



# UbiGroup: Um Modelo de Recomendação Ubíqua de Conteúdo para Grupos Dinâmicos de Aprendizes

*UbiGroup: A Model of Ubiquitous Recommendation of Content for Dynamic Groups of Learners*

**Luis Gustavo Araujo Ferreira**

Universidade do Vale do Rio dos Sinos (Unisinos)  
Av. Unisinos, 950, São Leopoldo – RS Brasil  
[luis.gustavo.af@gmail.com](mailto:luis.gustavo.af@gmail.com)

**Jorge Luis Victória Barbosa**

Universidade do Vale do Rio dos Sinos (Unisinos)  
Av. Unisinos, 950, São Leopoldo – RS – Brasil  
[jbarbosa@unisinos.br](mailto:jbarbosa@unisinos.br)

**João Carlos Gluz**

Universidade do Vale do Rio dos Sinos (Unisinos)  
Av. Unisinos, 950, São Leopoldo – RS – Brasil  
[jcgluz@unisinos.br](mailto:jcgluz@unisinos.br)

**Rosa Vicari**

Universidade Federal do Rio Grande do Sul (UFRGS)  
Instituto de Informática - Caixa Postal 15064  
Porto Alegre – RS – Brasil  
[rosa@inf.ufrgs.br](mailto:rosa@inf.ufrgs.br)

## Resumo

*A necessidade do professor buscar e selecionar materiais educacionais adequados para sua turma é um fato comum no meio educacional. Entretanto, a grande disponibilidade de materiais, a heterogeneidade dos perfis dos alunos e a diversidade de atividades pedagógicas que podem ser realizadas, tornam esta tarefa bastante trabalhosa. Neste cenário, este trabalho apresenta um modelo de recomendação ubíqua de conteúdo educacional para grupos de aprendizes criados dinamicamente, que visa auxiliar o professor no processo de busca e seleção de materiais educacionais levando em conta os perfis dos alunos e o contexto onde eles estão inseridos. A estratégia adotada neste trabalho se diferencia dos trabalhos relacionados por efetuar a recomendação de materiais educacionais considerando de forma conjunta os perfis de um grupo de aprendizes, a dinamicidade na formação destes grupos e o contexto no qual eles se encontram. Com base em uma validação por cenários foi possível verificar a viabilidade do modelo.*

**Palavras-Chave:** *Recomendação para Grupos. Sensibilidade a Contexto. Ontologia. Agente de Software.*

## Abstract

*The necessity of teachers to search and to select appropriate educational materials for their classes is a common fact in the educational environment. However, the wide availability of materials, the heterogeneity of the students' profiles and the diversity of pedagogical activities that can be conducted, make this task laborious. In this scenario, this work presents a model for ubiquitous recommendation of educational content for dynamically created groups of learners, which aims to help teachers to search and to select educational materials taking into consideration the profile of the group and the teaching context. The strategy adopted in this work differs from related work by making the recommendation of educational content considering jointly the profiles of group of learners, the dynamics of group formation and context in which they find themselves. Based on validation scenarios, it*

*was possible to verify the feasibility of the model.*

**Keywords:** *Recommendation to Groups. Context awareness. Ontology. Software Agent.*

## 1. Introdução

Atualmente, percebe-se um aumento dos recursos tecnológicos disponíveis para apoiar os alunos no seu processo de aprendizagem. Estes recursos também podem auxiliar os professores na organização das suas aulas. Existem diversas possibilidades de utilização pelo professor, no entanto, uma questão específica na utilização destes recursos é a constante busca de materiais educacionais para utilização junto aos alunos. Pelo fato de existirem muitos repositórios, com muitos materiais digitais educacionais, o trabalho de busca, bem como a avaliação destes, torna-se custoso e demorado para o professor.

Em uma consulta a dois repositórios de materiais, foi verificada uma quantidade expressiva de materiais disponíveis. No repositório BIOE [1] encontram-se disponíveis aproximadamente 17.000 materiais educacionais e mais 1.900 em avaliação. Já no repositório Merlot [2] são disponibilizados cerca de 38.000 materiais. Além destes, existem outros repositórios que também disponibilizam materiais educacionais, como por exemplo, RIVED [3] e Cesta [4].

Neste cenário, a busca de materiais educacionais se torna algo que demanda considerável esforço do professor, primeiro pelo fato de existirem muitos materiais disponíveis, segundo por estes materiais estarem distribuídos em diversos repositórios. A busca e a seleção manual destes materiais podem não ser efetivas ao ponto de atender as necessidades do professor.

Os materiais contidos nestes repositórios em sua maioria estão padronizados, sendo conceitualmente chamados de Objetos de Aprendizagem (OA) [5]. Existem diversos tipos de padronização para os OAs, como por exemplo, LOM [6], DCIM [7] e OBAA [8]. Pelo fato dos OAs estarem padronizados, é possível desenvolver ferramentas capazes de buscar estes materiais de forma sistemática. Este processo de busca normalmente está limitado à utilização de filtros de pesquisa por assunto, palavras-chave, autor, entre outras informações. Esta busca auxilia o professor a encontrar os materiais, mas não resolve o problema que consiste em analisar todos os materiais, verificando quais destes estão adequados ao seu planejamento pedagógico.

Analisando esta situação, pode-se considerar processos automatizados, que utilizem as Tecnologias atuais de Informação e Comunicação (TIC) para a busca e análise destes materiais, como alternativa para propiciar um ganho de tempo ao professor e uma maior qualidade na seleção de OAs.

Considerando a gama de informações disponíveis nos metadados dos OAs é possível identificar os materiais, por exemplo, por tipo de mídia, suporte a dispositivos, tipo de recurso de aprendizagem, grau de interatividade,

entre outras informações. Estas informações podem ser diretamente relacionadas ao perfil de aprendizado do grupo de alunos, deste modo, sendo possível identificar quais os OAs mais adequados aos alunos em questão.

Complementando esta abordagem, deve-se levar em conta a evolução do processo de aprendizado com os estudos na Educação Ubíqua [9]. Neste cenário, onde a educação se apoia na Computação Ubíqua [10], um novo modelo educacional se torna possível, a fim de explorar, por meio de dispositivos móveis, uso de recursos pedagógicos em contextos de ensino que podem ocorrer em qualquer hora e em qualquer lugar.

O professor, ao realizar o planejamento de suas aulas, pode conceber a execução de suas aulas em diferentes locais, como por exemplo, sala de aula, laboratório, biblioteca, entre outros. Neste sentido, pode-se considerar o *contexto* onde os alunos estão inseridos no seu processo de seleção de materiais. Aqui contexto é definido como qualquer informação que pode ser usada para caracterizar a situação de uma entidade (pessoas, lugares ou objetos) que são considerados relevantes para a interação entre o usuário e uma aplicação [11].

Um contexto de ensino só se torna real quando há um grupo de alunos presente no contexto. Tais grupos têm um caráter inerentemente dinâmico, no sentido que os alunos podem entrar ou sair do contexto em qualquer instante. Além disso, não apenas a localização, mas o horário também é determinante na definição de um contexto. Em relação ao grupo de alunos, pode-se considerar que mecanismos avançados de agregação podem combinar as informações individuais de cada aluno (perfil e preferências), sendo capazes de gerar um perfil representativo para um dado grupo, ou para toda a turma, e este pode ser utilizado para a busca e seleção dos OAs.

Ao se considerar o contexto dos alunos no refinamento da busca de OAs aliadas a possibilidade de se enviar materiais educacionais em qualquer lugar e em qualquer horário, pode-se dizer que este é um processo de recomendação ubíqua de conteúdo.

A partir da padronização dos OAs e da disponibilidade destes pelos repositórios é possível imaginar soluções para minimizar o tempo do professor na procura de materiais adequados. Uma ferramenta de apoio pedagógico capaz de localizar materiais adequados para as necessidades de grupos de aprendizes, baseado em informações de contexto previamente cadastradas pelo professor e que atendessem ao plano de desenvolvimento pedagógico, seria uma alternativa de solução tecnológica para este problema.

Aliando a padronização de materiais em repositórios e os recursos provenientes da educação ubíqua, pode-se modelar uma infraestrutura a fim de otimizar o processo

de localização de materiais. Neste cenário, este artigo propõe o UbiGroup, um modelo ubíquo para recomendação de conteúdo educacional focado em grupos de alunos formados dinamicamente. O modelo sugere conteúdo educacional para grupos de alunos considerando seus perfis e o contexto onde os aprendizes estão inseridos.

Existem trabalhos de recomendação de grupos, ou seja, modelos que sugerem aos aprendizes a formação dos grupos. No UbiGroup a formação dos grupos é automática, tendo como base os aprendizes inseridos nos contextos.

O texto está estruturado da seguinte forma. A seção 2 apresenta os conceitos básicos. A seção 3 aborda os trabalhos relacionados. A seção 4 apresenta o modelo desenvolvido. A seção 5 apresenta os aspectos de implementação do modelo. A seção 6 aborda o processo de avaliação. Por fim, a seção 7 relata as conclusões e considerações finais.

## 2. Conceitos Básicos

Nesta seção são apresentados os conceitos básicos envolvidos no desenvolvimento deste trabalho.

### 2.1. Sistemas de Recomendação

Os Sistemas de Recomendação (SR) têm por objetivo auxiliar as pessoas sugerindo algum tipo de informação por meio de ferramentas de apoio para determinadas tarefas. Visam fornecer itens de informação (por exemplo, páginas *web*, livro, filmes, músicas e documentos) que são de interesse para um determinado usuário [12].

Para sugerir itens para o usuário ou para um grupo de usuários, os sistemas usam diferentes fontes de dados, como preferências ou características dos usuários. No âmbito deste artigo as principais informações a serem utilizadas serão os perfis dos alunos e o contexto no qual estão inseridos. Os SRs são classificados em três categorias, sistemas de recomendação baseado no conteúdo, sistemas de recomendação baseado na filtragem colaborativa e sistemas híbridos [13].

Um dos fatores relevantes para a pesquisa realizada é a formação dos grupos de indivíduos (colega, alunos, professores, entre outros) em meio aos sistemas de recomendação. Nesta condição de grupo é necessário levar em conta as preferências e características dos alunos contidos no grupo. Dentre as características podem-se considerar as informações como um curso, uma turma, um grupo de estudos e também informações do contexto, como posição via GPS, dispositivos e horários.

A recomendação para um grupo de alunos é uma forma diferenciada de sugerir conteúdos aos aprendizes. Neste caso, a recomendação de um determinado item pode ser realizada levando em conta o grupo de pessoas. Os grupos em um sistema de recomendação podem ser

divididos em quatro categorias, grupo estabelecido, grupo ocasional, grupo randômico e grupo automaticamente identificado [12].

A informação de tipo do grupo é relevante para que se possa dar o direcionamento correto na estruturação da recomendação. Inicialmente é preciso saber o grupo de pessoas que se deseja atingir para após iniciar os passos para prover a recomendação.

### 2.2. Sensibilidade a contexto

A sensibilidade a contexto [9] é um tema de pesquisa relacionado à Computação Ubíqua [11] criada por Weiser em 1991. Em 2001, Satyanarayanan enquadrava a visão de Weiser dentro do estado tecnológico existente na época. Na sua publicação [14], ele especifica áreas de pesquisa da computação ubíqua, sendo uma destas, a sensibilidade a contexto.

Satyanarayanan entende que para um sistema seja minimamente intrusivo, é necessário que ele seja capaz de reconhecer a situação do usuário e seus arredores, e adaptar seu comportamento de acordo com esta informação. Esta informação é chamada de “contexto” e sistemas capazes deste tipo de adaptação são chamados “sensíveis a contexto”.

O contexto foi definido por vários autores, entretanto a definição que se tornou referência é a realizada por Dey em 2001 [9]. No UbiGroup, o contexto é a informação mais relevante para agrupar os aprendizes, pois ela identifica os alunos por meio de sua posição física e com isso os agrupa em contextos de interesse comum.

### 2.3. Sistemas multiagentes

Os sistemas multiagentes [15] são sistemas compostos de agentes coordenados através de suas relações. Sendo que o Agente é uma entidade autônoma que pode adaptar-se e interagir com o seu ambiente [16]. Nesta estrutura cada agente de software possui seu papel e responsabilidade, e por meio de seus objetivos, planos e relações, visam atender as necessidades do sistema.

A base tecnológica do UbiGroup é fundamentada na utilização de agentes. Com base nesta estruturação é possível delegar responsabilidades isoladas para cada agente permitindo que atuem de forma independente. A comunicação entre os agentes e as trocas de mensagens é padronizada por meio da utilização de ontologia, desta forma garantindo a interoperabilidade. Assim o sistema torna-se flexível e pode ser integrado a outros sistemas multiagentes. Esta flexibilidade é importante para o desenvolvimento de trabalhos futuros relacionados ao modelo UbiGroup.

## 3. Trabalhos Relacionados

A seleção dos trabalhos relacionados teve como objetivo buscar trabalhos que se assemelhassem ao UbiGroup.

Durante a pesquisa não foram encontrados trabalhos específicos de recomendação de OAs para Grupos, desta forma, buscou-se os trabalhos por meio de duas frentes de pesquisa, trabalhos que fossem sobre recomendação para grupo, mas que não recomendassem OAs, e, trabalhos que recomendassem OAs, mas não para grupos de aprendizes. Adotando esta abordagem, foi possível efetuar uma comparação dos trabalhos levando em conta os principais aspectos do UbiGroup que são Grupos, Contextos e OAs.

### 3.1. Recomendação para Grupos

Os estudos desenvolvidos em [17] consistem na definição de um modelo para identificação automática de grupos de usuários e um algoritmo para recomendação para grupos utilizando a técnica de filtragem colaborativa. Para identificação do grupo, o modelo utiliza as recomendações individuais, as combina com as preferências dos usuários e usa os dois como entrada para o algoritmo de agrupamento. Sobre recomendação para grupos, o artigo apresenta o algoritmo ImprovedGRA, este gera predições a partir da matriz de classificação utilizada na identificação dos grupos e adiciona as predições dos itens não classificados pelos usuários, sendo seu diferencial na pesquisa realizada. A avaliação ocorreu a partir de recomendações de filmes para grupos de usuários.

No trabalho [18] é apresentada a plataforma GroupRecoPF. Ela fornece aos desenvolvedores um suporte para a construção de sistemas de recomendação para grupos. A geração da recomendação ocorre a partir da união de três estratégias: predição mais alta dentre as médias do grupo, maior do grupo e a menor individual. O trabalho apresenta uma infraestrutura composta pela *GroupRecoServer* onde ocorre o processo de recomendação e comunicação, e, o *GroupRecoEditor* que suporta aplicações para *desktop* e para dispositivos móveis. Os artefatos recomendados durante o processo de avaliação do modelo foram filmes.

Em [19] é apresentado o *GRec\_OC* que consiste no desenvolvimento de um procedimento para recomendação para grupos em comunidades *on-line*. Sua técnica de recomendação é baseada na criação de um perfil para o grupo representando a agregação das preferências dos membros do grupo. Seu processo de recomendação consiste em duas fases, a primeira é a geração de um conjunto de recomendações para o grupo utilizando o método de filtragem colaborativa. A segunda é a remoção de itens irrelevantes para melhor satisfazer as preferências dos membros através de um método de filtragem baseado no perfil individual dos membros do grupo.

No sistema proposto em [20] a semelhança de usuários é identificada com base em informações implícitas sobre interações do usuário e sua navegação. Então regras de associação são usadas para melhorar o processo de filtragem colaborativa. Por fim, o processo de filtragem

com base em conteúdo utiliza técnicas semânticas para identificar mensagens relevantes.

### 3.2. Recomendação de OAs

No trabalho apresentado por [21] foi proposto um modelo de recomendação de OA, chamado e-LORS. Sua abordagem para recomendação de conteúdo é baseada no relacionamento entre perfis e objetos de aprendizagem. Seu processo de recomendação é baseado em filtragem de informações baseada em conteúdo. O processo de recomendação de objetos de aprendizagem é realizado por uma filtragem em três etapas. As etapas selecionam os OAs com base nas informações referente ao tema a ser apresentado aos alunos, no perfil de aprendizagem do aluno e nas restrições tecnológicas que caracterizam o ambiente em que o aluno interage.

No trabalho desenvolvido em [22], foi apresentado um modelo de sistema de recomendação de OAs com base no desenvolvimento de competências em EAD. O recomendador utiliza como base no seu processo a relação das competências que se deseja desenvolver nos alunos, o perfil do usuário e os objetos de aprendizagem disponíveis no repositório em questão. O processo de recomendação definido no modelo consiste em três etapas. A primeira etapa é a seleção dos OAs a serem utilizados com a turma visando o desenvolvimento de competências dos alunos. A segunda etapa consiste nos alunos responderem a um questionário referente ao seu nível de conhecimento em determinadas competências além de informações demográficas. Com base nestas informações o sistema pode recomendar os objetos de aprendizagem mais adequados. A última etapa é a responsável por efetuar a recomendação. Um mecanismo executa a filtragem de informação selecionando os OAs a serem recomendados aos alunos, a partir do seu perfil e das competências que o mesmo necessita desenvolver ao longo do curso.

No estudo proposto em [23] é apresentado o MobiLE, uma abordagem baseada em agentes para recomendação sensível ao contexto de OAs. A proposta para recomendação apresentada no trabalho é baseada em um fator de recomendação influenciado pela área de interesse, local e horário preferido do aluno. O modelo implementado é baseado em um sistema de recomendação que considera o conteúdo dos OAs em relação ao perfil do aprendiz e também a comparação de alunos com preferências semelhantes.

## 4. UbiGroup

O UbiGroup é um modelo de recomendação de conteúdo para grupos de aprendizes que possui como principal objetivo a recomendação de OAs para grupos de aprendizes, levando em conta os perfis dos aprendizes e o contexto onde estão inseridos. Sua arquitetura é composta por cinco agentes conforme pode ser visto na Figura 1. O Agente Apoio Pedagógico (AP) é o responsável por obter

as informações do aprendiz e de informá-lo sobre novas recomendações. O Agente Gestor de Perfis (GP) mantém atualizados os perfis dos aprendizes e gera a similaridade entre eles. O Agente Gestor de Contextos (GC) gerencia os contextos do sistema. O Agente Recomendador (RE) mantém as regras de recomendação. Por sua vez, o Agente Comunicador (CO) efetua a comunicação com os repositórios de OAs. As próximas subseções abordam os agentes do UbiGroup.

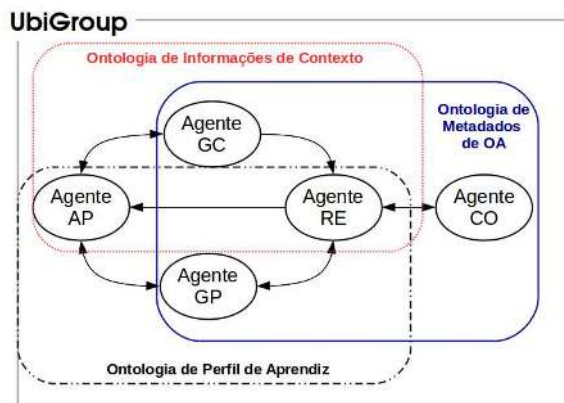


Figura 1: Arquitetura do UbiGroup

#### 4.1. Agente Apoio Pedagógico

O Agente Apoio Pedagógico (AP) realiza a interface com o aprendiz. Ele envia as informações atualizadas dos perfis dos aprendizes para o Agente Gestor de Perfis. Também envia a posição atualizada para o Agente Gestor de Contextos. O agente monitora as ações do aprendiz e disponibiliza as informações de perfil e contexto.

Este agente é executado no dispositivo móvel, como Smartphones e Tablets, dos aprendizes. A partir de sua interface gráfica, o aprendiz pode acessar o UbiGroup e ingressar em um contexto, desta forma, estando apto a receber recomendações de conteúdo.

O primeiro passo para o aprendiz receber uma recomendação é ingressar em um contexto, para que isso ocorra, é papel do agente AP monitorar as movimentações dele e enviar esta informação para o agente GC. Caso o agente GC identifique que o aprendiz está em um contexto, ele avisa o agente AP que, por sua vez, notifica o aluno.

Durante o processo de recomendação, a responsabilidade do agente AP é notificar o aprendiz de que existe uma recomendação para ele. O agente também é responsável por apresentar os OAs recomendados para o aprendiz. O aprendiz ao receber a recomendação poderá aceitá-la ou rejeitá-la. Caso ele aceite, os dados do OAs são apresentados para o aprendiz. Após a visualização do OA, o aprendiz é convidado a classificar se a recomendação foi satisfatória ou não.

#### 4.2. Agente Gestor de Perfis

O Agente Gestor de Perfis (GP) mantém os perfis dos aprendizes atualizados, além de ser responsável por calcular a similaridade entre os perfis dos aprendizes contidos no contexto. O agente GP também é responsável pelo controle de acesso dos aprendizes no sistema, verificando se as credenciais de acesso do aprendiz enviadas pelo agente AP estão liberadas ou não. Após receber estas informações o agente AP informa o aprendiz do retorno da operação.

O perfil de cada aprendiz contém suas informações pessoais, os relacionamentos com outros aprendizes, desempenhos, preferências e seu portfólio. O agente GP utiliza um modelo ontológico para representar estas informações. O objetivo da utilização das ontologias é auxiliar os agentes nas trocas de mensagens, fornecendo um padrão no tratamento das informações do sistema. Para este fim, foi criada uma ontologia OWL cujas classes e propriedades foram definidas de acordo com o padrão PAPI [24] de metadados de aprendizes (Figura 2).

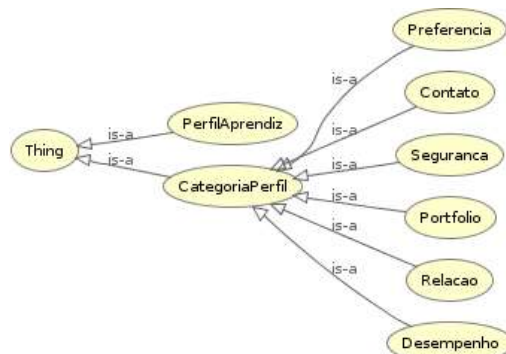


Figura 2: Classes da Ontologia de Perfil de Aprendiz

As informações contidas no perfil de aprendiz no UbiGroup estão divididas em seis categorias: Contato, Segurança, Relação, Portfólio, Preferência e Desempenho, cada uma representada por uma classe da ontologia.

As informações pessoais do aprendiz estão relacionadas na classe Contato. Através de atributos de dados (*data properties*) os objetos (indivíduos) desta classe possuem as seguintes informações: 1) *TemIdAprendiz*: identificador único do aprendiz no sistema; 2) *TemNome*: nome do aprendiz; 3) *TemTipo*: tipo do aprendiz no sistema, podendo ser do tipo Aluno ou Professor; 4) *TemEmail*: e-mail do aprendiz; 5) *TemTelefone*: telefone de contato; 6) *TemEndereco*: endereço completo.

Na classe Seguranca são informadas as credenciais de acesso do aprendiz. Ela possui as seguintes informações: 1) *TemIdAprendiz*: identificador único do aprendiz no sistema; 2) *TemLogin*: login de acesso do aprendiz; 3) *TemSenha*: senha de acesso; 4) *TemLogado*: indica se o aprendiz esta acessando o sistema; 5) *TemContextoAtual*:

indica de qual contexto o aprendiz faz parte.

Os relacionamentos do aprendiz com outros aprendizes são mantidos na classe *Relacao*, que possui as seguintes informações: 1) *TemIdAprendiz*: identificador único do aprendiz no sistema; 2) *TemIdAprendizRelacao*: identificador único do aprendiz com o qual houve uma relação de contato; 3) *TemDataHora*: data e hora que houve a relação com o aprendiz; 4) *TemContexto*: contexto onde ocorreu esta relação; 5) *TemGrauSimilaridade*: grau de similaridade entre os aprendizes.

Na classe *Portfolio* são mantidos as informações referentes aos trabalhos produzidos pelo aprendiz. As informações contidas nesta classe são as seguintes: 1) *TemIdAprendiz*: identificador único do aprendiz no sistema; 2) *TemIdPortfolio*: identificador único do material do aprendiz; 3) *TemTipoPortfolio*: tipo do material podendo ser um Artigo Completo, Artigo Resumido, Pôster, Vídeo, Áudio, Livro, Apresentação e Outros; 4) *TemTitulo*: título do material; 5) *TemDescricao*: breve descrição do material; 6) *TemEvento*: evento onde o material foi publicado; 7) *TemLocal*: local do evento; 8) *TemAno*: ano da publicação.

A classe *Desempenho* contém o histórico do aprendiz listando seu desempenho e avaliações. Esta classe possui as seguintes informações: 1) *TemIdAprendiz*: identificador único do aprendiz no sistema; 2) *TemIdDesempenho*: identificador único do histórico de desempenho; 3) *TemTipoDesempenho*: tipo de historio de desempenho podendo ser Prova, Disciplina, Certificação e Outros; 4) *TemNomeDesempenho*: nome referente ao desempenho; 5) *TemDescritivo*: breve descrição; 6) *TemInstituicao*: nome da instituição onde foi realizado; 7) *TemAvaliacao*: nota obtida, se houver; 8) *TemDataReferencia*: data em que foi realizado.

As informações relativas às preferências do aprendiz estão relacionadas na classe *Preferencia*. Esta classe possui as seguintes informações: 1) *TemIdAprendiz*: identificador único do aprendiz; 2) *TemFormatoOA*: formato de OA preferido pelo aprendiz (Áudio, Vídeo, Documento, Páginas Web, entre outros); 3) *TemTipoDispositivo*: tipo de dispositivo preferido pelo aprendiz (*smartphone*, *tablet* ou *notebook*); 4) *TemEstiloAprendizagem*: estilo de aprendizagem do aprendiz; 5) *TemContexto*: contexto preferido para receber recomendação.

Outra funcionalidade do agente GP é manter os perfis dos aprendizes atualizados. O agente recebe do AP a atualização do perfil do aprendiz toda vez que ele acessa a aplicação ou quando atualiza seu perfil manualmente. Assim que o aprendiz for identificado em um contexto, ele receberá os perfis atualizados de todos os membros do contexto para geração do grau de similaridade entre eles.

O grau de similaridade do aprendiz para com os demais aprendizes do contexto é gerado a partir do cálculo

de similaridade baseado nas informações das classes *Preferencia*, *Desempenho* e *Portfolio*. O cálculo consiste em gerar um valor de equivalência individual para cada item contido nestas classes e gerar uma média sobre todos os valores obtidos. A equação (1) apresenta o cálculo de similaridade.

$$S(PA_1, PA_2) = \frac{\sum_{i=1}^n ve(PA_1.item_i, PA_2.item_i)}{n} \quad (1)$$

A similaridade ( $S$ ) entre dois perfis de aprendizes  $PA_1$  e  $PA_2$  é igual ao somatório do valor de equivalência ( $ve$ ) de cada item do perfil dos aprendizes, dividido pelo número total de itens avaliados ( $n$ ).

A geração do valor de equivalência individual entre os itens dos perfis dos aprendizes é definida pelo algoritmo *editDistance* [25]. Este algoritmo compara dois *strings* e determina o número mínimo de operações necessárias para transformar uma *string* em outro. Com o resultado da função é possível calcular o valor de similaridade entre as informações a partir da divisão do número de operações necessárias, obtida pelo algoritmo e pelo número máximo de operações possíveis.

A informação de similaridade entre os perfis é utilizada pelo Agente Recomendador para definir qual o perfil mais representativo do grupo contido no contexto. Com base neste perfil é realizada a recomendação. Na seção 4.4 é detalhado este processo.

### 4.3. Agente Gestor de Contextos

O Agente Gestor de Contextos (GC) é responsável por manter os contextos utilizados pelo UbiGroup e por informar o Agente Recomendador sobre alterações ocorridas com relação à entrada ou saída de membros do contexto. O agente GC também utiliza um modelo ontológico para representar as informações de contexto. As classes e propriedades desta ontologia são baseadas em [9]. As informações que representam o contexto estão organizadas em quatro categorias Localização, Atividade, Temporal e Grupo (Figura 3).

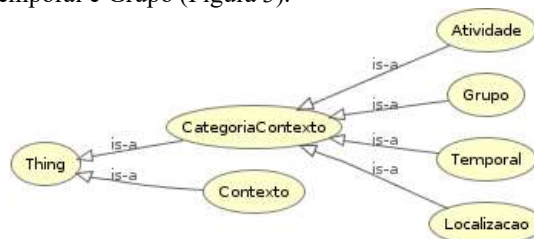


Figura 3: Classes da Ontologia de Informações de Contexto

A classe *Localizacao* contém as informações de posicionamento geográfico do contexto. A classe possui as seguintes informações: 1) *TemLocal*: nome representativo para a localização; 2) *TemPosicaoGPSInicial*: posição GPS do quadrante superior esquerdo do contexto; 3) *TemPosicaoGPSFinal*: posição GPS do quadrante inferi-

or direito do contexto;

A classe *Atividade* contém as informações referentes às atividades ou situações que representam um contexto. A classe possui as seguintes informações: 1) *TemTipo*: tipo de atividade a ser realizada, podendo ser definido como Aula, Prova, Laboratório, Extraclasse, Biblioteca, Bar, Auditório e Outros; 2) *TemNome*: nome que representa a atividade em questão; 3) *TemDescricao*: breve descrição sobre a atividade; 4) *TemRegraRecomendacao*: Regra de recomendação associada à atividade; 5) *TemPalavrasChave*: palavras-chave relativas à atividade, esta informação servirá de base para a busca de OAs; 6) *TemGrauEnsino*: define o grau de ensino da atividade no contexto, podendo ser Ed. Infantil, Ensino Fundamental, Ensino Médio, Ensino Superior ou Pós-graduação.

Na classe *Temporal* são relacionadas às informações de contexto referente à limitação de tempo para o contexto existir. A classe possui as seguintes informações: 1) *TemDiasdaSemana*: dias da semana no qual o contexto existe; 2) *TemData*: data específica para o contexto existir; 3) *TemHoraInicio*: horário inicial de acompanhamento do contexto; 4) *TemHoraFim*: horário final de acompanhamento da formação do contexto. Para os itens 1 e 2, pelo menos um deles deve estar preenchido.

A classe *Grupo* contém informações referentes à formação de grupos no contexto. Esta classe possui as seguintes informações: 1) *TemTipoGrupo*: tipo dos membros que podem fazer parte do grupo no contexto, podendo ser dos tipos Alunos, Professores ou Misto; 2) *TemNumeroMinimoAprendizes*: número mínimo de aprendizes dentro do contexto para a formação de um grupo.

A classe *Localizacao* serve de base para saber se o aprendiz está na localização de um contexto ou não. Já a classe *Atividade* serve para informar o aluno da atividade a ser realizada no contexto, mas principalmente por manter as palavras-chave do tema que se deseja recomendar e a regra de recomendação. As classes *Temporal* e *Grupo* são a base para geração do contexto, pois determinam o dia e horário da existência do contexto e a formação do grupo, respectivamente.

O agente GC além de possuir todos os dados dos contextos do sistema, também mantém todos os contextos que estão ativos. Entende-se por contexto ativo aquele contexto que possui pelo menos um aprendiz contido nele. A condição para o contexto estar ativo não significa que ele está apto a receber uma recomendação, pois, para isso, deve satisfazer a todas as condições das classes *Temporal* e *Grupo*.

Um grupo é formado dinamicamente quando um conjunto mínimo de estudantes (que é um requisito definido na classe *Grupo*) entra em um dado contexto no horário certo (isto é, em um horário que satisfaz os requisitos

definidos na classe *Temporal*). O grupo se desfaz, quando não há um número suficiente de estudantes no contexto ou quando seu tempo de vida se encerrar.

O agente GC é o responsável por iniciar o processo de recomendação. Ele recebe do agente AP a informação da entrada e saída de membros no contexto. Com esta informação o agente GC verifica se o contexto está apto a receber uma recomendação, ou seja, verifica se todas as condições para existência do contexto estão atendidas (data, hora, posição geográfica e formação em grupo). Caso esteja apto, ele informa o agente RE e se inicia o processo de recomendação.

#### 4.4. Agente Recomendador

O Agente Recomendador (RE) é o responsável por gerenciar o processo de recomendação. Ele gerencia as regras de recomendação que controlam este processo, executando as etapas necessárias para preparar, executar e encaminhar a recomendação.

Uma regra de recomendação é um conjunto de mapeamentos entre o perfil do aprendiz, o contexto e os metadados dos OAs. Estes mapeamentos fornecem as informações necessárias para construir filtros de busca capazes de selecionar OAs apropriados ao contexto, pelas informações contidas nos metadados do OA. Os mapeamentos de uma regra de recomendação correlacionam informações particulares contidas em um perfil de aprendiz ou no contexto de ensino, com informações que podem aparecer nos metadados de OA. Todas as informações utilizadas nestas regras são representadas de acordo com a correspondente ontologia OWL. Portanto, as informações de perfil de aprendiz e de contexto são representadas pelas classes e propriedade das ontologias apresentadas, respectivamente, nas Figuras 2 e 3. Para os metadados de OA foi empregada a ontologia de metadados OBAA definida em [26] que cobre todos os metadados IEEE-LOM.

Cada mapeamento é formado pela informação de contexto ou do perfil de aprendiz, seguida da definição de qual será o metadado de OA correspondente a esta informação de perfil ou contexto. Na Tabela 1 são apresentados alguns exemplos de mapeamentos criados entre o perfil do aprendiz e os metadados dos OAs. Na Tabela 2 são apresentados os mapeamentos entre o contexto e os metadados de OAs.

A partir dos mapeamentos contidos nas regras de recomendação, é possível gerar uma consulta aos OAs utilizando as informações dos seus metadados para filtros de pesquisa. A consulta gerada é enviada para o agente CO, que encaminha ao repositório para que sejam pesquisados os OAs que atendam aos filtros informados.

| Perfil do Aprendiz |      | Metadados OA |                      |
|--------------------|------|--------------|----------------------|
| Contato            | Tipo | Educational  | IntendedEndUserRoles |



|             |                    |              |                        |
|-------------|--------------------|--------------|------------------------|
| Preferencia | FormatoOA          | Segmentation | SegmentMediaTypels     |
| Preferencia | TipoDispositivo    | Technical    | SupportedPlatformsIs   |
| Preferencia | EstiloAprendizagem | Educational  | LearningResourceTypels |
| Preferencia | EstiloAprendizagem | Educational  | InteractivityLevels    |
| Preferencia | EstiloAprendizagem | Educational  | InteractivityTypels    |

**Tabela 1:** Mapeamentos entre Perfil do Aprendiz e metadados de OA.

| Contexto  |               | Metadados OA |                        |
|-----------|---------------|--------------|------------------------|
| Atividade | PalavrasChave | General      | Title                  |
| Atividade | PalavrasChave | General      | Description            |
| Atividade | PalavrasChave | General      | Keyword                |
| Atividade | Tipo          | Educational  | LearningResourceTypels |

**Tabela 2:** Mapeamentos entre Contexto e metadados de OA.

O professor define quais mapeamentos ele irá utilizar para compor sua regra de recomendação. A definição desta regra propicia ao professor buscar materiais atrelados ao plano de ensino desejado.

O processo de recomendação é iniciado quando existe um contexto pronto para receber uma recomendação. Para isso, é necessário que as condições deste contexto estejam plenamente satisfeitas. As condições são a localização dos aprendizes, a formação em grupo e o dia e a hora atual. Estando o contexto apto a receber a recomendação, inicia-se a preparação dos dados.

O processo de seleção dos OAs recomendados é realizado pelo agente RE e consiste em três etapas: obtenção do perfil mais representativo para o grupo de aprendizes, a busca dos OAs no repositório cadastrado e a classificação dos itens selecionados com base nas avaliações dos membros do grupo.

A primeira etapa é a análise dos perfis contidos no contexto. Esta análise consiste em identificar qual o perfil mais representativo para o grupo e com isso tomar este como base na consulta aos OAs. Para identificar este perfil é utilizado o grau de similaridade entre os aprendizes. O perfil escolhido é aquele que tiver a maior média do grau de similaridade entre os membros do grupo.

Após isso, inicia-se a segunda etapa que é a busca dos OAs no repositório. Com base no perfil, contexto e a regra de recomendação (informada no contexto), o agente RE define os filtros de pesquisa a serem utilizados para buscar os OAs e envia para o agente CO. Este agente executa a consulta no repositório e retorna para o agente RE. O resultado obtido com esta busca é a listagem prévia de OAs a serem recomendados.

Para finalizar o processo de recomendação, o agente RE classifica os OAs, previamente selecionados, com base nas avaliações realizadas pelos alunos. Sendo assim, os itens melhor avaliados serão os primeiros da listagem dos itens a serem recomendados para o grupo. O agente RE seleciona os primeiros OAs da listagem e envia para o agente AP.

## 4.5. Agente Comunicador

O Agente Comunicador (CO) é o responsável por efetuar a comunicação com os repositórios de OAs. A partir de uma solicitação do agente RE, ele efetua o acesso ao repositório cadastrado e retornar as informações solicitadas. Este acesso ocorre de forma automática para o aprendiz, sendo necessário apenas o repositório ser previamente configurado. A comunicação com os repositórios irá ocorrer por meio de comunicação *Webservice* onde o agente CO irá acessar os serviços disponíveis no repositório. O principal processo a ser executado nos repositórios é a consulta aos OAs.

O agente CO recebe do agente RE os filtros de pesquisa que deseja executar sobre os metadados dos OAs. O agente transforma estes dados em uma consulta SPARQL [27] e esta é enviada ao repositório. O repositório, por sua vez, retorna uma listagem dos OAs em XML. Esta listagem é encaminhada para o agente RE.

## 5. Aspectos de Implementação

Para apoiar no processo de avaliação do UbiGroup foi implementado um protótipo. Primeiramente foi realizado o projeto do sistema, onde foi desenvolvida a documentação técnica, e logo após a etapa de desenvolvimento.

Na etapa de projeto foi utilizada a linguagem UML 2.1 para modelar os agentes e suas funcionalidades. Inicialmente, criaram-se os diagramas de relação de dependência e logo após os diagramas de UML. Na etapa de desenvolvimento foram implementadas as funcionalidades previstas em cada agente e a comunicação entre eles.

O modelo de arquitetura adotada no projeto do sistema foi o MVC (*Model View Controller*). Esta arquitetura consiste na divisão do sistema em camadas, facilitando o entendimento e eventuais manutenções no código-fonte. Esta arquitetura está dividida em três camadas, *Model*, *View* e *Controller*. Na camada *Model*, são definidas as informações trocadas pelo sistema, ou seja, a definição das estruturas de dados. A camada *Controller* é a responsável por armazenar as regras de negócio da aplicação e gerenciar a comunicação entre as camadas. A camada *View* (visão) é a responsável por apresentar os dados ao usuário final e por receber as interações do usuário junto ao sistema. Na Figura 4 pode ser observada a arquitetura utilizada na implementação do UbiGroup, a divisão entre as três camadas e a subdivisão destas em pacotes.


**Figura 4:** Arquitetura de Implementação do UbiGroup

No desenvolvimento foi utilizada a linguagem de programação Java em conjunto de tecnologias, como Tomcat, Android SDK e REST/Jersey, para suportar a estrutura desenvolvida no UbiGroup.

## 6. Metodologia de Avaliação

Nesta seção é apresentada a metodologia de avaliação aplicada no UbiGroup. Para isso, foram definidos cenários simulando a utilização da aplicação como uma ferramenta de apoio pedagógico. Para auxiliar neste processo foi necessária a implementação de um protótipo. A estratégia de validação por cenários é uma abordagem que vem sendo utilizada pela comunidade científica para avaliação de ambientes sensíveis ao contexto [28] e ambientes ubíquos [29].

O objetivo desta avaliação é realizar simulações da utilização do UbiGroup em atividades propostas pelo professor, considerando diferentes grupos de aprendizes e diferentes contextos a fim de analisar o processo de recomendação de materiais. Para realização desta avaliação foram definidos dois cenários.

Antes de iniciar o processo de utilização da aplicação, se fez necessário realizar a preparação da aplicação por meio do cadastramento de algumas informações. Esta etapa é discutida na subseção 6.1. Nas subseções 6.2 e 6.3 são descritos os cenários de avaliação do UbiGroup.

### 6.1. Preparação da Aplicação

Antes da realização dos cenários foi necessário cadastrar algumas informações na aplicação, como os locais para criação dos contextos, as configurações para acesso aos repositórios de OAs e os perfis dos alunos.

Os locais cadastrados na aplicação serão utilizados posteriormente na criação dos contextos. A tela disponibilizada na aplicação para realização deste cadastro é apresentada na Figura 5. Foram definidos quatro locais reais existentes na Universidade do Vale do Rio dos Sinos (Unisinos): Sala 6B624, Laboratório de Informática, Biblioteca e o Bar 1.



Figura 5: Tela de cadastramento de locais para os contextos.

Além do cadastramento dos locais disponíveis na aplicação é necessário configurar o acesso aos repositórios de OAs. Estes repositórios são responsáveis por disponibilizar os materiais para o processo de recomendação. Nos cenários foi utilizado o repositório de OAs do projeto OBAA MILOS [30].

As informações dos perfis dos alunos podem ser obtidas pela instituição no momento em que eles efetuam sua matrícula, por meio do sistema administrativo. Algumas informações específicas do UbiGroup, necessárias na geração da similaridade, como por exemplo, as preferências dos alunos, podem ser preenchidas a partir da tela de manutenção de perfis (Figura 6).



Figura 6: Tela de manutenção de perfis de alunos.

Após o cadastro destas informações na aplicação é possível disponibilizá-la para o professor efetuar o cadastramento das suas regras de recomendação. Para o professor realizar esta atividade ele precisa conduzir uma análise junto ao seu plano de ensino para identificar os momentos em que deseja utilizar a ferramenta como apoio às suas aulas.

Para realização dos cenários foi utilizado um exemplo de uma parte de um plano de ensino da disciplina de Modelagem e Simulação de um curso de graduação de Ciências da Computação. O professor para abordar o conteúdo durante o semestre definiu um roteiro para a execução das suas aulas. Ele distribui o conteúdo programático diversificando entre as atividades a serem realizadas durante a aula, locais onde a aula ocorrerá e quais serão os tópicos de ensino programados. Na Tabela 3 é apresentada a programação de execução das aulas.

|   |                                                                                                                                                                        |                      |
|---|------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----------------------|
| 1 | Atividades: Aula expositiva.<br>Tópicos: - Apresentação do programa de ensino da disciplina;<br>- Aspectos gerais sobre a simulação e seu uso.                         | Local: Sala de aula. |
| 2 | Atividades: Aula expositiva.<br>Tópicos: - Noções gerais sobre a simulação;<br>- Evolução do conceito de simulação;<br>- Vantagens e Desvantagens do uso da Simulação. | Local: Sala de aula. |
| 3 | Atividades: Aula expositiva e elaboração de uma resenha.<br>Local: Sala de aula.<br>Tópico: Modelos e sistemas de simulação.                                           |                      |
| 4 | Atividades: Estudo dirigido e discussão sobre o conteúdo.<br>Local: Sala de aula.<br>Tópico: Conceitos Gerais sobre Números Aleatórios.                                |                      |

|   |                                                                                                                                              |
|---|----------------------------------------------------------------------------------------------------------------------------------------------|
| 5 | Atividades: Estudo dirigido em grupo e entrega de trabalho.<br>Local: Laboratório de informática.<br>Tópico: Geradores de Números Aleatórios |
| 6 | Atividades: Prova individual. Local: Sala de aula.<br>Tópico: Prova 1.                                                                       |

**Tabela 3:** Roteiro das aulas.

A partir deste roteiro, o professor é capaz de identificar em quais aulas ele poderá utilizar o UbiGroup como ferramenta de apoio pedagógico. Nos cenários apresentados nesse artigo, o professor identificou duas aulas (aulas 3 e 4) onde a disponibilização de materiais poderá auxiliar os alunos na realização das atividades propostas. Para utilização do UbiGroup nestas aulas é necessário que o professor identifique o contexto e a regra de recomendação adequada para atender a sua aula.

Para a aula 3, o planejamento do professor prevê uma aula sobre o conteúdo de modelos e sistemas de simulação. Na primeira parte da aula o professor irá realizar a apresentação deste tópico de forma expositiva. No segundo momento, os alunos deverão escrever uma resenha sobre este tópico. Para apoiar os alunos nesta atividade, eles receberão materiais educacionais recomendados a partir do UbiGroup. Na Figura 7 é apresentada a tela da aplicação onde as informações de contexto foram cadastradas para atender a esta aula.

Na definição do contexto o professor indica qual regra de recomendação deseja utilizar. A regra de recomendação fornecesse a associação entre o perfil do grupo de alunos e o contexto. Com base nisso, o sistema localiza os OAs que melhor se adequam as estas informações. Na Figura 8 é a apresentada a tela de cadastramento da regra de recomendação para atender a proposta da aula 3.

**Figura 7:** Tela de cadastro de contexto.

Detalhando a estrutura desta regra de recomendação, pode ser observado que o professor deseja localizar OAs relacionados ao tema “Modelos de Simulação”, para isso atrela a informação de palavra-chave contida no contexto ao título, descrição e as palavras-chave contidas nos metadados dos OAs. A regra de recomendação também leva em conta o tipo de aprendizes no grupo. Através da informação de tipo de contato (classe Contato e campo Tipo) o professor esperar buscar materiais adequados ao tipo “Alunos”. Referente às preferências do grupo, a

regra leva em conta o formato dos OAs, tipo de dispositivo e o estilo de aprendizagem. A partir destas informações a regra irá buscar os OAs que mais se adequam a estas preferências do grupo.

O cadastro de uma regra de recomendação na aplicação começa com a definição de um identificador e uma descrição para a regra. Após o professor precisa cadastrar os mapeamentos entre o Perfil dos Alunos, Contexto e Metadados dos OAs (Figura 8).

**Figura 8:** Cadastro de mapeamentos de regra de recomendação.

Esta etapa é estratégica, pois é a partir destas informações que os OAs serão localizados. Para definir estes mapeamentos, o professor precisa indicar a origem da informação (contexto ou perfil), a classe e também o campo que deseja utilizar. Depois disso, ele precisa indicar sobre qual classe e qual campo dos Metadados dos OAs eles serão filtrados.

Para cada mapeamento é necessário definir o tipo de associação entre os campos (Figura 9). Caso este tipo seja “Direta”, a própria informação contida nas classes do tipo Perfil ou Contexto serão utilizadas como filtro nos metadados dos OAs. Caso o tipo de associação seja “de-para”, para cada tipo de informação cadastrada nos campos das classes Perfil ou Contexto, existirá uma informação correspondente nos metadados dos OAs. A Figura 9 mostra a tela para cadastro de associações do tipo “de-para”.

**Figura 9:** Cadastro de associações de mapeamento “de-para”.

No exemplo apresentado na Figura 9, pode ser visto que se for selecionado o tipo “Aluno” através do campo “Contato.Tipo”, será efetuado a busca por “Learner”. Ou seja, durante o processo de busca, por meio dos metadados dos OAs, será procurado por “Learner”, utilizando para isso as informações contidas no campo “Educational.IntendedEndUserRoleIs”. Quando for selecionado o tipo “Professor” será procurado por “Teacher” nos metadados dos OAs.

Após a realização destes cadastros, a preparação do sistema para a recomendação está completa. O professor pode lecionar sua aula e aguardar que o processo de recomendação seja realizado automaticamente.

## 6.2. Cenário da Aula 3

No cenário aula 3, o professor preparou o sistema para recomendar materiais para os alunos após uma exposição de conteúdo. O objetivo é fornecer materiais para que os alunos possam elaborar uma resenha sobre o tema estudado. No início da aula o professor orienta os alunos a acessarem a aplicação UbiGroup e se autenticarem mediante usuário e senha previamente cadastrados (Figura 10a). Isso é necessário só no primeiro acesso, pois após esta autenticação, a aplicação se mantém ativa no dispositivo móvel atualizando suas informações de localização a fim de identificar contextos educacionais disponíveis.

O professor realiza sua aula prevista, onde apresenta de forma expositiva o conteúdo e os alunos vão para o intervalo. Ao retornarem do intervalo os alunos se locomovem em direção a sala de aula. Ao entrarem nas limitações da posição física do contexto, os alunos são notificados, através da aplicação móvel, que eles estão contidos no contexto definido pelo professor (Figura 10b). Além das informações de localização, para o contexto estar disponível, também são verificadas as informações de data e hora. Os alunos vão se locomovendo em direção a aula e à medida que vão entrando no contexto, eles além de serem automaticamente identificados, são vinculados ao contexto. Neste momento também é gerada a similaridade entre eles com o objetivo de identificar o perfil mais representativo para o grupo.

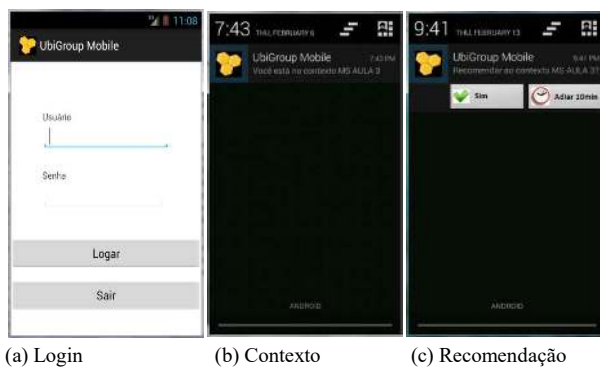


Figura 10: UbiGroup Mobile.

Através da tela de contexto pode ser observado o grupo de alunos contido no contexto, bem como a similaridade entre eles, conforme Figura 11.

| Nome Aluno | Similaridade |
|------------|--------------|
| Aluno A    | 49%          |
| Aluno B    | 43%          |
| Aluno C    | 32%          |
| Aluno D    | 44%          |
| Aluno E    | 42%          |

Figura 11: Alunos contidos no contexto.

Com a entrada dos alunos, ao ser atingido o número mínimo de alunos para o contexto, a aplicação executa o processo de recomendação. O primeiro passo é perguntar ao professor se a recomendação pode ser efetuada ou se ele deseja aguardar mais algum tempo (Figura 10c). Esta situação ocorre caso o professor identifique que estão faltando muitos alunos. No cenário, o professor postergou o início da recomendação em mais 10 minutos. Quando o horário postergado é atingido, a aplicação solicita novamente a confirmação do professor para executar a recomendação. Desta vez o professor confirma e o processo de recomendação é iniciado.

Com base nos perfis contidos no contexto, é possível identificar o perfil mais representativo para o grupo a partir da geração da similaridade entre estes perfis. O perfil que obtiver a maior média de similaridade será o perfil utilizado como referência no processo de recomendação. Conforme apresentado na Figura 11, o perfil mais representativo para o grupo de alunos é o “Aluno A”. Depois de identificado o perfil mais representativo do grupo, é localizada a regra de recomendação associada ao contexto. No caso, a regra de recomendação cadastrada é a “Regra de Recomendação 01”. A partir da regra de recomendação é possível definir os filtros de pesquisa que serão utilizados sobre os metadados dos OAs

Definidos os filtros de pesquisa, é necessário identificar a configuração de acesso ao repositório de OAs. O tipo de acesso ao repositório do OBAA MILOS é do tipo *Webservice* e o tipo de consulta disponível é SPARQL. Para que seja efetuada a consulta no repositório é necessário primeiramente montá-la neste padrão. A Figura 12 mostra parte da consulta gerada.

```
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX owl: <http://www.w3.org/2002/07/owl#>
PREFIX obaa: <http://obaa.unisinos.br/obaa22.owl#>
PREFIX myfn: <java:br.org.jena.>

SELECT distinct ?lobj ?key ?title ?desc ?loc WHERE {
?lobj a obaa:LearningObject .
?lobj obaa:hasMetadata ?mdata.
?mdata obaa:itsKeywordIs ?key.
?lobj obaa:hasMetadata ?mdtit.
?mdtit obaa:itsTitleIs ?title .
?lobj obaa:hasMetadata ?mddesc.
```

```

?mddesc obaa:itsDescriptionIs ?desc .
?lobj obaa:hasMetadata ?mdloc.
?mdloc obaa:itsLocationIs ?loc .

FILTER (regex(?key, '<PalavraChave>', 'i')).
FILTER (regex(?title, '<PalavraChave>', 'i'))
FILTER (regex(?desc, '<PalavraChave>', 'i'))
...
    
```

**Figura 12:** Parte da consulta SPARQL gerada.

Esta consulta é enviada para o repositório para que ocorra a pesquisa dos OAs disponíveis na base. Como resultado da execução da consulta no repositório é recebido uma listagem de OAs resultantes desta pesquisa. Para definir os OAs a serem recomendados é considerada a média das avaliações realizadas por outros alunos que já visualizaram tal material. Primeiramente é atribuída aos OAs a média de suas avaliações e após são ordenados por esta média de forma decrescente. Desta forma, os OAs melhores avaliados são recomendados para os alunos. Na Tabela 4 é apresentada a listagem dos OAs gerada no cenário, após a ordenação realizada com base na média das avaliações.

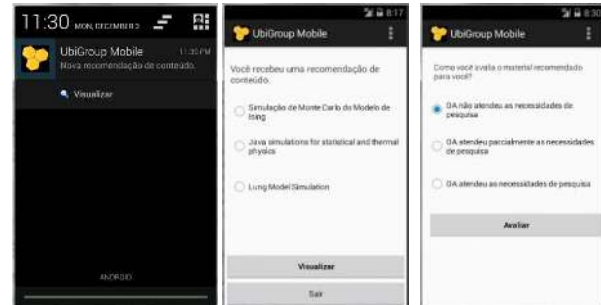
Para compor a lista dos OAs a serem recomendados aos alunos, são selecionados os dois primeiros OAs desta listagem ordenada e um OA que ainda não possui nenhuma avaliação. Desta forma, a aplicação favorece a renovação de materiais recomendados, onde OAs recentemente adicionados ao repositório tenham a possibilidade de serem avaliados pelos alunos.

| # | Nome                                                 | Referência                                                                                                                                                          | Avaliação |
|---|------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------|
| 1 | Simulação de Monte Carlo do Modelo de Ising          | <a href="http://objetoseducacionais2.mec.gov.br/handle/mec/16176">http://objetoseducacionais2.mec.gov.br/handle/mec/16176</a>                                       | 6,11      |
| 2 | Java simulations for statistical and thermal physics | <a href="http://www.opensourcephysics.org/binary/osp_stp.jar">http://www.opensourcephysics.org/binary/osp_stp.jar</a>                                               | 4,44      |
| 3 | Modelo exponencial: Crescimento populacional         | <a href="http://objetoseducacionais2.mec.gov.br/handle/mec/19738">http://objetoseducacionais2.mec.gov.br/handle/mec/19738</a>                                       | 2,27      |
| 4 | Lung Model Simulation                                | <a href="http://demonstrations.wolfram.com/LungModelSimulation/">http://demonstrations.wolfram.com/LungModelSimulation/</a>                                         | -         |
| 5 | The Bohr model                                       | <a href="http://www.upscale.utoronto.ca/PVB/Harrison/BohrModel/Flash/BohrModel.html">http://www.upscale.utoronto.ca/PVB/Harrison/BohrModel/Flash/BohrModel.html</a> | -         |

**Tabela 4:** OAs ordenados conforme avaliação.

Após a definição dos OAs, eles são enviados para os alunos. Por meio dos seus dispositivos móveis, os alunos recebem uma notificação informando que possuem uma nova recomendação (Figura 13a). Quando os alunos escolhem visualizar a notificação, são direcionados para a aplicação onde poderão acessar os materiais recomendados. Na Figura 13b é apresentada a tela do aplicativo contendo os materiais disponibilizados para os alunos. No cenário, o processo de recomendação apresentou para os alunos três materiais para sua visualização. Os alunos

podem selecionar o material desejado e escolherem a “visualizar”. Conforme o tipo de material, o UbiGroup executa a aplicação padrão do sistema operacional do dispositivo para o tipo de arquivo.


**Figura 13:** Recomendação de conteúdos no UbiGroup Mobile.

Após a visualização os alunos são convidados a realizar uma avaliação do material, conforme apresentado na Figura 13c. A avaliação consiste em indicar se o material atendeu plenamente, parcialmente ou não atendeu as necessidades de pesquisa do aluno. A avaliação realizada pelos alunos é mantida na aplicação, pois servirá de base para a classificação do OA em futuras recomendações.

Após a avaliação do material o processo de recomendação é finalizado. Com base nos materiais disponibilizados os alunos então passam ao desenvolvimento da atividade proposta pelo professor. Eles elaboram a resenha e entregam para o professor. Os alunos permanecem autenticados na aplicação tendo sua posição monitorada para que, ao se locomoverem pela instituição, possam fazer parte de novos contextos.

Este cenário descreveu o processo de recomendação de materiais para um grupo de alunos. Este cenário permite verificar como o modelo realiza recomendações de conteúdo considerando o perfil do grupo de alunos, por meio da identificação do perfil mais representativo para o grupo. Também é possível observar se o material recomendado atenderia as necessidades do professor.

### 6.3. Cenário da Aula 4

No cenário referente à aula 4 o professor tem como objetivo realizar uma aula expositiva e após o retorno do intervalo realizar um debate sobre o tema da aula. A recomendação neste cenário se torna um apoio para os alunos consolidarem o conhecimento adquirido. Primeiramente, o professor realiza a apresentação do conteúdo de forma expositiva conforme seu planejamento, e logo após, os alunos saem para o intervalo. Os alunos ao chegarem ao Bar são inseridos no contexto previamente cadastrado pelo professor. Alguns alunos retornam para a sala de aula e também são inseridos no outro contexto cadastrado pelo professor.

Desta forma, alguns alunos ficam fisicamente no Bar enquanto outros estão na sala de aula. O cadastro do contexto realizado no sistema é apresentado na Figura 14.



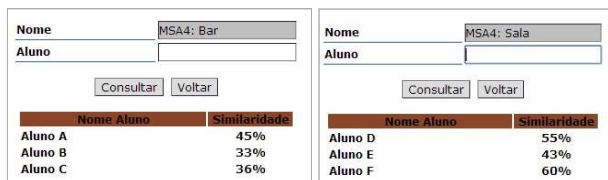
Figura 14: Cadastro de Contextos.

Ambos contextos usam a mesma regra de recomendação. Porém o professor deseja realizar recomendações diferenciadas para os alunos considerando seu contexto. O professor espera que os alunos na sala de aula recebam materiais mais apropriados para leitura, como por exemplo, artigos ou textos. Já para os alunos no Bar, o professor espera que estes recebam materiais mais facilmente analisados considerando o local onde eles estão, como por exemplo, vídeos ou apresentações. Para tanto o professor cria diferentes mapeamentos de contexto da regra de recomendação (Figura 15). Através do campo “Atividade.Tipo” define-se que para o tipo “Sala de Aula” devem ser localizados materiais do tipo texto, caso seja “Bar”, os materiais devem ser vídeos ou apresentações.



Figura 15: Informações dos contextos “Sala de aula” e “Bar”.

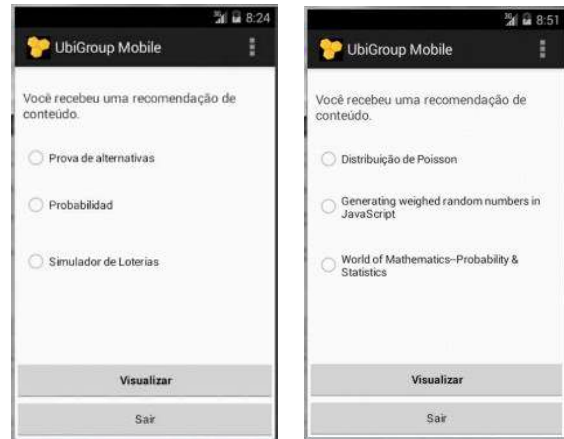
No momento em que é gerada a recomendação os alunos estão distribuídos da seguinte forma, os alunos “Aluno A”, “Aluno B” e “Aluno C” estão no Bar enquanto os alunos “Aluno D”, “Aluno E” e “Aluno F” já estão em sala de aula. A similaridade entre os perfis, em cada contexto, é apresentada na Figura 16.



(a) Similaridade no contexto Bar (b) Similaridade contexto Aula

Figura 16: Similaridade nos contextos.

Por ordem aleatória, primeiramente a aplicação realiza a recomendação para o grupo que está no Bar. É solicitada a confirmação do professor para realização da recomendação, ele confirma e o material é disponibilizado (Figura 17a). Logo em seguida é solicitada ao professor a confirmação da recomendação para o contexto referente à Sala de Aula. O professor confirma novamente, inicia o processo de recomendação e os materiais são disponibilizados para os alunos conforme Figura 17b.



(a) Recomendação contexto Bar (b) Recomendação contexto Aula

Figura 17: Recomendações para contextos diferentes.

Desta forma, a recomendação atendeu aos objetivos do professor, pois cada grupo recebeu o material mais adequado para seu contexto. Os alunos visualizam o material e realizam a avaliação do mesmo.

Os alunos do Bar retornam para sala de aula e o professor inicia a atividade. Os materiais recomendados para os alunos atenderam as necessidades do professor, pois foram do tema proposto em aula. O material serviu para complementar o conhecimento dos alunos, auxiliando na discussão em sala de aula.

Este cenário descreveu o processo de recomendação de materiais para um grupo de alunos. Este cenário permite verificar como o modelo realiza recomendação de conteúdos levando em conta o contexto e as preferências do grupo de alunos.

## 7. Considerações Finais

O modelo UbiGroup permite que sejam recomendados materiais educacionais para grupos dinâmicos de aprendizes que compartilhem o mesmo contexto. Na geração das recomendações são utilizadas informações sobre o contexto onde os alunos estão inseridos e o agrupamento de seus perfis. Regras de recomendação podem ser criadas pelo professor para guiar este processo e alinhar a busca de materiais ao plano de ensino.

A partir dos cenários simulados foi possível observar que o UbiGroup pode atender as necessidades do profes-

sor como uma ferramenta de apoio pedagógico. Neste sentido, ele pode ajudar a diminuir o tempo na busca e seleção de materiais educacionais adequados ao grupo de alunos e ao contexto onde eles estão inseridos. Por meio da definição de contextos e das regras de recomendação o professor pode direcionar a pesquisa de materiais a ser realizada para atender ao plano de ensino. No cenário da aula 3, foi possível observar, com base nos materiais recebidos, que o processo de recomendação para esta aula levou em conta as regras de localização, data/hora, definição da atividade e formação em grupo, contidas no contexto. Além disso, também se pode verificar que os materiais recomendados atenderiam as necessidades dos alunos, pois se referiam ao tema estudado. O cenário da aula 4 previa a recomendação de materiais para que os alunos pudessem realizar um debate sobre o tema estudado. Para personalizar a recomendação, foram criadas regras que levassem em conta o contexto onde os alunos estivessem inseridos. Isso de fato ocorreu, pois o tipo de material recomendado para os alunos que estavam no Bar foi diferente do tipo de material recomendado para o grupo que estava em sala de aula. Também foi considerado o perfil mais representativo para o grupo no tipo de material recomendado.

Estes cenários mostram que o UbiGroup é capaz de apoiar o professor nas atividades de busca e seleção de materiais, podendo diminuir o tempo destas atividades com o uso repetido do sistema e o reaproveitamento das regras de recomendação. Os cenários também demonstram que a pesquisa de OAs no UbiGroup considera as preferências o grupo de alunos, o contexto onde estão inseridos e o caráter dinâmico da formação do grupo. A Tabela 5 apresenta uma comparação do UbiGroup com os trabalhos relacionados.

| Trabalho                 | Tipo       | Item             | Contexto | Disp. Movel |
|--------------------------|------------|------------------|----------|-------------|
| ImprovedGRA [17]         | Grupo      | Filmes           | Não      | Não         |
| GroupRecommendation [18] | Grupo      | Músicas e Filmes | Não      | Não         |
| GRec_OC [19]             | Grupo      | Livros           | Não      | Não         |
| [20]                     | Grupo      | Mensagens        | Não      | Não         |
| e-LORS [21]              | Individual | OA               | Não      | Não         |
| [22]                     | Individual | OA               | Não      | Não         |
| MobiLE [23]              | Individual | OA               | Sim      | Sim         |
| UbiGroup                 | Grupo      | OA               | Sim      | Sim         |

Tabela 5: Comparação entre os trabalhos relacionados.

Nesta tabela podem ser observados alguns diferenciais importantes do UbiGroup. Em termos de sensibilidade a contexto e ao suporte para dispositivos móveis, os trabalhos em sua maioria não apresentam estes recursos em seus modelos. O MobiLE [23] é o único que dispõe destes recursos. Nos trabalhos relacionados também não foi identificada a possibilidade de criação e alteração das regras de recomendação. Dentre os trabalhos que tinham

como área de aplicação o ambiente educacional, nenhum se propôs a dispor desta funcionalidade. As regras de recomendação eram previamente estabelecidas na aplicação. O UbiGroup permite tal funcionalidade, dando ao professor a possibilidade de definir a regra de recomendação que melhor se adéqua as suas necessidades de ensino. Os modelos propostos em sua maioria utilizam apenas as informações de título, descrição e palavras-chave dos metadados dos OAs. No UbiGroup o professor pode usar todas as informações disponíveis nos metadados dos OAs. Uma etapa de refinamento no processo de recomendação é a utilização de opiniões expressadas por outros usuários. Alguns trabalhos se propõem a considerar estas informações. Entretanto, um ponto a ser considerado é que são constantemente adicionados novos materiais aos repositórios. Como estes OAs não possuem avaliação, tendem a não ser recomendados. Esta situação é tratada pelo UbiGroup, pois o seu processo de recomendação prevê a indicação de pelo menos um material que não possua avaliação.

O trabalho [20] assemelha-se a proposta do UbiGroup, pelo fato de inferir automaticamente a formação de grupos, usar regras para selecionar de forma colaborativa o material recomendado e aplicar técnicas semânticas para filtrar o conteúdo retornado. No entanto, existem duas diferenças principais que consistem na contribuição do UbiGroup. As técnicas utilizadas para formar grupos dinamicamente e selecionar materiais baseadas nas informações de contexto. Esses aspectos não são considerados em [20]. Outra diferença está relacionada com detalhes específicos da abordagem semântica para filtrar os resultados. Diferentemente das técnicas utilizadas no dicionário [20], o UbiGroup usa uma abordagem ontológica completa para lidar com a semântica dos conteúdos dos OAs a serem filtrados.

Durante o desenvolvimento do UbiGroup foram identificadas algumas propostas de trabalhos futuros: (1) tratamento de múltiplos contextos, definindo regras que permitam o acesso dos alunos à contextos que interseccionam a mesma posição física; (2) realização de consultas a múltiplos repositórios de OAs; (3) explorar oportunidades pedagógicas por meio de trilhas [31] aplicadas as informações de ocorrência de relacionamentos (data, aluno e contexto) entre os alunos; (4) estender aplicação móvel para o sistema operacional IOs; (5) realizar uma avaliação com aprendizes, obtendo uma avaliação mais detalhada referente à usabilidade da ferramenta.

## Referências

- [1] Banco Internacional de Objetos Educacionais. <http://objetoseducacionais2.mec.gov.br/>, Abr. 2014.
- [2] Merlot - Multimedia Educational Resources for Learning and Ponline Teaching. <http://www.merlot.org/merlot/index.htm>, Abr. 2014.

- [3] RIVED – Rede Interativa Virtual de Educação. <http://rived.mec.gov.br/>, Abr. 2014.
- [4] Cesta - Coletânea de entidades de suporte ao uso de tecnologia na aprendizagem. <http://cesta2.cinted.ufrgs.br/xmlui/>, Abr. 2014.
- [5] D. A. Wiley. Connecting learning objects to instructional design theory: A definition, a metaphor, and a taxonomy. In D. A. Wiley (Ed.), *The Instructional Use of Learning Objects*. <http://reusability.org/read/chapters/wiley.doc>, Abr. 2014.
- [6] Std 1484.12.1 - IEEE Learning Technology Standard Committee (LTSC) - Standard for Learning Object Metadata (LOM). 2002.
- [7] The Dublin Core Metadata Initiative. <http://dublincore.org/>, Abr. 2014.
- [8] R. Vicari, J. Gluz, L. Passerino. The OBAA Proposal for Learning Objects Supported by Agents. *Proceedings of MASEIE Workshop – AAMAS*, Toronto, Canada, 2010.
- [9] D. Barbosa. Learning in a Large-Scale Pervasive Environment”. *2nd IEEE Int. W. on Pervasive Computing (PerEl)*, New York, IEEE Press. 2006
- [10] M. Weiser. The Computer for the Twenty-First Century. *Scientific American*, pages 94–10. 1991.
- [11] A. Dey, G. Abowd, D. Salber. Conceptual Framework and a Toolkit for Supporting the Rapid Prototyping of Context-Aware Applications. *Human-computer Interaction*. Vol. 16, 2001.
- [12] L. Boratto, S. Carta. State-of-the-Art in Group Recommendation and New Approaches for Automatic Identification of Groups. *Studies for Computational Intelligence*, Springer Verlag, 2010.
- [13] G. Adomavicius, A. Tuzhilin. Toward the Next Generation of Recommender Systems: A Survey of the State-of-the-Art and Possible Extensions. *IEEE Transactions on Knowledge and Data Engineering*, p. 734-749, Jun. 2005.
- [14] M. Satyanarayanan. Pervasive computing: vision and challenges. *Personal Communications*, IEEE, v. 8, n. 4, p. 10-17. 2001.
- [15] J. Odell. *Agent Technology: An Overview*. [http://www.jamesodell.com/Agent\\_Technology-An\\_Overview.pdf](http://www.jamesodell.com/Agent_Technology-An_Overview.pdf), Abr. 2014.
- [16] S. Russel, P. Norvig. *Artificial Intelligence: A Modern Approach*. p.932. ed. New Jersey: Prentice-Hall, 1995.
- [17] L. Boratto. Group Identification and Individual Recommendations in Group Recommendation Algorithms. *Practical Use of Recommender Systems, Algorithms and Technologies (PRSAT)*. 2010.
- [18] T. Gross, C. Beckmann, M. Schirmer. GroupRecoPF: Innovative Group Recommendations in a Distributed Platform. *19<sup>th</sup> Internacional Euromicro Conference on Parallel, Distributed and Network-based Processing*, IEEE. 2011.
- [19] J. Kim, H. Kim, H. Oh, Y. Ryu. A group recommendation system for online communities. Elsevier, *International Journal of Information Management*, v.30, n.3, p.212–219, 2009.
- [20] A. A. Kardan, M. Ebrahimi. A novel approach to hybrid recommendation systems based on association rules mining for content recommendation in asynchronous discussion groups. *Information Sciences*, v. 219, p. 93-110, 2013.
- [21] L. Zaina, G. Bressan, M. Cardieri, J. Rodrigues. e-LORS: Uma Abordagem para Recomendação de Objetos de Aprendizagem. *Revista Brasileira de Informática na Educação*, v.20, n.1, 2012.
- [22] S. Cazella, K. Silva, P. Bhear, D. Schneider; R. Freitas. Recomendando Objetos de Aprendizagem baseado em Competências em EAD. *RENOTE - Revista Novas Tecnologias na Educação - CINTED-UFRGS*, v.9, n.2, Porto Alegre, 2011.
- [23] L. Silva, F. Neto, L. Junior. MobiLE: Um ambiente Multiagente de Aprendizagem Móvel para Apoiar a Recomendação Sensível ao Contexto de Objetos de Aprendizagem. *22<sup>o</sup> Simpósio Brasileiro de Informática na Educação*, Aracaju, 2011.
- [24] PAPI - Draft standard for learning technology. public and private information for learners (PAPI Learner), IEEE. 2000.
- [25] V. Levenshtein. Binary codes capable of correcting deletions, insertions. *Soviet Physics Doklady*, v.10, n.8, p.707-710, 1966.
- [26] J. C. Gluz, R. M. Vicari. An OWL Ontology for IEEE-LOM and OBAA Metadata. *Proceedings of ITS 2012*, Chania, Crete. LNCS. New York: Springer, v.7315. p.696 – 698, 2012.
- [27] Query Language for RDF (SPARQL) – W3C. <http://www.w3.org/TR/rdf-sparql-query>, Abr. 2014.
- [28] M. Kich, J. Barbosa, M. C. Kich, A. Klein. Um Modelo Sensível ao Contexto Orientado à Gestão por Competências. *Revista Brasileira de Informática na Educação*, v. 21, p. 69-84, 2014.
- [29] A. Wagner, J. Barbosa, D. Barbosa. A Model for Profile Management Applied to Ubiquitous Learning Environments. *Expert Systems with Applications*, v. 41, p. 2023-2034, 2014.
- [30] L. R. J. Silva, J. C. Gluz. MSSearch: Busca Semântica de Objetos de Aprendizagem OBAA com Suporte a Alinhamento Automático de Ontologias. *Anais do XXIII SBIE*, Rio de Janeiro, 2012.
- [31] J. Silva, J. Rosa, J. Barbosa, D. Barbosa, L. Palazzo. Content Distribution in Trail-aware Environments. *Journal of the Brazilian Computer Society (Impresso)*, v. 16, p. 163-176, 2010.