

UNIVERSIDADE FEDERAL DO MARANHÃO
CENTRO DE CIÊNCIAS EXATAS E TECNOLOGIA
DEPARTAMENTO DE ENGENHARIA DE ELETRICIDADE
CURSO DE PÓS-GRADUAÇÃO EM ENGENHARIA DE ELETRICIDADE

JOSÉ FLÁVIO GOMES BARROS

**MODELAGEM E DESENVOLVIMENTO DO AGENTE DICA DO AMBIENTE DE
AVALIAÇÃO INTEGRADO AO SOFTWARE VIRTUAL-TANEB APLICADO À
GESTÃO ESCOLAR DE RESULTADOS EDUCACIONAIS**

São Luís

2010

JOSÉ FLÁVIO GOMES BARROS

**MODELAGEM E DESENVOLVIMENTO DO AGENTE DICA DO AMBIENTE DE
AVALIAÇÃO INTEGRADO AO SOFTWARE VIRTUAL-TANEB APLICADO À
GESTÃO ESCOLAR DE RESULTADOS EDUCACIONAIS**

Dissertação de Mestrado apresentada ao curso de Pós-Graduação em Engenharia de Eletricidade da Universidade Federal do Maranhão, como parte dos requisitos para a obtenção do título de Mestre em Engenharia de Eletricidade, na área de Ciência da Computação.

Orientador: Prof. Dr. Sofiane Labidi

São Luís

2010

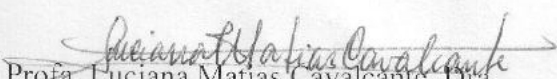
**MODELAGEM E DESENVOLVIMENTO DO AGENTE DICA DO
AMBIENTE DE AVALIAÇÃO INTEGRADO AO SOFTWARE
VIRTUAL-TANEB APLICADO À GESTÃO ESCOLAR DE
RESULTADOS EDUCACIONAIS**

José Flávio Gomes Barros

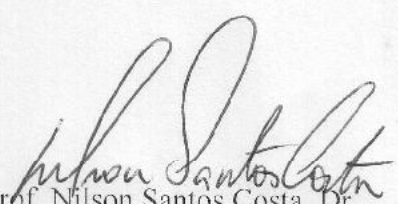
Dissertação aprovada em 10 de dezembro de 2010.



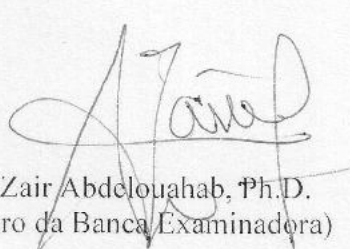
Prof. Sofiane Labidi, Dr.
(Orientador)



Profa. Luciana Matias Cavalcante, Dra.
(Membro da Banca Examinadora)



Prof. Nilson Santos Costa, Dr.
(Membro da Banca Examinadora)



Prof. Zair Abdelouahab, Ph.D.
(Membro da Banca Examinadora)

***Em memória da minha inesquecível
Avó, Terezinha Lopes de Alcântara.***

AGRADECIMENTOS

Em primeiro lugar a Deus, por todas as bênçãos que me tem concedido.

À minha ESPOSA Helanne Cristina e FILHAS Larha Theresa e Layna Talita, pela COMPREENSÃO, AMOR, DEDICAÇÃO demonstradas nos momentos difíceis vividos aqui nesta etapa da VIDA.

Ao Prof. Dr. Sofiane Labidi, MEU ORIENTADOR E AMIGO, por ter me acolhido e confiado no meu trabalho a ser DESENVOLVIDO junto ao LSI.

Ao Prof. Dr. Nilson Santos, MEU AMIGO, por suas sugestões e correções que contribuíram para melhorar a qualidade deste trabalho.

À minha FAMÍLIA, Francisca Antônia, Francisco de Assis, Emanuel de Alcântara, Ithayara Gomes etc, pelo apoio e confiança em minha dedicação.

Aos meus TIOS de São Luís, Júlia e Ivandir Moraes, pela confiança e apoio dado ao chegar a São Luís para uma nova caminhada a ser conquistada.

Aos meus AMIGOS(AS) que aqui fiz, Lucas Drumond, Djeffersson Smith, Geraldo Abrantes, Raimundo Osvaldo, Vladimir Oliveira, Fernando Sérvulo, Ariel Soares, Alcides Neto, Luis Claudio, Lianna Duarte e Eduardo Devidson.

Aos meus AMIGOS do LSI, Carlos Eduardo, Pedriana, Pedro Brandão, Adriano, Jonatas, Fernando Pinheiro e Rafael Cunha.

Aos meus AMIGOS, Cleber Pereira e Raimundo Neto ("O PIQUENO") que me deram FORÇA nos momentos difíceis durante o mestrado.

Ao meu AMIGO Paulo Cardoso, pela contribuição durante a implementação do trabalho.

Ao meu AMIGO/IRMÃO conquistado em São Luís, Christian Diniz, durante minha caminhada para a conquista do título de mestre.

Ao meu AMIGO/IRMÃO DE LONGOS DESAFIOS, Jaclason Machado, pelo APOIO e INCENTIVO no ingresso e durante o Mestrado.

Ao meu AMIGO João, pelo lanche de todas as manhãs e pelo HUMOR FANTÁSTICO.

A todos os MEUS atuais e ex-colegas de trabalho.

Enfim, a todas as pessoas, que direta ou indiretamente, contribuíram para a concretização deste sonho.

RESUMO

Este trabalho visa a modelagem e desenvolvimento do agente DICA e da Ontologia ONTOGRAM usada no Ambiente de Avaliação Educacional integrado ao *software* Virtual-TANEB, tendo como objetivo a disponibilização de informações relevantes e confiáveis sobre o desempenho dos alunos na resolução de questões de matemática (grandezas e medidas) no quinto ano do ensino fundamental (antiga quarta série), proporcionando um diagnóstico mais preciso da situação do aluno nesta área de conhecimento.

Palavras-Chave: Agente de Software. Ontologia. Avaliação Educacional. Virtual-TANEB.

ABSTRACT

This work is related to the modeling and development of agent "DICA" and the ONTOGRAM Ontology used in an Educational Assessment Environment that is integrated to the Virtual-TANEB software, having as main goal the availability of relevant and reliable information about student performance when solving math questions (quantities and measures) in the fifth year of elementary school (previous fourth grade), providing a more accurate diagnosis of the situation of students in this area.

Keywords: Software Agent. Ontology. Educational Evaluation. Virtual-TANEB.

LISTA DE FIGURAS

Figura 1 - Tipos de Gestão Escolar.....	19
Figura 2 - Estrutura de um Agente.	28
Figura 3 - Comunicação direta entre agentes.	29
Figura 4 - Comunicação por comunicação assistida.....	30
Figura 5 - Estrutura de um Sistema Multiagente.	33
Figura 6 - JADE Remote Agent Management GUI (RMA).....	35
Figura 7 - Ciclo de Vida de um agente definido pela FIPA.....	37
Figura 8 - Tipos de Ontologias de acordo com seu nível de dependência de uma determinada tarefa ou ponto de vista.	41
Figura 9 - Arquitetura do <i>Software Virtual-TANEB</i>	46
Figura 10 - Visão geral do <i>Software Virtual-TANEB</i>	47
Figura 11 - Interface de autenticação no <i>Software Virtual-TANEB</i>	49
Figura 12 - Interface para recuperar senha no <i>Software Virtual-TANEB</i>	49
Figura 13 - Interface da página principal do usuário “Administrador do Sistema” no <i>Software Virtual-TANEB</i>	50
Figura 14 - Interface de cadastro de usuários no <i>Software Virtual-TANEB</i>	51
Figura 15 - Interface de visualização de itens cadastrados no <i>Software Virtual-TANEB</i>	52
Figura 16 - Estrutura do modelo proposto.....	54
Figura 17 - Diagrama de Caso de Uso “Realizar Prova”.	56
Figura 18 - Visualização de modelos abstraindo conceitos e relacionamentos na metodologia TROPOS.....	57
Figura 19 - Modelo de Raciocínio Estratégico do agente DICA.	58
Figura 20 - Hierarquia de Classes da Ontologia ONTOGRAM.....	59
Figura 21 - Rede semântica com Instâncias da subclasse “volume e massa” da Ontologia ONTOGRAM.....	60
Figura 22 - Rede semântica com Instâncias da subclasse “tempo e comprimento” da Ontologia ONTOGRAM.....	60

Figura 23 - Rede semântica com população das subclasses (volume, tempo, massa e comprimento) da Ontologia ONTOGRAM.	61
Figura 24 - Parte da modelagem do BD do Ambiente de Avaliação do VT.....	62
Figura 25 - Página principal do aluno no ambiente de avaliação do <i>Software Virtual-TANEB</i>	63
Figura 26 - Questão no ambiente feminino de avaliação do <i>Software Virtual-TANEB</i> com referência o descritor D8.	64
Figura 27 - Questão no ambiente feminino de avaliação do <i>Software Virtual-TANEB</i> com referência o descritor D11.	65
Figura 28 - Questão no ambiente masculino de avaliação do <i>Software Virtual-TANEB</i> com referência o descritor D9.	66
Figura 29 - Questão no ambiente masculino de avaliação do <i>Software Virtual-TANEB</i> com referência o descritor D10.	67
Figura 30 - Solicitação de DICA no ambiente masculino de avaliação do <i>Software Virtual-TANEB</i>	67
Figura 31 - Solicitação de DICA no ambiente feminino de avaliação do <i>Software Virtual-TANEB</i>	68
Figura 32 - Finalização de prova no ambiente de avaliação do <i>Software Virtual-TANEB</i>	68
Figura 33 - Inexistência de prova a ser realizada no ambiente de avaliação do <i>Software Virtual-TANEB</i>	69
Figura 34 - Desempenho do aluno no ambiente de avaliação do <i>Software Virtual-TANEB</i>	70
Figura 35 - Trecho do código da classe cadastrarDAO.java do pacote virtualTaneb.Item do <i>Software Virtual-TANEB</i>	71

LISTA DE TABELAS

Tabela 1 - Estrutura da Educação Brasileira	18
Tabela 2 - Matriz de Referência de Matemática: Tema II (grandezas e medidas) e seus Descritores - 4 ^a série/5 ^o ano do Ensino Fundamental	24
Tabela 3 - Documentação do Caso de Uso realizar prova	56
Tabela 4 - Parte dos procedimentos aplicados nos testes de unidade e testes de funcionalidade	72

LISTA DE ABREVIATURAS E SIGLAS

ACC	Agent Communication Channel
ACL	Tipo de mensagem utilizada pelos Agentes
AID	Lista de identificadores de agentes
CAIs	Instrução Assistida por Computador
CONSED	Conselho Nacional do Livro Didático
DAML	DARPA Agent Markup Language
DF	Directory Facilitador
ENCCEJA	Exame Nacional para Certificação de Competências de Jovens e Adultos
ENEM	Exame Nacional do Ensino Médio
FIPA	Foundation for Intelligent Physical Agents
FIPA-ACL	Linguagem de comunicação entre agentes
GUI	Interface Gráfica com o Usuário
HTML	HyperText Markup Language
IDE	Ambiente Integrado de Desenvolvimento
INEP	Instituto Nacional de Estudos e Pesquisas Educacionais Anísio Teixeira
JADE	Java Agent DEvelopment Framework
JavaScript	Linguagem de Programação para Web
JOE	Java Ontology Editor
JSP	JavaServer Pages
KQML	Knowledge Query and Manipulation Language
Logado	Usuário cadastrado e autenticado no sistema
Login	Nome de usuário com sua respectiva senha
MAS	Agent Managment System
MEC	Ministério da Educação
OCDE	Países participantes da organização para a cooperação e desenvolvimento econômico
OIL	Ontology Inference Layer (ou Ontology Interchange Language)
ONTOEDIT	Ambiente gráfico para edição de Ontologias
ONTOGRAM	Ontologia no domínio da matemática, grandezas e medidas.

ONTOLÍNGUA	Ambiente distribuído para pesquisar, criar, editar e uso de Ontologias
PDE	Plano de Desenvolvimento da Escola
PISA	Programa Internacional de Avaliação de Alunos
PROTÉGÉ	Ferramenta para criar, editar, modificar e uso de Ontologias
PROUNI	Programa Universidade para Todos
RDF	Resource Description Framework
RMA	Remote Management Agent
SAEB	Sistema de Avaliação da Educação Básica
TAOM4E	Tool for Agent Oriented Modeling
Thread	Forma de um processo dividir a si mesmo em duas ou mais tarefas que podem ser executadas concorrentemente
TRI	Teoria da Resposta ao Item
TROPOS	Requirements-Driven Development for Agent Software
UML	Unified Modeling Language
VT	Virtual-TANEB
WEB	World Wide Web
XML	eXtensible Markup Language

SUMÁRIO

LISTA DE FIGURAS.....	7
LISTA DE TABELAS	9
LISTA DE ABREVIATURAS E SIGLAS.....	10
1 INTRODUÇÃO.....	14
1.1 Justificativa e Relevância.....	14
1.2 Objetivos do Trabalho.....	15
1.3 Estruturação.....	15
2 EDUCAÇÃO BÁSICA BRASILEIRA	18
2.1 Gestão Escolar	19
2.2 Mecanismos de Avaliação da Educação Básica.....	20
2.3 Metodologia Adotada	24
2.4 Teoria da Resposta ao Item.....	25
2.5 Considerações Finais.....	26
3 AGENTES DE SOFTWARE	27
3.1 Agentes de Software.....	27
3.2 Comunicação entre Agentes.....	29
3.2.1 Comunicação Direta	29
3.2.2 Comunicação Indireta.....	30
3.3 Linguagens de Comunicação entre Agentes	30
3.3.1 KQML.....	31
3.3.2 FIPA-ACL.....	31
3.4 Sistemas Multiagentes.....	32
3.5 Desenvolvimento de Agentes	33
3.5.1 Framework JADE.....	33
3.5.2 JADE e FIPA.....	36
3.6 Considerações Finais.....	38
4 ONTOLOGIA	39
4.1 Tipos de Ontologias	40
4.2 Ferramentas para construir Ontologias.....	42
4.3 Trabalhos Relacionados	43
4.4 Considerações Finais.....	44
5 SOFTWARE VIRTUAL-TANEB	45
5.1 Arquitetura geral do Software Virtual-TANEB	45

5.2 Ambiente do Software Virtual-TANEB.....	48
5.3 Considerações Finais.....	53
6 MODELAGEM DO AMBIENTE DE AVALIAÇÃO DO SOFTWARE VIRTUAL-TANEB	54
6.1 Modelo Proposto.....	54
6.2 Diagrama de Caso de Uso do Ambiente de Avaliação do Virtual-TANEB	55
6.3 Modelagem do agente DICA.....	57
6.4 ONTOGRAM.....	58
6.5 Modelagem do Banco de Dados.....	61
6.6 Considerações Finais.....	62
7 IMPLEMENTAÇÃO E TESTE DO AMBIENTE DE AVALIAÇÃO DO SOFTWARE VIRTUAL-TANEB	63
7.1 Ambiente de Avaliação	63
7.2 Teste no Virtual-TANEB.....	70
7.3 Considerações Finais.....	73
8 CONCLUSÃO	74
8.1 Resultados Alcançados.....	74
8.2 Principais Contribuições.....	74
8.3 Perspectivas Futuras	75
REFERÊNCIAS	77
APÊNDICE	84
ANEXO A	89
ANEXO B	92

1 INTRODUÇÃO

Com a introdução da informática na educação surgem novas possibilidades e desafios, que conduzem à busca de formas inovadoras de ensinar e de aprender, alterando a concepção de ensino e aprendizagem. Os sistemas baseados em computador objetivam auxiliar no processo de aquisição do conhecimento, automatizando e apoiando a realização das inúmeras atividades humanas. Neste contexto, a criação de metodologias para o desenvolvimento de aplicações baseada em agentes de *Software* e Ontologias, tem sido item de pesquisa constante, por demonstrar grandes diferenciais na construção de sistemas de alta complexidade, onde é necessário operar em diversos ambientes imprevisíveis e com mudanças frequentes. As técnicas da Inteligência Artificial são utilizadas em diversas aplicações, que vão desde pequenos sistemas, até sistemas complexos com missão crítica para aplicações industriais (BELLIFEMINE; CAIRE; GREENWOOD, 2007).

Diante disso, há indícios bastante evidentes que estamos ingressando numa nova era em relação à Educação Básica Brasileira. Assim, segundo Novello (2005), o Sistema Nacional de Avaliação da Educação Básica (SAEB) tem como principal objetivo oferecer subsídios para a formulação, reformulação e monitoramento de políticas públicas, contribuindo, dessa maneira, para a universalização do acesso e a ampliação da qualidade, da equidade e da eficiência da Educação Brasileira.

1.1 Justificativa e Relevância

Com o intuito de disponibilização de dados, informações relevantes e confiáveis, e um maior preparo por parte dos gestores de política pública para lidar com os meios de informação, surge desta forma, a referida proposta de dissertação de mestrado, que desenvolve uma modelagem e implementação do agente DICA (Recurso da Inteligência Artificial) e da Ontologia ONTOGRAM (Engenharia do Conhecimento) para integrar ao *Software* Virtual-TANEB.

Este sistema é baseado na Teoria da Resposta ao Item (TRI) (ANDRADE; TAVARES; VALLE, 2000), visando oferecer um suporte à gestão educacional através do SAEB na aquisição de informações sobre o desempenho dos alunos na resolução de questões perante o Ensino da Matemática (grandezas e medidas) (WAKABAYASHI, 2007), no 5º ano do Ensino Fundamental (antiga 4ª série), proporcionando um diagnóstico mais preciso da situação do aluno nesta área de conhecimento.

1.2 Objetivos do Trabalho

Esta pesquisa tem como objetivo geral a utilização da Inteligência Artificial e parte da Engenharia do Conhecimento no desenvolvimento de um agente de *Software* denominado DICA e de uma Ontologia (ONTOGRAM) no domínio da matemática no contexto de grandezas e medidas, para ser integrado ao *Software* Virtual – TANEB durante as avaliações realizadas pelos alunos, que esse Ambiente proporciona.

Para alcançar esse objetivo geral, planeja-se atingir os seguintes objetivos específicos:

- Entender a estrutura da Gestão Educacional do Sistema de Avaliação da Educação Básica (SAEB);
- Analisar aspectos e estrutura dos agentes e Ontologias;
- Analisar a estrutura do Ambiente Virtual - TANEB;
- Contribuir com uma modelagem e implementação do Agente DICA para integrar ao *Software* Virtual - TANEB;

1.3 Estruturação

A estrutura dessa dissertação compreende oito capítulos, sendo eles, incluindo esta introdução, os seguintes:

O Capítulo 2 traz uma revisão referente a Educação Básica, a Gestão Escolar e dos Sistemas de Avaliação Brasileiros, englobando desde a parte histórica até a forma de aplicação destas avaliações, passando ainda por uma análise detalhada dos diversos tipos de sistemas de avaliações realizados aqui no Brasil para avaliar o Ensino Brasileiro, são eles: Prova Brasil, o Programa Internacional de Avaliação de Alunos (PISA), o Exame Nacional do Ensino Médio (ENEM), o Exame Nacional para Certificação de Competências de Jovens e Adultos (ENCCEJA), a Provinha Brasil e o SAEB, este último torna-se um dos tópicos mais importantes para o embasamento do presente trabalho; Por último, é realizada uma abordagem sobre a Teoria da Resposta ao Item (TRI), definida como uma modelagem estatística utilizada em medidas, principalmente na área de avaliação de habilidades e conhecimentos.

Já o capítulo 3, aborda os agentes de *Software* e os Sistemas Multiagentes, assunto de grande importância no contexto dessa pesquisa; Sintetiza-se a comunicação entre agentes de forma direta e indireta; aborda as linguagens de comunicação entre agentes, KQM e FIPA-ACL, pois a forma de interação que ocorre entre os agentes é um fator muito importante na integração destes; E, finalmente, apresenta-se uma das ferramentas utilizadas para o desenvolvimento de agentes e usada no presente trabalho, o *framework* JADE.

O capítulo 4 apresenta as Ontologias, utilizadas para o armazenamento do conhecimento diante de um domínio de interesse; Expõe-se os componentes básicos de uma Ontologia, que servem para representar conhecimento, são eles: conceitos, relações, instâncias e axiomas; Apresenta-se algumas ferramentas para se construir Ontologias, como: *Ontolíngua*, *Java Ontology Editor* (JOE), *OntoEdit* e o *Protégé* (utilizada no presente trabalho). E, por último, alguns trabalhos relacionados às ontologias são mostrados.

Já no capítulo 5, uma arquitetura geral do *Software* Virtual-TANEB é apresentada; Também, sintetiza-se a visão geral do *Software* VT; E, finalmente, é exposto algumas Interfaces do Ambiente do *Software* Virtual-TANEB.

Em seguida, o capítulo 6 descreve o modelo proposto do Ambiente de Avaliação do *Software* Virtual-TANEB e que se caracteriza como uma das principais contribuições científicas resultantes deste trabalho; Apresenta-se o diagrama de Caso de Uso do Ambiente de Avaliação do VT; Mostra a modelagem do Agente

DICA, utilizando à metodologia TROPOS; Apresenta a estrutura da Ontologia ONTOGRAM, baseada no domínio da matemática do quinto ano do Ensino Fundamental, referente ao tema grandezas e medidas. E, finalmente, apresenta-se a especificação do Banco de Dados utilizado no Ambiente de Avaliação do VT.

O capítulo 7 mostra a implementação e testes com o protótipo do Ambiente de Avaliação do *Software* VT, demonstrando algumas interfaces.

E, finalmente, o capítulo 8 expõe as conclusões acerca do trabalho, com ênfase nos trabalhos relacionados, nas principais contribuições, nos resultados alcançados e nas perspectivas futuras.

2 EDUCAÇÃO BÁSICA BRASILEIRA

A Educação Básica (ou Ensino Básico) no Brasil vem sofrendo mudanças gradativamente e ganhando contornos bastante complexos principalmente nos últimos oito anos (CURY, 2002). Neste cenário, Cury (2002) ainda relata que analisar a Educação Brasileira não é fácil, exatamente porque as contingências que a cercam são múltiplas e os fatores que a determinam têm sido objeto de leis, políticas e programas nacionais, alguns dos quais em convênio com órgãos internacionais.

O ensino básico é um conceito mais do que inovador para um país, pois designa um conjunto de atividades educativas, formais, não formais e informais. Com isso, resulta que a Educação Infantil é a base da Educação Básica, o Ensino Fundamental é o seu tronco e o ensino médio é seu acabamento. É partindo de uma visão do todo como base que se pode ter uma visão consequente das partes (CURY, 2002).

A tabela 1 mostra a estrutura Educacional Brasileira.

Tabela 1 - Estrutura da Educação Brasileira.

EDUCAÇÃO INFANTIL	ENSINO FUNDAMENTAL	ENSINO FUNDAMENTAL (9 anos)	ENSINO MÉDIO
Creche	Alfabetização	1º ano	1º ano
Pré-Escola	1ª Série	2º ano	2º ano
	2ª Série	3º ano	3º ano
	3ª Série	4º ano	
	4ª Série	5º ano	
	5ª Série	6º ano	
	6ª Série	7º ano	
	7ª Série	8º ano	
	8ª Série	9º ano	

Para Frigotto e Ciavatta (2003), a Educação Básica deve estruturar-se em consonância com o avanço do conhecimento científico e tecnológico, fazendo da cultura técnica um componente da formação geral, articulada com o trabalho produtivo. Já na visão institucional, a Educação é um direito social de todos, passa a

ser cada vez mais encarada como um serviço. Isso pressupõe a vinculação da ciência com a prática, bem como a superação das dicotomias entre humanismo e tecnologia, e entre formação teórica geral e técnica instrumental.

2.1 Gestão Escolar

Segundo o CONSED (2008), uma gestão escolar de qualidade é pré-condição para alcançar as cinco grandes metas educacionais, até 2022, ano do bicentenário da Independência do Brasil: 1) Todas as crianças de 4 a 17 anos estarão na escola; 2) Toda criança de 8 anos saberá ler e escrever; 3) Todo aluno aprenderá o que é esperado para a sua série; 4) Todos os alunos vão concluir a Educação Básica; e 5) Os recursos da Educação serão garantidos e bem geridos.

Diante da sistemática organizacional no setor educacional, é importante analisar pelo menos de forma sucinta os tipos de gestão e seus respectivos objetivos no espaço educacional. Dessa forma, o CONSED (2008) descreve os tipos de gestões. (*Vide Figura 1*)



Figura 1 - Tipos de Gestão Escolar.

A gestão de resultados educacionais abrange processos e práticas de gestão voltadas para assegurar a melhoria dos resultados de desempenho da escola como o rendimento, frequência e a proficiência dos alunos. Esta gestão destaca-se como indicadores de gestão de resultados: avaliação e melhoria contínua do projeto pedagógico da escola; análise, divulgação e utilização dos resultados alcançados.

Já a gestão participativa cuida dos processos e práticas que respondam ao princípio de gestão democrática do ensino público. São destacados como indicadores de qualidade: planejamento e avaliação do projeto pedagógico e dos planos de ação da escola, de forma participativa; a atuação de órgãos colegiados como os conselhos escolares, grêmios estudantis dentre outros.

Outra dimensão de avaliação citado pelo CONSED (2008) é a gestão pedagógica que executa práticas de gestão pedagógica orientados para assegurar a aprendizagem dos alunos, em consonância com o projeto pedagógico da escola. Destacam-se como indicadores de qualidade: atualização periódica da proposta curricular; inovação pedagógica e políticas de inclusão com equidade, etc.

A gestão de pessoas também faz parte das dimensões de avaliações, que tem como intuito a realização de práticas de gestão, visando envolvimento e compromisso de professores e demais profissionais, pais e alunos com o projeto pedagógico da escola. Destacam-se como indicadores de qualidade: a integração entre os profissionais da escola, pais e alunos; o desenvolvimento profissional contínuo; a avaliação de desempenho e reconhecimento do trabalho escolar.

Por fim, a gestão que abrange processos e práticas de gestão dos serviços de apoio, recursos físicos e financeiros é chamada de gestão de serviços e recursos. Os indicadores de qualidade desta gestão são: a organização dos registros escolares; a utilização das instalações e equipamentos, preservação do patrimônio escolar e aplicação de recursos didáticos e financeiros, dentre outros.

2.2 Mecanismos de Avaliação da Educação Básica

Diante dos ensinamentos de Rodrigues (2007), o processo de avaliação educacional é complexo e abrangente e, por isso, definido de várias formas. A

realização da avaliação no âmbito educacional pode alcançar diversos propósitos e ser realizada sob várias perspectivas. O autor ainda conceitua de uma outra forma a avaliação da aprendizagem educacional, como sendo qualquer método usado para melhor entender o conhecimento corrente que um estudante possui, incluindo desde a simples avaliação realizada pelo julgamento subjetivo baseado numa única observação do desempenho do estudante, até um teste padronizado complexo que exige técnicas de análise sofisticadas (RODRIGUES, 2007).

Neste contexto, Gama (1998) relata sobre a Teoria das Inteligências Múltiplas e suas implicações para a Educação. A Teoria das Inteligências Múltiplas é uma alternativa para o conceito de inteligência como uma capacidade inata, geral e única, que permite aos indivíduos uma performance, maior ou menor, em qualquer área de atuação (HOWARD GARDNER, 1985 apud GAMA, 1998). Portanto, as implicações da teoria de Gardner para a Educação são claras quando se analisa a importância dada às diversas formas de pensamento, aos estágios de desenvolvimento das várias inteligências e à relação existente entre estes estágios, a aquisição de conhecimento e a cultura.

O INEP é o Instituto Nacional de Estudos e Pesquisas Educacionais “Anísio Teixeira” que é vinculado ao Ministério da Educação (MEC). O INEP tem como missão promover estudos, pesquisas e avaliações sobre o sistema educacional brasileiro com o objetivo de subsidiar a formulação e implementação de políticas públicas para a área educacional a partir de parâmetros de qualidade e equidade, bem como produzir informações claras e confiáveis aos gestores, pesquisadores, educadores e público em geral (BRASIL, 2009).

Com o intuito de gerarem dados e estudos educacionais, o INEP também tem como função realizar levantamentos estatísticos e avaliativos em algumas etapas da Educação Básica (BRASIL, 2009). Portanto, existem vários Mecanismos de Avaliação na Educação Brasileira, tais como:

- **Prova Brasil** - é um instrumento de avaliação do sistema educacional brasileiro criado para auxiliar no desenvolvimento e implementação de políticas públicas educacionais, tendo como principal função avaliar o que as crianças e jovens sabem em termos de habilidades e competências, e não simplesmente de conteúdo, sendo aplicada a cada dois anos a alunos

de séries finais de ciclos da Educação Básica: 4ª e 8ª série do Ensino Fundamental (INEP, 1a) (INEP, 1c) (BRASIL, 2009). A Prova Brasil foi criada em 2005, e de acordo com o INEP (1a), avalia as escolas públicas localizadas em área urbana nas habilidades em Língua Portuguesa (foco em leitura) e Matemática (foco na resolução de problemas).

- **Programa Internacional de Avaliação de Alunos (PISA)** - é considerado o Programa Internacional de Avaliação de alunos, tem como foco de avaliação internacional padronizada, desenvolvido conjuntamente pelos países participantes da organização para a cooperação e desenvolvimento econômico (OCDE), aplicada a alunos de 15 anos, cujas avaliações são realizadas a cada três anos, abrange as áreas de Linguagem, Matemática e Ciências (Sociais, Naturais, entre outras), não somente quanto ao domínio curricular, mas também quanto aos conhecimentos relevantes e às habilidades necessárias à vida adulta (BRASIL, 2009).

- **Exame Nacional do Ensino Médio (ENEM)** - foi criado em 1998, é outro mecanismo de avaliação, sendo individual, de caráter voluntário, oferecido anualmente aos estudantes que estão concluindo ou que já concluíram o Ensino Médio em anos anteriores que possibilita uma referência para auto-avaliação, a partir das competências e habilidades que o estruturam (BRASIL, 2009). O MEC (1a) relata que o ENEM é utilizado como critério de seleção para os estudantes que pretendem concorrer a uma bolsa no Programa Universidade para Todos (ProUni). Atualmente, diversas Universidades utilizam o resultado ENEM como critério de seleção para o ingresso no Ensino Superior, seja complementando ou substituindo o vestibular. E a partir de 2011, 35 das 59 universidades federais do Brasil terão o ENEM como a única prova de seleção para o ingresso de novos estudantes (FARRUGIA, 2010).

- **Exame Nacional para Certificação de Competências de Jovens e Adultos (ENCCEJA)** - é uma ferramenta de avaliação da Educação Brasileira onde tem o intuito de avaliar as habilidades e competências básicas de jovens e adultos que não tiveram oportunidade de acesso à escolaridade regular na idade apropriada (MEC, 1b). Esse mecanismo de

avaliação é aplicado a brasileiros residentes no Brasil e no exterior, onde o participante se submete a uma prova e, alcançando o mínimo de pontos exigido, obtém a certificação de conclusão daquela etapa educacional, portanto, o exame é aplicado anualmente e visa, ainda, sinalizar, para educadores, estudantes e interessados, a natureza e a função de uma avaliação de competências fundamentais ao exercício pleno da cidadania (BRASIL, 2009) (MEC, 1b).

- **Provinha Brasil** - é uma avaliação diagnóstica aplicada aos alunos matriculados no segundo ano do Ensino Fundamental, com a intenção de oferecer aos professores e gestores escolares um instrumento que permita acompanhar, avaliar e melhorar a qualidade da alfabetização e do levantamento inicial oferecidos às crianças (MEC, 1c). Outro ponto crucial a ser considerado em relação a esse mecanismo, é que foi criada em consonância com o objetivo do Plano de Desenvolvimento da Educação (PDE) do MEC de viabilizar ações que contribuam para a equidade e a qualidade da Educação pública Brasileira (BRASIL, 2009).

- **Sistema Nacional de Avaliação da Educação Básica (SAEB)** - é o principal mecanismo de estudo do referido trabalho, onde pode-se afirmar que esse Sistema tem o objetivo de avaliar a educação em larga escala oferecendo subsídios à formulação, reformulação e monitoramento de políticas públicas e programas de intervenção ajustados às necessidades diagnosticadas nas áreas e etapas de ensino avaliadas, identificando os problemas e as diferenças regionais do ensino (INEP, 1b). O SAEB foi aplicado pela primeira vez em 1990. Nesta ferramenta de avaliação, os alunos fazem prova de Língua Portuguesa (foco em leitura) e Matemática (foco na resolução de problemas), sendo realizada em dois em dois anos, com foco nos estudantes de 4^a e 8^a séries do Ensino Fundamental e também estudantes do 3^o ano do Ensino Médio da rede pública e da rede privada (INEP, 1a).

Entretanto, faz-se necessário destacar que segundo Castro (1998), além de identificar os níveis de desempenho e aprendizagem efetivamente alcançados pelos alunos, os resultados obtidos por esse mecanismo (SAEB), permitem estabelecer algumas correlações importantes sobre os fatores associados ao

rendimento escolar. Com isso, a avaliação dos alunos é feita utilizando-se uma grande quantidade de questões, o que lhe confere maior validade curricular, pois contempla uma amplitude maior de conteúdos e habilidades, abrangendo grande parte daquilo que é proposto nos currículos estaduais, por isso, com as informações obtidas, é possível monitorar as políticas voltadas para a melhoria da qualidade, da equidade e da eficiência do ensino básico brasileiro (CASTRO, 1998).

2.3 Metodologia Adotada

O SAEB tem utilizados diversas metodologias, desde a sua implantação, para atender a demanda social por informações sobre a qualidade, equidade e eficiência do ensino brasileiro (BRASIL, 2009).

A Matriz de Referência (em anexo A) é uma das metodologias adotadas pelo SAEB, pois consiste num documento onde estão descritas as orientações para a elaboração dos itens (questões) das provas do SAEB. Diante disso, sua composição é realizada numa ampla consulta e análise das propostas curriculares dos estados brasileiros, professores das capitais, regentes das redes municipal, estadual e privada das séries que serão aplicadas as provas, nas disciplinas de Língua Portuguesa e Matemática (BRASIL, 2009).

Além disso, a Matriz de Referência é utilizada para formulação das questões que compõem a prova, constituída por um conjunto de Descritores - cruzamento ou associação entre conteúdos curriculares e operações mentais, competências e habilidades, que estão organizados em tópicos, temas e assuntos, distribuídos para cada série/ano que o SAEB é aplicado (BRASIL, 2009).

Na tabela 2 podemos visualizar parte da Matriz de Referência em matemática da 4ª série (5º ano) do Ensino Fundamental, mostrando os Descritores do Tema II, grandezas e medidas.

Tabela 2 - Matriz de Referência de Matemática: Tema II (grandezas e medidas) e seus Descritores - 4ª série/5º ano do Ensino Fundamental.
Fonte: (BRASIL, 2009).

Descritores	4ª série (5º ano) EF
Estimar a medida de grandezas utilizando unidades de medida convencionais ou não	D6
Resolver problemas significativos utilizando unidades de medida padronizadas como km/m/cm/mm, kg/g/mg, l/ml	D7
Estabelecer relações entre unidades de medida de tempo	D8
Estabelecer relações entre o horário de início e término e/ou o intervalo da duração de um evento ou acontecimento	D9
Num problema, estabelecer trocas entre cédulas e moedas do sistema monetário brasileiro, em função de seus valores	D10
Resolver problema envolvendo o cálculo do perímetro de figuras planas, desenhadas em malhas quadriculadas	D11
Resolver problema envolvendo o cálculo ou estimativa de áreas de figuras planas, desenhadas em malhas quadriculadas	D12

2.4 Teoria da Resposta ao Item

Segundo Reise *et al.*(2005) apud Tezza *et al.*(2009) a Teoria da Resposta ao Item (TRI) é definida como um conjunto de modelos matemáticos e estatísticos que são utilizados na análise de itens e escalas, para criar e administrar medidas, e para medir indivíduos ou organizações. Com isso, Andrade *et al.*(2000) mostra estatisticamente que uma pessoa j com maior proficiência será capaz de acertar um item i da prova, e também um item ser acertado por acaso por um indivíduo sem habilidade para respondê-lo.

Além disso, a TRI é sustentada basicamente por três fundamentos, são eles: função resposta de um item, função de informação e invariância (REISE *et al.*, 2005 apud TEZZA *et al.*, 2009).

Para Andrade *et al.*(2000), a TRI foi usada no Brasil pela primeira vez, em 1995 na análise dos dados do Sistema Nacional de Ensino Básico (SAEB). A introdução da Teoria da Resposta ao item permitiu que os desempenhos de alunos de quarta e oitava séries do Ensino Fundamental e de terceira série do Ensino Médio pudessem ser comparados e colocados em uma escala única de conhecimento.

2.5 Considerações Finais

Este capítulo apresentou uma revisão referente à Educação Básica, a Gestão Escolar e dos Sistemas de Avaliação Brasileiros, englobando desde a parte histórica até a forma de aplicação destas avaliações, abordando uma análise detalhada dos diversos tipos de sistemas de avaliações realizados aqui no Brasil para avaliar o ensino brasileiro, são eles: Prova Brasil, o PISA (Programa Internacional de Avaliação de Alunos), o ENEM (Exame Nacional do Ensino Médio), o ENCCEJA (Exame Nacional para Certificação de Competências de Jovens e Adultos), a Provinha Brasil e o SAEB, focando especificamente este último, com a utilização da metodologia, Matriz de Referência. Finalmente, foi realizada uma abordagem sobre a Teoria da Resposta ao Item (TRI), definida como uma modelagem estatística utilizada em medidas, principalmente na área de avaliação de habilidades e conhecimentos.

O próximo capítulo tratará dos Agentes e Sistemas Multiagentes, uma das contribuições e etapa importante para a evolução do referido trabalho.

3 AGENTES DE SOFTWARE

3.1 Agentes de Software

Atualmente, a criação de metodologias para o desenvolvimento de *Software* baseado em agentes tem sido item de pesquisa constante, pois tem mostrado grandes vantagens nas construções desses sistemas baseados em agentes e com alta complexidade, onde necessitam operar em diversos ambientes imprevisíveis, e com mudanças frequentes.

Um agente de *Software* é definido como sendo uma entidade autônoma que percebe seu ambiente através de sensores e age sobre o mesmo utilizando-se dos executores (também conhecido como atuadores), tendo como característica fundamental a autonomia (RUSSEL e NORVIG, 2004).

Com isso, um agente sendo uma entidade de *Software* que funciona de forma contínua e autônoma, também deve ser capaz de perceber e atuar no seu ambiente, de forma flexível e inteligente, sem requerer intervenção ou orientação humana constante, aprender através da experiência, comunicar-se e cooperar com outros agentes que porventura co-existam no mesmo ambiente (SILVA, 2006). Outro ponto a ser considerado sobre os agentes, é que, para satisfazer seus objetivos, pode mover-se em locais distintos.

Neste cenário, segundo Silva (2006) existem diversos tipos de agentes e suas mais variadas diferenças, podendo ser: móveis, situados ou estacionários, competitivos, coordenados ou colaborativos, reativos e cognitivos.

Os agentes móveis tem como característica principal a mobilidade, isto é, uma capacidade de mover-se seja por uma rede interna local (intranet) ou até mesmo, pela Internet, transportando-se pelas plataformas levando dados e códigos.

Já os agentes situados ou estacionários, como são conhecidos, tem característica oposta aos móveis, pois são fixos em um mesmo ambiente e/ou plataforma, não se movimentam em uma rede e muito menos na Internet.

Os agentes competitivos se caracterizam por competir entre si para a realização de seus objetivos ou tarefas, não havendo colaboração entre os agentes.

Outro tipo de agente a ser considerado são os agentes coordenados ou colaborativos, como são conhecidos, cuja finalidade é de alcançar um objetivo maior, pois realizam tarefas específicas, porém coordenando-as entre si de forma que suas atividades se completem.

Os agentes reativos são aqueles que reagem a estímulos sem ter memória do que já foi realizado no passado e nem previsão da ação a ser tomada no futuro, com isso, não possuem representação do seu ambiente ou de outros agentes e são incapazes de prever e antecipar ações.

E, por fim, os agentes cognitivos, que tem característica opostas aos agentes reativos, pois esse tipo de agente pode raciocinar sobre as ações tomadas no passado e planejar ações a serem tomadas no futuro. Esses agentes são capazes de resolver problemas por ele mesmo, e com objetivos e planos explícitos os quais permitem atingir seu objetivo final. Outro ponto a ser considerado, é que os agentes cognitivos possuem uma representação interna e seus mecanismos de inferência lhe permitem atuar independentemente dos outros agentes e lhe dão uma grande flexibilidade na forma de expressão de seu comportamento (SILVA, 2006).

Na figura 2, a seguir, é mostrada com detalhe a estrutura de um Agente, tendo assim, um melhor entendimento sobre agentes e seu ambiente.

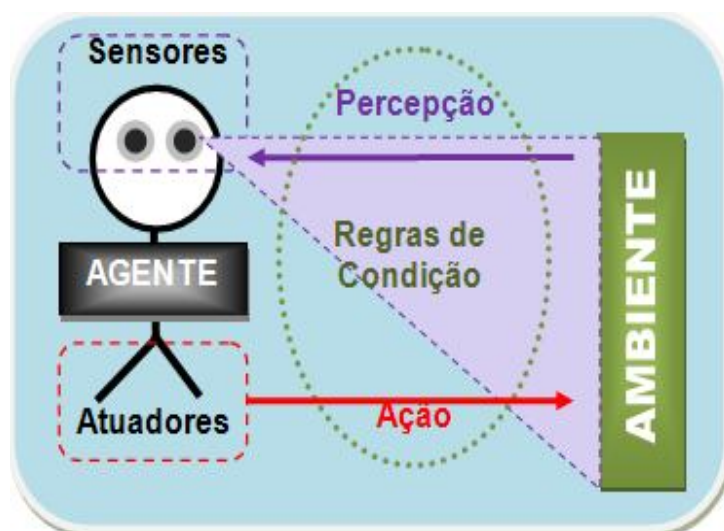


Figura 2 - Estrutura de um Agente.

Fonte: Adaptado de (BORDINI; HÜBNER; WOOLDRIDGE, 2007).

Na figura 2 é exibido o processo de um agente de *software* genérico com sua estrutura básica, possuindo memória interna que irá atualizar-se com a chegada

de novas percepções do ambiente, que são capturadas através de sensores. Assim, essa memória é utilizada nos procedimentos de tomada de decisão, os quais irão gerar ações que serão executadas através de executores, ou seja, atuadores (LINDOSO, 2006).

3.2 Comunicação entre Agentes

3.2.1 Comunicação Direta

De acordo com os ensinamentos de Gonzaga *et al.*(1998), a comunicação de forma direta é realizada através de trocas de mensagens. Na comunicação direta, ou comunicação via troca de mensagens direta, cada agente comunica diretamente com qualquer outro agente sem qualquer intermediário (SILVA, 2003).

A figura 3, a seguir, mostra este tipo de comunicação.

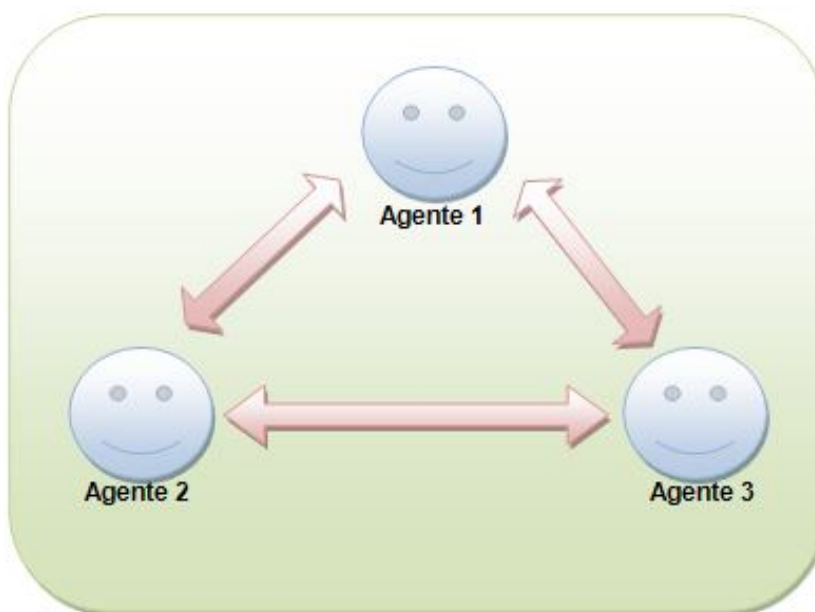


Figura 3 - Comunicação direta entre agentes.

Fonte: Adaptado de Silva (2003).

Na seção seguinte, é mostrada outra forma de comunicação entre agentes, ou seja, de maneira indireta.

3.2.2 Comunicação Indireta

A comunicação de forma indireta, utilizada no presente trabalho, pode ser através de um *blackboard*, ou seja, arquitetura que permite a integração de módulos ou programas individuais em uma aplicação única e integrada (GONZAGA *et al.*, 1998). Entretanto, a comunicação realizada por *blackboard* é mais simples, porém sua implementação em uma arquitetura distribuída não é eficiente.

Já Silva (2003), descreve que a comunicação indireta pode comunicar-se através de um agente “facilitador” especial em sistema “federado” (comunicação assistida), onde os agentes utilizam algum sistema ou agente especial para coordenar suas atividades. A figura 4, a seguir, mostra como os agentes se comunicam de forma indireta.

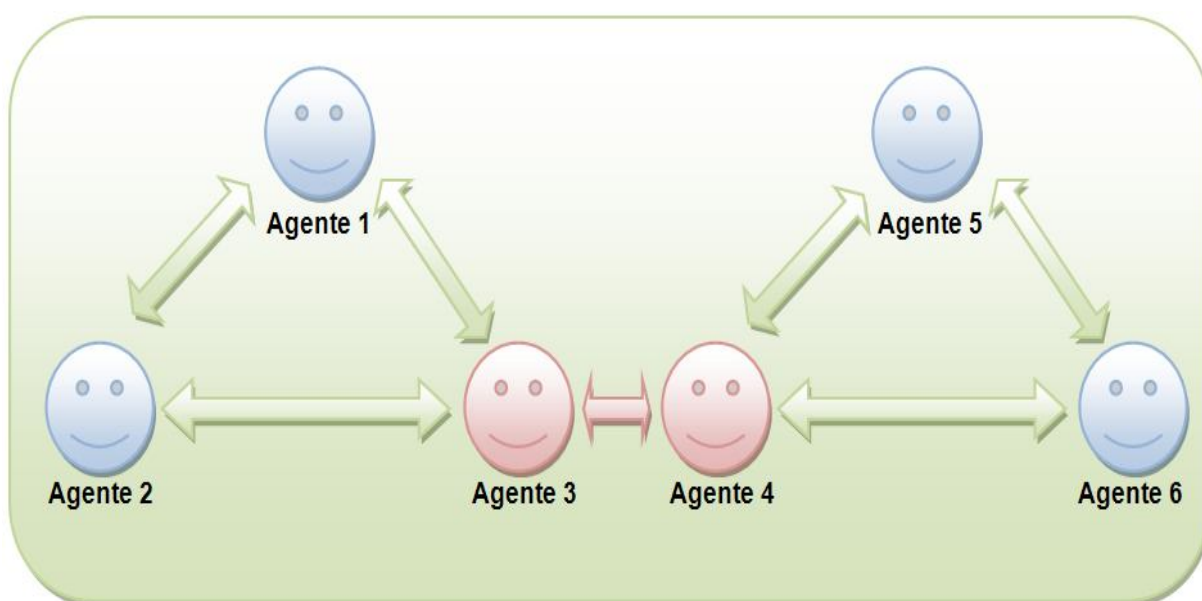


Figura 4 - Comunicação por comunicação assistida.

Fonte: Adaptado de (SILVA, 2003).

Assim, uma estrutura hierárquica de agentes é definida e a troca de mensagens dá-se através de agentes especiais designados facilitadores ou mediadores.

3.3 Linguagens de Comunicação entre Agentes

A forma de interação que ocorre entre os agentes é um fator muito importante na integração destes. As linguagens de comunicação e sua expressividade definem a capacidade de comunicação de cada agente. Ela deve ser universal e partilhada por todos os agentes, ser concisa e ter um número limitado com primitivas de comunicação (SILVA, 2003). Assim, as principais linguagens de comunicação entre agentes são a KQM e FIPA-ACL.

3.3.1 KQML

A Knowledge Query and Manipulation Language (KQML) (em anexo B) é uma linguagem e um protocolo de comunicação de alto nível para troca de mensagens independente de conteúdo e da Ontologia aplicável (SILVA, 2003). Desta forma, a linguagem KQML serve como um formato de mensagem e um protocolo de gerenciamento de mensagens, além disso, esta linguagem não se preocupa muito com o conteúdo da mensagem, mas sim com a especificação da informação necessária à compreensão do conteúdo.

Segundo Fialho (2003) apud Veras (2010), a KQML é uma linguagem versátil de propósito geral que suporta a comunicação entre vários agentes com um conjunto de primitivas reservadas chamadas *performative*. O autor adotou o termo *performative* para referir-se a uma declaração feita pelo usuário emissor e que é executada pelo agente receptor, simplesmente porque o emissor declara ou afirma. Esta linguagem, também possui como característica interessante, a de ser responsável por oferecer uma forma de acesso à informação mesmo para programas que não sejam com agentes.

3.3.2 FIPA-ACL

A FIPA-ACL (em anexo B) é uma linguagem de comunicação entre agentes, como o KQML, baseada em ações de fala e contém um conjunto de parâmetros de uma ou mais mensagens. A sua sintaxe é bastante semelhante ao KQML, porém o conjunto de performativas (atos comunicativos) é diferente (BELLIFEMINE; CAIRE; GREENWOOD, 2007).

Neste contexto, a especificação de uma mensagem FIPA-ACL consiste de um conjunto de tipos de mensagens e descrições dos efeitos da mensagem sobre os agentes que a enviam e sobre o que a recebem (BELLIFEMINE; CAIRE; GREENWOOD, 2007) (SILVA, 2003). Além disso, a estrutura de uma mensagem FIPA-CL é muito parecida com a estrutura das mensagens KQML, pois possui uma semântica bem definida.

3.4 Sistemas Multiagentes

De acordo com Gonzaga *et al.*(1998), os agentes computacionais são capazes de interagir com outros agentes e de participarem de sociedades em algumas situações específicas, portanto, uma sociedade é composta por dois ou mais agentes ou resolvedores de problemas, eles interagem através de um ambiente comum para solucionar um determinado problema. O autor ainda relata que esta sociedade é denominada de sistema multiagente, sendo definida como uma rede de resolvedores de problemas que trabalham juntos para resolver tarefas que estão além das suas capacidades individuais.

O sistema multiagente, ou seja, sociedade de agentes, pode ser classificado de três maneiras diferentes de acordo com Gonzaga *et al.*(1998):

- O tipo de agente;
- O número de agentes; e
- Suas regras de comportamento.

A classificação de acordo com o tipo de agente divide as sociedades em homogêneas, onde todos os agentes são idênticos, e heterogêneas, que possui agentes diferentes. Já a classificação segundo o número de agentes divide as sociedades em fechadas, quando há um número fixo e único de agentes, e abertas quando o número de agentes pode variar sendo incluídos ou excluídos novos agentes. E, por fim, a classificação de acordo com suas regras de comportamento, que divide as sociedades em baseadas em leis, quando existem regras que determinam o comportamento dos agentes, e sem lei, pois não há regras para reger os agentes.

A figura 5, a seguir, mostra uma visão geral de um sistema multiagente.

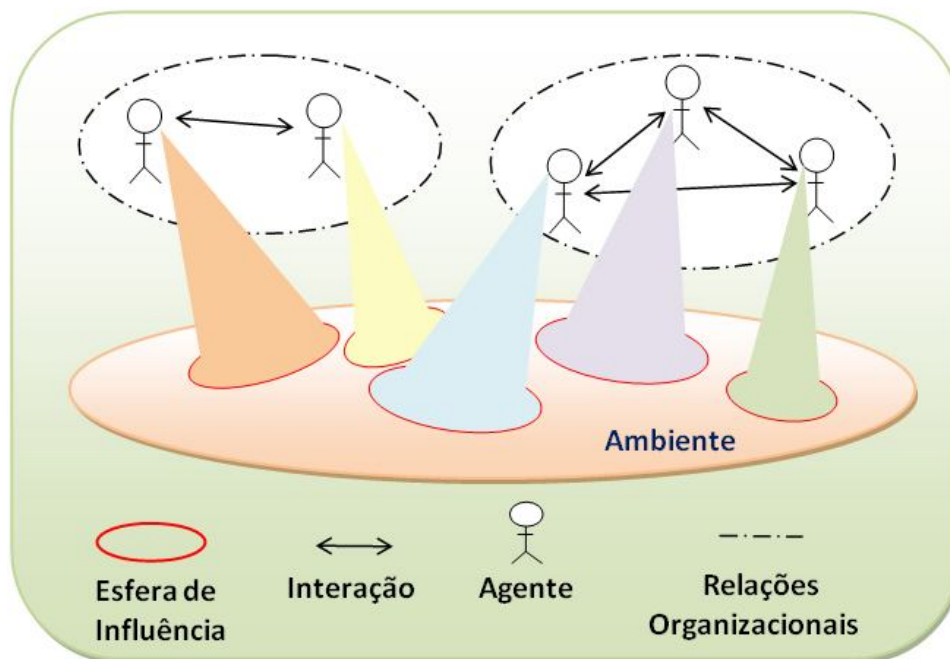


Figura 5 - Estrutura de um Sistema Multiagente.

Fonte: Adaptado de (BORDINI; HÜBNER; WOOLDRIDGE, 2007).

Na parte inferior da figura (BORDINI; HÜBNER; WOOLDRIDGE, 2007), é observado um ambiente compartilhado onde os agentes ocupam, cada agente tem uma "esfera de influência" neste ambiente, ou seja, uma área do ambiente que eles são capazes de controlar.

Alem disso, pode ser que um agente tenha a capacidade única de controle ou parte de seu ambiente (PEREIRA, 2006). Na parte superior do ambiente (BORDINI; HÜBNER; WOOLDRIDGE, 2007) é mostrado os próprios agentes, que estão em diferentes relações organizacionais. Finalmente, esses agentes terão algum conhecimento de si (PEREIRA, 2006), embora possa existir um caso em que um agente não tem conhecimento completo dos outros agentes no sistema.

3.5 Desenvolvimento de Agentes

3.5.1 Framework JADE

O *Java Agent DEvelopment Framework* (JADE) (em anexo B) é um ambiente para desenvolvimento de aplicações baseada em agentes conforme as especificações da *Foundation for Intelligent Physical Agents* (FIPA) para

interoperabilidade entre sistemas multiagentes totalmente implementado em JAVA (BELLIFEMINE; CAIRE; GREENWOOD, 2007).

De acordo com Silva (2003), o principal objetivo do *framework* JADE é simplificar e facilitar o desenvolvimento de sistemas multiagentes garantindo um padrão de interoperabilidade entre sistemas multiagentes através de um abrangente conjunto de agentes de serviços de sistema, pois facilitam e possibilitam a comunicação entre agentes, de acordo com as especificações da FIPA:

- Naming Service (serviço de nomes) e Yellow-Page Service (páginas amarelas);
- Transporte de mensagens;
- Serviços de codificação e decodificação de mensagens; e
- Biblioteca de protocolos de interação (padrão FIPA) pronta para ser usada.

Além disso, toda comunicação entre agentes no Framework JADE é realizada via troca de mensagens.

RMA (Remote Management Agent)

O gerenciador remoto de agentes considera-se como um console gráfico para o controle e gerenciamento da plataforma JADE, permitindo o controle dos estados do ciclo de vida de todos os agentes em execução inclusive os distribuídos. Além disso, serve como um controle principal onde estão centralizadas algumas funcionalidades como, por exemplo, chamar as outras ferramentas do JADE (BELLIFEMINE; CAIRE; GREENWOOD, 2007). (*Vide Figura 6*)

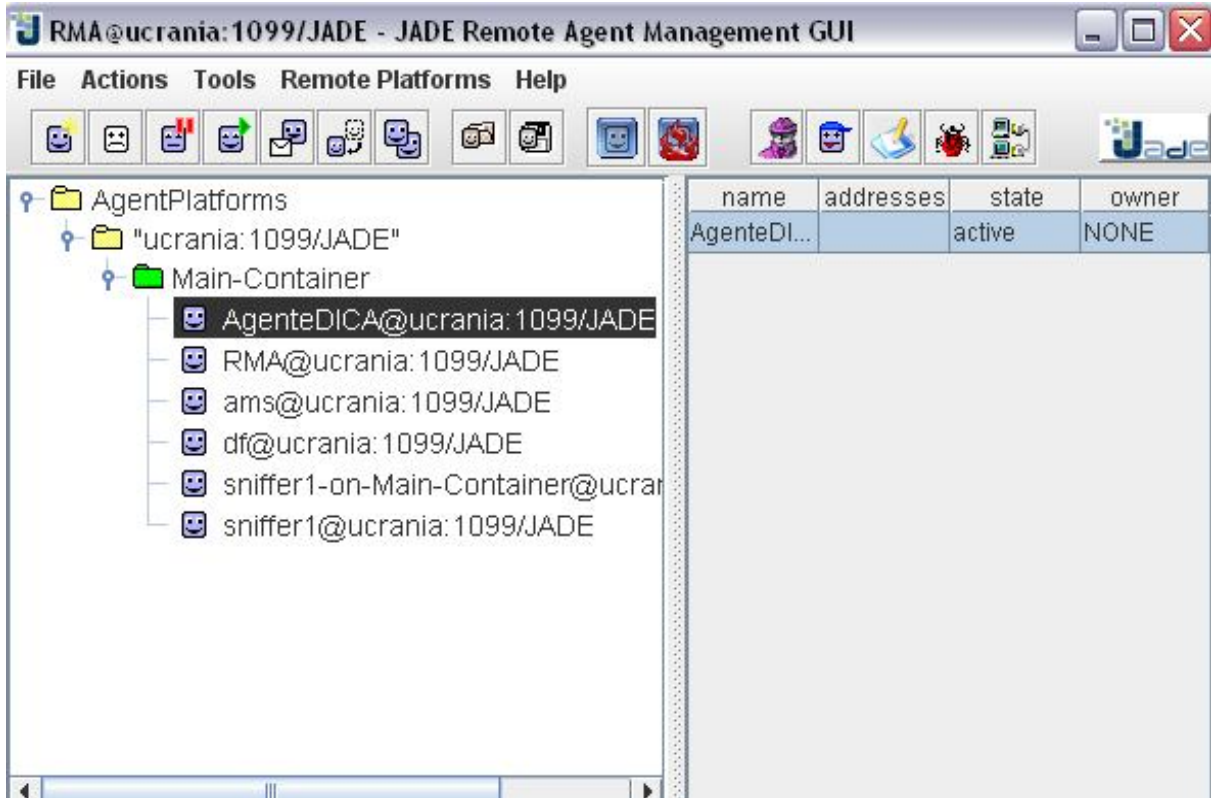


Figura 6 - JADE Remote Agent Management GUI (RMA).

Fonte: (BELLIFEMINE; CAIRE; GREENWOOD, 2007).

Diante do exposto na figura 6, o *Framework* JADE também possui algumas ferramentas e recursos para manipulação e monitoramento de Agentes JADE, são elas: Dummy Agent, Sniffer Agent, dentre outras.

O Dummy Agent (em anexo B) é uma ferramenta gráfica de monitoramento para agentes JADE. Neste cenário é possível criar e enviar mensagens ACL para outros agentes e listar todas as mensagens ACL enviadas e recebidas com informações detalhadas. Portanto, é possível salvar essa lista de mensagens em disco e recuperá-la depois (BELLIFEMINE; CAIRE; GREENWOOD, 2007).

Já o *Sniffer* Agente (em anexo B) é uma ferramenta que mostra a troca de mensagens graficamente, pois tem notação semelhante aos diagramas de seqüências da Linguagem de Modelagem Unificada (UML) entre determinados agentes. Neste contexto, quando um usuário decide fazer um *Sniffer* em um agente ou grupo de agentes, toda mensagem direcionada a este agente, ou grupo de agentes, é rastreada e disponibilizada na Interface Gráfica com o Usuário (GUI). Em virtude disso, o usuário poderá ver todas mensagens, salvá-las em disco como

arquivo-texto ou binário para uso posterior (BELLIFEMINE; CAIRE; GREENWOOD, 2007).

Na próxima seção é enfatizado o JADE de acordo com o padrão FIPA.

3.5.2 JADE e FIPA

Segundo Braga (2006), o JADE é composto por diversos pacotes de classes que implementam as especificações FIPA. Além deles, é fornecido um pacote de ferramentas para facilitar a administração da plataforma e do desenvolvimento da aplicação.

Neste contexto, algumas dessas ferramentas são (BRAGA, 2006):

- Um agente gráfico de gerenciamento da plataforma;
- Um agente para monitoração que permite enviar e receber mensagens ACL;
- Um agente de escuta para interceptar as trocas de mensagens ACL da comunidade;
- Um agente para monitorar o ciclo de vida dos agentes;
- Uma interface visual para controlar o DF (ou seja, páginas amarelas).

Diante da conformidade com a FIPA, na figura 7 é mostrado o ciclo de vida de um agente JADE obedecendo a uma máquina de estados. (*Vide Figura 7*)

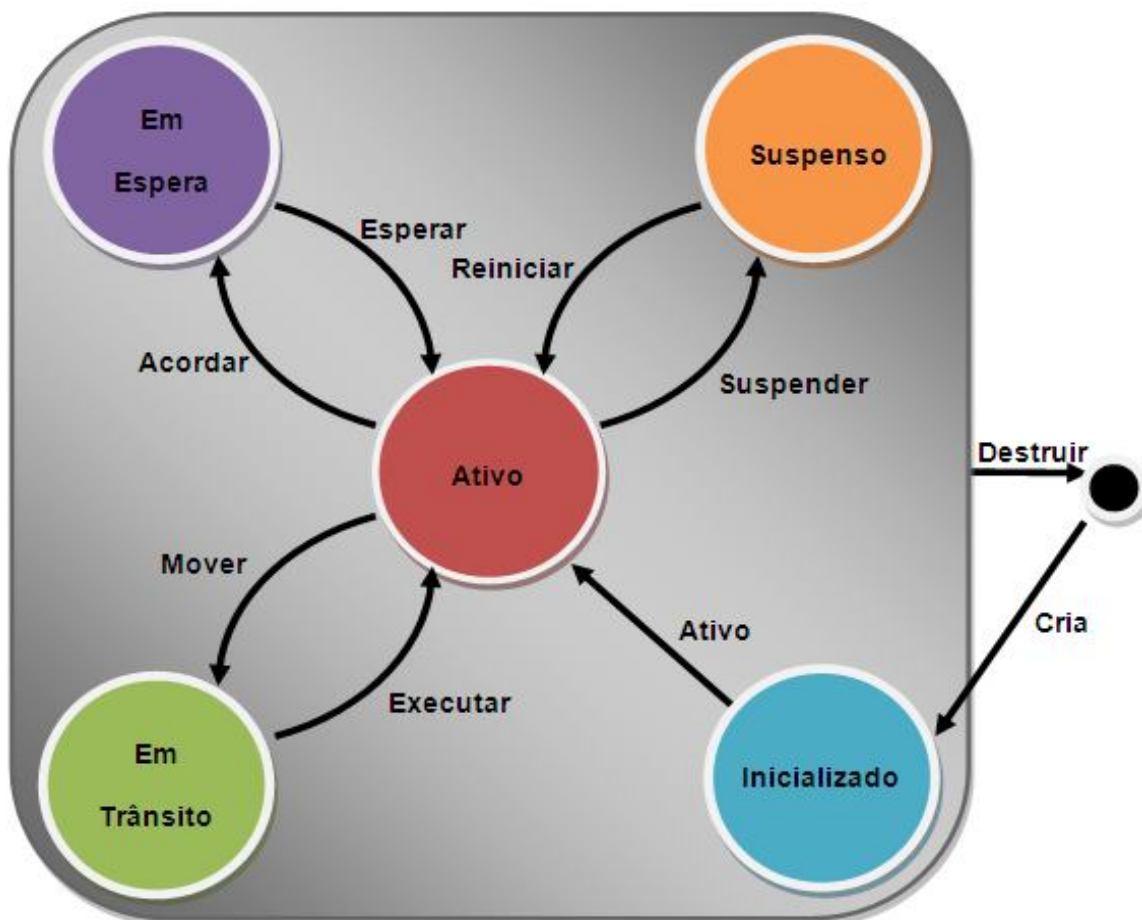


Figura 7 - Ciclo de Vida de um agente definido pela FIPA.

Fonte: Adaptado de (BRAGA, 2006).

Esses estados são representados em JADE como constantes estáticas da *classe agente*, são eles (SILVA, 2003):

- **Inicializado:** o objeto da *classe agente* foi instanciado, mas ainda não se registrou no AMS, não possui um identificador, ou seja, não tem nome, nem endereço, e não pode se comunicar com outros agentes.
- **Ativo:** o objeto da *classe agente* está registrado no AMS, possui um identificador, ou seja, tem nome formal e endereço, e acesso a todas funcionalidades do JADE, podendo executar seus serviços.
- **Suspenso:** o objeto da *classe agente* está no momento interrompido. Sua Thread interna está suspensa e nenhum comportamento está sendo executado.
- **Esperando:** o objeto da *classe agente* está bloqueado, esperando por algum evento. Sua Thread interna está “dormindo” sob um monitor

JAVA e irá acordar quando alguma condição ocorrer, ou seja, quando uma mensagem chegar.

- **Removido:** o objeto da *classe agente* está terminado, pois sua Thread interna acabou sua execução e o agente não está mais registrado no AMS.
- **Em Trânsito:** um agente móvel entra neste estado quando está migrando para um novo local. O sistema continua a armazenar as mensagens enviadas a este agente, que lhe serão passadas quando ele estiver no novo local.

3.6 Considerações Finais

Neste capítulo, foi realizado no primeiro momento, uma abordagem histórica sobre agentes de *Software* e Sistemas Multiagentes, assunto de grande importância no contexto desse trabalho. Em seguida foi exposto dois tipos de comunicação entre agentes, a forma direta e indireta, pois, a comunicação é fundamental para permitir que haja colaboração, negociação, cooperação entre entidades independentes. Também foram detalhadas as linguagens de comunicação entre agentes, KQM e FIPA-ACL, pois a forma de interação que ocorre entre os agentes é um fator muito importante na integração destes. E, finalmente, apresentou-se uma das ferramentas utilizada para o desenvolvimento de agentes e utilizada na presente pesquisa, o *Java Agent DEvelopment Framework* (JADE).

O próximo capítulo tratará sobre Ontologias e seus componentes básicos, além disso, também é mostrada algumas ferramentas para se construir Ontologias, como: *Ontolíngua*, *Java Ontology Editor* (JOE), *OntoEdit* e o *Protégé*.

4 ONTOLOGIA

Com o passar dos anos, o aumento crucial dos dados disponíveis tem conferido importância significativa às técnicas de organização da informação, com isso, essas técnicas fazem parte de estudos que buscam melhores tratamentos nesses dados. Uma possível solução para organização e manipulação de dados é a utilização das Ontologias, pois as mesmas oferecem um meio de lidar com a representação de recursos de informação. Finalmente, essas Ontologias estão sendo usadas em diversas áreas para organização dessas informações.

As Ontologias são utilizadas para o armazenamento do conhecimento diante de um domínio de interesse. Diante disso, Lindoso (2006) relata que num modo mais simples, uma Ontologia descreve uma hierarquia de conceitos relacionados entre si. Em casos mais complexos, acrescentam-se axiomas adequados, que são sentenças compostas por predicados unidos através de conectores lógicos, e que servem para expressar restrições a sua interpretação (SOUZA, 2003) (LINDOSO, 2006). O autor Lindoso (2006), ainda descreve que uma Ontologia é a representação do vocabulário de um domínio, mais precisamente, não é simplesmente o vocabulário como tal o que qualifica a Ontologia, mas os conceitos que os termos do vocabulário pretendem capturar.

Outra característica importante sobre Ontologia é ressaltada em Carneiro e Brito (2005), onde representa um modelo abstrato com o intuito de representar conceitos de um determinado domínio de forma clara e objetiva, o qual deve ser formal, compartilhável e composto por um conjunto de conceitos e regras bem formalizadas.

Os componentes básicos de uma Ontologia, que servem para representar conhecimento são (LINDOSO, 2006): conceitos, relações, instâncias e axiomas.

Os conceitos são as ideias básicas que se tentam formalizar, normalmente organizados em taxonomias, podendo ser classes de objetos, métodos, planos, estratégias, processos, entre outros.

Já as relações, representam relacionamentos semânticos entre os conceitos do domínio, sendo alguns exemplos típicos (é_uma, instância_de, subclasse_de e parte_de).

Outro componente básico de uma Ontologia é a instância, onde representa determinados objetos de um conceito, considerado como um tipo. E, por fim, os axiomas, como sendo as regras declaradas sobre relações que os elementos da Ontologia devem cumprir, podendo-se inferir através deles novos conhecimentos.

Mais formalmente, segundo Drumond (2009), uma Ontologia pode ser definida de acordo com a equação: $O = (C, H, I, R, P, A)$.

Portanto, o “C” representa o conjunto de entidades da Ontologia, ou seja, descreve as entidades do domínio sendo modelado. Já o “H” é o conjunto das relações taxonômicas entre os conceitos, onde tais relações definem a hierarquia de conceitos. O “I” representa o conjunto de relacionamentos entre classes e instâncias (relacionamento “é_um”) de uma Ontologia. Entretanto, o conjunto de relacionamentos que não são nem taxonômicos nem de instanciação entre classes e instâncias de uma Ontologia é representado por “R”. O “P” é o conjunto de propriedades das entidades de uma Ontologia, em virtude disso, tais propriedades relacionam conceitos a um tipo básico de dados (ex. inteiro, real ou string) ou podem relacionar instâncias a valores específicos dos tipos de dados. E, finalmente, o “A” representa um conjunto de axiomas, regras que permitem checar a consistência da Ontologia e deduzir novos conhecimentos através de algum mecanismo de inferência (DRUMOND, 2009).

Uma razão fundamental para a utilização de Ontologias no processo de desenvolvimento de Sistemas Multiagente é que elas são necessárias para viabilizar a comunicação entre agentes (LINDOSO, 2006). Como mostrado na seção anterior, os agentes se comunicam através de mensagens. E, a comunicação das mensagens no ambiente das Ontologias, contém expressões formuladas em termos de uma Ontologia.

4.1 Tipos de Ontologias

Segundo Guarino (1998), é possível desenvolver diferentes tipos de Ontologias de acordo com seu nível de generalidade, como mostrado na figura a seguir. (*Vide Figura 8*)

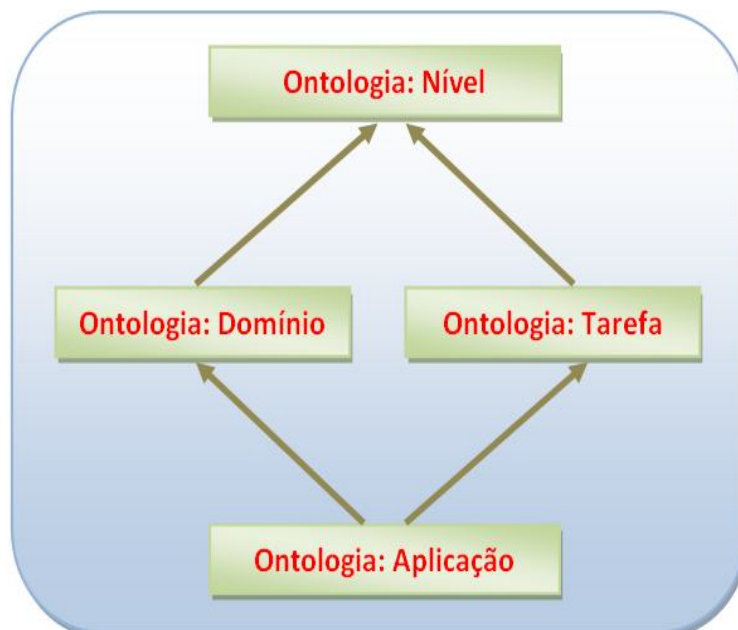


Figura 8 - Tipos de Ontologias de acordo com seu nível de dependência de uma determinada tarefa ou ponto de vista.

Fonte: Adaptado de (GUARINO, 1998).

Baseado no exposto na figura 8, é verificado que o *Nível Superior* das Ontologias descrevem conceitos muito gerais como espaço, tempo, matéria, objeto, evento, ação etc, que são independentes de um problema particular ou domínio (GUARINO, 1998).

Já no nível de *Domínio* e *Tarefa*, as Ontologias descrevem, respectivamente, o vocabulário relacionado a um domínio genérico (como a medicina, automóveis) ou uma tarefa genérica ou atividade (como diagnosticar, venda), especializando os termos introduzidos na Ontologia de nível superior (GUARINO, 1998).

Finalmente, no nível de *Aplicação* de Ontologias, descrevem conceitos em função tanto de um domínio particular, quanto no nível de tarefa, que muitas vezes são especializações de ambas as Ontologias relacionadas (GUARINO, 1998).

É importante considerar que, segundo consta nos relatos de Lindoso (2006), alguns princípios são de essencial observância durante a construção de uma Ontologia, são eles:

- Os termos devem ser acompanhados de definições objetivas;
- Coerência para permitir realizar inferências que sejam consistentes com as definições;

- Flexibilidade para a inclusão de novos termos sem revisão das definições existentes;
- Diversificação das hierarquias para aproveitar ao máximo os mecanismos de herança múltipla;
- Padronização dos nomes sempre que possível.

Com isso, Carneiro e Brito (2005) relata inúmeros benefícios quando se define um domínio de interesse com Ontologias, são eles: Compartilhamento do conhecimento; Aplicação de uma Ontologia genérica para um domínio de conhecimento específico; Compreensão semântica dos dados do domínio; e Evitar interpretações ambíguas.

4.2 Ferramentas para construir Ontologias

Dentre as ferramentas disponíveis para criação de Ontologias está a *Ontolíngua*, *Java Ontology Editor* (JOE), *OntoEdit*, *Protégé* (utilizada no presente trabalho), dentre outras.

A *Ontolíngua* (MCGUINNESS; FIKES; FEIGENBAUM, 2005) fornece um ambiente distribuído para pesquisar, criar, editar, modificar e uso de Ontologias. O servidor suporta mais de 150 usuários ativos, alguns dos quais nos fornecem descrições de seus projetos. Além disso, esta ferramenta possui um conjunto de serviços que possibilitam a construção de Ontologias compartilhadas entre grupos, permite acesso a uma biblioteca de Ontologias, tradutores para linguagens e um editor para criar e navegar pela Ontologia (FARQUHAR; FIKES; RICE, 1996).

O *Java Ontology Editor* (JOE) é uma ferramenta que fornece uma interface gráfica ao usuário para a criação ou edição de Ontologias, bem como formular consultas. Além disso, proporciona gerenciamento do conhecimento em ambientes abertos, heterogêneos e com diversos usuários. Portanto, o JOE é uma ferramenta escrita em JAVA, oferecendo vantagens na distribuição, segurança e portabilidade (MAHALINGAM; HUHNS, 1997).

Já o *OntoEdit* (MAEDCHE; VOLZ, 2002) é um ambiente gráfico para edição de Ontologias que permite inspeção, navegação codificação e alteração de Ontologias. O modelo conceitual é armazenado usando um modelo de Ontologia que

pode ser mapeado em diferentes linguagens de representação. As Ontologias são armazenadas em bancos relacionais e podem ser implementadas em XML, RDF e DAML+OIL

E, por fim, o *Protégé*, versão 3.3.1, um ambiente interativo para se criar projetos de Ontologias (PROTÉGÉ, 2010). Este *software* foi utilizado no presente trabalho, é de código aberto, oferece uma interface gráfica para edição de Ontologias e uma arquitetura para a criação de ferramentas baseadas em conhecimento. Segundo Noy and McGuinness (2010), a arquitetura do *Protégé* é modulada e permite a inserção de novos recursos. O *Protégé* é ainda, uma plataforma desenvolvida na linguagem de programação JAVA, criada pelo grupo de pesquisa *Stanford Medical Informatics* da escola de medicina da Universidade de Stanford (PROTÉGÉ, 2010).

4.3 Trabalhos Relacionados

Existem alguns trabalhos relacionados às ontologias, portanto, são propostos em (CARNEIRO; BRITO, 2005), (SANTOS; CAMPOS; VILLELA, 2003) e (PRADO, 2005), dentre outros.

Carneiro e Brito (2005) propõe à criação de uma ontologia em linguagem OWL, que permite a formalização de uma ontologia para representação de conteúdos educacionais generalistas, não direcionada a determinada área do conhecimento. Já o trabalho de Santos, Campos e Villela (2003) apresenta o estudo de uma ontologia para o domínio da educação mediada pela Web, apresentando uma proposta de taxonomia com os conceitos mais utilizados na área subdivididos em dois submodelos: categoria e funcionalidades. E, finalmente, Prado (2005) propõe a arquitetura de um sistema para reforço em EAD baseado em ontologias e agentes de *software*, contemplando o conteúdo das disciplinas e a forma com que o material de aprendizagem é armazenado.

4.4 Considerações Finais

Este capítulo teve como propósito a abordagem sobre Ontologias, uma descrição formal dos conceitos e relacionamentos que existem dentro de um domínio. Em seguida, foram expostos os componentes básicos de uma Ontologia, que servem para representar conhecimento, são eles: conceitos, relações, instâncias e axiomas. Apresentou-se algumas ferramentas para se construir Ontologias, como: *Ontolíngua*, *Java Ontology Editor (JOE)*, *OntoEdit* e o *Protégé*. Esta última foi utilizada na presente pesquisa para o desenvolvimento de uma Ontologia no domínio da matemática, grandezas e medidas (será abordada na seção 6.4 desta dissertação). E, por último, alguns trabalhos relacionados foram abordados.

O próximo capítulo trata sobre o *Software Virtual-TANEB*, enfatizando sua arquitetura e visão geral do *Software Virtual-TANEB*, assim como algumas interfaces do Ambiente do *Software VT*.

5 SOFTWARE VIRTUAL-TANEB

O Virtual-TANEB (VT) é um *Software* com a finalidade de obter informações sobre o desempenho de alunos do quinto ano do ensino fundamental, na resolução de questões de matemática, do Ensino Fundamental. Esse *Software* tem como base a Teoria da Resposta ao Item (TRI), onde sua característica principal é analisar como elemento central as questões que compõem uma prova, pois é um instrumento poderoso nos processos quantitativos de avaliação educacional, por permitir a construção de escalas de habilidades (VERAS *et al.*, 2009).

Para Andrade e Justino (2007 apud VERAS *et al.*, 2009), a TRI assume a existência de um traço latente, isto é, uma habilidade que não pode ser diretamente observável, sendo difícil de mensurá-la. Entretanto, a habilidade pode ser estimada a partir de um modelo matemático que leva em consideração características das questões e a probabilidade do indivíduo acertar uma determinada questão dado o seu grau de habilidade.

É importante ressaltar também que, o *Software* Virtual-TANEB permitiu um controle de dados cadastrais tanto no âmbito administrativo, com informações de professores, escolas, pedagogos, entre outros, como no âmbito dos respectivos itens armazenando-os em uma base de itens. Tendo em vista que, os resultados obtidos pelo *Software*, fornecem apoio a professores e pedagogos na gestão dos resultados educacionais (VERAS *et al.*, 2009).

5.1 Arquitetura geral do Software Virtual-TANEB

A arquitetura do *Software* Virtual-TANEB é constituída por uma Interface Gráfica com o Usuário (GUI) e o Núcleo do *Software*, onde pode ser visto com detalhes na figura a seguir. (*Vide Figura 9*)

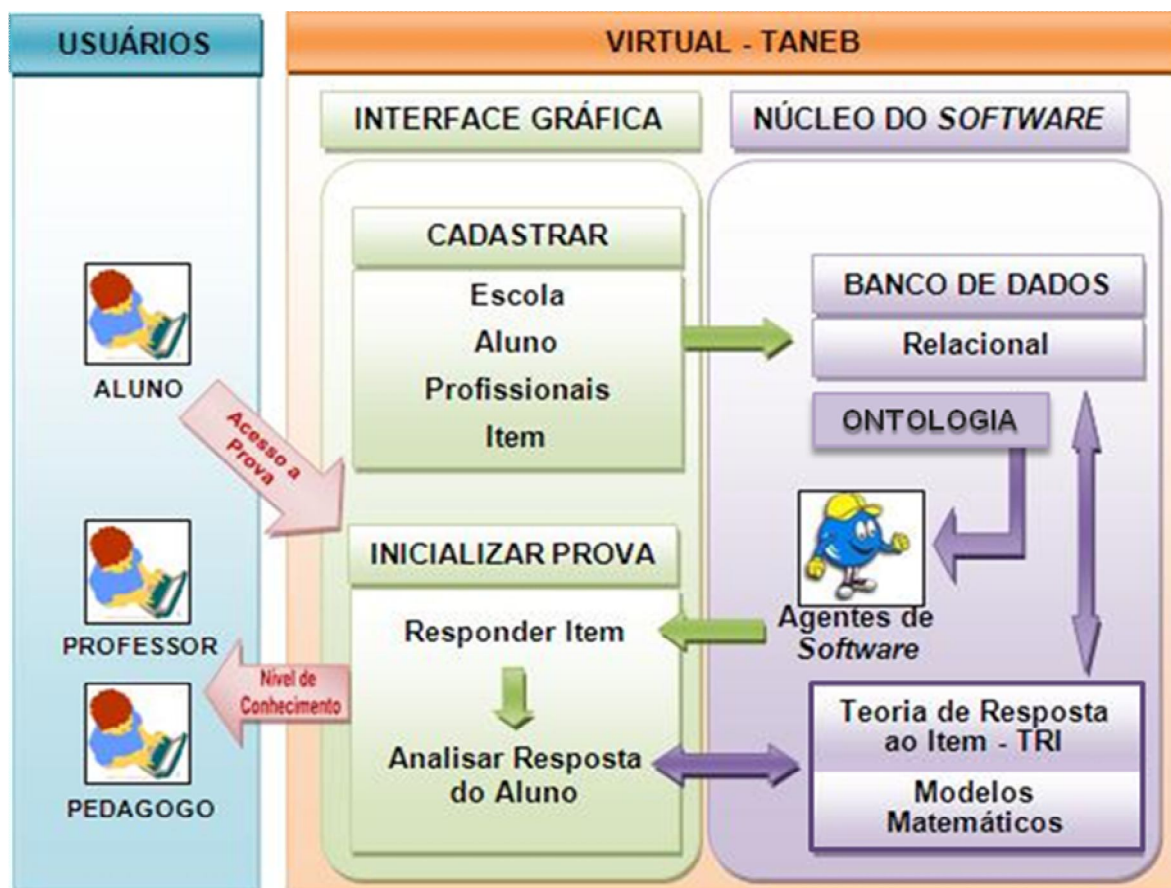


Figura 9 - Arquitetura do Software Virtual-TANEB.

Fonte: Adaptado de (VERAS et al., 2009).

Conforme mostra a figura 9, a Interface com o Usuário irá auxiliar o estudante na realização de sua prova e, posteriormente, permitindo que professores e pedagogos possam avaliar o nível de conhecimento destes alunos sobre uma determinada área. Com isso, esta interação possibilita que através dessas respostas, possa-se fazer uma análise, e através do resultado da análise inferir na qualidade do Ensino Fundamental na área de matemática em uma determinada escola. Em virtude disso, a GUI é responsável por permitir o contato do aluno com as provas formuladas a partir da seleção das questões armazenadas em um banco de dados através de agente de *Software* (VERAS et al., 2009).

Já o Núcleo do *Software* é composto por agentes de *Software*, uma base de dados e da Teoria da Resposta ao Item. Diante disso, os agentes de *Software*, ficam responsáveis pela escolha dos itens armazenados em um banco de dados para a elaboração de uma prova, pelo monitoramento em relação à resolução dos itens, como por exemplo, o tempo gasto para responder cada item e, por último, terá

um agente responsável por disponibilizar informações (dicas) sobre um determinado assunto, abordado no item (VERAS *et al.*, 2009).

Segundo Veras *et al.*(2009), no Virtual-TANEB, já estão definidos os agentes participantes do sistema (seletor, monitor, dica) e seus respectivos papéis. Portanto, a figura 10, a seguir, exhibe uma visão geral, dividida em módulos do funcionamento do *Software* VT, analisando a estrutura organizacional em que os agentes de *Software* estão inseridos. (*Vide Figura 10*)

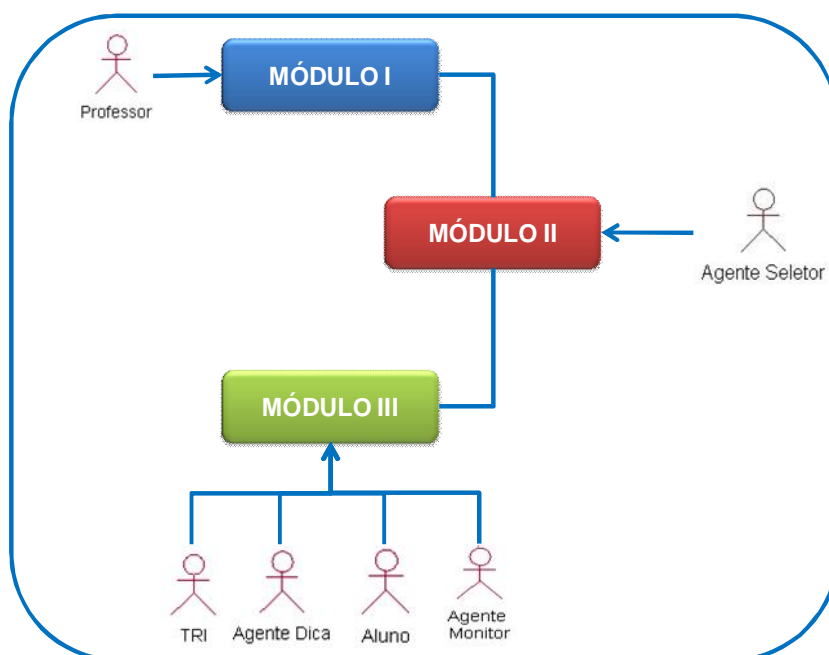


Figura 10 - Visão geral do *Software* Virtual-TANEB.

Fonte: (VERAS *et al.*, 2009).

Diante do exposto na figura 10, o Módulo I é a parte do sistema que fica direcionado ao cadastramento dos dados pessoais dos profissionais envolvidos neste *Software* (professores, pedagogos, alunos, gestores e funcionários administrativos) envolvidos na formulação e resolução dos itens, também cadastrados nesta seção (VERAS *et al.*, 2009).

Já o Módulo II, é parte responsável pela elaboração da prova, aonde temos uma interação entre o agente Seletor e um repositório de dados no processo de selecionar itens para a formulação da prova a ser realizada pelos alunos (VERAS *et al.*, 2009).

E, por fim, o Módulo III, possui o agente Monitor (monitora o desempenho do aluno, desde o tempo gasto para responder cada item, até a verificação do nível

de desempenho por itens acertados) e o agente DICA, responsável por disponibilizar informações que possam auxiliar o discente na realização da prova (VERAS *et al.*, 2009).

Veras *et al.*(2009) relata ainda que, o *Software* Virtual-TANEB oferece um suporte amigável e interativo para que alunos, professores, pedagogos e gestores possam usufruir do *Software*, visando identificar inicialmente os problemas existentes com o ensino da Matemática, ou seja, apontar falhas e soluções a serem implantadas relacionadas a este Ensino.

5.2 Ambiente do Software Virtual-TANEB

O Ambiente do *Software* Virtual-TANEB foi desenvolvido utilizando tecnologias voltadas para WEB, oferecendo um suporte amigável e interativo para que alunos, professores, pedagogos e gestores possam usufruir do *Software*, visando identificar inicialmente os problemas existentes com o ensino da matemática.

Utilizou-se o *Java Server Pages* (JSP), *HyperText Markup Language* (HTML), a *JavaScript*, linguagem baseada em scripts com sintaxe semelhante à do JAVA, dentre outros recursos. Além destas linguagens utilizadas no desenvolvimento do *Software* VT, também foi necessário o uso de um Ambiente Integrado de Desenvolvimento (IDE), e diante de várias IDE's disponíveis para desenvolvimento na WEB, optou-se pelo uso do NetBeans IDE 6.8 (GONÇALVES, 2008), que é caracterizado por ser gratuito e de código aberto para desenvolvedores na linguagem JAVA, dentre outras linguagens.

Dentre as etapas para navegar no *Software* Virtual-TANEB, a autenticação é necessária para acesso ao sistema. Após o usuário devidamente cadastrado no VT, executa-se o *login*, informando nome de usuário com sua respectiva senha, por meio da Interface de *login* do Virtual-TANEB. Na figura 11 é possível observar com detalhes, este processo de autenticação. (*Vide Figura 11*)



The screenshot shows the login page of the Virtual-TANEB software. At the top, there is a blue header with the Virtual-TANEB logo on the left, the text "Virtual-TANEB" in the center, and the UFMA logo on the right. Below the header, the main content area is light green. It features a login form with two input fields: "Login:" containing the text "Flavio" and "Senha:" containing five dots. To the right of the password field is a yellow padlock icon with a key. Below the fields is a yellow "Entrar" button. Further down, there is a green link that says ":: Esqueceu Senha !!" with a blue question mark icon below it. At the bottom of the page, a blue footer contains the text "Copyright © 2009 Virtual-TANEB v.1.0".

Figura 11 - Interface de autenticação no Software Virtual-TANEB.

O Software Virtual-TANEB possui um procedimento que permite ao usuário recuperar sua senha, caso ele a esqueça. Na figura 12, a seguir é mostrada através de uma interface, a funcionalidade de recuperação da senha e *login* do usuário do VT, onde o usuário receberá informações sobre seu cadastro, via email.



The screenshot shows the password recovery page of the Virtual-TANEB software. It has the same blue header as Figure 11, with the Virtual-TANEB and UFMA logos. The main content area is light green. It features a form with an "Email:" label and a text input field containing "flathbarros@gmail.com". To the right of the input field is a blue question mark icon. Below the input field is a yellow "Reenviar Senha" button. At the bottom of the page, a blue footer contains the text "Copyright © 2009 Virtual-TANEB v.1.0".

Figura 12 - Interface para recuperar senha no Software Virtual-TANEB.

Já na figura 13, a seguir, é demonstrada a página principal do Software VT. (Vide Figura 13)



Figura 13 - Interface da página principal do usuário “Administrador do Sistema” no Software Virtual-TANEB.

A figura acima, ilustra a interface principal do VT, na ocorrência de prévia autenticação do usuário, utilizando *login* e senha. É observado um “painel de controle”, onde o usuário autenticado como “Administrador do Sistema” tem total permissão para manipular todos os recursos oferecidos pelo Virtual-TANEB, como: cadastrar usuários, visualizar usuários, cadastrar itens, liberar prova, dentre outras funções.

A figura, a seguir, é mostrado a interface de cadastro de usuários no Software Virtual-TANEB, são eles: Administrador do Sistema, Gestor, Técnico Administrativo, Professor, Pedagogo e Aluno. (Vide Figura 14)

Copyright © 2009 Virtual-TANEB v.1.0

Figura 14 - Interface de cadastro de usuários no Software Virtual-TANEB.

Outra atividade possível no *Software* Virtual-TANEB é o procedimento para cadastrar os itens, seguindo os parâmetros estabelecidos pelo SAEB (BRASIL, 2009), que serão aproveitados na elaboração da prova de matemática, a ser respondida pelos alunos do quinto ano do Ensino Fundamental. Diante disso, os itens antes do cadastro passarão por uma análise e, em seguida, é possível realizar correções, caso haja problemas. Esta verificação é feita por meio de observações dadas pelos pedagogos baseados nos Descritores estabelecidos pelo SAEB (BRASIL, 2009) para a Matemática do quinto ano do Ensino Fundamental.

Após o cadastro dos itens, faz-se necessário uma interface de visualização dos mesmos. Na figura 15 é possível observar este processo. (*Vide Figura 15*)

OPERAÇÃO	NIVEL	DESCRITOR	PERGUNTA	IMAGEM PERGUNTA	ALTERNATIVAS	ALTERNATIVA CORRETA
Alterar Excluir	Básico	D6	O carro de João consome 1 litro de gasolina a cada 10 quilômetros percorridos. Para ir da sua casa ao sítio, que fica distante 63 quilômetros, o carro consome?		1) um pouco menos de 6 litros de gasolina Imagem: 2) exatamente 6 litros de gasolina Imagem: 3) um pouco mais de 6 litros de gasolina Imagem: 4) exatamente 7 litros de gasolina Imagem:	3
Alterar Excluir	Básico	D7	Ao usar uma régua de 20 cm para medir uma mesa, Henrique observou que ela cabia 27 vezes no comprimento da mesa. Ele multiplicou esses valores e encontrou 540 cm. Em metros, o comprimento da mesa é de?		1) 0,54 m Imagem: 2) 5,4 m Imagem: 3) 54 m Imagem: 4) 540 m Imagem:	2
Alterar Excluir	Básico	D8	Faltam 31 dias para o aniversário de João. Quantas semanas completas faltam para o aniversário dele?		1) 3 Imagem: 2) 4 Imagem: 3) 5 Imagem: 4) 6 Imagem:	2

Copyright © 2009 Virtual-TANEB v.1.0

Figura 15 - Interface de visualização de itens cadastrados no Software Virtual-TANEB.

A figura exibida acima, traz a Interface que mostra todos os itens efetivamente cadastrados pelo “Professor” seguindo as observações efetuadas pelos pedagogos conforme os Descritores estabelecidos através do Sistema Nacional de Avaliação da Educação Básica (SAEB). Estes itens serão utilizados na formulação das provas realizadas pelos “Alunos”.

Portanto, na etapa da realização da prova, alguns procedimentos são usados (VERAS, 2010): 1) o “Gestor Escolar” solicita ao “Administrador do Sistema” um agendamento determinando o dia e a hora que será realizada a prova; 2) o “Administrador” faz este agendamento no *Software Virtual-TANEB*; 3) no dia e na hora estabelecida o *Software* dispara o agente Seletor, ficando responsável pela escolha dos itens que farão parte da prova a ser respondida naquele momento; e 4) o “Aluno” responde a prova, cuja avaliação foi formada a partir da seleção de itens por meio de agente de *Software*

5.3 Considerações Finais

Neste capítulo foi descrita a arquitetura do *Software Virtual-TANEB* com o objetivo de identificar algumas funcionalidades deste sistema, bem como as contribuições referentes à arquitetura geral do sistema. E, por fim, foram expostas algumas Interfaces do Ambiente do *Software Virtual-TANEB*, assim como: a Interface de Autenticação; a Interface para recuperar senha; a Interface da página principal do usuário “Administrador do Sistema”; a Interface de cadastro de usuários; e a Interface de visualização de itens cadastrados no *Software Virtual-TANEB*.

No próximo capítulo apresenta-se o modelo proposto do Ambiente de Avaliação do *Software Virtual-TANEB*, que se caracteriza como uma das principais contribuições deste trabalho.

6 MODELAGEM DO AMBIENTE DE AVALIAÇÃO DO SOFTWARE VIRTUAL-TANEB

6.1 Modelo Proposto

No modelo proposto para o Ambiente de Avaliação do *Software* Virtual-TANEB, o aluno terá acesso ao módulo de avaliação do VT, após estar devidamente cadastrado no sistema. Uma vez autorizado a realizar a prova, o aluno inicializará o teste através da sua conta de usuário. É também possível observar três Atores principais: Aluno, Professor e o Pedagogo. Portanto, a figura, a seguir, mostra com detalhes o modelo proposto, onde irá ser integrado no *Software* Virtual-TANEB. (Vide Figura 16)

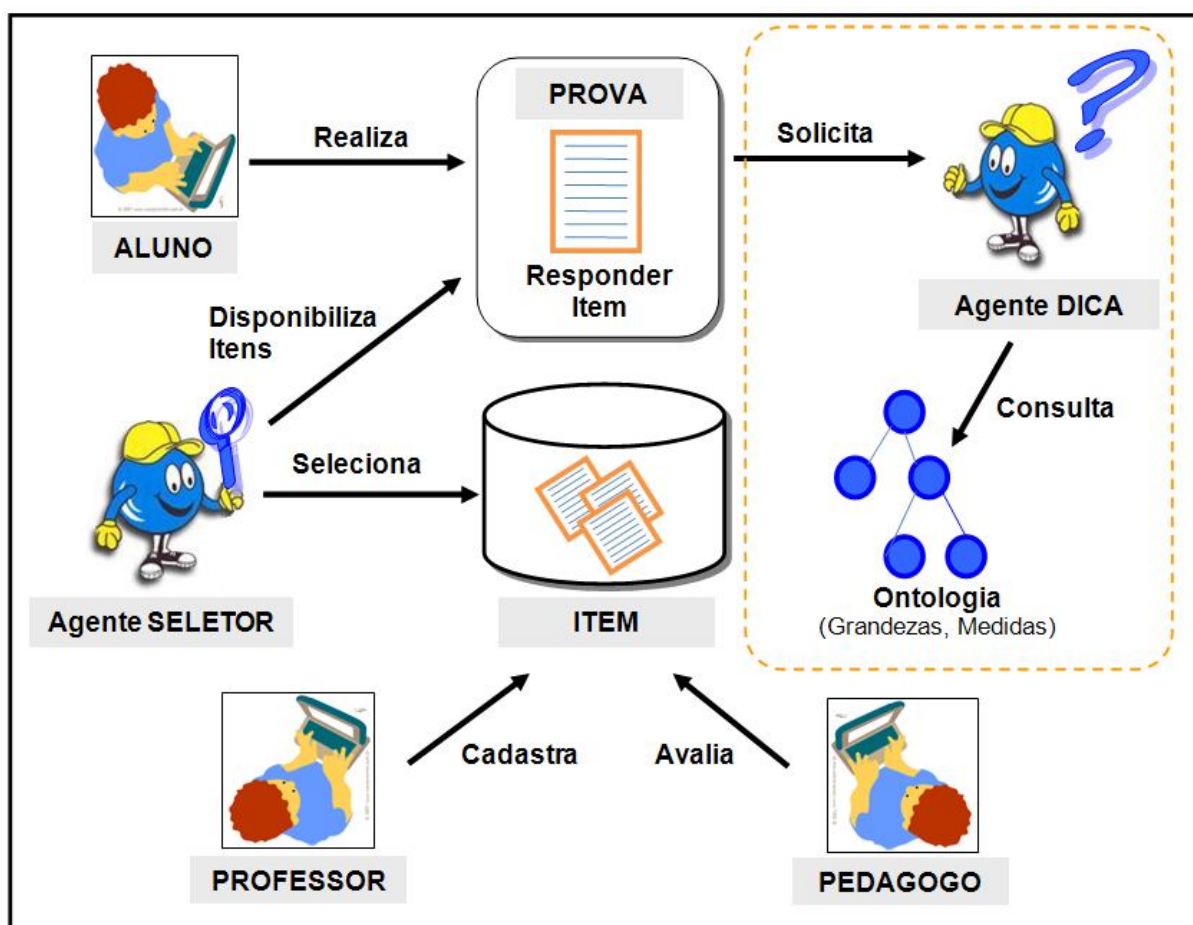


Figura 16 - Estrutura do modelo proposto.

Diante do exposto na figura 16, o Ator “Aluno” realizará uma prova, onde a mesma contém os referidos Itens de acordo com o nível de cada aluno. O agente Seletor é responsável por organizar esses itens e seleciona de forma aleatória, portanto, o agente busca os itens no banco de dados, disponibilizando para o “Aluno”. Ao responder os itens, o aluno poderá solicitar do agente DICA uma consulta em uma Ontologia. A Ontologia disponibiliza informações que possam auxiliar o discente na realização da prova. O agente e a Ontologia aqui utilizados, são focos de pesquisa do referido trabalho.

Já o professor no modelo proposto, na figura 16, fica responsável por formular os itens e cadastrá-los em um banco de dados. Ao serem armazenadas esses itens, não estarão habilitados. Portanto, o agente seletor só seleciona os itens habilitados pelo pedagogo, formulando assim a prova, e disponibilizando para o discente.

E, por fim, no modelo o pedagogo avalia os itens recém cadastrados não habilitados, e caso precise reformular, ele retorna (reenvia o item) para o professor, descrevendo através de comentários, e assim, informando ao professor as devidas alterações. Com isso, o professor, após realizar as alterações solicitadas pelo pedagogo, reenvia o referido item ao pedagogo para ser habilitado. O pedagogo, após uma avaliação final do item, pode ou não habilitá-lo, para ser utilizado pelo agente seletor, fazendo parte da formulação das provas.

6.2 Diagrama de Caso de Uso do Ambiente de Avaliação do Virtual-TANEB

Foi utilizado no presente trabalho o Processo Unificado (BOOCH *et al.*, 2000) com a notação UML (Guedes 2008) para descrever as tarefas e interações dos agentes envolvidos no processo de elaboração e resolução dos itens da prova.

Diante da Visão Geral do Virtual-TANEB apresentado na figura 10 (no presente trabalho), identificou-se os Casos de Uso do Módulo III, realizar prova demonstrada na figura 17. Neste módulo, o aluno será analisado com o objetivo de identificar as suas aptidões (habilidades) e deficiências sobre o tema abordado nos itens a serem respondidos. (*Vide Figura 17*)

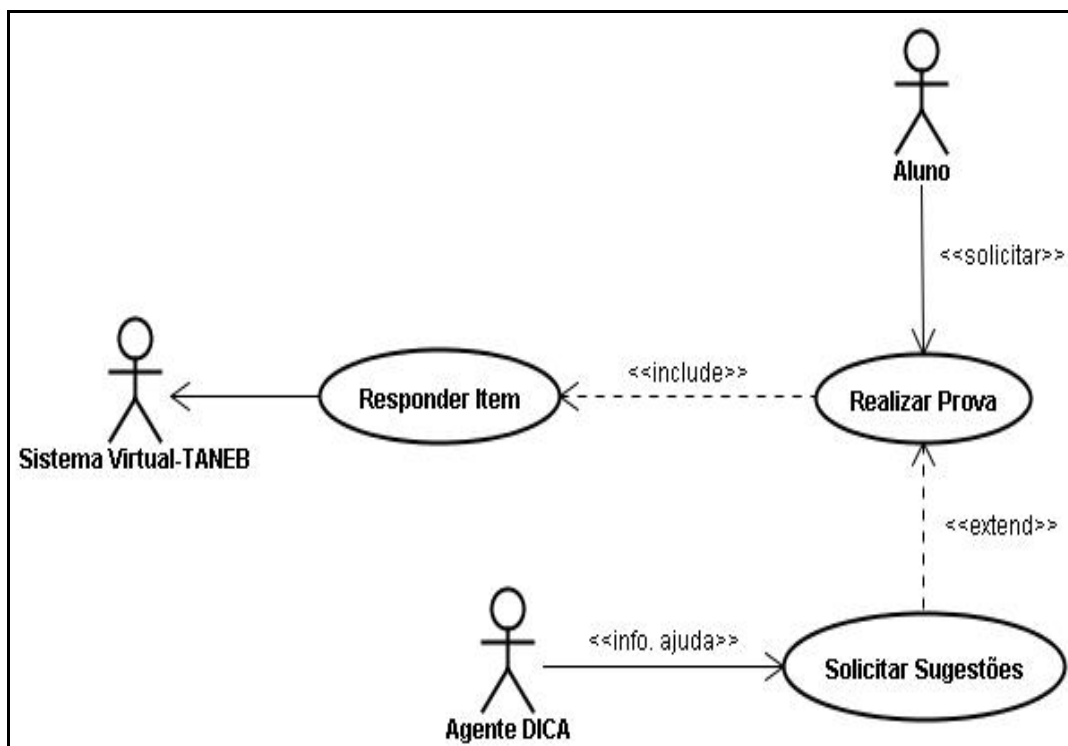


Figura 17 - Diagrama de Caso de Uso “Realizar Prova”.

Na Tabela 3, a seguir, estão expostos os atores e casos de usos que fazem parte do diagrama de Caso de Uso “Realizar Prova”, da figura 17.

Tabela 3 - Documentação do Caso de Uso realizar prova.

Nome do Caso de Uso	Realizar Prova
Ator Principal	Aluno
Atores Secundários	Agente DICA, <i>Software</i> Virtual-TANEB
Resumo	Este Caso de Uso informa como será executado o processo de realização da prova pelo aluno
Pré-Condições	A prova precisa estar disponível para o aluno, e o itens formulados pelo agente seletor
Ações do Ator	Ações do Sistema
1) Solicitar a realização da prova	
	2) Aplicar a prova
3) Dar auxílio aos alunos com sugestões sobre os itens	
	4) Gerar relatório sobre o desempenho do aluno
Restrições / Validações	1) Para responder a prova, faz-se necessário que o aluno, a escola e a disciplina estejam cadastrados no sistema.

Na próxima seção será apresentada uma Modelagem do agente DICA, utilizando uma metodologia denominada TROPOS.

6.3 Modelagem do agente DICA

Para a especificação do agente DICA foi utilizado à metodologia TROPOS, pois nos permite definir os agentes incluindo suas metas, tarefas e recursos.

Giunchiglia *et al.*(2002) define a TROPOS, como uma metodologia de desenvolvimento de *Software* orientado a agentes que se baseiam em características chaves: as noções de agente, objetivo, plano, bem como vários outros conceitos em nível de conhecimento. Tais são primitivas fundamentais e uniformemente usadas durante todo o processo, além disso, um papel crucial é atribuído à análise e especificação de requisitos quando o futuro sistema é analisado em relação ao seu ambiente alvo.

A figura 18 mostra os modelos abstraindo os conceitos e relacionamentos da metodologia TROPOS (PERINI; SUSI, 2004) (SILVA, 2005), são eles: Objetivos, Planos, Recursos, Dependências, Capacidades e Crenças.

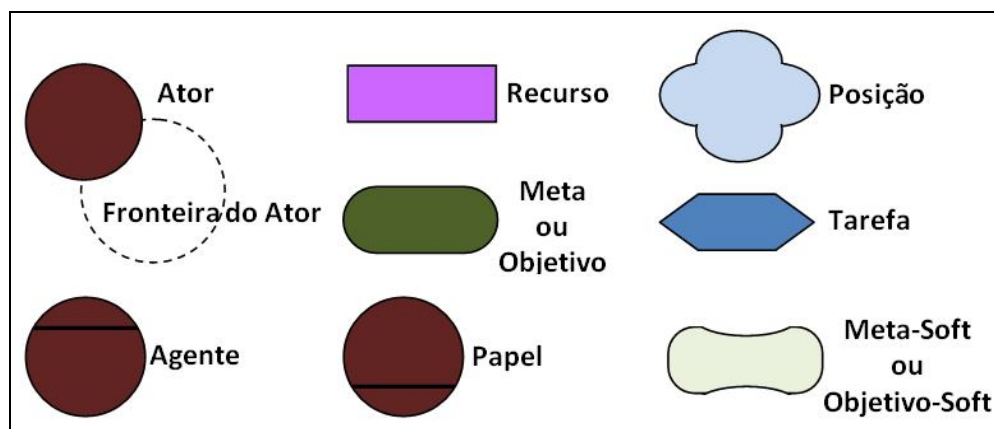


Figura 18 - Visualização de modelos abstraído conceitos e relacionamentos na metodologia TROPOS.

Fonte: Adaptado de (SILVA, 2005).

A TROPOS possui algumas ferramentas para realizar a modelagem segundo suas especificações. Uma delas, e utilizada no presente trabalho, é a *Tool for Agent Oriented Modeling* (TAOM4E) (PERINI, 2010).

Na figura 19, a seguir, é mostrado um modelo de raciocínio estratégico para o agente DICA, pois tem como *Meta*, a de informar uma sugestão de um item, e também deve satisfazer uma *Meta-Soft* de “Auxiliar o aluno na resolução da prova”. Para realizar isto, ele executa a tarefa “Gerar Dica” que precisará realizar uma Subtarefa “Consultar a ONTOGRAM”. (Vide Figura 19)

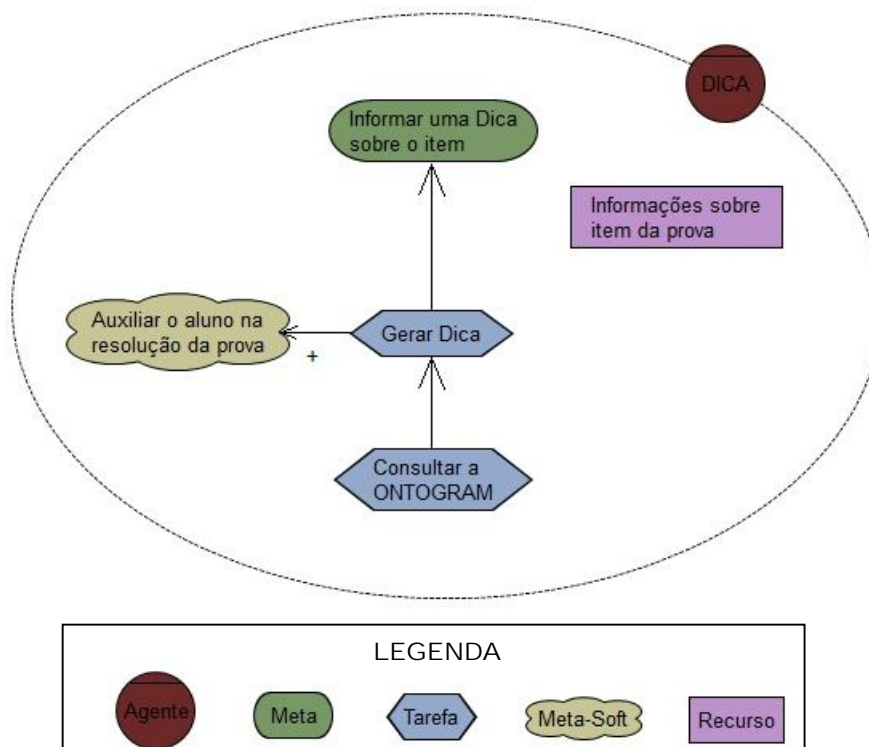


Figura 19 - Modelo de Raciocínio Estratégico do agente DICA.

Diante do exposto na figura acima, é necessário fazer uso do recurso “Informações sobre o item da prova”, para realização das tarefas que envolvem o agente DICA.

6.4 ONTOGRAM

Para o desenvolvimento da Ontologia ONTOGRAM, foi considerada a Matriz de Referência de Matemática do Sistema de Avaliação da Educação Básica (SAEB) (BRASIL, 2009) com os seus respectivos Descritores relativos ao conteúdo programático do quinto ano do Ensino Fundamental, referentes ao tema grandezas e

medidas. A seleção dos conceitos da ONTOGRAM fez parte de uma pesquisa aos materiais didáticos e consultas a especialistas, professores e pedagogos do quinto ano.

A partir do conhecimento adquirido em relação ao domínio da Ontologia, utilizou-se a metodologia 101 (NOY; MCGUINNESS, 2010) para se criar as classes no *Software Protégé*. Estas classes são demonstradas em uma hierarquia, apresentadas na “Guia Classes do *Protégé*”, do lado esquerdo da ferramenta, conforme figura a seguir. (*Vide Figura 20*)

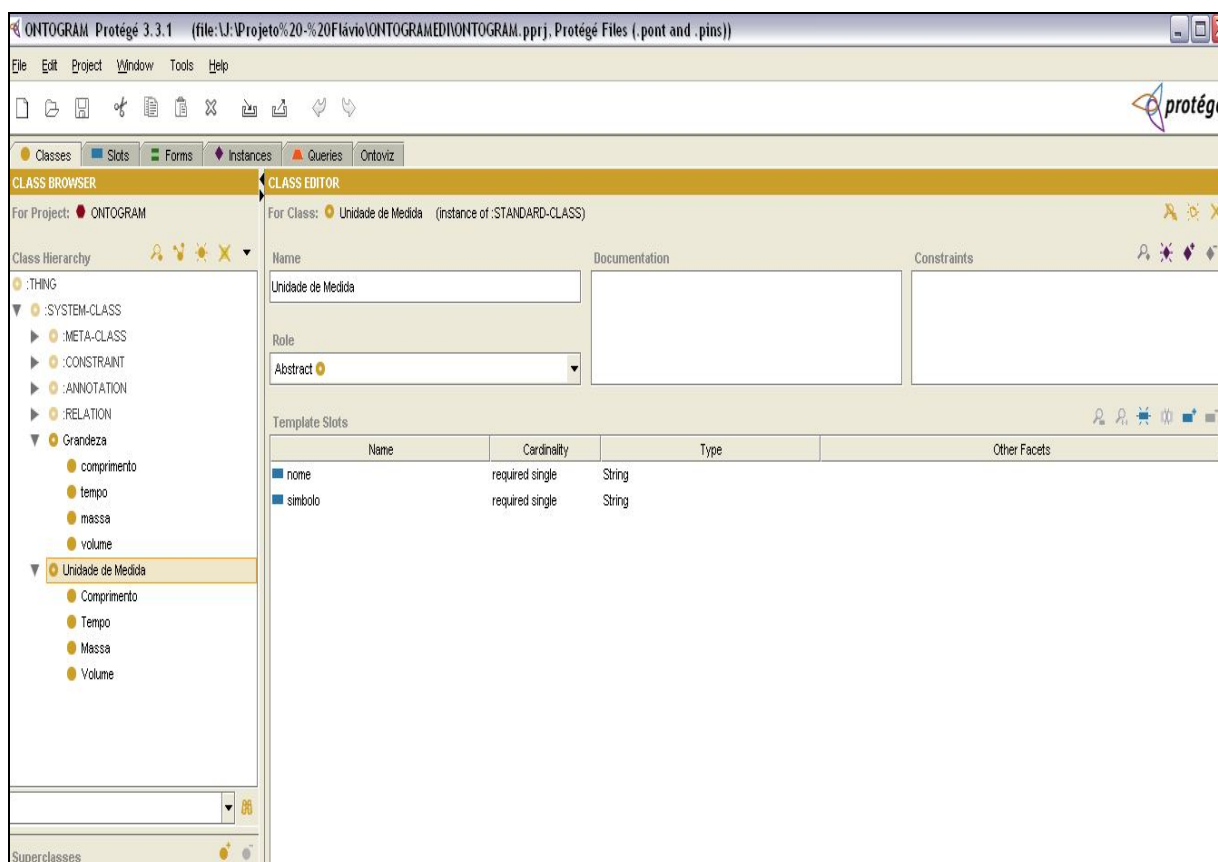


Figura 20 - Hierarquia de Classes da Ontologia ONTOGRAM.

Na figura 20, para cada conceito presente na Ontologia foram identificados *slots* que representam as suas propriedades. Por exemplo, na classe Unidade de Medida foram identificados os *slots* “nome” e “símbolo”.

Já na figura 21 é ilustrada parte da rede semântica com os conceitos da ONTOGRAM. A partir dos conceitos relacionados ao domínio foram derivados os axiomas, ou seja, regras declaradas sobre relações que os elementos da Ontologia

devem cumprir. Na figura também é possível observar as instâncias das subclasses “volume” e “massa”. (Vide Figura 21)

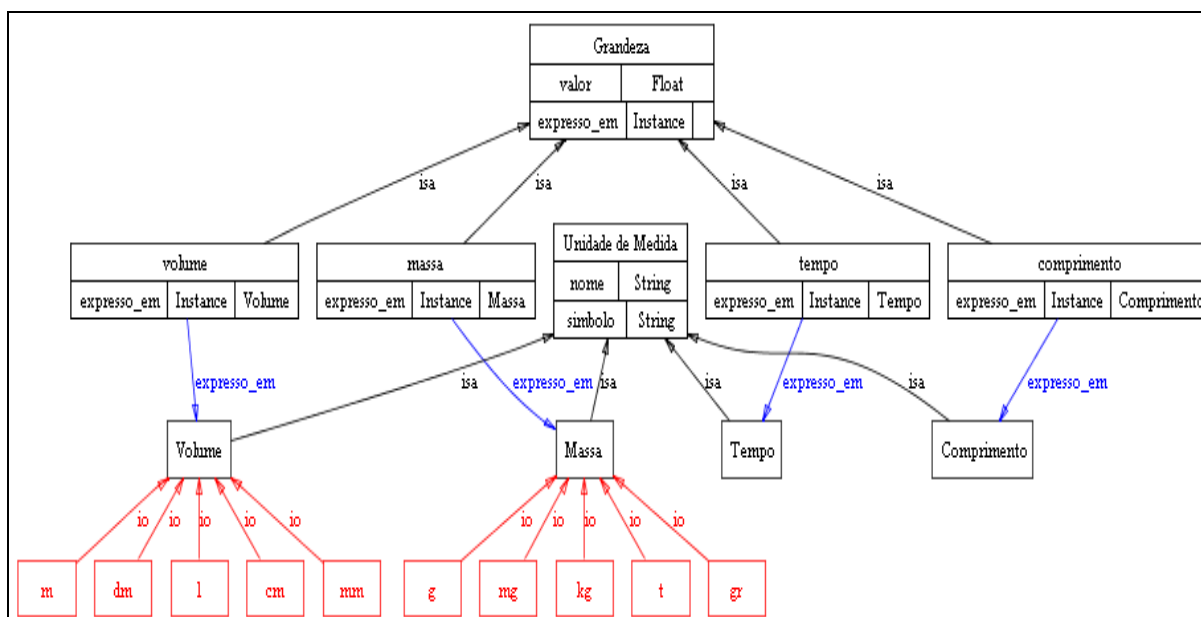


Figura 21 - Rede semântica com Instâncias da subclasse “volume e massa” da Ontologia ONTOGRAM.

A figura 21 demonstra as subclasses “massa”, “tempo”, “comprimento” e “volume”. Todas as subclasses das classes “Grandeza” e “Unidades de Medida” foram instanciadas conforme previsto no domínio da Matemática, grandezas e medidas. Já na figura 22, a seguir, destaca as instâncias das subclasses “tempo” e “comprimento”, relacionada à classe Unidade de Medida.

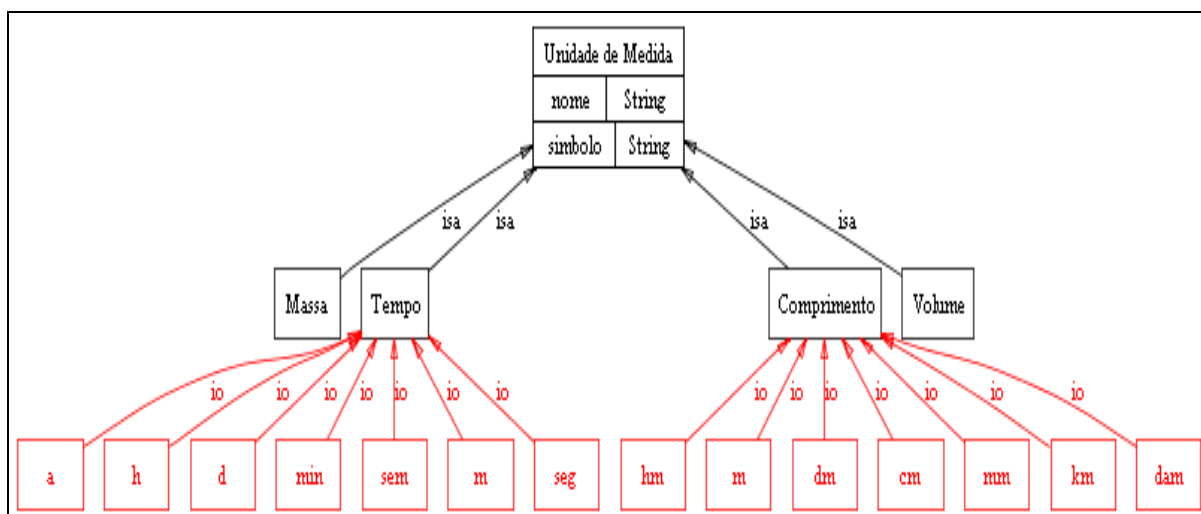


Figura 22 - Rede semântica com Instâncias da subclasse “tempo e comprimento” da Ontologia ONTOGRAM.

A figura 23 demonstra parte da Ontologia ONTOGRAM, classe “Grandeza”, com as quatro subclasses, “volume”, “tempo”, “massa” e “comprimento” devidamente populada.

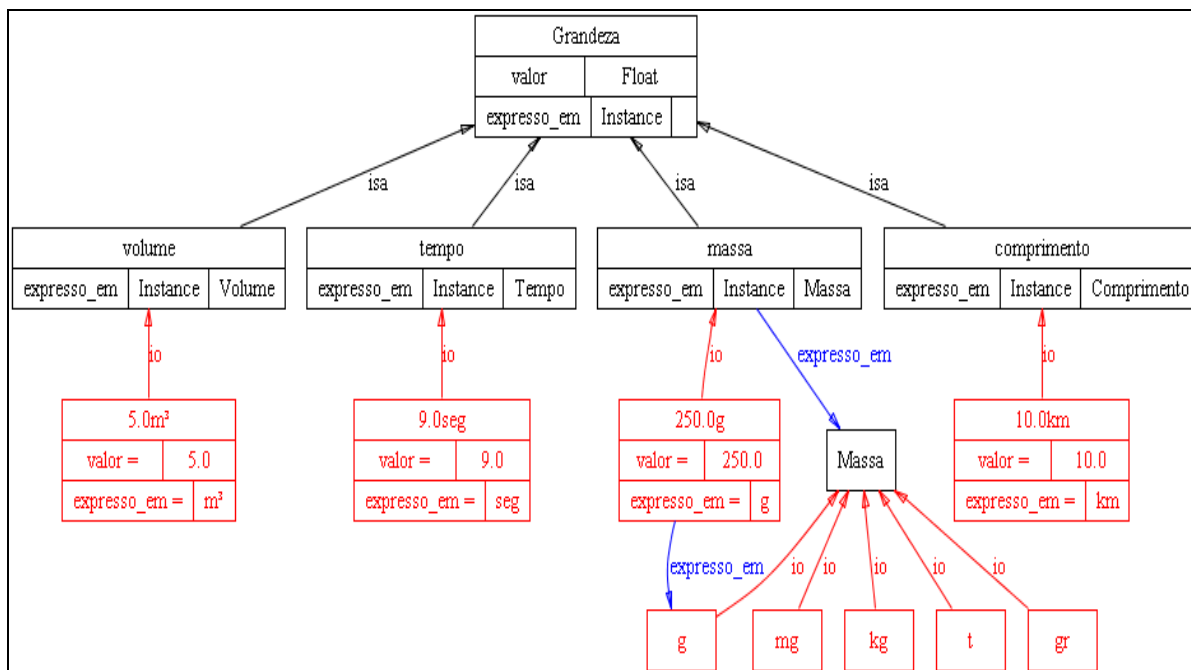


Figura 23 - Rede semântica com população das subclasses (volume, tempo, massa e comprimento) da Ontologia ONTOGRAM.

6.5 Modelagem do Banco de Dados

Para a especificação do Banco de Dados (BD) do Ambiente de Avaliação do *Software* Virtual-TANEB, utilizou-se uma ferramenta denominada MySQL Workbench – Visual Database Design For Professionals, versão 5.1.16 OSS. O MySQL Workbench é uma multiplataforma, ferramenta de design visual de dados e disponível como uma ferramenta GUI nativa em Windows, Linux e Mac OS X em diferentes edições (ZINNER, 2008).

Na figura, a seguir, é demonstrada parte da modelagem do BD do Ambiente de Avaliação do VT. (*Vide Figura 24*)

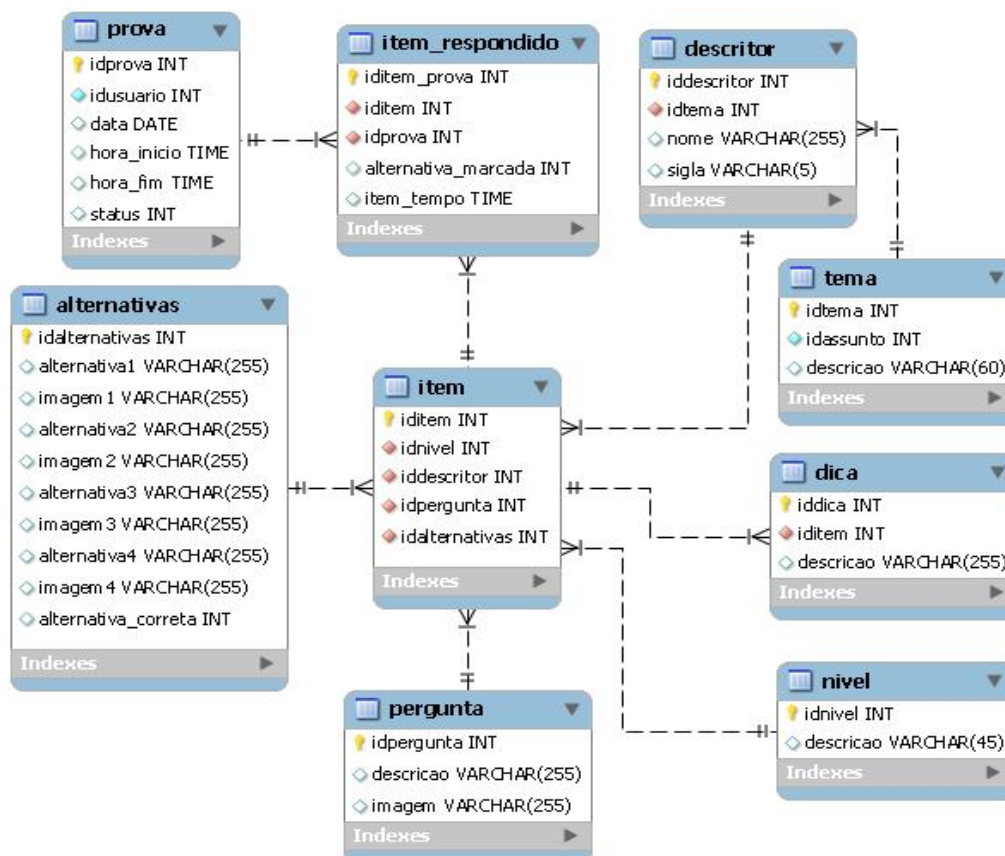


Figura 24 - Parte da modelagem do BD do Ambiente de Avaliação do VT.

A figura 24 mostra parte da modelagem utilizada no referido trabalho. Portanto, pode ser visualizada por completo no Apêndice desta dissertação.

6.6 Considerações Finais

Neste capítulo, apresentou-se o modelo proposto do Ambiente de Avaliação do *Software* Virtual-TANEB, que se caracteriza como uma das principais contribuições científica resultante deste trabalho, além disso, apresentou-se: Diagrama de Caso de Uso do Ambiente de Avaliação do VT; Modelagem do agente DICA, utilizando a TROPOS; Estrutura da Ontologia ONTOGRAM, baseada no domínio da matemática do quinto ano do Ensino Fundamental, referente ao tema grandezas e medidas; e, por último, é exposto parte da modelagem do banco de dados utilizado no protótipo do Ambiente de Avaliação do *Software* VT.

O próximo capítulo mostra como foi implementado o Ambiente de Avaliação do *Software* Virtual-TANEB, mostrando algumas interfaces do *software*.

7 IMPLEMENTAÇÃO E TESTE DO AMBIENTE DE AVALIAÇÃO DO SOFTWARE VIRTUAL-TANEB

7.1 Ambiente de Avaliação

O Ambiente de Avaliação do *Software* Virtual-TANEB está desenvolvido numa versão 2.0/2010. Este protótipo propõe a utilização de um ambiente gráfico para a realização das provas de forma amigável, cujo foco se concentra na interatividade do aluno com a ferramenta, tendo em consideração diversos elementos, tais como interesses e curiosidades para a execução dos testes no VT.

Portanto, o aluno terá acesso ao Ambiente de Avaliação do Virtual-TANEB, após estar devidamente cadastrado no sistema (mostrado na figura 11). Uma vez autorizado a realizar a prova, o aluno inicializará o teste através da navegabilidade no painel de controle do Ambiente. Desta maneira, a figura 25 demonstra à página principal do Ambiente de Avaliação do *Software* Virtual-TANEB.



Figura 25 - Página principal do aluno no ambiente de avaliação do *Software* Virtual-TANEB.

As figuras 26 à 29, exibem telas de acesso ao ambiente da prova do Virtual-TANEB que o aluno acessará através da WEB. Desta forma, este ambiente possibilitará ao aluno interface personalizadas, que se apresentam de acordo com o sexo do aluno, como masculino, numa interação com o personagem “Joãozinho” e como feminino, cuja personagem é “Mariazinha”. Em ambos os casos, os personagens oferecem auxílio aos alunos na resolução dos itens que compõem a prova de matemática, cujo tema é grandezas e medidas.

Na figura a seguir, a pergunta tem como referência o descritor “D8 – Estabelecer relações entre unidades de tempo” (BRASIL, 2009). (Vide Figura 26)



Figura 26 - Questão no ambiente feminino de avaliação do Software Virtual-TANEB com referência o descritor D8.

Nas figuras 26 e 27, o aluno poderá pedir ajuda durante a realização da prova ao agente DICA por meio da interação com a personagem animada “Mariazinha” do Software. O agente de Software do Ambiente de Avaliação do Virtual-TANEB funciona como um assistente em sala de aula, maneira tradicional, e

irá auxiliar o aluno a responder o item da prova quanto à explicação dos elementos que pertencem à pergunta sem induzi-lo à resposta correta.



Figura 27 - Questão no ambiente feminino de avaliação do Software Virtual-TANEB com referência o descritor D11.

Na figura 27, a pergunta tem como referência o descritor “D11 - Resolver problema envolvendo o cálculo de perímetro de figuras planas, desenhadas em malhas quadriculadas” (BRASIL, 2009).

Nas figuras 28 e 29, o aluno também poderá pedir ajuda durante a realização da prova ao agente DICA por meio da interação com o personagem animado “Joãozinho” do *Software*, funcionando como um assistente na resolução dos itens. Na figura 28, a seguir, a pergunta tem como referência o descritor “D9 – Estabelecer relações entre o horário de início e término e/ou intervalo da duração de um evento ou acontecimento” (BRASIL, 2009). (Vide Figura 28)

The screenshot displays the Virtual-TANEB assessment environment. At the top, there is a blue header with the Virtual-TANEB logo, the UFMA logo, and a user profile icon labeled 'SAIR jacason Aluno'. Below the header, the interface is divided into several sections. On the left, there is a 'PAINEL DE CONTROLE' (Control Panel) with two buttons: 'Realizar Prova' (Take Test) and 'Visualizar Desempenho' (View Performance). The main area is titled 'Avaliação' (Assessment) and contains a question labeled 'Pergunta 4'. The question text is: 'Quando Maria colocou um bolo para assar, o relógio marcava (ver figura). O bolo ficou pronto em 30 minutos. Que horário o relógio estava marcando quando o bolo ficou pronto?'. Below the text is a circular clock face showing the time 11:50. The hour hand is between 11 and 12, and the minute hand is pointing at 10. Below the clock, there are four multiple-choice options: 1) 11 horas e 50 minutos, 2) 12 horas, 3) 12 horas e 5 minutos, and 4) 12 horas e 10 minutos. A 'Responder' (Answer) button is located below the options. On the right side of the assessment area, there is a cartoon character wearing a yellow hard hat and a blue shirt, with a 'Quer Dica?' (Want a Hint?) button below it. At the bottom of the interface, there is a copyright notice: 'Copyright © 2009-2010 Virtual-TANEB v.2.0'.

Figura 28 - Questão no ambiente masculino de avaliação do Software Virtual-TANEB com referência o descritor D9.

Na figura 29, a seguir, a pergunta tem como referência o descritor “D10 – Em um problema, estabelecer trocas entre cédulas e moedas do sistema monetário brasileiro, em função de seus valores” (BRASIL, 2009). (Vide Figura 29)

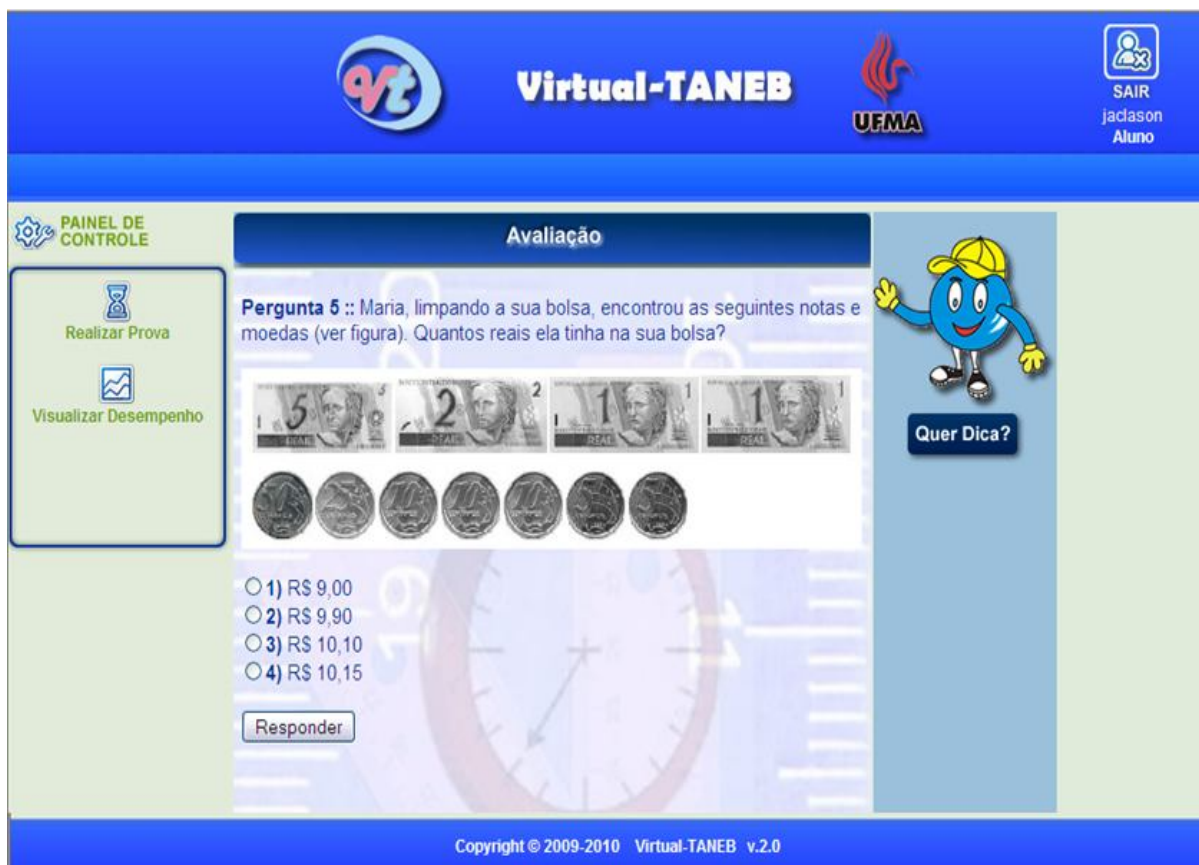


Figura 29 - Questão no ambiente masculino de avaliação do Software Virtual-TANEB com referência o descritor D10.

Na figura 30, a seguir, é mostrada a interação com o personagem “Joãozinho”, que representa o agente DICA, mostrando uma dica, solicitada pelo aluno, de acordo com o item a ser respondido.



Figura 30 - Solicitação de DICA no ambiente masculino de avaliação do Software Virtual-TANEB.

Já na figura 31, a seguir, é mostrada a interação com a personagem “Mariazinha”, que também representa o agente DICA. (Vide Figura 31)

Neste caso, o agente DICA mostra que não está disponível, nenhum dica para o item correspondido.



Figura 31 - Solicitação de DICA no ambiente feminino de avaliação do Software Virtual-TANEB.

Após o aluno realizar a prova é gerado um resultado preliminar, mostrando as respostas marcadas, alternativas corretas e o registro do tempo, de cada item respondido, assim, como os códigos que identificam o item e a prova, respectivamente. Desta forma, pode ser observado na figura 32, a seguir, o resultado da prova realizada.

PERGUNTA	ID PROVA	ID ITEM	ALTERNATIVA MARCADA	ALTERNATIVA CORRETA	TEMPO ITEM
1	5	3	2	2	03:06:11
2	5	6	4	4	03:07:53
3	5	4	3	3	03:09:02
4	5	1	3	3	03:09:50
5	5	7	3	3	03:10:34

Copyright © 2009-2010 Virtual-TANEB v.2.0

Figura 32 - Finalização de prova no ambiente de avaliação do Software Virtual-TANEB

O aluno, após ter finalizado a prova, caso tente realizar novamente uma outra prova, será necessário um novo agendamento com data e horário, de acordo com a disponibilidade do aluno. Caso isso não ocorra, o aluno ao tentar realizar uma prova não agendada, irá se deparar com uma exceção, ou seja, uma tela informando a não existência de prova agendada.

Para melhor visualização, é possível observar esta restrição na figura 33, a seguir.



Figura 33 - Inexistência de prova a ser realizada no ambiente de avaliação do Software Virtual-TANEB.

A figura 34 mostra todo registro de provas realizadas, de cada aluno *logado* (ou seja, cadastrado e autenticado) no Software Virtual-TANEB. (Vide Figura 34)



Figura 34 - Desempenho do aluno no ambiente de avaliação do Software Virtual-TANEB.

Na próxima seção é mostrado alguns testes com o Software aqui proposto.

7.2 Teste no Virtual-TANEB

As atividades de Teste têm contribuído para melhorar a qualidade do Software VT, tendo em vista a melhoria do sistema, foram aplicados no Virtual-TANEB Teste de Unidade e Teste de Funcionalidade. A seguir é realizado uma descrição do Ambiente e os itens analisados.

- **Itens a serem analisados e testados:** Foram testado as classes `cadastrarDAO.java` do pacote `virtual taneb.usuários` e `cadastrarDAO.java` do pacote `virtual taneb.item`, estas classes fazem chamadas aos métodos das outra classes e conecta ao banco de dados como é mostrado na figura 35.
- **Abordagem:** Foi utilizado o *Junit* (BIASI, 2006) para automatizar a aplicação de Teste de Unidade, levando em consideração os requisitos do

Virtual-TANEB e seus diagramas. Na atividade de Teste de Funcionalidade foram analisados os menus, executando tarefas de usuário “administrador do sistema” tais como cadastrar um novo usuário, cadastrar itens, dentre outros mostrado na figura 15 desta dissertação.

- **Ambiente:** Os Teste foram desenvolvidos no Laboratório de Sistemas Inteligentes (LSI) da UFMA em um Hardware com dois gigabytes de memória RAM, processador Dual core de 2GHz de frequência, sistema operacional Windows XP Server Pack 3, IDE utilizada foi o NetBeans 6.8 que vem com o *Framework Junit*.

```

1  /**
2   * @author Flávio Barros
3   */
4   package virtualTaneb.usuario;
5   import virtualTaneb.*;
6   import java.sql.SQLException;
7   import javax.swing.JOptionPane;
8
9   public class cadastrarDAO {
10      // CADASTRAR USUÁRIO
11      public void cadastrarUsuarios(usuarios usr) {
12          try {
13              conexao.conectar();
14              conexao.st.execute("INSERT INTO usuario "
15                  + "VALUES(" + usr.getIdusuario() + "," + usr.getIdendereco() + ","
16                  + usr.getIdtipo() + "," + usr.getIdinstituicao() + ","
17                  + usr.getIdpais() + "," + usr.getIdtipo_nacionalidade() + ","
18                  + usr.getIdcor() + "," + usr.getIdfuncao() + ","
19                  + usr.getIdcargo() + "," + usr.getNome() + ","
20                  + usr.getEmail() + "," + usr.getData_Nasc() + ","
21                  + usr.getSexo() + "," + usr.getNome_Mae() + ","
22                  + usr.getIdentidade()+ "," + usr.getOrgao_Emissor_Ident()+","
23                  + usr.getData_Expedicao_Ident() + "," + usr.getLogin() + ","
24                  + usr.getSenha() + ")");
25          } catch (SQLException e) {
26              JOptionPane.showMessageDialog(null, e.getMessage() + "\nErro no"
27                  + "acesso ao banco de dados!!!\nNão foi possível cadastrar "
28                  + "usuário!", "ERRO",JOptionPane.ERROR_MESSAGE);
29          } finally {
30              conexao.desconectar();
31          }
32      } // FIM DO CADASTRAR DADOS
33  }

```

Figura 35 - Trecho do código da classe cadastrarDAO.java do pacote virtualTaneb.Item do Software Virtual-TANEB.

A figura 35, mostra trecho da classe cadastrarDAO.java do Software Virtual-TANEB, onde a mesma foi utilizada nos Testes aqui proposto.

Na tabela 4, a seguir, é mostrado parte dos procedimentos aplicados no Teste do *Software* VT.

Tabela 4 - Parte dos procedimentos aplicados nos Testes de Unidade e testes de Funcionalidade.

Teste de Unidade	Verificação
1) Método <code>testcadastrarDisciplina</code> testa o pacote os métodos das classes pertencentes ao pacote <code>virtualTaneb.Item</code> ; passa os parâmetros (6, matemática);	1) A saída foi a esperada, ou seja, foi verificado que a disciplina matemática e o seu <i>id</i> correspondente (6) foram cadastrados no banco de dados, sendo assim, pode ser notado que os outros métodos das classes pertencente ao pacote <code>virtualTaneb.Item</code> estão recebendo os parâmetros corretamente (ex. métodos <i>get</i> e <i>set</i>);
2) O método <code>testcadastrarCargo</code> (<code>cadastrarDAOTest.java</code>) testa no pacote os métodos das classes pertencentes ao pacote <code>virtualTaneb.usuario</code> passar os parâmetros, (5 Analista);	2) A saída foi a esperada, ou seja, foi verificado que foi criado no banco de dados o cargo de analista e o seu <i>id</i> correspondente (5), sendo assim, pode ser notado que os outros métodos das classes pertencente ao pacote <code>virtualTaneb.usuario</code> estão recebendo os parâmetro corretamente(métodos <i>get</i> e <i>set</i>);
3) O método <code>testcadastrarFunção</code> (<code>cadastrarDAOTest.java</code>) testa o pacote os métodos das classes pertencentes ao pacote <code>virtualTaneb.usuario</code> passar os parâmetros (4, Administração);	3) A saída foi a esperada, ou seja, foi verificado que foi criado no sistema a função de administrador e o seu <i>id</i> correspondente (4), sendo assim, pode ser notado que os outros métodos das classes pertencente ao pacote <code>virtualTaneb.usuario</code> estão recebendo os parâmetro corretamente (ex. métodos <i>get</i> e <i>set</i>);
Teste de Funcionalidade	Verificação
1) Foram testados os menus 'cadastrar usuário', 'visualizar usuário', 'cadastra item' e 'visualizar item' da seção Administrador e da área destinada aos discentes foram os menus 'visualizar desempenho' e 'liberar prova'. Estes testes foram baseados nas atividades a serem realizadas pelo usuário e nas especificações de funcionalidade.	1) Os menus estão de acordo com especificação do sistema, quando realizada a atividade 'cadastrar usuário' foi realizada de forma eficiente e rápida, e ainda verificamos se estava salvo no banco de dados.

7.3 Considerações Finais

O objetivo deste capítulo foi apresentar uma implementação que comprove a efetividade da modelagem proposta no Capítulo 6 do Ambiente de Avaliação do VT, observando conceitos das tecnologias usadas. Além disso, apresentaram-se algumas interfaces do protótipo do Ambiente de Avaliação do *Software Virtual-TANEB*, que se caracteriza como uma das principais contribuições científica resultante deste trabalho. E, finalmente, é mostrada a realização de alguns Testes (de Unidade e de Funcionalidade) no *Software Virtual-TANEB*.

O próximo capítulo expõe conclusões do referido trabalho, com ênfase nos trabalhos relacionados, nas principais contribuições, nos resultados alcançados e nas perspectivas futuras.

8 CONCLUSÃO

O presente trabalho apresentou a arquitetura do agente de *Software* “Agente DICA” acessando uma Ontologia no domínio da matemática, grandezas e medidas, a ONTOGRAM, aqui desenvolvida e integrada ao Virtual-TANEB, *Software* relacionado à Teoria da Resposta ao Item (TRI), a fim de avaliar o nível de conhecimento do aluno referente à matemática lecionada no quinto ano do ensino fundamental. Este foi aceito para publicação no XXI Simpósio Brasileiro de Informática na Educação (SBIE 2010) (BARROS *et al.*, 2010).

8.1 Resultados Alcançados

Resultados favoráveis foram alcançados durante aplicação de Teste Funcional e de Unidade, realizados com o *Software* VT (mostrado na seção 7.2). Durante os Testes Funcionais (ou Teste Baseado na Especificação), foram levados em consideração os requisitos do sistema, sem realçar detalhes de implementação interna.

A aplicação do Teste de Unidade foi realizada na menor parte funcional do *software*, com objetivo de comparar sua funcionalidade, de um determinado módulo ao que foi proposto na especificação do mesmo, de forma a assegurar que este não entre em contradição. Portanto, os Testes realizados como o *Software* VT foram satisfatórios.

8.2 Principais Contribuições

As principais contribuições deste trabalho são:

- Desenvolvimento do agente DICA para auxiliar o aluno no entendimento das questões;

- Criação da Ontologia ONTOGRAM contendo os principais conceitos do tema, grandezas e medidas;
- Fornecimento de dicas aos alunos para esclarecer os conceitos quando apresentadas dificuldades. Portanto, as dicas são representadas obedecendo uma especificação de Ontologia;
- Agilidade na correção, uma vez que no Ambiente de Avaliação do Virtual-TANEB, a correção é automática ao final da prova, evitando assim custos com avaliadores;
- Mecanismos Computacionais utilizando Inteligência Artificial para melhorar e aperfeiçoar a avaliação do nível de conhecimento dos alunos em relação à matemática do quinto ano do ensino fundamental;
- Automatização no processo de Avaliação da Educação Básica;
- Redução no tempo de resposta atual para a execução, análise e resultado do diagnóstico da prova do SAEB;

8.3 Perspectivas Futuras

Haja vista que o desenvolvimento desse trabalho procurou determinar a representação da arquitetura do agente DICA utilizado no Ambiente de Avaliação do *Software* Virtual-TANEB, têm-se como perspectivas futuras novas pesquisas:

- Expansão da Implementação do Virtual-TANEB permitindo que o pedagogo faça a avaliação do item a ser cadastrado de forma automática, pois a avaliação atualmente é feita manualmente;
- Implementação e integração de um *Software* que automatize a Teoria da Resposta ao Item (TRI);
- Implementar a interação dos agentes “Monitor” e “Seletor”, permitindo que o Virtual-TANEB se adapte ao nível de conhecimento dos alunos;
- Extensão da ONTOGRAM para abranger o conteúdo de outras séries da disciplina de matemática;

- Criação de Ontologias com o conteúdo de outras disciplinas;
- Geração automática de gráficos e planilhas que permitam a análise do resultado das provas;
- Execução de testes, como por exemplo, em escolas, para uma melhor análise de adaptação do *Software* aos usuários finais.

REFERÊNCIAS

- [1] ANDRADE, Dalton F.; TAVARES, Heliton R.; VALLE, Raquel da C. **Teoria da Resposta ao Item: Conceitos e Aplicações**. SINAPE, São Paulo, 2000.
- [2] BARROS, José Flávio Gomes; LABIDI, Sofiane; COSTA, Nilson S., *et al.* **VIRTUAL-TANEB: Um Ambiente de Avaliação utilizando Agente de Software e Ontologia no Processo de Avaliação da Aprendizagem da Matemática do Quinto Ano do Ensino Fundamental**. WORKSHOP, SBIE2010, Nov. 2010.
- [3] BELLIFEMINE, Fabio; CAIRE, Giovanni; GREENWOOD, Dominic. **Developing Multi-Agent Systems with JADE**. Wiley, Chichester. 2007.
- [4] BIASI, Luciano B. **Geração Automatizada Drives e Stubs de teste para Junit a partir das especificações da U2TP**. Dissertação (Mestrado), Porto Alegre, 2006.
- [5] BOOCH, G., RUMBAUGH, J; JACOBSON, I. **UML: Guia do Usuário**. Editora Campus. Rio de Janeiro - RJ, 2000.
- [6] BORDINI, Rafael; HÜBNER, Jomi; WOOLDRIDGE, Michael. **Programming Multi-Agent Systems in AgentSpeak using Jason**. Wiley, Chichester. 2007.
- [7] BRAGA, Ciro de Carvalho. **Estudo do JADE e do CIM no contexto de um Sistema SCADA**. Monografia (Especialização) - Programa de Pós-Graduação em Informática da Universidade Tecnológica Federal do Paraná, Curitiba, 2006.
- [8] BRASIL, Ministério da Educação. **PDE – PROVA BRASIL: Plano de Desenvolvimento da Educação: matrizes de referência, tópicos e descritores**. Brasília, 2009.
- [9] CARNEIRO, Raquel E.; BRITO, Parcilene F. de. **Definição de uma Ontologia em OWL para Representação de Conteúdos Educacionais**. In: VII ENCONTRO DE ESTUDANTES DE INFORMÁTICA DO ESTADO DO TOCANTINS, Palmas: 2005.

- [10] CASTRO, Maria H. G. de. **Avaliação do Sistema Educacional Brasileiro: Tendências e Perspectivas**. Brasília: Instituto Nacional de Estudos e Pesquisas Educacionais, 1998.
- [11] CASTRO, Pedriana de J. P.; LABIDI, Sofiane; COSTA, Nilson S., *et al.* **Aplicação da teoria da resposta ao Item e dos critérios de avaliação do saeb para avaliação do desempenho dos alunos da 4ª Série do Ensino Fundamental na Disciplina Matemática no Conteúdo da Geometria no Software Proposto Virtual-TANEB**. INTERTECH2010, Bahia, BA, 2010.
- [12] CONSED. **Prêmio Nacional de Referência em Gestão Escolar: Ano-Base: 2008**. Manual de Orientações. Brasília, Nov. 2008.
- [13] CURY, Carlos R. J. **A Educação Básica no Brasil**. Educação & Sociedade., Campinas, vol. 23, n. 80, set. 2002.
- [14] DRUMOND, Lucas R. **Aquisição automatizada de hierarquias de conceitos de Ontologias utilizando aprendizagem estatística relacional**. Dissertação (Mestrado em Ciência da Computação) – Curso de Pós-Graduação em Engenharia de Eletricidade, Universidade Federal do Maranhão, São Luís, 2009.
- [15] FARQUHAR, Adam; FIKES, Richard and RICE, James. **The Ontolingua Server: a Tool for Collaborative Ontology Construction**. Stanford University, 1996.
- [16] FARRUGIA, Beatriz. **35 Universidades Federais usarão ENEM como Vestibular 2011**. 9, ago, 2010. Disponível em: <<http://www.enem2010.org/2010/08/09/35-universidades-federais-usarao-enem-como-vestibular-2011/>>. Acesso em: set. 2010.
- [17] FRIGOTTO, Gaudêncio; CIAVATTA, Maria. **Educação básica no Brasil na década de 1990: subordinação ativa e consentida à lógica do mercado**. Educação & Sociedade. Campinas, vol. 24, n. 82, abr. 2003.
- [18] GAMA, Maria Clara S. S. **A Teoria das Inteligências Múltiplas e suas implicações para Educação**. 1998. Disponível em:

- <<http://www.homemdemello.com.br/psicologia/intelmult.html>>. Acesso em: 23 ago. 2010.
- [19] GIUNCHIGLIA, Fausto, MYLOPOULOS, John, PERINI, Anna. **The Tropos Software Development Methodology: Processes, Models and Diagrams**. Proceedings of Third International Workshop on Agent-oriented Software Engineering – AOSE, 2002.
- [20] GONÇALVES, Edson. **Desenvolvendo aplicações WEB com NetBeans IDE 6**. Editora Ciência Moderna. São Paulo, 2008.
- [21] GONZAGA, Thiago; BENTES, Cristiana, CASTRO, Maria C. S., *et al.* **Uso de Memória Compartilhada Distribuída na Coordenação de Multi-Agentes**. 1998.
- [22] GUARINO, Nicola. **Formal Ontology in Information Systems**. Proceedings of FOIS'98, Trento, Italy, 6-8 June 1998. Amsterdam, IOS Press, pp. 3-15.
- [23] GUEDES, Gilleanes T. A. **UML: Uma Abordagem Prática**. 3 ed. Novatec Editora. São Paulo – SP, 2008.
- [24] INEP, 1a. **Prova Brasil e SAEB: Semelhanças e Diferenças**. 2009. Disponível em: <http://provabrasil.inep.gov.br/index.php?option=com_content&task=view&id=1&Itemid=98>. Acesso em: 06 nov. 2009.
- [25] INEP, 1b. **Prova Brasil e SAEB: Objetivos das Avaliações**. 2009. Disponível em: <http://provabrasil.inep.gov.br/index.php?option=com_content&task=view&id=52&Itemid=63>. Acesso em: 26 dez. 2009.
- [26] INEP, 1c. **Prova Brasil e SAEB: Avaliações Orientam Políticas Públicas**. 2009. Disponível em: <http://provabrasil.inep.gov.br/index.php?option=com_content&task=view&id=33&Itemid=60>. Acesso em: 26 dez. 2009.

- [27] **Java Ontology Editor (JOE)**. Disponível em: <<http://www.cse.sc.edu/research/cit/demos/java/joe/>>. Acesso em: jun. 2010.
- [28] LINDOSO, Alisson N. **Uma metodologia baseada em Ontologias para a engenharia de aplicações multiagente**. Dissertação (Mestrado em Ciência da Computação) – Curso de Pós-Graduação em Engenharia de Eletricidade, Universidade Federal do Maranhão, São Luís, 2006.
- [29] MACHADO, Carlos E. A.; LABIDI, Sofiane; COSTA, Nilson S., *et al.* **Automatização Computacional do Processo de Avaliação da “Gestão Escolar” Baseado nas Diretrizes da Secretaria Executiva do CONSED**. WORKSHOP, SBIE2009, Nov. 2009.
- [30] MAEDCHE, Alexander and VOLZ, Raphael. **The Ontology Extraction & Maintenance Framework Text-To-Onto**. 2002.
- [31] MAHALINGAM, Kuhanandha and HUHNS, Michael N. **An Ontology Tool for Distributed Information Environments**. 1997.
- [32] MCGUINNESS, Deborah L.; FIKES, Richard; FEIGENBAUM, Ed. **Ontolingua**. Stanford University, 2005. Disponível em: <<http://ksl.stanford.edu/software/ontolingua/>>. Acesso em: jun. 2010.
- [33] MEC, 1a. **ENEM – Apresentação**. 2009. Disponível em: <http://portal.mec.gov.br/index.php?option=com_content&view=article&id=183&Itemid=86>. Acesso em: 26 dez. 2009.
- [34] MEC, 1b. **Encceja**. 2009. Disponível em: <http://portal.mec.gov.br/index.php?option=com_content&view=article&id=12485&Itemid=784>. Acesso em: 28 dez. 2009.
- [35] MEC, 1c. **Provinha Brasil**. 2009. Disponível em: <http://portal.mec.gov.br/index.php?option=com_content&view=article&id=211&Itemid=328>. Acesso em: 28 dez. 2009.

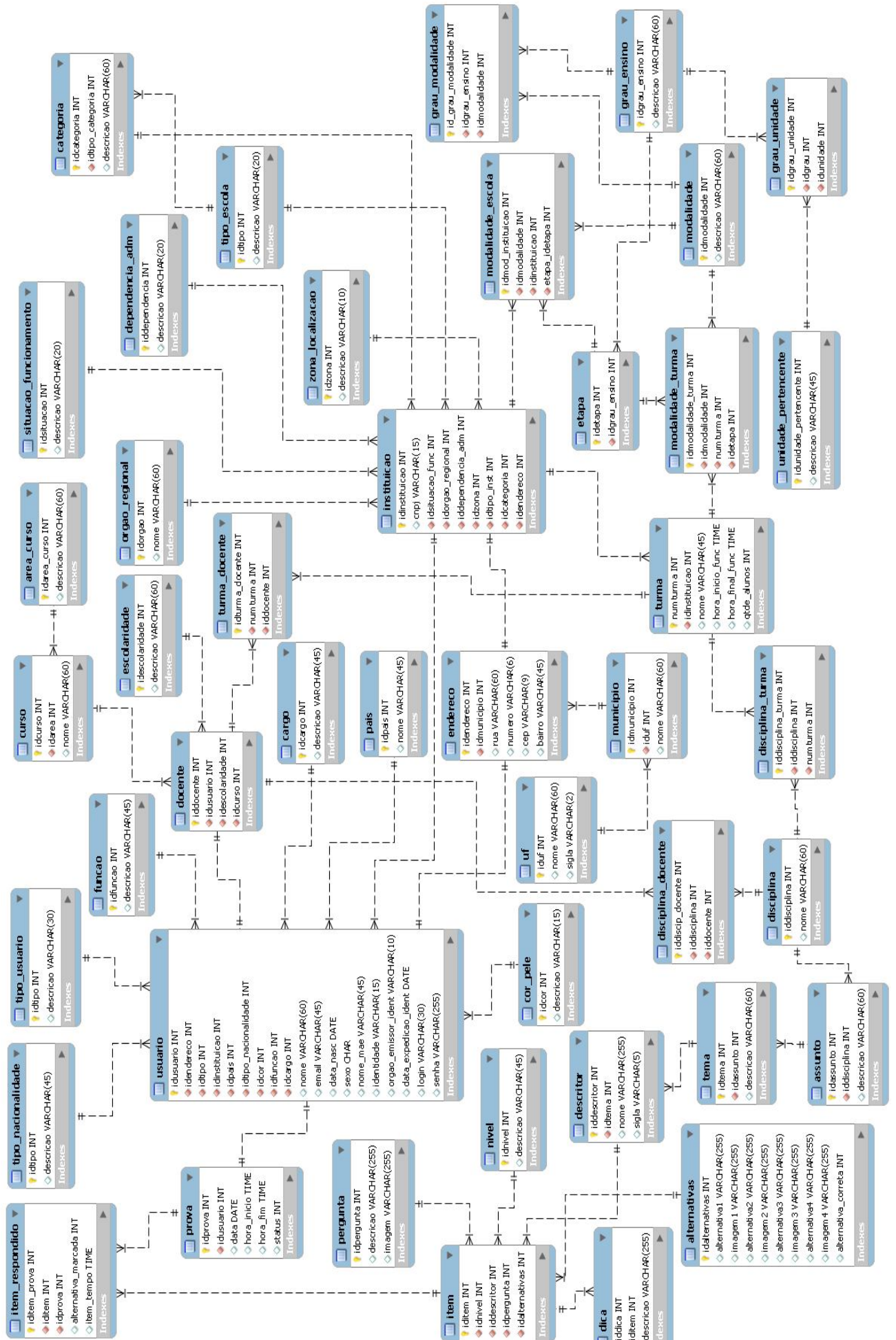
- [36] NOVELLO, Tanise Paula. **Sistema de Autoria e Tutor Inteligente – SATI: Desenvolvimento e Validação na Formação Continuada**. XXV Congresso da Sociedade Brasileira de Computação, São Leopoldo - RS, 2005.
- [37] NOY, N.; MCGUINNESS, D. **Ontology Development 101: A guide to creating your first ontology**. Disponível em: <<http://www.ksl.stanford.edu/people/dlm/papers/ontology101/ontology101-noy-mcguinness.html>>. Acesso em: mar. 2010.
- [38] PEREIRA, Mauro H. J. **Uma metodologia e uma ferramenta para o reuso gerativo na Engenharia de Domínio Multiagente**. Dissertação (Mestrado em Ciência da Computação) – Curso de Pós-Graduação em Engenharia de Eletricidade, Universidade Federal do Maranhão, São Luís, 2006.
- [39] PERINI, A. and SUSI, A. **Developing Tools for Agent-Oriented Visual Modeling**. In: Proc. of MATES'04, Vol. 3187 of LNCS. pp. 169–182, Springer-Verlag, 2004.
- [40] PERINI, Anna *et al.* **Tool for Agent Oriented Modeling (TAOME4E)**. Disponível em: <<http://sra.itc.it/tools/taom4e/>>. Acesso em: abr. 2010.
- [41] PRADO, Simone das G. D. **Um experimento no uso de ontologias para reforço da aprendizagem em educação à distância**. São Paulo. Tese (Doutorado em Engenharia de Computação) – Curso de Pós-Graduação em Engenharia de Computação e Sistemas Digitais, Escola Politécnica, Universidade de São Paulo, 2005.
- [42] PROTÉGÉ. Disponível em: <<http://protege.stanford.edu>>. Acesso em: abr. 2010.
- [43] RODRIGUES, Margarida M. M. **Avaliação Educacional Sistêmica na Perspectiva dos Testes de Desempenho e de seus Resultados: Estudo do SAEB**. Tese (Doutorado) – Universidade de Brasília – UnB, Brasília, abr. 2007.
- [44] RUSSEL, Stuart; NORVIG, Peter. **Inteligência Artificial**. Tradução 2.ed. Rio de Janeiro: Campus, 2004.

- [45] SANTOS, Neide; CAMPOS, Fernanda; BRAGA, Regina. **Ontologia para domínio da Educação mediada pela Web**. in 8^{vo} Taller Internacional de Software Educativo, Tise'03, Santiago, Chile, 2003.
- [46] SECCO, Rosemeire Lima. **Um Ambiente Interativo para Aprendizagem em Fração**. In SBIE 2009.
- [47] SILVA, Isabel; FREITAS, Carla. **Ontologia para Sistemas Configuracionais Urbanos**. Instituto de Informática – UFRGS, Porto Alegre, 2009.
- [48] SILVA, Ismênia G. L. **Projeto e Implementação de Sistemas Multi-Agentes: O Caso Tropos**. Dissertação (Mestrado) - Universidade Federal de Pernambuco, Recife, Ago.2005.
- [49] SILVA, Leonardo A. de M. **Estudo e Desenvolvimento de Sistemas Multiagentes usando JADE: Java Agent DEvelopment framework**. Monografia de Conclusão de Curso, Centro de Ciências Tecnológicas – CCT, UNIFOR, Fortaleza, Ceará, 2003.
- [50] SILVA, Mauro L. C. **Modelo de IDS Remoto baseado na tecnologia de Agentes, Web Services e MDA**. Dissertação (Mestrado em Ciência da Computação) – Curso de Pós-Graduação em Engenharia de Eletricidade, Universidade Federal do Maranhão, São Luís, 2006.
- [51] SOUZA, Vítor E. S.; FALBO, Ricardo de A. **Construindo Axiomas e Avaliando Ontologias em ODEd**. XVII SBES, Manaus, AM, 2003.
- [52] TEZZA, Rafael; BORNIA, Antonio C. **Teoria da Resposta ao Item: Vantagens e Oportunidades para a Engenharia de Produção**. In. XXIX Encontro Nacional de Engenharia de Produção. Salvador-BA, 2009.
- [53] VERAS, Jaclason M. ; LABIDI, Sofiane; COSTA, Nilson S., *et al.* **Uma Proposta de Modelagem para o Software Virtual-TANEB Baseado na Teoria da Resposta ao Item para Avaliar o Rendimento dos Alunos da Quarta Série do Ensino Fundamental em Relação ao Ensino da Matemática**. WORKSHOP, SBIE2009, Nov. 2009.

- [54] VERAS, Jaclason M. **Modelagem para o Software Virtual-TANEB baseado na teoria da resposta ao Item para avaliar o rendimento dos alunos.** Dissertação (Mestrado em Ciência da Computação) – Curso de Pós-Graduação em Engenharia de Eletricidade, Universidade Federal do Maranhão, São Luís, Fev. 2010.
- [55] WAKABAYASHI, Jukie K. **Infância Feliz: Matemática - 5º Ano - 4ª Série Ensino Fundamental.** Escala Educacional, Nacional, 2007.
- [56] ZINNER, Michael G. *et al.* **MySQL Workbench - Visual Database Design For Professionals.** v5.1.16 OSS. 2008. Disponível em: <<http://wb.mysql.com/>>. Acesso em: jan. 2010.

APÊNDICE

TABELAS DO BANCO DE DADOS DO VIRTUAL-TANEB - V.2.0



CLASSE CADASTRAR ITEM DO AMBIENTE VIRTUAL-TANEB V2.0

```
package virtualTaneb.Item;
import java.sql.ResultSet;
import virtualTaneb.conexao;
import java.sql.SQLException;
import java.util.ArrayList;
import javax.swing.JOptionPane;

// CADASTRAR ITEM
public class cadastrarDAO {
    public void cadastrarDisciplina(disciplina disc){
        .
    }// FIM CADASTRAR DADOS

    public void cadastrarAssunto(assunto ass){
        .
    }// FIM CADASTRAR DADOS

    public void cadastrarDescriptor(descriptor desc){
        .
    }// FIM CADASTRAR DADOS

    public void cadastrarTema(tema tem){
        .
    }// FIM CADASTRAR DADOS

    public void cadastrarAlternativas(alternativas alt){
        .
    }// FIM CADASTRAR DADOS

    public void cadastrarItemResp(ItemRespondido iResp){
        .
    }// FIM CADASTRAR DADOS

    public ArrayList<respItem> responderItem (int idProva) throws
    SQLException{
        .
        return lista;
    }

    public void atualizarItemResp(ItemRespondido iResp){
        .
    }// FIM CADASTRAR DADOS
}
```

**CLASSE ITEM RESPONDIDO DO AMBIENTE DE AVALIAÇÃO DO VIRTUAL-
TANEB**

```

package virtualTaneb.Item;
public class ItemRespondido {
    private int idItem_prova;
    private int idItem;
    private int idprova;
    private int idalternativa_marcada;
    private String Item_tempo;

    public int getIdItem() {
        return idItem;
    }
    ...

    public void setIdItem(int idItem) {
        this.idItem = idItem;
    }
    ...

    public ItemRespondido(int idItem, int idprova, int idalternativa_marcada,
String Item_tempo) {
        ...
    }
}

```

**CLASSE RESPONDER ITEM DO AMBIENTE DE AVALIAÇÃO DO VIRTUAL-
TANEB**

```

package virtualTaneb.Item;
public class respItem {
    private int idItem;
    private String pergunta;
    private String imgPerg;
    private String alt1;
    private String alt1img;
    private String alt2;
    private String alt2img;
    private String alt3;
    private String alt3img;
    private String alt4;
    private String alt4img;

    public int getIdItem() {
        return idItem;
    }
    ...

    public void setIdItem(int idItem) {
        this.idItem = idItem;
    }
}

```



```

    public respItem(int idItem, String pergunta, String imgPerg, String alt1,
String alt1img, String alt2, String alt2img, String alt3, String alt3img,
String alt4, String alt4img) {
        ...
    }
}

```

CLASSE DICA DO AMBIENTE DE AVALIAÇÃO DO VIRTUAL-TANEB

```

package virtualTaneb.Item;
public class dica {
    private int iddica;
    private int idItem;
    private String descricao;
    //MÉTODOS GETS
    public int getIddica() {
        return iddica;
    }
    ...
    //MÉTODOS SETS
    public void setIddica(int iddica) {
        this.iddica = iddica;
    }
    ...
    public dica(int iddica, int idItem, String descricao) {
        ...
    }
}

```

ANEXO A

A MATRIZ DE REFERÊNCIA DE MATEMÁTICA: Temas e seus Descritores - 4ª série/ 5º ano do Ensino Fundamental (BRASIL, 2009).

Tema I. Espaço e Forma

Descritores	4ª série (5º ano) EF
Identificar a localização/movimentação de objeto em mapas, croquis e outras representações gráficas	D1
Identificar propriedades comuns e diferenças entre poliedros e corpos redondos, relacionando figuras tridimensionais com suas planificações	D2
Identificar propriedades comuns e diferenças entre figuras bidimensionais pelo número de lados e pelos tipos de ângulos	D3
Identificar quadriláteros observando as posições relativas entre seus lados (paralelos, concorrentes, perpendiculares)	D4
Reconhecer a conservação ou modificação de medidas dos lados, do perímetro, da área em ampliação e/ou redução de figuras poligonais usando malhas quadriculadas	D5

Tema II. Grandezas e Medidas

Descritores	4ª série (5º ano) EF
Estimar a medida de grandezas utilizando unidades de medida convencionais ou não	D6
Resolver problemas significativos utilizando unidades de medida padronizadas como km/m/cm/mm, kg/g/mg, l/ml	D7
Estabelecer relações entre unidades de medida de tempo	D8
Estabelecer relações entre o horário de início e término e/ou o intervalo da duração de um evento ou acontecimento	D9
Num problema, estabelecer trocas entre cédulas e moedas do sistema monetário brasileiro, em função de seus valores	D10
Resolver problema envolvendo o cálculo do perímetro de figuras planas, desenhadas em malhas quadriculadas	D11
Resolver problema envolvendo o cálculo ou estimativa de áreas de figuras planas, desenhadas em malhas quadriculadas	D12

Tema III. Números e Operações/Álgebra e Funções

Descritores	4ª série (5º ano) EF
Reconhecer e utilizar características do sistema de numeração	D13

decimal, tais como agrupamentos e trocas na base 10 e princípio do valor posicional	
Identificar a localização de números naturais na reta numérica	D14
Reconhecer a decomposição de números naturais nas suas diversas ordens	D15
Reconhecer a composição e a decomposição de números naturais em sua forma polinomial	D16
Calcular o resultado de uma adição ou subtração de números naturais	D17
Calcular o resultado de uma multiplicação ou divisão de números naturais	D18
Resolver problema com números naturais, envolvendo diferentes significados da adição ou subtração: juntar, alteração de um estado inicial (positiva ou negativa), comparação e mais de uma transformação (positiva ou negativa)	D19
Resolver problema com números naturais, envolvendo diferentes significados da multiplicação ou divisão: multiplicação comparativa, idéia de proporcionalidade, configuração retangular e combinatória	D20
Identificar diferentes representações de um mesmo número racional	D21
Identificar a localização de números racionais representados na forma decimal na reta numérica	D22
Resolver problema utilizando a escrita decimal de cédulas e moedas do sistema monetário brasileiro	D23
Identificar fração como representação que pode estar associada a diferentes significados.	D24
Resolver problema com números racionais expressos na forma decimal envolvendo diferentes significados da adição ou subtração	D25
Resolver problema envolvendo noções de porcentagem (25%, 50%, 100%)	D26

Tema IV. Tratamento da Informação

Descritores	4ª série (5º ano) EF
Ler informações e dados apresentados em tabelas	D27
Ler informações e dados apresentados em gráficos (particularmente em gráficos de colunas)	D28

ANEXO B

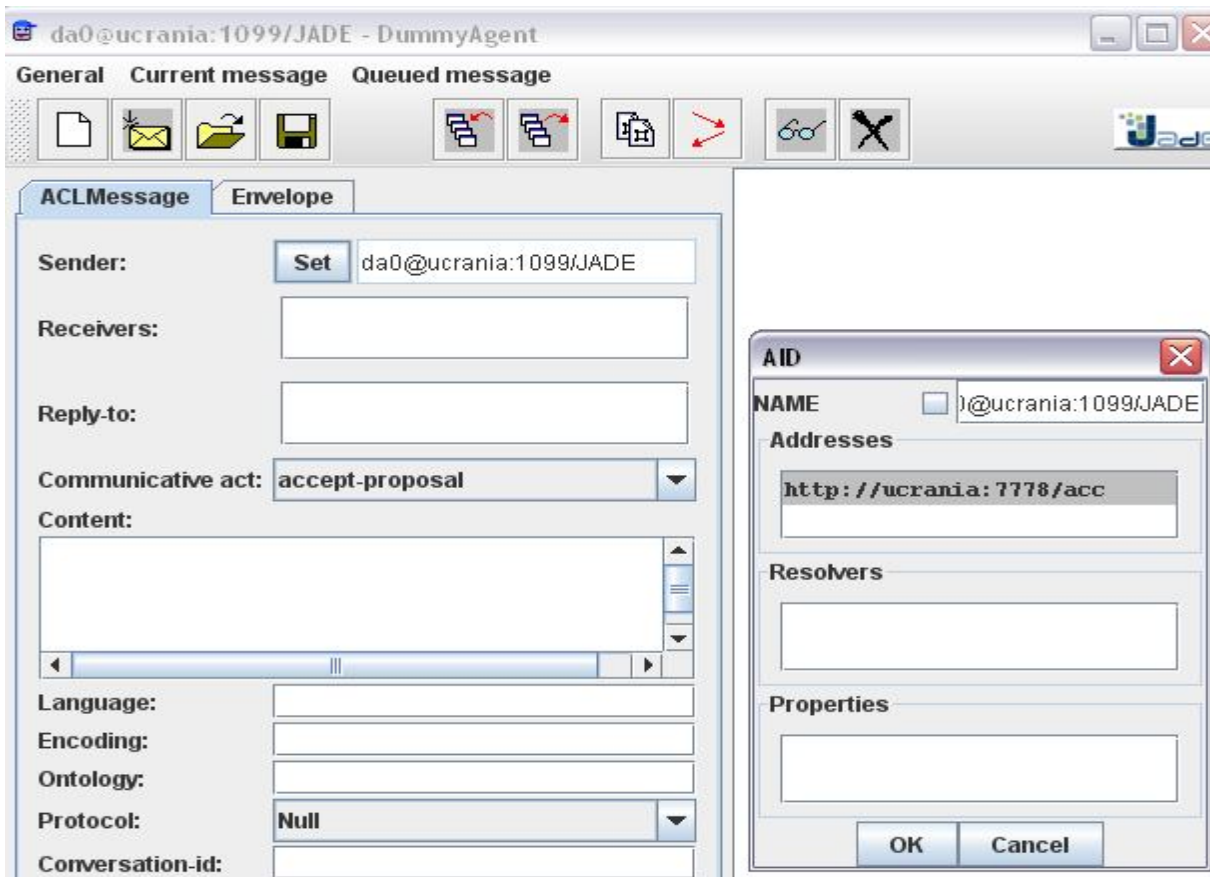
Demonstração de uma estrutura básica de uma mensagem KQML (SILVA, 2003).

```
(KQML-performative
  :sender <word>
  :receiver <word>
  :reply-with <word>
  :language <word>
  :ontology <word>
  :content <expression>
  ..... )
```

Demonstração da estrutura básica de uma mensagem FIPA-ACL (BELLIFEMINE; CAIRE; GREENWOOD, 2007)

```
(request
  :sender (agent-identifier :name alice@mydomain.com)
  :receiver (agent-identifier :name bob@yourdomain.com)
  :ontology travel-assistant
  :language FIPA-SL
  :protocol fipa-request
  :content
    ""((action
      (agent-identifier :name bob@yourdomain.com)
      (book-hotel :arrival 15/10/2006
                  :departure 05/07/2002 ... )
    )) ""
)
```

Demonstração da ferramenta Dummy Agent (BELLIFEMINE; CAIRE; GREENWOOD, 2007).



Demonstração da ferramenta Sniffer Agent (BELLIFEMINE; CAIRE; GREENWOOD, 2007).

