



UNIVERSIDADE FEDERAL RURAL DO SEMI-ÁRIDO
PRÓ-REITORIA DE GRADUAÇÃO
DEPARTAMENTO DE CIÊNCIAS EXATAS E NATURAIS
BACHARELADO EM CIÊNCIA DA COMPUTAÇÃO

DANILO GOMES CARLOS

**APLICANDO PBL EM UM AMBIENTE VIRTUAL DE APRENDIZAGEM APOIADO
POR AGENTES DE SOFTWARE**

MOSSORÓ

2016

DANILO GOMES CARLOS

**APLICANDO PBL EM UM AMBIENTE VIRTUAL DE APRENDIZAGEM APOIADO
POR AGENTES DE SOFTWARE**

Monografia apresentada à Universidade Federal Rural do Semi-Árido como requisito para obtenção do título de Bacharel em Ciência da Computação.

Orientador: Francisco Milton Mendes Neto, Prof. Dr.

Coorientadora: Laysa Mabel de Oliveira Fontes, Prof^ª. Ma.

MOSSORÓ

2016

© Todos os direitos estão reservados a Universidade Federal Rural do Semi-Árido. O conteúdo desta obra é de inteira responsabilidade do (a) autor (a), sendo o mesmo, passível de sanções administrativas ou penais, caso sejam infringidas as leis que regulamentam a Propriedade Intelectual, respectivamente, Patentes: Lei nº 9.279/1996 e Direitos Autorais: Lei nº 9.610/1998. O conteúdo desta obra tomar-se-á de domínio público após a data de defesa e homologação da sua respectiva ata. A mesma poderá servir de base literária para novas pesquisas, desde que a obra e seu (a) respectivo (a) autor (a) sejam devidamente citados e mencionados os seus créditos bibliográficos.

C284a Carlos, Danilo Gomes.
Aplicando PBL em um ambiente virtual de
aprendizagem apoiado por agentes de software /
Danilo Gomes Carlos. - 2016.
91 f. : il.

Orientador: Francisco Milton Mendes Neto.
Coorientadora: Laysa Mabel de Oliveira Fontes.
Monografia (graduação) - Universidade Federal
Rural do Semi-árido, Curso de Ciência da
Computação, 2016.

1. Agente Pedagógico. 2. MOODLE. 3. Educação a
Distância. 4. Aprendizagem Baseada em Problema. 5.
Sistema Multiagente. I. Mendes Neto, Francisco
Milton, orient. II. Fontes, Laysa Mabel de
Oliveira, co-orient. III. Título.

O serviço de Geração Automática de Ficha Catalográfica para Trabalhos de Conclusão de Curso (TCC's) foi desenvolvido pelo Instituto de Ciências Matemáticas e de Computação da Universidade de São Paulo (USP) e gentilmente cedido para o Sistema de Bibliotecas da Universidade Federal Rural do Semi-Árido (SISBI-UFERSA), sendo customizado pela Superintendência de Tecnologia da Informação e Comunicação (SUTIC) sob orientação dos bibliotecários da instituição para ser adaptado às necessidades dos alunos dos Cursos de Graduação e Programas de Pós-Graduação da Universidade.

DANILO GOMES CARLOS

APLICANDO PBL EM UM AMBIENTE VIRTUAL DE APRENDIZAGEM APOIADO
POR AGENTES DE SOFTWARE

Monografia apresentada à Universidade
Federal Rural do Semi-Árido como requisito
para obtenção do título de Bacharel em
Ciência da Computação.

Defendida em: 31 / 05 / 2016.

BANCA EXAMINADORA

Francisco Milton Mendes Neto, Prof. Dr. (UFERSA)
Presidente

Laysa Mabel de Oliveira Fontes, Profª. Ma. (UFERSA)
Membro Examinador

Paulo Gabriel Gadelha Queiroz, Prof. Dr. (UFERSA)
Membro Examinador

Daniel Cavalcante Lopes, Prof. Dr. (UFERSA)
Membro Examinador

A Deus, em primeiro lugar, que me confia o Amor e a graça de viver. Que me deu o dom da inteligência e a capacidade de acreditar em mim mesmo.

Aos meus avôs, Francisco de Menezes Neto e Antônio Araújo Leite, que me ensinaram a viver com dignidade e trabalho duro.

A minha tia Wálbia Carlos, que sempre me incentivou a buscar me aperfeiçoar meus conhecimentos.

Aos meus pais, sem vocês nada disso seria possível. Obrigado pelo apoio, carinho e compreensão.

AGRADECIMENTOS

A minha família, por acreditar e investir em mim. Em especial a minha mãe por todo cuidado, amor e carinho conferidos a mim. Ao meu pai pela presença e pelo exemplo que, por muitas vezes rígido, me motivou a buscar sempre mais. Ao meu irmão pelos incentivos, pelos momentos compartilhados e pelo espelho que se tornou em minha vida.

Ao professor Francisco Milton Mendes Neto, pela orientação, confiança e paciência dedicados. Por ter insistido nos momentos de fraqueza sempre me apoiando e se mostrando presente.

A Laysa Mabel de Oliveira Fontes pela coorientação, por, em todos os momentos, ter sido prestativa e por me motivar a concluir este trabalho.

A Laura, minha namorada, que se mostrou paciente e me incentivou a buscar os meus objetivos. Obrigado também, por todos os momentos maravilhosos que passamos e passaremos juntos, sua presença em minha vida é uma graça divina.

A UFERSA pelo ambiente criativo e amigável que proporciona. Aos meus professores, colegas e todos aqueles que fizeram parte da minha vida acadêmica pelo companheirismo e por terem me auxiliado nessa experiência única.

Aos meus amigos, pelas alegrias, tristezas e dores compartilhadas. Vocês foram essenciais pra chegar onde estou.

A todos aqueles que de alguma forma estiveram e estão próximos de mim, fazendo a vida valer a pena.

Que os vossos esforços desafiem as impossibilidades, lembrai-vos de que as grandes coisas do homem foram conquistadas do que parecia impossível.

Charles Chaplin

RESUMO

A Aprendizagem Baseada em Problema é uma metodologia de aprendizagem na qual os estudantes aprendem em grupo através da resolução de um problema. O problema é aplicado de forma que os estudantes colaborem entre si, auxiliados por um professor que deve atuar como facilitador do aprendizado. Nesta metodologia, os conhecimentos são aplicados em situações semelhantes aos que enfrentariam durante sua atuação profissional, motivando os estudantes a resolverem problemas que poderiam ocorrer em seu cotidiano profissional. No entanto, sua aplicação pode apresentar algumas dificuldades. A Educação a Distância é uma modalidade de ensino que vem sendo cada vez mais empregada devido o avanço tecnológico e a facilidade de acesso a essas tecnologias. Porém, aplicar metodologias de aprendizagem alternativas, como a Aprendizagem Baseada em Problema, pode ser um desafio à parte. Algumas destas dificuldades podem ser contornadas utilizando técnicas de Inteligência Artificial. Esta monografia apresenta o desenvolvimento e avaliação de um sistema multiagente que visa auxiliar o facilitador e os estudantes na implantação da Aprendizagem Baseada em Problema no Ambiente Virtual de Aprendizagem MOODLE.

Palavras-chave: Agente Pedagógico. Aprendizagem Baseada em Problema. Educação a Distância. MOODLE. Sistema Multiagente.

LISTA DE FIGURAS

Figura 1 - O ciclo da PBL.....	19
Figura 2 - Modelos da metodologia MAS-CommonKADS+.....	25
Figura 3 - Componentes do Padrão FIPA.....	27
Figura 4 - Estrutura básica de um algoritmo genético.....	31
Figura 5 - Estrutura básica de um controlador baseado em lógica fuzzy.	33
Figura 6 - Abordagem multiagente de apoio à PBL na EaD.	35
Figura 7 - Esquema do sistema multiagente proposto por Fontes (2013).	36
Figura 8 - Esquema do sistema multiagente proposto neste trabalho.	37
Figura 9 - Modelo de Tarefas.	44
Figura 10 - Modelo de Papéis.....	45
Figura 11 - Modelo de Organização.	46
Figura 12 - Formulário de definição do problema.....	56
Figura 13 - Formulários de metas de aprendizagem e definição de requisitos.....	57
Figura 14 - Tela de listagem de grupos.	58
Figura 15 - Tela de visualização de grupos.	59
Figura 16 - Tela de listagem de sessões.	59
Figura 17 - Formulário de criação de sessão.	60
Figura 18 - Tela de visualização de sessão.....	61
Figura 19 - Formulário de avaliação de sessão.....	61
Figura 20 - Formulário da solução do problema.	62
Figura 21 - Formulário de avaliação final do estudante.	63
Figura 22 - Formulário de edição de perfil do estudante.....	64
Figura 23 - Tela de visualização do problema.....	65
Figura 24 - Formulário de termos desconhecidos.	66
Figura 25 - Tela de listagem de sessões.	66
Figura 26 - Formulário de relatório de sessão.	67
Figura 27 - Formulários de relatório final e solução do problema.	68
Figura 28 - Tela de visualização de grupo.....	68
Figura 29 - Formulário de avaliação de par.....	69
Figura 30 - Gráfico das médias das notas relacionadas à facilidade de uso percebida.	76
Figura 31 - Gráfico das médias das notas relacionadas à utilidade percebida.	76
Figura 32 - Gráfico das médias das notas relacionadas à intenção de uso percebida.....	77
Figura 33 - Gráfico das médias das notas relacionadas à complexidade da metodologia.....	78
Figura 34 - Gráfico das médias das notas relacionadas ao treinamento aplicado.	78

LISTA DE TABELAS

Tabela 1 - <i>Template</i> textual do AgP.	47
Tabela 2 - <i>Template</i> textual do AgMP.	48
Tabela 3 - <i>Template</i> textual do AgGG.	49
Tabela 4 - <i>Template</i> textual do AgMG.	50
Tabela 5 - <i>Template</i> textual do AgMS.	51
Tabela 6 - <i>Template</i> textual do AgDP.	52
Tabela 7 - <i>Template</i> textual do AgR.	53
Tabela 8 - <i>Template</i> textual do Agente DF.	54
Tabela 9 - Variáveis, conceitos e dimensões.	72
Tabela 10 - Conjunto dos dados obtidos do questionário.	74
Tabela 11 - Resultados dos cálculos dos coeficientes de correlação e variação de Pearson.	79

SUMÁRIO

1 INTRODUÇÃO.....	12
1.1 Problemática	12
1.2 Trabalhos Relacionados.....	13
1.3 Objetivo Geral.....	14
1.4 Organização do Trabalho.....	14
2 EDUCAÇÃO A DISTÂNCIA	15
3 APRENDIZAGEM BASEADA EM PROBLEMA	17
3.1 Estrutura de um Grupo na PBL.....	18
3.2 Ciclo da Aprendizagem Baseada em Problema	18
4 AGENTES E SISTEMAS MULTIAGENTE.....	20
4.1 Características	20
4.2 Tipos De Agentes.....	21
4.3 Agentes Pedagógicos	22
4.4 Sistemas Multiagente	22
4.4.1 Organização dos Sistemas Multiagente.....	23
5 MODELAGEM DE SMAs.....	24
5.1 Metodologia MAS-CommonKADS+	24
5.2 Comunicação e Gerenciamento de Agentes	26
6 ALGORITMOS GENÉTICOS.....	29
6.1 Conceitos Básicos dos Algoritmos Genéticos	29
6.1.1 Genes.....	29
6.1.2 Cromossomos	29
6.1.3 População	30
6.1.4 Geração.....	30
6.1.5 Função de Aptidão	30
6.2 Funcionamento dos Algoritmos Genéticos.....	30

7 LÓGICA FUZZY	32
7.1 Conjuntos Fuzzy	32
7.2 Regras Fuzzy.....	32
7.3 Inferência Fuzzy	33
7.4 Fuzzificação e Defuzzificação	33
8 SISTEMA MULTIAGENTE DE APOIO À PBL.....	34
8.1 Ferramentas Utilizadas.....	34
8.1.1 Moodle	34
8.1.2 Netbeans	34
8.1.3 Jade.....	34
8.2 Abordagem Multiagente de Apoio à PBL na EaD	34
8.3 Descrição do Sistema Multiagente.....	39
8.3.1 Agente Problema - AgP.....	39
8.3.2 Agente Monitorador de Problemas - AgMP	39
8.3.3 Agente Gerenciador de Grupos - AgGG	39
8.3.4 Agente Monitorador de Grupos - AgMG.....	40
8.3.5 Agente Monitorador de Sessões - AgMS	41
8.3.6 Agente Detector de Problemas - AgDP.....	41
8.3.7 Agente Recomendador - AgR	42
8.3.8 Agente Directory Facilitator - DF	42
8.4 Modelagem do Sistema Multiagente	43
8.4.1 <i>Template</i> textual do AgP	47
8.4.2 <i>Template</i> textual do AgMP	48
8.4.3 <i>Template</i> textual do AgGG	49
8.4.4 <i>Template</i> textual do AgMG.....	50
8.4.5 <i>Template</i> textual do AgMS	51
8.4.6 <i>Template</i> textual do AgDP	52

8.4.7 <i>Template</i> textual do AgR.....	53
8.4.8 <i>Template</i> textual do DF	54
8.5 Cenários de Uso	55
8.5.1 Cenários de Uso do Facilitador	55
8.5.2 Cenários de Uso do Estudante.....	63
9 ESTUDO DE CASO	70
9.1 Ferramentas Utilizadas.....	70
9.1.1 Google Forms.....	70
9.1.2 Microsoft Excel.....	70
9.2 Metodologia	70
9.2.1 Objetivos	71
9.2.2 Amostragem	72
9.2.3 Obtenção dos dados.....	72
9.3.4 Análise dos Dados.....	73
9.4 Apresentação dos Resultados.....	74
10 CONSIDERAÇÕES FINAIS E TRABALHOS FUTUROS	81

1 INTRODUÇÃO

O mercado de trabalho está se tornando cada vez mais competitivo e os empregadores passaram a exigir maior qualificação dos empregados. Por exemplo, cargos que exigiam que o candidato fosse apenas alfabetizado, agora requerem o ensino médio completo. Entretanto, o ensino tradicional, que requer a presença do aluno em sala de aula, não é acessível para algumas pessoas. É nesse contexto que a Educação a Distância (EaD) se estabeleceu.

A EaD é uma modalidade de educação mediada por tecnologias onde o professor e o aluno estão separados de forma espacial e/ou temporal. Atualmente vem ganhando bastante visibilidade por pesquisadores da área da informática aplicada à educação. Algumas pesquisas abordam, por exemplo, a implantação de diferentes tipos de metodologias de aprendizagem no âmbito da EaD. Uma dessas metodologias é a aprendizagem baseada em problema.

A aprendizagem baseada em problema, do inglês *Problem-Based Learning* (PBL), é uma metodologia inovadora que, através da discussão de uma problemática, estabelece uma conexão teoria-prática de maneira dinâmica (AKINOĞLU; TANDOĞAN, 2007). Segundo Hmelo-Silver (2004), quando aplicada corretamente, a PBL pode oferecer alguns benefícios, dentre eles:

- Desenvolver o pensamento crítico e a criatividade nos estudantes;
- Aumentar a capacidade de resolução de problemas e de liderança; e
- Estimular a aplicação dos conhecimentos adquiridos em situações reais.

1.1 Problemática

Na EaD, utiliza-se ferramentas tecnológicas para mediar o ensino e o aprendizado, tendo como objetivo atender o maior número possível de estudantes. Entretanto, em alguns casos, é necessário um suporte informatizado adequado às características de cada indivíduo (SILVA, 2012). Além disso, as ferramentas disponíveis nesses sistemas podem não oferecer um suporte suficiente para aquisição de conhecimento no processo de ensino e aprendizagem (JAQUES et al., 2002).

Algumas dificuldades podem oferecer riscos à efetividade relacionada a PBL. A complexidade na construção do problema, o acompanhamento efetivo na colaboração e no processo de aprendizagem, o método de avaliação complexo e a necessidade do facilitador conhecer as estratégias de aprendizagem são algumas das dificuldades encontradas durante a

aplicação da PBL (HMELO-SILVER, 2004; HMELO-SILVER; BARROWS, 2006, SAVERY, 2006).

Para contornar esses problemas, pesquisadores da área de informática aplicada à educação têm confiado na utilização de técnicas de Inteligência Artificial (IA) para auxiliar no projeto e desenvolvimento de ambientes de ensino e aprendizagem computadorizados (SANTOS et al., 2001). Entretanto, é importante avaliar a aceitação dessas ferramentas por parte dos usuários.

Não foi encontrado na literatura um trabalho que implantasse a PBL na EaD utilizando uma abordagem multiagente, que tenha sido realizado um estudo de caso para avaliar sua aceitação por parte dos usuários.

1.2 Trabalhos Relacionados

Felix e Tedesco (2008) apresentam uma ferramenta apoiada por uma sociedade de agentes para fazer acompanhamento, sugestão e formação de grupos de aprendizagem baseados nas informações de contexto dos aprendizes.

Bremgartner e Netto (2011) apresentam um Sistema Multiagente (SMA) que objetiva adaptar Ambientes Virtuais de Aprendizagem (AVAs) para auxiliar estudantes na execução de atividades. O SMA recomenda o auxílio por parte de estudantes com perfis adequados a ajudar outros estudantes com dúvidas nas atividades propostas pelo professor.

Melo-Solarte e Baranauskas (2008) propõem a utilização da PBL no AVA Sakai, utilizando ferramentas que complementam o ambiente para dar suporte à metodologia. Além das ferramentas que já existem no Sakai, foram criados os componentes de gestão das sessões (ACBPUser e Socialize) e um componente usado pra permitir acesso ao banco de dados.

Pinto et al. (2011) apresentam um ambiente virtual intitulado *Problem Based Learning - Virtual Enviroment* (PBL-VE), que tem como objetivo apoiar atividades relacionadas ao método PBL, tanto na educação presencial como na educação a distância.

Sancho et al. (2011) propõem uma abordagem que implementa a PBL no MOODLE. Nessa abordagem os autores utilizam ferramentas internas do AVA e alguns *plugins* específicos para dar suporte a formação e gerenciamento de grupos e avaliação dos alunos.

Mandal (2011) apresenta um *plugin* para o MOODLE, de apoio à PBL. Esse *plugin* consiste em um módulo de atividades para apoiar todos os passos que regem a PBL. Nesse trabalho, foram apresentadas todas as adaptações realizadas no MOODLE para atender à PBL.

Em (FONTES, 2013), foi proposta uma abordagem multiagente para atender as principais metas relacionadas ao auxílio no cumprimento da PBL, auxiliando o facilitador na detecção de possíveis problemas na colaboração e promovendo a aprendizagem dos estudantes durante o processo de aplicação da PBL. A solução integra ao MOODLE um módulo criado especificamente pra atender aos principais requisitos para o cumprimento das etapas da PBL e um sistema multiagente que visa apoiar o facilitador e os estudantes durante sua aplicação.

1.3 Objetivo Geral

O presente trabalho tem como objetivo geral implantar a proposta de Fontes (2013), descrita anteriormente, em um cenário real e realizar um estudo de caso para validar a solução. Para tanto, algumas alterações foram necessárias e serão explicadas no decorrer do trabalho.

1.4 Organização do Trabalho

Este trabalho está organizado da seguinte forma: a Seção 2 apresenta conceitos de educação a distância. A Seção 3 descreve aspectos relacionados à aprendizagem baseada em problema. A Seção 4 traz conceitos de agentes e SMAs. A Seção 5 apresenta conceitos sobre modelagem de SMAs. A Seção 6 apresenta a fundamentação teórica sobre algoritmos genéticos. A Seção 7 apresenta os conceitos de lógica *fuzzy*. A Seção 8 descreve o sistema multiagente de apoio à PBL desenvolvido neste trabalho, detalhando os agentes e como estes foram modelados. Esta sessão também exemplifica alguns cenários de uso do MOODLE. A Seção 9 detalha o estudo de caso realizado neste trabalho. E, por fim, a Seção 10 apresenta as considerações finais e os trabalhos futuros.

2 EDUCAÇÃO A DISTÂNCIA

Segundo Schuelter (2005), a Educação a Distância (EaD) é um conceito que remonta a antiguidade, conhecido desde a época de Platão que enviava cartas a seus discípulos para transmitir seus pensamentos filosóficos. No entanto, essa modalidade começou a se consolidar a partir do século XX quando passa a utilizar de recursos audiovisuais para a transmissão de informações. A partir da década de 1980, os avanços tecnológicos contribuíram ainda mais para a expansão da EaD (CHERMANN; BONINI, 2000).

A EaD é uma variante de ensino/aprendizagem em que o aluno está separado do professor espacialmente ou temporalmente. Esta modalidade de ensino utiliza-se, em geral, de ferramentas computacionais para promover o ensino e estabelecer a conexão aluno-professor (VERMEERSCH et al., 2006).

Na EaD, não é necessária a presença do aluno em sala de aula como é exigida no ensino tradicional. Em alguns cursos, todas as aulas são realizadas não presencialmente. Outros cursos adotam a modalidade semipresencial, que requer a presença em alguns horários específicos. Para os alunos que trabalham ou gostam de estudar por conta própria, pode ser uma grande vantagem. Outros estudantes, no entanto, se veem desmotivados pela falta de contato direto com o professor (FREITAS, 2005).

Flexibilidade e presença são as características que mais diferem este método de educação dos demais. O estudante muitas vezes não possui um horário fixo para o acompanhamento das aulas e a realização das tarefas acontece no seu próprio ritmo (CAPELETTI, 2014).

Em sua utilização através da Internet, pode ser aplicada em um Ambiente Virtual de Aprendizagem (AVA), onde o instrutor disponibiliza o conteúdo aos estudantes para que possam realizar suas tarefas. A comunicação é estabelecida por meio de ferramentas sociais promovidas pelo ambiente, como, por exemplo, fóruns e chats, que são utilizados para promover a interação entre o aluno, professor e colegas (CAPELETTI, 2014).

A EaD vem se tornando cada vez mais presente na realidade brasileira. Em 2007, 2.504.483 brasileiros estavam matriculados em cursos de EaD, segundo a Associação Brasileira de Educação a Distância (ABED) (LEMGRUBER, 2009). Em uma pesquisa realizada entre 2008 a 2010 pela ABED e pelo Ministério da Educação (MEC), a demanda por cursos de especialização a distância aumentou 60% durante esse período (MAIA, 2011). Ainda segundo a ABED, em 2012, foram registradas 5.772.466 matrículas, que equivalem a um aumento de 52,5% em relação a 2011 (ABED, 2013). Em 2014, o Instituto Nacional de

Estudos e Pesquisas Educacionais Anísio Teixeira (INEP) confirmou a existência de mais de 1,2 mil cursos a distância no Brasil, que equivalem a uma participação superior a 15% nas matrículas de graduação (INEP, 2014).

Contudo, apesar de a EaD apresentar um cenário promissor, alguns desafios dificultam sua implantação, como os descritos a seguir (CAPELETTI, 2014):

- O aluno se torna responsável pela aquisição do conhecimento e deve saber administrar o seu tempo para que possa cumprir os prazos exigidos na EaD;
- Os estudantes e professores devem ter acesso às tecnologias necessárias e ter conhecimento das ferramentas disponibilizadas;
- O professor deve colocar em prática estratégias pedagógicas.

Os impactos ocasionados por essas dificuldades podem ser minimizados com a criação de um sistema que: se adapte de acordo com o perfil do aluno; possa identificar e sugerir estratégias pedagógicas; e possa auxiliar o professor e os estudantes durante aplicação de metodologias de ensino/aprendizagem (FONTES, 2013; PONTES, 2010; SILVA, 2012).

3 APRENDIZAGEM BASEADA EM PROBLEMA

A PBL é uma metodologia de aprendizagem colaborativa que permite ao estudante desenvolver seu aprendizado a partir do trabalho em equipe, analisando quais conhecimentos são necessários para a resolução de um determinado problema, estimulando a coletividade e a autoaprendizagem (AKINOĞLU; TANDOĞAN, 2007).

Os problemas devem ser contextualizados em situações realistas, abordando vários conteúdos e, se possível, possuírem mais de uma solução para promover flexibilidade no aprendizado. Também é importante que o problema aborde temas que estejam ligados à realidade dos alunos para motivá-los a resolvê-los. Para resolver os problemas estabelecidos pelo professor, os alunos necessitam fundamentar sua resposta baseados em seus conhecimentos prévios, discutindo entre si o que é necessário ser aprendido para resolver o problema (HMELO-SILVER, 2004). Desse modo, o problema deve ser construído de forma que os estudantes explorem o que já sabem e evidenciem suas deficiências.

Contrariamente à metodologia de ensino convencional, onde o professor atua como um transmissor de informações, na PBL, o mesmo atua como um facilitador, orientando os alunos na discussão do problema e no desenvolvimento de estratégias para resolução do mesmo. O facilitador deve ser capaz de modelar boas estratégias de aprendizagem, estimular o pensamento crítico ao invés de apenas conhecer o conteúdo, motivar os alunos e acompanhar o progresso de uma grande quantidade de estudantes (HMELO-SILVER; BARROWS, 2006).

O aluno agora passa a ser o centro das atividades e, com o apoio do facilitador e dos outros membros da equipe, torna-se apto a desenvolver habilidades, tais como: (i) administração do tempo; (ii) disciplina; (iii) autocrítica; (iv) raciocínio; (v) elaboração de estratégias; e (vi) trabalho em equipe (AKINOĞLU; TANDOĞAN, 2007). Aos poucos o facilitador deve ir assumindo um papel mais próximo de espectador, à medida que os estudantes vão assumindo papéis de facilitador por si só (HMELO-SILVER, 2004). Dessa forma, é possível observar que o papel do facilitador é de fundamental importância para o sucesso na prática desta metodologia.

A colaboração entre os membros da equipe é essencial para a resolução do problema. Normalmente os problemas são desenvolvidos com complexidade, o que dificulta a resolução por um único estudante. A divisão em pequenos grupos promove a distribuição das funções de predição, teorização, resumo dos resultados e conclusões de acordo com a habilidade de cada um, permitindo que os mesmos possam potencializar seu aprendizado em determinada

área. O facilitador tem o papel fundamental de assegurar o envolvimento de todos os integrantes nesse processo (HMELO-SILVER, 2004).

3.1 Estrutura de um Grupo na PBL

Usualmente um grupo na PBL é constituído por no máximo dez alunos. Em cada sessão, um aluno é escolhido como coordenador e outro como relator, com o intuito de que outros membros do grupo tenham a oportunidade de desempenhar esses papéis (SANTOS et al., 2007). O coordenador instrui os alunos a discutir de acordo com uma metodologia estabelecida e dá suporte ao relator, além de encorajar a participação efetiva de todos os componentes. O relator é responsável pela pauta dos encontros e por transcrever as ideias discutidas em um relatório.

3.2 Ciclo da Aprendizagem Baseada em Problema

Inicialmente, o problema é definido e apresentado aos usuários. Os estudantes devem ler atentamente o problema e identificar os termos que são desconhecidos. Durante a primeira sessão, são idealizadas hipóteses a partir dos conhecimentos prévios dos membros do grupo. As hipóteses são analisadas e resumidas, descartando as que não são importantes. Em seguida, metas de aprendizagem são criadas pelo coordenador e direcionadas a cada membro. O relator deve ainda elaborar um relatório sobre tudo o que ocorre durante a sessão. É importante ressaltar que o objetivo da primeira sessão é apenas organizar e planejar o estudo, sendo papel do facilitador evitar que o grupo se desvie do ciclo. Portanto, os alunos não devem buscar novas informações durante as fases da primeira sessão.

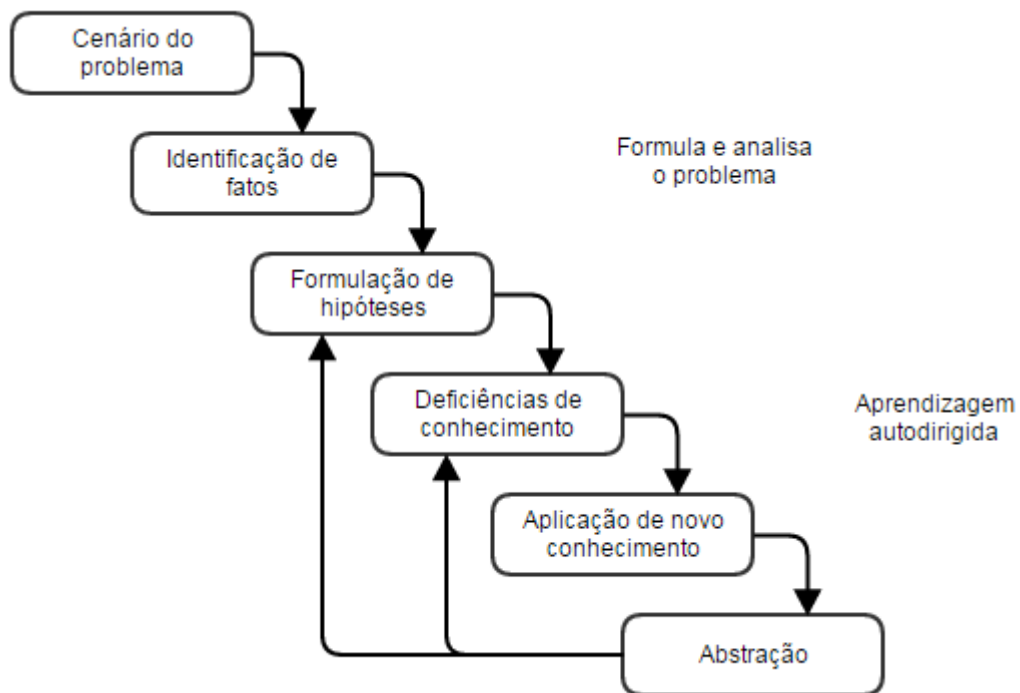
Após o término da primeira sessão, inicia-se a fase de estudo individual (autoaprendizagem). Nesse intervalo, o aluno deverá buscar novos conhecimentos com base nas metas de aprendizagem atribuídas na primeira sessão. Já o facilitador, deverá avaliar a participação dos alunos na sessão, inclusive a atuação do coordenador e do relator. Essa avaliação serve apenas como incentivo para que os alunos tenham um melhor proveito durante as sessões e não apresenta pontuação. O facilitador deve propor o que deve ser melhorado e o que deve ser mantido na atuação dos estudantes.

Na segunda sessão, a resolução do problema é discutida a partir do estudo individual de cada membro do grupo. O aprendizado deve ser compartilhado com todos os membros e, a partir do conhecimento adquirido, deve-se tentar resolver o problema.

Após a segunda sessão, o facilitador deve orientar os alunos a refletirem sobre o conhecimento absorvido. Caso o grupo não tenha atingido as expectativas de aprendizagem, o ciclo pode ser refeito inteiramente. No entanto, o facilitador pode encerrar o problema assim mesmo. Por fim, é realizada a avaliação de pares, onde cada membro do grupo avalia os demais membros e a si mesmo, destacando as deficiências encontradas e refletindo sobre o aprendizado do grupo e individual.

A Figura 1 ilustra o ciclo de desenvolvimento da PBL.

Figura 1 - O ciclo da PBL.



Fonte: Adaptado de (HMELO-SILVER, 2004).

4 AGENTES E SISTEMAS MULTIAGENTE

Ainda não se tem um consenso sobre a definição do que é um agente. No entanto, através das definições sugeridas por pesquisadores da área, é possível construir um conceito (ARTERO, 2009; FONTES, 2013; PONTES, 2010; SILVA, 2012).

Uma definição simples do que é um agente é sugerida por Russel e Norvig (2003), que diz que um agente é tudo o que pode ser considerado capaz de perceber seu ambiente por meio de sensores e de atuar sobre esse ambiente através de atuadores. Já Wooldrigde (2002), acrescenta que os agentes devem atuar de forma autônoma no ambiente na intenção de cumprir os objetivos para os quais foram projetados.

Outra definição, segundo Henderson-Sellers e Giorgini (2005), diz que agentes são entidades de software ou não que possuem autonomia na realização de suas tarefas, além de serem proativos e direcionados a objetivos.

Segundo Artero (2009), agentes inteligentes são programas que executam um conjunto de operações no lugar de um usuário, utilizando uma representação do conhecimento que contém os objetivos do usuário. Complementando, o autor ainda afirma que agentes são programas que realizam diálogos para negociar e coordenar transferências de informação.

4.1 Características

A seguir são descritas algumas características relacionadas a agentes encontradas em diversas definições (ARTERO, 2009; RUSSEL; NORVIG, 2003; WOOLDRIGDE, 2002):

- Autonomia: permite que o agente opere sem a intervenção direta dos usuários;
- Sociabilidade: capacidade de interagir com outros agentes a fim de realizar tarefas;
- Reatividade: refere-se à aptidão de análise do ambiente e resposta a suas alterações;
- Pró-atividade: tomada de iniciativa na realização de tarefas;
- Mobilidade: capacidade de se mover através da rede.

Além dessas características, para Russel e Norvig (2003), outra característica a ser levada em consideração é a racionalidade. Essa racionalidade é influenciada em quatro fatores (RUSSEL; NORVIG, 2003):

- Medida de desempenho, que é o fator que define o critério de sucesso;
- Conhecimento prévio que o agente possui;

- Ações que o agente pode realizar;
- Sequência de percepções captadas pelo agente até o momento.

Através desses fatores, a racionalidade permite ao agente maximizar sua medida de desempenho com base em uma sequência de percepções e em qualquer outro conhecimento interno do agente (RUSSEL; NORVIG, 2003).

4.2 Tipos De Agentes

Os agentes podem variar de acordo com o seu comportamento. A seguir são citados quatro tipos básicos de agentes (FONTES, 2013; RUSSEL; NORVIG, 2003; SILVA, 2012):

- Agentes reativos simples: é o tipo mais simples de agente. Selecionam ações com base apenas na percepção atual. Sua função interpreta a entrada resultando em um estado, associa esse estado a uma regra correspondente e realiza a ação com base na regra estabelecida. Em decorrência da simplicidade, tem a inteligência muito limitada e, com frequência, ocorrem laços de repetição infinitos quando operado em ambientes parcialmente observáveis. No entanto, é possível escapar de laços de repetição infinitos se o agente puder tornar algumas de suas decisões aleatórias;
- Agentes reativos baseados em modelos: este tipo de agente baseia-se no conhecimento de como o mundo funciona para realizar ações e evitar laços de repetição infinitos. Dessa forma, mesmo só observando o ambiente parcialmente, ele pode controlar a parte do ambiente que ainda não consegue enxergar, baseado num histórico de percepções. A partir do estado interno, que depende do histórico de percepções, o agente reflete alguns dos aspectos não observados do estado atual;
- Agentes baseados em objetivos: utiliza além de uma descrição do estado atual, alguma informação sobre os objetivos que descreve situações desejáveis. Combinando os resultados de ações possíveis com as informações sobre os objetivos, o agente pode escolher as ações que alcancem o objetivo desejado. O seu comportamento pode ser alterado com mais facilidade, visto que o conhecimento que apoia suas decisões é representado explicitamente. Assim, modificando o conhecimento, modifica o comportamento do agente;
- Agentes baseados na utilidade: se os agentes baseados em objetivos visam alcançar um objetivo desejado, os agentes baseados na utilidade visam chegar ao objetivo

desejado com mais eficiência, escolhendo os estados que tenham maior utilidade para o agente. Uma função de utilidade mapeia um estado (ou uma sequência de estados) em um valor, definindo um grau de “felicidade” do agente.

Esses tipos básicos de agentes também podem fazer uso de algoritmos de aprendizagem, melhorando, conseqüentemente, o seu desempenho (RUSSEL e NORVIG, 2003). Os agentes com aprendizado possuem uma grande adaptabilidade às mudanças do ambiente, em decorrência do aprendizado que o ambiente oferece (ARTERO, 2009; FONTES, 2013; SILVA, 2012).

4.3 Agentes Pedagógicos

Ao apoiar uma atividade educacional, os agentes são ditos pedagógicos. Sua principal finalidade é fornecer acesso personalizado ao estudante, atuando por conta própria ou com o auxílio dos professores. A combinação de agentes pedagógicos e AVAs gera uma abordagem promissora para o aprendizado eficaz auxiliado por computador (FONTES, 2013; SOLIMAN; GUETL, 2010).

Agentes pedagógicos atuam na supervisão do desempenho do estudante, motivando-o, detectando suas deficiências, fornecendo dicas e assessorando-o na resolução dos problemas (REATEGUI; MORAES, 2006).

São divididos em: orientados pela utilidade e dirigidos por objetivos. Os agentes orientados pela utilidade baseiam-se nas necessidades do estudante, ajudando-o na realização das tarefas. Já os agentes dirigidos por objetivos são mais autônomos. De acordo com seus objetivos, estabelecem metas e perspectivas, tomam decisões e atuam no ambiente pedagógico (SILVA; BERNARDI, 2009).

4.4 Sistemas Multiagente

Dado o conceito e as características dos agentes, pode-se definir o conceito de Sistema Multiagente (SMA) como um sistema constituído de dois ou mais agentes que competem ou cooperam entre si com o intuito de desempenhar tarefas ou atingir um ou mais objetivos (GIORGINI, 2005; HENDERSON-SELLERS; LESSER, 1999).

Uma das aplicações mais comuns para os SMAs é a de transformar grandes problemas em problemas menores, possibilitando que cada agente possa utilizar suas habilidades para

tratar os pequenos problemas, tornando o problema menos complexo (ARTERO, 2009; FONTES, 2013; SILVA, 2012).

Os SMAs também podem ser classificados quanto à arquitetura. Na abordagem denominada “Quadro Negro”, os agentes trabalham de forma autônoma sem a necessidade direta de comunicação entre eles. Toda a comunicação é realizada através de uma estrutura de dados central (quadro negro) que é compartilhada por todos os agentes. Assim, os agentes podem ler e escrever informações no quadro negro, compartilhando-as com outros agentes (ARTERO, 2009; FONTES, 2013; REIS, 2003; SILVA, 2012).

Analogamente, outra estratégia destaca a capacidade de interagir com outros agentes utilizando protocolos de interação social de forma a realizar uma comunicação direta. No entanto, a flexibilidade do SMA acaba sendo comprometida, pois os agentes são obrigados a se identificarem de forma precisa (ARTERO, 2009; FONTES, 2013; REIS, 2003; SILVA, 2012).

4.4.1 Organização dos Sistemas Multiagente

Os SMAs também possuem uma classificação. No entanto, é levado em conta como os agentes estão organizados no SMA. Os três tipos mais comuns são (ARTERO, 2009; FONTES, 2013; SILVA, 2012):

- Hierárquica: um agente superior controla e toma as decisões, repassando-as para os demais agentes, que pertencem a um nível hierárquico inferior;
- Comunidade de especialistas: todos os agentes estão em um mesmo nível, tendo cada agente uma especialidade em certo domínio. Nesse tipo de organização, a interação entre os agentes ocorre de acordo com regras previamente estabelecidas;
- Comunidade científica: os problemas são solucionados localmente e, em seguida, essas soluções são testadas e refinadas por agentes solucionadores de problemas.

Neste trabalho, optou-se pela organização de SMA comunidade de especialistas, onde cada agente é especializado em realizar determinada tarefa. A escolha se deu por este trabalho dar continuidade a um trabalho que já utilizava esse modelo organizacional.

5 MODELAGEM DE SMAS

Para contornar a complexidade associada ao desenvolvimento de SMAs, técnicas e ferramentas de engenharia de software podem ser aplicadas ao longo do projeto (FONTES, 2013; SILVA, 2012). Com isso, várias metodologias para a modelagem de SMAs foram propostas nos últimos anos. Para entender melhor cada uma dessas metodologias, é recomendado consultar as análises comparativas realizadas em (MORAIS II, 2010) e (PONTES, 2010).

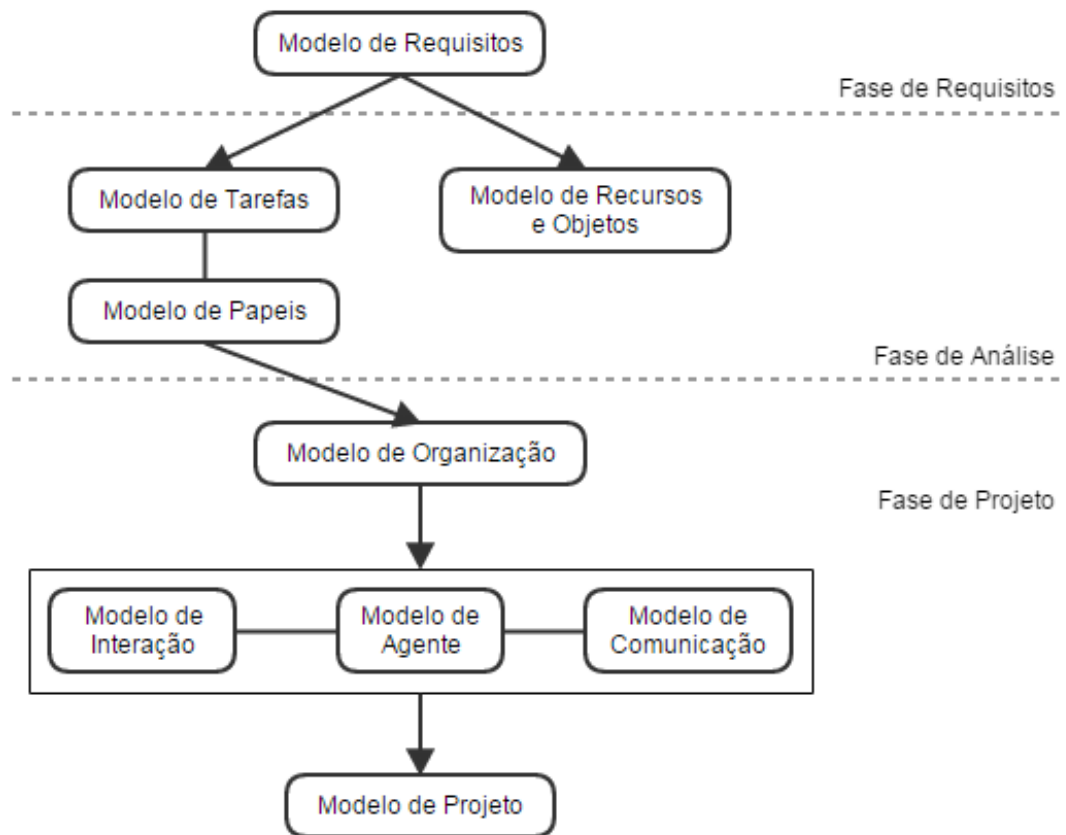
A metodologia escolhida para este trabalho foi a MAS-CommonKADS+, proposta em (MORAIS II, 2010). A escolha se deu analisando os principais pontos fortes e fracos de cada metodologia, principalmente através dos trabalhos apresentados em (MORAIS II, 2010) e em (PONTES, 2010). Na subseção a seguir, são apresentados detalhes dessa metodologia.

5.1 Metodologia MAS-CommonKADS+

A metodologia MAS-CommonKADS+ é uma extensão da metodologia MAS-CommonKADS. Ela mantém muitos dos modelos já propostos na metodologia MAS-CommonKADS, porém realiza algumas modificações e adiciona novos conceitos. Ao invés de definir sete modelos, a MAS-CommonKADS+ contém nove modelos. Esses modelos são representados através de notações da AML (*Agent Modeling Language*), que consiste em uma linguagem de modelagem especificada como uma extensão da UML 2.0, com o intuito de especificar, modelar e documentar SMAs (FONTES, 2013; SILVA, 2012). Foram adicionados à metodologia os modelos de requisitos, de papéis e de recursos, enquanto que os modelos de organização, de interação e de projeto foram alterados, com o intuito de complementar a especificação dos diagramas da AML.

O modelo de agentes foi o que sofreu mais alterações, possibilitando agora demonstrar como o agente irá perceber e atuar no ambiente de acordo com seus comportamentos e planos. A Figura 3 mostra a arquitetura da MAS-CommonKADS+ (FONTES, 2013; SILVA, 2012).

Figura 2 - Modelos da metodologia MAS-CommonKADS+.



Fonte: Adaptado de (MORAIS II, 2010).

O modelo de requisitos é utilizado para descrever os requisitos do sistema, sendo dividido em análise de casos de uso e cenários, análise por objetivos e análise do ambiente.

O modelo de tarefas descreve todas as atividades (chamadas de tarefas) que devem ser realizadas com o intuito de alcançar determinada meta. As tarefas e subtarefas são demonstradas em um diagrama de tarefas, seguindo uma abordagem *top-down*. Também são utilizados *templates* textuais para descrever as entradas, saídas, pré-condições, pós-condições e meta de cada uma das tarefas.

O modelo de papéis tem como objetivo a identificação dos papéis do sistema e a representação de papéis que realizam as tarefas descritas no modelo de tarefas. Um papel é uma abstração que define as tarefas que um agente deve realizar dentro de uma organização. Um único agente pode ser responsável por vários papéis em um sistema. Logo, a inclusão deste modelo é de extrema relevância, pois permite modelar os vários papéis que um agente pode exercer, o que não é possível ser realizado na MAS-CommonKADS.

O modelo de recursos e objetos foi adicionado com o intuito de possibilitar a modelagem de objetos e recursos, permitindo, assim, uma melhor definição do sistema, visto que não era algo possível de ser modelado na MAS-CommonKADS.

O modelo de organização descreve a estrutura organizacional de papéis do sistema, e não mais a organização de agentes, como acontecia na MAS-CommonKADS.

O modelo de interação consiste na junção dos modelos de coordenação e de comunicação da MAS-CommonKADS. Nele, são descritas, através da AML, todas as interações entre agentes. Cada interação deve obedecer a um protocolo de interação, o qual estabelece como os agentes podem se comunicar.

O modelo de agentes especifica os agentes, por quais papéis eles são responsáveis, as percepções, os atuadores, as condições de ativação e de parada, e as características do agente. A primeira etapa para construção desse modelo é descrever os agentes do sistema, identificando-os e definindo os seus papéis. Em seguida, é realizada a identificação dos comportamentos internos do agente de acordo com suas características.

O modelo de comunicação descreve as interações entre um agente humano e um agente de software.

O modelo de projeto descreve as características do local onde o sistema será instalado, os diagramas de implantação e informações a respeito da mobilidade dos agentes. Este modelo tem o intuito de facilitar o entendimento da infraestrutura, além de trazer informações sobre como componentes do SMA podem ser localizados, distribuídos e conectados.

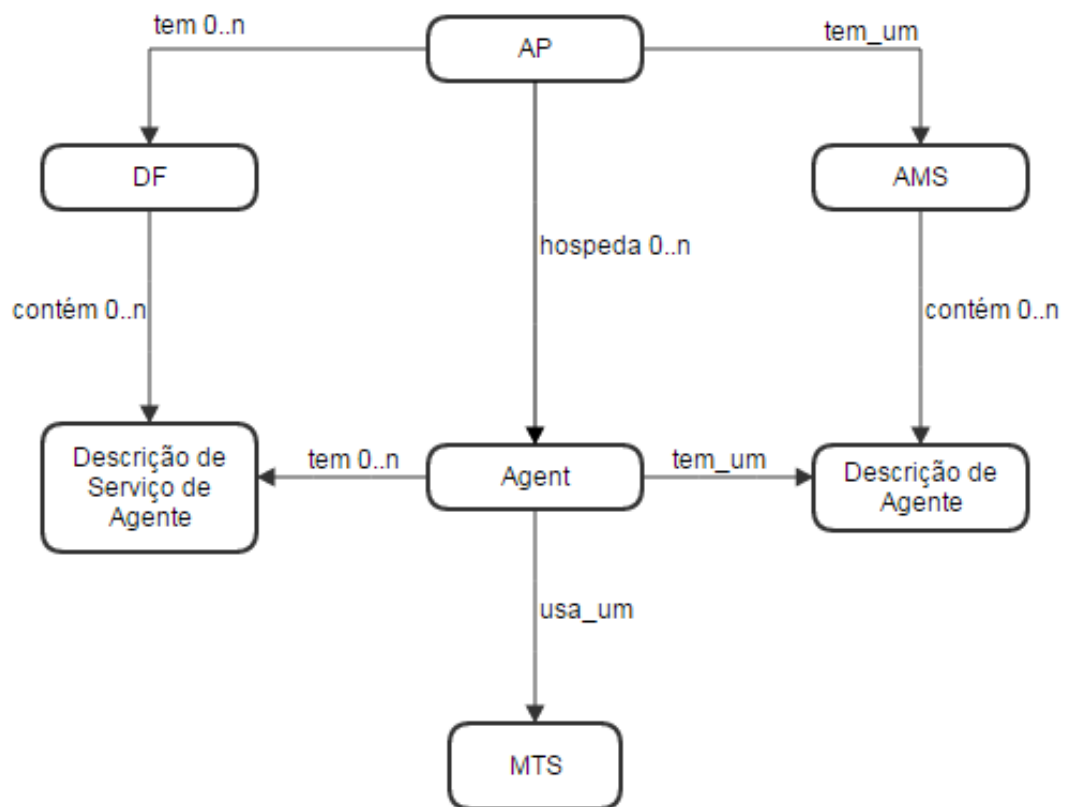
5.2 Comunicação e Gerenciamento de Agentes

Um dos componentes chaves de um SMA é a comunicação entre os agentes. Na realidade, agentes precisam se comunicar para que sejam capazes de cooperar, colaborar e negociar entre si. De uma forma geral, os agentes interagem entre si através de algumas linguagens de comunicação especiais, chamadas linguagens de comunicação entre agentes.

Atualmente, a linguagem de comunicação entre agentes mais difundida e utilizada é a FIPA (*Foundation for Intelligent Physical Agents*) ACL (*Agent Communication Language*). As principais características da FIPA ACL são a possibilidade de utilizar linguagens de conteúdos diferentes e o gerenciamento de conversações através de protocolos de interação predefinidos (BELIFEMINE; CAIRE; GREENWOOD, 2007; FONTES, 2013; SILVA, 2012).

FIPA é um conjunto de padrões da IEEE (*Institute of Electrical and Electronics Engineers*) cujo objetivo é promover (i) as tecnologias baseadas em agentes, (ii) a interoperabilidade desses padrões com outras tecnologias, (iii) a interoperação de agentes heterogêneos e (iv) os serviços que eles podem representar (FIPA, 2011; FONTES, 2013; SILVA, 2012). Além de prover um padrão e uma linguagem comum através da qual os agentes podem se comunicar, FIPA define um modelo lógico de referência para gerenciamento dos agentes. Desta forma, as plataformas que atendem à sua especificação devem implementar esse modelo, possibilitando, assim, a criação, o registro, a localização, a comunicação, a migração e a operação dos agentes (BELIFEMINE; CAIRE; GREENWOOD, 2007; FONTES, 2013; SILVA, 2012). A Figura 4 mostra os componentes desse padrão.

Figura 3 - Componentes do Padrão FIPA.



Fonte: Adaptado de (BELIFEMINE; CAIRE; GREENWOOD, 2007).

Como pode ser visto na Figura 4, os seguintes componentes podem ser visualizados nesse padrão (FONTES, 2013; SILVA, 2012):

- *Agent Platform* (AP): provê a infraestrutura física mínima na qual os agentes são executados. Consiste de máquinas, sistemas operacionais, componentes para

gerenciamento de agentes FIPA (descritos a seguir), os agentes em si e qualquer software de suporte adicional;

- *Agent Management System* (AMS): componente obrigatório de uma AP responsável por gerenciar as operações desta, tais como criação e destruição de agentes, e por fiscalizar as migrações dos agentes entre APs. Em cada AP, existe apenas um AMS e, caso a AP seja composta de várias máquinas, o AMS será a autoridade máxima entre todas elas. Cada agente deve se registrar com o AMS para que possa obter um AID (*Agent Identifier*), identificador que o identifica de forma única dentro da AP. O AMS mantém, então, um diretório contendo todos os AIDs e os respectivos estados dos agentes (ex. ativo, suspenso ou em espera), serviço este chamado de páginas brancas;
- *Agent*: processo computacional que reside em uma AP e oferece um ou mais serviços computacionais, que podem ser publicados como uma descrição de serviço. Cada agente deve ter, obrigatoriamente, um AID;
- *Directory Facilitator* (DF): componente opcional de uma AP que provê, para outros agentes, o serviço de páginas amarelas. Esse serviço consiste em uma lista de todos os agentes e os respectivos serviços oferecidos por cada um destes. Todo e qualquer agente que deseje publicar seus serviços para outros agentes deve encontrar um DF apropriado e requisitar o registro da sua descrição de serviço. Além disso, os agentes podem retirar e modificar o seu próprio registro de um DF e buscar neste, de acordo com um critério de busca, por um registro de serviço fornecido por outro agente. Esse componente é de extrema importância para o SMA proposto neste trabalho;
- *Message Transport Service* (MTS): também conhecido como ACC (*Agent Communication Channel*), o MTS é o serviço provido por uma AP para transportar mensagens FIPA-ACL entre os agentes em uma determinada AP e entre agentes localizados em APs distintas.

6 ALGORITMOS GENÉTICOS

Os Algoritmos Genéticos (AGs) correspondem a uma categoria de algoritmos de busca e otimização baseada nos algoritmos evolucionários. Tem como principal finalidade resolver problemas para os quais não existe um algoritmo conhecido. (ARTERO, 2009; FONTES, 2013; SILVA, 2012).

Os AGs, assim como os demais tipos de algoritmos evolucionários, não podem garantir que uma geração será melhor do que a sua geração antecessora, pois não consiste em um processo dirigido à obtenção da solução ótima. O processo simplesmente consiste em fazer competir um conjunto de indivíduos, fazendo com que sobrevivam aqueles indivíduos que são mais aptos (LINDEN, 2008; PETROLI NETO, 2011).

6.1 Conceitos Básicos dos Algoritmos Genéticos

Para que se possa entender melhor o funcionamento de um AG é necessário compreender alguns conceitos. Nas subseções a seguir, serão descritos os principais conceitos relacionados aos AGs.

6.1.1 Genes

Em AGs, os genes correspondem a uma representação de algum parâmetro de interesse, seguindo algum alfabeto pré-estabelecido, podendo ser representados por valores inteiros, reais e cadeias de caracteres (ARTERO, 2009). Normalmente se utiliza os valores 0 e 1 para representar cada gene (ARTERO, 2009; FONTES, 2013; PETROLI NETO, 2011; SILVA, 2012).

6.1.2 Cromossomos

Os cromossomos são compostos por uma cadeia de genes e representam os indivíduos da população, que representam as soluções encontradas em um problema de otimização (ARTERO, 2009).

6.1.3 População

É formada por um conjunto de soluções (cromossomos) que irão competir para sobreviver e reproduzir, perpetuando suas características para as gerações seguintes (ARTERO, 2009; FONTES, 2013; SILVA, 2012).

6.1.4 Geração

Representa os valores dos indivíduos obtidos após um determinado número de iterações (ARTERO, 2009; FONTES, 2013; SILVA, 2012).

6.1.5 Função de Aptidão

Mede a habilidade do indivíduo para sobreviver e se reproduzir (ARTERO, 2009). É fundamental para o funcionamento correto de um AG, pois ela faz a ligação do algoritmo com o problema real (PETROLI NETO, 2011).

6.2 Funcionamento dos Algoritmos Genéticos

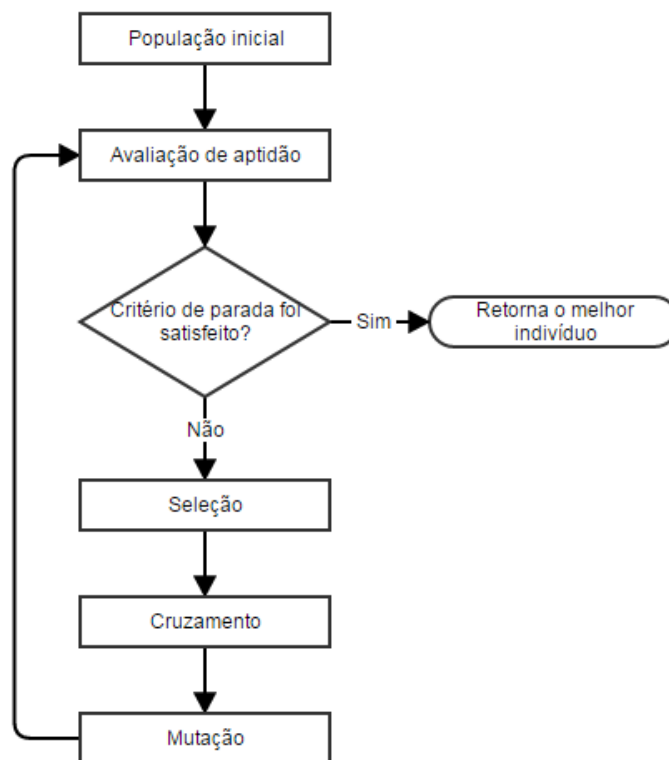
Com base nos conceitos vistos anteriormente, um AG pode ser descrito da seguinte forma (FONTES, 2013; LINDEN, 2008; PETROLI NETO, 2011; SILVA, 2012):

1. Uma população é inicializada aleatoriamente;
2. Avalia-se cada um dos cromossomos na população através da função de aptidão, de modo a encontrar uma classificação dos indivíduos mais aptos a sobreviver;
3. A partir de uma estratégia de seleção previamente definida, são escolhidos os indivíduos que servirão como pais para criação de uma nova população;
4. São aplicadas as operações de cruzamento e mutação aos indivíduos para gerar os indivíduos da nova população;
5. Eliminam-se os cromossomos da população antiga, de forma que os novos cromossomos gerados possam ser inseridos sem alterar o tamanho da população inicial;
6. A função de aptidão é aplicada a todos os novos cromossomos, selecionando os melhores e inserindo-os na população anterior, gerando uma nova população;

7. Se o resultado da população atual representar o resultado esperado ou se a quantidade de gerações atingiu o limite ou ainda se o algoritmo não conseguir mais mostrar evolução, a execução deve parar. Do contrário, deve-se reiniciar o algoritmo a partir do passo 3.

Concluída a execução do AG, espera-se que a população de cromossomos gerada seja a que melhor represente o resultado do problema. A Figura 4 apresenta a estrutura básica de um Algoritmo Genético.

Figura 4 - Estrutura básica de um algoritmo genético.



Fonte: Adaptado de (ZINI, 2009).

Porém, ainda é necessário compreender aspectos mais complexos que devem ser tratados, tais como: a escolha de uma codificação dos cromossomos adequada; definir o tamanho da população; definir como será realizada a mutação; e selecionar uma função de aptidão adequada ao problema (LINDEN, 2008; PETROLI NETO, 2011; SILVA, 2012).

Para uma melhor compreensão do funcionamento dos AGs, recomenda-se a leitura de (SILVA, 2012), onde cada um desses aspectos, bem como suas características, são descritos de forma detalhada.

7 LÓGICA FUZZY

A lógica convencional trata de distinguir se um elemento pertence ou não a um conjunto. Por outro lado, a lógica fuzzy utiliza um parâmetro, denominado grau de pertinência, que descreve numericamente quão próximo um elemento está de pertencer a um conjunto. Dessa forma, a premissa pode ser verdadeira, falsa, parcialmente verdadeira ou parcialmente falsa (MARRO et al., 2010).

7.1 Conjuntos Fuzzy

Na lógica convencional, os elementos são classificados utilizando sistemas binários. Nessa abordagem, um elemento qualquer possui apenas duas possibilidades: pertencer ou não a um dado conjunto. Em adição, para uma determinada premissa, se um elemento pertence a um dado conjunto, ele não poderá pertencer a nenhum outro. A lógica fuzzy, no entanto, estabelece a classificação dos conjuntos de acordo com o grau de pertinência do elemento que pode variar de 0 a 1, tal que um representa total pertinência e zero a total exclusão. Assim, um elemento pode assumir sua pertinência a mais de um conjunto, semelhante ao funcionamento do pensamento humano (GOMIDE; GUDWIN, 1994; IBRAHIM, 2003; MARRO et al., 2010).

Para representar um conjunto fuzzy, é necessário primeiro definir uma função de pertinência adequada ao problema. Além disso, deve-se definir também as variáveis linguísticas, que irão representar os elementos, e os valores que essas variáveis poderão alcançar, bem como os modificadores que modificam a intensidade dos valores fuzzy. Por exemplo: “João é muito alto”, onde: “João” é a variável linguística e o modificador “muito” intensifica o valor “alto” (MARRO et al., 2010).

7.2 Regras Fuzzy

As regras fuzzy são uma forma de expressar um determinado conhecimento. A maneira mais comum de expressar esses conhecimentos é através de regras do tipo condição-ação, onde um conjunto de condições é associado a uma ação. Como, por exemplo, se o aluno possui conhecimento avançado sobre o assunto, então ele é adequado para resolver o problema (GOMIDE; GUDWIN, 1994).

7.3 Inferência Fuzzy

A inferência fuzzy é um processo que avalia as entradas com o objetivo de obter conclusões com base nas regras pré-estabelecidas. É comum utilizar modelos de inferência de acordo com o problema a ser aplicado. Um dos modelos mais utilizados é o de Mamdani que tem como intuito de representar experiências da vida real (MARRO et al., 2010). Esse modelo utiliza o operador de agregação, e os operadores de máximo e mínimo correspondem às operações de união e intercessão dos conjuntos. Um exemplo de regra baseado nesse modelo pode ser visto a seguir:

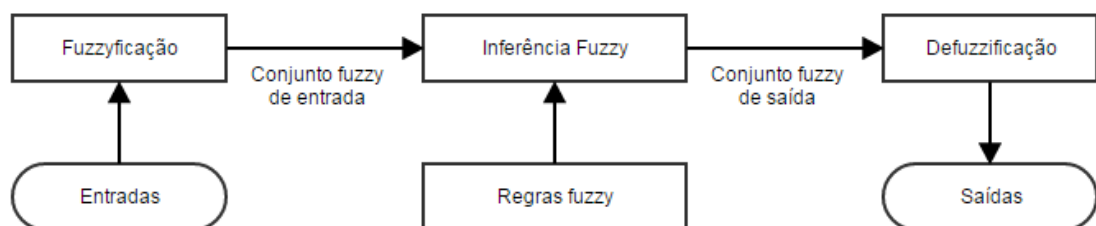
$$\text{SE } a = x \text{ E } b = y \text{ ENTÃO } c = z$$

7.4 Fuzzificação e Defuzzificação

A fuzzificação é uma operação que transforma um conjunto definido em um conjunto difuso. Ou seja, traduz a entrada, que pode ser um valor ou uma medida, em conceitos linguísticos. Ou ainda, mapeia os números do domínio real para o domínio fuzzy. Analogamente, a defuzzificação produz um efeito inverso ao da fuzzificação, transformando um conceito linguístico em um valor ou medida. Ou seja, a variável no domínio fuzzy é mapeada para uma variável no domínio real (GABRIEL, 2008; IBRAHIM, 2003).

A Figura 5 representa a estrutura básica de um controlador baseado em lógica fuzzy.

Figura 5 - Estrutura básica de um controlador baseado em lógica fuzzy.



Fonte: Adaptado de (IBRAHIM, 2003).

8 SISTEMA MULTIAGENTE DE APOIO À PBL

8.1 Ferramentas Utilizadas

8.1.1 Moodle

O MOODLE é um ambiente de aprendizagem robusto, distribuído de forma gratuita sob a licença GNU (*General Public License*). Possui uma comunidade ativa e global, tendo traduções para diversas línguas e uma grande quantidade de usuários, tanto do meio acadêmico quanto no meio empresarial. A documentação é simples e de fácil acesso, e seu código fonte aberto permite modificá-lo da maneira que o usuário desejar, além de disponibilizar diversos módulos. Segundo o seu site oficial, existem 63.285 sites registrados no mundo utilizando o MOODLE, sendo que desses, 3.830 são no Brasil (MOODLE, 2015).

8.1.2 Netbeans

O NetBeans é um ambiente de desenvolvimento integrado que promove o desenvolvimento rápido em diversas linguagens de programação. É uma ferramenta gratuita e de código aberto, e possui uma vasta opção de configurações e *plug-ins* (NETBEANS, 2015). Neste trabalho, o NetBeans foi utilizado para desenvolver o módulo para o MOODLE, utilizando a linguagem PHP (*Hypertext Preprocessor*) (PHP, 2015), e o SMA em Java (ORACLE, 2016).

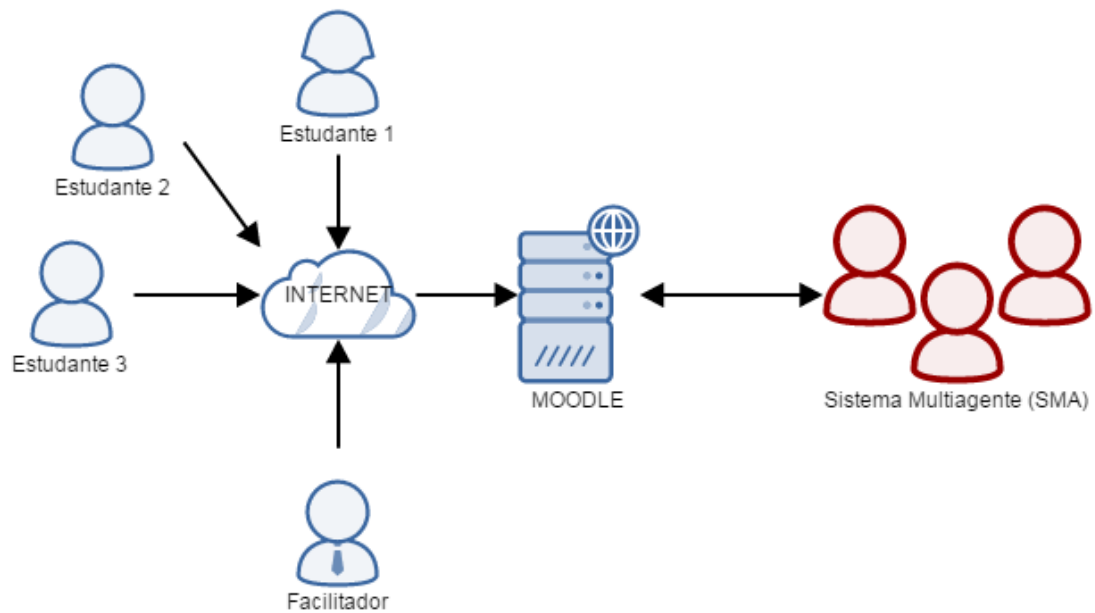
8.1.3 Jade

O JADE (*Java Agent DEvelopment Framework*) é um *framework* totalmente implementado em Java que simplifica a criação de SMAs seguindo as especificações do padrão FIPA. É distribuído de forma gratuita sob licença LGPL (*Lesser General Public License*) e tem seu código aberto (JADE, 2014).

8.2 Abordagem Multiagente de Apoio à PBL na EaD

Para que pudesse auxiliar a implantação da PBL em um AVA, foi proposta, neste trabalho, a abordagem multiagente representada na Figura 4.

Figura 6 - Abordagem multiagente de apoio à PBL na EaD.



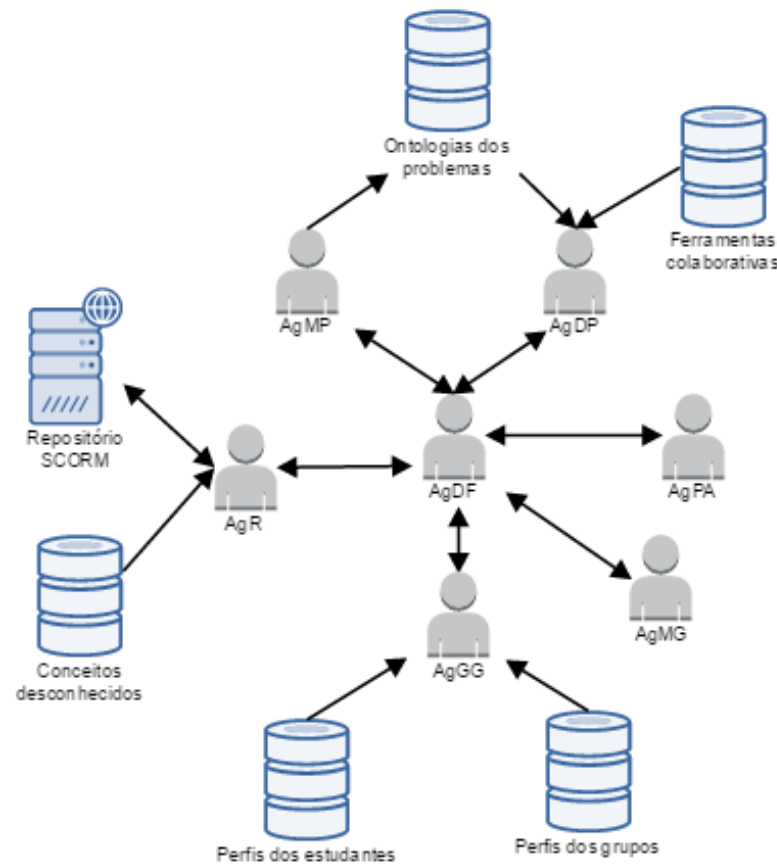
Fonte: Dados produzidos pelo autor.

Como pode ser visto na Figura 4, os estudantes e o facilitador acessam o MOODLE através da Internet. Por meio de um módulo, os usuários terão acesso às ferramentas e interfaces criadas especificamente para atender às necessidades da PBL.

O SMA atua de forma que os estudantes e o facilitador não percebam suas ações, executando tarefas que visam auxiliar os estudantes e o facilitador durante todo o ciclo da PBL.

Para atender às necessidades da PBL, Fontes (2013) propôs um SMA do tipo comunidade de especialistas, composto de sete agentes: Agente Pedagógico Animado (AgPA), Agente Monitorador de Problemas (AgMP), Agente Detector de Problemas (AgDP), Agente Monitorador de Grupos (AgMG), Agente Gerenciador de Grupos (AgGG), Agente recomendador (AgR) e Agente DF (*Directory Facilitator*), como ilustra a Figura 5.

Figura 7 - Esquema do sistema multiagente proposto por Fontes (2013).

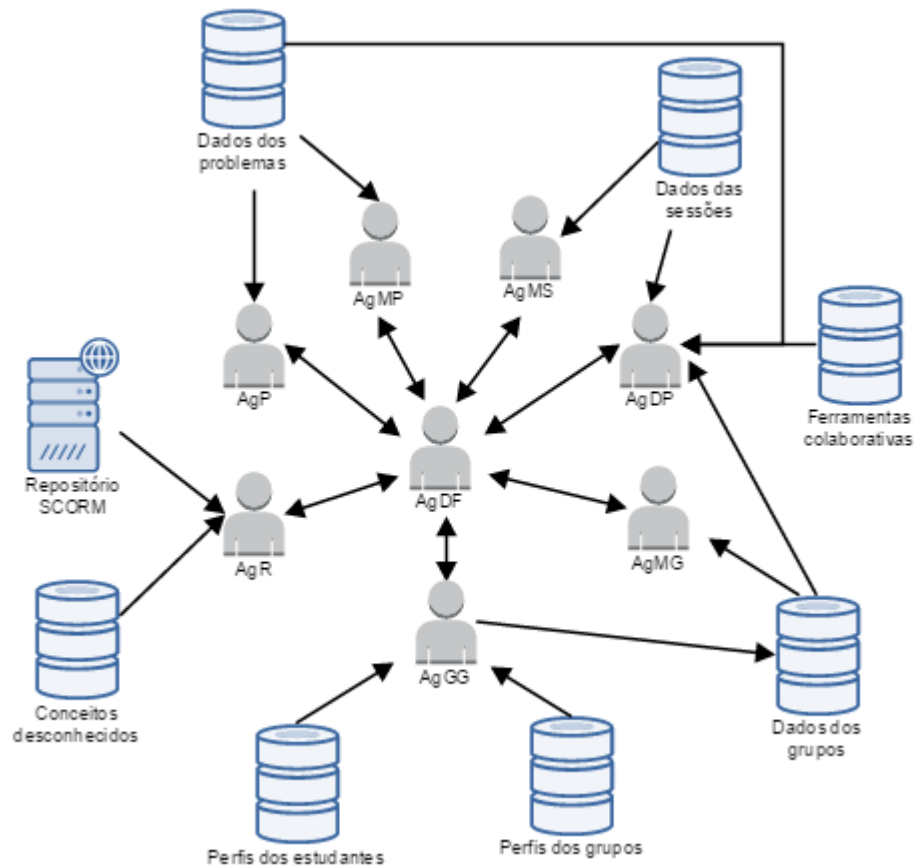


Fonte: Adaptado de (FONTES, 2013).

Visando otimizar as tarefas dos agentes, algumas alterações foram realizadas no SMA proposto por Fontes (2013). Foram acrescentados dois novos agentes: o Agente Problema, que tem por intuito encontrar novos problemas para serem monitorados, e o Agente Monitorador de Sessões, que visa monitorar as sessões criadas pelo facilitador. Além disso, foi removido o Agente Pedagógico Animado devido à incompatibilidade entre a tecnologia usada para desenvolvê-lo e os *browsers* atuais.

Assim sendo, o SMA desenvolvido neste trabalho é composto por oito agentes: Agente Problema (AgP), Agente Monitorador de Problema (AgMP), Agente Detector de Problemas (AgDP), Agente Monitorador de Grupo (AgMG), Agente Gerenciador de Grupos (AgGG), Agente Recomendador (AgR), Agente Monitorador de Sessões (AgMS) e Agente DF (*Directory Facilitator*). Este SMA é ilustrado na Figura 6.

Figura 8 - Esquema do sistema multiagente proposto neste trabalho.



Fonte: Dados produzidos pelo autor.

A organização do SMA é do tipo comunidade de especialistas, pois cada um dos oito agentes criados para este trabalho encontra-se no mesmo nível, sendo cada um deles especialista em determinada tarefa.

Os principais componentes dessa abordagem são:

- SMA: é o sistema multiagente criado para auxiliar na implantação da PBL no MOODLE;
- AVA: é o ambiente virtual de aprendizagem na qual a metodologia será implantada. Para a realização deste trabalho, foi escolhido o AVAMOODLE, que é um ambiente de código-fonte aberto, sendo este muito utilizado para a realização de cursos a distância (MOODLE, 2011);
- Perfis dos estudantes: refere-se à base de perfis dos estudantes. Os estudantes preenchem seus respectivos perfis, via interface do MOODLE, através do módulo de apoio à PBL, na interface “Meu perfil”. Os perfis são incrementados após a resolução de um problema, mediante avaliação do facilitador. Os perfis dos estudantes são

consultados no processo de formação de grupos e recomendação de Objetos de Aprendizagem (OAs);

- Perfis dos grupos: referem-se às características requisitadas para a resolução de cada problema. Esses perfis são criados pelo facilitador, via interface do MOODLE, através do módulo de apoio à PBL, na interface de “*Visualização do problema*”. Os perfis dos grupos são consultados no processo de formação de grupos;
- Dados dos problemas: são os dados armazenados que contêm as informações referentes aos problemas propostos durante a aplicação da PBL. O facilitador é responsável por criar e definir esses problemas. Algumas das informações que podem ser encontradas nessa base são: o título do problema, a definição do problema, as palavras relacionadas ao problema, as metas de aprendizagem, as áreas de conhecimento que o problema aborda, dentre outras;
- Dados dos grupos: são os dados que correspondem às informações dos grupos, como, por exemplo, se o grupo está ou não vinculado ao problema, se o problema já foi finalizado para esse grupo, os seus respectivos membros e o nome do grupo;
- Dados das sessões: são as informações relacionadas às sessões. Essas informações são fundamentais para a análise do ciclo da PBL e correspondem a data de início da sessão, o estado finalizado ou não finalizado, se é a última sessão para o grupo, o relatório e a avaliação da sessão;
- Ferramentas colaborativas: são as ferramentas usadas pelos estudantes e facilitador para colaboração e comunicação durante a resolução do problema. As informações referentes a essas interações são armazenadas na base de dados e consultadas durante a detecção de problemas. As duas ferramentas utilizadas nesta proposta foram o *chat* e o fórum;
- Conceitos desconhecidos: referem-se aos conceitos que os estudantes desconhecem na definição do problema proposto. Esses conceitos desconhecidos são fornecidos pelo próprio estudante, via interface do MOODLE, através do módulo de apoio à PBL, na interface de “*Visualização de sessão*”. Essas informações, dentre outras, são consideradas durante a recomendação de OAs e devem ser atualizadas pelos estudantes a cada sessão.

8.3 Descrição do Sistema Multiagente

Os agentes interagem entre si através de um protocolo de comunicação previamente estabelecido. Para o desenvolvimento dos agentes, foram utilizadas as bibliotecas do JADE. Nas subseções a seguir, serão descritos cada agente e suas funcionalidades.

8.3.1 Agente Problema - AgP

O AgP tem como objetivo identificar novos problemas inseridos pelo facilitador passíveis a serem monitorados e gerenciados pelo SMA. Ao encontrar um problema, o AgP solicita que o AgMP monitore o problema encontrado. Apenas um agente desse tipo é criado no SMA.

8.3.2 Agente Monitorador de Problemas - AgMP

O AgMP é responsável por monitorar um problema, identificando suas necessidades iniciais e verificando se existem grupos vinculados que possam resolvê-lo. Cada vez que um novo problema é encontrado pelo AgP, um novo AgMP é instanciado.

Ao ser instanciado pelo AgP, o AgMP solicita a formação de grupos ao AgGG. Em seguida, executa um comportamento cíclico que verifica se existem grupos vinculados ao problema que ainda não estão sendo monitorado pelo AgMG e que ainda não finalizaram suas atividades. Se algum grupo for encontrado, é solicitado o monitoramento desse grupo. Caso todos os grupos vinculados tenham finalizado suas tarefas, o AgMP termina suas atividades.

8.3.3 Agente Gerenciador de Grupos - AgGG

O AgGG tem como função principal formar grupos para resolver problemas específicos.

O processo de formação de grupos é efetuado da seguinte forma: os estudantes preenchem seus respectivos perfis, via interface do MOODLE, que alimentam uma base de perfis que será usada no processo de formação de grupos. Além disso, cada vez que um estudante participa da resolução de um problema, seu perfil é atualizado de acordo com a avaliação final realizada pelo facilitador que, por sua vez, utiliza as metas de aprendizagens definidas na descrição do problema. O perfil dos estudantes é composto por conhecimentos,

onde cada um possui um nível, que pode ser baixo, médio ou alto, podendo um estudante ter um ou mais conhecimentos.

Durante a definição do problema, o facilitador preenche os perfis dos grupos desejados para cada problema, adicionando requisitos através de uma interface Web, de forma semelhante aos estudantes. Um perfil de grupo é composto por conhecimentos, cada um possuindo um nível, que pode ser baixo, médio ou alto, bem como um valor fuzzy, que varia de 0.1 a 1.0 e que define o grau de importância do requisito. Após o facilitador construir os perfis dos grupos, haverá uma base de perfis de grupos que será consultada pelo AgGG no processo de formação de grupos. É importante salientar que um estudante que tenha um perfil aproximado ao desejado terá aptidão maior para atingir as metas de aprendizagem.

A formação de grupos pode ser dividida em duas etapas:

1. A primeira etapa consiste em ranquear os alunos, dando um valor a cada um destes, que diz o quanto as características deste aluno correspondem ao que foi requisitado pelo facilitador. Para isso, um trecho implementado em Java (VERONESE et al., 2002) analisa os perfis dos estudantes e os perfis dos grupos. Após essa análise, ele gera um arquivo. A partir desse arquivo, um algoritmo implementado em Prolog (GOMES et al., 2002) realiza o processo de fuzzificação, que dará o valor correspondente a cada estudante. O código fonte do trecho em Prolog pode ser visto no Apêndice A;
2. A segunda etapa é um AG que tem como objetivo formar grupos que possuam quantidades niveladas de estudantes mais ou menos aptos. O AG leva em consideração o *ranking* gerado pela etapa anterior, distribuindo os estudantes em grupos, de forma que, ao final da sua execução, os grupos possuam a média dos valores dos seus membros semelhantes.

Lembrando que os grupos formados serão apenas indicados ao facilitador, que pode ou não aceitar a sugestão do SMA. Para que o grupo possa ser aceito, basta que o facilitador vincule o grupo ao problema, via interface do MOODLE, através da tela “*Visualização de Grupos*”, no módulo de apoio à PBL.

8.3.4 Agente Monitorador de Grupos - AgMG

O AgMG tem como objetivo monitorar um grupo vinculado ao problema. É instanciado um agente desse tipo para cada grupo vinculado ao problema que não tenha tido

suas atividades finalizadas. Seu papel é identificar se o grupo iniciou e finalizou suas atividades corretamente, monitorando as atividades realizadas pelos membros do grupo.

8.3.5 Agente Monitorador de Sessões - AgMS

O AgMS tem como propósito acompanhar os estudantes e o facilitador durante as sessões, identificando o estado de cada sessão.

Seus comportamentos consistem em verificar, caso exista alguma sessão marcada para o grupo, se a sessão está em andamento ou se já foi finalizada. Se existir uma sessão que já foi finalizada, solicita ao AgDP que seja realizada a detecção de problemas na sessão e, se essa sessão ainda não for a última, solicita ao AgR a recomendação de OAs.

8.3.6 Agente Detector de Problemas - AgDP

O AgDP possui dois comportamentos principais: um que auxilia na detecção de problemas de colaboração em sessões; e outro que detecta problemas de colaboração na finalização do problema.

O primeiro comportamento é acionado quando solicitado pelo AgMS e é responsável por detectar estudantes passivos, detectar conversações fora do contexto do problema, detectar se o relator enviou o relatório da sessão e detectar se o facilitador já realizou a avaliação da sessão. Esse comportamento é executado quando uma sessão é finalizada. Esse comportamento foi criado com o intuito de auxiliar o facilitador na avaliação do comportamento dos estudantes durante o processo de aplicação da PBL. Desta forma, uma vez sendo detectado um problema, o AgDP irá notificar o facilitador e os estudantes sobre o problema encontrado.

O segundo comportamento é acionado pelo AgMG quando a última sessão é finalizada para um determinado grupo, e permanece ativo até que o problema seja finalizado pelo grupo. Sua função é identificar se o relatório final, a solução e a avaliação de pares foram submetidos pelos estudantes e se o facilitador submeteu a avaliação final de todos os estudantes. De forma similar ao primeiro comportamento, o AgDP envia notificações para os estudantes e para o facilitador, caso algum problema seja identificado.

8.3.7 Agente Recomendador - AgR

O AgR é acionado na fase de autoaprendizagem, ou seja, sempre que uma sessão é finalizada, com exceção da última sessão. Esse agente tem o papel de recomendar OAs, baseando-se nos dados do problema e nos conceitos que os estudantes desconhecem na descrição do problema.

O AgR tem o intuito de detectar OAs adequados ao contexto do estudante, levando em consideração: (i) as informações providas pelos seus respectivos perfis; (ii) os assuntos que os estudantes desconhecem; e (iii) as informações dos OAs disponíveis no repositório de OAs SCORM (*Sharable Content Object Reference Model*) (SCORM, 2013).

O AgR utiliza um AG para identificar os OAs a serem recomendados. O AG considera para recomendação de OAs: (i) a incidência das palavras-chave na descrição do curso do qual o estudante está participando; (ii) a incidência das áreas de interesse do estudante nas palavras-chave do OA; (iii) o horário de estudo preferido do estudante; e (iv) a incidência dos conceitos desconhecidos na descrição, no título e nas palavras-chave do OA. Tal recomendação poderá auxiliar os estudantes a sanar as possíveis dúvidas durante a fase de identificação dos fatos.

A utilização do AG é justificada pela complexidade do problema de recomendação, o que pode depreciar o desempenho do AgR quando o número de OAs for suficientemente grande. Portanto, a aplicação de um algoritmo exato seria inviável. Neste caso, um algoritmo aproximativo viabiliza a busca por uma solução próxima da ótima.

8.3.8 Agente Directory Facilitator - DF

O papel de mediador da comunicação entre os agentes é realizado pelo agente DF, o qual é provido pela própria plataforma JADE, conforme exigência da especificação FIPA (FIPA, 2011). No sistema multiagente proposto, foi necessário apenas codificar a forma como os agentes criados se comunicam com o DF.

A principal vantagem em utilizar o JADE reside no fato de que é possível utilizar todos os componentes da especificação FIPA, visto que o JADE atende todas as especificações deste padrão e ainda o estende, sendo que um desses componentes é o agente DF (FONTES, 2013; SILVA, 2012).

Outra vantagem dessa plataforma consiste na possibilidade de consultar e utilizar, através do MTS (*Message Transport Service*), DFs que estejam sendo executados em outras

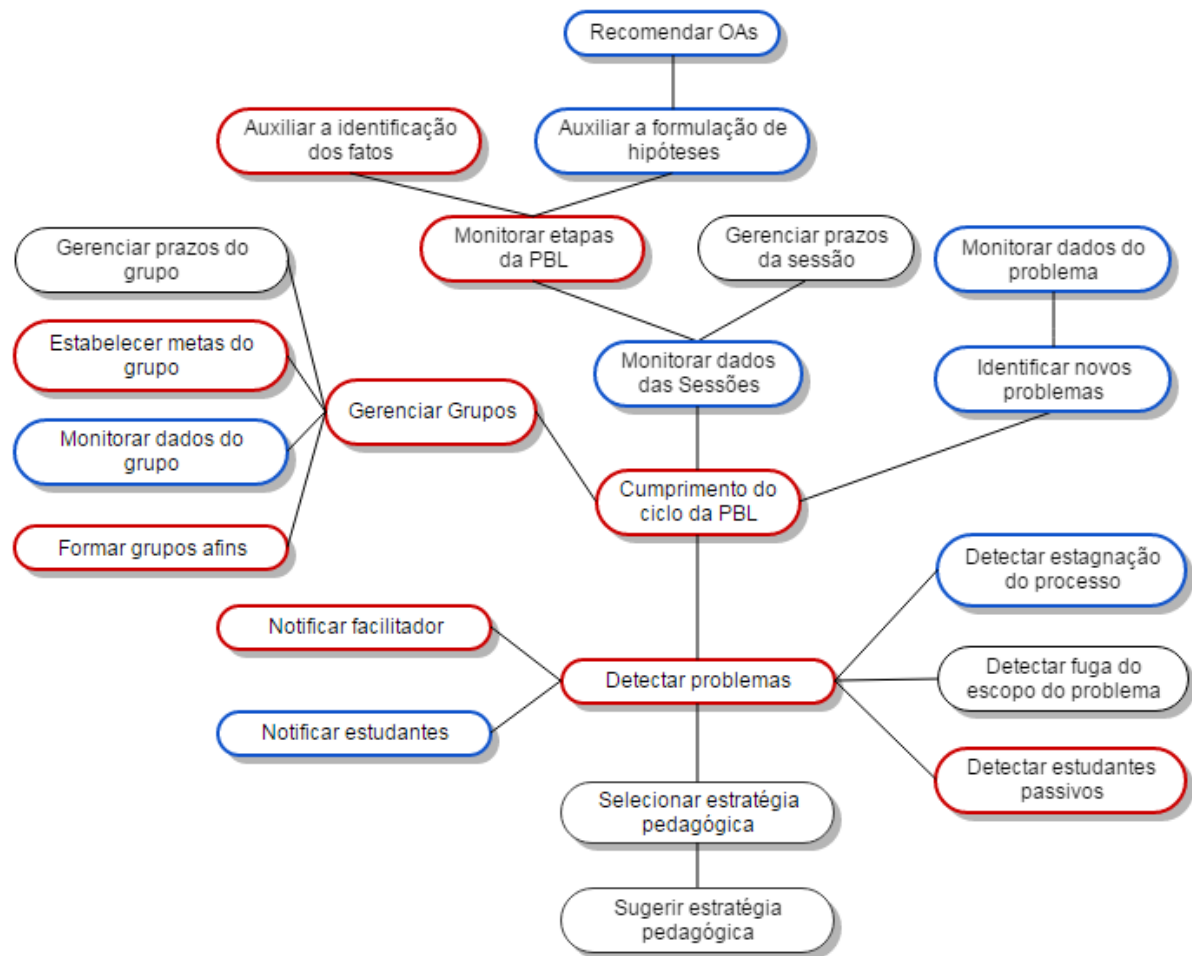
plataformas de agentes (APs). Assim, o DF utilizado na plataforma JADE deste trabalho pode, por exemplo, ser consultado, por meio da Internet, por um agente que esteja sendo executado em outro local, através do protocolo MTP (*Message Transport Protocol*). Essa característica permite a criação de uma rede de colaboração entre os agentes, mesmo que estejam localizados, por exemplo, em universidades distintas (FONTES, 2013; SILVA, 2012). No presente trabalho, este protocolo foi utilizado para comunicação entre agentes que se encontravam na mesma plataforma, visto que os sete agentes criados e o DF se encontram no mesmo servidor.

8.4 Modelagem do Sistema Multiagente

Seguindo os passos da metodologia MAS-CommonKADS+, essa subseção apresenta a modelagem e descrição dos elementos que compõem o SMA desenvolvido neste trabalho. Nem todos os diagramas dos respectivos modelos são necessários para a compreensão da abordagem multiagente proposta, visto que alguns deles acabam fornecendo informações que se tornam redundantes. Logo, decidiu-se representar cada modelo através de um diagrama, contribuindo assim para a compreensão do SMA. A escolha dos modelos a serem inseridos na modelagem depende do tamanho e da complexidade do SMA a ser modelado (FONTES, 2013; SILVA, 2012).

Baseando-se nas necessidades da PBL, o SMA deve realizar uma série de tarefas, as quais podem ser observadas no Modelo de Tarefas, exibido na Figura 7. As tarefas destacadas em vermelho foram contempladas neste trabalho.

Como dito anteriormente, este trabalho estende a solução proposta em (FONTES, 2013). No entanto, algumas alterações foram realizadas. No Modelo de Tarefas, ilustrado na Figura 7, o que está colorido foi contemplado neste trabalho, onde as tarefas em azul foram adicionadas ou modificadas a partir do trabalho de Fontes (2013).

Figura 9 - Modelo de Tarefas.

Fonte: Adaptado de (FONTES, 2013).

A tarefa principal desse SMA é o cumprimento do ciclo da PBL. Essa tarefa, por sua vez, foi decomposta em outras subtarefas, que têm como intuito final auxiliar no atendimento da tarefa principal.

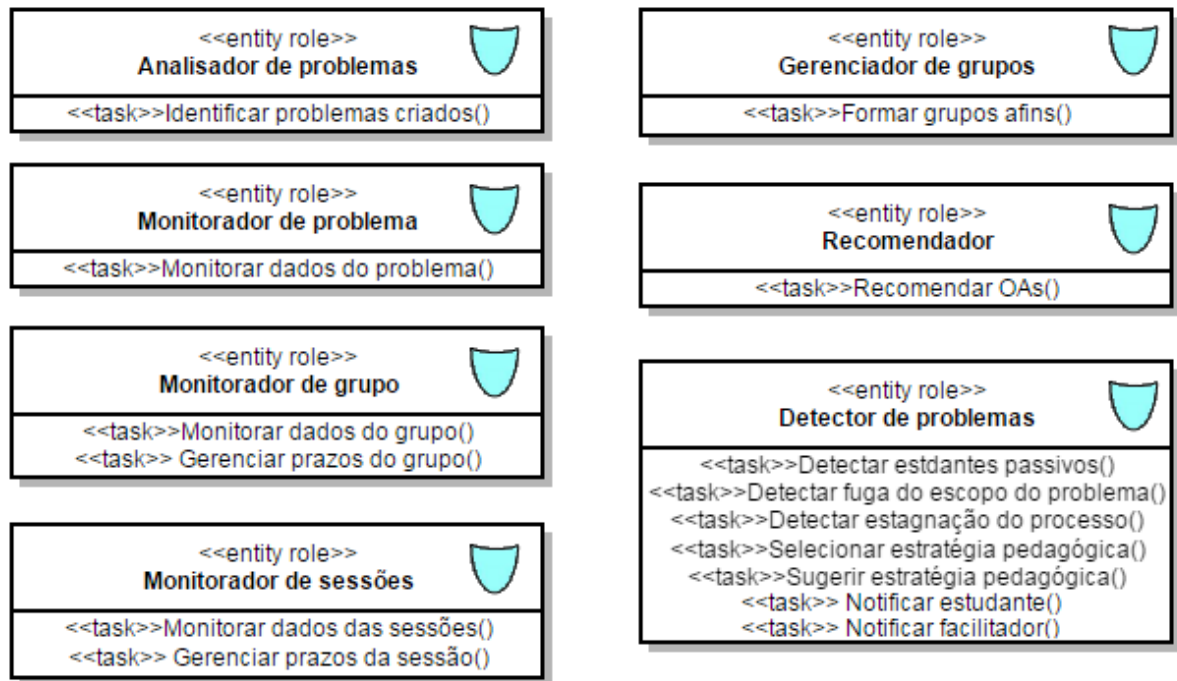
Como pode ser visto na Figura 7, em relação ao Modelo de Tarefas proposto por Fontes (2013), foram adicionadas as tarefas: (i) identificar novos problemas; (ii) monitorar dados do problema; e (iii) monitorar sessões. A recomendação de OAs agora é feita na formulação de hipóteses, etapa que corresponde ao período entre o fim de uma sessão e início de outra. Além disso, a detecção de estagnação do processo realizada pelo AgDP foi contemplada. Agora além da tarefa de gerenciar prazos do grupo, que passou a pertencer o monitoramento do grupo, existe a tarefa gerenciar prazos da sessão que pertence ao monitoramento de sessões.

A tarefa de motivar estudantes, proposta por Fontes (2013), foi substituída pela tarefa de notificar estudantes, já que o presente trabalho não contempla o Agente de Interface que

havia sido desenvolvido em Fontes (2013). Por fim, a tarefa de monitorar a criação do perfil do grupo foi substituída pela tarefa de monitorar os dados do grupo.

Baseando-se nas tarefas elencadas no Modelo de Tarefas da Figura 7, foi definido o Modelo de Papéis apresentado na Figura 8.

Figura 10 - Modelo de Papéis.



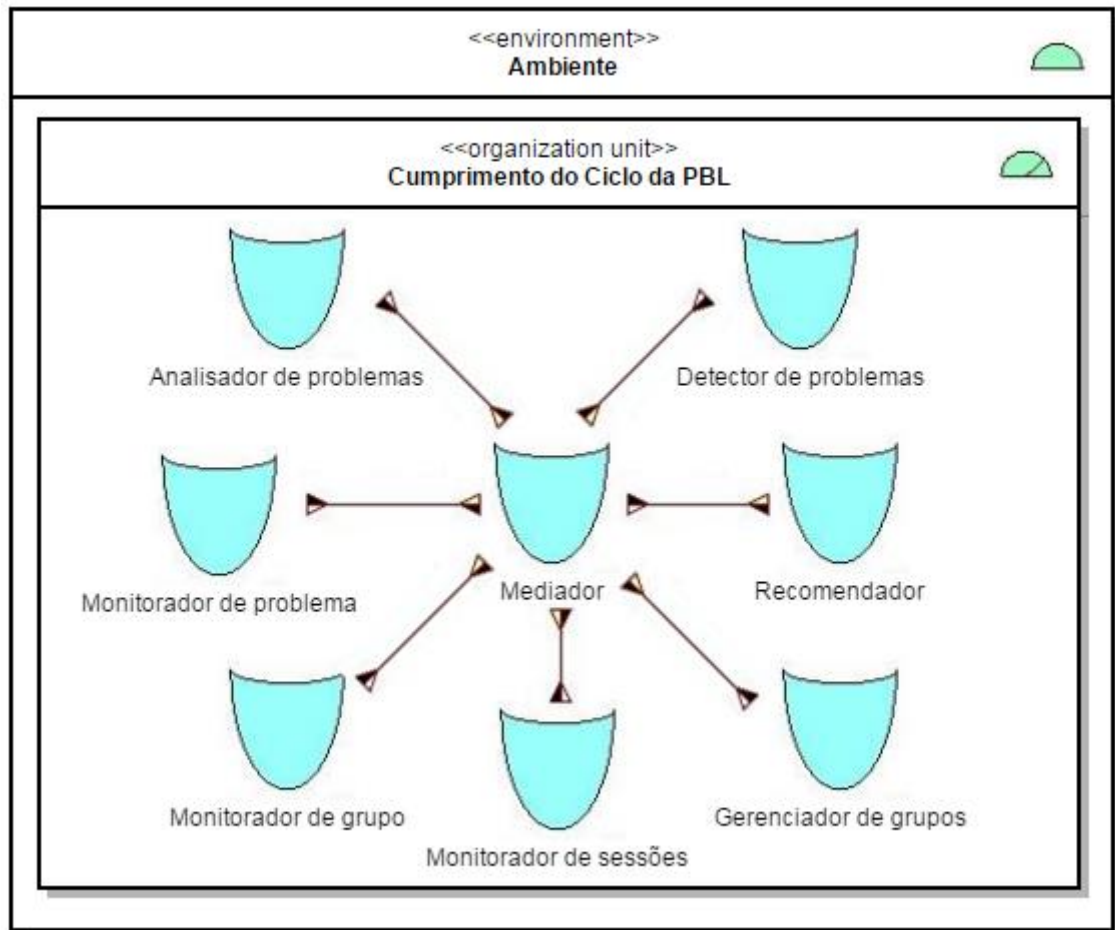
Fonte: Adaptado de (FONTES, 2013).

Observando a Figura 8, é possível perceber que foram definidos sete tipos de papéis para atender às tarefas inerentes ao Modelo de Tarefas: analisador de problemas, recomendador, detector de problemas, gerenciador de grupos, monitorador de problema, monitorador de grupo e monitorador de sessões. Vale ressaltar que as tarefas de selecionar estratégia pedagógica e de sugerir estratégia pedagógica, do papel de detector de problemas, a tarefa gerenciar prazos do grupo, do papel monitorador de grupo e a tarefa gerenciar prazos da sessão do papel monitorador de sessões não foram contempladas neste trabalho.

Como no Modelo de Tarefas, o Modelo de Papéis também sofreu algumas alterações com relação ao mesmo modelo proposto por Fontes (2013). Nesse modelo foi removido o papel de Adjutor, responsável por motivar os estudantes, e adicionado os papéis de Analisador de problemas e Monitorador de sessões.

Uma vez construído o Modelo de Papéis, foi desenvolvido o Modelo de Organização, que serve para descrever a estrutura organizacional dos papéis no sistema, ou seja, como os papéis estão relacionados entre si. A Figura 9 mostra o Modelo de Organização correspondente aos papéis exibidos na Figura 8.

Figura 11 - Modelo de Organização.



Fonte: Adaptado de (FONTES, 2013).

Nesse modelo é possível perceber a presença de outro papel, o mediador. O mediador permitirá que os agentes se comuniquem sem que, inicialmente, eles se conheçam. O motivo dele não fazer parte do Modelo de Papéis é que o mesmo não realiza tarefas que compõem a finalidade principal do SMA, ou seja, não realiza uma das tarefas que compõem o Modelo de Tarefas. Mesmo assim, essa tarefa é de extrema relevância, pois é através dela que os agentes conseguem informações a respeito dos serviços de outros agentes e conseguem se comunicar uns com os outros. Ou seja, ele funciona como um intermediador, permitindo que agentes cadastrem informações e pesquisem por informações de outros agentes.

O Modelo de Organização apresenta a organização interna e externa dos papéis. A organização interna apresenta os relacionamentos entre os papéis do SMA, enquanto que a organização externa demonstra os relacionamentos entre as organizações em si. O SMA deste trabalho possui apenas uma unidade organizacional (*organization unit*), logo não existe uma representação externa. Como pode ser visto no Modelo de Organização da Figura 9, na representação interna da unidade “Cumprimento do ciclo da PBL”, os oito papéis estão relacionados através de um relacionamento do tipo *peer*, que é usado para papéis com a mesma autoridade e é representado por uma seta parcialmente cheia (FONTES, 2013; SILVA, 2012). Uma vez sabendo como os diferentes papéis em um SMA estão organizados entre si, é necessário definir quais agentes serão responsáveis por cada papel. É necessário também definir os objetivos e as características dos agentes, como entrada de dados, condições de ativação do agente e tipos de informação disponíveis.

As subseções a seguir descrevemos *templates* textuais de cada agente, com os objetivos e características, em formato de tabelas para facilitar o entendimento.

8.4.1 *Template* textual do AgP

O AgP realiza o papel de “*Analisador de problemas*”. A Tabela 1 apresenta o seu *template* textual.

Tabela 1 - *Template* textual do AgP.

Agente: AgP
Objetivos: Este agente é responsável por verificar se existem novos problemas que possam ser monitorados e gerenciados pelo SMA.
Parâmetros de Entrada: Identificador do curso, identificador do problema e dados relacionados ao problema.
Parâmetros de Saída: Envia uma requisição ao AgMP, do tipo “ <i>monitorar-problema</i> ”, enviando o identificador do problema.
Condição de Ativação: Quando o SMA é iniciado.

Condição de Finalização: Quando o SMA é encerrado.
Informação Associada: Possui um comportamento (tipo <i>Ticker Behavior</i> (BELIFEMINE; CAIRE; GREENWOOD, 2007)) que atua verificando, dentre os cursos existentes, se há algum problema criado recentemente. Se algum problema nessas condições foi encontrado, envia uma mensagem para o AgMP, solicitando a execução de seus comportamentos.
Descrição: Um único agente desse tipo é criado no SMA. Esse agente é responsável por encontrar novos problemas para serem monitorados.

Fonte: Dados produzidos pelo autor.

8.4.2 *Template* textual do AgMP

O AgMP realiza o papel de “*Monitorador de problema*”. A Tabela 2 apresenta o seu *template* textual.

Tabela 2 - *Template* textual do AgMP.

Agente: AgMP
Objetivos: O AgMP é responsável por monitorar um problema, identificando suas necessidades iniciais e verificando se foi concluído corretamente. Este agente ainda cadastra um serviço do tipo “ <i>monitorador-de-problema</i> ” no DF, para que o AgP possa solicitar seus comportamentos.
Parâmetros de Entrada: Identificador do curso, identificador do problema e dados relacionados ao problema.
Parâmetros de Saída: Solicita a formação de grupos ao AgGG e o monitoramento de grupos ao AgMG.
Condição de Ativação: Quando o agente recebe do AgP uma mensagem com o identificador de conversação do tipo “ <i>monitorar-problema</i> ”.
Condição de Finalização: Quando todos os grupos vinculados a ele concluírem suas atividades.
Informação Associada:

Possui também outro comportamento (tipo *Ticker Behavior*) que verifica se existem grupos vinculados ao problema que ainda não estão sendo monitorados pelo AgMG e que ainda não finalizaram suas atividades. Se sim, envia uma mensagem ao AgMG do tipo “*monitorar-grupos*”. Também possui um comportamento cíclico (tipo *Cyclic Behaviour* (BELIFEMINE; CAIRE; GREENWOOD, 2007)) para ficar monitorando por requisições do AgP. Ao ser acionado, o agente aciona um comportamento (tipo *One Shot Behavior*) que solicita a formação de novos grupos ao AgGG, enviando uma mensagem do tipo “*formar-grupos*”.

Descrição:

É criado um AgMP para cada problema inserido no curso que tenha sido analisado pelo AgP. Esse agente é responsável por monitorar os problemas.

Fonte: Adaptado de (FONTES, 2013).

8.4.3 *Template* textual do AgGG

O AgGG realiza o papel “*Gerenciador de Grupos*” e tem como função principal formar grupos para resolver problemas específicos. A Tabela 3 apresenta o seu *template* textual.

Tabela 3 - *Template* textual do AgGG.

Agente: AgGG
Objetivos: Uma vez em posse das informações enviadas pelo AgMP, o AgGG forma, automaticamente, grupos afins com base nos perfis dos estudantes e requisitos do problema. Também cadastra um serviço do tipo “<i>formador-de-grupos</i>” no DF para que o AgMP possa encontrar esse serviço e se comunicar com ele.
Parâmetros de Entrada: Perfis dos estudantes matriculados no curso e perfis dos grupos.
Parâmetros de Saída: O agente grava no banco de dados os grupos formados indicados para resolver o problema.
Condição de Ativação: Quando o agente recebe do AgMP uma mensagem com o identificador de conversação do tipo “<i>formar-grupos</i>”.
Condição de Finalização:

Quando o agente grava no banco de dados os grupos formados.
Informação Associada: Através do identificador do curso e do identificador do problema, o agente capta as informações sobre os perfis dos estudantes matriculados no curso e as informações relacionadas aos requisitos do problema que corresponde ao perfil desejado para os grupos que serão formados. Uma vez em posse dessas informações, um comportamento de execução única (tipo <i>One Shot Behaviour</i>) é acionado para formar os grupos. Também possui um comportamento cíclico (tipo <i>Cyclic Behaviour</i>) para ficar monitorando as requisições do AgMP.
Descrição: Este agente visa auxiliar o facilitador na tarefa de formação de grupos. Ele é responsável pela alocação automatizada de estudantes a grupos candidatos a participar da resolução de um dado problema.

Fonte: Adaptado de (FONTES, 2013).

8.4.4 *Template* textual do AgMG

O AgMG realiza o papel “*Monitorador de Grupo*” que tem como objetivo monitorar um grupo vinculado ao problema. A Tabela 4 apresenta o seu *template* textual.

Tabela 4 - *Template* textual do AgMG.

Agente: AgMG
Objetivos: Identificar se o grupo iniciou e finalizou suas atividades corretamente. Este agente cadastra um serviço do tipo “ <i>monitorador-de-grupo</i> ” no DF para que o AgMP possa solicitar seus serviços.
Parâmetros de Entrada: Identificador do problema e identificador do grupo.
Parâmetros de Saída: Solicita o monitoramento de sessões e a detecção de problemas, enviando o identificador do curso, o identificador do problema e o identificador do grupo para o AgMS e AgDP.
Condição de Ativação: Quando recebe do AgMP uma mensagem do tipo “ <i>monitorar-grupos</i> ” ou quando recebe do

AgMS uma resposta do tipo “ <i>monitorar-sessoes</i> ”.
Condição de Finalização: Quando todas as atividades do grupo forem finalizadas.
Informação Associada: O AgMG possui um comportamento cíclico (tipo <i>Cyclic Behaviour</i>) para ficar monitorando por requisições do AgMP. Quando é identificada uma requisição do tipo “ <i>monitorar-grupo</i> ”, aciona um comportamento (tipo <i>OneShot Behavior</i>) que envia uma mensagem do tipo “ <i>monitorar-sessoes</i> ” ao AgMS. Outro comportamento (tipo <i>Ticker Behavior</i>) aguarda uma resposta do AgMS, identificando que a última sessão do grupo já foi finalizada. Quando isso ocorre, o AgMG aciona o AgDP para verificar se o grupo enviou a solução final, se realizou a avaliação de pares e se o facilitador realizou a avaliação final.
Descrição: É criado um AgMG para cada grupo vinculado ao problema que não tenha tido suas atividades finalizadas. Esse agente é responsável por monitorar as atividades realizadas pelos estudantes e facilitador, até que o grupo conclua suas tarefas.

Fonte: Adaptado de (FONTES, 2013).

8.4.5 *Template* textual do AgMS

O AgMS realiza o papel “*Monitorador de Sessões*”. A Tabela 5 apresenta o seu *template* textual.

Tabela 5 - *Template* textual do AgMS.

Agente: AgMS
Objetivos: O AgMS tem como propósito acompanhar os estudantes e o facilitador durante as sessões, identificando o estado de cada sessão. Este agente cadastra um serviço do tipo “ <i>monitorador-de-sessoes</i> ” no DF para que o AgMG possa solicitar seus serviços.
Parâmetros de Entrada: Identificador do curso, identificador do problema e identificador do grupo.
Parâmetros de Saída: Uma mensagem para o facilitador informando a necessidade de se criar uma sessão, uma mensagem para o AgR solicitando a recomendação de OAs para os estudantes e uma

mensagem para o AgDP solicitando a detecção de problemas de sessão.
Condição de Ativação: Quando o agente recebe uma solicitação do tipo “ <i>monitorar-sessoes</i> ” do AgMG.
Condição de Finalização: Quando a última sessão for finalizada.
Informação Associada: Possui um comportamento (tipo <i>Ticker Behavior</i>) que verifica se há alguma sessão criada para o grupo. Caso não haja, avisa ao facilitador para criar uma nova sessão. Se existir uma sessão que já tenha sido finalizada, solicita ao AgDP que seja realizada a detecção de problemas na sessão e, se essa sessão ainda não for a última, solicita ao AgR a recomendação de OAs. Possui também um comportamento cíclico (tipo <i>Cyclic Behaviour</i>) para ficar monitorando por requisições do AgMG.
Descrição: É criado um AgMS cada vez que um AgMG inicia suas atividades. Esse agente é responsável por monitorar as atividades realizadas pelos estudantes e facilitador durante as sessões.

Fonte: Dados produzidos pelo autor.

8.4.6 *Template* textual do AgDP

O AgDP realiza o papel “*Detector de problemas*”. A Tabela 6 apresenta o seu *template* textual.

Tabela 6 - *Template* textual do AgDP.

Agente: AgDP
Objetivos: O AgDP tem a função de disponibilizar um serviço do tipo “ <i>detector-de-problemas</i> ” no DF, de modo que o AgMG e o AgMS possam encontrar esse serviço para se comunicar com ele. Ele também é responsável por detectar problemas de colaboração, por parte dos estudantes, e de gerenciamento, por parte do facilitador.
Parâmetros de Entrada: Identificador do problema; identificador do grupo; identificador da sessão; informações sobre quais atividades os estudantes estão executando nas ferramentas colaborativas durante

a sessão; informações sobre o conteúdo das discussões nas ferramentas colaborativas.
Parâmetros de Saída: . E-mail para o facilitador e para os estudantes com informações sobre problemas encontrados.
Condição de Ativação: Quando o agente recebe do AgMP uma mensagem com o identificador de conversação do tipo " <i>detectar-problemas-de-grupo</i> " ou do AgMS uma mensagem com identificador de conversação do tipo " <i>detectar-problemas-de-sessão</i> ".
Condição de Finalização: Quando o agente percebe que o ciclo da PBL foi encerrado.
Informação Associada: Este agente possui dois comportamentos periódicos (tipo <i>Ticker Behavior</i>). Quando requisitado pelo AgMS, este agente executa o comportamento " <i>detectar-problemas-de-sessão</i> ", que consiste em detectar alunos passivos, verificar se o relatório da sessão foi enviado pelo relator e verificar se a avaliação da sessão foi realizada pelo facilitador. Já quando requisitado pelo AgMG, este agente executa o comportamento " <i>detectar-problemas-de-grupo</i> ", que consiste em verificar se a solução do problema foi enviada, verificar se a avaliação de pares foi realizada pelos estudantes e verificar se a avaliação final foi realizada pelo facilitador. O AgDP também possui um comportamento cíclico (tipo <i>Cyclic Behaviour</i>) para ficar monitorando por requisições dos agentes AgMG e AgMS .
Descrição: Este agente é responsável por detectar problemas de colaboração que podem vir a acontecer durante o processo de aplicação da PBL.

Fonte: Adaptado de (FONTES, 2013).

8.4.7 *Template* textual do AgR

O AgR realiza o papel de “*Recomendador*”, baseando-se nos dados do problema e nos conceitos que os estudantes desconhecem na descrição do problema. A Tabela 7 apresenta o seu *template* textual.

Tabela 7 - *Template* textual do AgR.

Agente: AgR

Objetivos:
Detectar OAs adequados ao contexto do estudante, de acordo com (i) as informações providas pelos seus respectivos perfis, (ii) os assuntos que os estudantes desconhecem e (iii) as informações obtidas dos OAs disponíveis no repositório. Cadastra um serviço do tipo “ <i>recomendador-de-oas</i> ” no DF para que o AgMS possa solicitar seus serviços.
Parâmetros de Entrada:
Identificador do estudante, os conceitos que os estudantes desconhecem do problema proposto e informações dos OAs disponíveis no repositório de OAs SCORM.
Parâmetros de Saída:
Possíveis OAs adequados ao contexto do estudante.
Condição de Ativação:
Quando recebe do AgMS uma mensagem de conversação do tipo “ <i>recomendar-oas</i> ”.
Condição de Finalização:
Quando o agente envia para o estudante, via e-mail, as informações do OA recomendado.
Informação Associada:
Toma as decisões de acordo com o resultado da execução de um AG. Essa recomendação auxilia os estudantes a sanar suas dúvidas na fase de identificação dos fatos, durante a resolução do problema. Este agente também possui um comportamento cíclico (tipo <i>Cyclic Behaviour</i>) para ficar monitorando por requisições do AgMS.
Descrição:
Este agente possui o comportamento voltado para a recomendação de OAs. Ele é responsável por detectar e realizar a recomendação de OAs.

Fonte: Adaptado de (FONTES, 2013).

8.4.8 *Template* textual do DF

O papel de “*Mediador*” da comunicação entre os agentes é realizado pelo agente DF. A Tabela 8 apresenta o seu *template* textual.

Tabela 8 - *Template* textual do Agente DF.

Agente: DF
Objetivos:
Possibilitar que os agentes possam cadastrar e procurar por serviços oferecidos por outros agentes (serviço de páginas amarelas).

Parâmetros de Entrada:
Registros de serviços por parte dos agentes.
Parâmetros de Saída:
Lista de serviços e os respectivos agentes responsáveis pelos mesmos.
Condição de Ativação:
Quando a plataforma JADE é inicializada.
Condição de Finalização:
Caso a plataforma JADE seja finalizada.
Informação Associada:
É um agente inicializado pela própria plataforma JADE como um dos componentes integrantes do padrão FIPA.
Descrição:
Este agente possui o comportamento voltado para a mediação entre os outros agentes. Sua função principal é fornecer uma arquitetura do tipo “quadro-negro”, onde agentes escrevem informações, procuram por informações escritas por outros agentes e conseguem, através do serviço provido pelo DF, se comunicar com o agente que escreveu aquela informação.

Fonte: Adaptado de (FONTES, 2013).

8.5 Cenários de Uso

8.5.1 Cenários de Uso do Facilitador

Para que o módulo possa ser utilizado, é necessário criar primeiramente um curso e inscrever nele os alunos e o facilitador que participarão da resolução do problema. Esse processo pode ser realizado utilizando as ferramentas convencionais do MOODLE.

Em seguida, deve-se criar uma instância do módulo “*problem*”, levando em conta que o facilitador é o responsável por elaborar e apresentar o problema a um grupo de estudantes. Para isso, o facilitador deve acessar o curso no qual o problema será aplicado e clicar no botão “Ativar edição” e, em seguida, clicar no *link* “Adicionar uma atividade ou recurso”. Uma janela será aberta, exibindo os módulos disponíveis. O facilitador deverá selecionar o módulo “*problem*” e clicar em adicionar. Uma nova página será carregada, onde o facilitador poderá definir um problema, incluindo as informações em um formulário. A Figura 10 exibe o formulário de definição do problema.

Figura 12 - Formulário de definição do problema.

Título do problema:*

Descrição*

Parágrafo

B

I

Fonte: Dados produzidos pelo autor.

Após definir o problema, o facilitador deverá definir os requisitos e as metas de aprendizagem. Os requisitos servirão de apoio na formação de grupos, auxiliando o facilitador a selecionar quais estudantes estão mais aptos a participar de um determinado grupo. Já as metas de aprendizagem, serão utilizadas na avaliação, para que o avaliador possa adicionar novos conhecimentos à lista de características dos estudantes e que os próprios estudantes possam utilizar como referência na avaliação de pares e autoavaliação. A Figura 11 mostra os formulários de definição de requisitos e metas de aprendizagem.

Figura 13 - Formulários de metas de aprendizagem e definição de requisitos.

✉ Metas de aprendizagem

Descrição

Construir o modelo de requisitos nos padrões da UML.

— Remover

Descrição

Adicionar meta de aprendizagem

✉ Requisitos do problema

Descrição	Nível	Significância	
-----------	-------	---------------	--

Descrição

Nível de conhecimento

☐ Baixo
 ☐ Médio
 ☐ Alto

Significância

Média

Adicionar área de conhecimento

Fonte: Dados produzidos pelo autor.

Uma vez tendo criado o problema e definido suas características, é necessário que o facilitador crie um grupo e o vincule ao problema. Nessa etapa, o facilitador pode gerar os problemas manualmente, através das ferramentas de criação de grupos do MOODLE ou o SMA pode identificar a necessidade de criar e sugerir grupos automaticamente. É importante salientar que o perfil desejado, construído pelo facilitador, é o que melhor se adéqua à resolução do problema. Assim, um estudante que tenha um perfil aproximado ao desejado terá as competências necessárias à resolução do problema proposto.

Para vincular um grupo ao problema, o facilitador deve visualizar o problema e clicar em “Listagem de grupos”. Nessa tela, serão exibidos os grupos de estudantes cadastrados, separados em vinculados e não vinculados ao problema. Clicando no botão “Vincular”, na listagem de grupos não vinculados, o facilitador vincula o grupo ao problema. Analogamente, para desvincular um grupo de um problema, é necessário clicar no botão “Desvincular”, na listagem de grupos vinculados. A tela listagem de grupos pode ser vista na Figura 12.

Figura 14 - Tela de listagem de grupos.



Fonte: Dados produzidos pelo autor.

A Figura 13 ilustra a interface de visualização de grupos. Nessa interface existem duas abas, “Grupo” e “Sessões”. Na aba “Grupo”, o facilitador pode identificar se o estudante já foi avaliado, visualizar as avaliações de pares e os perfis dos estudantes. Clicando na aba “Sessões”, o facilitador pode visualizar, editar e criar sessões. É importante lembrar que para que uma nova sessão seja criada é necessário que a sessão anterior tenha sido finalizada.

Figura 15 - Tela de visualização de grupos.

[🏠 Voltar ao início](#)

Grupo

Sessões

Dados do Grupo: Grupo A		
Nome	Avaliação final	Avaliação de pares
Eduardo Lopes	Não avaliado Avaliar usuário	Visualizar avaliação de pares
Carmen Júlia	Não avaliado Avaliar usuário	Visualizar avaliação de pares
Amanda Mendes	Não avaliado Avaliar usuário	Visualizar avaliação de pares
Carlos Júnior	Não avaliado Avaliar usuário	Visualizar avaliação de pares
Diego Lima	Não avaliado Avaliar usuário	Visualizar avaliação de pares
Marta Lopes	Não avaliado Avaliar usuário	Visualizar avaliação de pares
José Oliveira	Não avaliado Avaliar usuário	Visualizar avaliação de pares
Calos Bento	Não avaliado Avaliar usuário	Visualizar avaliação de pares

Fonte: Dados produzidos pelo autor.

A interface ilustrada na Figura 14 mostra a listagem de sessões.

Figura 16 - Tela de listagem de sessões.

[🏠 Voltar ao início](#)

Grupo

Sessões

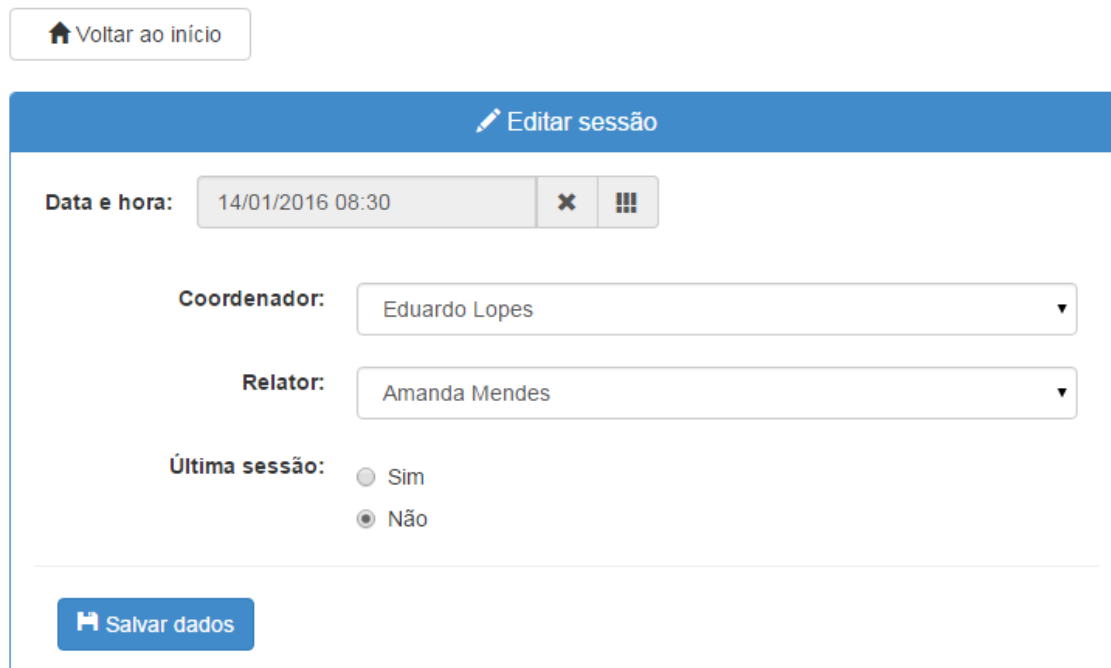
Dados das sessões		
Data	Status	Ações
14/01/2016 08:30	Próxima sessão	👁 Visualizar 🗑 Excluir ✎ Editar

Fonte: Dados produzidos pelo autor.

Para criar uma sessão, o facilitador deve clicar no botão “Criar nova sessão”, na página de listagem de sessões. Ao clicar no botão, será exibido o formulário, onde o facilitador deve escolher uma data para início da sessão. O SMA pode recomendar uma data

que seja mais viável para todos os membros do grupo. Também devem ser selecionados o coordenador e relator da sessão. Por fim, o facilitador deve identificar se a sessão a ser criada será ou não a última sessão. A Figura 15 ilustra o formulário de criação de sessão.

Figura 17 - Formulário de criação de sessão.



O formulário de criação de sessão é composto por um cabeçalho azul com o texto "Editar sessão" e um ícone de lápis. Abaixo do cabeçalho, há um botão "Voltar ao início" com um ícone de casa. O formulário principal contém os seguintes campos:

- Data e hora:** Um campo de texto com o valor "14/01/2016 08:30", um ícone de "X" para limpar e um ícone de calendário.
- Coordenador:** Um campo de seleção com o valor "Eduardo Lopes" e uma seta para baixo.
- Relator:** Um campo de seleção com o valor "Amanda Mendes" e uma seta para baixo.
- Última sessão:** Duas opções de rádio: "Sim" (desselecionada) e "Não" (selecionada).

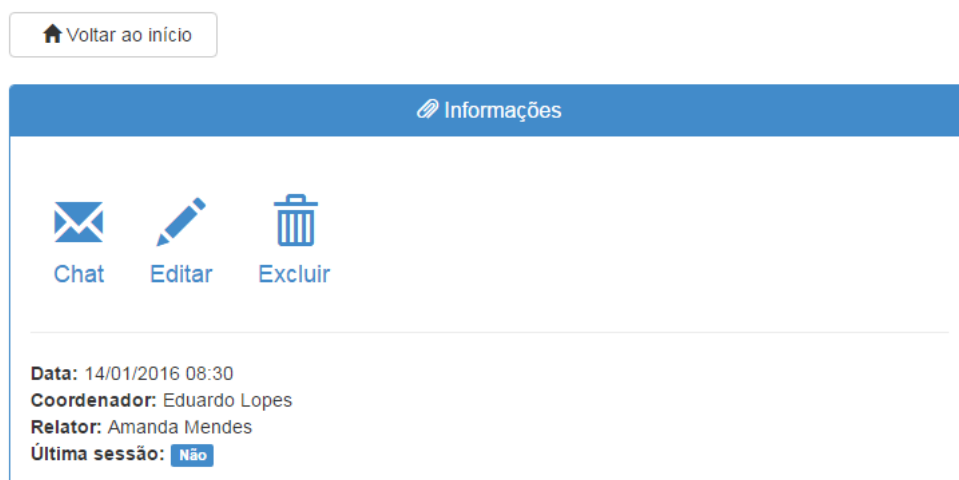
Na base do formulário, há um botão azul "Salvar dados" com um ícone de salvar.

Fonte: Dados produzidos pelo autor.

Tendo em vista a necessidade de interação síncrona entre os estudantes para a realização das sessões tutoriais, foi escolhida a ferramenta *chat*, do próprio MOODLE, para cumprir esse papel. Essa ferramenta tem como característica promover a comunicação, em tempo real, entre estudantes, podendo ou não ser em grupos. Em outras palavras, para o módulo “*problem*”, uma sessão é uma reunião, com data e hora previamente definidas, onde os membros de um grupo utilizam o *chat* do MOODLE para se comunicar.

Para acessar a sessão, o facilitador deverá clicar no botão “Visualizar” na listagem. Na tela de visualização de sessão, é possível ver detalhadamente os dados da sessão. O botão “*Chat*” dá acesso à ferramenta de comunicação, que deve ser acessada pelos estudantes e pelo facilitador, na data e hora marcada. A Figura 16 exibe a tela de visualização de sessão.

Figura 18 - Tela de visualização de sessão.



A interface de visualização de uma sessão. No topo, há um botão "Voltar ao início" com um ícone de casa. Abaixo, uma barra azul com o título "Informações". O conteúdo principal apresenta três ícones: um envelope para "Chat", uma caneta para "Editar" e um lixeira para "Excluir". Abaixo desses, há um formulário com os seguintes campos: "Data: 14/01/2016 08:30", "Coordenador: Eduardo Lopes", "Relator: Amanda Mendes" e "Última sessão:" seguido de um botão "Não".

Fonte: Dados produzidos pelo autor.

Depois de realizada a sessão, o facilitador pode finalizá-la na tela de listagem de sessões, clicando no botão “*Finalizar sessão*”. Feito isso, o relator deverá enviar o relatório de sessão. O facilitador deverá então realizar a avaliação da sessão, utilizando o formulário de avaliação de sessão, que pode ser acessado na página de visualização de sessão. O formulário de avaliação de sessão pode ser visto na Figura 17.

Figura 19 - Formulário de avaliação de sessão.



O formulário de avaliação de sessão. Possui um cabeçalho azul com o título "Avaliação de sessão". O formulário contém três campos de texto rotulados "Coordenador:", "Relator:" e "Grupo:". Cada campo possui um ícone de lápis no canto inferior direito, indicando que é editável. No rodapé, há um botão azul "Salvar dados" com um ícone de salvar.

Fonte: Dados produzidos pelo autor.

Após a última sessão ser finalizada, o facilitador terá acesso ao relatório final e à solução do problema, ambos disponíveis na interface de visualização de grupos. Além disso, como visto anteriormente, nessa mesma interface, o facilitador pode ter acesso às avaliações de pares realizadas pelos alunos. É importante que o facilitador visualize essas avaliações, já que a competência de avaliador dos alunos faz parte do currículo PBL. A Figura 18 exibe o formulário da solução do problema.

Figura 20 - Formulário da solução do problema.

Fonte: Dados produzidos pelo autor.

O facilitador poderá realizar a avaliação final clicando em “*Avaliar usuário*”. Na interface que será exibida, o facilitador irá encontrar as metas de aprendizagem que cadastrou durante a definição do problema. Para cada meta, existe uma caixa de seleção correspondente, onde o facilitador deverá atribuir uma nota de 0 a 10 para o aprendizado daquele usuário com relação à meta. Um campo de avaliação de grupo também estará presente no formulário. Nesse campo, o facilitador poderá comentar sobre a colaboração, interação e o resultado final. Porém, só é necessário preencher esse campo uma vez por grupo. A Figura 19 mostra o formulário de avaliação final de estudante.

Figura 21 - Formulário de avaliação final do estudante.

 Voltar ao início

✉ Avaliação de aluno baseado nas metas de aprendizagem

Nome do aluno: Eduardo Lopes

Meta de aprendizagem	Valor
Construir o modelo de requisitos nos padrões da UML.	0 0 1 2 3 4 5 6 7 8 9 10

Avaliação do grupo:

 Salvar avaliação

Fonte: Dados produzidos pelo autor.

Por fim, o facilitador poderá finalizar o problema para um dado grupo de usuários clicando no botão “*Finalizar*”, na tela de listagem de grupos. Esse processo irá fazer com que o SMA deixe de agir nesse grupo.

8.5.2 Cenários de Uso do Estudante

Para visualizar um problema, o estudante deverá estar em um grupo vinculado ao problema. Além disso, só é possível visualizar os dados do problema após o início da primeira sessão.

Para que o facilitador possa criar os grupos, é necessário que ele conheça os estudantes. O módulo “*problem*” permite que os estudantes acrescentem mais informações aos seus perfis. Para adicionar novas características, o estudante pode acessar o problema e clicar em “*Perfil*”. Na página que irá abrir, o estudante poderá escolher os melhores horários para acessar o MOODLE e adicionar conhecimentos que adquiriu e que podem ser importantes para o seu aprendizado. Outra aplicação para o perfil de usuário é a

recomendação de OAs, oferecida pelo SMA. A Figura 20 mostra o formulário de edição de perfil do estudante.

Figura 22 - Formulário de edição de perfil do estudante.

Problema

Grupo

Sessões

Meu perfil

Horários disponíveis para estudo

Domingo:

☐ Manhã ☐ Tarde ☐ Noite

Segunda:

☐ Manhã ☐ Tarde ☐ Noite

Terça:

☐ Manhã ☐ Tarde ☐ Noite

Quarta:

☐ Manhã ☐ Tarde ☐ Noite

Quinta:

☐ Manhã ☐ Tarde ☐ Noite

Sexta:

☐ Manhã ☐ Tarde ☐ Noite

Sábado:

☐ Manhã ☐ Tarde ☐ Noite

Salvar dados

Características

Nenhuma característica encontrada.

Adicionar nova característica

Descrição

Valor

0 ▼

Adicionar característica

Fonte: Dados produzidos pelo autor.

Para ter acesso a tela de visualização do problema, é necessário clicar no *link* do problema, dentro do curso. Nessa tela, o estudante poderá visualizar o título, a descrição do

problema, o formato do produto final e as áreas de conhecimentos que o problema aborda, conforme ilustrado na Figura 21.

Figura 23 - Tela de visualização do problema.



🏠 Problema Grupo 📅 Sessões 👤 Meu perfil

Dados do problema

Levantamento de requisitos

Uma artesã alterna seu trabalho entre produzir peças de artesanatos e o gerenciamento do seu negócio. O controle de gasto e de estoque é feito de forma manual utilizando papel e caneta, o que provoca uma enorme perda de tempo. Ela então, decidiu facilitar seu trabalho adquirindo um computador. Mas para utilizá-lo de forma efetiva, é necessário um software que possibilite gerenciar seu material de produção mais rápida e facilmente. Nossa empresa se propôs a ajudá-la, e pra facilitar a implementação utilizaremos a UML para projetar um sistema que controle os gastos com materiais para produção e estoque de peças produzidas.

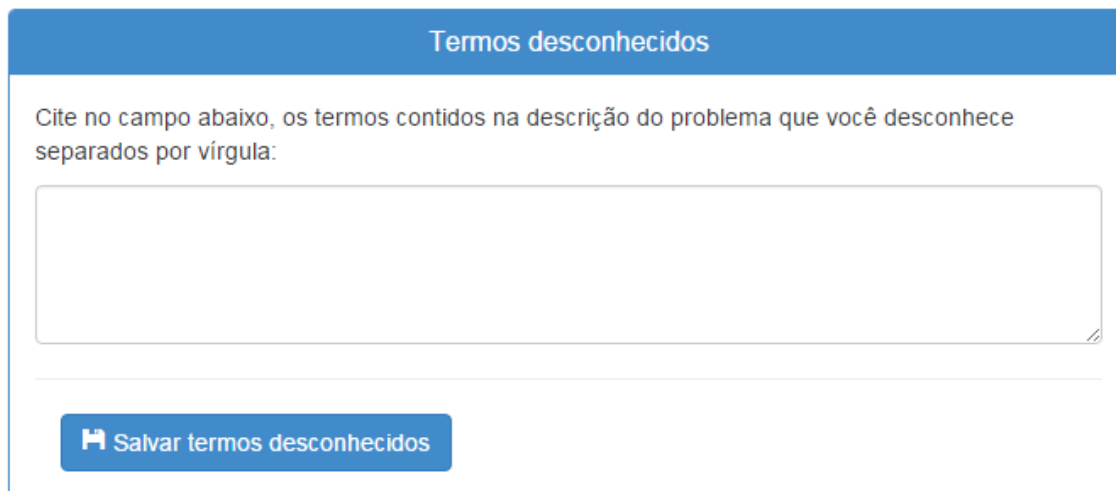
Produto final: Um modelo de levantamento de requisitos do sistema no padrão UML.

Áreas de conhecimento: Análise e projeto de sistemas, Engenharia de software.

Fonte: Dados produzidos pelo autor.

O estudante pode ainda inserir, no formulário de termos desconhecidos, os termos desconhecidos identificados na descrição do problema. A Figura 22 ilustra o formulário de termos desconhecidos.

Figura 24 - Formulário de termos desconhecidos.

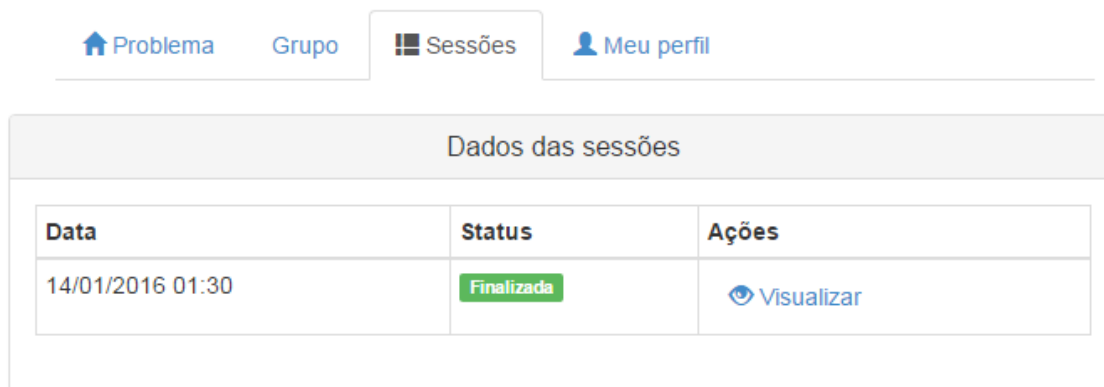


O formulário tem um cabeçalho azul com o título "Termos desconhecidos". Abaixo, há um texto explicativo: "Cite no campo abaixo, os termos contidos na descrição do problema que você desconhece separados por vírgula:". Segue um campo de texto grande e vazio para a entrada. No rodapé, há um botão azul com o ícone de um documento e o texto "Salvar termos desconhecidos".

Fonte: Dados produzidos pelo autor.

Clicando na aba “Sessões”, o estudante tem acesso à lista de sessões. Nessa página, o estudante pode ver a data e o status da sessão, conforme visto na Figura 23.

Figura 25 - Tela de listagem de sessões.



A interface apresenta uma barra de navegação superior com quatro abas: "Problema" (com ícone de casa), "Grupo", "Sessões" (com ícone de calendário e selecionada) e "Meu perfil" (com ícone de pessoa). Abaixo, há uma seção intitulada "Dados das sessões" que contém uma tabela com as seguintes informações:

Data	Status	Ações
14/01/2016 01:30	Finalizada	 Visualizar

Fonte: Dados produzidos pelo autor.

Clicando em "Visualizar", o estudante terá acesso aos demais dados da sessão, como, por exemplo, o coordenador e o relator da sessão, se essa é ou não a última sessão, o botão de acesso ao *chat*, o relatório da sessão e avaliação da sessão.

O relatório de sessão deverá ser realizado pelo relator, após a sessão ser finalizada. Para isso, o relator deve preencher o formulário de envio do relatório de sessão, na página de visualização da sessão, descrevendo o que ocorreu durante a sessão. Para visualizar o relatório da sessão, o estudante ou facilitador deverá acessar a página de visualização da sessão. A Figura 24 mostra o formulário de relatório da sessão.

Figura 26 - Formulário de relatório de sessão.



The image shows a web form titled 'Relatório' in a blue header bar. Below the header, the text 'Relatório da sessão:' is centered. Underneath is a toolbar with buttons for 'Texto normal' (with a dropdown arrow), 'Negrito', 'Itálico', 'Sublinhado', bulleted list, numbered list, decrease indent, increase indent, undo, and redo. Below the toolbar is a large text area labeled 'Dados do relatório'. At the bottom left of the form is a blue button with a floppy disk icon and the text 'Salvar dados'.

Fonte: Dados produzidos pelo autor.

A solução do problema deverá ser enviada após a última sessão ser finalizada, podendo ser enviada por qualquer estudante pertencente ao grupo. Na aba de visualização do problema, o estudante deve enviar o(s) arquivo(s) referente(s) à solução do problema, através do formulário de envio da solução do problema. No mesmo local, é exibido o formulário de relatório final. A Figura 25 exibe os formulários de relatório final e solução do problema.

Figura 27 - Formulários de relatório final e solução do problema.

Relatório final

Parágrafo B I [Listas] [Link] [Imagem] [Outros ícones]

Caminho: p

Solução: Tamanho máximo para novos arquivos: 3Mb, máximo de anexos: 99

Arquivos

Você pode arrastar e soltar arquivos aqui para adicioná-los.

Salvar mudanças

Fonte: Dados produzidos pelo autor.

Na aba “Grupo”, o estudante pode visualizar os demais estudantes que compõem o grupo, além de realizar as avaliações, conforme ilustrado na Figura 26.

Figura 28 - Tela de visualização de grupo.

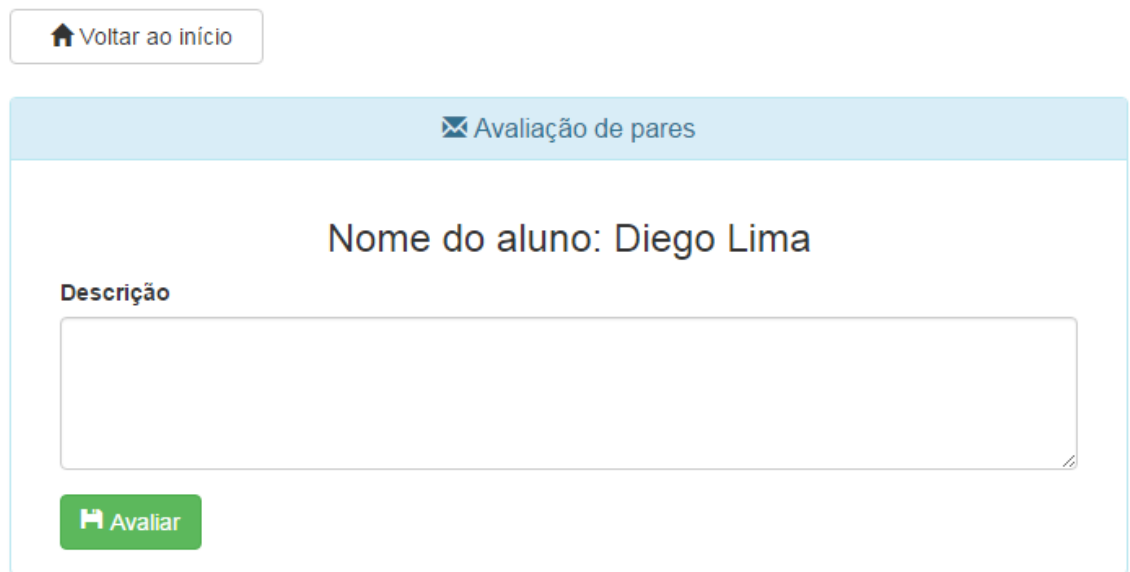
Problema	Grupo	Sessões	Meu perfil
--------------------------	--------------	-------------------------	----------------------------

Dados do Grupo: Grupo A	
Nome	Avaliação de pares
Eduardo Lopes	Avaliar Visualizar avaliação
Carmen Júlia	Avaliar Visualizar avaliação
Amanda Mendes	Avaliar Visualizar avaliação
Carlos Júnior	Avaliar Visualizar avaliação
Diego Lima	Avaliar Visualizar avaliação
Marta Lopes	Avaliar Visualizar avaliação
José Oliveira	Avaliar Visualizar avaliação
Calos Bento	Avaliar Visualizar avaliação

Fonte: Dados produzidos pelo autor.

A avaliação de pares e autoavaliação devem ser realizadas após o envio da solução do problema. Seguidamente, o estudante deverá escolher um membro do grupo e clicar no *link* “Avaliar”. Na janela que irá abrir, o estudante poderá adicionar comentários sobre a participação do avaliado, durante as sessões e o autoaprendizado. O estudante deve avaliar inclusive a si mesmo, sendo este o processo de autoavaliação. A Figura 27 mostra o formulário de avaliação de pares.

Figura 29 - Formulário de avaliação de par.



O formulário de avaliação de pares apresenta uma interface com um cabeçalho azul claro contendo o ícone de envelope e o texto "Avaliação de pares". Abaixo, o nome do aluno "Nome do aluno: Diego Lima" é exibido em uma fonte maior. À esquerda, há uma seção rotulada "Descrição" com um campo de texto grande e vazio para o comentário. No canto inferior esquerdo, um botão verde com o ícone de uma seta para cima e o texto "Avaliar" permite submeter a avaliação. No topo esquerdo do formulário, um botão com um ícone de casa e o texto "Voltar ao início" oferece uma opção de navegação.

Fonte: Dados produzidos pelo autor.

9 ESTUDO DE CASO

Esta seção tem como objetivo descrever e apresentar os resultados do estudo de caso realizado no presente trabalho.

9.1 Ferramentas Utilizadas

Para apoiar a realização do estudo de caso foram utilizadas duas ferramentas, o Google Forms e o Microsoft Excel. As subseções a seguir descrevem cada uma dessas ferramentas.

9.1.1 Google Forms

Segundo o site oficial, o *Google Forms* é uma ferramenta on-line de criação de formulários onde se tem a possibilidade de criar questionários de forma rápida e intuitiva. Essa ferramenta além de tornar fácil a criação de formulários, possui integração com software *Planilhas* da mesma empresa. Dessa forma o pesquisador pode visualizar os resultados diretamente de uma planilha sem a necessidade de exportação. Mas, caso seja necessário, a planilha também pode ser salva em diversos formatos, permitindo a análise dos dados em outros softwares.

9.1.2 Microsoft Excel

O *Microsoft Excel* é um software de gerenciamento de planilhas digitais que permite a estruturação, formatação, organização e análise de dados de forma eficiente. Oferece uma grande variedade de recursos para facilitar e potencializar sua utilização em diversas áreas.

9.2 Metodologia

Em (OLIVEIRA, 2015), foi realizada uma avaliação dos principais modelos de validação de uma tecnologia. Nesse estudo, o autor cita sete modelos, descrevendo suas características e a forma na qual cada um é aplicado.

Com base no estudo realizado em (OLIVEIRA, 2015), no presente trabalho, foi escolhido o Modelo de Aceitação da Tecnologia ou *Technology Acceptance Model* (TAM)

para realizar a validação da solução proposta. Essa metodologia foi selecionada por sua facilidade de uso e porque seu objetivo é avaliar a intenção de uso de uma tecnologia através da observação da utilidade e da facilidade de uso do sistema.

O TAM foi proposto por Davis (1989) na década de 80, com o objetivo de auxiliar e prever o uso dos sistemas. O objetivo é esclarecer o comportamento dos usuários e identificar o porquê da não aceitação de um sistema ou tecnologia específica e fazer as medidas corretivas.

O modelo TAM foi idealizado para que pudesse interpretar variáveis externas de aceitação dos usuários e o real uso da tecnologia. De acordo com Davis (1989), a finalidade era entender o comportamento do indivíduo a partir da utilidade e da facilidade de utilização percebida por ele.

A utilidade percebida é o quanto um indivíduo acha que a tecnologia irá melhorar seu desempenho no trabalho. A facilidade de utilização percebida é se utilizar essa tecnologia vai exigir pouco esforço. A junção desses dois construtos resulta nas intenções de uso.

As pessoas formam intenções para desempenhar ações que tenham um sentimento positivo. O esforço economizado, motivado pela melhora na facilidade de uso detectada, pode ser aplicado em outras tarefas, permitindo assim que um indivíduo realize mais trabalho com o mesmo esforço.

9.2.1 Objetivos

O objetivo geral dessa validação é testar empiricamente as hipóteses no Modelo de Aceitação da Tecnologia para medir os seus construtos. Pretende-se chegar ao objetivo de medir a influência de cada construto, testando as seguintes hipóteses:

- **Hipótese 1:** o treinamento dado ao facilitador e aos estudantes influencia na facilidade de uso do sistema percebida pelo usuário;
- **Hipótese 2:** a complexidade que envolve o entendimento da metodologia influencia diretamente na facilidade de uso do sistema percebida pelo usuário;
- **Hipótese 3:** a experiência do usuário influencia na facilidade de uso do sistema percebida pelo usuário;
- **Hipótese 4:** a facilidade de uso percebida pelo usuário influencia na utilidade percebida pelo usuário;

- **Hipótese 5:** a utilidade percebida influencia a intenção de uso da ferramenta pelo usuário;
- **Hipótese 6:** a facilidade de uso percebida influencia a intenção de uso da ferramenta pelo usuário.

9.2.2 Amostragem

Foi selecionada uma amostra de 21 indivíduos. Os entrevistados se dividem em 8 estudantes da disciplina Análise e Projeto de Sistemas do curso de Ciência da Computação da Universidade Federal Rural do Semi-Árido (UFERSA) e 13 da disciplina Computação e Música do curso Licenciatura em Música da Universidade do Estado do Rio Grande do Norte (UERN).

9.2.3 Obtenção dos dados

Os dados foram coletados através de um questionário disponível on-line, utilizando a ferramenta *Google Forms*. O questionário possui vinte questões, conforme descrito na Tabela 9.

Tabela 9 - Variáveis, conceitos e dimensões.

Dimensões/Conceitos		Variáveis
Facilidade de Uso Percebida		FUP1 - Em minha concepção o módulo é de fácil utilização.
		FUP2 - Em minha concepção o módulo é de fácil compreensão.
		FUP3 - A interação com o módulo é simples.
		FUP4 - Acho que utilizar o módulo com facilidade depende do treinamento aplicado.
		FUP5 - É fácil saber o que fazer em cada etapa da utilização do módulo.
		FUP6 - Utilizar o módulo não requer conhecimentos avançados sobre o MOODLE.
		FUP7 - A utilização do módulo fica mais fácil à medida que é usado.
Utilidade Percebida		UP1 - Acho que o módulo facilita a aplicação da Aprendizagem

	Baseada em Problema no MOODLE.
	UP2 - Acho que utilizar o módulo simplifica a experiência com a Aprendizagem Baseada em Problema no MOODLE.
	UP3 - A utilização do módulo melhora a produtividade na experiência com a Aprendizagem Baseada em Problema no MOODLE.
	UP4 - Acho que é mais eficaz aplicar a Aprendizagem Baseada em Problema no MOODLE utilizando o módulo.
Intenção de Uso Percebida	IUP1 - Indicaria a utilização da ferramenta para aplicar a Aprendizagem Baseada em Problema em outras ocasiões.
	IUP2 - Utilizaria a ferramenta em outras ocasiões.
Variáveis Externas	VE1 - Aprender novas metodologias não é um desafio para mim.
	VE2 - A Aprendizagem Baseada em Problema é de simples compreensão para mim.
	VE3 - O treinamento dado foi simples e didático.
	VE4 - O treinamento dado me auxiliou no entendimento da utilização da ferramenta.
	VE5 - Você já teve alguma experiência com ensino a distância?
	VE6 - Você já teve alguma experiência com o MOODLE?
	VE7 - Você já teve alguma experiência com a Aprendizagem Baseada em Problema?
Fonte: Dados produzidos pelo autor.	

O formulário é composto de questões do tipo escala *Linkert* e de múltipla escolha. Neste trabalho, as questões do tipo escala *Linkert* possuem opções de 1 a 7, assumindo que de 1 a 3 o entrevistado discorda da afirmação, o valor 4 representa indecisão e os valores de 5 a 7 assume que o entrevistado concorda com a afirmação.

9.3.4 Análise dos Dados

Os cálculos estatísticos foram realizados utilizando o software *Microsoft Excel*. Para avaliar as hipóteses, foram gerados os gráficos dos valores retirados do questionário, bem como realizado o cálculo do Coeficiente de Correlação Linear de Pearson. Este método é usualmente utilizado para medir a correlação entre duas variáveis. Seu valor (ρ) varia de -1 a

1, onde -1 indica uma perfeita correlação inversa entre as variáveis, e 1 indica uma perfeita correlação positiva entre as variáveis. O valor 0 indica que não há correlação linear entre as variáveis. No entanto, podem ocorrer diferentes valores entre -1 e 1. Segundo Callegari-Jacques (2003), o coeficiente de correlação pode ser descrito de forma qualitativa, como apresentado a seguir:

- se $0,00 < \rho < 0,30$, existe uma correlação linear fraca entre as variáveis;
- se $0,30 \leq \rho < 0,60$, existe uma correlação linear moderada entre as variáveis;
- se $0,60 \leq \rho < 0,90$, existe uma correlação linear forte entre as variáveis;
- se $0,90 \leq \rho < 1,00$, existe uma correlação linear muito forte entre as variáveis.

O Coeficiente de Correlação de Pearson pode ser calculado segundo a equação a seguir:

$$\rho = \frac{\sum_{i=1}^n (x_i - \bar{x}) \cdot (y_i - \bar{y})}{\sqrt{\sum_{i=1}^n (x_i - \bar{x})^2} \cdot \sqrt{\sum_{i=1}^n (y_i - \bar{y})^2}}$$

O Coeficiente de Variação (c_v) explica o quanto a variação de uma variável influencia na variação de outra. Para obter o c_v , basta calcular ρ elevado ao quadrado. Ainda é possível ser expresso na forma de percentagem se multiplicado por 100.

$$c_v = \rho^2 * 100$$

9.4 Apresentação dos Resultados

Para realizar os testes foi construída uma tabela com base nas respostas obtidas pelo questionário descrito na Tabela 9.

Tabela 10 - Conjunto dos dados obtidos do questionário.

Alunos	FUP	UP	IUP	Complexidade	Treinamento	Experiência com EaD
Aluno 1	5	5	6	4	7	Não
Aluno 2	5	5	5	5	5	Não
Aluno 3	5	5	5	6	7	Sim
Aluno 4	5	5	5	5	6	Não

Aluno 5	5	5	6	5	7	Sim
Aluno 6	6	7	7	6	7	Sim
Aluno 7	6	5	6	4	5	Sim
Aluno 8	6	5	5	6	6	Sim
Aluno 9	4	4	5	4	3	Sim
Aluno 10	6	6	6	6	6	Não
Aluno 11	5	5	6	6	6	Não
Aluno 12	4	4	4	6	5	Sim
Aluno 13	6	7	5	6	7	Não
Aluno 14	4	5	3	5	3	Não
Aluno 15	4	5	7	4	7	Não
Aluno 16	6	6	7	3	7	Sim
Aluno 17	5	4	6	6	4	Sim
Aluno 18	5	5	6	5	6	Não
Aluno 19	5	5	6	5	6	Não
Aluno 20	6	6	7	6	7	Não
Aluno 21	5	5	7	2	5	Sim

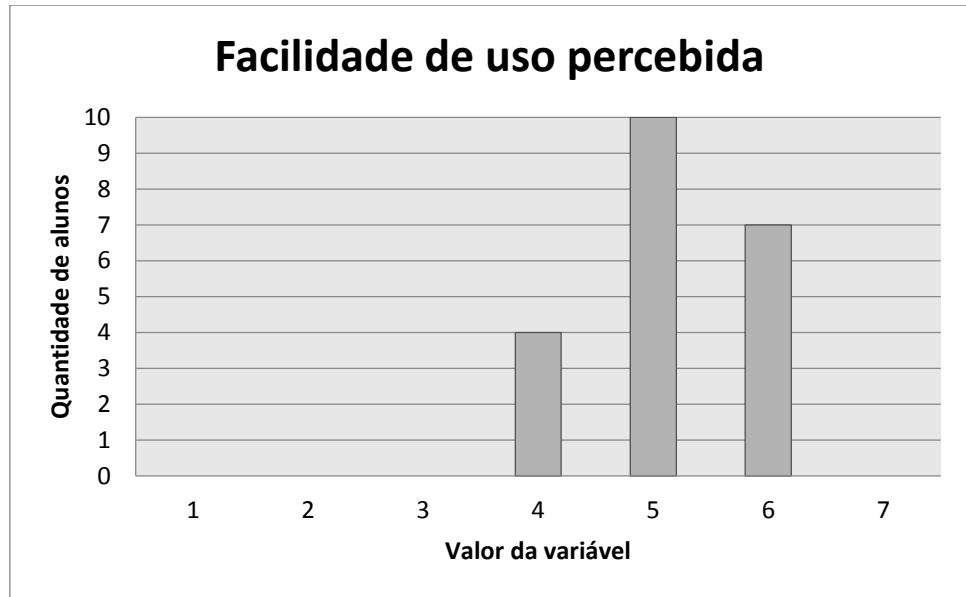
Fonte: Dados produzidos pelo autor.

A coluna FUP, contém a média das respostas das questões FUP1, FUP2, FUP3 e FUP5. Já a coluna UP, contém os valores da média das questões UP1, UP2, UP3 e UP4. Na coluna IUP, estão as médias dos valores relacionados às questões IUP1 e IUP2. A coluna Complexidade contém a média dos valores relacionados às questões VE1 e VE2. Na coluna Treinamento, estão contidos os valores das médias das questões VE3 e VE4. E, por fim, a coluna Experiência com EAD contém os valores referentes à questão VE4.

Com base nos dados obtidos da Tabela 10, pode-se observar que poucas foram as ocasiões em que os entrevistados reagiram de forma negativa ou indecisa.

Na Figura 30, é possível observar que a maioria dos usuários considerou a ferramenta de fácil utilização. Alguns usuários ficaram indecisos com relação à facilidade de uso percebida. Contudo, pode-se dizer que a ferramenta se mostrou simples de ser utilizada.

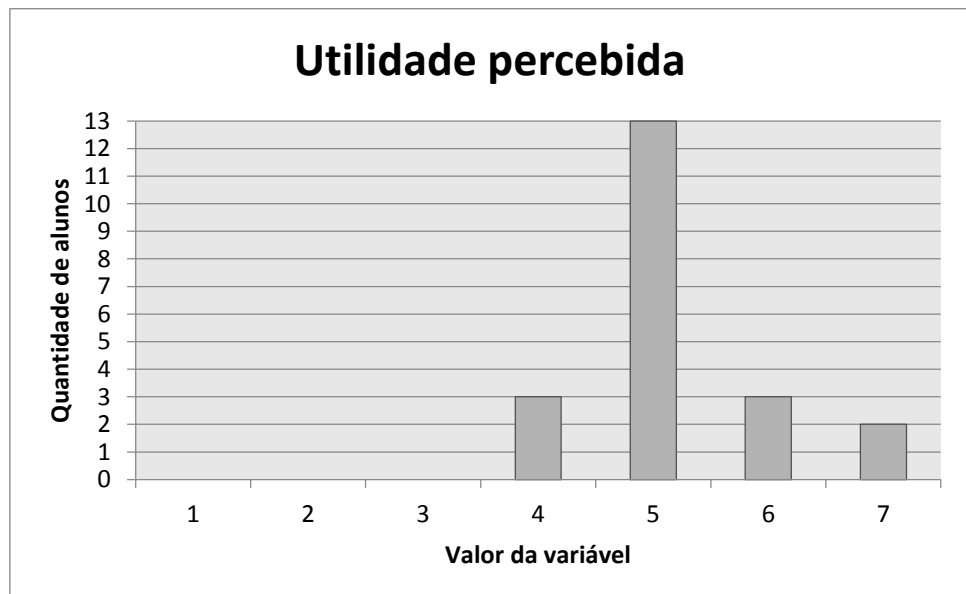
Figura 30 - Gráfico das médias das notas relacionadas à facilidade de uso percebida.



Fonte: Dados produzidos pelo autor.

Outra variável que teve retorno positivo foi com relação à utilidade percebida. Grande parte dos usuários considerou a ferramenta útil para se aplicar a PBL na EaD, como pode ser visto na Figura 31.

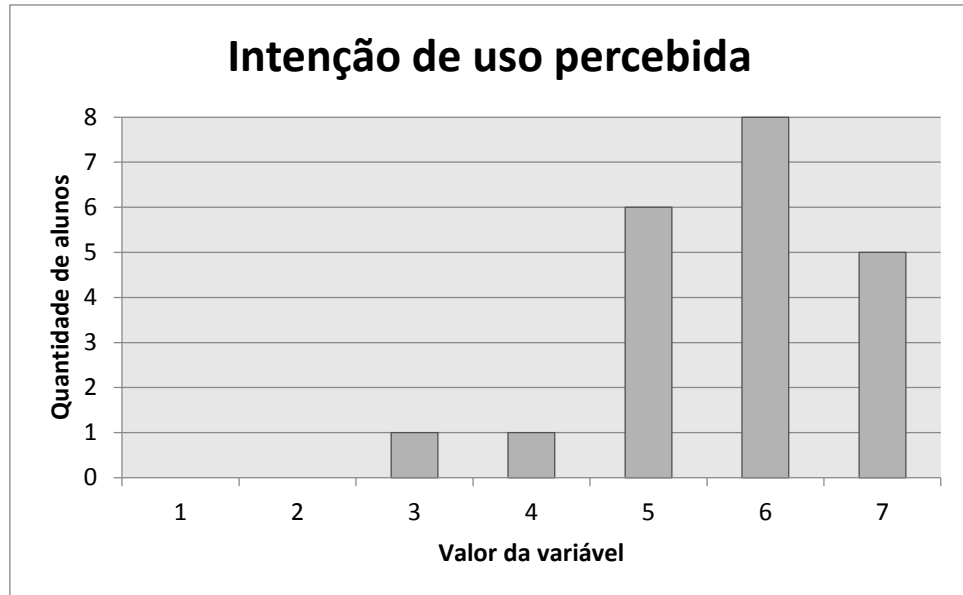
Figura 31 - Gráfico das médias das notas relacionadas à utilidade percebida.



Fonte: Dados produzidos pelo autor.

Com base na Figura 32, é possível observar que a maioria dos entrevistados demonstrou interesse em utilizar a ferramenta. Apenas dois dos entrevistados não concordaram que utilizaria a ferramenta em outras ocasiões.

Figura 32 - Gráfico das médias das notas relacionadas à intenção de uso percebida.

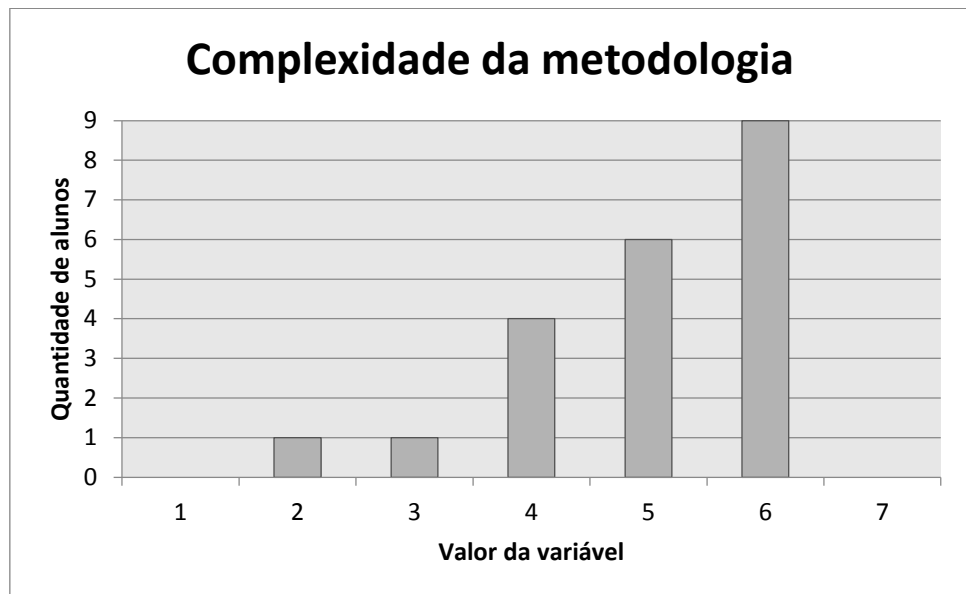


Fonte: Dados produzidos pelo autor.

A complexidade relacionada à metodologia PBL foi o alvo de maior cuidado. Essa preocupação foi justificada com base nos resultados obtidos das questões relacionadas a essa variável.

Conforme o gráfico da Figura 33, alguns usuários demonstraram uma leve dificuldade em assimilar a metodologia. Outros usuários ficaram indecisos com relação à complexidade. Esse dado é importante para que mais esforços sejam apontados para minimizar os impactos causados por essa variável. É importante esclarecer que quanto maior o valor dado para a complexidade, menos o usuário considera a metodologia complexa.

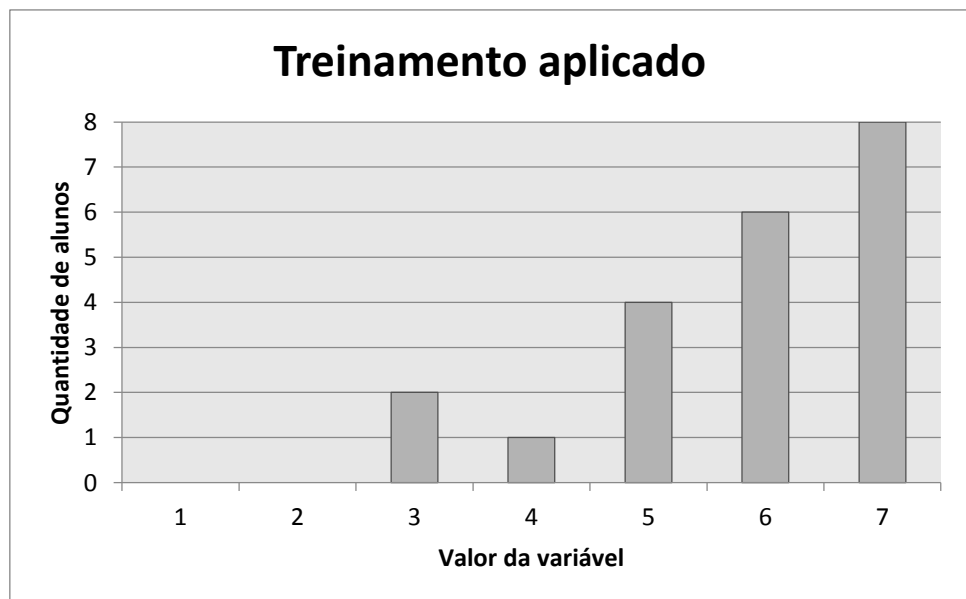
Figura 33 - Gráfico das médias das notas relacionadas à complexidade da metodologia.



Fonte: Dados produzidos pelo autor.

Com relação ao treinamento aplicado, a maioria dos usuários concordou que o treinamento foi adequado. No entanto alguns entrevistados ainda apresentaram dificuldades, como pode ser visto na Figura 34.

Figura 34 - Gráfico das médias das notas relacionadas ao treinamento aplicado.



Fonte: Dados produzidos pelo autor.

Além de observar os gráficos, pode-se também avaliar a relação entre as variáveis. A Tabela 11 apresenta os dados obtidos dos cálculos realizados para obter os coeficientes de correlação e variação de Pearson.

Tabela 11 - Resultados dos cálculos dos coeficientes de correlação e variação de Pearson.

Variáveis	ρ	c_v
Treinamento x FUP	0,54	29,66
Complexidade x FUP	0,16	2,58
FUP x UP	0,62	38,24
FUP x IUP	0,53	27,82
UP x IUP	0,37	13,97

Fonte: Dados produzidos pelo autor.

Analisando os valores da Tabela 11, pode-se realizar o teste das hipóteses definidas para o estudo de caso. A seguir será apresentada a verificação de cada uma dessas hipóteses.

Hipótese 1: o treinamento dado ao facilitador e aos estudantes influencia na facilidade de uso do sistema percebida pelo usuário.

Como pode ser observado na Tabela 11, o coeficiente de correlação para o treinamento e a facilidade de uso é de 0,54. Isso significa que existe uma correlação moderada entre essas duas variáveis.

Pode-se dizer que o treinamento dado aos estudantes explica em 29,66% a facilidade de uso percebida.

Hipótese 2: a complexidade que envolve o entendimento da metodologia influencia diretamente na facilidade de uso do sistema percebida pelo usuário.

O coeficiente de correlação calculado para a complexidade da metodologia e a facilidade de uso é de 0,16. Esse valor representa uma correlação fraca entre as variáveis. Com base no coeficiente de variação, é possível dizer que a complexidade atrelada a metodologia explica em apenas 2,58% a variação da facilidade de uso percebida.

Hipótese 3: a experiência influencia na facilidade de uso do sistema percebida pelo usuário.

Considerando que todos os usuários utilizaram o sistema pela primeira vez durante a validação, pôde-se observar, como mostram os dados descritos no gráfico exibido na Figura 30, que a falta de experiência com a ferramenta não dificultou a sua utilização. No entanto, alguns usuários concordam que utilizar o módulo fica mais fácil à medida que é usado. Como relata um dos entrevistados dizendo:

Recurso interessante, mas deve ser utilizado com certa frequência para um melhor aproveitamento.

Hipótese 4: a facilidade de uso percebida pelo usuário influencia na utilidade percebida pelo usuário.

Pode-se observar também na Tabela 11 que o coeficiente de correlação entre as variáveis facilidade de uso e utilidade foi de 0,62. Esse dado representa uma correlação forte entre as variáveis. Ou seja, a facilidade de uso percebida pelo usuário explica em 38,24% a variação da utilidade percebida.

Hipótese 5: a utilidade percebida influencia a intenção de uso da ferramenta pelo usuário.

O coeficiente de correlação entre as variáveis utilidade percebida e intenção de uso foi de 0,37. Esse valor indica uma correlação moderada entre as variáveis. Ou seja, a variação da utilidade percebida explica em 13,97% a variação da intenção de uso percebida pelo usuário.

Hipótese 6: a facilidade de uso percebida influencia a intenção de uso da ferramenta pelo usuário.

Com relação à facilidade de uso e a intenção de uso, obteve-se o coeficiente de correlação 0,53. Isso significa que a correlação entre essas variáveis é moderada, ou ainda, a facilidade de uso percebida pelo usuário explica em 27,82% a intenção de uso.

10 CONSIDERAÇÕES FINAIS E TRABALHOS FUTUROS

Neste trabalho foi apresentada uma solução para a aplicação da PBL no MOODLE, utilizando uma abordagem multiagente para apoiar os estudantes e o facilitador durante o processo de implementação da PBL, bem como um estudo de caso para validar a solução.

Para a realização do estudo, foi utilizada a metodologia TAM, proposta por Davis (1986), na intenção de verificar a aceitação da ferramenta por um grupo de alunos que estavam cursando a disciplina Análise e Projeto de Sistemas na UFERSA e Computação e Música na UERN.

Foi construído um formulário contendo 20 questões e disponibilizado para os alunos através da ferramenta *Google Forms*. Após todos os alunos responderem o questionário, os dados foram colhidos, abstraídos e analisados através da ferramenta Microsoft Excel.

A partir dos resultados obtidos no estudo de caso, foi possível perceber que a ferramenta se mostrou eficiente durante sua aplicação. Os entrevistados demonstraram facilidade ao utilizar a ferramenta e a caracterizaram como uma boa solução para se aplicar a PBL na EaD. Também foi possível identificar uma alta aceitação da ferramenta, como pode ser observado na experiência descrita a seguir por um dos entrevistados:

A experiência foi bastante proveitosa, uma vez que a ferramenta virtual facilita as reuniões do grupo, no horário mais propício para cada um, sem necessitar de tomar muito o tempo, que estaria sendo dedicado a outras atividades. As sessões curtas são estimulantes, pois evitam a fadiga e fazem com que nos foquemos precisamente na resolução da questão.

Por fim, foi possível perceber que a utilidade percebida pelos usuários está fortemente relacionada à facilidade de uso da ferramenta, como mostraram os dados obtidos nos testes de hipóteses discutidas anteriormente. Alguns entrevistados apresentaram ainda dificuldade em compreender o funcionamento da metodologia. Esse fator tem uma relevância moderada na facilidade de uso. Entretanto, esse obstáculo não atrapalhou a experiência dos entrevistados, como pode ser visto a seguir:

Sobre a aprendizagem baseada em problemas por ser uma experiência nova não aceitei muito inicialmente, mais depois vi que é uma boa forma de estimular ao aluno para que ele construa seu próprio conhecimento.

Como proposta para trabalhos futuros, pretende-se implementar todas as tarefas contidas no modelo de tarefas, ilustrado na Figura 9, que não foram contempladas neste trabalho.

REFERÊNCIAS

- ABED. Associação Brasileira de Educação a Distância. Site oficial da ABED. 2013. Disponível em: <http://www.abed.org.br/censoead/censoEAD.BR_2012_pt.pdf>. Acesso em: 01 mai. 2016.
- AKINOĞLU, O.; TANDOĞAN, R. O. The Effects of Problem-Based Active Learning in Science Education on Students' Academic Achievement, Attitude and Concept Learning. *Eurasia Journal of Mathematics, Science & Technology Education*, 2007.
- ARTERO, A. O. Inteligência Artificial - Teoria e Prática. 1ª. ed. São Paulo: Livraria da Física, 2009.
- BELIFEMINE, F.; CAIRE, G.; GREENWOOD, D. Developing multi-agent systems with JADE. Liverpool, Inglaterra: John Wiley& Sons, Ltd, 2007.
- CALLEGARI-JACQUES, S. M. Bioestatística: princípios e aplicações. Porto Alegre: Artemed, 2003. 255p.
- CAPELETTI, A. M. Ensino a Distância: Desafios Encontrados por Alunos do Ensino Superior. *Revista Eletrônica Saberes da Educação*, São Roque, v. 5, n. 1, 2014.
- CASTRO, J.; GIORGINI, P.; KOLP, M.; MYLOPOULOS, J. Tropos: A requirements-driven methodology for agent-oriented software. In: HENDERSON-SELLERS, B.; GIORGINI, P. *Agent-Oriented Methodologies*. Hershey, PA: IDEA Group Publishing, 2005. Cap. II, p. 20-45.
- CERVENKA, R.; TRENCANSKY, I. AML, The Agent Modeling Language: A Comprehensive Approach to Modeling Multi-Agent Systems. Basel, Suíça: Birkhauser Verlag, v. Whitestein Series in Software Agent Technologies and Automatic Computing, 2007.
- CHERMANN, M., BONINI M. L. Educação à distância: novas tecnologias em ambientes de aprendizagem pela internet. São Paulo: EPN Editoria e Projetos S/C, 2000.
- CONSENTINO, M. From requirements to code with the passi methodology. In: HENDERSON-SELLERS, B.; GIORGINI, P. *Agent-Oriented Methodologies*. Hershey, PA: IDEA Group Publishing, 2005. Cap. IV, p. 79-106.
- DAVIS, F. D. Perceived usefulness, perceived ease of use, and user acceptance of information technology. *MIS Quarterly*, Minneapolis (MN), v. 13, n. 3, p. 319-339, 1989.
- DELOACH, S.; KUMAR, M. Multiagent systems engineering: An overview and case study. In: HENDERSON-SELLERS, B.; GIORGINI, P. *Agent-Oriented Methodologies*. Hershey, PA: IDEA Group Publishing, 2005. Cap. XI, p. 317-340.
- FIPA. Welcome to the Foundation for Intelligent Physical Agents. Site Oficial do Padrão FIPA, 2011. Disponível em: <<http://www.fipa.org/>>. Acesso em: 08 jan. 2016.

FONTES, L. M. O. Uma arquitetura multiagente de apoio à aprendizagem baseada em problema. 2013. 153 f. Dissertação (Mestrado em Ciência da Computação) - Universidade Estadual do Rio Grande do Norte, Universidade Federal Rural do Semi-Árido, Mossoró, 2013.

FREITAS, K. S. Um panorama geral sobre a história do ensino a distância. In: ARAUJO, B.; FREITAS, K. S. (Org.). Educação a distância no contexto brasileiro: algumas experiências da UFBA. v. 1. Salvador: ISP/UFBA, 2005. p. 57-68.

GABRIEL, C. P. C. Aplicação da Lógica Fuzzy para Avaliação do Faturamento do Consumo de Energia Elétrica e Demanda de uma Empresa de Avicultura de Postura. 2008. 97 f. Tese (Doutorado em Agronomia) - Faculdade de Ciências Agronômicas da UNESP. Bocatú, 2008.

GARIJO, F. J.; GÓMEZ-SANZ, J. J.; MASSONET, P. The MESSAGE Methodology for Agent-Oriented Analysis and Design. In: HENDERSON-SELLERS, B.; GIORGINI, P. Agent-Oriented Methodologies. Hershey, PA: IDEA Group Publishing, 2005. Cap. VIII, p. 203-234.

GOMES, A. K.; BERNADINI, F. C.; MONARD, M. C.; BATISTA, G. E. A. P. A. Uma Sintaxe Padrão Prolog para Classificadores Simbólicos. São Carlos: n. 154, 2002.

GOMIDE, F. A. C.; GUDWIN, R. R. Modelagem, Controle, Sistemas e Lógica Difusa. Controle & Automação. v. 4, n. 3. Campinas: UNICAMP, 1994. p. 97-115.

GOOGLE FORMS. Formulários Google: crie e analise pesquisas gratuitamente. Site oficial do Google Forms. 2016. Disponível em: <<https://www.google.com/forms/about/>>. Acesso em: 03 mai. 2016.

GPL. The GNU General Public License v3.0 - GNU Project - Free Software Foundation (FSF). GNU General Public License, 2007. Disponível em: <<http://www.gnu.org/copyleft/gpl.html>>. Acesso em: 25jan. 2016.

HENDERSON-SELLERS, B.; GIORGINI, P. Agent-Oriented Methodologies. Guernsey, Reino Unido: IGI Global, 2005.

HMELO-SILVER, C. E. Problem-Based Learning: What and How Do Students Learn? Educational Psychology Review, v. 16, n. 3, 2004.

HMELO-SILVER, C. E.; BARROWS, H. S. Goals and strategies of a problem-based learning facilitator. The Interdisciplinary Journal of Problem-based Learning, v. 1, n. 1, p. 21-39, 2006. Disponível em: <<http://docs.lib.purdue.edu/ijpbl/vol1/iss1/4/>>. Acesso em: 02 fev. 2016.

IBRAHIM, A. M. Fuzzy Logic for Embedded Systems applications. Elsevier Science (USA), 2003. 287 f.

IGLESIAS, C. A.; GARIJO, M. The Agent-Oriented Methodology MAS-CommonKADS. In: HENDERSON-SELLERS, B.; GIORGINI, P. Agent-Oriented Methodologies. Hershey, PA: IDEA Group Publishing, 2005. Cap. III, p. 46-78.

INEP. Instituto Nacional de Estudos e Pesquisas Educacionais Anísio Teixeira. Site oficial do INEP. 2014. Disponível em: <http://portal.inep.gov.br/visualizar/-/asset_publisher/6AhJ/content/matriculas-no-ensino-superior-crescem-3-8>. Acesso em: 08 jan. 2016.

JADE. JAVA Agent DEvelopment Framework: is an open source platform for peer-to-peer agent based applications. Site Oficial do JADE, 2014. Disponível em: <<http://jade.tilab.com/>>. Acesso em: 20 jan. 2016.

ORACLE. Software Java: Oracle BR. Site Oficial do JAVA, 2016. Disponível em: <<http://www.oracle.com/br/java/overview/index.html>>. Acesso em: 30 Mai. 2016.

LEMGRUBER, M.S. Educação à distância: expansão, regulamentação e mediação docente. In: Educ. foco, Juiz de Fora, v. 14, n. 1, p. 145-159, mar/ago 2009.

LINDEN, R. Algoritmos Genéticos - Uma importante ferramenta da Inteligência Computacional. 2ª.ed. Rio de Janeiro, RJ: Brasport, 2008.

MANDAL, S. Problem Based Learning Tool as a Plug-in for Moodle. 2011. 111 f. Dissertação (Mestrado em Ciência da Computação e Engenharia) – Indian Institute of Technology, Bombay, 2011.

MARRO, A. A.; SOUZA, A. M. C.; CAVALCANTE, E. R. S.; BEZERRA, G. S.; NUNES, R. O. Lógica Fuzzy: conceitos e aplicações. Universidade Federal do Rio Grande do Norte (UFRN) Natal – RN – Brasil, 2010.

MELO-SOLARTE, D. S.; BARANAUSKAS, M. C. C. Uma Abordagem para EaD Baseada em Resolução de Problemas. In: SBIE 2008 - Simpósio Brasileiro de Informática na Educação. Fortaleza. Anais do SBIE 2008, XIX Simpósio Brasileiro de Informática na Educação. p. 716 - 725, v. XIX. Disponível em <<http://www.br-ie.org/pub/index.php/sbie/article/viewFile/760/746>>. Acesso em: 14 mar. 2016.

MENDES NETO, F. M. E-grupo: Um ambiente para suporte à aprendizagem colaborativa baseada na web. 2000. 147 f. Dissertação (Mestrado em Informática) - Universidade Federal da Paraíba, Campina Grande, 2000.

MICROSOFT EXCEL. Microsoft Excel: software de criação de planilha. Site oficial do Microsoft Excel. 2016. Disponível em: <<https://products.office.com/pt-br/excel>>. Acesso em: 02 mai. 2016.

MOODLE. About Moodle. Site oficial do Moodle, 2015. Disponível em: <http://docs.moodle.org/29/en/About_Moodle>. Acesso em: 14 mar. 2016.

MORAIS II, M. J. D. O. MAS-CommonKADS+: Uma Extensão à Metodologia Mas-CommonKADS para Suporte ao Processo Detalhado de Sistemas Multiagentes Racionais. Dissertação de Mestrado. Universidade Estadual do Ceará - UECE. Fortaleza, CE. 2010.

MYSQL. MySQL: The world's most popular open source database. Site oficial do MySQL, 2011. Disponível em: <<http://www.mysql.com/>>. Acesso em: 18 fev. 2016.

NETBEANS. NetBeans IDE - The Smarter and Faster Way to Code. Site Oficial do NetBeans, 2015. Disponível em <<https://netbeans.org/features/index.html>>. Acesso em: 18 fev. 2016.

OLIVEIRA, P. R. D. Uma avaliação de metodologias para validação de software. 2015. 68 f. Monografia (Graduação em Ciência da Computação) - Universidade Federal Rural do Semi-Árido, Mossoró, 2015.

PAVÓN, J.; GOMEZ-SANZ, J. J.; FUENTES, R. The INGENIAS Methodology and Tools. In: HENDERSON-SELLERS, B.; GIORGINI, P. Agent-Oriented Methodologies. Hershey, PA: IDEA Group Publishing, 2005. Cap. IX, p. 236-276.

PETROLI NETO, S. Computação Evolutiva: desvendando os algoritmos genéticos. Ubiquidade - Revista de Estudos de Tecnologia da Informação e Comunicações, v. 1, n.1, p. 34-45, 2011.

PHP. PHP: Hypertext Preprocessor. Site Oficial do PHP, 2011. Disponível em: <<http://www.php.net/>>. Acesso em: 18 dez. 2015.

PINTO, G. R. P. R.; SENA, C. P. P.; COSTA, R. A.; SILVA FILHO, S. S.; PEREIRA, H. B. B. PBL-VE: Um ambiente virtual para apoiar a aprendizagem baseada em problemas. In: XXXIX Congresso Brasileiro de Educação em Engenharia, 2011, Blumenau. Formação Continuada e Internalização. Blumenau: ODORIZZI, 2011.

PONTES, A. A. A. Uma Arquitetura de Agentes para Suporte à Colaboração na Aprendizagem Baseada em Problemas em Ambientes Virtuais de Aprendizagem. 2010. 114 f. Dissertação (Mestrado em Ciência da Computação) - Universidade Estadual do Rio Grande do Norte, Universidade Federal Rural do Semi-Árido, Mossoró, 2010.

REATEGUI, E.; MORAES, M. Agentes Pedagógicos Animados: Concepção, Desenvolvimento e Aplicação. In: ANAIS DO SIMPÓSIO BRASILEIRO DE INFORMÁTICA NA EDUCAÇÃO – SBIE, 17. Brasília: SBC, 2006.

ROUGEMAILLE, S.; ARCANGELI, J. P.; GLEIZES, M. P.; MIGEON, F. ADELFE Design, AMAS-ML in Action. In: ARTIKIS, A. A. P. G. A. V. L. Engineering Societies in the Agents World IX. Berlin, Heidelberg: Springer-Verlag, 2009. p. 105-120.

RUSSEL, S. J.; NORVIG, P. Artificial Intelligence: A Modern Approach. 2ª Edição. Upper Saddle River, Nova Jersey: Prentice Hall, 2003.

SANCHO, P.; TORRENTE, J.; MARCHIORI, E. J.; FERNÁNDEZ-MANJÓN, B. Enhancing Moodle to Support Problem Based Learning: The Nucleo Experience. In: IEEE Engineering Conference (EDUCON). 2011. Disponível em <http://www.e-ucm.es/drafts/e-UCM_draft_175.pdf>. Acesso em: 18 mai. 2016.

SANTOS, D.; REZENDE, G.; BERTONI, F.; SENA, C.; BITTENCOURT, R. Aplicação do método de Aprendizagem Baseada em Problemas no curso de Engenharia de Computação da Universidade Estadual de Feira de Santana. In Anais do XXXV Congresso Brasileiro de Educação em Engenharia. Curitiba, 2007.

SAVERY, J. R. Overview of Problem-based Learning: Definitions and Distinctions. *The Interdisciplinary Journal of Problem-based Learning*, v. 1, n. 1, p. 9-20, 2006.

SCHREIBER, G.; AKKERMANS, H.; ANJEWIERDEN, A.; HOOG, R.; SHADBOLT, N.; VELDE, W. V.; WIELINGA, B. *Knowledge Engineering and Management - The CommonKADS Methodology*. Cambridge, MA: MIT Press, 2000.

SCORM. ADL SCORM. Site Oficial do SCORM, 2013. Disponível em <<http://www.adlnet.gov/capabilities/scorm>>. Acesso em: 18 jan. 2016.

SCHUELTER, W. Ambiente virtual de aprendizagem: reflexões sobre as mudanças na metodologia de ensino e o papel do professor. Universidade de Santa Catarina – UNISUL- 05/2005 155-TC-C5.

SILVA, L. C. N. *MobiLE - Um Ambiente Multiagente de Aprendizagem Móvel para Apoiar a Recomendação Ubíqua de Objetos de Aprendizagem*. 2012. 108 f. Dissertação (Mestrado em Ciência da Computação) - Universidade Estadual do Rio Grande do Norte, Universidade Federal Rural do Semi-Árido, Mossoró, 2012.

SILVA, T. G.; BERNARDI, G. Cal: um Agente Pedagógico Animado para Apoio em um Objeto de Aprendizagem para o Ensino de Matemática. In: ANAIS DO SIMPÓSIO BRASILEIRO DE INFORMÁTICA NA EDUCAÇÃO – SBIE, 20. Florianópolis: SBIE, 2009.

SOLARTE, D. S. M. *Aprendizagem Colaborativa Baseada em Problemas: ACBP Modelo Conceitual e Ferramentas*. 2009. 107 f. Dissertação (Mestrado em Ciência da Computação) - UNICAMP, Campinas, 2009.

SOLIMAN, M.; GUETL, C. Intelligent Pedagogical Agents in Immersive Virtual Learning Environments: A Review. In: PROCEEDINGS OF THE INTERNATIONAL CONVENTION - MIPRO. Opatija, Croatia, p. 827-832, 2010.

VERMEERSCH, J. et al. *Iniciação ao Ensino a Distância*. Bruxelas: Het Gemeenschapsonderwijs, 2006. 118 f.

VERONESE, G.; CORREA, A.; WERNER, C.; JEZINI NETTO, F. ARES: Uma Ferramenta de Engenharia Reversa Java-UML. In: SIMPÓSIO BRASILEIRO DE ENGENHARIA DE SOFTWARE – SBES, 16. Gramado: p. 347-352, 2002.

YU, E. *Modelling strategic relationships for process reengineering*. Tese de Doutorado. University of Toronto. Toronto, Canadá. 1995.

ZAMBONELLI, F.; JENNINGS, N.; WOOLDRIDGE, M. Multi-agent systems as computational organizations: The gaia methodology. In: HENDERSON-SELLERS, B.; GIORGINI, P. *Agent-Oriented Methodologies*. Hershey, PA: IDEA Group Publishing, 2005. Cap. VI, p. 136-171.

ZINI, E. D. O. C. *Algoritmo Genético Especializado na Resolução de Problemas com Variáveis Contínuas e Altamente Restritos*. 2009. 149 f. Dissertação (Mestrado em Engenharia Elétrica) - Universidade Estadual Paulista, Ilha Solteira, SP, 2009.

APÊNDICE A

A seguir é apresentado o código fonte do algoritmo fuzzy de geração do rank de grupos em Prolog.

```
% Para realizar uma consulta:
% ?- agente_forma_grupo('Percepcoes.txt','Acoes.txt').
agente_forma_grupo(Percepcoes,Acoes):-
ver(Percepcoes),
ação(Acoes).
/*****
***
*****/
ver(Percepcoes):-
see(Percepcoes),
vê_quantidade_alunos,
vê_parâmetros_similaridade,
vê_situação_desejada_e_Importâncias,
vê_situações_correntes,
seen.
:- dynamicquantidade_alunos/1.
vê_quantidade_alunos:-
read(Q),
asserta(quantidade_alunos(Q)).
:- dynamicparâmetros_similaridade/2.

vê_parâmetros_similaridade:-
read(P),
read(N),
asserta(parâmetros_similaridade(P,N)).
:- dynamicsituação_desejada/1.
:- dynamicImportâncias/1.
vê_situação_desejada_e_Importâncias:-
read(SD),
read(ValoresImportância),
quantizar(SD,NiveisDesejados),
asserta(situação_desejada(NiveisDesejados)),
asserta(Importâncias(ValoresImportância)).
:- dynamicaluno_candidato/3.
vê_situações_correntes:-
read(Aluno),
Aluno = [ID_Aluno, Nome, SC],!,
quantizar(SC,NiveisCorrentes),
asserta(aluno_candidato(ID_Aluno,Nome,NiveisCorrentes)),
vê_situações_correntes.
vê_situações_correntes.
/*****
***
*****/
quantizar([Char],[LReais]):-
!,converter(Char,LReais).
quantizar([Char|RLChar],[LReais|RLReais]):-
converter(Char,LReais),
quantizar(RLChar,RLReais).
```

```

converter(b,[1.0, 1.0, 0.9, 0.7, 0.5, 0.2, 0.0, 0.0, 0.0, 0.0, 0.0]).
converter(m,[0.0, 0.0, 0.0, 0.8, 0.9, 1.0, 0.7, 0.0, 0.0, 0.0, 0.0]).
converter(a,[0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.3, 0.5, 1.0, 1.0, 1.0]).
/*****
***
*****/
ação(Acoes):-
situação_desejada(NiveisDesejados),
Importâncias(ValoresImportância),
parâmetros_similaridade(P,N),
aval(NiveisDesejados,ValoresImportância,0,AlunosCandidatosAvaliados,P,N)
,
ordenar(AlunosCandidatosAvaliados,AlunosCandidatosOrdem),
salvar_limparBaseconhecimento(AlunosCandidatosOrdem,Acoes).
/*****
***
*****/
aval(_,_Q,_,_):-
quantidade_alunos(Q),!.
aval(NiveisDesejados,ValoresImportancia,X,[candidato(ID_Aluno,Nome,Aval)
|R]
,P,N):-
ID_Alunois X+1,
aluno_candidato(ID_Aluno,Nome,NiveisCorrentes),
Diferençass(NiveisDesejados,NiveisCorrentes,Diferençass,P,N),
f_avalia(Diferençass,ValoresImportancia,Aval),
aval(NiveisDesejados,ValoresImportancia,ID_Aluno,R,P,N).
/*****
***
*****/
Diferençass([NivelDesejado],[NivelCorrente],[Diferenças],P,N):-!,
Diferenças(NivelDesejado,NivelCorrente,Diferenças,P,N).
Diferençass([NivelDesejado|NiveisDesejados],[NivelCorrente|NiveisCorrentes]
,[Diferenças|Diferençass],P,N):-
Diferenças(NivelDesejado,NivelCorrente,Diferenças,P,N),
Diferençass(NiveisDesejados,NiveisCorrentes,Diferençass,P,N).
/*****
***
*****/
Diferenças(NivelDesejado,NivelCorrente,Diferenças,P,N):-
pares_valores_participação(NivelDesejado,NivelCorrente,L1),
subtrações_pares(L1,L2),
modulos(L2,L3),
elevados(L3,L4,P),
somatoria(L4,S),
E is 1/P,
E1 is S**E,
E2 is N**E,
Diferenças is E1/E2.
pares_valores_participação([],[],[]):-!.
pares_valores_participação([X|Y],[Z|W],[par(X,Z)|L]):-
pares_valores_participação(Y,W,L).
subtrações_pares([],[]):-!.
subtrações_pares([par(A,B)|Y],[Z|W]):-
Z is A-B,
subtrações_pares(Y,W).
modulos([],[]):-!.
modulos([X|Y],[Z|W]):-
modulo(X,Z),
modulos(Y,W).

```

```

modulo(N,N):-
N>=0.0,! .
modulo(N,M):-
N <0.0,
M is (-1.0)*N.
elevados([],[],_):-!.
elevados([X|Y],[Z|W],P):-
Z is X**P,
elevados(Y,W,P).
somatoria(L,S):-
somat(L,0.0,S).
somat([],Acum,Acum):-!.
somat([X|Y],Acum,S):-
Acum_auxis X + Acum,
somat(Y,Acum_aux,S).
/*****
***
*****/
f_avalua(Diferenças,ValoresImportancia,Aval):-
um_menos(Diferenças,Similaridades),
soma_ponderada(ValoresImportancia,Similaridades,Avalaux),
somatoria(ValoresImportancia,Somaimp),
Aval isAvalaux/Somaimp.
um_menos([X],[Y]):-!, Y is1-X.
um_menos([X|Y],[Z|W]):-
Z is1-X,
um_menos(Y,W).
soma_ponderada(Imp,Sim,Aval):- s_p(Imp,Sim,0,Aval).
s_p([],[],Acum,Aval):-!, Aval = Acum.
s_p([Imp|RImp],[Sim|RSim],Acum,Aval):-
AcumauxisAcum+ Imp*Sim,
s_p(RImp,RSim,Acumaux,Aval).
/*****
***
*****/
ordenar([],[]):-!.
ordenar(Lava,[candidato(ID_Aluno,Nome,Aval)|RLava_ordenada]):-
buscar_maximo(Lava,candidato(ID_Aluno,Nome,Aval)),
eliminar_aluno(candidato(ID_Aluno,Nome,Aval),Lava,L1),
ordenar(L1,RLava_ordenada).
buscar_maximo([candidato(ID_Aluno,Nome,Aval)],candidato(ID_Aluno,Nome,Aval):-!.
buscar_maximo([candidato(ID_Aluno,Nome,Aval),candidato(_,A)|RLava],Min):-
A =<Aval,!,
buscar_maximo([candidato(ID_Aluno,Nome,Aval)|RLava],Min).
buscar_maximo([_,candidato(ID_Aluno,Nome,Aval)|RLava],Min):-
buscar_maximo([candidato(ID_Aluno,Nome,Aval)|RLava],Min).
eliminar_aluno(O,[O|N],N):-!.
eliminar_aluno(O,[X|Y],[X|Z]):-
eliminar_aluno(O,Y,Z).
/*****
***
*****/
salvar_limparBaseconhecimento(AlunosCandidatosOrdem,Acoes):-
tell(Acoes),
imprime(AlunosCandidatosOrdem),
told,
retract(quantidade_alunos(_)),
retract(parâmetros_similaridade(_, _)),

```

```
retract(situação_desejada(_)),  
retract(Importâncias(_)),  
abolish(aluno_candidato/3).  
imprime([]):-!.  
imprime([A|RA]):-  
write(A),nl,  
imprime(RA).
```