

UNIVERSIDADE DE CAXIAS DO SUL  
CENTRO DE COMPUTAÇÃO E TECNOLOGIA DA INFORMAÇÃO  
BACHARELADO EM CIÊNCIA DA COMPUTAÇÃO

MATHEUS PEREIRA

**Um sistema de *following* para  
interação em Comunidades de Prática**

Elisa Boff  
Orientadora

João Luis Tavares da Silva  
Coorientador

Caxias do Sul, Dezembro de 2013

*"Seja a mudança que você quer ver no mundo."*

DALAI LAMA

## AGRADECIMENTOS

Aos meus pais, José Silmar Ferreira Pereira e Clarice Degasperi Pereira, pelo empenho em sempre oferecer as melhores condições possíveis, diria até impossíveis, para minha educação e desenvolvimento como pessoa.

Aos meus avós, Eliseu Vitorino Pereira e Judite Ferreira Pereira, pois estiveram sempre presentes e ensinaram-me lições importantes desde o princípio. Em especial, durante os últimos quatro anos e meio, fui acolhido em sua residência, pude conviver e aprender com eles valores que carregarei comigo para sempre. Posso considerá-los como meus segundos pais.

À minha irmã, Gisele Degasperi Pereira, pela oportunidade de crescer e aprender juntos, além de todos meus familiares, não citados aqui.

À Dra. Elisa Boff que durante todo o desenvolvimento deste trabalho dedicou seu tempo e orientou as atividades e esforços para que a execução do mesmo fosse possível.

Ao Dr. João Luis Tavares da Silva mentor e co-orientador deste trabalho. Professor e amigo, esteve sempre disposto a colaborar para que o desenvolvimento e sucesso do trabalho fosse alcançado.

Ao Dr. Tiago Thompsen Primo companheiro de infindáveis conversas dos mais diversos assuntos, desde representações de interesses de usuários para recomendações até a forma como fazer um bom chimarrão ou capuccino.

Aos meus amigos e colegas de trabalho e estudos Luciano Camargo Cruz, João Toss Molon e Felipi Medeiros Macedo.

Ao Dr. Alexandre Moretto Ribeiro e ao Dr. João Luis Tavares da Silva, pela oportunidade profissional que proporcionam a todos os colegas na HaDi.Com e Instituto Communitas.

A todos aqueles que de alguma forma colaboraram para este trabalho e, por motivos desconhecidos, não foram citados nesses agradecimentos.

A todos vocês, minha sincera gratidão.

Matheus Pereira

# SUMÁRIO

<b>LISTA DE ACRÔNIMOS</b> . . . . .	6
<b>LISTA DE FIGURAS</b> . . . . .	8
<b>LISTA DE TABELAS</b> . . . . .	10
<b>LISTA DE TRECHOS DE CÓDIGO</b> . . . . .	11
<b>RESUMO</b> . . . . .	12
<b>ABSTRACT</b> . . . . .	13
<b>1 INTRODUÇÃO</b> . . . . .	14
1.1 Objetivo e organização do texto . . . . .	15
<b>2 TECNOLOGIA DE COMUNIDADES DE PRÁTICA</b> . . . . .	16
2.1 Orientações e Atividades Colaborativas . . . . .	18
2.2 Ferramentas de colaboração e comunicação . . . . .	21
2.3 Conclusões sobre o capítulo . . . . .	23
<b>3 SISTEMAS DE RECOMENDAÇÕES</b> . . . . .	24
3.1 Definições . . . . .	24
3.2 Técnicas de recomendação . . . . .	27
3.2.1 Filtragem Colaborativa . . . . .	28
3.2.2 Filtragem Baseada em Conteúdo . . . . .	29
3.2.3 Filtragem Demográfica . . . . .	29
3.2.4 Filtragem Baseada em Utilidade . . . . .	30
3.2.5 Filtragem Baseada em Conhecimento . . . . .	30
3.2.6 Filtragem Híbrida . . . . .	30
3.2.7 Descritores de Itens . . . . .	31
3.3 Conclusões sobre o capítulo . . . . .	32

<b>4</b>	<b>PLATAFORMA DE COMUNIDADES DE PRÁTICA . . . . .</b>	<b>34</b>
4.1	Tecnologia . . . . .	35
4.1.1	Python . . . . .	35
4.1.2	Zope . . . . .	35
4.1.3	Plone . . . . .	36
4.2	<i>Framework</i> . . . . .	38
4.2.1	Tipos de conteúdo . . . . .	38
4.2.2	Ferramentas implementadas . . . . .	38
4.2.3	Temas, usuários e seus relacionamentos . . . . .	39
4.3	Conclusões sobre o capítulo . . . . .	40
<b>5</b>	<b>SISTEMA DE FOLLOWING . . . . .</b>	<b>41</b>
5.1	Requisitos . . . . .	41
5.1.1	Casos de uso . . . . .	42
5.2	Arquitetura . . . . .	50
5.2.1	Banco de dados . . . . .	52
5.2.2	Camada de persistência . . . . .	53
5.2.3	Camada de negócio . . . . .	54
5.2.4	Camada de apresentação . . . . .	56
5.3	Conclusões sobre o capítulo . . . . .	56
<b>6</b>	<b>EXTRAINDO ATRIBUTOS DO SISTEMA DE FOLLOWING . . . . .</b>	<b>57</b>
6.1	Representação do <i>following</i> . . . . .	57
6.2	Cálculo de similaridade . . . . .	58
6.2.1	Similaridade do cosseno . . . . .	59
6.2.2	Força do relacionamento . . . . .	59
6.3	Filtragem colaborativa item-item . . . . .	61
6.4	Cenários de recomendação . . . . .	61
6.4.1	Cenário de recomendação 1 . . . . .	62
6.4.2	Cenário de recomendação 2 . . . . .	63
6.4.3	Cenário de recomendação 3 . . . . .	64
6.4.4	Cenário de recomendação 4 . . . . .	65
6.5	Conclusões sobre o capítulo . . . . .	66
<b>7</b>	<b>CONSIDERAÇÕES FINAIS . . . . .</b>	<b>67</b>
7.1	Contribuições . . . . .	68
7.2	Trabalhos futuros . . . . .	68
	<b>REFERÊNCIAS . . . . .</b>	<b>69</b>

## LISTA DE ACRÔNIMOS

<b>AJAX</b>	<i>Asynchronous Javascript and XML</i>
<b>CdP</b>	comunidade de prática
<b>CoPPLA</b>	<i>Communities of Practice Plataform</i>
<b>CMF</b>	<i>Content Management Framework</i>
<b>CMS</b>	<i>Content Management System</i>
<b>DTML</b>	<i>Document Template Markup Language</i>
<b>GUI</b>	<i>Graphical User Interface</i>
<b>GPL</b>	<i>General Public License</i>
<b>HTML</b>	<i>HyperText Markup Language</i>
<b>I18N</b>	<i>Internationalization</i>
<b>LDAP</b>	<i>Lightweight Directory Access Protocol</i>
<b>ORM</b>	<i>Object Relational Mapper</i>
<b>PAS</b>	<i>Pluggable Authentication Service</i>
<b>PZP</b>	Python/Zope/Plone
<b>TAL</b>	<i>Template Attribute Language</i>
<b>TALES</b>	<i>Template Attribute Language Expression Syntax</i>
<b>ZCML</b>	<i>Zope Configuration Mark-up Language</i>
<b>ZMI</b>	<i>Zope Management Interface</i>
<b>ZODB</b>	<i>Zope Object Data Base</i>

<b>Zope</b>	<i>Z Object Publishing</i>
<b>ZPT</b>	<i>Zope Page Template</i>
<b>ZTK</b>	<i>Zope Toolkit</i>



## LISTA DE FIGURAS

Figura 2.1: Ciclo de vida de comunidades de prática (WENGER et al., 2005).	17
Figura 2.2: Quadro sinóptico das atividades relacionando ferramentas tecnológicas às tensões encontradas em comunidades de prática (WENGER et al., 2005).	22
Figura 3.1: Estrutura do Sistema de Recomendação (SCHAFER; KONSTAN; RIELD, 2000).	25
Figura 3.2: Avaliações de usuários (estrelas) no site Amazon.com.	26
Figura 3.3: Filtragem Híbrida (CAZELLA, 2006).	31
Figura 3.4: Exemplo de um descritor de item (REATEGUI; LORENZATTI, 2005).	31
Figura 4.1: Interface de uma comunidade do <i>Framework CoPPLA (Communities of Practice Plataform)</i> .	34
Figura 4.2: <i>Workflow</i> padrão para publicação de objetos em Plone.	37
Figura 4.3: Permissões para o estado Pendente do <i>workflow</i> de publicação.	37
Figura 5.1: Casos de uso do sistema de <i>following</i> .	42
Figura 5.2: Ação de <i>Follow / Unfollow</i> disponível na lista de participantes das comunidades.	45
Figura 5.3: Ação de <i>Follow / Unfollow</i> disponível no perfil do usuário.	45
Figura 5.4: Gerenciamento do <i>following</i> de temas.	46
Figura 5.5: <i>Portlet</i> para visualização do <i>following</i> de temas.	47
Figura 5.6: Visualização do <i>following</i> de usuários.	47
Figura 5.7: <i>Portlet</i> para visualização do <i>following</i> de temas.	48
Figura 5.8: Conteúdos categorizados com o tema "Escola".	49
Figura 5.9: Lista de usuários que seguem o tema "Escola".	50
Figura 5.10: Modelo de aplicação em três camadas para o sistema de <i>following</i> .	51
Figura 5.11: Modelo da estrutura de banco de dados para o sistema de <i>following</i> .	53
Figura 5.12: Diagrama de classe para camada de persistência.	54

Figura 6.1: Exemplo de uma estrutura de <i>following</i> . . . . .	58
Figura 6.2: Representação do cenário de recomendação 1. . . . .	62
Figura 6.3: Tabela de similaridade para o cenário de recomendação 1. . . . .	63
Figura 6.4: Representação do cenário de recomendação 2. . . . .	63
Figura 6.5: Tabela de similaridade para o cenário de recomendação 2. . . . .	64
Figura 6.6: Representação do cenário de recomendação 3. . . . .	64
Figura 6.7: Tabela de similaridade para o cenário de recomendação 3. . . . .	65
Figura 6.8: Representação do cenário de recomendação 4. . . . .	65
Figura 6.9: Tabela de similaridade para o cenário de recomendação 4. . . . .	66

## LISTA DE TABELAS

Tabela 3.1: Técnicas de Recomendação (BURKE, 2002) . . . . .	27
Tabela 3.2: Comparativo entre técnicas de recomendação (Adaptada de BURKE (2002)). . . . .	33
Tabela 5.1: Cadastrar usuário - Caso de uso 1 . . . . .	43
Tabela 5.2: Cadastrar usuário - Caso de uso 2 . . . . .	43
Tabela 5.3: Cadastrar tema - Caso de uso . . . . .	44
Tabela 5.4: Categorizar conteúdo - Caso de uso . . . . .	44
Tabela 5.5: <i>Follow / Unfollow</i> de usuário - Caso de uso . . . . .	45
Tabela 5.6: <i>Follow / Unfollow</i> de tema - Caso de uso . . . . .	46
Tabela 5.7: Visualizar <i>following</i> de usuário - Caso de uso . . . . .	47
Tabela 5.8: Visualizar <i>following</i> de tema - Caso de uso . . . . .	48
Tabela 5.9: Visualizar conteúdos categorizados pelo tema - Caso de uso . . .	49
Tabela 5.10: Visualizar usuários que seguem tema - Caso de uso . . . . .	50

## LISTA DE TRECHOS DE CÓDIGO

5.1	<i>Zope Page Template</i> para seguir usuário. . . . .	54
5.2	Código SQLAlchemy para seguir usuário. . . . .	55
5.3	Código Python para seguir usuário. . . . .	55
5.4	Código Python para seguir usuário. . . . .	55
5.5	Código Python para seguir usuário. . . . .	56
6.1	Busca em amplitude para cálculo da força do relacionamento entre usuários. . . . .	60

## RESUMO

Este trabalho apresenta um sistema de following, de usuários e temas, desenvolvido para ampliar e estimular a interação entre os participantes de comunidades de prática. Neste ambiente, conhecer e interagir com outros indivíduos é fundamental no processo de aprendizagem e a existência de ferramentas que apoiem as práticas é o que define o sucesso de uma comunidade. Esta rede de relacionamentos representa um interesse distribuído dos participantes por determinados assuntos. O sistema de following captura e representa estes interesses de modo a considerá-los para recomendações na plataforma CoPPLA. É por meio destas sugestões oferecidas pelo sistema de recomendação, que os usuários são convidados a colaborar com outros participantes ou a conhecer novos temas de interesse.

**Palavras-chave:** Comunidades de Prática, Ambientes de Aprendizagem, Colaboração, Sistemas de Recomendação, Python, Zope, Plone.

## English Abstract

# ABSTRACT

This paper presents a Following System, for users and topics, designed to improve and to encourage interaction among participants in communities of practice. In this environment, is very important to meet and to interact with other individuals in the learning process and also the supply of tools to support the practices is what defines the success of a community. This relationships network can be compared with an interest of participants distributed by certain subjects. The Following System captures and represents these interests in order to consider them for recommendations on the CoPPLA platform. It is through these suggestions offered by the recommendation system, that users are invited to collaborate with other participants or to discover new topics of interest

**Keywords:** Communities of Practice, Learning Environments, Collaboration, Recommendation Systems, Python, Zope, Plone.

# 1 INTRODUÇÃO

Devido à rápida expansão e democratização da internet, a produção de conteúdo segue em ritmo acelerado. Como resultado, os usuários são expostos a uma grande oferta de informação (ZAIANE, 2000), dificultando a busca por conteúdos específicos, tornando complexa a atividade de aprendizagem.

Este trabalho enquadra-se no contexto de aprendizagem colaborativa através da interação de usuários em comunidades virtuais. Em FIORIO; SILVA; RIBEIRO (2011) e RIBEIRO et al. (2011) é proposto um *framework* de comunidades de prática com o objetivo de oferecer uma plataforma de construção de comunidades virtuais voltado ao desenvolvimento de práticas e habilidades relacionadas a um domínio específico. O principal foco desta abordagem é a modelagem de um *framework* que usa um modelo de referência ontológico para representar uma comunidade, seus participantes, o conhecimento e os perfis de interesse e domínio em um contexto de comunidades de prática. Segundo WENGER (1998), comunidades de prática consistem em grupos de pessoas que compartilham determinado interesse comum. A aprendizagem e a produção do conhecimento ocorrem a partir da interação dos indivíduos. Desta maneira, a comunidade de prática atribui ao aprendiz não somente o papel de consumidor, mas também de produtor da informação, desenvolvendo a competência de seus membros que, com o tempo, passam a ser especialistas e socializadores do conhecimento.

A plataforma de comunidades de prática em questão, *CoPPLA* (*Communities of Practice Platform*), compreende um conjunto de ferramentas que possibilita a interação de seus participantes. Estes tem a habilidade de criar conteúdos diversos, como textos, imagens, páginas web, eventos, fóruns de discussão, entre outros. A plataforma faz com que a criação e a manipulação destes objetos sejam mais flexíveis e intuitivas, promovendo maior participação dos usuários. A ampla atividade de pessoas nestas comunidades de aprendizagem gera uma grande quantidade de conteúdo que precisa ser acessado e encontrado por seus pares.

Uma das funcionalidades a ser desenvolvida é a possibilidade de conhecer e seguir (*Follow/Following*) outros participantes e temas. Temas são termos de categorização

do conhecimento relacionados a cada conteúdo compartilhado nas comunidades que representam o domínio da comunidade. A possibilidade de seguir temas define uma parte dinâmica do interesse do usuário, que pode evoluir com as interações nas comunidades. Por outro lado, seguir pessoas é uma função agora tradicional em redes sociais que relaciona o conteúdo postado por determinada pessoa aos interesses do seguidor, colocando este conteúdo a disposição de forma implícita.

A principal forma de disponibilizar estes conteúdos e relacionamentos é através de recomendação. Um dos enfoques deste trabalho é definir como representar estes interesses, por temas e pessoas, através de uma estrutura que possa ser utilizada por sistemas de recomendações em comunidades de prática. Os sistemas de recomendação surgem com a proposta de recuperar informações e apresentar resultados personalizados conforme o perfil do usuário. Diferentes técnicas são empregadas para produzir recomendações, dentre elas pode-se citar a Filtragem Baseada em Conteúdo, a Filtragem Colaborativa e a Filtragem Híbrida (CAZELLA; NUNES; REATEGUI, 2010).

Com base neste cenário, este projeto envolve um estudo sobre técnicas de recomendação para estimular a colaboração em comunidades de prática e ampliar a interação entre usuários e temas de interesse, buscando responder a seguinte questão de pesquisa: Como a relação entre temas e Usuários pode ser utilizada na recomendação em comunidades de prática?

## 1.1 Objetivo e organização do texto

O principal objetivo deste trabalho é desenvolver uma solução para relacionar usuários e temas, em comunidades de prática, de forma que os interesses possam ser armazenados e recuperados para servir como base para recomendações.

O Capítulo 2 apresenta conceitos relativos a tecnologia de comunidades de prática, seus elementos estruturantes, ciclo de vida, orientações e principais ferramentas utilizadas para sua instrumentalização. O Capítulo 3 aborda conceitos relativos aos sistemas de recomendação e identifica as principais técnicas de recomendação, relacionando-as ao contexto de comunidades de prática. O Capítulo 4 consiste no estudo da plataforma *CoPPLA*, sua tecnologia e ferramentas já desenvolvidas.

O sistema de *following* é descrito em detalhes no Capítulo 5. O Capítulo 6 demonstra como o *following* armazena e recupera informações relevantes para recomendação. O último capítulo apresenta as conclusões, contribuições, atividades e aplicações futuras verificadas durante o desenvolvimento do presente trabalho.



## 2 TECNOLOGIA DE COMUNIDADES DE PRÁTICA

O conceito comunidade de prática é relativamente novo e pode ser definido como grupos de pessoas, que compartilham um interesse ou um conjunto de problemas e que aprendem através de interações regulares. A intenção de aprender em comunidades de prática pode não ser explícita, mas suas discussões, atividades e relacionamentos constroem uma história de aprendizagem conjunta. O fenômeno de organizar-se em comunidades é antigo, pode-se observar sua formação em ambientes como escola, trabalho e atividades de lazer. Grupos como vizinhos de um bairro, amigos que partilham interesse por determinado esporte ou colegas de trabalho podem ser considerados comunidades, mas segundo WENGER (2008), para que eles possam ser considerados comunidades de prática, seus elementos estruturais devem atender a três características fundamentais:

**Domínio** - Uma rede de pessoas conectadas não define, por si só, uma comunidade de prática. Ela deve possuir um propósito que orienta suas atividades e estabelece suas fronteiras, um domínio de interesses partilhado. O domínio também define a identidade de seus membros, uma vez que estes possuem competências que os diferem de pessoas que não fazem parte do grupo.

**Comunidade** - Os membros, ao desenvolverem atividades e discussões em conjunto, estabelecem relacionamentos e compartilham informações. Estas relações proporcionam aos participantes aprender uns com os outros e são essenciais para a caracterização da comunidade.

**Prática** - É um conjunto de esquemas de trabalho, ferramentas, ideias, estilos, linguagem, histórias e documentos decorrentes das interações na comunidade. A prática representa o conhecimento desenvolvido, compartilhado e armazenado, que torna-se fonte de conhecimento para seus membros e outros que possam se beneficiar destes recursos.

Segundo WENGER (2000), para que as comunidades de prática sejam bem sucedidas é necessário que elas tenham a capacidade de serem sistemas de aprendizagem social.

HEALY (2009) define aprendizagem como o processo de evolução de um estado sem conhecimento para um onde há conhecimento. O aspecto social está nas interações humanas, aprender através do contato e associações regulares. A possibilidade de interagir e trocar experiências, fazendo com que os membros aprendam sobre um domínio e também sobre a própria participação, torna a comunidade de prática apta a suportar a aprendizagem social, ao mesmo tempo que estimula o processo de participação e aprofundamento do domínio de aprendizagem (RIBEIRO et al., 2011).

As relações e o sentimento de pertencer à comunidade são fundamentais no processo de aprendizagem social. A partir do momento que o participante adquire confiança e se identifica com seus pares estas relações se intensificam. Inicia-se um processo de motivação intrínseca, onde os participantes sentem-se a vontade para participar de discussões, expor suas ideias e socializar o conhecimento.

As comunidades de prática também apresentam um ciclo de vida (Figura 2.1), onde as etapas não possuem tempo definido de duração, mas estão diretamente ligadas a participação e interesse de seus integrantes. As etapas são: potencial/criação, início/expansão, maturação, ativa e dispersão/transformação.

Figura 2.1: Ciclo de vida de comunidades de prática (WENGER et al., 2005).



**Potencial/Criação** - Inicialmente pessoas com questões e necessidades similares formam redes imprecisas que, com o tempo, são agregadas e formam comunidades. Esta fase tem como principal objetivo constituir a comunidade em torno de seus três elementos estruturantes, domínio, comunidade e prática.

**Início/Expansão** - Em seguida existe um engajamento, por parte dos participantes, em atividades de aprendizagem que consolidam o grupo como comunidade, aumentando o grau de confiança e compreensão entre os indivíduos.

**Maturação** - A comunidade se responsabiliza por suas práticas e expande-se, estipula-se padrões e define-se agendas, ritmo e identidade própria. O foco passa a ser o desenvolvimento do conhecimento de forma coletiva.

**Ativa** - A comunidade passa a apresentar ciclos de atividades, onde a participação oscila entre picos altos e baixos. A comunidade de prática já está estabelecida e possui conhecimento sólido, mas necessita estar aberta a novas ideias e novos participantes. Novos relacionamentos devem ser criados para que a comunidade continue ativa, mesmo durante períodos de baixa atividade.

**Dispersão/Transformação** - Por desenvolverem novos interesses ou a comunidade tratar de um assunto desatualizado ou sem grande relevância, os participantes acabam se distanciando da comunidade. Neste momento a comunidade pode sofrer transformações que geram outras comunidades ou se dispersar.

O sucesso de uma comunidade de prática implica em compreender estas etapas e assegurar que as atividades e ferramentas atendam as necessidades de cada uma delas.

## 2.1 Orientações e Atividades Colaborativas

Em comunidades de prática o aprendizado é construído de diferentes maneiras e o modo com que os participantes interagem define seu estilo. A comunidade de prática pode apresentar mais de um estilo simultaneamente, e isso pode mudar ao longo do tempo. Cada estilo ou orientação é associado a um conjunto de ferramentas que proporcionam suporte ao padrão de suas atividades. WENGER; WHITE; SMITH (2010) definem nove diferentes orientações que comunidades de prática podem assumir: encontros, conversas em aberto, projetos, conteúdo, acesso a perícia (*expertise*), relacionamentos, participação individual, cultivar a própria comunidade e contribuir num contexto específico.

**Encontros** - Algumas comunidades tem ênfase em encontros regulares, onde os participantes sobretudo compartilham atividades por algum tempo. A participação visível dos membros é o que define a existência da comunidade. Flexibilidade suficiente na agenda para interações espontâneas, ritmo adequado de encontros,

com frequência e agenda adequados e prática de encontros da comunidade são fatores importantes para o sucesso das comunidades orientadas a encontros.

As principais atividades desenvolvidas em comunidades de prática com esta orientação são: interações síncronas e assíncronas, *feedback* para membros e tomada de decisão, participação à distância em encontros face a face, criação e distribuição de notas compartilhadas e/ou colaborativas para encontros online ou face a face.

**Conversas em aberto** - Comunidades onde as conversações são as principais forma de aprendizagem, que caracterizam e mantém a comunidade unida. Variedade suficiente de tópicos para manter o interesse, participação ativa de um segmento representativo da comunidade e arquivos bem organizados das conversações para evitar conversas circulares e ajudar iniciantes a se situarem são fundamentais para o sucesso deste estilo de comunidade.

Algumas das principais atividades desenvolvidas nestas comunidades são: conversas sobre tópicos, tradução entre línguas, desenvolvimento de subgrupos com controle de acesso, destaque de pontos chave de aprendizagem e arquivamento.

**Projetos** - As comunidades são focadas em projetos, na resolução de projetos específicos ou na produção de artefatos úteis. Normalmente os projetos envolvem subgrupos e os membros desenvolvem suas práticas juntos. Estas comunidades devem apresentar uma definição coletiva dos projetos relacionados ao domínio da comunidade, definir coordenação, liderança e comunicação adequada entre subgrupos e o resto da comunidade.

A Criação de conteúdo em conjunto, desenvolvimento de subgrupos, gestão de projetos e comunicação com o restante das comunidades são algumas das atividades desenvolvidas em comunidades de prática com esta orientação.

**Conteúdo** - Determinadas comunidades estão principalmente interessadas em criar, compartilhar e dar acesso a documentos, ferramentas e outros conteúdos. O conteúdo bem organizado é um recurso importante para os membros, logo, a comunidade deve organizá-lo cuidadosamente, refletindo a visão da comunidade sobre seu domínio. Também deve apresentar uma taxonomia flexível que permita crescimento e evolução, facilidade de publicação interna ou para um público mais amplo e boa capacidade de busca.

As atividades características dessas comunidades são: *upload* e compartilhamento de arquivos e documentos, comentários, anotações e discussão de conteúdo, acesso a conteúdo interno e externo, avaliação de contribuições e arquivamento.

**Acesso a perícia (*expertise*)** - Este estilo tem como objetivo fornecer acesso ao conhecimento da comunidade, tanto internamente quando externamente. Essas comunidades são focadas em responder perguntas e de forma colaborativa resolver

problemas. O conhecimento pode ser produzido e mantido por todo o grupo, ou por apenas um pequeno conjunto de peritos. O acesso a informação confiável ou às respostas dos peritos deve ser rápido, as pessoas devem saber onde encontrar o conhecimento. O grau de confiança de uma resposta é avaliado através da reputação de quem respondeu ou através de processos explícitos de validação.

Perguntas e respostas, localizar perícia, validar ou avaliar respostas e promover questões ainda não respondidas ou com respostas inadequadas, compartilhar problemas resolvidos e seguir peritos são algumas das atividades desenvolvidas nestas comunidades.

**Relacionamentos** - As comunidades possuem foco no relacionamento construído entre seus membros, enfatizando o aspecto interpessoal da aprendizagem social. Comunidades com essa orientação são voltadas a criar, sustentar e representar relacionamentos. A possibilidade de conhecer outras pessoas e proporcionar conexões informais entre elas são fatores importantes para este estilo de comunidade de prática.

As principais atividades deste estilo são: relacionar e encontrar outros participantes, descobrir informações expressas na identidade pessoal de cada usuário, saber quem está conectado na comunidade, podendo interagir informalmente e seguir outros participantes.

**Participação individual** - A aprendizagem, mesmo quando em grupo, acontece de forma individual. Pessoas carregam diferentes experiências, estilos de comunicação e aspirações quanto a sua participação na comunidade, o que torna o processo de aprendizagem heterogêneo. Em comunidades orientadas a participação individual os membros tem a possibilidade de desenvolver seu próprio estilo de participação e a comunidade acolhe, suporta e evolui com essa diversidade.

Personalização, assinaturas e outros mecanismos de alertas, disponibilizar modos de interação, gerenciar a participação individual publicamente, gerenciar a própria privacidade e prover suporte explícito para vários membros são atividades que caracterizam comunidades de participação individual.

**Cultivar a própria comunidade** - Tem como objetivo evoluir de forma organizada e eficaz, proporcionando um ambiente favorável ao trabalho. As atividades da comunidade são bem planejadas e seus materiais de referência são bem produzidos e organizados.

*Feedback* da comunidade, anúncios e outras informações enviadas diretamente aos membros, organização das interações e gratificações por empenho na comunidade são atividades características deste estilo.

**Contribuir num contexto específico** - O objetivo central de comunidades que apresentam essa orientação é servir a um contexto específico que caracteriza a identidade e a forma como a comunidade opera. Elas podem ser utilizadas dentro

de organizações, ter a missão de fornecer recursos de aprendizagem para o mundo ou buscar interações com outras comunidades cujo domínio complementa o seu próprio.

Criar interfaces, oferecendo conteúdo da comunidade para o exterior, constelações de comunidades relacionadas, convites públicos e recrutamento de membros são atividades presentes nesta orientação de comunidades.

A próxima seção enfoca as ferramentas tecnológicas que, quando combinadas adequadamente, se tornam apropriadas a estilos específicos de comunidades de prática.

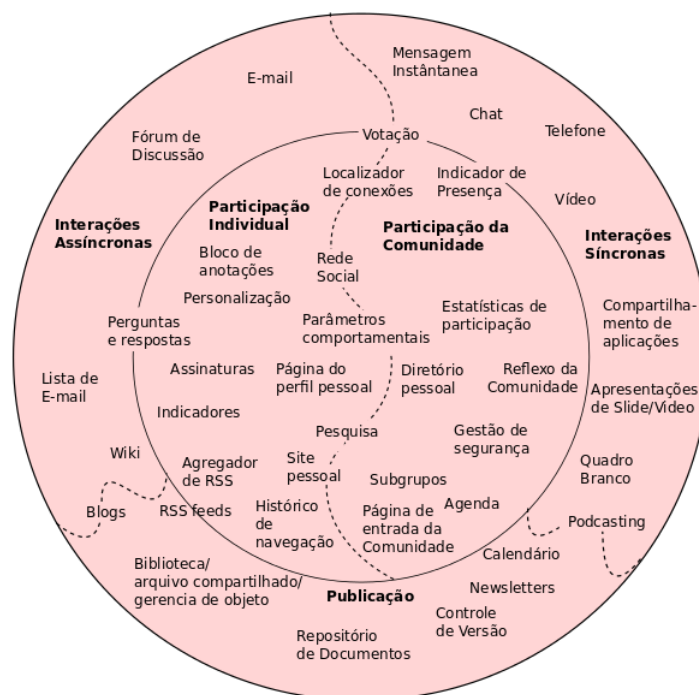
## **2.2 Ferramentas de colaboração e comunicação**

As comunidades de prática utilizam ferramentas tecnológicas tradicionais para sua instrumentalização, alterando apenas seus objetivos e formas de utilização. A orientação da comunidade é fundamental para a definição de suas ferramentas básicas. Aquelas que não se enquadram ao estilo tornam-se ociosas, enquanto que a falta de outras, que deveriam estar presentes, desmotivam e limitam a participação no ambiente.

WENGER et al. (2005) sugere diversas ferramentas para a instrumentalização em comunidades de prática, desde ferramentas de comunicação tradicionais como chat, fórum de discussão e e-mail até ferramentas mais sofisticadas como vídeo-conferência e podcast. Também são necessárias ferramentas de gestão e publicação (Figura 2.2). Sendo assim, a inovação não está nas ferramentas, mas sim no enfoque de seu uso, que permite aos membros da comunidade de prática a disseminação do conhecimento (SILVA; RIBEIRO; FIORIO, 2011).

Visando definir possibilidades tecnológicas que atendam as necessidades dos membros de uma comunidade de prática, WENGER et al. (2005) identifica as interfaces requeridas por meio de tensões presentes nas relações de troca entre os membros e a comunidade gerando três tipos de necessidades: Interação (síncrona e assíncrona), Publicação e Tendência (participação individual e cultivo da comunidade), ilustrado na Figura 2.2.

Figura 2.2: Quadro sinóptico das atividades relacionando ferramentas tecnológicas às tensões encontradas em comunidades de prática (WENGER et al., 2005).



O gerenciamento das relações entre as ferramentas de interação necessárias para colaboração nas comunidades de prática visa atender as necessidades de manutenção da comunidade. O uso das ferramentas tecnológicas tradicionais necessita de especificações orientadas aos estilos de comunidades. As ferramentas essenciais para a gestão das comunidades de prática são:

**Perfil** - É fundamental que os participantes possam visualizar o perfil dos outros usuários. A partir do perfil é possível verificar experiências e conhecimentos dos indivíduos, o que permite a definição de mapeamentos de especializações e competências comuns.

**Perfil da comunidade de prática** - A comunidade de prática deve apresentar alguns elementos estruturantes como moderadores, regras, domínio de interesse e objetivos. Esse perfil é definido pelo moderador que possui acesso diferenciado dos demais participantes.

**Lista de participantes** - Deve ser possível localizar todos os participantes de uma comunidade, observando assim seu Perfil e Domínio de Interesse. O conhecimento mútuo dos participantes contribui para o aumento de sua confiança, estimulando a participação na comunidade.

**Eventos** - É um importante mecanismo para a manutenção das atividades na comunidade. Eventos valorizam a comunicação, os participantes, divulgam resultados e familiarizam os membros, aumentando sua confiança e fortalecendo a comunidade. Os eventos podem ser tanto virtuais como presenciais.

**Níveis de atividade** - Acompanhar a frequência de colaboração permite a identificação do estágio do ciclo de vida da comunidade, bem como seu ritmo atual.

**Histórico** - Deve permitir a pesquisa por registros de interações, que precisam estar associados a seus produtores.

**Convites** - Participantes aprendem e produzem conhecimento a medida que interagem nas comunidades. Convidar outros usuários possibilita à comunidade de prática crescer, renovar ideias e continuar agregando conhecimento.

**Recomendações** - É necessário um mecanismo de busca e recomendações de interesses envolvidos na comunidade ou conjunto de comunidades que o usuário pertence, isso garante uma participação de diferentes níveis e mantém o canal de diálogo aberto.

As relações existentes entre as ferramentas, experiências e interesses pessoais fazem transparecer o domínio da comunidade de prática. Através deles é possível a inferência de interesses dos indivíduos para associá-los às comunidades, outros usuários ou temas que apresentam os mesmos interesses. Estas associações tem o objetivo de favorecer a disseminação do conhecimento, ampliar a abrangência das práticas e aumentar as relações, recomendando novas e fortalecendo as existentes.

Relações interpessoais não formam necessariamente uma comunidade de prática, mas à medida que condições favoráveis se apresentam, em termos de ferramentas e estrutura, existe uma grande possibilidade que destas relações possam emergir Comunidades de Prática com os mais diversos domínios.

## 2.3 Conclusões sobre o capítulo

Ao longo do capítulo foram apresentados os elementos estruturantes de uma comunidade de prática, seu ciclo de vida, orientações e ferramentas que se enquadram a cada um destes estilos. Também foi discutido como o conhecimento é produzido e de que maneira as interações colaboram para seu desenvolvimento neste contexto.

A troca de experiência entre os usuários é o principal objetivo do modelo de comunidades de prática. As ferramentas citadas no decorrer do capítulo proporcionam diferentes níveis de interação entre os usuário e o ambiente e, de todas estas, o trabalho concentra-se nas que incentivam a comunicação entre os participantes.



## 3 SISTEMAS DE RECOMENDAÇÕES

Atualmente a disponibilidade de informação na internet dificulta buscas precisas, a grande variedade de opções apresentadas a um indivíduo pode comprometer seu acesso ao conhecimento, muitas vezes este possui pouca experiência pessoal e não é capaz de discernir entre as várias alternativas que lhe são apresentadas (REATEGUI; CAZELLA, 2005; ALMAZRO et al., 2010).

Em um ambiente de comunidades de prática, onde as interações são incentivadas e a produção de conteúdo facilitada, a busca por informações específicas também pode ser desgastante. Neste cenário os sistemas de recomendação são ferramentas essenciais, que orientam e auxiliam o usuário a encontrar conteúdo relevante entre os itens disponíveis.

### 3.1 Definições

Sistemas de recomendação representam as preferências do usuário com o propósito de sugerir, de forma personalizada, novos itens de interesse. Eles são fundamentais para o comércio eletrônico e para o acesso a informação, sugerindo conteúdo, dentre um vasto espaço de possibilidades, que direcionam o usuário àqueles itens que melhor se enquadram em suas necessidades e preferências (BURKE, 2002).

Segundo CAZELLA; NUNES; REATEGUI (2010), a capacidade de personalização de um sistema depende de um conjunto específico de funções. Toda a interação do usuário deve ser monitorada, a fim de coletar informações, como itens visualizados, interesses e preferências, que são a base para manter-se um modelo de usuário. O sistema pode apresentar, esconder ou enfatizar fragmentos de uma página, a fim de exibir somente as informações apropriadas, em um nível adequado de dificuldade ou detalhe, ou ainda modificar essa apresentação de modo a sugerir ao usuário seus próximos passos.

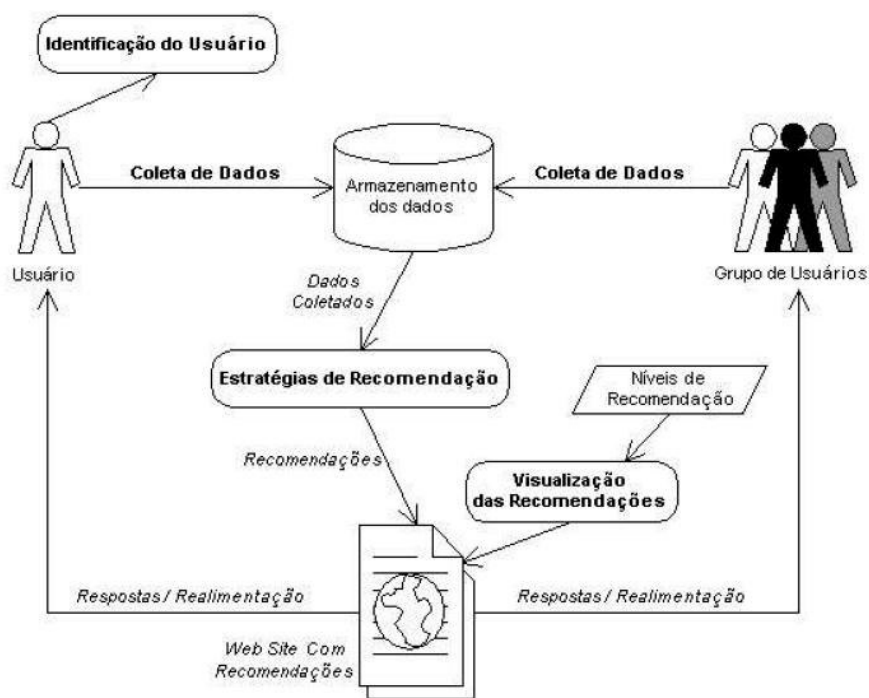
Na internet, a personalização de sistemas é largamente utilizada com o intuito de cativar o usuário. Os sistemas de recomendações são umas das ferramentas que

colaboram com essa personalização, já que são capazes de identificar preferências e sugerir itens relevantes para cada usuário a partir da análise de seu comportamento de navegação, histórico de interação e preferências.

Para que seja possível recomendar produtos, conteúdos ou pessoas a um usuário é necessário conhecê-lo. A eficiência das recomendações está diretamente relacionada à definição do perfil do usuário e à coleta de informações. Identificar qual o tipo de informação será relevante para a geração de recomendações é tão importante quanto capturá-la e armazená-la.

Segundo BARCELLOS et al.(2007, apud SCHAFER; KONSTAN; RIELD, 2000) a estrutura dos sistemas de recomendação é dividida em quatro etapas: identificação do usuário, coleta de informações, estratégias de recomendação e visualização das recomendações (Figura 3.1).

Figura 3.1: Estrutura do Sistema de Recomendação (SCHAFER; KONSTAN; RIELD, 2000).



O processo de identificação do usuário é opcional, porém, ao fazê-lo, é possível personalizar o sistema de acordo com as características daquele perfil. O perfil de um usuário expressa suas características, incluindo todas as informações diretamente solicitadas tal como as aprendidas durante sua interação no ambiente. O perfil é uma base de dados onde as informações sobre o usuário, incluindo seus interesses e preferências, são armazenadas e pode ser dinamicamente mantido (CAZELLA; NUNES; REATEGUI, 2010).

A coleta de informações para a construção do perfil do usuário pode ser feita de forma explícita ou implícita (BARCELLOS et al., 2007). Na coleta explícita,

o usuário indica o que lhe é importante, utilizando mecanismos como formulários, que exigem atenção, paciência e veracidade das informações fornecidas. Na forma implícita as informações são coletadas através do monitoramento das interações do usuário, páginas consultadas, tempo e frequência de visualização de conteúdo entre outros. O perfil do usuário é inferido com base no comportamento de outros usuários com padrão de comportamento similar ao seu.

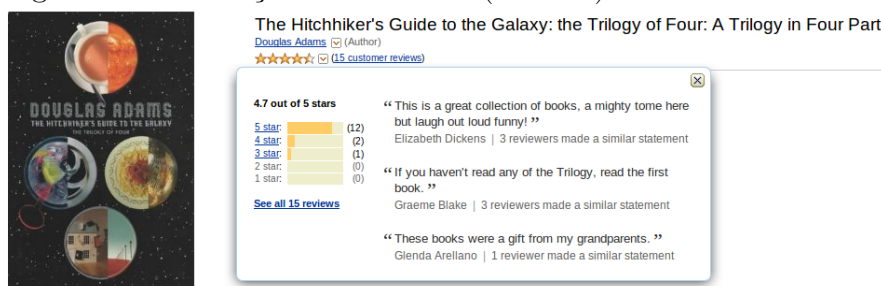
As recomendações precisam ser exibidas ao usuário de maneira que possam ser facilmente visualizadas e compreendidas. Elas podem ser apresentadas através de e-mail, lista de recomendações, navegação, atendimento online e personificação (BARCELLOS et al., 2007).

Segundo REATEGUI; CAZELLA (2005) diferentes estratégias, com diferentes graus de complexidade, podem ser utilizadas para recomendar conteúdo aos usuários.

**Listas de recomendação** - Mantém-se listas dos itens mais populares organizados por tipos de interesse, por exemplo, itens mais vendidos. Listas de recomendações são implementadas facilmente, mas suas sugestões não são dirigidas, o conteúdo a ser apresentado será o mesmo a todos, sem distinção de usuário.

**Avaliações de usuários** - Após acessar um item o usuário pode avaliar o item. Comumente são apresentados ícones, que possibilitam a indicação de quanto aquele item é de interesse do indivíduo (Figura 3.2). Do mesmo modo é possível representar seu interesse apenas de forma positiva, sem manter uma escala, assim como fazem redes sociais atuais como LinkedIn<sup>1</sup> (Gostei) e Facebook<sup>2</sup> (Curtir). As avaliações também são importantes pois asseguram a qualidade e utilidade dos itens. É a estratégia mais utilizada em sistemas de recomendações e sua implementação é simples, apenas sendo necessário armazenar e disponibilizar as avaliações de cada item, apresentando-os no momento apropriado.

Figura 3.2: Avaliações de usuários (estrelas) no site Amazon.com.



**Individualização** - As recomendações são geradas especificamente para cada usuário levando em consideração suas preferências explícitas e/ou implícitas.

<sup>1</sup><http://www.linkedin.com/>

<sup>2</sup><http://www.facebook.com/>

**Correlação** - É baseada na associação de itens que tendem a serem acessados de maneira conjunta. A correlação também pode ser definida através de regras de associação, que demonstram como os itens se relacionam uns com os outros (BARCELLOS et al., 2007).

**Associação por conteúdo** - Faz as recomendações baseada no conteúdo de determinado item, como um autor ou compositor, encontrando relações em um escopo mais restrito, por exemplo, dois filmes de determinado ator, que frequentemente são vendidos em conjunto.

### 3.2 Técnicas de recomendação

Segundo BURKE (2002) sistemas de recomendações possuem basicamente: dados de *background*, que é a informação que o sistema possui antes de iniciar o processo de recomendação; dados de entrada, que é a informação que o sistema necessita para gerar as recomendações; e um algoritmo, que combina os dados de *background* e de entrada para produzir suas sugestões. Identificando, deste modo, cinco diferentes técnicas de recomendação como mostrado na Tabela 3.1, onde  $I$  é o conjunto de itens sobre o qual as recomendações são feitas,  $U$  é o conjunto de usuários cujas preferências são conhecidas,  $u$  é o usuário para quem os itens são recomendados e  $i$  é o item que a preferência de  $u$  deve ser inferida.

Tabela 3.1: Técnicas de Recomendação (BURKE, 2002)

Técnica	Dados de <i>Background</i>	Dados de Entrada	Processo
<b>Colaborativa</b>	Avaliações de $U$ de itens em $I$ .	Avaliações de $u$ de itens em $I$ .	Identificar os usuários em $U$ similares a $u$ , e a partir deles inferir suas avaliações de $i$
<b>Baseada em Conteúdo</b>	Características de itens em $I$ .	Avaliações de $u$ de itens em $I$ .	Gerar um classificador que ajuste o comportamento de avaliação de $u$ para usá-lo em $i$ .
<b>Demográfica</b>	As informações demográficas de $U$ e suas classificações de itens em $I$ .	Informações demográficas de $u$ .	Identificar os usuários que são demograficamente semelhantes a $u$ , e a partir deles inferir suas avaliações de $i$ .
<b>Baseada em Utilidade</b>	Características de itens em $I$ .	Uma função de utilidade sobre os itens em $I$ que descrevem as preferências de $u$ .	Aplicar a função aos itens e determinar a classificação de $i$ .
<b>Baseada em Conhecimento</b>	Características de itens $I$ . O conhecimento de como esses itens atendem às necessidades do usuário.	Uma descrição de interesses e necessidades de $u$ .	Inferir uma correspondência entre $i$ e as necessidades de $u$ .

Além destas, BURKE (2002) identifica a filtragem híbrida que, a partir da combinação de duas ou mais técnicas, busca se beneficiar das qualidades e reduzir

as limitações que cada uma pode apresentar. REATEGUI; CAMPBELL; TORRES (2004) ainda apresentam a técnica baseada em descritores de itens.

### 3.2.1 Filtragem Colaborativa

É a mais popular e madura das técnicas de recomendação. Importantes projetos de pesquisa, como Tapestry (GOLDBERG et al., 1992) e GroupLens (SARWAR et al., 1998), assim como sites de comércio eletrônico como Amazon.com<sup>3</sup> utilizam-se desta técnica.

A filtragem colaborativa tem em sua essência a troca de experiências entre usuários que possuem interesses comuns (CAZELLA; NUNES; REATEGUI, 2010). Os itens são filtrados levando-se em consideração avaliações feitas pelos usuários sobre um determinado assunto. A ideia básica é fazer com que participantes de determinada comunidade possam ser beneficiados pela experiência uns dos outros. Porém a maioria dos sistemas de filtragem colaborativa requer que os próprios usuários especifiquem o relacionamento de predição entre suas opiniões, ou indiquem os itens de interesse, ou indiquem pontuações dos itens pelos usuários (HERLOCKER, 2000). Um usuário de um sistema de filtragem colaborativa deve, portanto, pontuar cada item experimentado, indicando o quanto este item combina com sua necessidade de informação. A abordagem de filtragem colaborativa apresenta então uma média de pontuações para cada item com potencial de interesse. Assim, o sistema indica ao usuário itens que são considerados de interesse pelo grupo e evita os itens de menor interesse.

Em geral, nos sistemas tradicionais de filtragem colaborativa, a decisão sobre a recomendação baseia-se no histórico de avaliações comuns e no valor de predição previamente calculado. Tipicamente o perfil do usuário em um sistema colaborativo consiste em um vetor de itens e suas avaliações, que cresce conforme as interações do indivíduo com o sistema. Neste tipo de filtragem é possível identificar nichos de interesse distantes através do reconhecimento de similaridades entre usuários e a compreensão ou reconhecimento do conteúdo não é necessário.

Esta técnica pode apresentar limitações como: problema do primeiro avaliador, quando um novo item, por não ter sido avaliado, não é recomendado; pontuação esparsa, quando o número de usuários é pequeno em relação ao volume de informação; e problema de similaridade, quando determinado usuário tem preferências que variam do normal, dificultando a busca por usuários com gostos similares, o que gera recomendações pobres (REATEGUI; CAZELLA, 2005).

---

<sup>3</sup><http://www.amazon.com/>

### 3.2.2 Filtragem Baseada em Conteúdo

A filtragem baseada em conteúdo consiste em categorizar e definir associações entre itens. O item a ser recomendado é aquele que apresenta o maior nível de similaridade com as escolhas prévias do indivíduo. Esta técnica é chamada de filtragem baseada em conteúdo porque o sistema realiza a filtragem baseada na análise de conteúdo do item e no perfil do usuário (CAZELLA, 2006). O perfil de interesse do usuário é definido a partir de suas escolhas prévias e depende do método de aprendizagem empregado. Árvores de decisão, redes neurais e representações baseadas em vetores são utilizadas com esta finalidade (BURKE, 2002).

A filtragem puramente baseada em conteúdo ignora a preferência de outros usuários e pode apresentar três limitações (ADOMAVICIUS; TUZHILIN, 2005): a análise de conteúdo limitada, que é a dificuldade de categorizar itens pouco estruturados como imagens, áudio e vídeo; a super especialização, que ocorre quando o sistema recomenda apenas itens semelhantes a itens avaliados positivamente, não apresentando conteúdo diferente do perfil do usuário, ou ainda, não apresentando conteúdo com alta similaridade a outros já recomendados, a fim de evitar redundâncias; e o problema do novo usuário, que precisa avaliar um número suficiente de itens para que o sistema recomende novos conteúdos. Segundo ADOMAVICIUS; TUZHILIN (2005), novos usuário, com poucas avaliações, não seriam capazes de receber recomendações precisas.

### 3.2.3 Filtragem Demográfica

A filtragem demográfica visa categorizar o usuário a partir de seu perfil e então gerar recomendações baseado em sua classe demográfica. O perfil do usuário é criado a partir da classificação dos usuários em estereótipos que representam as características de uma classe de usuários. As informações são coletadas através de formulários de registros e são utilizadas para a caracterização dos usuários e de seus interesses (REATEGUI; CAZELLA, 2005).

O processo de recomendação consiste em cruzar os interesses de usuários da mesma categoria, encontrando itens ainda não avaliados pelo usuário alvo. A filtragem demográfica forma correlações entre pessoas assim como a filtragem colaborativa, mas requer menor esforço computacional pois não precisa raciocinar sobre os dados do usuário e tem claras as semelhanças entre os indivíduos.

KRULWICH (1997) utiliza esta técnica no sistema *Prizm* para gerar recomendações para mais de 40,000 usuários. O sistema classifica a população dos Estados Unidos em 62 grupos demográficos a partir de seus históricos de compras e leva em consideração mais de 600 variáveis, cada uma caracterizando um estilo de vida, compra ou atividade.

O benefício desta abordagem é que ela não necessita de um histórico de avaliações do usuário, diferente da filtragem colaborativa e baseada em conteúdo, mas apresenta limitações como o problema de similaridade entre usuários.

### 3.2.4 Filtragem Baseada em Utilidade

A filtragem baseada em utilidade não se preocupa em construir uma generalização sobre o perfil de seus usuários, o foco está em definir uma função de utilidade que pode ser aplicada ao conjunto de itens disponíveis. Esta técnica gera as recomendações baseada na utilidade calculada a partir da função para cada item ao usuário. O problema central está em como criar a função para cada usuário. Tête-à-Tête e PersonaLogic (GUTTMAN, 1998), sistemas que ajudam consumidores a identificar quais produtos melhor atendem suas necessidades, são exemplos de utilização desta técnica. O benefício da recomendação baseada em utilidade é que ela pode levar em consideração atributos externos aos itens, como confiabilidade do fornecedor e disponibilidade, recomendando, por exemplo, itens em liquidação a usuários com necessidades imediatas (BURKE, 2002).

### 3.2.5 Filtragem Baseada em Conhecimento

A filtragem baseada em conhecimento tem ciência de como um item preenche determinada necessidade de determinado usuário, conhecimento funcional, podendo raciocinar sobre a relação entre uma necessidade e sua possível recomendação. O perfil do usuário pode ser qualquer estrutura de conhecimento que suporte esta inferência. Ele pode ser simplesmente uma consulta formulada pelo usuário como no site de buscas Google<sup>4</sup>, porém esta representação, das necessidades do usuário, pode ser mais detalhada. O conhecimento utilizado também pode assumir diversas formas. Google utiliza informações sobre links entre páginas para inferir a popularidade e a relevância de um item. Entretanto, os sistemas atuais não utilizam inferência, sendo necessário que o próprio usuário mapeie seus interesses.

### 3.2.6 Filtragem Híbrida

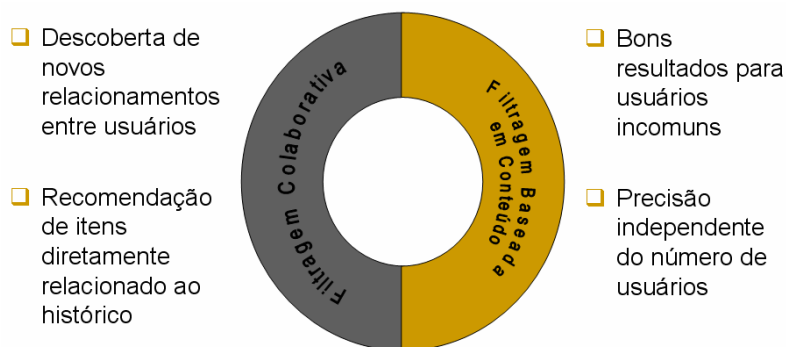
A filtragem híbrida procura aprimorar suas recomendações, através da combinação de duas ou mais abordagens, reduzindo suas limitações. É comum a filtragem colaborativa ser utilizada em conjunto com alguma outra a fim de reduzir o problema do primeiro avaliador, mas diversas combinações são possíveis (BURKE, 2002). Por exemplo, ao combinar as filtragens colaborativas e baseadas em conteúdo (Figura 3.3) é possível: implementar os dois métodos separadamente e combinar suas predições; incorporar características do método baseado em conteúdo na implementação do método colaborativo; incorporar características do

---

<sup>4</sup><https://www.google.com.br/>

método colaborativo na implementação do método baseado em conteúdo; ou construir um modelo unificado, incorporando características tanto do método baseado em conteúdo quanto do método colaborativo (ADOMAVICIUS; TUZHILIN, 2005).

Figura 3.3: Filtragem Híbrida (CAZELLA, 2006).



### 3.2.7 Descritores de Itens

Um descritor de item mantém características demográficas e comportamentais dos usuários para construir seu perfil e inferir suas preferências. As interações dos usuários são monitoradas e suas sequências de acesso são armazenadas com o intuito de aprender sobre seus padrões de interação (REATEGUI; CAZELLA, 2005).

Os descritores empregam parâmetros como fator de confiança (FConf) e fator de suporte (FSup) para indicar a força dos relacionamentos entre os itens. Em uma associação  $X \rightarrow Y$  o fator de suporte consiste na razão do número de usuários em que X e Y são verdadeiros, sobre o número total de usuários, enquanto que o fator de confiança consiste na razão do número de usuários em que X e Y são verdadeiros, sobre o número de usuários onde X é verdadeiro.

A figura 3.4, utilizada como exemplo no assistente virtual Cadinho (REATEGUI; LORENZATTI, 2005), apresenta um descritor onde o item alvo é a geração da série de Fibonacci, classificada como “Exercício”, e que tem “Estrutura Para..Faça”, “Cálculo da média dos números entre 8 e 80” e “Cálculo da soma números primos de 1 a 200” como itens relacionados.

Figura 3.4: Exemplo de um descritor de item (REATEGUI; LORENZATTI, 2005).

Item Alvo: Geração da Série de Fibonacci		Classe: Exercício	
Item Relacionado	Classe	FConf	FSup
Estrutura Para... Faça	Capítulo	0,70	0,62
Cálculo da média dos números entre 8 e 80	Exercício	0,59	0,57
Cálculo da soma números primos de 1 a 200	Exercício	0,52	0,52



A partir do fator de confiança é possível concluir que 70% dos alunos que acessam o capítulo “Estrutura Para..Faça” buscam resolver o exercício de geração da série de Fibonacci e que 62% de todos os alunos cadastrados acessam tanto o capítulo “Estrutura Para..Faça” quanto o exercício de geração da série de Fibonacci.

Para cada item a ser recomendado, o sistema cria um descritor e calcula a confiança entre o item alvo e os itens relacionados, provenientes da análise das interações do usuário. O sistema então calcula um *score* para o descritor que pode variar entre diferente (0) e similar (1). Calculado o *score* para cada descritor de item, aqueles que obtiverem o maior valor serão recomendados ao usuário.

### 3.3 Conclusões sobre o capítulo

Todas as técnicas analisadas durante o capítulo apresentam pontos fortes e fracos, dependendo do contexto onde são aplicadas. As vantagens e limitações são enumeradas a seguir.

#### Vantagens

1. Pode Identificar nichos de interesses diferentes dos já avaliados.
2. Conhecimento do domínio não é necessário.
3. Qualidade das recomendações cresce com o passar do tempo.
4. *Feedback* implícito é suficiente.
5. Itens e usuários não precisam de avaliações anteriores.
6. Sensível a alterações de preferências.
7. Pode incluir características externas ao produto.
8. Pode mapear necessidades do usuário para os produtos.

#### Desvantagens

9. Problema do novo usuário, cujo perfil não pode ser avaliado por falta de informações.
10. Problema do novo item, que por possuir poucas avaliações não é recomendado.
11. Problema de similaridade do usuário, quando um indivíduo tem gostos diferentes do padrão, que dificulta a busca por usuários semelhantes.
12. Qualidade depende do tamanho do conjunto de itens avaliados.
13. Problema de estabilidade x maleabilidade, quando o perfil do usuário mantém preferências antigas, que podem não ser do interesse atual do indivíduo.
14. Necessita de informações demográficas.
15. Usuário deve especificar a função de utilidade.
16. Habilidade de sugestão é estática.
17. Construção do conhecimento é necessária.

As características de cada técnica são apresentadas de forma resumida na Tabela 3.2.

Tabela 3.2: Comparativo entre técnicas de recomendação (Adaptada de BURKE (2002)).

<b>Técnica</b>	<b>Vantagem</b>	<b>Limitação</b>
<b>Colaborativa</b>	1, 2, 3, 4	9, 10, 11, 12, 13
<b>Baseada em Conteúdo</b>	2, 3, 4	9, 12, 13
<b>Demográfica</b>	1, 2, 3	9, 11, 12, 13, 14
<b>Baseada em Utilidade</b>	5, 6, 7	15, 16
<b>Baseada em Conhecimento</b>	5, 6, 7, 8	16, 17
<b>Descritores de Itens</b>	1, 3, 6, 8	9, 10, 12, 14

Levando em consideração que o requisito básico da plataforma *CoPPLA*, objeto deste trabalho, são as redes de relacionamento e os temas que conectam estas redes, apenas algumas características são importantes para nossa solução. Em relação aos itens de 1 a 4, a avaliação de outros interesses diferentes dos já avaliados é importante na medida que novos temas emergem nas interações dentro das comunidades; o conhecimento de domínio está implícito na comunidade e não precisa ser considerado em uma matriz de cálculos para recomendações; quanto mais proveitosas as recomendações, dando início a novas interações, melhor a sua qualidade na medida que as interações crescem em intervalos reduzidos de tempo.

As comunidades de prática focam principalmente nas trocas entre os parceiros e em quanto a necessidade de interação sobre determinado tema é inversamente proporcional ao tempo de aprendizagem em sua compreensão e aplicação, portanto não precisamos considerar os itens 7, 8 e 14 a 17.

A partir desta análise é possível identificar a técnica de filtragem colaborativa como sendo a que melhor se enquadra no contexto da aplicação de *following* de comunidades de prática, proposta neste trabalho. Através dela é possível gerar recomendações cruzadas, podendo sugerir itens completamente diferentes dos já avaliados, identificando usuários com preferências similares, declaradas explicitamente através do *following*, e indicando itens ainda não avaliados uns aos outros. Nesta técnica também não é necessário o reconhecimento do conteúdo a ser recomendado nem a coleta de informações demográficas.

## 4 PLATAFORMA DE COMUNIDADES DE PRÁTICA

O produto *Communities of Practice Plataforma*<sup>1</sup> provê uma plataforma genérica de comunidades de prática virtuais, disponibilizando uma série de ferramentas de comunicação e colaboração integradas em um ambiente voltado para o compartilhamento de conhecimento, onde a criação de conteúdo e a manipulação de objetos é flexível e intuitiva.

A plataforma oferece recursos para o gerenciamento da comunidade, possibilitando a criação, o armazenamento e o acesso a seus conteúdos e participantes. O conjunto de ferramentas de gestão, comunicação e publicação são configuráveis e envolvem a manipulação de textos, imagens, páginas web, links, eventos, fóruns de discussão e espaços para experiências de aprendizagem. Os usuários tem a habilidade de criar e gerenciar suas comunidades como um espaço para compartilhar conhecimento envolvendo atividades de aprendizagem.

Figura 4.1: Interface de uma comunidade do *Framework CoPPLA*.

The screenshot shows the Plone community interface for a Python community. At the top left is the Plone logo. To the right is a search bar with the text 'Buscar no Site' and a 'Buscar' button, with a checkbox for 'apenas nesta seção'. Below the search bar is a navigation menu with tabs for 'Página Inicial', 'Notícias', 'Eventos', 'Usuários', and 'Programação'. The main content area shows the Python logo and a welcome message from 'admin' posted on 16/06/2013 at 17:18. The right sidebar includes a search bar, a list of participants (2), and social media links for Twitter, Facebook, and YouTube.

<sup>1</sup><https://bitbucket.org/amribeiro/communities.practice/>

## 4.1 Tecnologia

O *framework* *CoPPLA* é desenvolvido utilizando a solução tecnológica PZP (Python/Zope/Plone) que disponibiliza um ambiente web para criação rápida e segura de sites, oferecendo diversas funcionalidades, por exemplo, login, restrições de acesso através de *workflows* que definem papéis e permissões aos usuários e mecanismos de criação, armazenamento, indexação e busca de conteúdo.

### 4.1.1 Python

Python é uma linguagem de programação de alto nível, interpretada, imperativa, orientada a objetos, de tipagem dinâmica e forte criada por Guido van Rossum no início dos anos 90 (PYTHON, 2013a).

Multipataforma e com licença livre, compatível com a *GPL* (*General Public License*), Python prioriza a legibilidade do código e o desenvolvimento rápido através de uma sintaxe clara e concisa <sup>2</sup>.

A linguagem conta com uma extensa biblioteca padrão e uma série de módulos e frameworks desenvolvidos por terceiros (PYTHON, 2013b), sendo utilizada em uma grande variedade de domínios de aplicação como desenvolvimento web e de jogos, *GUIs* (*Graphical User Interfaces*), banco de dados e educação (PYTHON, 2013c).

### 4.1.2 Zope

Zope é um servidor de aplicação web, de código aberto, escrito em Python, que oferece uma solução flexível, segura e escalável para criação, gerenciamento e distribuição de conteúdo (ZOPE, 2013a).

Seu nome é um acrônimo de *Z Object Publishing*. Os objetos publicados com Zope são escritos em Python e armazenados em um banco de dados orientado a objetos nativo, ZODB (*Zope Object Data Base*). Também escrito em Python, o ZODB persiste os objetos sem a necessidade de um mapeamento para tabelas de um banco relacional, indexando objetos e metadados que podem ser pesquisados por funções do Zope. A persistência dos objetos é transparente, tornando o código mais simples, robusto e legível (ZODB, 2013).

A indexação e pesquisa são responsabilidades do sistema de catálogo ZCatalog, que é composto por índices e metadados. Índices são campos que podem ser pesquisados e metadados são cópias de conteúdos dos objetos, que podem ser acessados sem a necessidade de acessar o próprio objeto (PLONE, 2013a). ZCatalog possui um mecanismo potente de consultas, podendo pesquisar por

---

<sup>2</sup>Parte da cultura da linguagem gira ao redor de *The Zen of Python* (PETERS, 2004), um conjunto de 19 aforismos que descrevem a filosofia de Python. Pode-se vê-lo através do comando Python:

```
>>> import this
```

múltiplos índices, inclusive atribuindo pesos a cada um deles. Como resultado de uma consulta, ZCatalog retorna uma lista de elementos chamados *Brains*. *Brains* são objetos que possuem todos os metadados vinculados àquele item como atributos e métodos para recuperação do objeto e de sua localização. *Brains* são extremamente rápidos pois são criados em tempo de execução, apenas no momento de uma requisição, e não instanciam os próprios objetos, evitando impactos de performance (ZOPE, 2013b).

Algumas vezes bancos de dados orientados a objetos podem não ser a melhor solução. A base de dados pode já existir e o Zope deve apenas utilizá-la, a aplicação pode apresentar dados que são melhor representados por modelos relacionais ou os dados precisam ser acessíveis de fora do Zope. Nestes casos, é possível integrar bancos de dados relacionais ao sistema (LERNER, 2002).

Zope também conta com um conjunto de bibliotecas reutilizáveis, ZTK (*Zope Toolkit*), voltadas para o desenvolvimento de aplicações web, um CMF (*Content Management Framework*) para a construção de aplicações de gerenciamento de conteúdo, um mecanismo de *templates* que interpreta fragmentos ZPT (*Zope Page Template*) e DTML (*Document Template Markup Language*) para gerar páginas HTML (*HyperText Markup Language*) dinamicamente, suporte a internacionalização - I18N (*Internationalization*) e políticas de segurança. Estes componentes são configurados através de arquivos ZCML (*Zope Configuration Mark-up Language*) e acessíveis por meio da ZMI (*Zope Management Interface*), uma interface de gestão web.

#### 4.1.3 Plone

Plone é um sistema de gerenciamento de conteúdo (CMS - *Content Management System*), escrito em Python e que roda sobre o servidor de aplicação Zope. Sua interface é amigável e foi traduzida para mais de 40 idiomas (PLONE, 2012). Focado em usuários com pouco conhecimento técnico, seu principal objetivo é facilitar a criação, edição, publicação e distribuição de conteúdo.

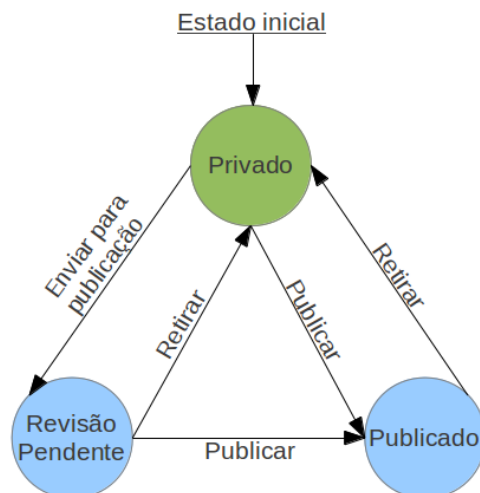
Plone possui um mecanismo completo de registro onde o administrador do site pode adicionar novos usuários, ou o próprio usuário pode efetuar seu cadastro com login e senha, ambos através de interfaces de fácil manuseio. Cada usuário pode criar e modificar suas informações pessoais bem como acessar conteúdos publicados no portal. O controle de acesso a este conteúdo é feito através de um sistema de *workflow*.

Um *workflow* define os possíveis estados de um objeto, suas transições e as permissões de acesso vinculadas a cada um dos estados (DELMONTE et al., 2009). As permissões não são definidas para cada um dos usuários e sim para papéis. O benefício desta abordagem é que um usuário pode ter diversos papéis

atribuídos simultaneamente, assim como pode receber papéis de forma global ou em um contexto específico (ASPELI, 2007).

Um *workflow* pode ser representado através de um grafo dirigido, os nodos são os estados do objeto e as arestas as possíveis transições entre os estados. O workflow padrão para publicação do plone pode ser visto na figura 4.2.

Figura 4.2: *Workflow* padrão para publicação de objetos em Plone.



Para cada estado do *workflow* são definidos os privilégios de acesso dos papéis. Por exemplo, na Figura 4.3, o estado Pendente habilita o acesso às informações do objeto para contribuidores (*Contributor*), editores (*Editor*), administradores (*Manager*), dono do conteúdo (*Owner*), leitores (*Reader*) e revisores (*Reviewer*) através da permissão de acesso a informação de conteúdo (*Access contents information*), mas permite a alteração do conteúdo (*Modify portal content*) apenas aos papéis de administrador e revisor.

Figura 4.3: Permissões para o estado Pendente do *workflow* de publicação.

Workflow State at [/Site/portal\\_workflow/simple\\_publication\\_workflow/states/pending](#)

When objects are in this state they will take on the role to permission mappings defined below. Only the [permissions managed by this workflow](#) are shown.

Permission	Roles
<b>Acquire permission settings?</b>	
	Anonymous Authenticated Contributor Editor Manager Member Owner Reader Reviewer
<input type="checkbox"/> Access contents information	<input type="checkbox"/> <input type="checkbox"/> <input checked="" type="checkbox"/> <input checked="" type="checkbox"/> <input checked="" type="checkbox"/> <input type="checkbox"/> <input checked="" type="checkbox"/> <input checked="" type="checkbox"/> <input checked="" type="checkbox"/>
<input type="checkbox"/> Change portal events	<input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input checked="" type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input checked="" type="checkbox"/>
<input type="checkbox"/> Modify portal content	<input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input checked="" type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input checked="" type="checkbox"/>
<input type="checkbox"/> View	<input type="checkbox"/> <input type="checkbox"/> <input checked="" type="checkbox"/> <input checked="" type="checkbox"/> <input checked="" type="checkbox"/> <input type="checkbox"/> <input checked="" type="checkbox"/> <input checked="" type="checkbox"/> <input checked="" type="checkbox"/>

Com uma comunidade atuante, Plone estende uma série de pacotes e ferramentas que atendem as mais variadas necessidades e demandas, além disso, por ser um

software de código aberto, pode ser facilmente alterado. Novas funcionalidades podem ser incorporadas com a utilização de *Archetypes*, um recurso desenvolvido para a criação de aplicações para Plone, que fornece um método simples para gerar novos conteúdos baseados em definições de esquemas (ARCHETYPES, 2013).

## 4.2 *Framework*

O *framework CoPPLA* utiliza *Archetypes* para estender os tipos de conteúdo já existentes do Plone, podendo ser instalado em qualquer portal com esta tecnologia. O produto conta ainda com um sistema de autenticação e um conjunto de *workflows* que define as permissões de acesso sobre os papéis dos membros da comunidade.

### 4.2.1 Tipos de conteúdo

Plone disponibiliza a criação de pastas, arquivos, imagens, links, eventos, páginas web e notícias, entretanto, para ter maior controle sobre os tipos de conteúdo disponíveis no ambiente de comunidades de prática virtuais, o *framework* estende estes tipos nativos, vinculando *templates* e *workflows* personalizados. A estilização dos *templates* visa proporcionar aos participantes uma interface que facilite a interação com o ambiente e o *workflow* se encarrega de definir os papéis e seus níveis de interação.

Uma comunidade de prática e seu conteúdo podem apresentar os estados público, restrito, e inativo. Os usuários podem receber os papéis de autenticado, participante e moderador. Uma comunidade ou conteúdo quando públicos podem ser acessados por qualquer usuário, itens restritos são visíveis apenas aos participantes da comunidade e itens inativos visíveis apenas aos moderadores e ao próprio dono.

### 4.2.2 Ferramentas implementadas

O *framework CoPPLA* disponibiliza espaços para a criação de conteúdo e interação entre os participantes através de:

**Calendário** - Onde eventos podem ser criados por participantes e moderadores;

**Acervo** - Armazena o conteúdo geral da comunidade, podem ser inseridos arquivos, imagens, links, páginas e pastas. Utilizado para publicação de produções coletivas;

**Portfólio** - Utilizado para publicação de produções individuais, disponibiliza a criação de arquivos, imagens e páginas;

**Tarefas** - Espaço para entrega de tarefas (em contextos educacionais);

**Fórum de discussão** - Nas comunidades discussões podem surgir em comentários dos itens publicados ou através de um ambiente de conversas que

utiliza o produto Ploneboard<sup>3</sup>;

**Notificações** - Utilizam-se duas formas de notificação nas comunidades. A primeira delas é um mecanismo que permite aos moderadores o envio direto de mensagens aos participantes e a segunda é através de um resumo diário das atividades da comunidade enviado por e-mail;

**Perfil dos usuários** - Exibe informações referentes ao usuário bem como seu conteúdo compartilhado;

**Lista de participantes** - Exibe os participantes e moderadores da comunidade, sendo possível a pesquisa e o acesso ao perfil de cada um;

**Nível de atividade** - Ferramenta desenvolvida para verificar o nível de interação nas comunidade, tanto coletivo quanto individual;

**Histórico** - Armazena todo o conteúdo produzido pelos participantes, pode-se pesquisar por conteúdos específicos utilizando filtros como nome, data de criação, quem publicou, temas relacionados, entre outros.

**Convite** - Ferramenta para envio de convites por e-mail, onde é possível anexar mensagens personalizadas.

**Perfil da comunidade** - Através da página inicial de uma comunidade é possível encontrar seu nome, imagem, descrição, lista de participantes e histórico.

#### 4.2.3 Temas, usuários e seus relacionamentos

Temas são termos de categorização do conhecimento, relacionados a cada conteúdo compartilhado, que representam o domínio da comunidade. No Plone os temas são palavras chaves representadas como entradas do índice *Subject* do ZCatalog. Um objeto pode ser marcado com um ou mais temas e, a partir destas palavras chave, é possível pesquisar por conteúdos relacionados. Os temas proporcionam maior flexibilidade na pesquisa e categorização de conteúdo do que índices, uma vez que um item pode ser marcado com mais de um tema simultaneamente (ZOPE, 2013b).

Em um site desenvolvido sobre PZP os usuários são objetos armazenados em uma pasta, no nível do Zope, chamada *acl.users*. Um usuário Zope possui nome, senha e papéis vinculados a ele. Para suportar propriedades adicionais são utilizados produtos externos para gerenciamento de usuários (ZOPE, 2013c). A autenticação e as informações de permissões são gerenciadas pelo *PAS (Pluggable Authentication Service)*, um sistema que implementa várias interfaces de autenticação, por exemplo LDAP (*Lightweight Directory Access Protocol*) e OpenID. No nível do Plone, o acesso as informações do usuário é realizado através da ferramenta *portal.membership* que as fornece através de metadados (PLONE, 2013b).

De forma nativa Plone não oferece ferramentas para o relacionamento direto

---

<sup>3</sup><http://plone.org/products/ploneboard>



entre temas e usuários, uma associação só é possível através dos itens publicados, que possuem um criador e podem ser marcados com temas, relacionando assim, os temas dos objetos a seus criadores. No entanto, essa abordagem torna difícil a execução de tarefas como a consulta por todos os temas relacionados a um usuário ou vice-versa. Também não abrange todos os interesses que um usuário pode apresentar, visto que o mesmo pode interessar-se por temas que não fazem parte de suas publicações.

### 4.3 Conclusões sobre o capítulo

Este capítulo apresentou uma visão geral sobre a plataforma de comunidades de prática a ser utilizada neste trabalho. Foram evidenciadas as principais tecnologias envolvidas na construção e compartilhamento do conhecimento neste framework. A ênfase nas ferramentas e na estrutura do conteúdo armazenado serve de base para a delimitação do escopo a ser utilizado na proposta de solução. Serão utilizadas apenas informações de relacionamentos entre participantes e os temas de interesse dos usuários em comunidades de prática. Desta forma, este trabalho não utilizará a estrutura completa de informação para concentrar-se apenas nos conteúdos diretamente relacionados aos seus membros criadores. Recomendações de conteúdo ou relações entre perfis, interesses e conteúdos podem ser abordados por sistemas mais robustos e complexos de recomendação, que estão fora do escopo deste trabalho.

Finalmente, a partir do estudo realizado sobre as ferramentas para instrumentalização de comunidades de prática e as já implementadas no *framework CoPPLA*, identificou-se a necessidade do desenvolvimento de mecanismos de relacionamento entre os interesses e conteúdos do usuário, assim como de relacionamento entre os participantes, promovendo uma maior interação entre os usuários, o que pode contribuir para o aumento de sua confiança e participação na construção do conhecimento.

## 5 SISTEMA DE FOLLOWING

O principal objetivo deste trabalho é disponibilizar um sistema de *following* que proporcione aos usuários novas formas de interação, facilitando o processo de aprendizagem, além da possibilidade de declarar suas preferências, de forma explícita, em comunidades de prática.

O sistema de *following* será integrado a um ambiente já em produção, por isso toda sua estrutura, criação das tabelas, mapeamento objeto relacional, regras de negócio e apresentação foram desenvolvidos em um novo produto Python/Zope/Plone que não cria dependências no *framework CoPPLA*. Deste modo, a instalação do sistema de *following* é opcional junto à plataforma.

Este capítulo será organizado em três seções. A primeira seção define os requisitos do sistema através de casos de uso e suas descrições. Para cada caso de uso também serão apresentadas as interfaces de usuário desenvolvidas. A segunda seção descreve a arquitetura do sistema pra atender a estas especificações. A terceira seção apresenta as conclusões do capítulo.

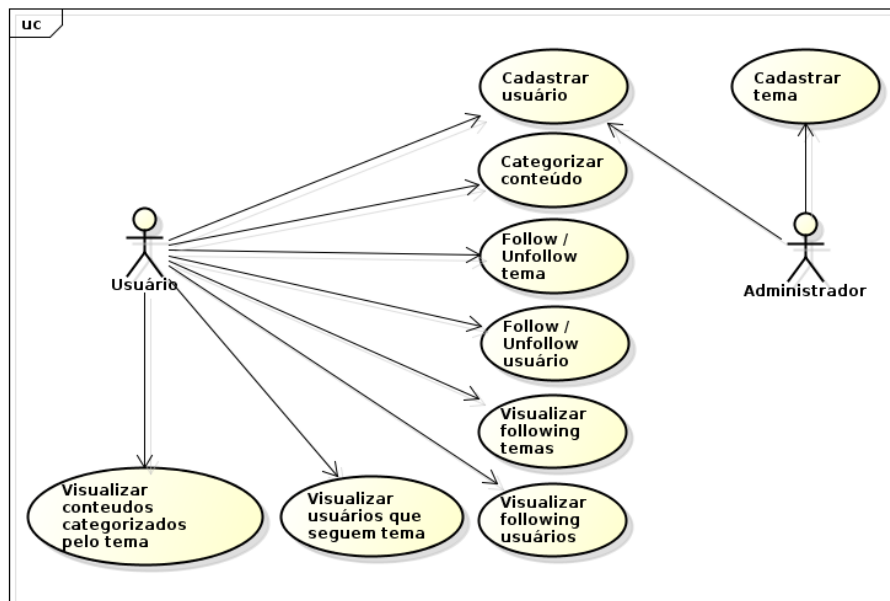
### 5.1 Requisitos

Os requisitos para o sistema de *following* foram definidos através de casos de uso modelados levando em consideração dois papéis: Administrador e Usuário. Administrador é o indivíduo que possui permissão de administrador do Zope ou administrador do portal Plone. Usuário é qualquer indivíduo que não possui alguma das permissões citadas na condição anterior. Sendo assim, tanto moderadores quanto participantes das comunidade de prática são considerados usuários. O administrador também tem acesso a todas as funcionalidades inerentes ao papel do usuário, não havendo representação de suas associações para uma melhor visualização dos casos de uso (Figura 5.1).

### 5.1.1 Casos de uso

Os casos de uso apresentados na Figura 5.1 contemplam todas as funcionalidades desenvolvidas no sistema de *following*. Cadastrar usuários, Cadastrar temas e Categorizar conteúdo são funcionalidades previstas e implementadas, de forma nativa, na solução tecnológica Python/Zope/Plone. Elas são apresentadas no escopo deste trabalho apenas com o objetivo de proporcionar um melhor entendimento do sistema como um todo.

Figura 5.1: Casos de uso do sistema de *following*.



#### 5.1.1.1 Cadastrar usuário

Para participar de uma comunidade de prática e posteriormente interagir através do sistema de *following* é necessário que o usuário esteja registrado no portal e autenticado. Alguns portais Plone permitem o auto-registro (Tabela 5.1), enquanto que outros só permitem o registro de novos usuários através do administrador do site (Tabela 5.2). É possível alterar a forma de registro em um site Plone por meio de suas configurações. Será apresentado um caso de uso descritivo para cada situação.

Tabela 5.1: Cadastrar usuário - Caso de uso 1

<b>Atores:</b>	Administrador do site
<b>Descrição:</b>	Administrador do site acessa área de gerência de usuário e grupos e insere novo usuário. Usuário acessa o portal e solicita uma redefinição de senha.
<b>Pré-condição:</b>	Estar autenticado no site Plone e possuir permissão de administrador.
<b>Pós-condição:</b>	Novo usuário cadastrado no portal.
<b>Fluxo Principal:</b>	<ol style="list-style-type: none"> <li>1. Administrador acessa área de gerenciamento de usuários e grupos</li> <li>2. Administrador inicia processo de inserção de usuário</li> <li>3. Administrador insere informações do novo usuário</li> <li>4. Administrador finaliza inserção</li> <li>5. Usuário acessa o portal e inicia processo de redefinição de senha</li> <li>6. Usuário recebe email de redefinição de senha</li> <li>7. Usuário escolhe nova senha e confirma a operação</li> </ol>
<b>Fluxos Alternativos:</b>	-

Tabela 5.2: Cadastrar usuário - Caso de uso 2

<b>Atores:</b>	Usuário
<b>Descrição:</b>	Usuário acessa o portal e inicia o processo de cadastro. Campos de informações pessoais, login e senha são apresentados para preenchimento pelo usuário. As informações obrigatórias devem ser verificadas, e caso não estejam preenchidas, as devidas mensagens devem ser exibidas.
<b>Pré-condição:</b>	Auto-registro deve estar habilitado nas configurações do portal Plone.
<b>Pós-condição:</b>	Novo usuário cadastrado no portal.
<b>Fluxo Principal:</b>	<ol style="list-style-type: none"> <li>1. Usuário acessa o portal</li> <li>2. Usuário acessa área de registro</li> <li>3. Usuário preenche informações necessárias, incluindo login e senha</li> <li>4. Usuário finaliza registro</li> </ol>
<b>Fluxos Alternativos:</b>	<ol style="list-style-type: none"> <li>3. Usuário preenche informações necessárias, incluindo login e senha <ol style="list-style-type: none"> <li>3.1. Sistema informa a falta de alguma informação obrigatória</li> <li>3.2. Usuário preenche informações solicitadas</li> </ol> </li> <li>4. Usuário finaliza registro</li> </ol>

### 5.1.1.2 Cadastrar tema

Com o propósito de mitigar inconsistências como sinonímia e variações linguísticas, o cadastro de temas é restrito ao administrador. A inserção livre de temas pode criar uma lista de termos semanticamente duplicados, reduzindo sua relevância. Por exemplo, pode-se utilizar os termos “banco de dado”, “banco de dados”, “database”, “db”. Todos termos que se referem a um mesmo assunto. A Tabela 5.3 apresenta o caso de uso para cadastro de temas, disponível durante a criação e edição de conteúdo.

Tabela 5.3: Cadastrar tema - Caso de uso

<b>Atores:</b>	Administrador do site
<b>Descrição:</b>	Todos os temas disponíveis para a categorização de conteúdo são cadastrados pelo administrador durante a inserção ou edição de conteúdo.
<b>Pré-condição:</b>	Estar autenticado no site e possuir permissão de administrador.
<b>Pós-condição:</b>	Novo tema cadastrado e disponível para categorização de conteúdo pelos usuários.
<b>Fluxo Principal:</b>	<ol style="list-style-type: none"> <li>1. Administrador inicia processo de criação ou edição de algum conteúdo</li> <li>2. Administrador insere um ou mais temas à lista de categorias do conteúdo</li> <li>3. Administrador finaliza inserção ou edição</li> </ol>
<b>Fluxos Alternativos:</b>	-

#### 5.1.1.3 Categorizar conteúdo

A categorização de conteúdo (Tabela 5.4) acontece durante a criação ou edição dos objetos. Usuários apenas podem escolher entre os temas previamente cadastrados. Todavia, a inserção de novos temas está disponível para administradores.

No contexto das comunidades de prática, moderadores possuem permissão para edição dos conteúdos inseridos em comunidades por eles gerenciadas, desta forma, quem é capaz de categorizar um conteúdo é o dono do objeto ou o moderador da comunidade.

Tabela 5.4: Categorizar conteúdo - Caso de uso

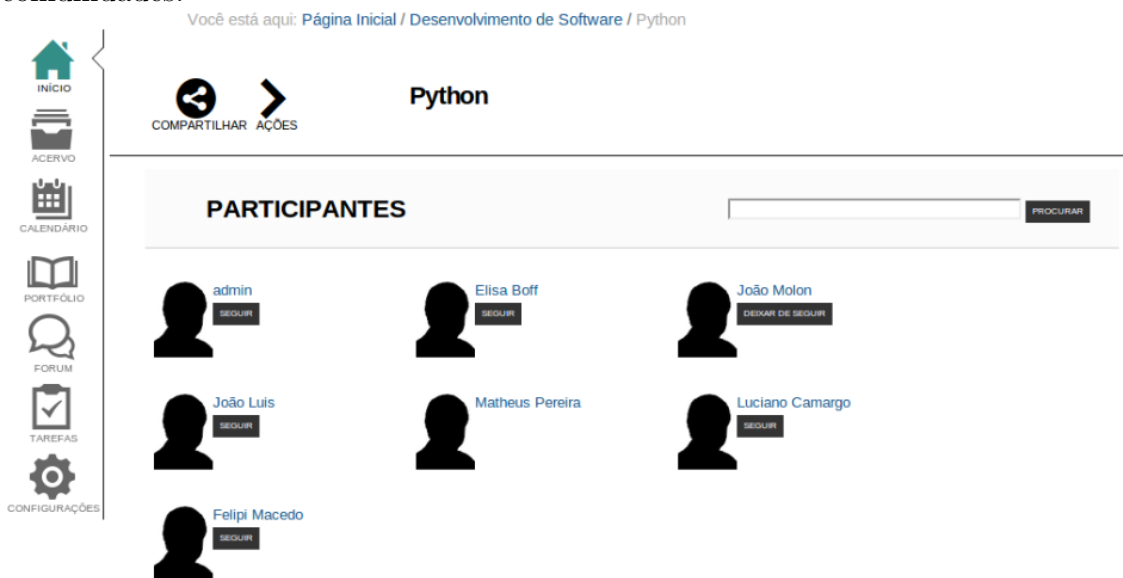
<b>Atores:</b>	Usuário
<b>Descrição:</b>	Ao inserir ou editar um conteúdo o usuário pode categorizá-lo utilizando temas previamente cadastrados pelo administrador.
<b>Pré-condição:</b>	Usuário deve estar autenticado e deve participar de uma comunidade. O conteúdo deve ser de autoria própria ou o usuário deve ser moderador da comunidade. Pelo menos um tema deve estar disponível para seleção.
<b>Pós-condição:</b>	Conteúdo categorizado com temas escolhidos pelo usuário.
<b>Fluxo Principal:</b>	<ol style="list-style-type: none"> <li>1. Usuário inicia o processo de inserção ou edição de um conteúdo</li> <li>2. Usuário categoriza o conteúdo com os temas disponíveis</li> <li>3. Usuário finaliza o processo de inserção ou edição do conteúdo</li> </ol>
<b>Fluxos Alternativos:</b>	-

#### 5.1.1.4 Follow / Unfollow de usuário

A funcionalidade de seguir usuários (Tabela 5.5) deve estar disponível na lista de participantes de uma comunidade (Figura 5.2) e na página de perfil dos usuários (Figura 5.3). Se um usuário não está autenticado nenhuma ação de *following* deve estar disponível.

Tabela 5.5: *Follow / Unfollow* de usuário - Caso de uso

<b>Atores:</b>	Usuário
<b>Descrição:</b>	Usuários deve ser capaz de seguir ou deixar de seguir outros usuários.
<b>Pré-condição:</b>	Usuário deve estar autenticado.
<b>Pós-condição:</b>	Usuário é inserido / removido da lista de <i>following</i> do usuário autenticado.
<b>Fluxo Principal:</b>	<ol style="list-style-type: none"> <li>1. Usuário acessa lista de participantes da comunidade de prática</li> <li>2. Usuário segue participante</li> </ol>
<b>Fluxos Alternativos:</b>	<ol style="list-style-type: none"> <li>1. Usuário acessa perfil de algum participante da comunidade de prática</li> <li>2. Usuário segue participante</li> </ol>

Figura 5.2: Ação de *Follow / Unfollow* disponível na lista de participantes das comunidades.Figura 5.3: Ação de *Follow / Unfollow* disponível no perfil do usuário.

#### 5.1.1.5 *Follow / Unfollow* de tema

A Figura 5.4 apresenta a área de gerenciamento de temas. O usuário pode pesquisar por temas disponíveis no portal e segui-los ou deixar de seguir temas que fazem parte da sua lista de *following*. Apenas usuários autenticados podem seguir

temas. A Tabela 5.6 apresenta o caso de uso expandido para a funcionalidade de *following* de temas.

Tabela 5.6: *Follow / Unfollow* de tema - Caso de uso

<b>Atores:</b>	Usuário
<b>Descrição:</b>	Em uma área de gerenciamento de temas, usuário consulta e segue temas de seu interesse ou deixa de seguir temas que já estão presentes em sua lista de <i>following</i> .
<b>Pré-condição:</b>	Usuário deve estar autenticado, deve existir um ou mais temas cadastrados pelo administrador.
<b>Pós-condição:</b>	Tema é inserido / removido da lista de <i>following</i> do usuário autenticado.
<b>Fluxo Principal:</b>	<ol style="list-style-type: none"> <li>1. Usuário acessa área de gerenciamento de temas</li> <li>2. Usuário pesquisa por tema de seu interesse</li> <li>3. Usuário encontra e segue determinado tema</li> </ol>
<b>Fluxos Alternativos:</b>	<ol style="list-style-type: none"> <li>1. Usuário acessa área de gerenciamento de temas</li> <li>2. Usuário deixa de seguir tema presente em sua lista de <i>following</i></li> </ol>

Figura 5.4: Gerenciamento do *following* de temas.



#### 5.1.1.6 Visualizar *following* de usuário

A Tabela 5.7 descreve a funcionalidade de visualização do *following* de usuários, que deve ser acessível de resumida (Figura 5.5) e, a partir desta visualização, deve ser possível acessar a lista completa de *following*. Na visualização completa do *following* deve ser possível seguir e deixar de seguir os usuários (Figura 5.6).

Tabela 5.7: Visualizar *following* de usuário - Caso de uso

<b>Atores:</b>	Usuário
<b>Descrição:</b>	O usuário deve ser capaz de acessar sua lista de seguidores e seguidos.
<b>Pré-condição:</b>	Usuário deve estar autenticado e possuir pelo menos uma ligação de <i>following</i> .
<b>Pós-condição:</b>	-
<b>Fluxo Principal:</b>	<ol style="list-style-type: none"> <li>1. Usuário acessa perfil para visualização do <i>portlet</i> de <i>following</i> de usuários</li> <li>2. A partir do <i>portlet</i> de <i>following</i> usuário acessa a visualização completa do <i>following</i> de usuário</li> </ol>
<b>Fluxos Alternativos:</b>	-

Figura 5.5: *Portlet* para visualização do *following* de temas.Figura 5.6: Visualização do *following* de usuários.



### 5.1.1.7 Visualizar following de tema

Duas formas de visualização são apresentadas na Tabela 5.8. Deve ser possível visualizar a lista de *following* de temas de maneira resumida (Figura 5.7) e, a partir desta visualização, acessar a lista completa de *following*. Na visualização completa do *following* deve ser possível seguir e deixar de seguir temas (Figura 5.4).

Tabela 5.8: Visualizar *following* de tema - Caso de uso

<b>Atores:</b>	Usuário
<b>Descrição:</b>	Usuário deve ser capaz de visualizar sua lista de temas seguidos.
<b>Pré-condição:</b>	Usuário deve estar autenticado e seguir pelo menos um tema.
<b>Pós-condição:</b>	-
<b>Fluxo Principal:</b>	1. Usuário acessa perfil para visualização do <i>portlet</i> de <i>following</i> de temas
<b>Fluxos Alternativos:</b>	1. A partir do <i>portlet</i> de <i>following</i> usuário acessa a visualização completa do <i>following</i> de temas.

Figura 5.7: *Portlet* para visualização do *following* de temas.



### 5.1.1.8 Visualizar conteúdos categorizados pelo tema

O principal objetivo da utilização de temas é organizar conteúdos em categorias, portanto, visualizar os conteúdos marcados por determinado tema é essencial. O usuário, ao acessar determinado tema, deve ser capaz de encontrar todo o conteúdo categorizado com aquela palavra chave (Tabela 5.9 e Figura 5.8).

Tabela 5.9: Visualizar conteúdos categorizados pelo tema - Caso de uso

<b>Atores:</b>	Usuário
<b>Descrição:</b>	Ao acessar a página de determinado tema o usuário pode encontrar todo o conteúdo marcado com aquela palavra chave.
<b>Pré-condição:</b>	-
<b>Pós-condição:</b>	-
<b>Fluxo Principal:</b>	<ol style="list-style-type: none"> <li>1. Usuário acessa página do tema a partir do <i>portlet</i> de following de temas</li> <li>2. Usuário visualiza conteúdo marcado com palavra chave</li> </ol>
<b>Fluxos Alternativos:</b>	<ol style="list-style-type: none"> <li>1. Usuário acessa determinado conteúdo das comunidades</li> <li>2. Usuário acessa página de algum tema marcado naquele conteúdo</li> <li>3. Usuário visualiza conteúdo marcado com palavra chave</li> </ol>

Figura 5.8: Conteúdos categorizados com o tema "Escola".

The screenshot displays a web application interface for a community of practice. The top navigation bar includes the site logo, user profile icons, and a search bar. The main content area is titled 'Conteúdos sobre "Escola"' and lists three items:

- Alexandre Ribeiro** posted on 27/08/2013 17:06: 'tst 2' (Post).
- André da Silva** posted on 23/08/2013 20:02: 'Página compartilhada do André' (Página compartilhada do André).
- Enio Gonçalves De Moraes** posted on 31/03/2013 11:35: 'Práticas Integrativas nos Centros de Atenção Psicossocial' (Espaço destinado a integração de profissionais envolvidos com as práticas realizadas nos CAPS, que promovam a autonomia e a melhora dos pacientes de CAPS, com o envolvimento de toda a rede de atenção Psicossocial).

### 5.1.1.9 Visualizar usuários que seguem tema

Acessando determinado tema deve ser possível visualizar todos os outros usuários que seguem este tema (Tabela 5.10 e Figura 5.9). Esta é uma funcionalidade relevante para as comunidades de prática pois incentiva a interação entre os usuários que possuem interesses comuns.

Tabela 5.10: Visualizar usuários que seguem tema - Caso de uso

<b>Atores:</b>	Usuário
<b>Descrição:</b>	Ao acessar a página de determinado tema o usuário pode encontrar todos os usuários que seguem aquela palavra chave.
<b>Pré-condição:</b>	-
<b>Pós-condição:</b>	-
<b>Fluxo Principal:</b>	<ol style="list-style-type: none"> <li>1. Usuário acessa página do tema a partir do <i>portlet</i> de following de temas</li> <li>2. Usuário visualiza a lista de todos os usuários que seguem aquela palavra chave</li> </ol>
<b>Fluxos Alternativos:</b>	<ol style="list-style-type: none"> <li>1. Usuário acessa determinado conteúdo das comunidades</li> <li>2. Usuário acessa página de algum tema marcado naquele conteúdo</li> <li>3. Usuário visualiza a lista de todos os usuários que seguem aquela palavra chave</li> </ol>

Figura 5.9: Lista de usuários que seguem o tema "Escola".



## 5.2 Arquitetura

O sistema de *following* é dividido em camadas. A arquitetura em camadas é comumente utilizada no desenvolvimento de software. Camadas oferecem abstração de implementação e modularidade, as dependências são minimizadas, a manutenção facilitada e o reuso dos componentes das aplicações é maximizado (FOWLER, 2002).

Por ser um ambiente Web, o desenvolvimento do *following* é dividido em três camadas: camada de apresentação, camada de negócio e camada de persistência. O acesso aos dados é separado do restante da aplicação, permitindo assim a comunicação de diferentes aplicativos à mesma base de dados através de interfaces de comunicação. As regras de negócio e processamento ficam do lado do servidor e a apresentação do lado do cliente. Cada módulo pode ser mantido e alterado sem comprometer os demais, desde que o mecanismo de comunicação entre eles não seja alterado.

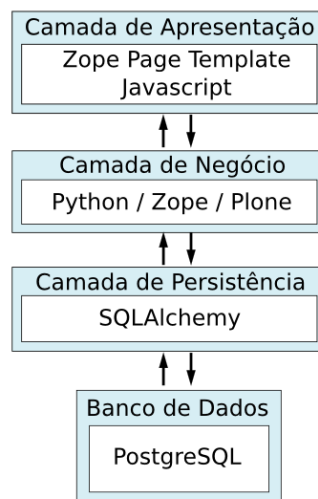
A camada de apresentação tem por objetivo implementar as funcionalidades de interação com o usuário, isolando a camada de negócio. Esta é uma característica importante que faz com que alterações de interface não afetem a camada de negócio. Esta camada é especialmente importante para o trabalho pois as funcionalidades do *following* serão agregadas a um produto externo, as

comunidades de prática, e não devem comprometer seu funcionamento. A camada de negócio é a camada que agrupa as funcionalidades da aplicação, a lógica de funcionamento, sem se preocupar com interfaces de usuário e persistência de dados. A camada de persistência agrupa as funcionalidades que tem como finalidade a criação, remoção, alteração e recuperação de dados persistentes. A camada de persistência não é o banco de dados mas sim o *front-end* para acesso a ele.

As camadas comunicam-se apenas com suas camadas adjacentes. Sendo assim, a camada de apresentação tem comunicação apenas com a camada de negócio, a camada de negócio comunica-se com a camada de apresentação e persistência, a camada de persistência tem comunicação com a camada de negócio e com o banco de dados. O banco de dados comunica-se apenas com a camada de persistência. Vale ressaltar que o banco de dados não é considerado uma camada da aplicação. Um banco de dados é uma aplicação completa, independente, que implementa interfaces de comunicação para disponibilizar acesso às suas funcionalidades.

A modelagem do banco de dados, as ferramentas e tecnologias utilizadas na implementação do *following* serão apresentadas em suas respectivas camadas conforme pode-se visualizar na Figura 5.10.

Figura 5.10: Modelo de aplicação em três camadas para o sistema de *following*.



### 5.2.1 Banco de dados

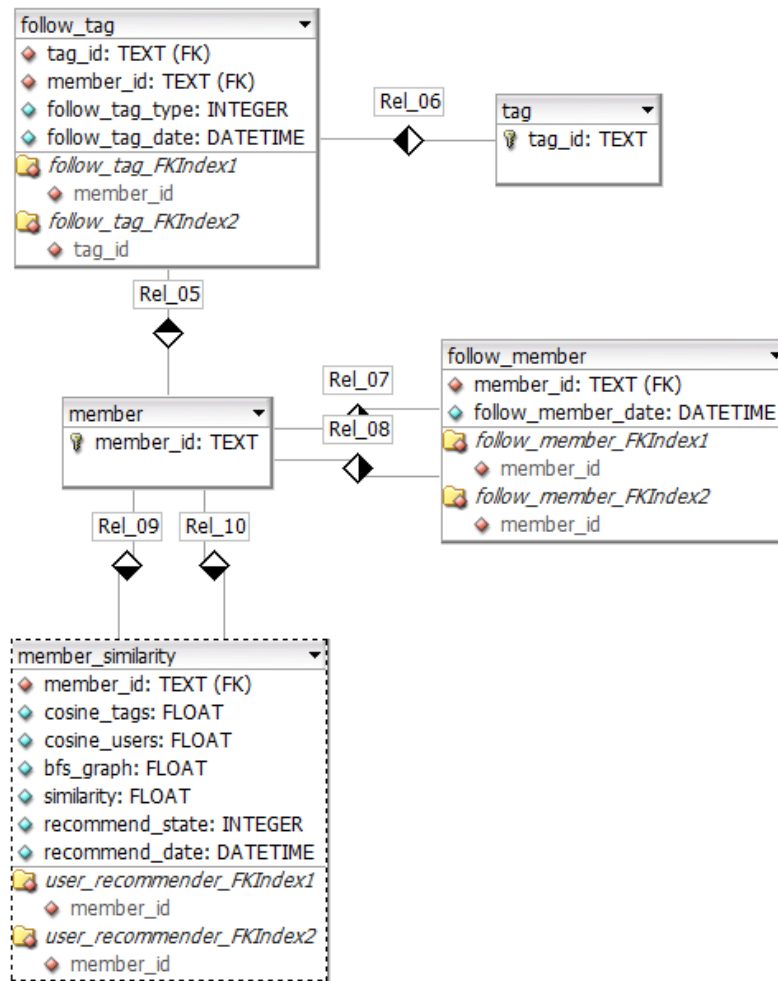
As relações entre usuários e temas foram armazenadas utilizando-se o banco de dados relacional PostgreSQL. Este foi escolhido por ser um banco de dados de código aberto, multiplataforma e com licença livre, assim como todas as ferramentas já utilizadas na plataforma. Além disso, PostgreSQL é um banco de dados objeto-relacional, desenvolvido há mais de 15 anos, que possui uma interface nativa para Python e ampla documentação (POSTGRESQL, 2013).

Conforme apresentado na Figura 5.11, foram representados no banco de dados relacional os usuários (*member*), os temas (*tag*), as relações entre usuários (*follow\_member*) e as relações entre usuários e temas (*follow\_tag*). A similaridade entre os usuários (*member\_similarity*), utilizada para validação do *following*, também será armazenada no banco de dados PostgreSQL.

As tabelas *member* e *tag* contam apenas com um atributo identificador, do tipo *string*, utilizado para mapear os usuários e os temas no Plone. A tabela *follow\_member* possui os atributos identificadores dos usuários relacionados como chaves estrangeiras, que juntas são a chave primária da tabela, assim como a data em que ocorreu o *following*. A tabela *follow\_tag* apresenta os atributos identificadores do usuário e tema relacionados, ambos chave estrangeira, que juntos formam a chave primária da tabela, a data em que ocorreu o *following* e o tipo de relacionamento. O atributo tipo de relacionamento não será tratado no escopo deste trabalho, mas trata-se da forma de relacionamento que o usuário atribui a determinado tema, por exemplo, interesse ou experiência.

A tabela *member\_similarity* possui os atributos identificadores dos usuários (*member\_id*), cosseno de temas (*cossine\_tags*), cosseno de usuários (*cossine\_users*), busca em amplitude no grafo (*bfs\_graph*), similaridade (*similarity*), estado da recomendação (*recommend\_state*) e data da recomendação (*recommend\_date*). Os atributos da tabela *member\_similarity* são discutidos no capítulo 6.

Figura 5.11: Modelo da estrutura de banco de dados para o sistema de *following*.

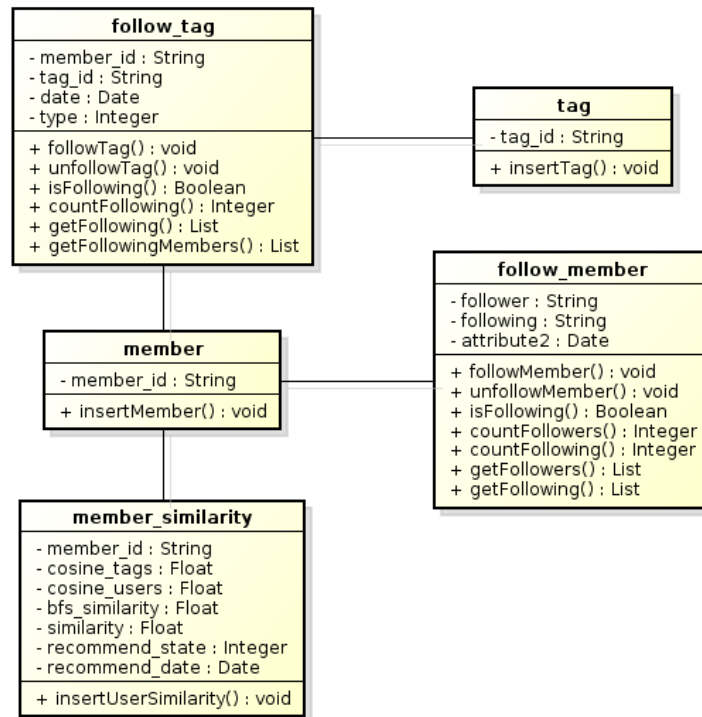


### 5.2.2 Camada de persistência

A conexão entre a aplicação e o PostgreSQL acontece através do SQLAlchemy, um conjunto de ferramentas e ORM (*Object Relational Mapper*) que faz o mapeamento das definições de classes Python em tabelas de bancos relacionais e dos objetos em entradas de tabelas (SQLALCHEMY, 2013).

Cada uma das tabelas apresentadas na Figura 5.11 possui uma classe Python associada. A manipulação dos dados, em nível de aplicação, é realizada através de instâncias dessas classes. As definições de classes e métodos associados aos objetos são apresentadas na Figura 5.12 e juntas compõem a camada de persistência do sistema de *following*.

Figura 5.12: Diagrama de classe para camada de persistência.



### 5.2.3 Camada de negócio

A camada de negócio compreende toda a lógica de funcionamento do sistema de *following*, comunicação com Python/Zope/Plone, acesso a informações do ZODB e PostgreSQL, controle de acesso e segurança, gerenciamento dos usuários e comunicação com *templates* ZPT.

O Trecho de Código 5.1 faz parte da camada de apresentação e é um fragmento *Zope Page Template* que interage com objetos Python para gerar código HTML. O Trecho de Código 5.2 faz parte da camada de persistência sendo responsável pela inserção de um novo *following* de usuário no PostgreSQL através do SQLAlchemy.

Trecho de Código 5.1: *Zope Page Template* para seguir usuário.

```

1   <div
2       tal:condition="python: member_id and view.is_following(member_id) == '
        not_following'">
3       <form method="post" >
4           <input type="hidden" name="member_id" tal:attributes="value
                member_id" />
5           <input type="hidden" name="transition" value="follow" />
6           <input type="submit" name="submit" id="submit" value="Seguir"
7               tal:attributes="name python: 'follow_%s'%(member_id) ">
8       </form>
9   </div>

```

Trecho de Código 5.2: Código SQLAlchemy para seguir usuário.

```

1     def followMember(follower_id, following_id):
2         try:
3             if not session.query(FollowMember).get((follower_id, following_id)):
4                 insertMember(follower_id)
5                 insertMember(following_id)
6                 date = datetime.now()
7                 follow = FollowMember(follower_id, following_id, date)
8                 session.add(follow)
9                 transaction.commit()
10            return True
11        except SQLAlchemyError:
12            return False

```

O Trecho de Código 5.3 demonstra a implementação da funcionalidade de seguir usuários. A função *follow\_member*, escrita em Python, utiliza recursos da tecnologia Python/Zope/Plone, por exemplo, as funções *getToolByName* e *getAuthenticatedMember*, aplica regras de negócio e faz a comunicação entre as camadas de apresentação e persistência.

Através de um método *POST* (Linhas 2, 3 do Trecho de Código 5.3 e Trecho de Código 5.1) a aplicação recebe as informações para o *following* da camada de apresentação e envia as alterações para a camada de persistência utilizando a função *followMember* (Linha 6 do Trecho de Código 5.3). O mesmo mecanismo é utilizado para o *following* de temas.

Trecho de Código 5.3: Código Python para seguir usuário.

```

1     def follow_member(self, user_id):
2         form = self.request.form
3         if 'follow' in form.values() and user_id in form.values():
4             portal_membership = getToolByName(self.context, 'portal_membership')
5             authenticated_member = portal_membership.getAuthenticatedMember()
6             followMember(authenticated_member.getId(), user_id)

```

A exibição das informações para o usuário acontece no sentido inverso. Para mostrar a opção de seguir ou deixar de seguir um usuário, por exemplo, o código Python faz as consultas no banco de dados através do SQLAlchemy (Trecho de Código 5.4 Linha 4 e Trecho de Código 5.5), interage com Python/Zope/Plone (Trecho de Código 5.4 Linhas 2, 3) e envia as informações para o ZPT (Trecho de Código 5.4 Linha 4). O sistema de *templates* do Zope interpreta as informações recebidas e gera o código HTML (Trecho de Código 5.1 Linhas 1 e 2).

Trecho de Código 5.4: Código Python para seguir usuário.

```

1     def is_following(self, user_id):
2         portal_membership = getToolByName(self.context, 'portal_membership')
3         authenticated_member = portal_membership.getAuthenticatedMember()
4         return isFollowing(authenticated_member.getId(), user_id)

```



Trecho de Código 5.5: Código Python para seguir usuário.

```

1 def isFollowing(follower_id, following_id):
2     try:
3         follow = session.query(FollowMember).get((follower_id, following_id))
4         if follow:
5             return 'following'
6         return 'not_following'
7     except SQLAlchemyError:
8         return False

```

#### 5.2.4 Camada de apresentação

As interfaces de usuário desenvolvidas para o sistema de *following*, conforme exibidas na seção 5.1, utilizam as tecnologias *Zope Page Template* e JavaScript.

*Zope Page Template* é o mecanismo utilizado para a comunicação com Python. Através dele é possível invocar funções Python e tratar seus retornos. ZPT tem a capacidade de iterar por listas, verificar condições, inserir e remover elementos, manipular atributos utilizando padrões TAL (*Template Attribute Language*) e expressões TALES (*Template Attribute Language Expression Syntax*).

JavaScript é a linguagem de programação Web responsável por injetar o sistema de *following* nas interfaces da plataforma *CoPPLA* utilizando requisições assíncronas de informações com AJAX (*Asynchronous Javascript and XML*). As requisições assíncronas também tornam o sistema de *following* mais interativo, não sendo necessário recarregar toda a página HTML a cada ação de *following*.

### 5.3 Conclusões sobre o capítulo

Ao longo deste capítulo foram apresentadas as funcionalidades, interfaces de usuário e arquitetura do sistema de *following* desenvolvido para a plataforma *CoPPLA*. Foram gerados artefatos que possibilitam o entendimento da solução e apontadas as tecnologias utilizadas.

A possibilidade de relacionar usuários e temas, disponível através do sistema de *following* nas comunidade de prática, constrói uma estrutura de interesses distribuídos. Esta estrutura contém informações relevantes da maneira como os usuários interagem nas comunidades, além de representar suas preferências declaradas explicitamente. O próximo capítulo trata da extração de atributos do sistema de *following* que podem ser utilizados para a validação deste trabalho.

## 6 EXTRAINDO ATRIBUTOS DO SISTEMA DE FOLLOWING

O propósito do sistema de *following* é estreitar a relação entre os participantes das comunidades motivando sua colaboração no ambiente virtual. Entretanto, quando um usuário segue temas ou participantes é porque tem algum interesse nos mesmos. Estas informações declaradas pelos usuários podem servir de fontes para os sistemas de recomendação.

Ao seguir temas e usuários, o indivíduo relaciona seus interesses aos de outros membros desta rede de relacionamentos. De forma indireta, ao seguir outros usuários, o indivíduo coloca este conteúdo a disposição de forma implícita. O sistema de *following* pretende capturar todas estas relações, gerenciando os interesses que estão distribuídos em uma vasta rede de relacionamentos, sem a necessidade de uma formalização substancial de conteúdos avaliados por uma grande quantidade de usuários, requisitados para um sistema de recomendação, por exemplo.

Isto posto, a validação do sistema de *following* acontece através da extração de atributos relevantes que serão utilizados em cenários de recomendação de usuários e temas nas comunidades de prática.

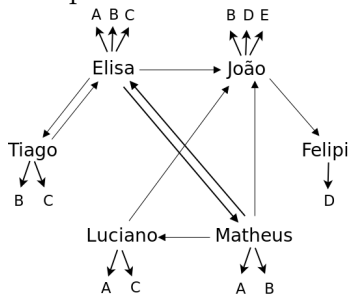
### 6.1 Representação do *following*

Redes sociais são muito dinâmicas, conexões surgem e desaparecem rapidamente. Ao seguir temas e usuários, os participantes das comunidades de prática desenvolvem uma base de interesses que pode ser representada através de um grafo dirigido. Este grafo de relações é construído de maneira colaborativa durante a navegação dos usuários que, ao expressarem suas preferências seguindo outros usuários e temas, geram novas conexões.

Usuários e temas são representados através de vértices e seus relacionamentos através de arestas. Usuários podem seguir e ser seguidos por outros usuários e esta diferença é representada pela direção da aresta no grafo dirigido. Seguir temas é

uma relação no sentido do usuário para o tema, não havendo relação no sentido inverso. A Figura 6.1 demonstra um possível grafo gerado pelo sistema de *following*.

Figura 6.1: Exemplo de uma estrutura de *following*.



Considerando esta representação é possível extrair características relevantes da rede de relacionamentos das comunidades de prática. Podemos, por exemplo, verificar quais usuários definem maior importância e interesse em relação aos outros usuários, calcular o quão distantes os participantes estão uns dos outros, encontrar usuários que possuem maior relevância na rede ou que possuem *expertise* em determinado assunto, correlacionar temas a partir dos interesses dos usuários, etc. Estas propriedades podem ser capturadas, utilizadas para calcular a similaridade entre os usuários e, a partir destes valores calculados, encontrar usuários semelhantes para gerar novas recomendações.

## 6.2 Cálculo de similaridade

Para a validação deste trabalho, provando que a recomendação pode acontecer utilizando-se das relações do *following* e não de uma formalização substancial de todo o conteúdo avaliado pelos usuários, a similaridade entre os usuários será calculada através da média aritmética de três características extraídas do sistema de *following*: cosseno do *following* de usuários ( $\cos(u)$ ), cosseno do *following* de temas ( $\cos(t)$ ) e força do relacionamento entre os usuários ( $fr$ ).

$$similaridade = \frac{\cos(u) + \cos(t) + fr}{3}$$

Como o objetivo do trabalho não é recomendar usuários e temas mas sim disponibilizar um mecanismo para ampliar o relacionamento entre os usuários e possibilitar a declaração de preferências em comunidades de prática, não é do seu escopo definir o peso da relação entre estas três métricas. O propósito de definir um valor de similaridade entre os usuários, utilizando apenas as informações de *following*, é validar a premissa de que a forma como os usuários se relacionam carrega informações relevantes que podem ser capturadas, armazenadas e recuperadas pelo sistema de *following*.

### 6.2.1 Similaridade do cosseno

O cosseno é utilizado para estimar a similaridade entre dois vetores e consiste em calcular o ângulo formado entre estes vetores. Para listas de palavras o resultado do cosseno é sempre um valor no intervalo  $[0..1]$ , sendo que 1 é o resultado para dois vetores que apontam para a mesma direção e 0 para vetores ortogonais. Esta medida independe do tamanho dos vetores.

$$\cos(\theta) = \frac{\sum_{i=1}^n A_i \times B_i}{\sqrt{\sum_{i=1}^n (A_i)^2} \times \sqrt{\sum_{i=1}^n (B_i)^2}}$$

O cosseno será utilizado para calcular a similaridade entre a lista de usuários e a lista temas que dois indivíduos seguem.

### 6.2.2 Força do relacionamento

A força do relacionamento entre os usuários leva em consideração, dentre os inúmeros aspectos presentes na estrutura de *following*, os seguintes atributos: proximidade, direção e relação inversa.

A proximidade entre os usuários é uma característica amplamente utilizada em sistemas de recomendação para redes sociais e baseia-se no quanto dois usuários estão distantes na rede de relacionamentos. A força de um relacionamento é inversamente proporcional à distância entre os usuários.

A direção do relacionamento é importante na medida que, ao seguir outros usuários, o indivíduo declara explicitamente seu interesse. A recíproca não é verdade, ao ser seguido por um usuário, o indivíduo pode ou não ter interesse por aquele determinado usuário e seus relacionamentos. Sendo assim, a força do relacionamento é maior quando o indivíduo segue um usuário do que quando é seguido por um usuário. Da mesma forma, quando existe a relação inversa e ambos os usuários possuem interesse um no outro a força do relacionamento deve ser reforçada.

O valor da força do relacionamento é calculado através de uma busca em amplitude no grafo gerado pelo *following* dos usuários. Em teoria dos grafos, a busca em amplitude consiste na pesquisa por determinado vértice, visitando-se primeiramente os vértices adjacentes. A pesquisa termina quando o vértice em questão é encontrado ou todos os vértices sejam visitados. O algoritmo para o cálculo da similaridade (Trecho de código 6.1) funciona da seguinte maneira:

Trecho de Código 6.1: Busca em amplitude para cálculo da força do relacionamento entre usuários.

```

1     def bfsSimilarity(current_id, target_id, max_depth = 4):
2         queue = []
3         visited = []
4         visited.append(current_id)
5         following_list = getNeighbors(current_id)
6         queue += [(f, current_id, 1, getStrength(f, current_id)) \
7                 for f in following_list]
8         match = False
9         while queue:
10            following = queue.pop(0)
11            visited.append(following[0])
12            if following[0] == target_id:
13                if not match or following[3]/following[2] > match[3]/match[2]:
14                    match = following
15            if not match and following[2] <= max_depth:
16                following_list = diff(getNeighbors(following[0]), visited)
17                level = following[2] + 1
18                queue += [(f, following[0], following[2]+1,
19                          following[3] + getStrength(f, following[0])/level) \
20                        for f in following_list]
21        if match:
22            return match[3]/match[2]
23        return 0

```

1. O algoritmo recebe o usuário para o qual será calculada a força do relacionamento (usuário corrente) e o usuário que se deseja encontrar no grafo (usuário alvo). A força do relacionamento é referente à ligação no sentido do usuário corrente para o usuário alvo. Também é possível parametrizar a quantidade de níveis que o algoritmo irá visitar na busca pelo usuário alvo.
2. Visita-se a lista de *following* do usuário corrente e para cada um deles é calculada a força de seu relacionamento. Para que a direção e a relação inversa do *following* sejam consideradas, a força do relacionamento é definida assim: se o usuário corrente seguir o usuário da lista de *following* soma-se à força do relacionamento 0.7, se o usuário corrente é seguido pelo usuário da lista de *following* soma-se 0.3. O valor máximo para a relação entre dois usuários adjacentes é 1, *following* mútuo.
3. A força do relacionamento entre os usuários adjacentes é calculada para cada um dos níveis até que o usuário alvo seja encontrado, o nível máximo de pesquisa seja alcançado ou todo o grafo tenha sido percorrido. A força do relacionamento é sempre dividida pela profundidade da pesquisa. Quanto maior a distância percorrida no grafo, menor deve ser a força dos relacionamentos.
4. Caso o usuário alvo seja encontrado, a força do relacionamento entre o usuário

corrente e o usuário alvo é a média da força de todos os relacionamentos dos usuários adjacentes visitados durante o percurso da pesquisa. Caso contrário, o valor da força do relacionamento é 0.

### 6.3 Filtragem colaborativa item-item

Existem inúmeras abordagens de recomendações baseadas em filtragem colaborativa, mas pode-se resumi-las em duas direções: baseada em usuário e baseada em item. A abordagem baseada em usuário procura por usuários que compartilham dos mesmos padrões de avaliações e estima o valor da avaliação de itens ainda não acessados pelo usuário para o qual será gerada a recomendação. A filtragem colaborativa baseada em item cria uma matriz item-item, determinando a similaridade entre pares de itens de modo assíncrono e, a partir dessa similaridade, as recomendações de itens são geradas (LINDEN; SMITH; YORK, 2003).

AGARWAL; BHARADWAJ (2012), em seu trabalho para recomendação de amigos em redes sociais, modela uma matriz item-item onde os itens da matriz são usuários. A similaridade entre os usuários é calculada utilizando três aspectos da rede: proximidade, correlação e força do relacionamento. A recomendação é a lista de  $N$  usuários que ainda não estão na lista de amigos do usuário alvo de recomendação.

Com base nesta abordagem item-item, onde os itens da matriz representam os usuários, um algoritmo de recomendação de usuários compreende os seguintes passos:

1. Computar a similaridade para cada par de vértices
2. Ranquear as similaridades em ordem decrescente
3. Determinar um limiar de corte
4. Recomendar o relacionamento dos pares até este limiar

Determinar a similaridade entre os usuários é o passo mais importante para esta abordagem. Por isso a validação do sistema de *following* concentra-se neste primeiro passo.

### 6.4 Cenários de recomendação

O processo de recomendação calcula a similaridade entre o usuário alvo de recomendação e os demais usuários que participam do sistema de *following*. A similaridade só é calculada entre aqueles que são seguidos ou seguem pelo menos um usuário ou tema.

Após definida a similaridade entre os usuários, a recomendação sugere novas

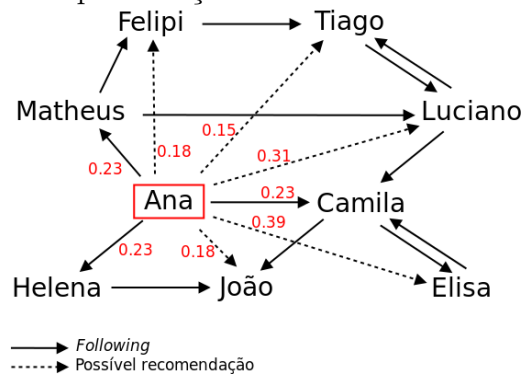
ligações no grafo. Os usuários recomendados são aqueles que possuem maior similaridade com o usuário alvo da recomendação e que ainda não fazem parte de sua lista de usuários seguidos. Os temas recomendados são aqueles presentes nas listas dos usuários mais similares ao alvo de recomendação e que ainda não fazem parte de sua lista de temas seguidos.

Para os cenários apresentados a seguir serão utilizados o conjunto de usuários  $U = \{\text{Ana, Camila, Elisa, Felipi, Helena, João, Luciano, Matheus, Tiago}\}$  e o conjunto de temas  $T = \{A, B, C, D, E\}$ . O *following* será definido para cada um dos cenários.

#### 6.4.1 Cenário de recomendação 1

O primeiro cenário de recomendação apresenta apenas ligações entre usuários e tem por objetivo demonstrar as relações de proximidade e direção do sistema de *following*. A estrutura das ligações é apresentada na Figura 6.2 juntamente com as possíveis recomendações e os valores de similaridade calculados para o alvo de recomendação (Ana).

Figura 6.2: Representação do cenário de recomendação 1.



Observando a Figura 6.3 percebe-se que a relação de proximidade e direção são respeitadas pelo algoritmo que calcula a força do relacionamento entre os usuários. Os maiores valores de força do relacionamento (*bfs\_graph*) são das ligações diretas de *following* com o usuário alvo. Após o *following* direto, o maior valor de força do relacionamento é referente a ligação com Elisa por conta do *following* mútuo entre Elisa e Camila, estando mais próxima de Ana do que os usuários João ou Felipi, por exemplo.

Os usuários recomendados são aqueles com maior valor de similaridade que ainda não são seguidos por Ana. Não considerando um limiar de corte a ordem seria: Elisa, Luciano, João, Felipe, Tiago.

Com as informações disponíveis através do *following* de usuários, João e Felipe possuem o mesmo valor de similaridade para Ana. Isto porque não foi levado em consideração, neste cenário, as preferências de temas.

Figura 6.3: Tabela de similaridade para o cenário de recomendação 1.

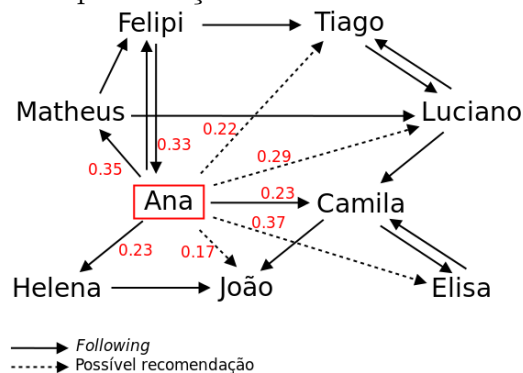
	target_member [PK] character varying	recommended_member [PK] character varying	cosine_tags double precision	cosine_users double precision	bfs_graph double precision	similarity double precision
1	ana	ana	1	1	1	1
2	ana	elisa	0	0.5773502691896	0.6	0.392450089729875
3	ana	luciano	0	0.4082482904638	0.525	0.311082763487954
4	ana	camila	0	0	0.7	0.233333333333333
5	ana	helena	0	0	0.7	0.233333333333333
6	ana	matheus	0	0	0.7	0.233333333333333
7	ana	joao	0	0	0.525	0.175
8	ana	felipi	0	0	0.525	0.175
9	ana	tiago	0	0	0.461111111111111	0.153703703703704
*						

9 rows.

#### 6.4.2 Cenário de recomendação 2

Este cenário (Figura 6.4) possui as mesmas ligações do cenário anterior com a adição das relações de *following*  $F = \{(Ana, Felipe), (Felipe, Ana)\}$ .

Figura 6.4: Representação do cenário de recomendação 2.



Observando a Figura 6.5 nota-se a alteração do valor da força do relacionamento entre Ana e Tiago, conforme esperado. Apesar da força do relacionamento ser maior entre os usuários Ana e Tiago do que entre Ana e Elisa, Elisa continua sendo o primeiro usuário a ser recomendado pelo sistema. Isto acontece porque Ana e Elisa possuem um *following* de usuário em comum (Camila) que fortalece o valor da similaridade entre os usuários.

Com esta nova relação entre Ana e Felipe, Tiago passa a ser mais próximo de Ana e a recomendação seria: Elisa, Luciano, Tiago, João.



Figura 6.5: Tabela de similaridade para o cenário de recomendação 2.

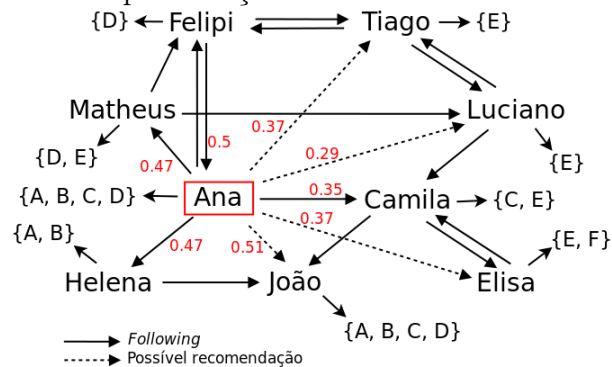
	target_member [PK] character varying	recommended_member [PK] character varying	cosine_tags double precision	cosine_users double precision	bfs_graph double precision	similarity double precision
1	ana	ana	1	1	1	1
2	ana	elisa	0	0.5	0.6	0.366666666666667
3	ana	matheus	0	0.3535533905932	0.7	0.351184463531091
4	ana	felipi	0	0	1	0.333333333333333
5	ana	luciano	0	0.3535533905932	0.525	0.292851130197758
6	ana	helena	0	0	0.7	0.233333333333333
7	ana	camila	0	0	0.7	0.233333333333333
8	ana	tiago	0	0	0.675	0.225
9	ana	joao	0	0	0.525	0.175
*						

9 rows.

### 6.4.3 Cenário de recomendação 3

Neste cenário de recomendação (Figura 6.6) foram inseridas as relações de *following* de temas e a ligação  $F = (\text{Tiago}, \text{Felipi})$ .

Figura 6.6: Representação do cenário de recomendação 3.



Nota-se que o cosseno do *following* de temas altera consideravelmente a relação dos usuários. Estas informações, declaradas explicitamente pelos usuários, são capazes de diferenciar usuários que estão em distâncias equivalentes no grafo, inclusive recomendando usuários mais distantes por conta das preferências dos temas.

O usuário João, que antes era a última opção para recomendação, passa a ser a primeira, mesmo tendo uma força de relacionamento menor do que os outros usuários. Este é um resultado induzido, mas serve para mostrar que o sistema de *following* é capaz de armazenar informações relevantes das preferências dos usuários e que, a partir de seus assuntos preferidos e *following* de usuários, é possível gerar recomendações personalizadas.

A Figura 6.7 apresenta os valores de similaridades calculados para Ana. A lista de usuários a serem recomendados é: João, Tiago, Elisa, Luciano.

Figura 6.7: Tabela de similaridade para o cenário de recomendação 3.

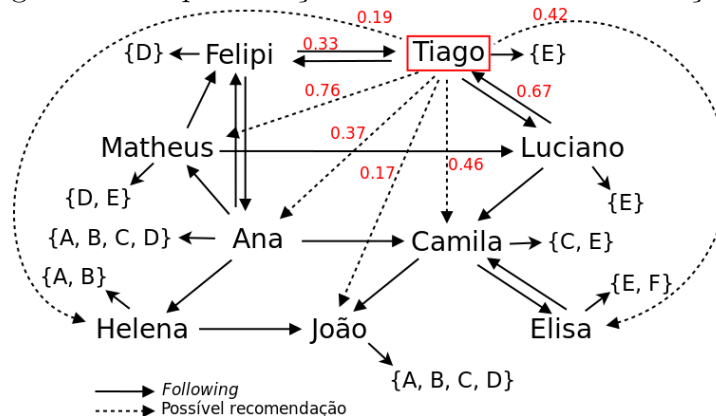
	target_member [PK] character varying	recommended_member [PK] character varying	cosine_tags double precision	cosine_users double precision	bfs_graph double precision	similarity double precision
1	ana	ana	1	1	1	1
2	ana	joao	1	0	0.525	0.5083333333333333
3	ana	felipi	0.5	0	1	0.5
4	ana	helena	0.70710678118654	0	0.7	0.469035593728849
5	ana	matheus	0.35355339059327	0.35355339059327	0.7	0.469035593728849
6	ana	tiago	0	0.35355339059327	0.75	0.367851130197758
7	ana	elisa	0	0.5	0.6	0.3666666666666667
8	ana	camila	0.35355339059327	0	0.7	0.351184463531091
9	ana	luciano	0	0.35355339059327	0.525	0.292851130197758
*						

9 rows.

#### 6.4.4 Cenário de recomendação 4

Finalmente, a Figura 6.8 possui as mesmas ligações do cenário anterior. A alteração está em quem receberá as recomendações, Tiago.

Figura 6.8: Representação do cenário de recomendação 4.



A lista de usuários que serão recomendados para Tiago é completamente diferente da lista de recomendações de Ana. Isso porque o sistema de *following* é capaz de armazenar as informações das preferências dos usuários de maneira individual, mesmo em uma única estrutura. A individualidade está também na forma como os dados são recuperados, o cálculo da similaridade, que atribui pesos diferentes para as relações de seguir e ser seguido. A Figura 6.9 apresenta os valores de similaridade para este cenário. A lista de recomendações para este cenário seria: Matheus, Camila, Elisa, Ana, Helena, João.

Figura 6.9: Tabela de similaridade para o cenário de recomendação 4.

	target_member [PK] character varying	recommended_member [PK] character varying	cosine_tags double precision	cosine_users double precision	bfs_graph double precision	similarity double precision
1	tiago	tiago	1	1	1	1
2	tiago	matheus	0.70710678118654	1	0.5777777777777778	0.761628186321442
3	tiago	luciano	1	0	1	0.6666666666666667
4	tiago	camila	0.70710678118654	0	0.675	0.460702260395516
5	tiago	elisa	0.70710678118654	0	0.5611111111111111	0.422739297432553
6	tiago	ana	0	0.3535533905932	0.75	0.367851130197758
7	tiago	felipi	0	0	1	0.3333333333333333
8	tiago	helena	0	0	0.5777777777777778	0.192592592592593
9	tiago	joao	0	0	0.5277777777777778	0.175925925925926
*						

9 rows.

## 6.5 Conclusões sobre o capítulo

Este capítulo demonstrou a relevância do sistema de *following* proposto para as comunidades de prática. Foram apresentados a estrutura gerada pelo *following*, a abordagem de recomendação e o cálculo de similaridade que pretende capturar alguns atributos relevantes da rede de relacionamentos. Também foram apresentados quatro cenários de recomendação.

A proposta de recomendação e seus cenários não abordaram todas as nuances que os relacionamentos dos usuários podem apresentar. Porém, serviram para provar que o sistema de *following* é capaz de representar os interesses distribuídos na rede de relacionamentos agora disponível nas comunidades de prática. Percebe-se também que a interação e a troca de experiências entre os participantes são incentivadas, uma vez que novos relacionamentos poderão ser recomendados.

## 7 CONSIDERAÇÕES FINAIS

Ao longo deste trabalho foi analisado como o conhecimento é gerado no contexto de comunidades de prática e de que forma a utilização de ferramentas de comunicação colabora com seu desenvolvimento no ambiente virtual.

Os sistemas de recomendação são ferramentas que proporcionam um mecanismo de busca personalizado, procurando exibir itens de maior relevância àquele usuário. Para que um sistema de recomendação possa gerar sugestões de qualidade, ele deve primeiramente conhecer o usuário e conseguir representar seus interesses. Foram apresentadas as técnicas de recomendação mais utilizadas e ao considerar, no contexto de comunidades de prática, as vantagens, desvantagens e características de cada uma das abordagens, definiu-se a filtragem colaborativa como a técnica de sistemas de recomendação a ser utilizada nesta proposta.

O capítulo referente à plataforma de comunidades de prática apresentou as ferramentas já implementadas no ambiente e a tecnologia Python/Zope/Plone, utilizada no desenvolvimento do *framework*. O sistema de *following* foi contextualizado e fatores importantes para seu desenvolvimento, como a representação de usuários, temas e suas relações, foram analisados.

Verificada a falta de um mecanismo que relacione usuários e temas, e que possa servir para a declaração de interesses nas comunidades de prática, foi proposto o desenvolvimento do sistema de *following*. Seus requisitos foram definidos e a arquitetura e tecnologias utilizadas foram apresentadas.

Esta nova ferramenta, disponível nas comunidades de prática, permite a representação de informações relevantes do perfil do usuário e, por consequência, pode ser utilizada para gerar recomendações de qualidade. Recomendar não fez parte do trabalho proposto, mas toda sua estrutura foi preparada e agora está disponível para estudos posteriores.

## 7.1 Contribuições

O presente trabalho abordou a importância dos relacionamentos entre os usuários no contexto de comunidades de prática e de que forma a utilização de ferramentas de comunicação colabora com o desenvolvimento social do conhecimento. O participante de uma comunidade de prática, que antes era produtor e consumidor de conhecimento, agora também é um agente socializador.

O sistema de *following* gera um grafo direcionado de interações, identificando a importância e a força entre as relações. A partir dele é possível conhecer e armazenar as preferências dos usuários. Esta estrutura de *following*, construída de forma dinâmica e colaborativa, é capaz de capturar as inúmeras relações entre usuários que podem ser utilizadas para gerar recomendações personalizadas.

O desenvolvimento do sistema de *following* agregou novas funcionalidades à plataforma de comunidades de prática, *CoPPLA*, proposta inicialmente em FIORIO; SILVA; RIBEIRO (2011) e RIBEIRO et al. (2011). Estas funcionalidades ampliam e estimulam as interações entre os participantes das comunidades de prática e colaboram com o processo de aprendizagem. Os sistemas de recomendação, que podem ser desenvolvidos utilizando como base as informações de *following*, proporcionam um ambiente com conteúdo personalizado que garante uma participação de diferentes níveis e mantém ativo o diálogo e práticas nas comunidades.

## 7.2 Trabalhos futuros

Utilizando as informações de *following* é possível gerar recomendações personalizadas de usuários e temas. Em um próximo nível, é possível identificar e recomendar comunidades que tratam de assuntos relacionados.

Outras características relevantes, diferentes das abordadas no trabalho, podem ser capturadas da estrutura de *following*, como por exemplo, a *expertise* de um usuário sobre determinado tema, a importância de um usuário para a rede e a correção de temas. Também é possível explorar características da estrutura do grafo como detecção de agrupamentos e descoberta de sub-grafos.

A abordagem temporal não foi explorada no sistema de *following*. A interação dos usuários pode ser monitorada e utilizada para fortalecer ou enfraquecer os relacionamentos uma vez que nossos interesses mudam com o passar do tempo.

Por fim, é possível armazenar o sistema de *following* em *Triple Store*, tornando a estrutura aderente ao modelo da web semântica, promovendo a uniformidade e a interoperabilidade com outras plataformas Web.

## REFERÊNCIAS

ADOMAVICIUS, G.; TUZHILIN, A. Toward the next generation of recommender systems: a survey of the state-of-the-art and possible extensions. **Knowledge and Data Engineering, IEEE Transactions on**, [S.l.], v.17, n.6, p.734–749, Junho 2005.

AGARWAL, V.; BHARADWAJ, K. K. A collaborative filtering framework for friends recommendation in social networks based on interaction intensity and adaptive user similarity. **Social Network Analysis and Mining**, [S.l.], Setembro 2012.

ALMAZRO, D. et al. A Survey Paper on Recommender Systems. **arXiv.org**, [S.l.], v.4, p.234–242, Junho 2010.

**Archetypes**. [S.l.]: Plone.org, 2013. Disponível em: <<http://plone.org/products/archetypes>>. Acessado em 16/06/2013.

ASPELI, M. **Professional Plone Development**. 1.ed. Birmingham, B27 6PA, UK: Packt Publishing Ltd, 2007. ISBN 978-1-847191-98-4.

BARCELLOS, C. D. et al. Sistema de Recomendação Acadêmico para Apoio a Aprendizagem. **RENOTE - Revista Novas Tecnologias na Educação**, [S.l.], v.5, n.2, Dezembro 2007.

BURKE, R. Hybrid recommender systems: survey and experiments. **User Modeling and User-Adapted Interaction**, [S.l.], v.12, n.4, p.331–370, 2002.

CAZELLA, S. C. **Aplicando a Relevância da Opinião de Usuários em Sistema de Recomendação para Pesquisadores**. 2006. Tese (Doutorado) — Universidade Federal do Rio Grande do Sul. Instituto de Informática. Programa de Pós-Graduação em Computação.

CAZELLA, S. C.; NUNES, M. A. S. N.; REATEGUI, E. A. A ciência da opinião: estado da arte em sistemas de recomendação. , [S.l.], v.1, p.161–216, 2010. André

Ponce de Leon F. de Carvalho; Tomasz Kowaltowski..(Org.). Jornada de Atualização de Informática-JAI 2010-CSBC2010. Rio de Janeiro: PucRIO.

DELMONTE, M. et al. **The Definitive Guide to Plone, Second Edition**. [S.l.]: Apress, 2009. 7p. ISBN-13 (pbk): 978-1-4302-1893-7.

FIORIO, M.; SILVA, J. L. T.; RIBEIRO, A. M. Um framework de comunidades de prática em ambientes virtuais de aprendizagem. **RENOTE - Revista Novas Tecnologias na Educação**, [S.l.], 2011.

FOWLER, M. **Patterns of Enterprise Application Architecture**. Boston, MA, USA: Addison-Wesley Longman Publishing Co., Inc., 2002.

GOLDBERG, D. et al. Using collaborative filtering to weave an information tapestry. **Communications of the ACM - Special issue on information filtering**, [S.l.], v.35, n.12, p.61–70, Dezembro 1992.

GUTTMAN, R. H. **Merchant Differentiation through Integrative Negotiation in Agent-mediated Electronic Commerce**. 1998. Dissertação (Mestrado) — School of Architecture and Planning, Program in Media Arts and Sciences, Massachusetts Institute of Technology.

HEALY, A. Communities of Practice as a Support Function for Social Learning in Distance Learning Programs. **M.D. Lytras, al. et (Eds.), WSKS 2009, CCIS 49**, [S.l.], p.49–56, 2009.

HERLOCKER, J. L. **Understanding and Improving Automated Collaborative Filtering Systems**. 2000. Tese (Doutorado) — University of Minnesota.

KRULWICH, B. Lifestyle Finder - Intelligent User Profiling Using Large-Scale Demographic Data. **Artificial Intelligence Magazine**, [S.l.], v.18, n.2, p.37–45, 1997.

LERNER, R. M. Databases and Zope. **linuxjournal**, [S.l.], Maio 2002. Disponível em: <<http://www.linuxjournal.com/article/5849>>. Acessado em 16/06/2013.

LINDEN, G.; SMITH, B.; YORK, J. Amazon.Com Recommendations: item-to-item collaborative filtering. **IEEE Internet Computing**, Piscataway, NJ, USA, v.7, n.1, p.76–80, Jan. 2003.

PETERS, T. **Python - The Zen of Python**. [S.l.]: Python.org, 2004. Disponível em: <<http://www.python.org/dev/peps/pep-0020/>>. Acessado em 04/06/2013.

PLONE. **Plone**. [S.l.]: Plone.org, 2012. Disponível em: <<http://plone.org/products/plone>>. Acessado em 08/06/2013.

PLONE. **Plone - Catalog**. [S.l.]: Plone.org, 2013. Disponível em: <[http://developer.plone.org/searching\\_and\\_indexing/catalog.html](http://developer.plone.org/searching_and_indexing/catalog.html)>. Acessado em 20/06/2013.

PLONE. **Plone - Member Manipulation**. [S.l.]: Plone.org, 2013. Disponível em: <[http://developer.plone.org/members/member\\_basics.html](http://developer.plone.org/members/member_basics.html)>. Acessado em 22/06/2013.

POSTGRESQL. **PostgreSQL - About**. [S.l.]: The PostgreSQL Global Development Group, 2013. Disponível em: <<http://www.postgresql.org/about/>>. Acessado em 20/06/2013.

PYTHON. **Python - History and License**. [S.l.]: Python.org, 2013. Disponível em: <<http://docs.python.org/2/license.html>>. Acessado em 04/06/2013.

PYTHON. **Python - About Python**. [S.l.]: Python.org, 2013. Disponível em: <<http://python.org/about/>>. Acessado em 04/06/2013.

PYTHON. **Python - Application Domains**. [S.l.]: Python.org, 2013. Disponível em: <<http://www.python.org/about/apps/>>. Acessado em 04/06/2013.

REATEGUI, E. B.; CAZELLA, S. C. Sistemas de Recomendação. **XXV Congresso da Sociedade Brasileira de Computação. Unisinos**, [S.l.], 2005. São Leopoldo, RS.

REATEGUI, E.; CAMPBELL, J. A.; TORRES, R. **Using Item Descriptors in Recommender Systems**. [S.l.: s.n.], 2004. Nineteenth National Conference on Artificial Intelligence (AAAI) - Workshop on Semantic Web Personalization. San Jose, USA.

REATEGUI, E.; LORENZATTI, A. Um Assistente Virtual para Resolução de Dúvidas e Recomendação de Conteúdo. **Proceedings V Encontro Nacional de Inteligência Artificial (ENIA 2005)**, [S.l.], Julho 2005. São Leopoldo, RS.

RIBEIRO, A. M. et al. Dos Ambientes de Aprendizagem às Comunidades de Prática. **Anais do XXII SBIE - XVII WIE**, [S.l.], 2011. Aracaju, 21 a 25 de novembro de 2011.

SARWAR, B. M. et al. Using Filtering Agents to Improve Prediction Quality in the GroupLens Research Collaborative Filtering System. **CSCW '98 Proceedings of the 1998 ACM conference on Computer supported cooperative work**, [S.l.], p.345–354, 1998.



SCHAFFER, J. B.; KONSTAN, J. A.; RIELD, J. Recommender Systems. **EC '99 Proceedings of the 1st ACM conference on Electronic commerce**, [S.l.], p.158–166, 2000.

SILVA, J. L. T.; RIBEIRO, A. M.; FIORIO, M. Um estudo de implantação de Comunidades de Prática em um portal institucional. **RECIIS – R. Eletr. de Com. Inf. Inov. Saúde**, [S.l.], v.3, n.3, 2011. Rio de Janeiro.

SQLALCHEMY. **SqlAlchemy - Object Relational Tutorial**. [S.l.: s.n.], 2013. Disponível em: <[http://docs.sqlalchemy.org/en/rel\\_0.8/orm/tutorial.html](http://docs.sqlalchemy.org/en/rel_0.8/orm/tutorial.html)>. Acessado em 20/06/2013.

WENGER, E. **Communities of practice: learning, meaning, and identity** (learning in doing: social, cognitive and computational perspectives). [S.l.]: Cambridge University Press, 1998. ISBN 978-0-521-66363-2.

WENGER, E. Communities of Practice and Social Learning Systems. , [S.l.], v.7, n.2, p.225–246, Maio 2000. Wenger Organization.

WENGER, E. **Communities of Practice: learning as a social system**. [S.l.: s.n.], 2008.

WENGER, E. et al. **Technology for Communities**. [S.l.: s.n.], 2005.

WENGER, E.; WHITE, N.; SMITH, J. D. **Digital Habitats: stewarding technology for communities**. [S.l.]: CPsquare, 2010.

ZAIANE, O. R. Web Mining: concepts, practices and research. **Simpósio Brasileiro de Banco de Dados, SBBD**, [S.l.], v.43, n.8, p.410–474, 2000. João Pessoa.

ZODB. **ZODB - a native object database for Python**. [S.l.: s.n.], 2013. Disponível em: <<http://zodb.org/>>. Acessado em 07/06/2013.

ZOPE. **Zope - Solutions Overview**. [S.l.]: Zope Corporation, 2013. Disponível em: <<http://www.zope.com/products/x255092005/Solutions-Overview>>. Acessado em 07/06/2013.

ZOPE. **Zope - Searching and Categorizing Content**. [S.l.]: Zope Corporation, 2013. Disponível em: <<http://docs.zope.org/zope2/zope2book/SearchingZCatalog.html>>. Acessado em 20/06/2013.

ZOPE. **Zope - Users and Security**. [S.l.]: Zope Corporation, 2013.  
Disponível em: <<http://docs.zope.org/zope2/zope2book/Security.html>>. Acessado em 22/06/2013.