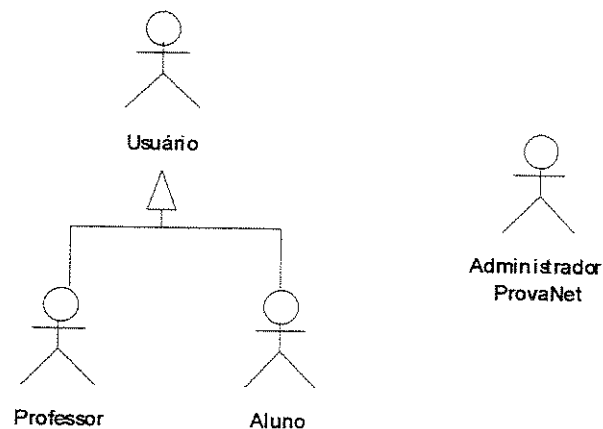


Figura 3.1: Atores identificados no sistema ProvaNet.

A Figura 3.2 ilustra o diagrama de fluxo de atividades do aplicativo, representando de forma genérica as atividades realizadas pelos usuários do sistema e o encadeamento das ações.

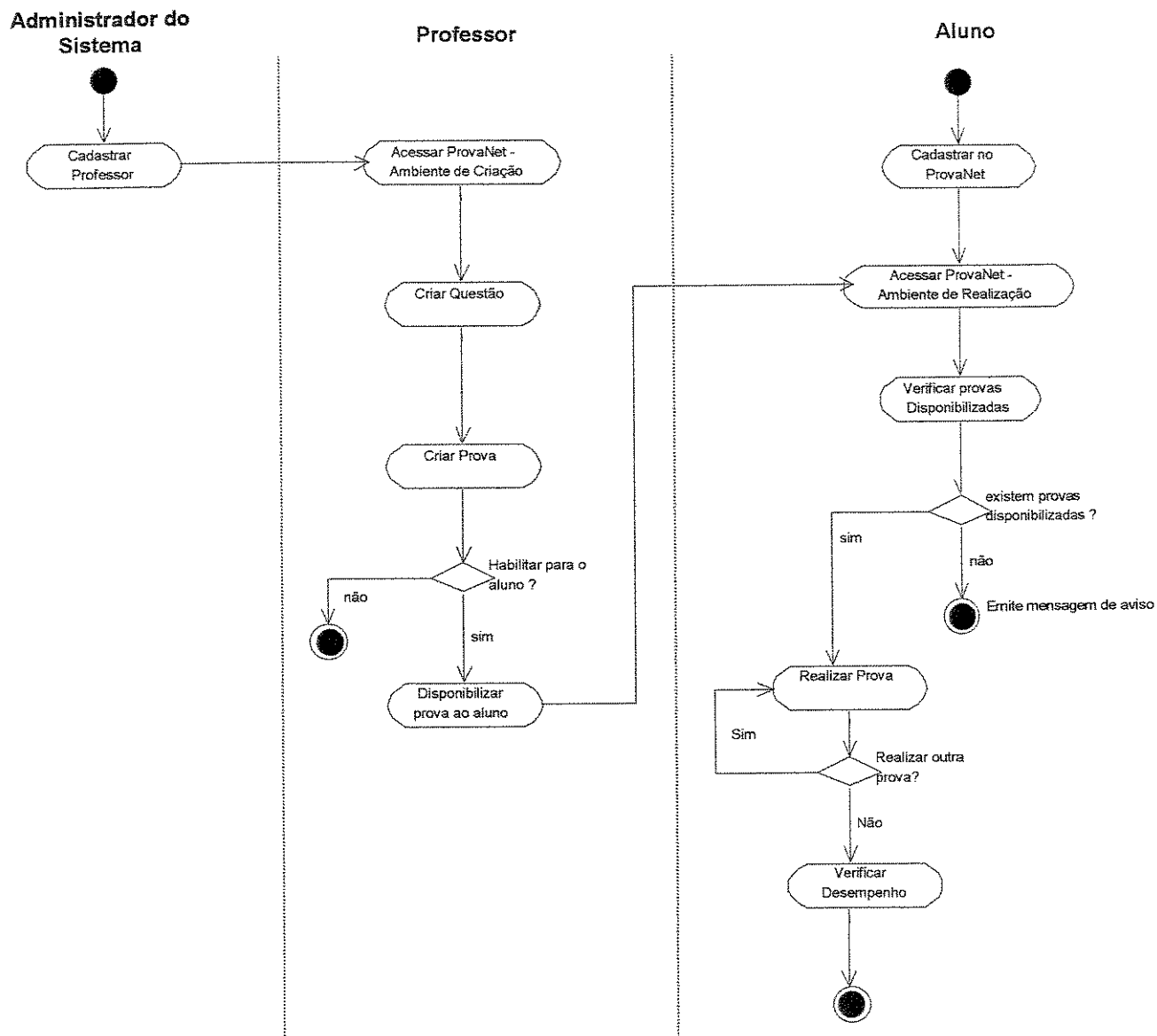


Figura 3.2: Diagrama de fluxo de atividades do ProvaNet.

Neste fluxo estão representadas as macroatividades realizadas pelos atores do sistema. O administrador realiza o cadastramento, permitindo que o professor acesse o ambiente de criação, crie questões e provas, e disponibilize as provas aos alunos. Por sua vez, um aluno pode se cadastrar no ProvaNet para acessar o ambiente de realização de provas, resolver as provas disponibilizadas pelos professores e verificar o desempenho.

Como forma de organizar os casos de uso identificados no ProvaNet, a modelagem das funcionalidades foi dividida em quatro pacotes distintos, que são:

- **Acesso e Ajuda:** funcionalidades de acesso ao sistema ProvaNet e funcionalidades de consulta ao manual de ajuda.
- **Questão:** funcionalidades que fazem referência ao gerenciamento de questões do ambiente (criação, alteração e eliminação).
- **Prova:** funcionalidades que fazem referência ao gerenciamento de provas do ambiente (criação, alteração e eliminação).
- **Resolução:** funcionalidades de resolução de uma prova por parte do aluno.

A seguir são detalhados os quatro pacotes identificados no sistema, apresentando-se os diagramas contemplados.

#### **3.4.1. Pacote Acesso e Ajuda**

Todos os atores identificados no sistema interagem com os casos de uso de acesso e ajuda, pois professores e alunos acessam o sistema e o administrador provê o cadastramento. O relacionamento entre os atores do sistema e os casos de uso mapeados no pacote de acesso e ajuda estão representados na Figura 3.3.

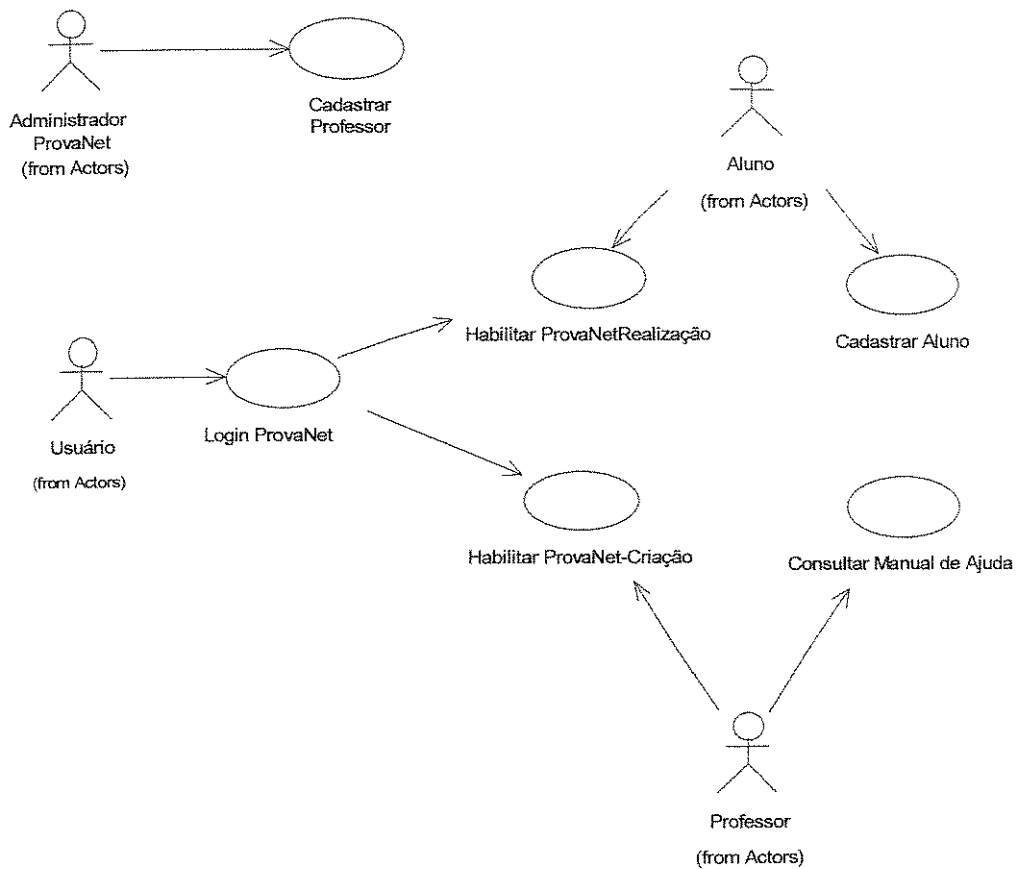


Figura 3.3: Diagrama de caso de uso - pacote acesso e ajuda.

As funcionalidades mapeadas neste pacote são:

### A. Login ProvaNet

Permite o acesso ao sistema e direciona para o ambiente apropriado de acordo com o perfil do usuário que está realizando o *login* no sistema

### B. Cadastrar Professor

Possibilita o cadastramento de professores interessados em utilizar o ProvaNet para disponibilizar provas em determinada disciplina. Este cadastramento, realizado pelo administrador do sistema, é inicializado quando

um professor acessa o sistema ProvaNet e solicita o cadastramento através do envio de um *email*, disponibilizado na página de cadastramento de professores do ProvaNet.

Com base nos dados fornecidos, o administrador cadastra este professor no sistema, especificando o *login* e a senha. Em seguida, envia um *email* de resposta para o usuário que solicitou o cadastramento fornecendo os dados necessários para o acesso ao sistema.

### **C. Cadastrar Aluno**

Permite o cadastramento de alunos que desejam utilizar o sistema ProvaNet para realizar as provas disponibilizadas pelos professores em determinadas disciplinas. Este cadastramento é realizado pelo próprio aluno em uma tela específica do aplicativo, na qual o *login* e a senha desejados são informados. Ao completar-se o formulário, o sistema armazena os dados fornecidos, permitindo o acesso imediato do aluno ao sistema.

### **D. Consultar Manual de Ajuda**

O ProvaNet apresenta tópicos de ajuda que fornecem todas as informações necessárias para a manipulação correta do sistema. Todo o conteúdo informativo pode ser acessado enquanto o sistema está sendo utilizado, pois é aberta uma nova janela para visualização do manual. O manual de ajuda é disponibilizado apenas para os professores.

### **E. Habilitar ProvaNet - Ambiente de Criação**

Permite o acesso ao ProvaNet para professores previamente cadastrados no sistema. O acesso se faz através do fornecimento do *login* e senha, enviados através de *email* enviado pelo administrador. Se os dados inseridos forem validados pelo sistema, o professor tem acesso ao ambiente de criação de provas.



## F. Habilitar ProvaNet - Ambiente de Realização

Permite o acesso ao sistema ProvaNet para alunos previamente cadastrados. Para que um aluno possa utilizar o aplicativo, deve fornecer o *login* e a senha que, se validados, permitem a entrada no ambiente de realização de provas.

### 3.4.2. Pacote Questão

Este pacote contém todas as funções de gerenciamento de questões do ProvaNet. Estas funcionalidades estão disponibilizadas apenas para o professor, único ator deste pacote.

A Figura 3.4 representa os casos de uso englobados por este pacote, descritos a seguir:

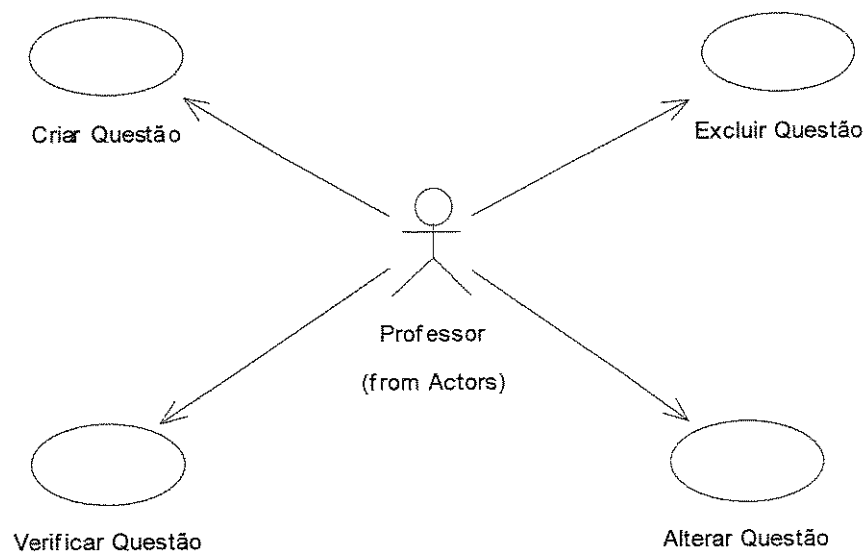


Figura 3.4: Diagrama de caso de uso - pacote questão.

## **A. Criar Questão**

Permite que o professor realize a criação de uma questão no sistema ProvaNet através da definição: da disciplina e do assunto da questão; do enunciado, que pode ser fornecido através da entrada de dados no sistema ou estar em um arquivo previamente criado pelo professor; das alternativas, que podem ter o texto inserido na entrada de dados do aplicativo ou contido no arquivo com o enunciado da questão; e do valor das alternativas como certas ou erradas.

## **B. Verificar Questão**

Possibilita que um professor possa visualizar uma questão previamente criada, fornecendo primeiramente a disciplina e o assunto associados à questão. Com estes dados, o sistema disponibiliza uma lista com todos os enunciados que atendem estas condições, permitindo assim que o professor escolha a questão desejada para visualização.

Todos os dados da questão são apresentados: disciplina, assunto, enunciado, alternativas e valor das respostas da questão.

## **C. Excluir Questão**

Permite que uma questão seja removida do sistema ProvaNet, na qual o professor deve identificar inicialmente a disciplina e o assunto da questão. Em função destes dados, o sistema apresenta uma lista com o enunciado de todas as questões criadas que possuem estas características e o professor escolhe, dentre as questões apresentadas, aquela que deseja eliminar.

Para que uma questão possa ser efetivamente removida, o sistema verifica se esta questão está inserida em alguma prova e se estiver, o sistema solicita uma confirmação. Se o usuário realmente confirmar a remoção, o ProvaNet atualiza todas as provas que possuem esta questão, mantendo a consistência dos dados.

#### **D. Alterar Questão**

Possibilita que uma questão seja alterada, e para isso o professor deve identificar a disciplina e o assunto da questão. Com estes dados, o sistema apresenta uma lista das questões criadas com estas características, exibindo parte do enunciado.

O professor deve escolher uma dentre as questões apresentadas e o sistema apresenta todos os dados inseridos na questão. Para alterar a questão, professor tem a opção de substituir o arquivo inicialmente inserido como enunciado por outro; alterar o texto inicialmente fornecido como enunciado; substituir um texto fornecido como enunciado por um arquivo previamente criado; alterar o texto e a resposta das alternativas e inserir ou remover uma alternativa.

A seguir, são apresentados os diagramas que descrevem as atividades relacionadas à criação de questões. As Figuras 3.5 e 3.6 mostram o diagrama de fluxo de atividades e o diagrama de seqüência, respectivamente, com o detalhamento da realização dos casos de uso do pacote questão.

No fluxo detalhado na Figura 3.5, estão apresentadas as atividades realizadas pelo professor que permitem a criação de uma questão. O professor estabelece a disciplina da questão, para em seguida determinar o assunto e o enunciado, definindo qual o tipo desejado. Em função da escolha, ocorre o desvio para um dos caminhos possíveis para definição do enunciado, e finalmente a confirmação da criação da questão pode ser feita.

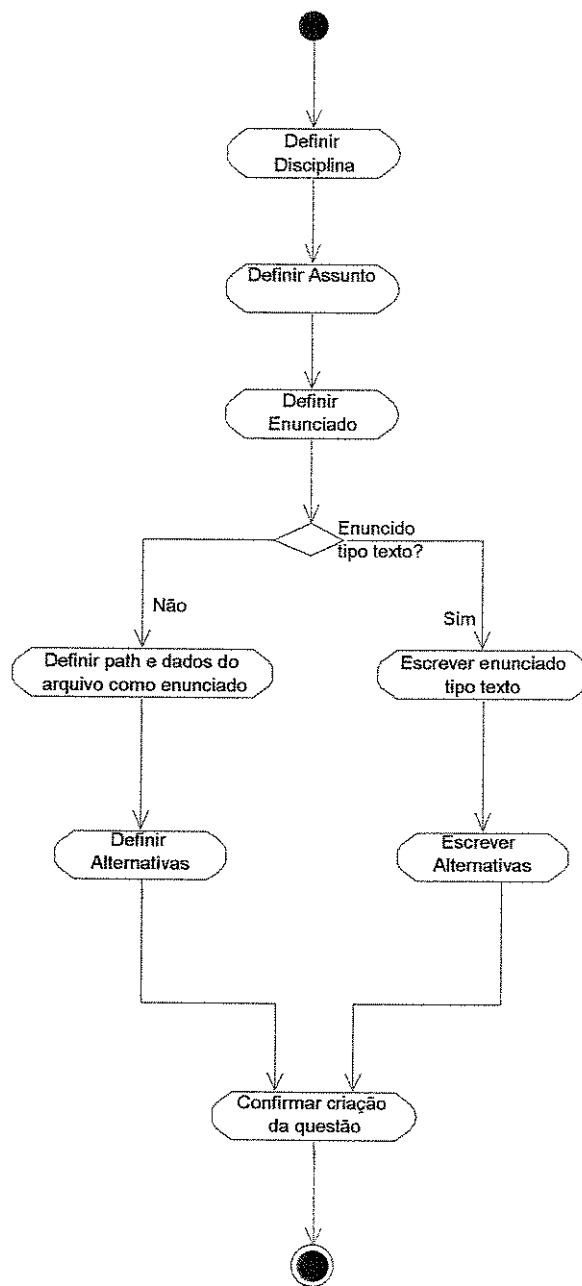


Figura 5.6 Diagrama de fluxo de atividades - criação de uma questão.

A figura 5.6 apresenta a colaboração entre o ator e os objetos envolvidos na criação de uma questão. Para realizar a criação de uma questão, o professor define a disciplina, o assunto, o enunciado e o número de

alternativas da questão. Estes eventos são representados pelas mensagens "definir dados da questão", "definir enunciado" e "definir número de alternativas" enviadas pelo professor para o objeto "questão enunciado".

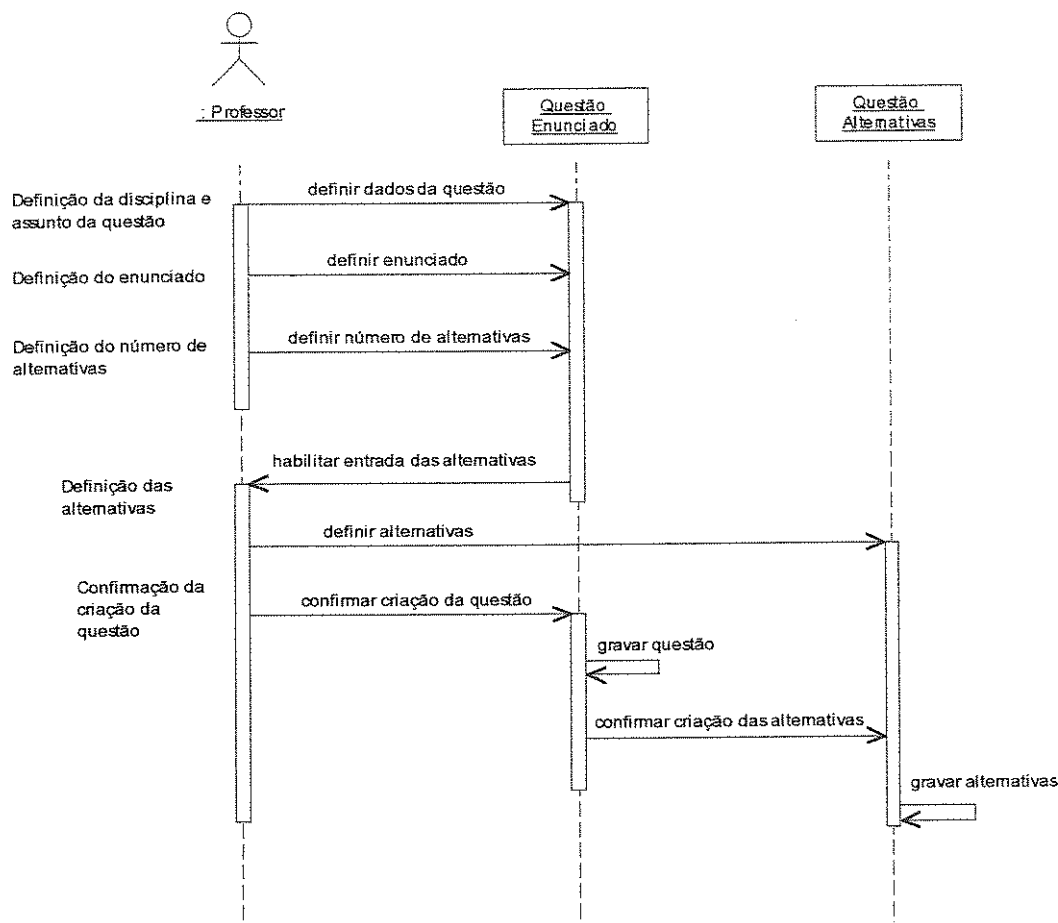


Figura 3.6: Diagrama de seqüência - criação de uma questão.

Em função do número de alternativas estabelecido para a questão, as entradas para a definição do enunciado das alternativas são habilitadas. O professor confirma a criação da questão, o enunciado da questão é gravado no objeto "questão enunciado" e em seguida são gravadas as alternativas no objeto "questão alternativas".

### 3.4.3. Pacote Prova

Este pacote contém as funções de gerenciamento de prova do ProvaNet. Estas funcionalidades estão disponibilizadas apenas para o professor, único ator deste pacote.

O relacionamento entre o professor e os casos de uso mapeados no pacote prova estão representados na Figura 3.7.

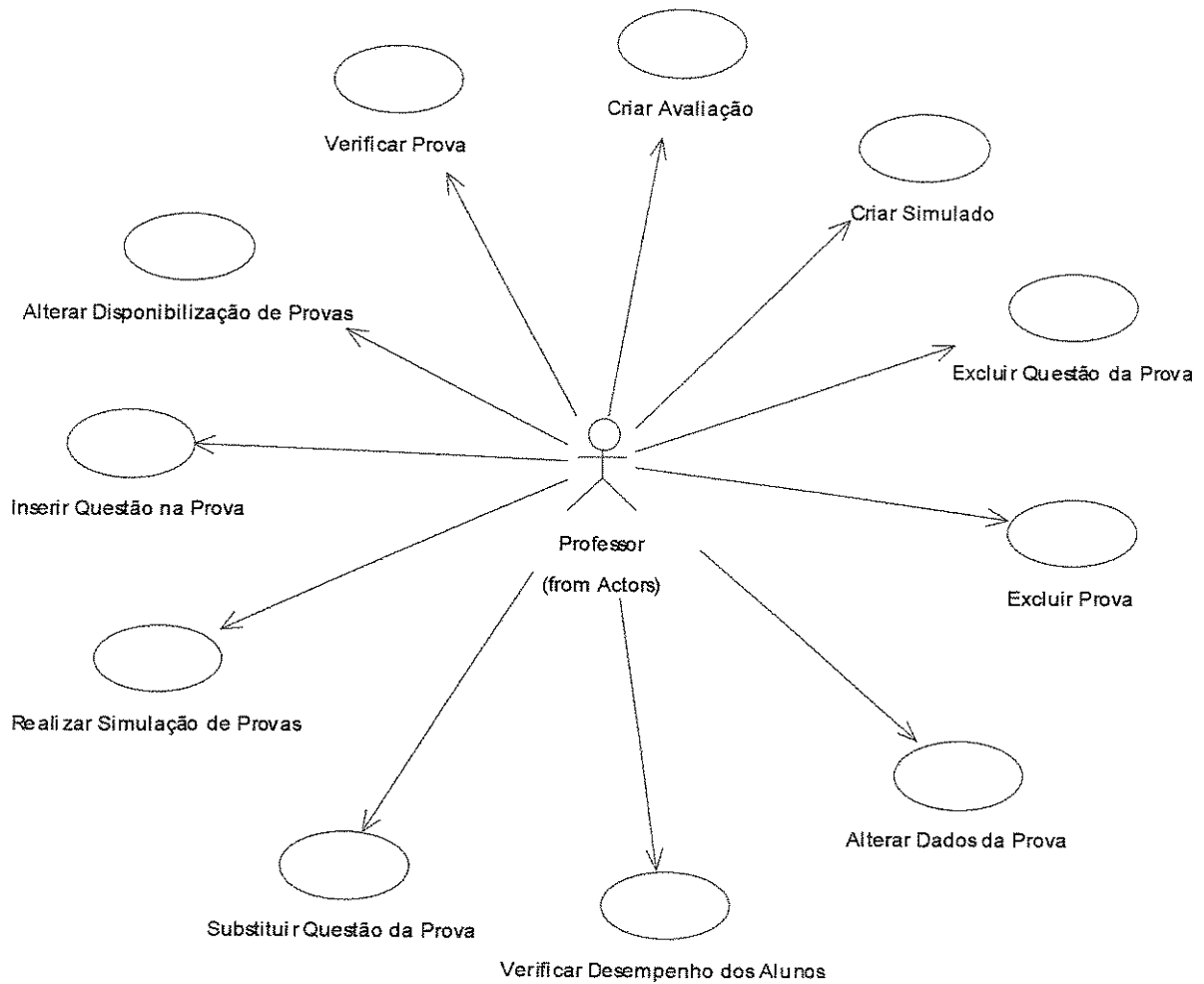


Figura 3.7: Diagrama de caso de uso - pacote prova.

## **A. Criar Avaliação**

Possibilita a criação de uma avaliação no ambiente ProvaNet através da definição inicial do tipo de prova, na qual o professor seleciona como tipo de prova uma avaliação; da definição da disciplina da qual fará parte; da definição do nível e peso da prova. O sistema deve armazenar a data e horário de criação, para posterior identificação por parte do professor.

As questões são inseridas de acordo com a disciplina e assunto, escolhidas em uma lista que exibe parte dos enunciados. O professor escolhe uma dentre as questões apresentadas e o sistema insere a questão na prova. A ordem de disponibilização obedece à ordem de inserção e uma questão só pode ser inserida uma vez na mesma prova.

O número de questões inseridas é arbitrário e, ao finalizar uma prova, o sistema oferece ao professor a possibilidade de disponibilização imediata aos alunos, segundo a seqüência de criação. A habilitação também pode ser feita em um momento posterior.

## **B. Criar Simulado**

A criação de um simulado ocorre da mesma forma que a criação de uma avaliação, porém o professor define como tipo desejado de prova o item "simulado".

## **C. Verificar Prova**

Permite que um professor possa visualizar uma prova previamente criada, fornecendo primeiramente a disciplina. Em seguida, o ProvaNet disponibiliza uma lista com a data de criação de todas as provas da disciplina, permitindo a escolha, e todos os dados da prova são apresentados, detalhando a disciplina e data de criação, questões inseridas e ordem escolhida para disponibilização aos alunos.

#### **D. Alterar Dados da Prova**

Possibilita que os dados de identificação de uma prova possam ser alterados; para isso, o professor deve identificar primeiramente a disciplina desejada. O sistema apresenta uma lista das provas criadas com esta característica e o professor escolhe uma dentre as provas apresentadas. O professor tem a opção de substituir peso, nível e a data de criação da prova.

#### **E. Inserir Questão na Prova**

Permite acrescentar um número arbitrário de questões em uma prova. Esta ação se faz através da definição do assunto, e a seleção das novas questões se faz através da escolha dentre as apresentadas.

#### **F. Excluir Prova**

Permite a remoção de uma determinada prova. O sistema verifica, primeiramente, se a prova está ou não disponibilizada aos alunos. Em caso afirmativo, o ProvaNet solicita uma confirmação para que esta prova possa ser efetivamente removida. Uma vez confirmada a eliminação de uma prova, a ordem de todas as demais provas é atualizada e a consistência dos dados é realizada.

#### **G. Substituir Questão da Prova**

Permite que o professor substitua uma questão, inicialmente inserida na prova, por outra. Para isso, deve escolher qual questão da prova deseja substituir e definir a nova questão, sempre procurando por disciplina e assunto.



## **H. Excluir Questão da Prova**

Possibilita a remoção de uma questão em uma prova previamente criada. Para isso, o professor deve escolher a questão a ser removida na prova.

## **I. Alterar Disponibilização de Provas**

Permite a alteração da ordem de disponibilização de uma prova criada. O ProvaNet disponibiliza todas as provas criadas numa mesma disciplina para que o professor possa alterar a ordem previamente estabelecida. O sistema permite que somente provas com questões inseridas possam ser disponibilizadas aos alunos.

## **J. Verificar Desempenho dos Alunos**

Possibilita que o professor verifique o desempenho individual ou coletivo dos alunos nas provas realizadas. Escolhendo uma prova específica, o professor obtém a média geral e as notas individuais dos alunos.

## **K. Realizar Simulação de Provas**

Permite que o professor possa simular a realização das provas criadas, sem que o desempenho seja contabilizado.

A seguir, são mostrados os diagramas que descrevem as funcionalidades relacionadas à criação de provas e à inserção de questões em provas.

A Figura 3.8 mostra o detalhamento do fluxo de atividades na criação de uma prova, na qual estão representadas as atividades realizadas. O professor define os dados da prova e de acordo com o tipo de prova escolhido, dois caminhos são permitidos, representados pelas atividades "criar avaliação" e "criar simulado". Em seguida as questões são inseridas e a prova é disponibilizada e finalizada.

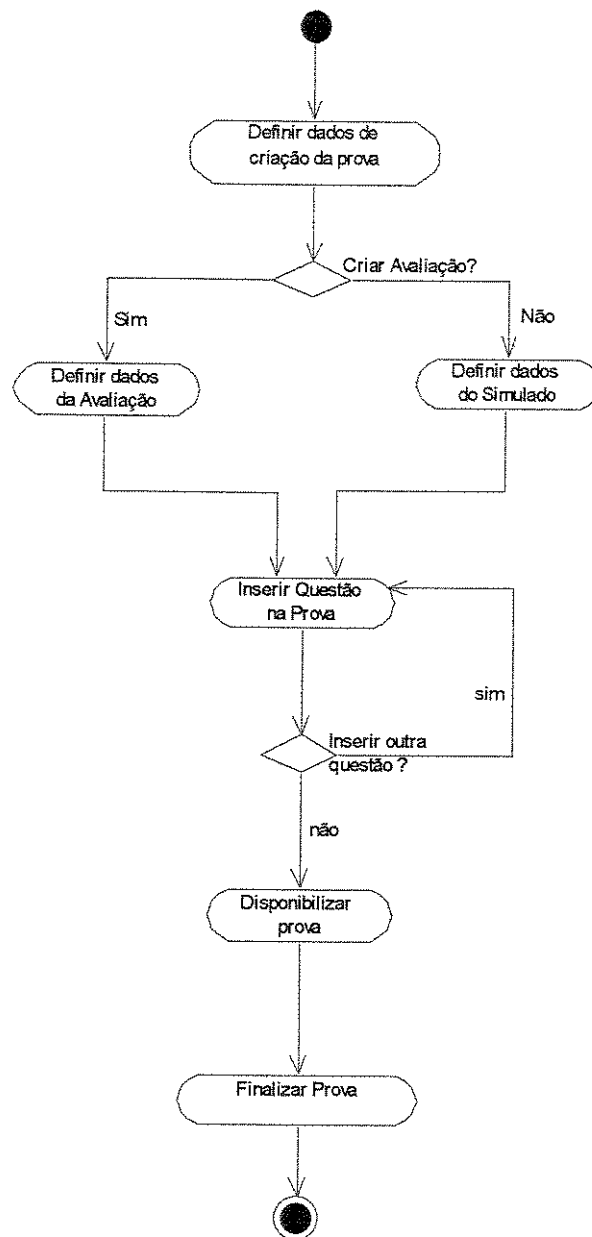


Figura 3.8: Diagrama de atividades - criação de uma prova.

O diagrama mostrado na Figura 3.9 apresenta a seqüência de atividades realizadas para a inserção de questões em uma prova. No fluxo, estão representados os eventos disparados pelo professor que permitem a inserção

de questões em uma prova. O professor define a disciplina e o assunto e, em função destes dados, escolhe a questão a ser inserida. Caso o professor solicite, são apresentados os detalhes, para em seguida ser definido o peso da questão. Estes eventos são realizados quantas vezes forem inseridas as questões na prova.

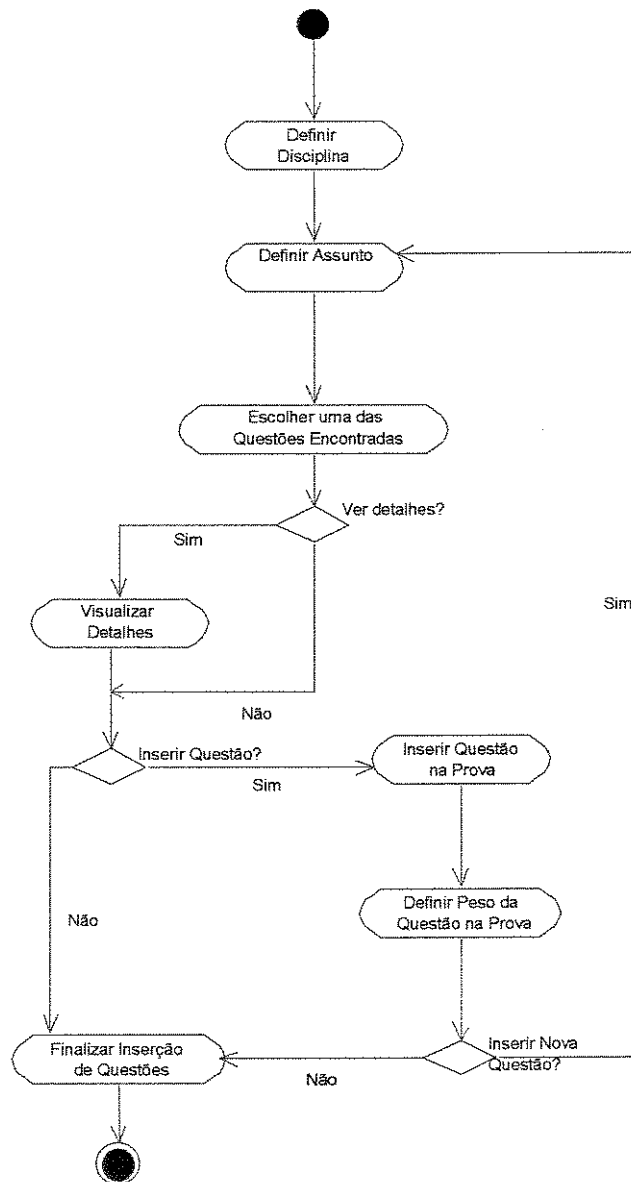


Figura 3.9: Diagrama de atividades - inserção de questões em uma prova.

A Figura 3.10 apresenta o diagrama de seqüência de criação de uma prova. No diagrama, está representada a colaboração entre o professor e os objetos envolvidos na criação de uma prova.

Para realizar a criação da prova, o professor define a disciplina, o peso, o nível e o tipo de prova, evento representado pela mensagem "definir dados da prova", enviada pelo professor para o objeto "prova". Em seguida o professor define o assunto, para que sejam visualizadas as questões que se enquadram nos dados fornecidos, e determina a questão a ser inserida na prova.

O professor pode solicitar a visualização dos detalhes, evento representado pela mensagem "visualizar detalhes" ou inserir a questão escolhida imediatamente, que é gravada na prova. Após todas as questões desejadas terem sido inseridas, a mensagem "disponibilizar prova" é disparada, solicitando a definição da ordem de apresentação ao aluno. A ordem é definida e gravada em seguida.

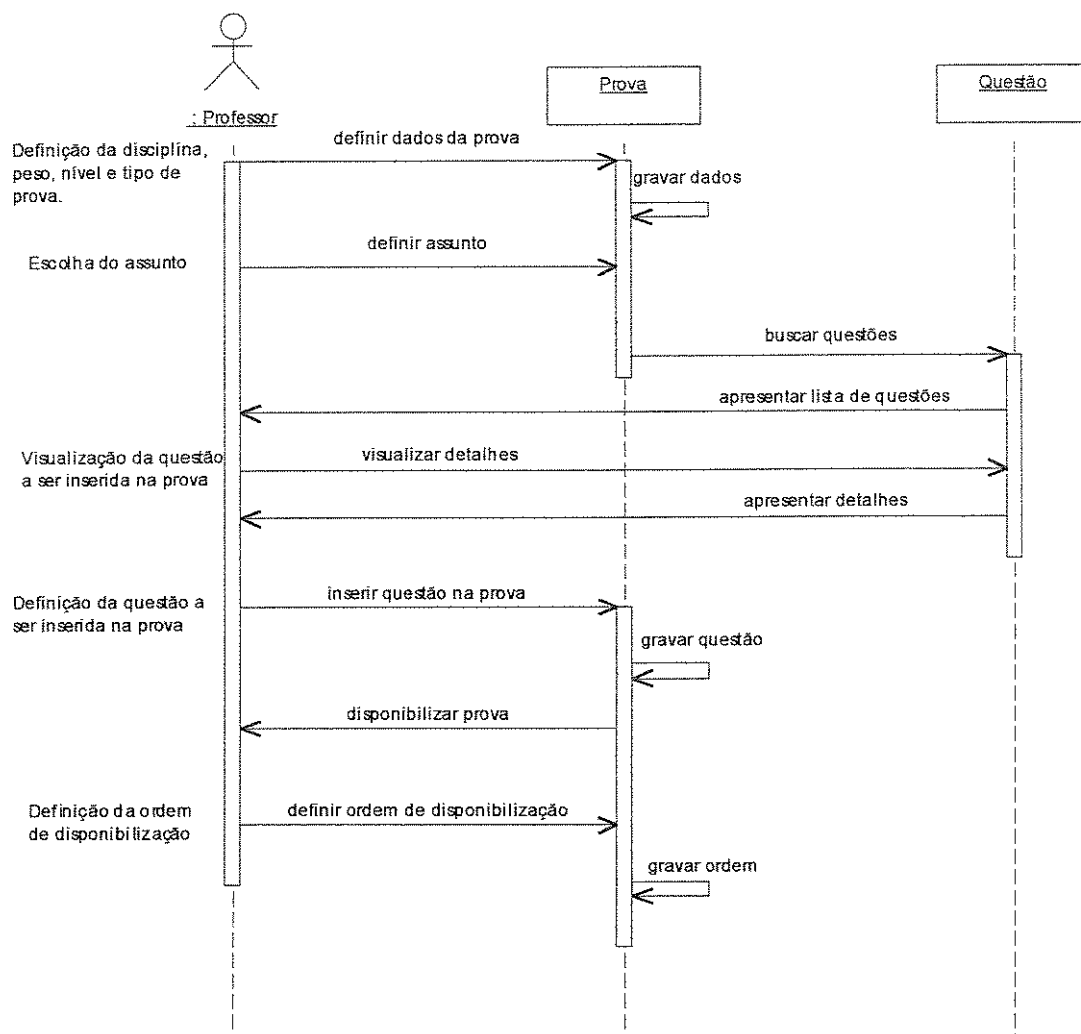


Figura 3.10: Diagrama de seqüência - criação de uma prova.

#### 3.4.4. Pacote Resolução

Este pacote possui as funções para resolução de provas. Estas funcionalidades estão disponibilizadas apenas para o aluno, o único ator deste pacote.

A Figura 3.11 representa o relacionamento entre o aluno e os casos de uso mapeados.

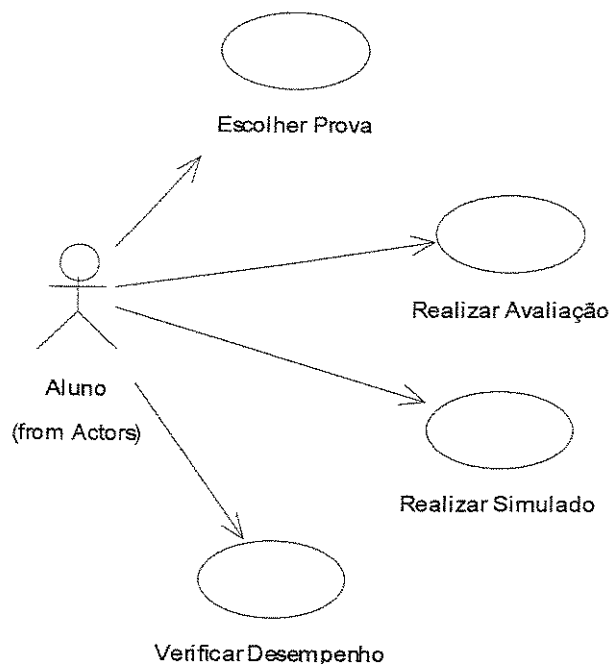


Figura 3.11: Diagrama de caso de uso - pacote resolução.

### A. Escolher Prova

Permite que o aluno tenha acesso ao ambiente de resolução de provas, no qual pode definir a disciplina e o tipo de prova que deseja resolver (simulado ou avaliação). O ProvaNet verifica se existem provas habilitadas pelo professor na disciplina e no tipo escolhidos e, caso não existam provas habilitadas, o sistema exibe uma mensagem de aviso.

### B. Resolver Avaliação

Quando um aluno tem acesso à avaliação disponibilizada pelo professor, a primeira questão da prova é visualizada através da apresentação do enunciado e alternativas definidas pelo professor.

O aluno deve escolher uma entre as alternativas apresentadas e passar para a próxima questão. Assim acontece sucessivamente, até que se esgotem todas as questões inseridas na prova.

A cada avaliação concluída, o ProvaNet calcula e armazena o desempenho, disponibilizando a seguir a próxima prova, que pode ser realizada naquele momento ou em um acesso posterior.

### **C. Realizar Simulado**

O mesmo conceito aplicado na realização de avaliações é válido para os simulados; porém, durante a resolução do simulado, o aluno pode voltar e resolver as questões erradas mais de uma vez. Em simulados, o ProvaNet não contabiliza o desempenho do aluno.

### **D. Verificar Desempenho**

Quando o aluno completa a resolução de uma prova, o sistema calcula a nota final e oferece uma opção para que o aluno verifique a média dos demais alunos que realizaram a prova.

A seguir, são mostrados os diagramas que descrevem as funcionalidades relacionadas à realização de provas.

No fluxo detalhado na Figura 3.12, estão representadas as atividades efetuadas pelo aluno que realiza uma prova (avaliação ou simulado). O aluno define a disciplina e o tipo de prova que deseja realizar e, em função do tipo escolhido, ocorre o desvio para um dos caminhos disponíveis para realização de uma prova, representados pelas atividades "responder questão avaliação" e "responder questão simulado". O fluxo apresentado para a resolução das provas obedece às regras estabelecidas para cada tipo, sendo que ao final da resolução, o aluno pode obter o desempenho geral dos que realizaram a prova.

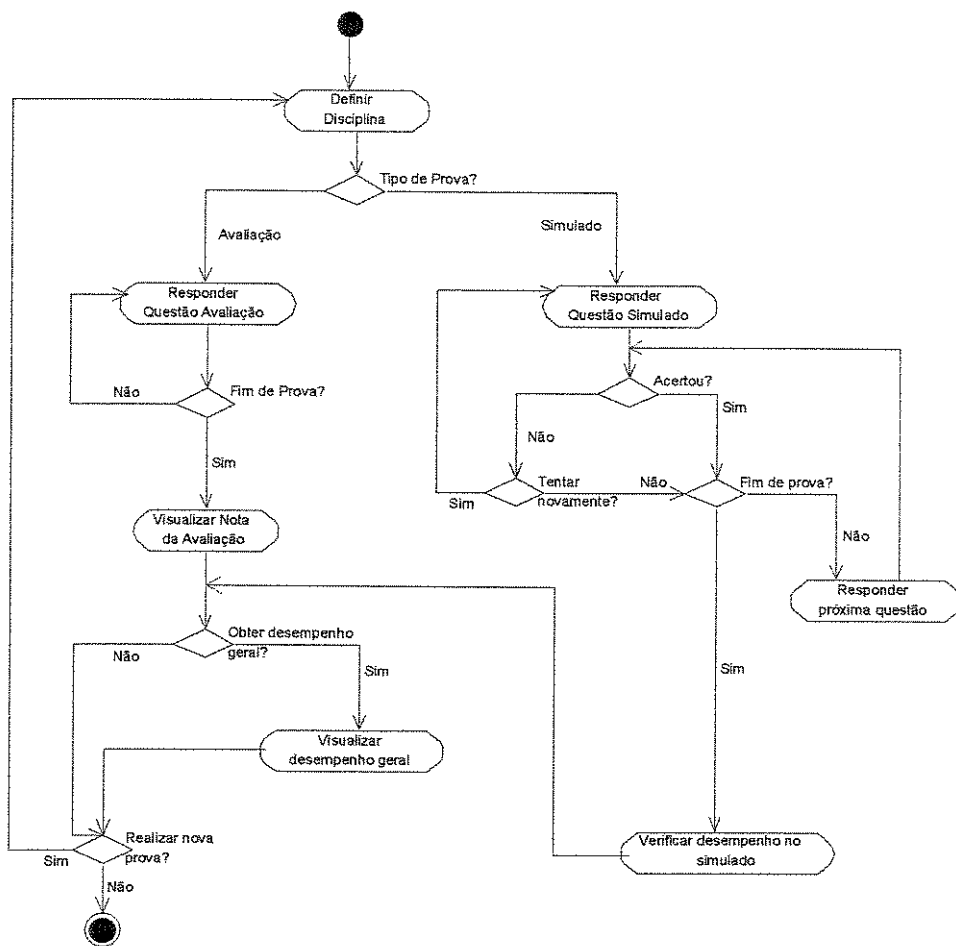


Figura 3.12: Diagrama de atividades - realização de uma prova.

As Figuras 3.13 e 3.14 mostram os diagramas de seqüência para a realização de uma avaliação e de um simulado, respectivamente. Os diagramas apresentam a colaboração entre o aluno e os objetos envolvidos na realização dos dois tipos de prova oferecidos pelo sistema (avaliação e simulado).

Para efetuar a resolução de uma prova, o aluno define qual o tipo e as provas habilitadas são recuperadas e as questões apresentadas. O aluno responde a questão, evento representado pela mensagem "responder questão", que é enviada pelo aluno para o objeto "prova" e a seqüência de mensagens trocadas em seguida obedecem às regras estabelecidas para cada tipo de prova.



No simulado o desempenho não é calculado; na avaliação, quando todas as questões forem respondidas, a nota final é calculada, gravada e apresentada ao aluno; eventos representados pelas mensagens "calcular nota final", "informar nota final" e "apresentar nota final".

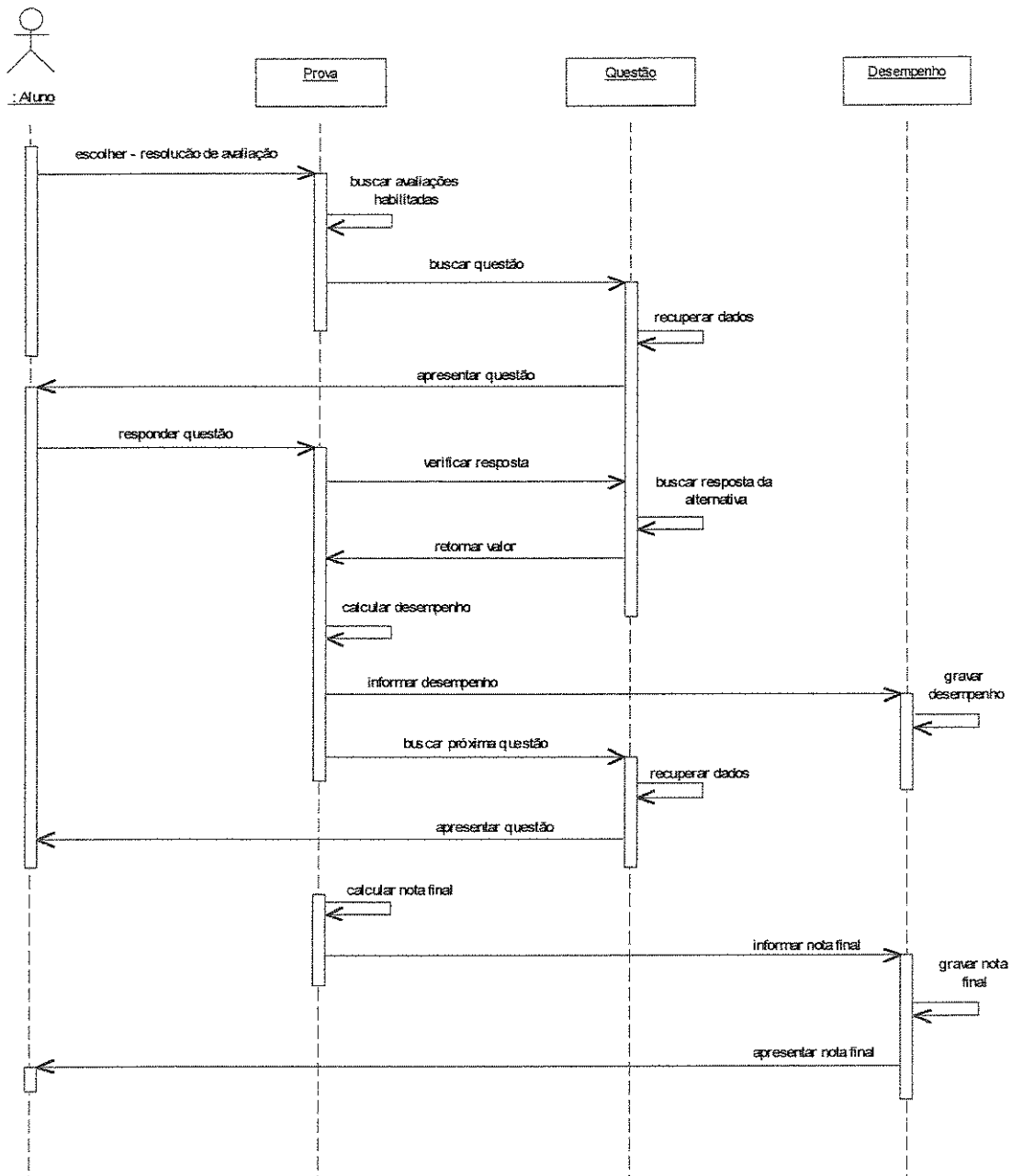


Figura 3.13: Diagrama de seqüência - realização de uma avaliação.

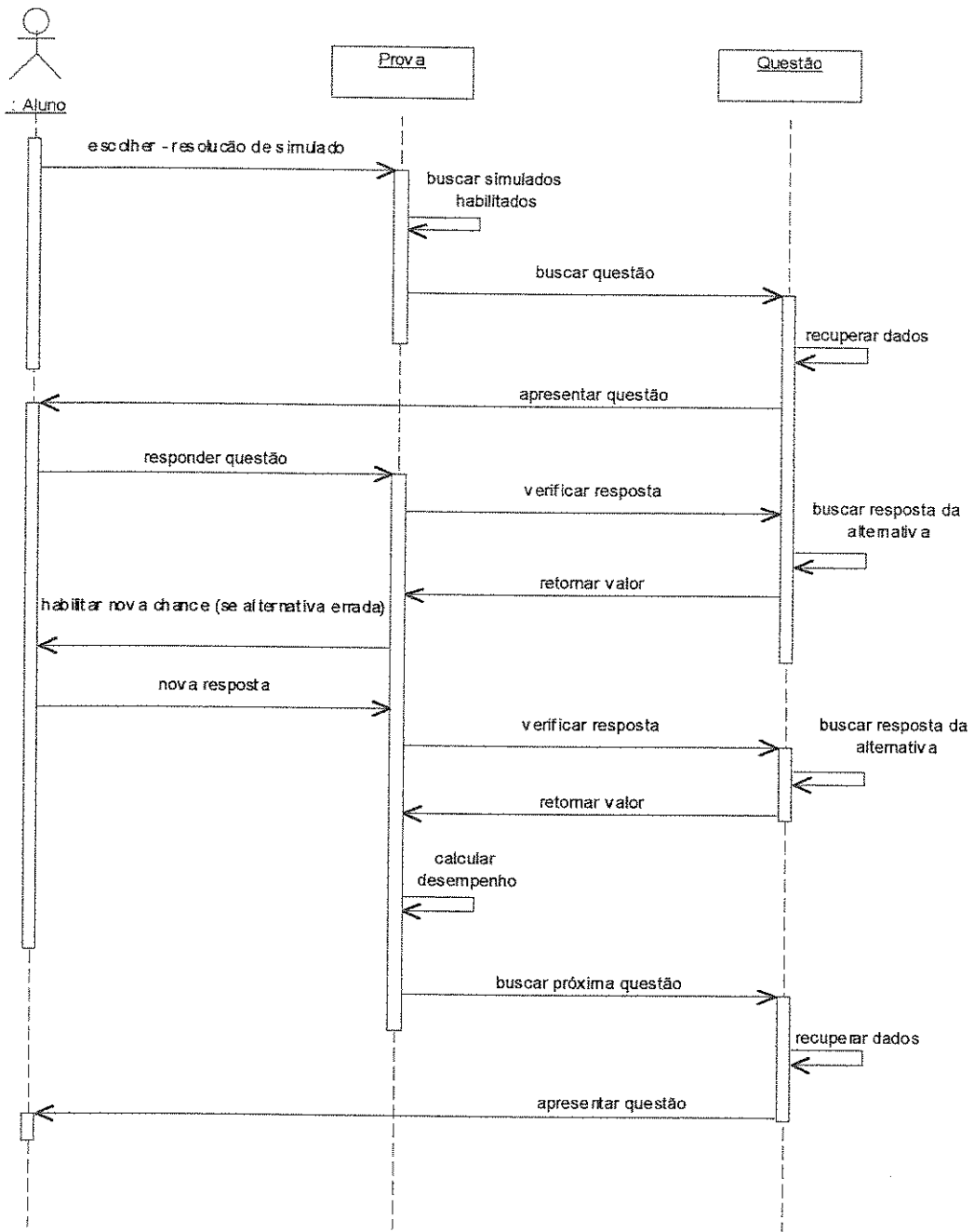


Figura 3.14: Diagrama de seqüência - realização de um simulado.

---

## Capítulo 4

### Desenvolvimento do Sistema ProvaNet

---

#### 4.1. Introdução

Este capítulo tem como objetivo apresentar o projeto do sistema ProvaNet em função da arquitetura concebida para a construção do aplicativo e das ferramentas utilizadas. Aqui são detalhadas as funcionalidades mapeadas, juntamente com o projeto lógico de construção do sistema e o projeto de banco de dados.

#### 4.2. Arquitetura

No modelo de desenvolvimento Web, o trabalho para a elaboração de um aplicativo se concentra nas partes específicas do sistema e na interação entre estas diferentes peças. Componentes já existentes podem ser aproveitados e o desenvolvimento se concentra no projeto das partes proprietárias da aplicação. Não é necessário, por exemplo, a construção de um navegador ou banco de dados, podendo ser utilizados aplicativos existentes que são integrados ao sistema.

A construção dos elementos do sistema foi baseada no conceito *Three-Tiered Architecture* (arquitetura em três camadas). O conceito das três camadas surge de uma análise funcional dos elementos ou serviços do sistema que exibem um comportamento similar e são, desta forma, agrupados de acordo com suas propriedades. As camadas ajudam a identificar, definir e, potencialmente, restringir como os elementos de cada camada interagem uns com os outros [Ambler, 1998], [Rumbaugh et al., 1994].

A arquitetura em três camadas de serviços é constituída pela camada de apresentação, pela camada de negócio e pela camada de dados.

A camada de apresentação contempla a interface com o usuário, oferecendo os meios para o acesso e a interação com o sistema, permitindo a visualização das informações na tela da estação de trabalho.

Os serviços da camada de apresentação consistem em programas que são executados no terminal do usuário. No caso da Internet, a interface básica do usuário é o navegador (*browser*). No entanto, podem existir outros componentes que são executados nesta camada. Por exemplo, o aplicativo pode necessitar receber certas informações fornecidas pelo usuário em um formato pré-estabelecido, sendo necessária a execução de um *script* que filtre os dados de entrada. Assim, ao abrir um arquivo enviado pelo servidor Web, o *browser* desempenha dois papéis básicos: apresentar a tela estabelecida pelo sistema, interpretando o código HTML e exibindo a tela definida para o usuário; e executar o código contido neste arquivo sob a forma de *scripts*.

A camada de negócio implementa as regras específicas do sistema e o fluxo de dados entre as camadas de apresentação e de dados. Ao atender uma solicitação do usuário, ou de outros serviços, esta camada executa uma tarefa específica do aplicativo, com o objetivo de realizar uma funcionalidade projetada de acordo com as necessidades de quem o utiliza. O desenvolvimento de componentes ou roteiros para realização dos serviços do sistema pode ser feito através de um programa a ser executado no servidor Web.

A camada de dados armazena e fornece as informações do sistema. Os serviços desta camada definem, acessam e alteram as informações de acordo com as solicitações efetuadas pelo aplicativo. A camada de dados pode ser fisicamente implementada através de um único sistema de banco de dados ou por uma coleção heterogênea, através de múltiplas plataformas ou mesmo através de uma combinação de máquinas de grande porte e de servidores de bancos de dados.

A Figura 4.1. representa a divisão e a comunicação entre as três camadas.

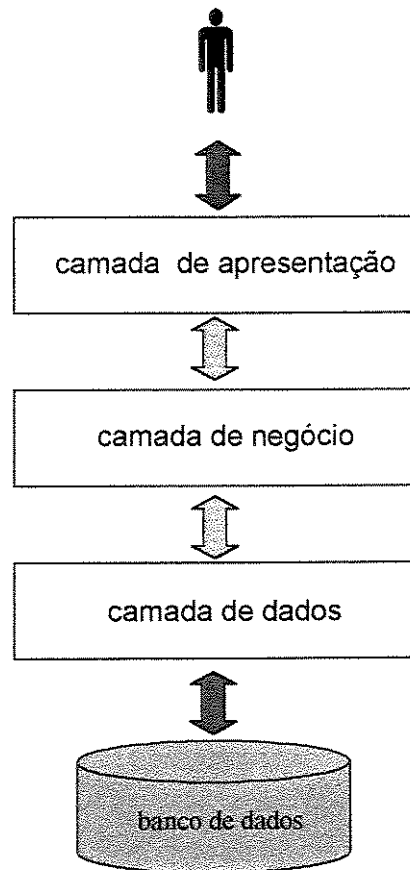


Figura 4.1: Arquitetura em três camadas.

As camadas de interface, de negócio e dados podem ser modeladas através da definição de classes, que são uma categoria de objetos semelhantes que encapsulam informação e comportamento, e fornecem os requisitos para a criação de objetos de um sistema [Ambler, 1995], [Boggs e Boggs, 1999], [Pressman, 1995], [Rumbaugh et al., 1994].

A divisão em três camadas de serviços é lógica e não física. A Figura 4.2 apresenta os serviços concebidos no ProvaNet divididos em camadas.

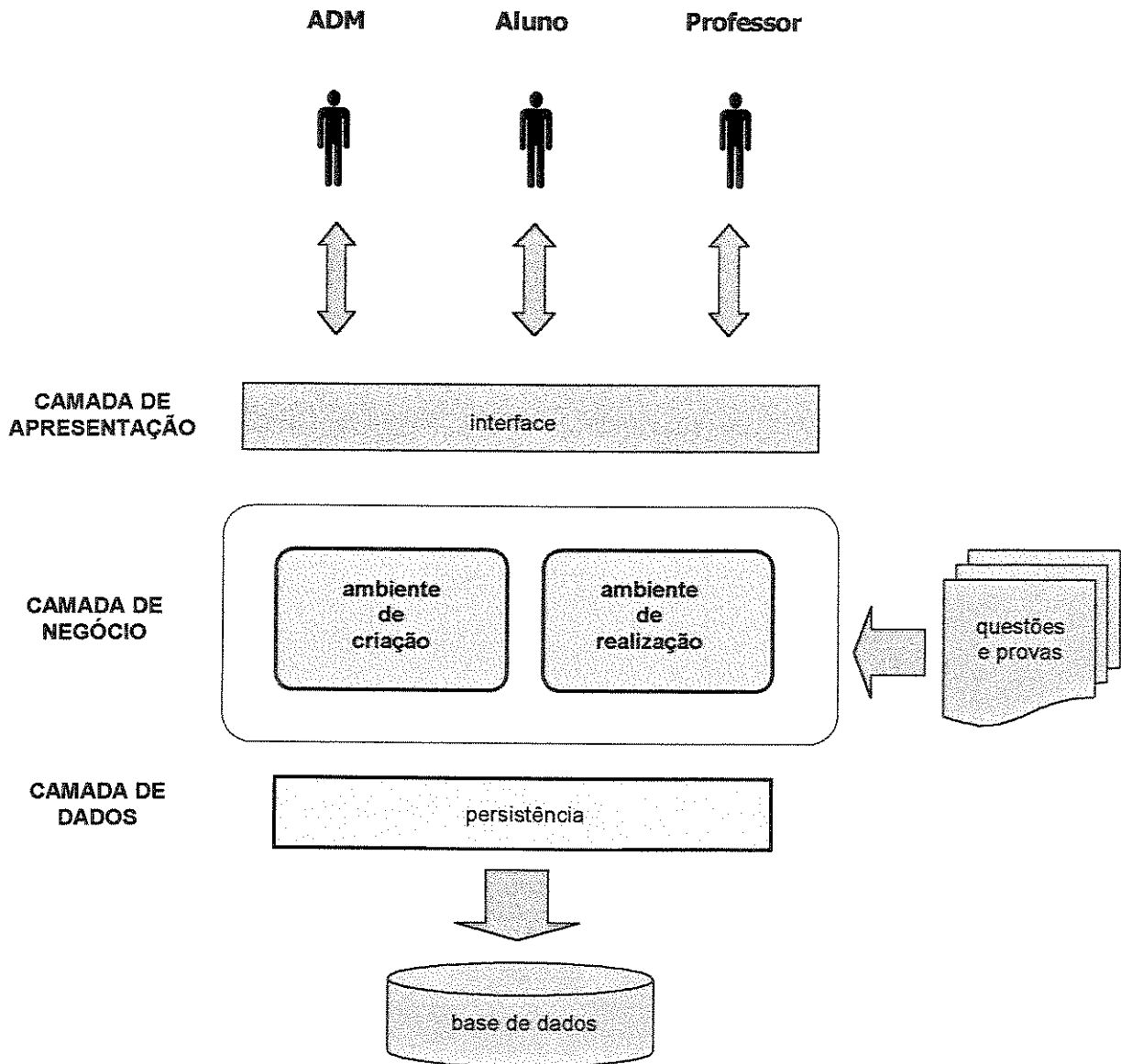


Figura 4.2: Arquitetura do ProvaNet.

Na camada de apresentação estão localizados os componentes referentes às telas de manipulação do sistema e à entrada de dados fornecidos por professores e alunos. Na camada de negócio estão as funções referentes ao gerenciamento de questões e provas e na camada de dados estão as funções que gerenciam a persistência dos dados utilizados no ProvaNet.

### **4.3. Ferramentas de Construção**

Diversas tecnologias para construção de aplicativos baseados em ambiente Internet poderiam ter sido utilizadas no desenvolvimento do ProvaNet.

Entre as mais utilizadas, se coloca a linguagem de programação Perl (*Practical Extraction and Report Language*), que tornou-se uma das tecnologias mais difundidas para a construção de aplicativos Web devido à facilidade de geração de documentos dinâmicos, manipulação de valores e cadeias de caracteres, além do suporte a expressões, essencial para extração e validação de dados. A linguagem Perl, entretanto, possui um sintaxe densa, na qual um comando de execução não raro se torna de difícil compreensão.

Outra alternativa é dada pelo PHP (*Hypertext Preprocessor*), uma linguagem de domínio público. Para o desenvolvimento do ProvaNet optou-se pela utilização do PHP pela sintaxe simples e facilidade de manipulação.

O XML (*Extensible Markup Language*) é outra tecnologia atualmente muito difundida no desenvolvimento de aplicativos Web [XML, 2001]. Esta linguagem é bastante utilizada principalmente por estar se tornando um padrão para a conservação de dados, ou seja, o XML pode ser usado como auxílio a qualquer programa de computador (ou páginas da Web) mantendo os dados armazenados.

A seguir são apresentadas as ferramentas utilizadas na construção do ProvaNet.

#### **4.3.1. JavaScript**

O JavaScript foi desenvolvido com o intuito de tornar as páginas HTML mais interativas. Sendo uma linguagem de *script* baseada em objetos, o JavaScript tem a vantagem de poder ser inserido diretamente no código de uma página, misturando-se com a linguagem HTML [Soares, 2000].

O JavaScript é processado inteiramente pelo *browser*, sendo esta sua principal característica. Estabelecendo-se como uma linguagem do tipo *Client-*

*Side*, muitas tarefas que antes eram feitas somente no lado do servidor (como por exemplo, a validação do preenchimento dos campos de um formulário HTML) passaram a ser feitas no lado do cliente com muito mais eficácia e rapidez. Este procedimento acarreta uma redução de sobrecarga de execuções do tipo *Server-Side*.

Uma das funcionalidades permitidas pelo JavaScript é o teste da consistência de dados de entrada inseridos pelo usuário antes que a aplicação execute a chamada ao servidor. Com isso a execução de aplicações Internet se torna mais rápida.

#### **4.3.2. PHP**

O PHP (*Hypertext Preprocessor*) é uma linguagem de domínio público que é incorporada em código HTML e processada pelo servidor de aplicação Web. Utilizada para criar páginas dinâmicas, tem como ponto forte a compatibilidade com a maioria dos bancos de dados atuais [Soares, 2000], [PHP, 2001].

Quando um código PHP é requisitado, o servidor envia este código primeiramente ao analisador do PHP, que executa todos os comandos e envia o resultado novamente ao servidor, que por sua vez os envia para o interpretador *browser* no cliente, que gera o código totalmente em HTML. Dentre as vantagens de utilização do PHP estão: a velocidade de execução; código nativo para muitos bancos de dados; e facilidade de utilização com grande poder de interatividade [Soares, 2000].

#### **4.3.3. MySQL**

O MySQL (*Structured Query Language*) é um sistema de gerência de banco de dados relacional, sendo que uma base de dados relacional consiste em um banco de armazenamento de dados em tabelas, ligadas através de relações entre seus campos, permitindo que um grande arsenal de informações seja manipulado. Esta característica imprime velocidade e flexibilidade à consulta de dados, agilizando o tempo de processamento de um aplicativo.



MySQL oferece uma gama de opções e capacidades que abrangem, entre outras, a habilidade de manipular um número ilimitado de usuários simultâneos, rápida execução de comandos, simplicidade e eficiência no sistema de privilégio de usuários [MySQL, 2001].

O PHP suporta comandos para acesso a tabelas definidas no banco de dados, fazendo a interface entre as páginas construídas em HTML e os dados tratados pelo MySQL [PHP, 2001].

#### **4.4. Implementação**

O ProvaNet foi desenvolvido utilizando a integração das ferramentas HTML, PHP, JavaScript e banco de dados MySQL [Righetto et al., 2001].

As regras de funcionalidade do ProvaNet, os comandos de acesso à base de dados e manipulação das variáveis resultantes deste processo foram desenvolvidos em linguagem PHP. A construção das tabelas para manipulação das informações foi realizada utilizando o banco de dados MySQL.

##### **4.4.1. Camada de Interface**

Todas as páginas exibidas pelo ProvaNet foram desenvolvidas em HTML e o JavaScript foi utilizado para validação dos campos de entrada de dados, realizando a verificação no ambiente do usuário.

Alguns campos de entrada são validados sem a necessidade de que um formulário seja enviado ao servidor para ser processado. Por exemplo, na criação de uma questão, não é permitido ao professor definir como enunciado uma entrada do tipo ASCII ao mesmo tempo que um arquivo previamente criado; o sistema permite que apenas uma destas possibilidades seja definida. Outro exemplo faz referência à criação de provas; o sistema somente permite que uma prova seja efetivamente criada se todos os campos de identificação estiverem preenchidos.

O conteúdo do manual de ajuda também foi desenvolvido em HTML. Para que a janela de exibição do manual fosse apresentada de forma independente

do aplicativo, permitindo que o professor pudesse visualizá-la sempre, foram utilizados recursos do JavaScript; uma janela filha é criada pelo sistema sempre que esta funcionalidade é requisitada. Desta forma o professor pode navegar pelas telas do ProvaNet e conservar o manual de ajuda sempre aberto para consulta.

As questões criadas no ProvaNet que possuem arquivos de diferentes formatos são visualizadas em diferentes *frames* na tela do aplicativo. Ou seja, a tela de execução do ProvaNet é particionada em duas; na parte superior, é exibido o arquivo atrelado ao enunciado, e na parte inferior são apresentadas as alternativas definidas, como mostra a Figura 4.3.

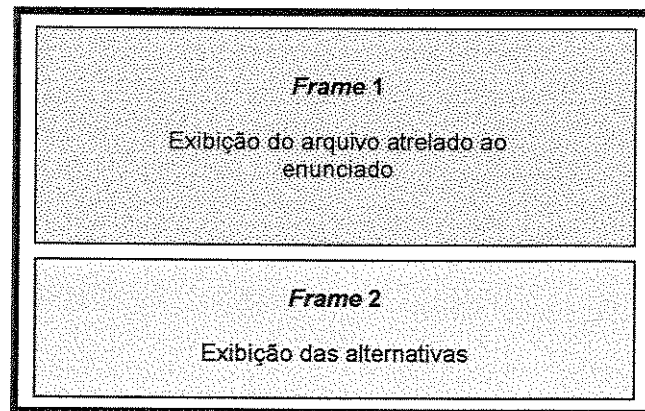


Figura 4.3: Divisão da tela de execução em *frames*.

Para que o sistema saiba qual forma de tela de execução apresentar, foram criadas rotinas em função do tipo de questão a ser exibida. Desta forma, o aplicativo particiona ou não a tela de execução para mostrar de forma correta as questões criadas.

As rotinas de apresentação de questões do aplicativo estão inseridas nas classes de interface que foram identificadas analisando-se os casos de uso mapeados no ProvaNet.

No mínimo, deve existir uma classe de interface para cada interação entre um ator e um caso de uso [Boggs e Boggs, 1999]. A classe de interface permite ao ator interagir com o sistema, como mostrado na Figura 4.4.

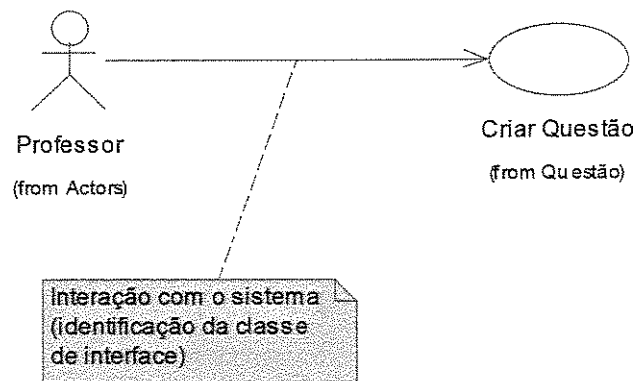


Figura 4.4: Identificação de uma classe de interface.

A Figura 4.5. apresenta as classes concebidas na camada de interface que fazem o gerenciamento de exibição de telas.

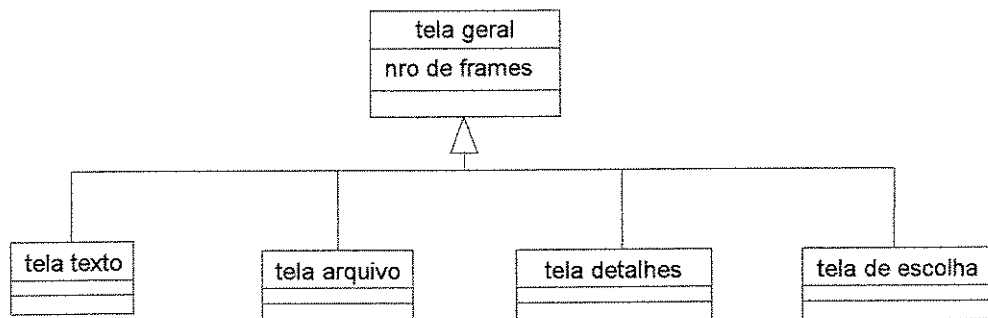


Figura 4.5: Diagrama de classes da camada de interface.

O uso de arquivos do tipo PDF nas questões e provas no ProvaNet exige alguns cuidados técnicos para melhor visualização. O navegador deve ser configurado para apresentar os arquivos na tela de execução do aplicativo, e não para armazená-los no diretório do usuário.

Para isso, um aplicativo que tem por objetivo aumentar a funcionalidade do navegador (*plug-in*) deve ser instalado e associado à execução de arquivos com extensão em PDF.

#### 4.4.2. Camada de Negócio

Na camada de negócio estão inseridas as ações referentes ao comportamento e às regras do sistema ProvaNet. Nesta camada estão mapeadas as funcionalidades de manipulação oferecidas aos professores e alunos.

A Figura 4.6 mostra os objetos criados nesta camada. Este diagrama de classes apresenta o modelo da estrutura do ProvaNet, ilustrando o relacionamento, herança e a colaboração entre os objetos [Pressman, 1995].

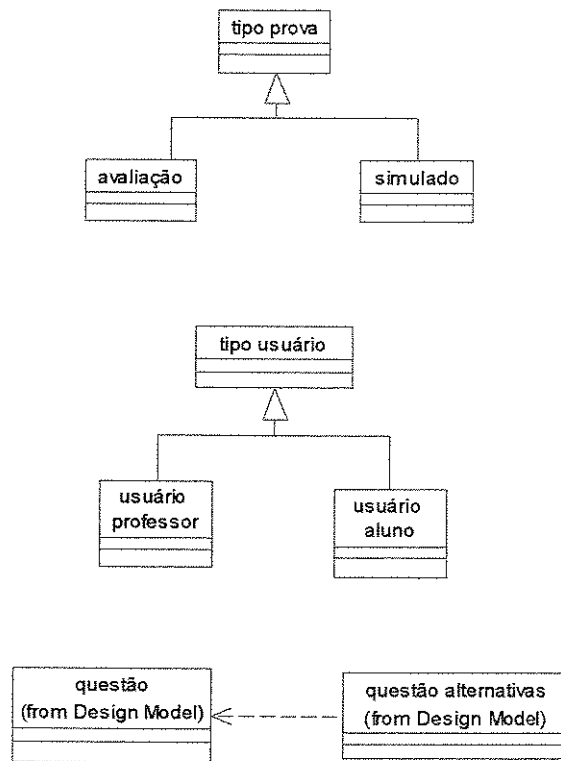


Figura 4.6: Diagrama de classes da camada de sistema.

Toda a funcionalidade da ferramenta ProvaNet foi implementada em PHP, no qual a lógica de processamento é baseada na troca de informações segundo a ação que está sendo realizada no momento. De acordo com uma determinada ação, variáveis específicas são enviadas de um formulário a

outro, sendo que as rotinas em PHP fazem o reconhecimento e executam as funções relativas ao gerenciamento ou armazenamento dos dados de acordo com a ação solicitada [Righetto et al., 2001].

### 4.4.3. Camada de Dados

Na camada de dados estão inseridas as funcionalidades de armazenamento dos dados pelo sistema.

O PHP possui funções próprias de acesso à base de dados MySQL, sistema gerenciador de banco de dados utilizado para a construção das tabelas no ProvaNet.

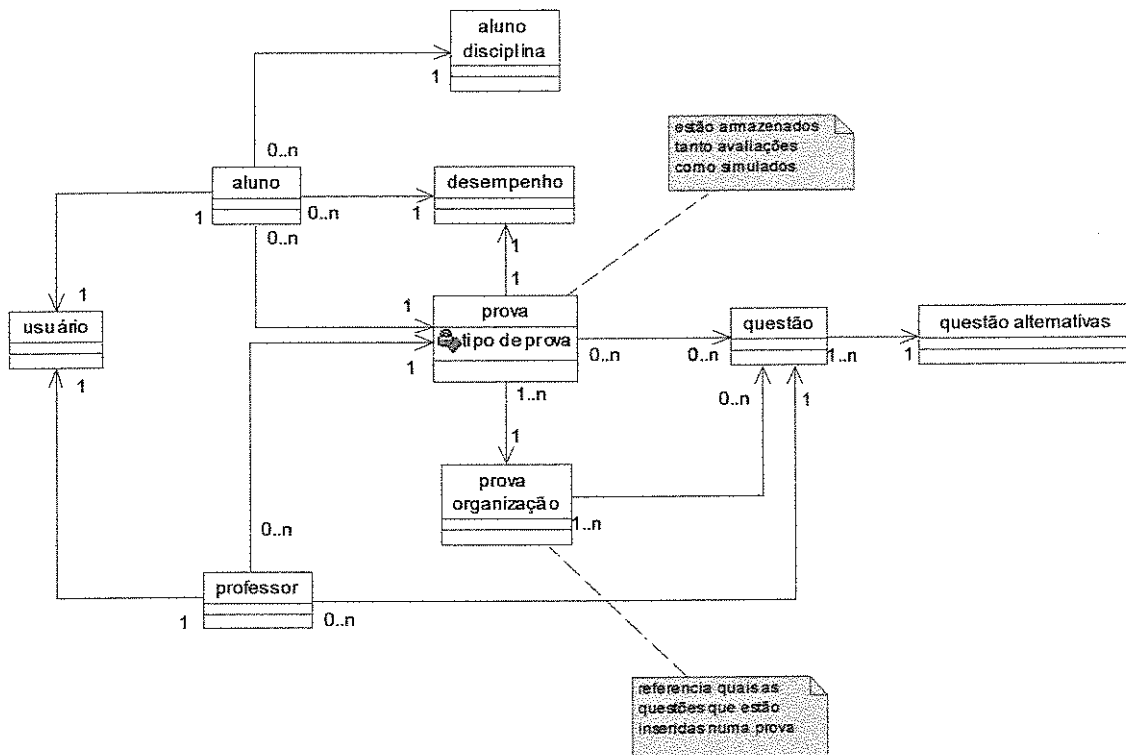


Figura 4.7: Tabelas do ProvaNet.

A Figura 4.7 representa as tabelas criadas no sistema para gerenciamento dos dados manipulados no ProvaNet e o relacionamento entre estas. Por

exemplo, o relacionamento entre as tabelas "questão" e "questão alternativas" é mapeado da seguinte forma: uma questão pode conter uma ou mais alternativas, porém, cada alternativa registrada na tabela pertence somente à uma questão.

O ambiente testa a consistência dos dados inseridos pelos usuários. Por exemplo, se um professor cria uma questão, insere-a em uma prova e posteriormente a exclui da base, o sistema acusa que a questão estava sendo utilizada, e atualiza todas as avaliações que continham a questão. Porém, o sistema não realiza o gerenciamento do acesso aos dados do sistema. Por exemplo, se um professor alterar os dados de uma questão exatamente ao mesmo tempo que um outro usuário, o sistema mantém a base atualizada com a última alteração realizada.

---

## Capítulo 5

### Exemplo de Utilização da Ferramenta ProvaNet

---

#### 5.1. Introdução

Este capítulo tem como objetivo apresentar um exemplo de utilização dos recursos presentes na ferramenta ProvaNet, mostrando em detalhes a manipulação de seus dois ambientes (criação e realização de provas) [Righetto et al., 2001].

O exemplo a seguir apresenta todo o ciclo de manipulação do aplicativo, desde a criação de questões até a realização de uma prova pelo aluno.

O sistema ProvaNet está disponível para utilização na página [http://www.athenas.dt.fee.unicamp.br/vanessa/provanet\\_v2](http://www.athenas.dt.fee.unicamp.br/vanessa/provanet_v2).

## 5.2. Utilização

Quando um usuário tem acesso ao sistema, a página inicial do ProvaNet é habilitada, segundo mostrado na Figura 5.1, que apresenta a tela de entrada para alunos e professores.

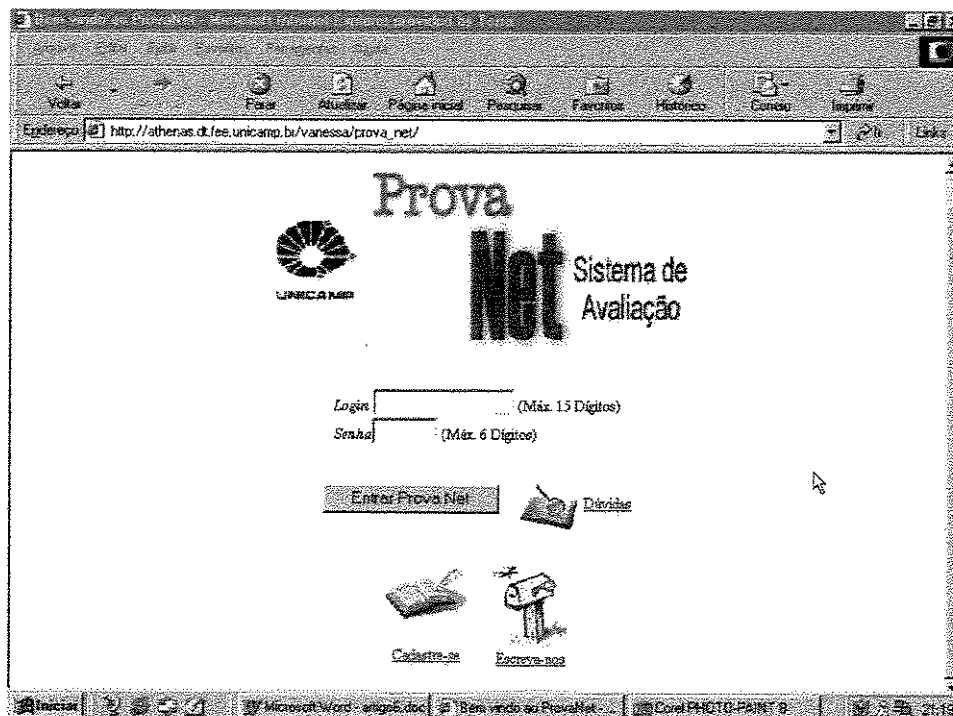


Figura 5.1: Tela inicial do ProvaNet.

### 5.2.1. Acesso e Ajuda

Através da página inicial, o sistema permite o acesso às telas de cadastramento e ajuda. A Figura 5.2. apresenta a tela de cadastramento de um aluno e a Figura 5.3 mostra o manual de ajuda disponibilizado para o professor.



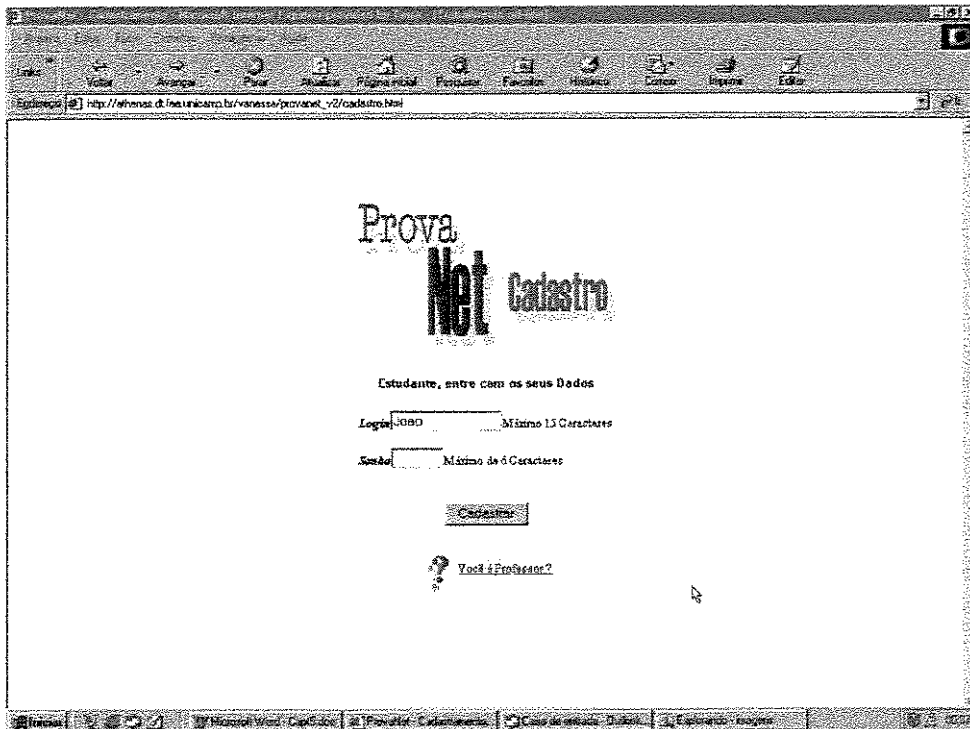


Figura 5.2: Cadastramento de um aluno.

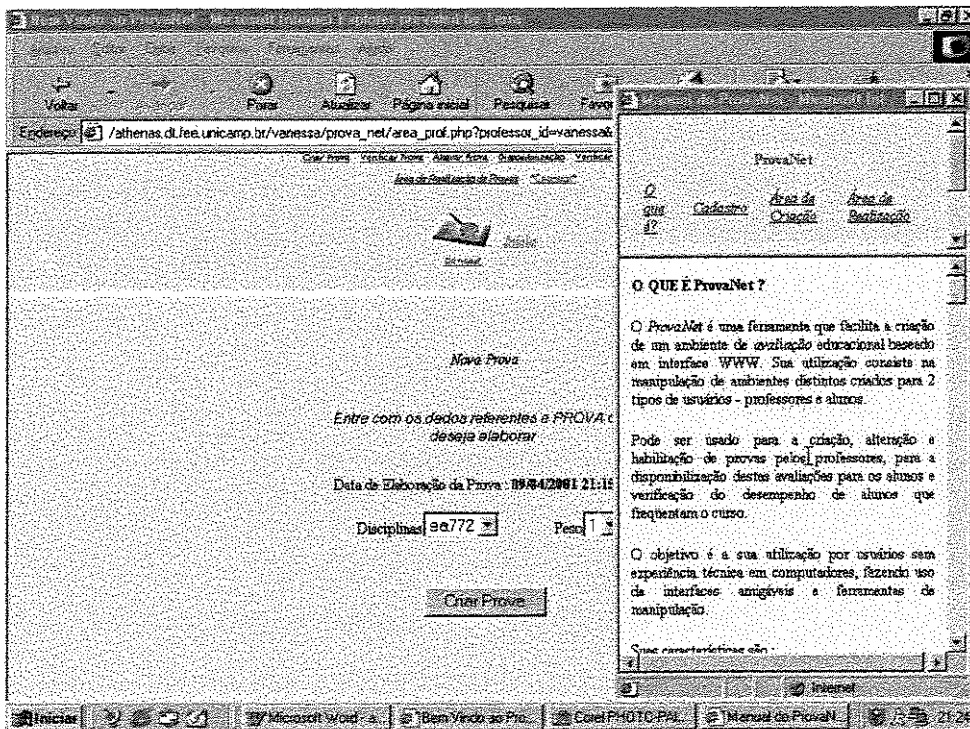


Figura 5.3: Manual de ajuda.

### 5.2.2. Ambiente de Criação de Provas

Ao entrar na área de criação de provas, uma página indicando o acesso é apresentada, como ilustrado na Figura 5.4. Em seguida, a tela de manipulação do ambiente de criação de provas é exibida.

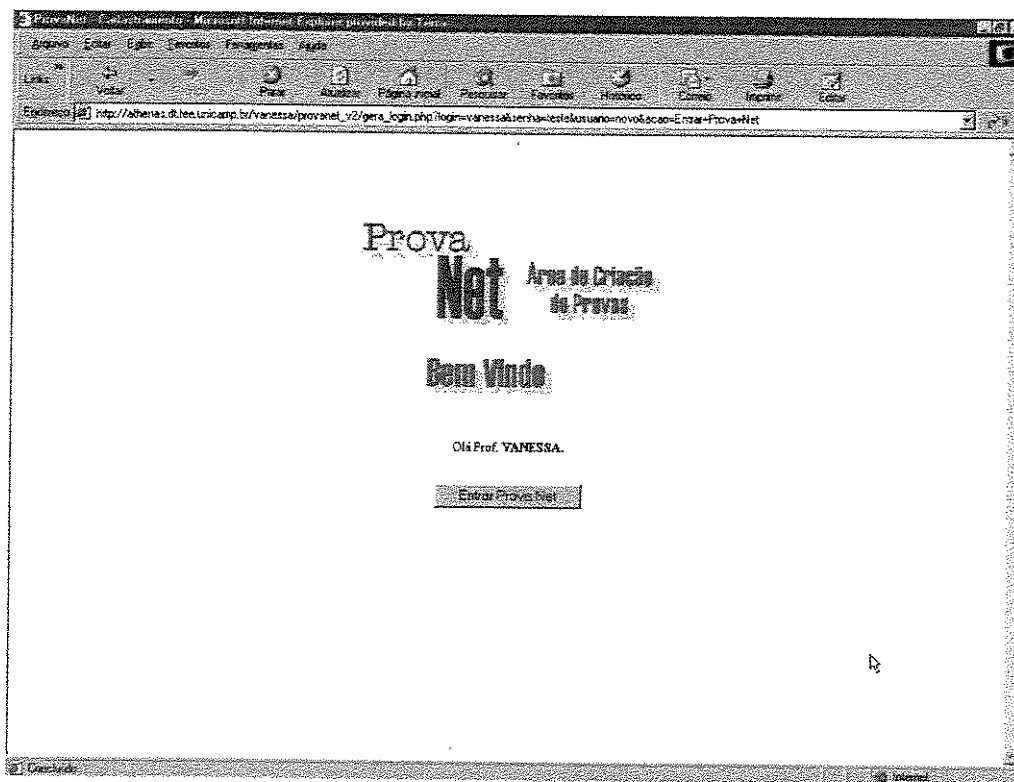


Figura 5.4: Entrada do ambiente de criação.

#### A. Criar Questão

Um total de quatro telas são exibidas para que o professor possa criar uma questão. O enunciado e as diferentes formas de estabelecer as alternativas são disponibilizadas: definição do enunciado (Figura 5.5) e valor das alternativas; ou definição somente do valor das alternativas (Figura 5.6).

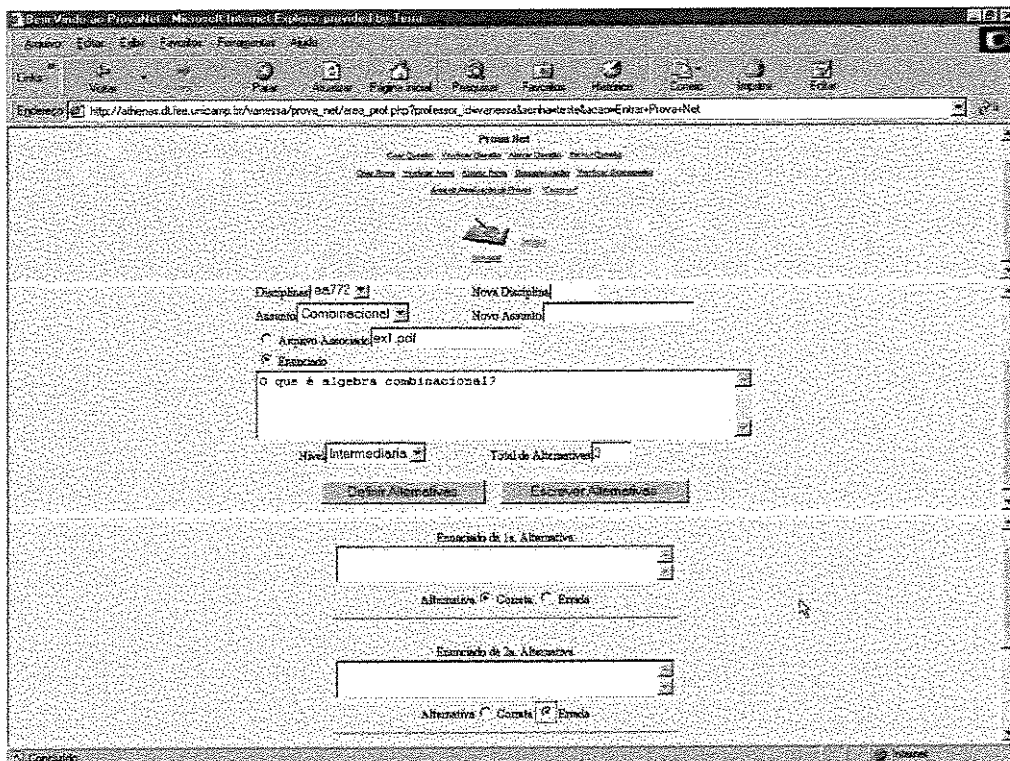


Figura 5.5: Criar questão - enunciado e escrita das alternativas.

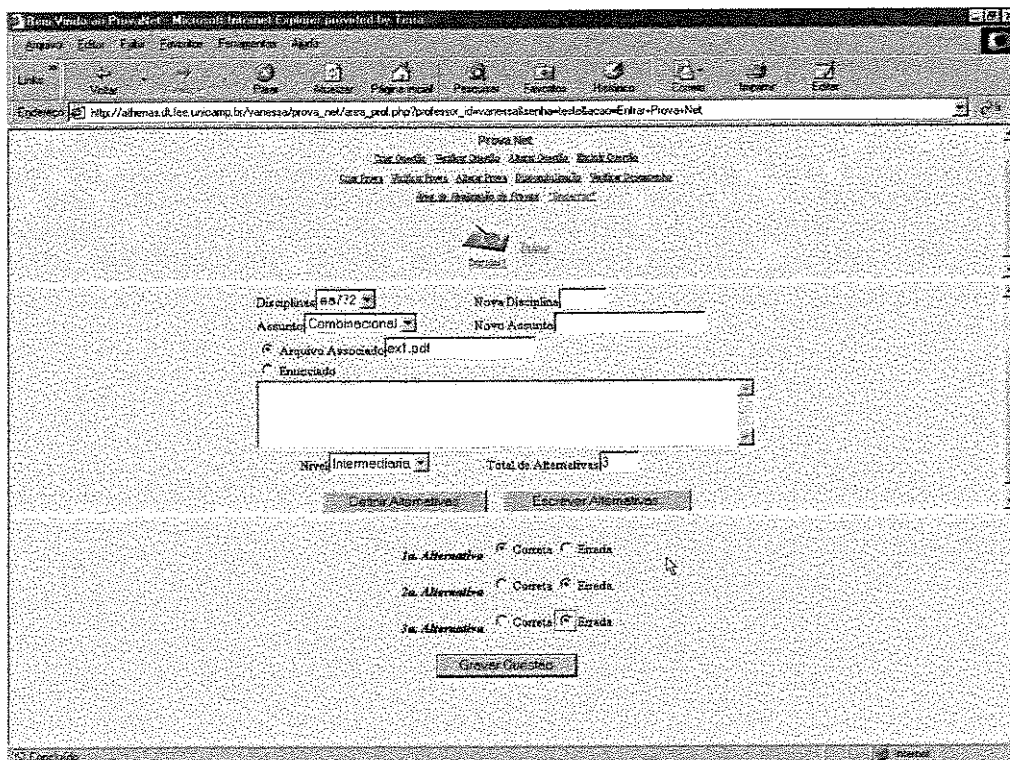


Figura 5.6: Criar questão - enunciado e definição do valor das alternativas.

Para a entrada dos dados do enunciado, é definida a disciplina, através da escolha no menu de seleção de disciplinas cadastradas, ou através da inserção de uma nova.

O assunto também é estabelecido da mesma forma, através da escolha no menu de seleção, que apresenta os assuntos de questões cadastradas, ou através da inserção de um novo.

Campos específicos para entrada de enunciado do tipo ASCII ou através do fornecimento de um arquivo previamente criado são apresentados na tela, assim como campos para definição do nível e total de alternativas da questão.

## B. Verificar Questão

A verificação de uma questão criada é realizada através da apresentação de quatro telas do sistema ProvaNet. Primeiramente o professor define a disciplina e o assunto da questão através de menus de seleção e uma lista com o enunciado das questões que atendem aos critérios estabelecidos de disciplina e assunto escolhidos é exibida.

O professor escolhe uma dentre as apresentadas, e verifica os detalhes da questão, como mostrado na Figura 5.7.

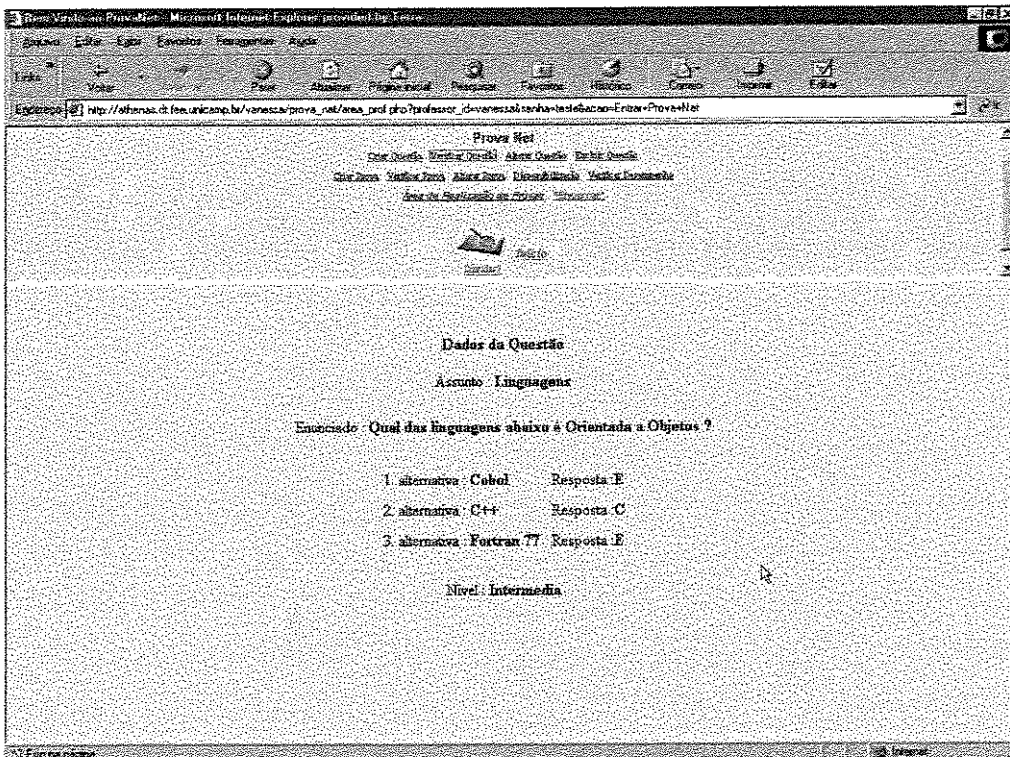


Figura 5.7: Verificar questão.

### C. Alterar Questão

Para que uma questão possa ser alterada, primeiramente o professor deve definir a disciplina e o assunto, fazendo esta escolha nos menus de seleção; uma lista com o enunciado das questões que atendem aos critérios de disciplina e assunto escolhidos é exibida e, em seguida, uma tela contendo todos os dados da questão para alteração é apresentada.

Além do enunciado da questão, de acordo com a Figura 5.8, as alternativas podem ser modificadas, tanto no enunciado como no número de alternativas apresentadas, como mostrado na Figura 5.9.

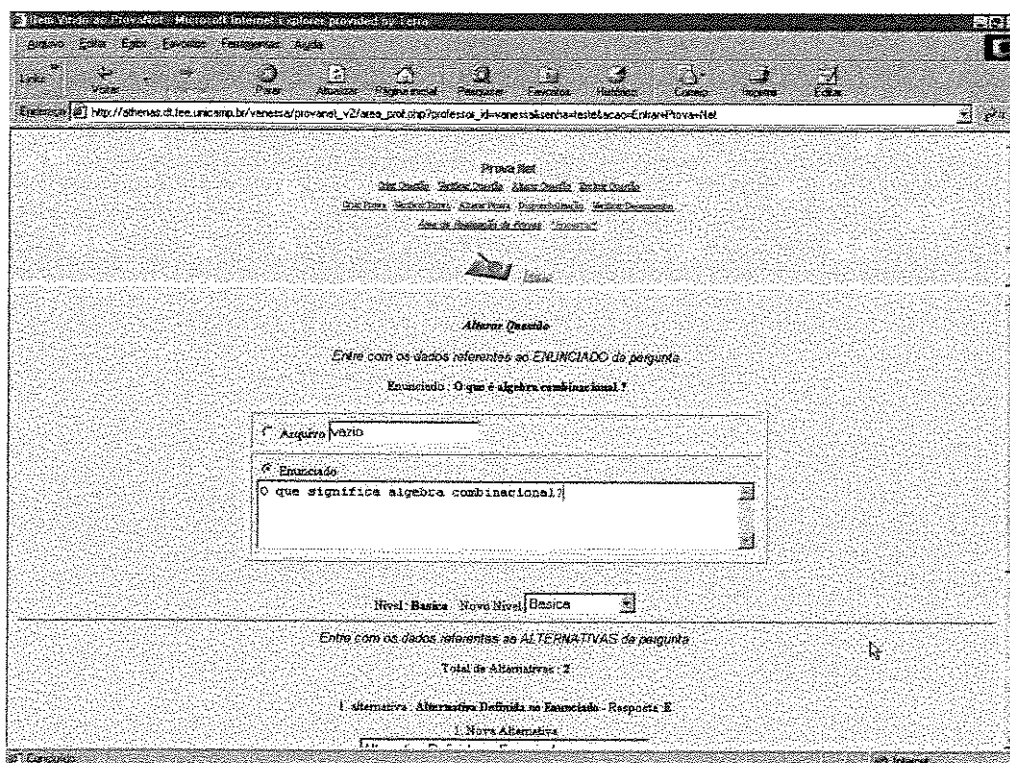


Figura 5.8: Alterar questão - enunciado.

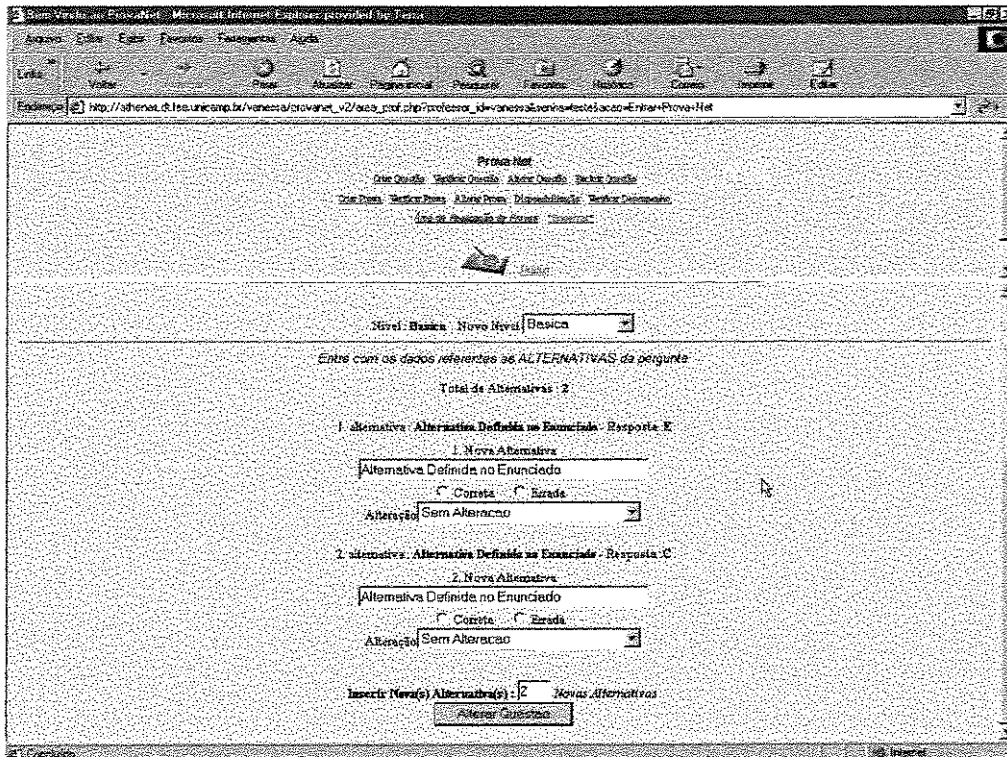


Figura 5.9: Alterar questão - alternativas.

O sistema emite uma mensagem de confirmação, informando que a alteração de uma questão foi bem sucedida.

## D. Excluir Questão

Para que uma questão possa ser excluída, o professor deve definir a disciplina e o assunto, fazendo esta escolha nos menus de seleção; a lista com o enunciado das questões que atendem aos critérios de disciplina e assunto escolhidos é exibida e a questão é selecionada.

Caso a questão a ser excluída esteja inserida em alguma prova, o ProvaNet disponibiliza uma tela de aviso, solicitando a confirmação. Em seguida uma nova tela contendo a confirmação da ação realizada pelo usuário (exclusão ou não da questão) é apresentada, conforme a Figura 5.10.

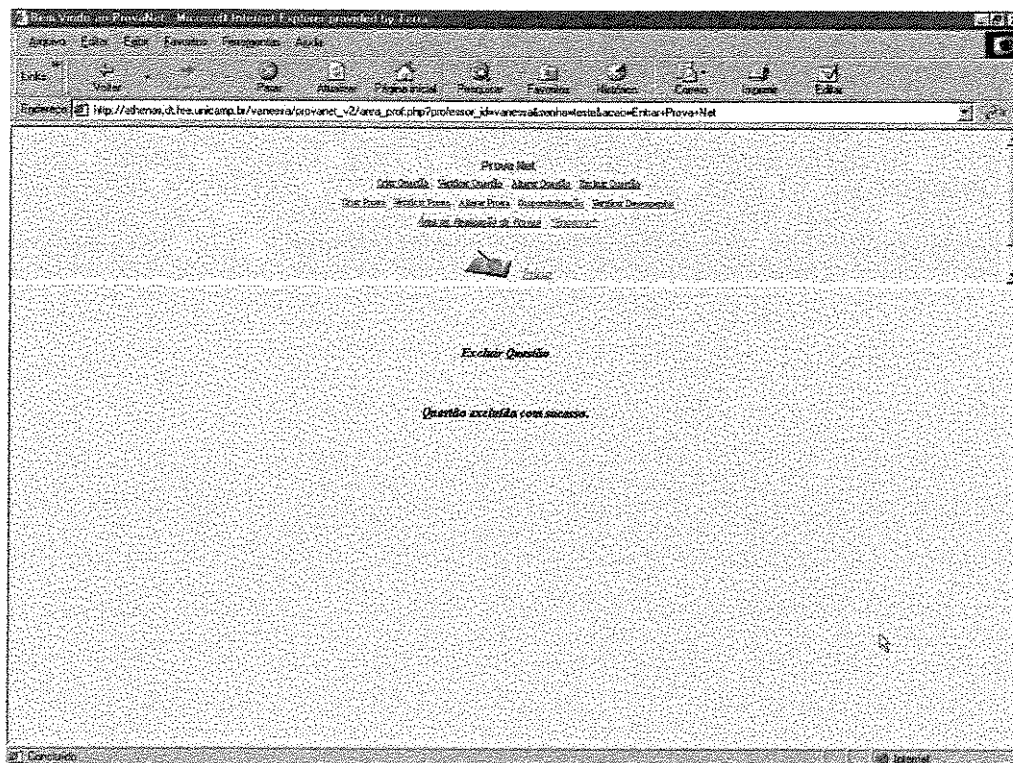


Figura 5.10: Excluir questão - confirmação da exclusão.





A questão a ser inserida é determinada escolhendo-se o assunto da questão desejada (a disciplina foi identificada no momento da criação da prova). Uma lista com o enunciado das questões que atendem aos critérios de disciplina e assunto é exibida, e o professor pode inserir esta questão imediatamente ou solicitar a visualização dos detalhes.

O professor pode inserir a questão na prova clicando no botão "inserir questão" ou finalizar a prova, clicando no botão "finalizar". Uma vez realizada a inserção, o sistema exibe uma tela de confirmação e habilita a entrada de novas questões através do botão "nova questão".

O ProvaNet exibe uma tela de aviso toda vez que se tentar incluir uma questão que já foi inserida na prova.

Quando o professor finaliza a inserção de questões, o ProvaNet exibe a tela de definição da ordem de disponibilização da prova, conforme a Figura 5.12.

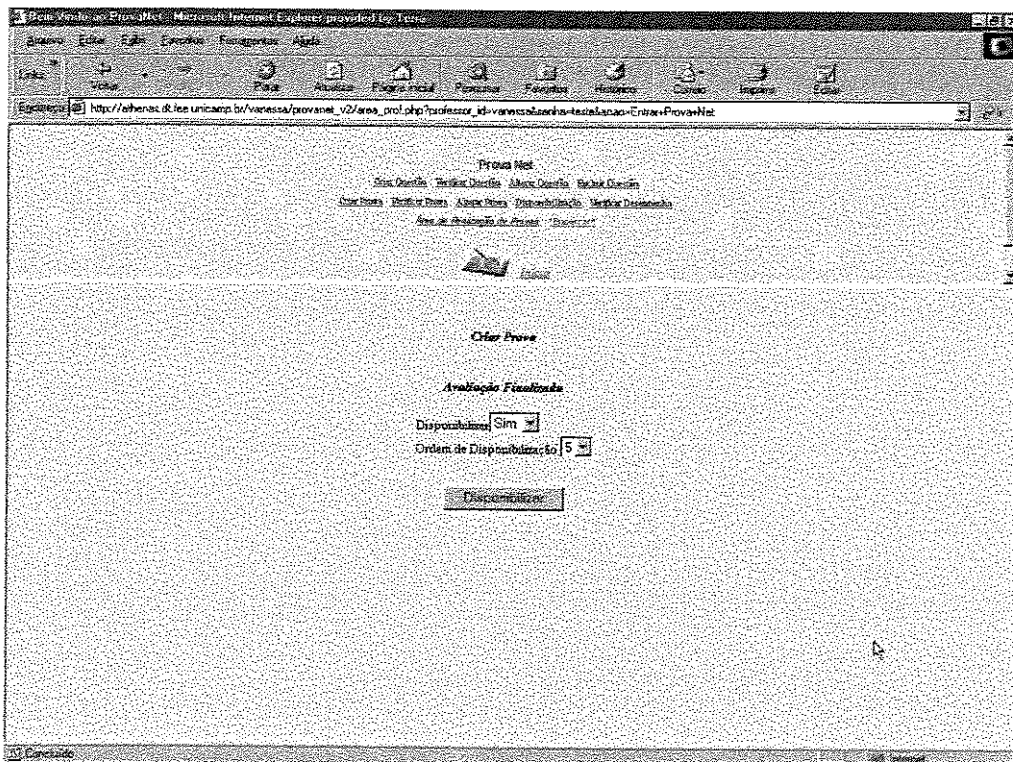


Figura 5.12: Criar prova - definição da ordem de disponibilização.

Finalizada a definição da disponibilização, o sistema exibe a tela de confirmação da ordem selecionada.

#### **F. Verificar Prova**

A verificação de uma prova criada é realizada através da exibição de três telas do sistema ProvaNet. Primeiramente o professor define a disciplina da prova e em seguida, as provas cadastradas (avaliações e simulados) são apresentadas. O professor escolhe uma dentre as apresentadas, e verifica os detalhes da prova (detalhes das questões inseridas).

#### **G. Alterar Prova**

Para que uma prova possa ser alterada, primeiramente o professor deve definir a disciplina, fazendo esta escolha no menu de seleção.

Uma lista contendo todas as provas criadas (em função da data de criação e do tipo) e as funções de alteração disponíveis são exibidas conforme a Figura 5.13.

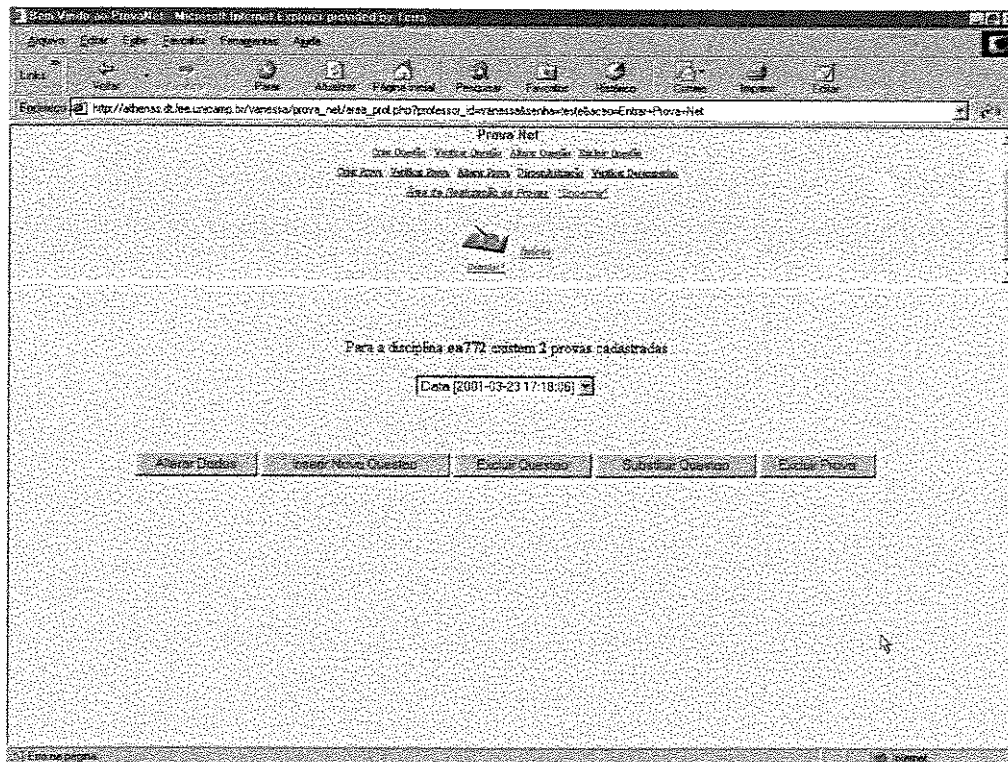


Figura 5.13: Alterar prova - funcionalidades.

Em função da alteração que deseja realizar, o professor clica em um dos cinco botões que fazem referência às funcionalidades de alteração habilitadas pelo sistema: "alterar dados", "inserir nova questão", "excluir questão", "substituir questão" e "excluir prova". Em seguida, o professor é remetido para a tela de manipulação da funcionalidade escolhida.

## H. Alterar Dados da Prova

Uma tela contendo todos os dados da prova e os campos de entrada para alteração é apresentada, como mostrado na Figura 5.14. O professor altera os dados inserindo novos valores e confirma a alteração.

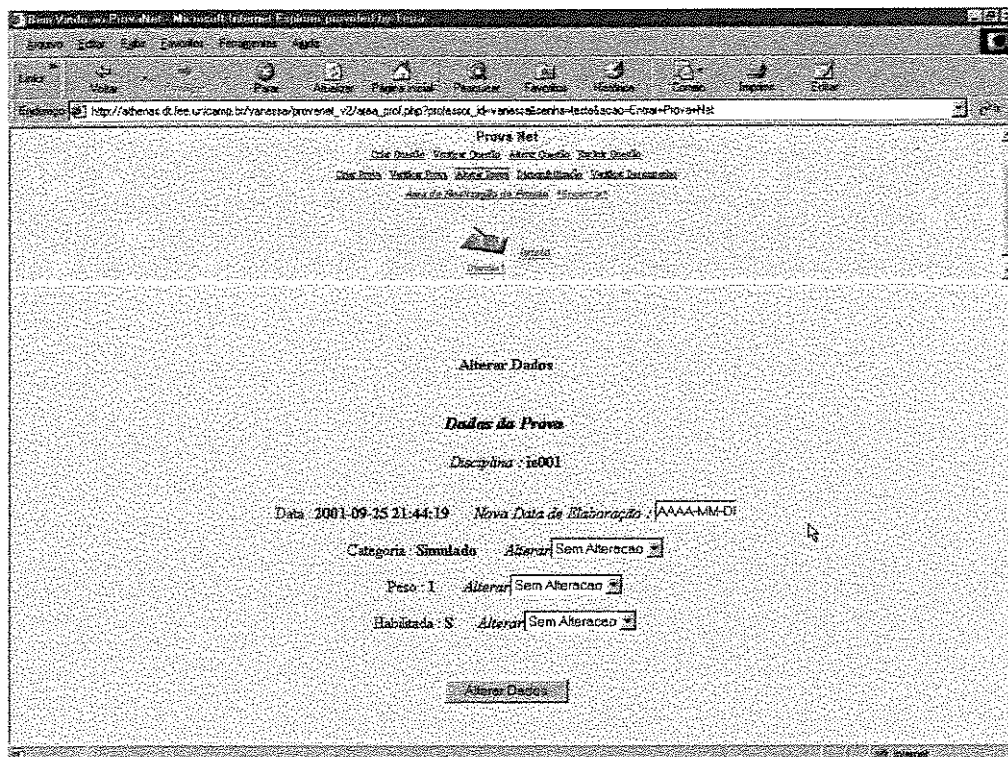


Figura 5.14: Alterar prova - alterar dados.

## I. Inserir Questão

Para que uma nova questão seja inserida em uma prova criada, o professor define o assunto (a disciplina foi inserida na tela inicial da funcionalidade alteração). Uma lista com o enunciado das questões que atendem aos critérios de disciplina e assunto é exibida, e o professor pode inserir esta questão imediatamente clicando no botão "inserir questão" ou solicitar a visualização dos detalhes, clicando no botão "detalhes". Uma vez realizada a inserção, o sistema exibe uma tela de confirmação.

Caso o professor deseje inserir outras questões na prova, deve clicar novamente no *link* "alterar prova" exibido no menu de funções gerais do ambiente de criação de provas.

#### **J. Excluir Questão da Prova**

Uma tela contendo todos os detalhes da prova é exibida pelo sistema e o professor escolhe, através de um menu de seleção contendo o número das questões daquela prova, uma dentre as apresentadas para que seja excluída.

O sistema emite uma mensagem de confirmação, informando que a exclusão da questão foi bem sucedida.

#### **K. Substituir Questão**

Uma tela contendo todos os detalhes da prova é exibida pelo sistema e o professor escolhe, através de um menu de seleção contendo o número das questões daquela prova, uma dentre as apresentadas para que seja substituída.

Uma vez estabelecida a questão a ser substituída, o professor define o assunto da nova questão (a disciplina foi inserida na tela inicial da funcionalidade alteração) através da escolha em um menu de seleção e uma lista com o enunciado das questões que atendem aos critérios de disciplina e assunto é exibida. O professor pode substituir esta questão imediatamente ou solicitar a visualização dos detalhes. Uma vez realizada a substituição, o sistema exibe uma tela de confirmação.

Caso o professor deseje substituir outras questões da prova, deve clicar novamente no *link* "alterar prova" exibido no menu de funções gerais do ambiente de criação de provas, para em seguida solicitar a funcionalidade "substituir questão".

## **L. Excluir Prova**

Para que uma prova possa ser excluída, primeiramente o professor deve clicar na função "excluir prova". O sistema disponibiliza uma nova tela contendo a lista de todas as provas criadas (em função da data de criação e do tipo) e o professor escolhe aquela que deverá ser excluída.

Em seguida, basta que seja clicado o botão "excluir prova" e o sistema exibe uma tela de confirmação da exclusão.

## **M. Disponibilizar Prova**

Para ter acesso aos dados de disponibilização das provas cadastradas no sistema, o professor deve definir inicialmente qual a disciplina das provas que deseja visualizar.

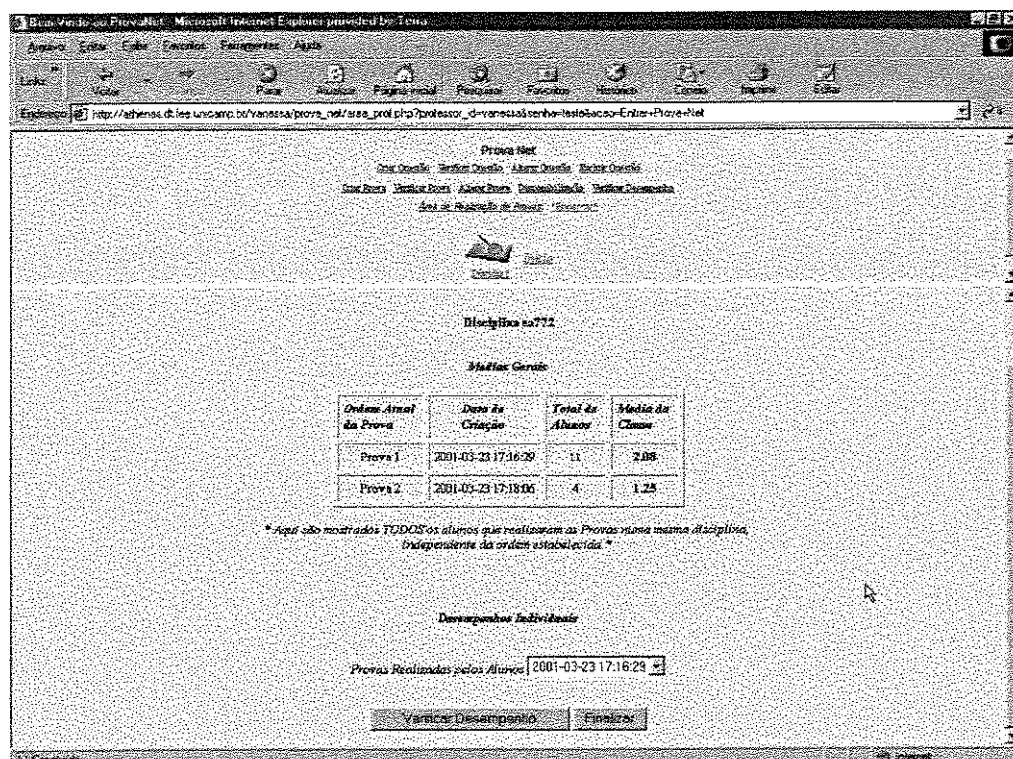
Todas as provas (avaliações ou simulados) que foram criadas na disciplina definida são listadas e o professor pode verificar ou alterar a ordem de disponibilização das provas. Ao alterar a ordem estabelecida, o professor pode habilitar ou desabilitar a disponibilização cadastrada manipulando os campos de entrada de cada prova apresentada, conforme mostrado pela Figura 5.15.





## N. Verificar Desempenho

Para ter acesso aos dados de desempenho dos alunos, o professor deve definir inicialmente a disciplina das provas realizadas e uma tela contendo a lista de todas as provas selecionadas é apresentada, juntamente com a média geral dos alunos, conforme mostrado na Figura 5.16.



The screenshot shows a web browser window with the URL `http://afenas.d.ies.uacamp.br/vanessa/prova_net/area_prof.php?professor_id=vanessa@cerbo-testelocao-Enter-Prova-Net`. The page displays a menu at the top with options like 'Prova Net', 'Data Questão', 'Respostas Questão', 'Aluno Questão', 'Respostas Questão', 'Data Prova', 'Verificar Prova', 'Aluno Prova', 'Disponibilidade', 'Verificar Desempenho', and 'Data de Realização de Prova'. Below the menu, there is a 'Disciplina' field with the value '772' and a 'Médias Gerais' table. The table has four columns: 'Ordem Anual da Prova', 'Data de Criação', 'Total de Alunos', and 'Média da Classe'. The data rows are: 'Prova 1' with date '2001-03-23 17:16:29', total of 11 students, and a class average of 2.08; and 'Prova 2' with date '2001-03-23 15:18:06', total of 4 students, and a class average of 1.25. Below the table, there is a note: '\* Aqui são mostrados TODOS os alunos que realizaram as Provas numa mesma disciplina, independente da ordem estabelecida.' At the bottom, there is a 'Desempenho Individual' section with a dropdown menu for 'Provas Realizadas pelos Alunos' set to '2001-03-23 17:16:29' and two buttons: 'Verificar Desempenho' and 'Finalizar'.

Ordem Anual da Prova	Data de Criação	Total de Alunos	Média da Classe
Prova 1	2001-03-23 17:16:29	11	2,08
Prova 2	2001-03-23 15:18:06	4	1,25

Figura 5.16: Verificar desempenho - desempenho geral.

O professor também pode visualizar o desempenho individual dos alunos, escolhendo uma prova dentre as apresentadas no menu de seleção e clicando no botão "desempenho individual". Desta forma, o ProvaNet disponibiliza uma tela contendo o nome de todos os alunos que realizaram aquela prova específica, juntamente com as suas notas individuais.

## O. Simulação de Provas

A partir do menu de funcionalidades gerais do ambiente de criação de provas, o professor tem acesso à área de simulação. Para isso, basta clicar no link "área de realização de provas" e o sistema disponibiliza uma tela de entrada na área de simulação, conforme mostrado na figura 5.17.

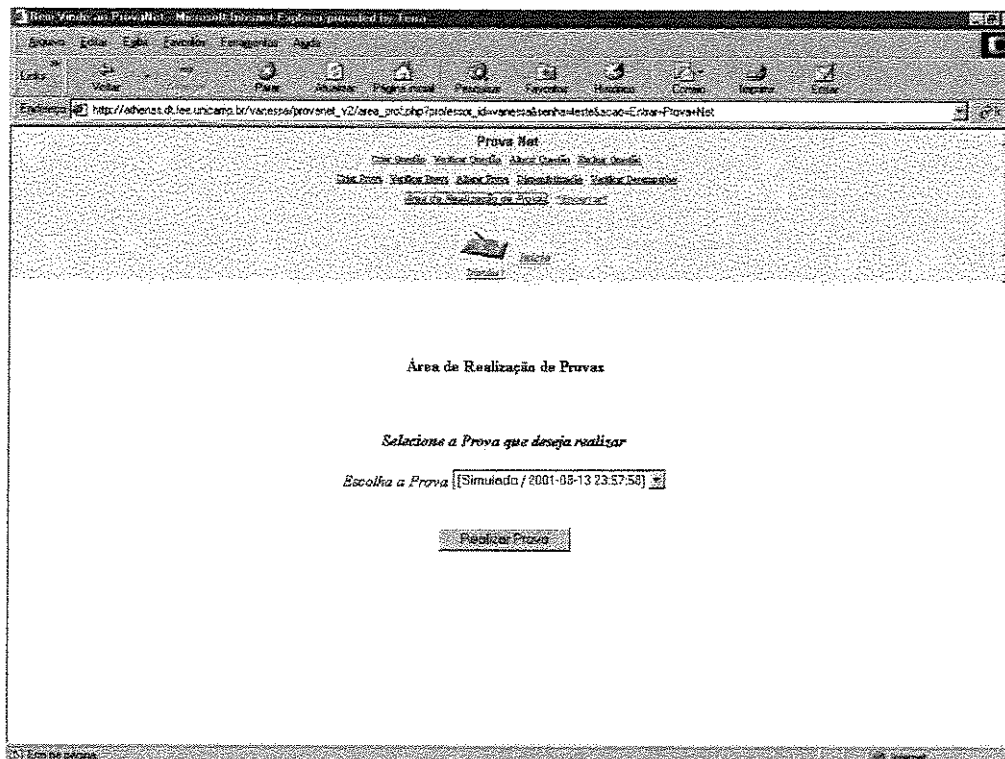


Figura 5.17: Simulação de provas.

Para que o professor possa simular a realização das provas criadas no sistema, inicialmente deve definir a disciplina da prova que deseja resolver, escolhendo uma dentre as apresentadas no menu de seleção. Uma vez definida a disciplina, o ProvaNet apresenta para a escolha todas as provas criadas que obedecem este critério. O professor define a prova que deseja realizar (avaliação ou simulado), clica no botão "realizar prova" e a primeira questão inserida é exibida pelo sistema.

A realização de uma prova obedece aos critérios estabelecidos para avaliações e simulados. Desta forma, caso o professor esteja resolvendo um simulado, após resolver uma questão, o ProvaNet apresenta uma tela com a análise da resposta escolhida. Se a resposta estiver errada, o sistema exibe uma tela intermediária que oferece a opção de retornar à questão anterior para uma nova tentativa de resolução. A Figura 5.18 apresenta esta tela.

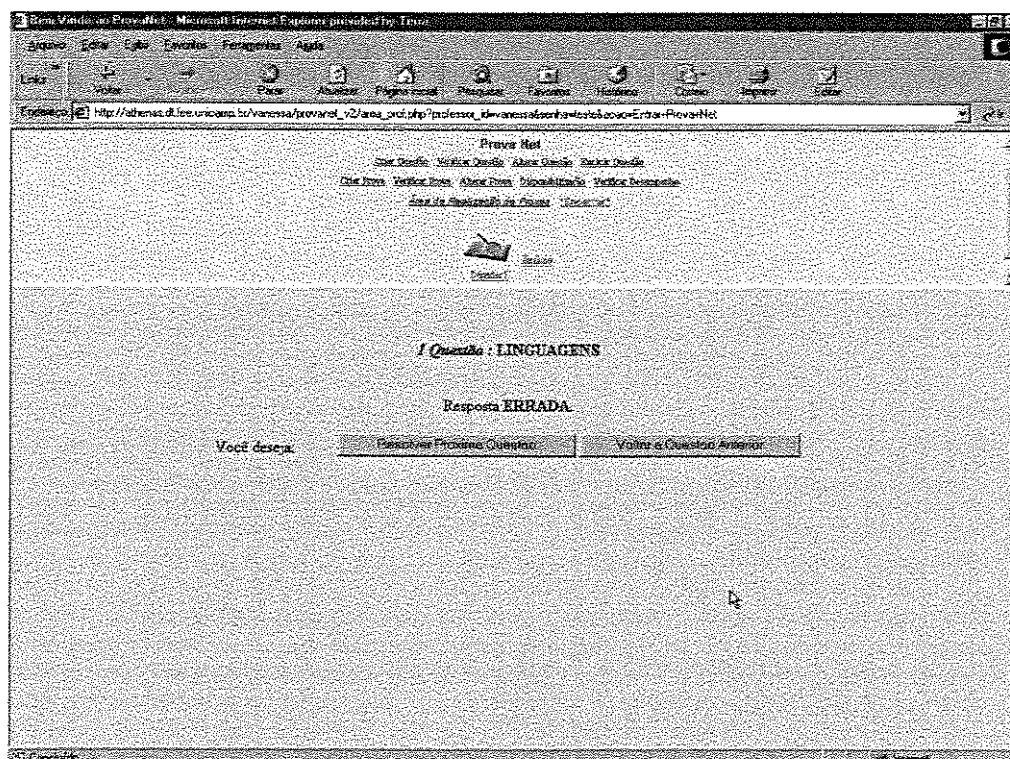


Figura 5.18: Realização de provas - análise da resposta (simulado).

Nas avaliações e simulados, as questões são apresentadas na ordem em que foram cadastradas e, ao se esgotarem, uma tela de finalização é exibida.

Se a prova que estiver sendo simulada for do tipo avaliação, a tela de finalização apresenta o desempenho através da nota final computada.

### 5.2.2. Ambiente de Realização de Provas

Ao entrar na área de realização de provas, uma página indicando o acesso é apresentada ao aluno, como ilustrado na Figura 5.19.

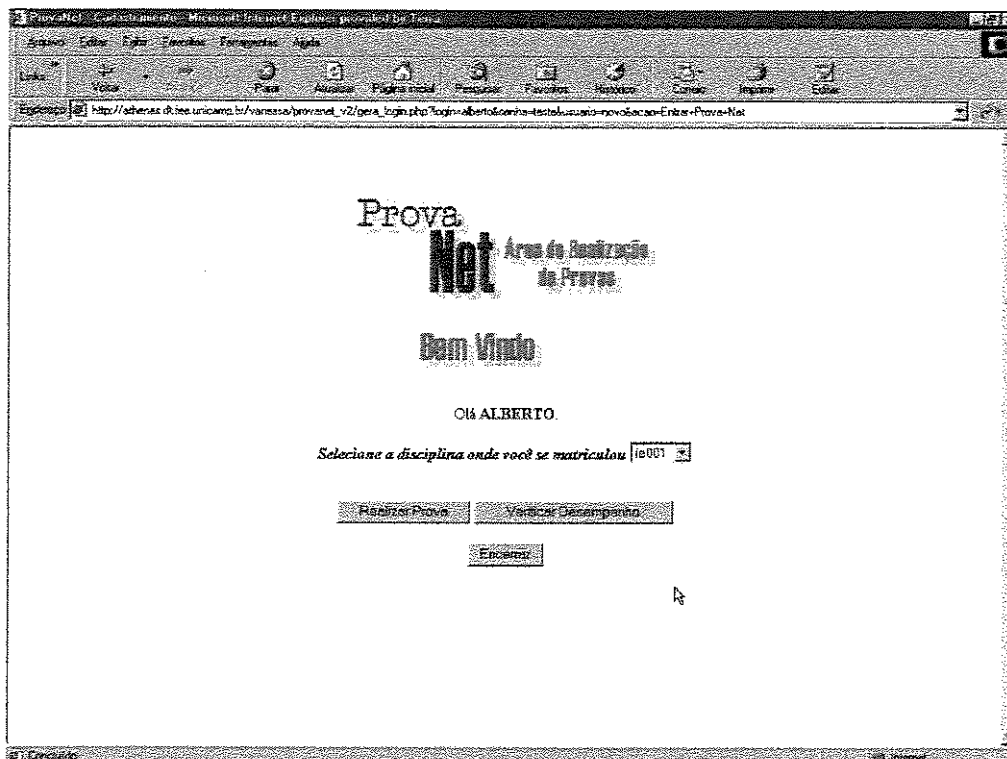


Figura 5.19: Entrada do ambiente de realização.

Em seguida, uma tela para definição da disciplina e do tipo de prova que se deseja realizar é exibida pelo sistema. O aluno pode realizar uma prova, clicando no botão "realizar prova" ou apenas visualizar seu desempenho, clicando no botão "verificar desempenho".

Caso o aluno deseje realizar uma prova, o ProvaNet recupera a prova que será apresentada, através de uma busca pelo do tipo de prova escolhido que ainda não foi resolvida pelo aluno naquela disciplina e que está habilitada pelo professor.

## A. Resolução de Avaliações

O ProvaNet apresenta para resolução do aluno as questões inseridas na avaliação, como mostrado na Figura 5.20. Uma vez respondida a questão, o aluno clica no botão "responder questão". A resposta da questão é analisada, o desempenho computado e a próxima questão inserida na avaliação é exibida para resolução.

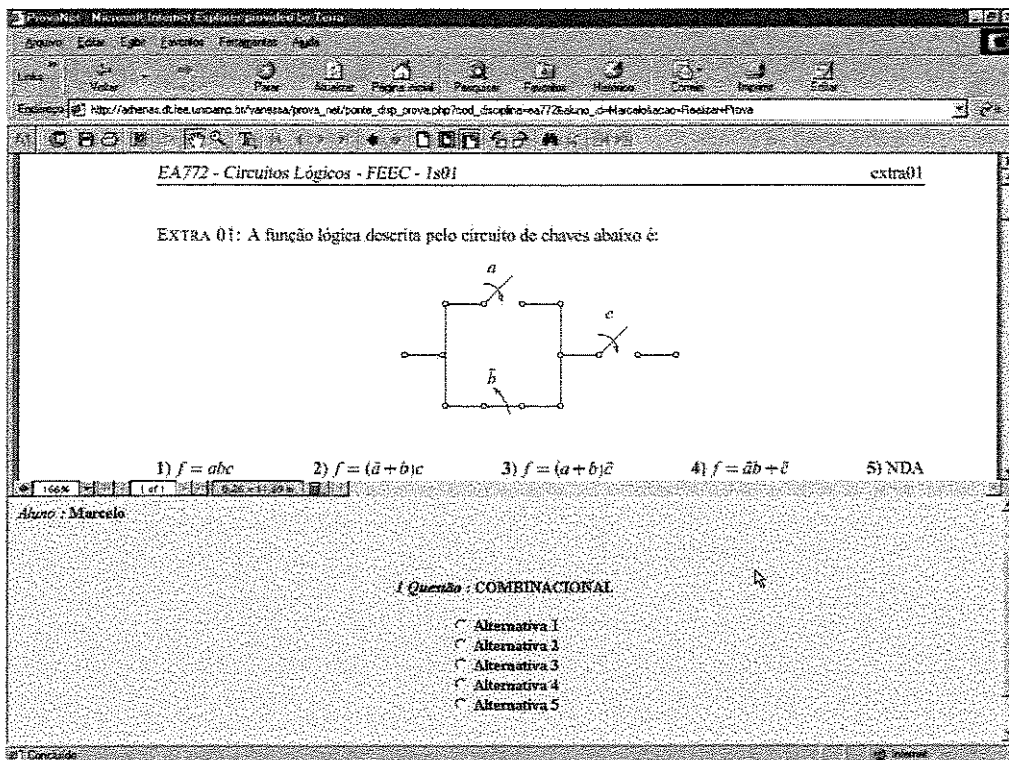


Figura 5.20: Resolução de avaliações - questão.

As questões são apresentadas na ordem em que foram cadastradas e, ao se esgotarem, uma tela de finalização é exibida, apresentando o desempenho do aluno através da nota final computada.

## B. Resolução de Simulados

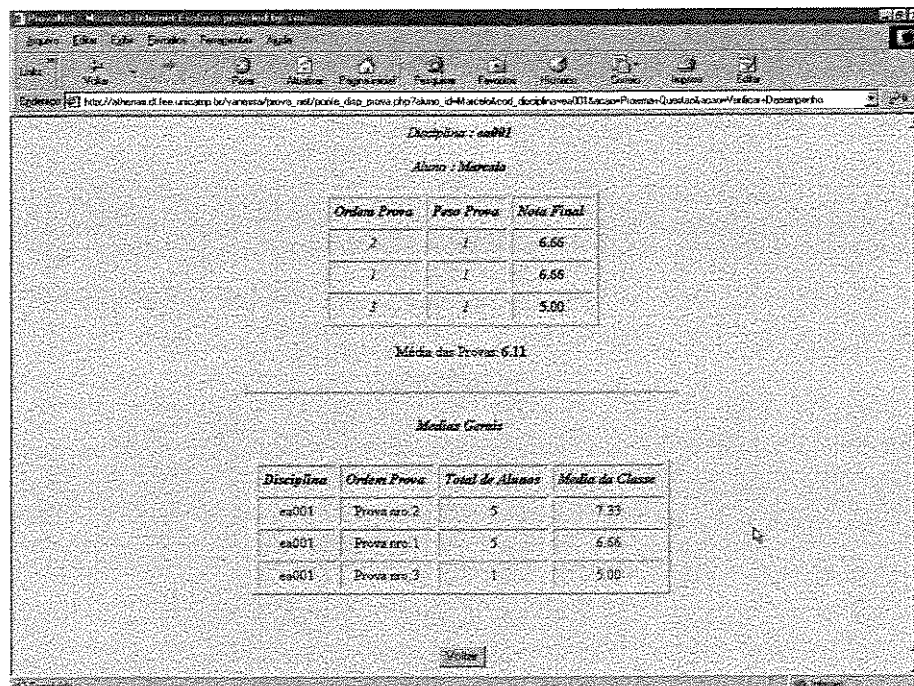
O ProvaNet apresenta para resolução do aluno as questões inseridas em um simulado. Uma vez respondida a questão, o aluno clica no botão "responder questão" e o sistema analisa a resposta escolhida. Se a resposta estiver errada, o ProvaNet apresenta uma tela intermediária que oferece a opção de retornar à questão anterior para uma nova tentativa de resolução.

As questões são apresentadas na ordem que em foram cadastradas e, ao se esgotarem, uma tela de finalização é exibida.

## C. Verificação de Desempenho

Na verificação do desempenho, o sistema apresenta uma tela contendo as notas do aluno em todas as avaliações e simulados realizadas por ele na disciplina escolhida (a disciplina foi inserida na tela anterior), juntamente com a média geral dos alunos que realizaram as mesmas provas.

A tela de desempenho disponibilizada a aluno é mostrada na Figura 5.21.



Disciplina : ea001

Aluno : Mareia

Ordem Prova	Fase Prova	Nota Final
2	1	6,66
1	1	6,66
3	2	5,00

Média das Provas: 6,11

Médias Gerais

Disciplina	Ordem Prova	Total de Alunos	Média da Classe
ea001	Prova nro. 2	5	7,33
ea001	Prova nro. 1	5	6,66
ea001	Prova nro. 3	1	5,00

Figura 5.21: Verificação de desempenho.

---

## Conclusões e Projetos Futuros

---

A educação avança cada vez mais na direção do aprendizado à distância, e a quantidade e variedade de aplicativos gerados nos últimos anos comprovam a importância desta metodologia de ensino.

Este trabalho apresentou um aplicativo que tem como principal objetivo contribuir para o desenvolvimento do aprendizado enfocando o aspecto da avaliação (tanto para quem cria, como para quem realiza), visando uma maior interação entre as partes através da disponibilização de material para estudo.

O professor pode avaliar o impacto do material disponibilizado através da análise da resolução das questões colocadas no ambiente, e reformular o conteúdo se assim achar conveniente. O aluno pode avaliar seu próprio aprendizado e, ao se dedicar às tarefas de resolução de problemas, transforma esta ação em conhecimento.

Em salas de aula presenciais, o ProvaNet pode ser empregado como um procedimento eficaz de avaliação individual automática através da Internet, cabendo ao professor o controle e a identificação dos usuários.

Como forma de dar continuidade ao trabalho de desenvolvimento do ProvaNet, outros aspectos podem ser abordados:

O controle de segurança do aplicativo pode ser desenvolvido, incorporando-se um módulo de proteção que impede o acesso e manipulação (intencional ou não) de informações contidas no sistema. Esta proteção pode ser desenvolvida em função das possíveis ameaças e riscos, formalizadas em termos do desenvolvimento de uma política completa de segurança.

Um módulo de controle de realização de provas pode ser desenvolvido, considerando que um aluno gostaria de voltar exatamente no mesmo ponto em que estava quando ocorreu uma queda do sistema, por exemplo.

O suporte a novas tecnologias para a construção de questões pode ser abordado, e também novos formatos de questões podem ser contemplados pelo sistema, como por exemplo questões dissertativas; novos tipo de mídia

para construção de questões, como áudio e vídeo; novas formas de avaliar as respostas fornecidas pelos alunos, por exemplo contabilizando de alguma maneira respostas certas, quase certas, erradas e muito erradas.

Um módulo para o gerenciamento do tempo de consulta às questões pode ser desenvolvido, a partir do qual o professor poderia realizar um análise da dificuldade das questões disponibilizadas, trazendo novas informações e contribuições ao sistema.

Uma nova versão do sistema poderá incluir a criação de testes adaptativos, nos quais a próxima questão a ser disponibilizada ao aluno depende da resposta dada na questão anterior. Maiores detalhes sobre testes adaptativos podem ser encontrados em [IMSProject, 1999].

E finalmente, a integração com outros sistemas pode ser contemplada, definindo o ProvaNet como um módulo de avaliação que complementa outros aplicativos voltados para o ensino à distância.



## Glossário e Lista de Acrônimos

---

### A

**ASCII** · American Standard Code for Information Interchange.

---

### B

**Booch** · Método criado por Grady Booch que consiste em técnicas de desenho orientado a objetos, utilizando-o como base para a criação de módulos de um sistema.

---

### C

**CASE** · Computer Aided System Engineering. Sistemas de engenharia assistida por computador.

**CGI** · Common Gateway Interface. Aplicação servidora utilizada geralmente para processar solicitações do navegador através de formulários HTML.

**Classes** · É uma abstração de um conjunto de objetos que possuem os mesmos tipos de características e comportamentos.

**Cliente-Servidor** · Arquitetura de comunicação utilizada pela Internet. Descreve a relação entre dois programas dos quais um, o cliente, faz uma solicitação de serviço para outro programa, o servidor, que cumpre o pedido.

**Client-Side** · Estação de trabalho que proporciona serviços de apresentação e computação apropriados, conectividade e interfaces relevantes para as necessidades do sistema.

---

### D

**Domínio** · Nome individual na Internet, que designa um *site* ou um conjunto de *sites* na rede. Domínios são hierarquizados, podendo conter subdomínios sucessivos.

---

**E**

**Email** · Termo que designa mensagem enviada por via eletrônica.

---

**F**

**Frame** · Recurso que permite a uma página Web possuir várias janelas controladas separadamente.

---

**H**

**HTML** · HyperText Markup Language. Linguagem de marcação utilizada para construção de páginas na Internet.

---

**I**

**Internet** · Rede mundial de computadores, também referida como Web ou *World Wide Web* (WWW).

---

**J**

**JavaScript** · Linguagem de programação que permite uma maior interatividade do usuário através da criação de páginas HTML dinâmicas.

---

**L**

**Link** · Apontador para outra fonte de informação.

**Login** · Termo que faz referência ao nome utilizado pelo usuário para sua identificação.

---

**M**

**MySQL** · Structured Query Language. Banco de dados relacional de domínio público, amplamente utilizado em aplicativos na Internet.

---

## O

**OMT** · Método criado por James Rumbaugh que consiste na modelagem semântica de dados.

**OOSE** · Método criado por Ivar Jacobson que consiste na análise dos requerimentos de um sistema utilizando-se casos de uso.

---

## P

**PDF** · Portable Document Format. Formato de arquivo texto bastante utilizado pela comunidade acadêmica.

**Perl** · Practical Extraction and Report Language. Linguagem de programação que permite a criação de páginas Web dinâmicas.

**PHP** · Hypertext Preprocessor. Linguagem de programação que proporciona maior dinamismo às páginas Web, e permite a integração do aplicativo com o bancos de dados.

**Plug-in** · Programa que é acoplado ao navegador, objetivando aumentar a sua funcionalidade.

---

## R

**Rational Rose** · Ferramenta da Rational Software que permite a criação de diagramas baseados na metodologia UML.

---

## S

**Script** · Forma de programação que permite a criação de conteúdos mais ricos em aplicativos Internet.

**Server-Side** · Referência ao lado Servidor da arquitetura cliente-servidor.

**Servidor** · No modelo de programação cliente-servidor, um servidor é um programa que espera e executa serviços requisitados pelo programa cliente.

**Site** · Conjunto de páginas no ambiente Web que é ocupado com informações como texto, fotos, animações gráficas, sons ou vídeos.

**Software** · Programas, dados e rotinas desenvolvidos para computadores.

**SGML** · Standard Generalized Markup Language. Linguagem de marcação de documentos.

---

**T**

**Three-Tiered Architecture** · Arquitetura em três camadas.

---

**U**

**UML** · Unified Modeling Language. Linguagem padronizada para modelagem de sistemas orientados a objetos.

**URL** · User Register Location. Endereço HTTP que identifica individualmente cada arquivo na rede Web.

---

**X**

**XML** · Extensible Markup Language. Linguagem utilizada no desenvolvimento de aplicativos Web. Atualmente bastante utilizada principalmente por estar se tornando um padrão para a conservação de dados.

---

**W**

**Web** · Referência a *World Wide Web* (WWW).

## Bibliografia

[Ambler, 1995] Ambler, S. W.; The Object Primer. The Application Developer's Guide to Object Orientation. Ed. SIGS Books (1995).

[Ambler, 1998] Ambler, S. W.; Análise e Projeto Orientados a Objeto, volume II: Seu Guia para Desenvolver Sistemas Robustos com Tecnologia de Objetos. Ed. Infobook (1998).

[Avalweb, 2001] Sistema Interativo de Gerência de Questões para Avaliação via Web (2001).  
<http://cemt.inf.ufrgs.br/avalweb/>

[Bernstein, 1999] Bernstein, T.; Segurança na Internet. Ed. Campus (1999).

[Boggs e Boggs, 1999] Boggs, W.; Boggs M.; Mastering UML with Rational Rose. Ed. SYBEX (1999).

[Booch et al., 1997] Booch, G.; Jacobson, I.; Rumbaugh, J.; The Unified Modeling Language User Guide. Ed. Addison-Wesley Object Technology Series (1997).

[CAA Centre, 2001] Computer Assisted Assessment (2001).  
<http://www.caacentre.ac.uk/index.html>

[Capa, 2001] Computer Assisted Personalized Approach (2001).  
<http://capa.mse.edu/>

[Cruz et al., 2001] Cruz, C. C.; Romano, M. I. R.; Vargas, P. A.; Guidetti, S. A.; Aprendizagem Individualizada. Relatório final da disciplina IA368F - Unicamp (2001).  
[http://www.decom.fee.unicamp.br/~cristia/ia368/Relatorio/relatorio\\_final.html](http://www.decom.fee.unicamp.br/~cristia/ia368/Relatorio/relatorio_final.html)

[Cunningham, 1995] Cunningham, B.; The QForm Package (1995).  
<http://www.satlab.hawaii.edu/space/hawaii/qform.html>

[Fróes, 1998] Fróes, J.; Educação e Tecnologia: O Desafio do Nosso Tempo. Artigos Selecionados de Informática e Educação (1998).  
<http://www.divertire.com.br>

[IMSProject, 1999] IMS Question & Interoperability Specification (1999).  
<http://www.imsproject.org/question/index.html>

[Jacobson et al., 1995] Jacobson, I.; Christerson, M.; Jonsson, P.; Overgaard, G.; Object Oriented Software Engineering: A Use Case Driven Approach. Ed. Addison-Weasley (1995).

[Laquey e Ryer, 1994] Laquey, T.; Ryer, J. C.; O Manual da Internet: Um Guia Introdotório para Acesso às Redes Globais. Ed. Campus (1994).

[Leiva, 1999] Leiva, W. D.; Ferramenta para Autoria de Questionários. Relatório interno da disciplina IA368G - Unicamp (1999).  
<http://www.icmc.sc.usp.br/~wdl/ia368g/RelatWillie.htm>

[Martin e Stewart, 1998] Martin, H.; Stewart, A.; Hot Potatoes: Free Tools for Creating Interactive Language Exercises for the World Wide Web. Eurocall "From Classroom Teaching to World-Wide Learning" (1998).  
<http://www.arts.kuleuven.ac.be/eurocall98/abstracts/holmesmartin.htm>

[McKnight et al., 1996] McKnight, C.; Dillon, A.; Richardson, J.; User-Centered Design of Hypertext/Hypermedia for Education. Handbook of Research for Educational Communications and Technology. David H. Jonassen (editor), 21:622-633, Macmillan, New York (1996).

[Moran, 1997] Moran, J. M.; Como Utilizar a Internet na Educação. Artigo publicado na Revista Ciência da Informação, 26(2):146-153 (1997).

[Moran, 1998] Moran, J. M.; Mudar a Forma de Ensinar e de Aprender com Tecnologias. Artigos Selecionados de Informática e Educação (1998).  
<http://www.divertire.com.br>

[MySQL, 2001] MySQL Reference Manual (2001).  
<http://www.mysql.com/documentation/mysql/bychapter>

[Pfaff-Harris, 1999] Pfaff-Harris, K. L.; QuizTest, Scripts for Educators: QuizTest (1999).  
<http://www.linguistic-funland.com/scripts>

[PHP, 2001] PHP Manual (2001).  
<http://www.php.net/manual/en>

[Pressman, 1995] Pressman, R.; Engenharia de software. Ed. Makron Books (1995).

[Question, 1997] Question Mark. QM Web - A Testing and Assessment Program for the World Wide Web (1997).  
<http://www.qmark.com/products/qmweb.html>

[Righetto et al., 2001] Righetto, V. L.; Peres, P. L. D.; Bonatti, I. S.; ProvaNet: Um Sistema de Avaliação de Aprendizado na Internet. Artigo submetido à Revista Brasileira de Informática na Educação (2001).

[Romiszowski e Manson, 1996] Romiszowski, A. J.; Manson, R.; Computer-Mediated Communication. Handbook of Research for Educational Communications and Technology. David H. Jonassen (editor), 14:438-456, Macmillan, New York (1996).

[Rumbaugh et al., 1994] Rumbaugh, J.; Blaha, M.; Premerlani, W.; Frederick, E.; Lorensen, W.; Modelagem e Projetos Baseados em Objetos. Ed. Campus (1997).

[Santoro et al., 1999] Santoro, F. M.; Borges, M. R. S.; Santos, N.; Um Framework para Estudo de Ambientes de Suporte à Aprendizagem Cooperativa. Revista Brasileira de Informática na Educação, 4:51-58 (1999).

[Santos, 1998] Santos, Jr. J. B.; Documentos Estruturados para o Domínio de Aplicação no Ensino: Modelagem, Autoria e Apresentação na WWW. Instituto de Ciências Matemáticas e de Computação de São Carlos/USP São Carlos, Dissertação de Mestrado (1998).

[Scapin, 1997] Scapin, R. H.; Desenvolvimento de uma Ferramenta para Criação e Correção Automáticas de Provas na World Wide Web. Instituto de Física de São Carlos/USP São Carlos, Dissertação de Mestrado (1997).

[Shute e Psootka, 1996] Shute, V. J.; Psootka, J.; Intelligent Tutoring Systems: Past, Present, and Future. Handbook of Research for Educational Communications and Technology. David H. Jonassen (editor), 19:570-600, Macmillan, New York (1996).

[Soares et al., 1995] Soares, L. F. G.; Lemos, G.; Colcher, S.; Redes de Computadores: das LANs, MANs e WANs às Redes ATM. Ed. Campus (1997).

[Soares, 2000] Soares, W.; Programando em PHP: Conceitos e Aplicações. Ed. Érica (2000).

[Tanenbaum, 1997] Tanenbaum, A. S.; Redes de Computadores. Ed. Campus (1997).

[XML, 2001] XML from the inside out (2001).  
<http://www.xml.com/pub/a/98/10/guide0.html>

[WebCT, 2001] WebCT Homepage (2001).  
<http://www.webct.com/>

[Wolz et al., 1998] Wolz, U.; Palm, J.; Anderson, P.; Dunne, J.; Karlsson, G.; Laribi, A.; Mannikko, S.; Spielvogel, R.; Walker, H.; Computer-Mediated Communication in Collaborative Educational Settings, 25(4):51-69 ACM SIGCUE Outlook (1998).