

# Gerência de Conhecimento na Engenharia de Requisitos

Denise F. Togneri<sup>1,2</sup>, Ricardo de A. Falbo<sup>2</sup>, Crediné S. de Menezes<sup>2</sup>, Bernardo S. Wernesback<sup>1</sup>,  
Diogo Q. de Almeida<sup>1</sup>, Marina F. Côrtes<sup>1</sup>

<sup>1</sup>FAESA - Faculdades Integradas Espírito-santenses  
Colegiado do Curso de Ciência da Computação  
Rua Anselmo Serrat, 199, Vitória-ES, Brazil, CEP: 29041-010  
<sup>2</sup>UFES - Universidade Federal do Espírito Santo  
Centro Tecnológico - Mestrado em Informática  
Avenida Fernando Ferrari, s/n, Vitória-ES, Brazil, CEP: 29060-900  
togneri@terra.com.br, {falbo, credine}@inf.ufes.br

## Resumo

*A Engenharia de Requisitos – uma das principais atividades da Engenharia de Software – é um processo sistemático de captura, modelagem e documentação de requisitos através de uma abordagem iterativa e cooperativa. É uma atividade que envolve um processo intensivo de conhecimento [5] e nesse sentido, é muito importante a adoção da Gerência de Conhecimento para apoiar sua realização. A área de Gerência de Conhecimento tem como objetivo promover o amadurecimento, a comunicação e a preservação do conhecimento formal e informal existente nas organizações. Este trabalho investiga como a área de Gerência de Conhecimento pode apoiar as atividades do processo de Engenharia de Requisitos e apresenta uma ferramenta que utiliza a tecnologia de agentes e se propõe a integrar essas duas áreas. Essa ferramenta foi desenvolvida para Web e foi modelada usando o paradigma da orientação a objetos e a metodologia GAIA [14] para modelagem dos agentes.*

## 1. Introdução

No contexto da Engenharia de Requisitos (ER), as organizações de software, como qualquer outra organização, sofrem com problemas relacionados à falta de gerência de conhecimento, uma vez que este está, em sua maior parte, na mente de seus membros [1]. Dentre os vários problemas a serem enfrentados, destacam-se [2]: (1) perda de conhecimento que pode ocorrer se houver receio de compartilhá-lo ou quando um profissional se desliga da organização; (2) escassez do conhecimento, que dificulta sua obtenção por parte de novos profissionais, diminuindo a produtividade; (3) falta de tempo para compartilhar o conhecimento; (4) dificuldade em localizar um conhecimento específico para resolver um problema e (5) dificuldade na reutilização do conhecimento quando do desenvolvimento de projetos similares.

Este artigo discute o uso de gerência de conhecimento na Engenharia de Requisitos e apresenta CRETA [13], uma ferramenta que apóia a gerência de conhecimento neste contexto. A ferramenta permite que o conhecimento criado, aprendido e utilizado durante o processo de Engenharia de Requisitos seja formalizado, armazenado e colocado à disposição dos membros da organização. Ela oferece recursos para atender a essas necessidades, possibilitando, ainda, a organização e disseminação de conhecimento através do uso de agentes.

Este artigo está estruturado da seguinte forma: a seção 2 aborda a Gerência de Conhecimento na Engenharia de Requisitos. Na seção 3, os requisitos de uma infra-estrutura para apoiar a Gerência de Conhecimento são apresentados. A seção 4 introduz CRETA, uma ferramenta de apoio à Engenharia de Requisitos Cooperativa, que explora o apoio à Gerência de Conhecimento na Engenharia de Requisitos e discute os serviços de gerência de conhecimento gerais e específicos que foram introduzidos em CRETA. A seção 5 discute os trabalhos correlatos. Finalmente, na seção 6, são apresentadas as conclusões deste trabalho.

## 2. Gerência de Conhecimento e Engenharia de Requisitos

A Gerência de Conhecimento (GC) consiste em coletar e armazenar sistematicamente o conhecimento adquirido, compartilhar este conhecimento através de uma memória organizacional e promover o surgimento de novo conhecimento [11]. Neste contexto, conhecimento pode ser definido como informação combinada com experiência, contexto, interpretação e reflexão, podendo ser explícito ou tácito [4]. Na literatura existem vários conjuntos de atividades propostos para a GC [1, 7]. Abecker et al. [1], por exemplo, afirmam que as atividades básicas da GC são: identificação, aquisição, desenvolvimento, disseminação, uso e preservação de conhecimento. Nessa visão, a memória organizacional está no centro do processo, sendo considerada como um repositório do conhecimento disponível na organização [9]. Para atingir seus objetivos, a GC pode se beneficiar de diversas tecnologias, tais como *data warehouses*, Internet, máquinas de busca, ontologias, agentes e ferramentas de *groupware* [2, 5, 10, 11, 12].

Uma organização de software deve possuir um processo de ER maduro, definido e bem documentado, com objetivo de implementar boas práticas. Devem ser documentados, dentre outros, os procedimentos, as atividades a serem realizadas, as competências individuais, as lições aprendidas, os padrões de documentos e os diversos artefatos produzidos, tais como planos e atas de entrevistas, especificações de requisitos de software e mensagens eletrônicas trocadas [8]. Esses documentos não são apenas produtos do trabalho, pois existe informação adicional dentro deles: (1) durante o projeto eles documentam as decisões; (2) após o fim do projeto, eles contêm o seu histórico. Podem ser reutilizados de diversas formas em projetos futuros, para que as pessoas possam aprender com eles, através da análise das soluções de diferentes problemas similares [12]. Dada a complexidade envolvida no processo de ER, uma preocupação crescente tem sido organizar, registrar e gerenciar o conhecimento, experiências, documentos e padrões envolvidos, de forma a promover reuso, disseminação, compartilhamento e aprendizado organizacional. Sendo assim, uma abordagem promissora consiste em utilizar a GC para apoiar o processo de ER. A criação de um repositório de conhecimento serviria como uma área para o compartilhamento do conhecimento, apoiando várias das atividades da ER, tais como a definição do processo de ER e a elicitação de requisitos.

## 3. Requisitos de uma Infra-estrutura para Gerência de Conhecimento

Para ser efetivamente utilizada, a GC deve estar integrada ao processo da organização, permitindo que o conhecimento relevante seja coletado e armazenado, à medida que é gerado no trabalho. Natali & Falbo [10] discutiram a importância da GC no desenvolvimento de software e apresentaram requisitos de uma infra-estrutura de Gerência de Conhecimento para organizações de software. Como a ER é parte do processo de desenvolvimento de software, esses requisitos também se aplicam a uma infra-estrutura de GC para apoiar a ER, tratando dos seguintes aspectos [10]:

- R1. Deve ser permitido o cadastro do conhecimento formal sobre processos de software, incluindo conhecimento sobre processos, atividades e procedimentos, dentre outros;
- R2. Deve ser permitido o cadastro dos conhecimentos informais existentes na organização, tais como competências individuais e lições aprendidas durante a realização de um processo de software;
- R3. Deve-se permitir a exclusão periódica de itens de conhecimento formais ou informais obsoletos ou sem utilidade, tomando por base o feedback dos usuários;
- R4. Deve-se permitir a indexação tanto do conhecimento formal como das lições aprendidas aprovadas;
- R5. Deve-se fornecer mecanismos de busca aos diversos itens de conhecimento;
- R6. Deve-se permitir que o usuário avalie o conhecimento utilizado, classificando-o e registrando comentários, se for necessário;
- R7. Deve-se fornecer mecanismos de disseminação do conhecimento de forma pró-ativa, ou seja, de iniciativa do próprio sistema. Os itens de conhecimento que o sistema julgar relevantes devem ser apresentados ao usuário como uma sugestão de ajuda na realização de uma atividade;
- R8. Deve-se permitir o registro do projeto de software pelo gerente de projeto, bem como do processo adotado para seu desenvolvimento de forma a possibilitar que possa ser controlado de forma efetiva;
- R9. Durante a definição do processo de software, o sistema deve oferecer conhecimento já cadastrado sobre processos de software de projetos semelhantes e fornecer sugestões para o gerente de projeto;
- R10. Deve-se possibilitar o apoio às diversas atividades do processo de software de um projeto específico, permitindo que os artefatos de software produzidos possam ser armazenados, reusados e disseminados.

A Engenharia de Software, e por conseguinte a ER, trata de trabalho em equipe e, para apoiar as interações e discussões em grupo, diversas aplicações de *groupware* podem ser utilizadas [3]. Nesse sentido, uma infraestrutura geral de GC também deve se beneficiar desse uso e, portanto, os seguintes requisitos gerais relativos à cooperação na GC também devem ser atendidos:

- R11. Deve-se permitir que as comunicações síncronas ou assíncronas mediadas por computador, relativas a um projeto de software, contendo as discussões, as idéias, as justificativas e as opiniões dos membros da equipe possam ser registradas;
- R12. Em relação à memória organizacional, os itens de conhecimento registrados referentes à memória do processo de interação e comunicação em grupo devem ser organizados e indexados para facilitar o compartilhamento, mesmo que os participantes estejam geograficamente distribuídos;
- R13. Deve-se permitir o registro das atividades do processo de software adotado, de forma a oferecer mecanismos para que um participante de um projeto visualize os demais participantes ativos, facilitando a comunicação, e perceba quem são seus parceiros, seu papel e as instruções para execução de cada atividade, bem como o período previsto e realizado e a situação de cada uma delas;
- R14. Deve-se possibilitar a automatização da coleta e apresentação para o grupo das informações sobre as atividades dos participantes individuais, dentro de um espaço de trabalho compartilhado, possibilitando atualização contínua das ações dos participantes e do progresso global do grupo.

Conforme destacado por Natali & Falbo [10], o requisito R7 que trata da disseminação do conhecimento, ainda que seja um requisito geral da Gerência do Conhecimento, deve ser definido através de ferramentas específicas, levando em conta seu funcionamento e características particulares, uma vez que não é possível oferecer ajuda pró-ativa sem conhecer detalhes sobre a tarefa que está sendo realizada.

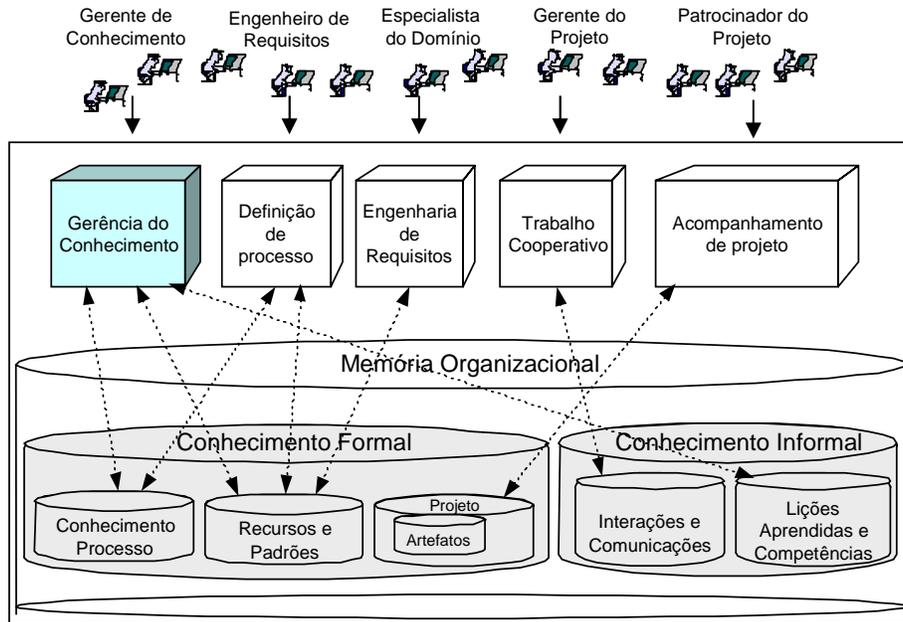
#### **4. Introduzindo Gerência de Conhecimento em CRETA - uma Ferramenta de Apoio à Engenharia de Requisitos Cooperativa**

CRETA – uma ferramenta de apoio à Engenharia de Requisitos Cooperativa [13] se propõe a integrar aplicações de *groupware* com a área de ER, auxiliando o trabalho de engenheiros de requisitos, especialistas de domínio, gerentes de projetos, patrocinadores e gerentes de conhecimento, que interagem com ela. Seu objetivo é apoiar as atividades principais do processo de ER, tais como elicitação, análise, negociação, documentação, validação, gerência de requisitos e acompanhamento do processo de ER. Os participantes fazem uso das ferramentas de *groupware* disponíveis - tais como agenda eletrônica, correio e compromisso eletrônico, listas de discussões, reuniões virtuais síncronas (*chat*) e assíncronas (fórum) - para facilitar a comunicação, cooperação e interação entre os envolvidos.

A implementação de CRETA foi baseada na Web, utilizando-se as tecnologias Java e JSP-*JavaServer Pages*, de forma a permitir que tanto os participantes geograficamente dispersos quanto os usuários internos de uma organização acessem a ferramenta. Entretanto, em sua versão original [13], CRETA não possuía apoio à GC. Sendo assim, um novo componente denominado *Gerência de Conhecimento* foi desenvolvido e integrado à CRETA, com objetivo de incorporar novas funcionalidades que possibilitam o registro do conhecimento criado ou capturado, a recuperação, o acesso, o uso, a disseminação e a manutenção do conhecimento armazenado, como mostra a figura 1.

A base do sistema de GC proposto é sua memória organizacional, que gerencia o conhecimento necessário à ER. Os diferentes itens de conhecimento existentes em CRETA foram agrupados em duas categorias:

- *Item de Conhecimento Formal*, que agrupa o conhecimento sobre o processo de ER, incluindo o conhecimento sobre atividades, modelos de ciclo de vida, métodos, padrões, recursos, os artefatos gerados durante a execução deste processo, dentre outros. Esses itens de conhecimento foram definidos com base na ontologia de processo de software proposta em [6]. Agrupa também o conhecimento sobre os padrões de documentos e listas de verificação usados na ER;
- *Item de Conhecimento Informal*, que agrupa o conhecimento sobre as competências dos indivíduos, as lições aprendidas, que podem ser tanto pontos positivos quanto oportunidades de melhoria, e as comunicações eletrônicas efetuadas pelos participantes do processo de ER. Essas comunicações podem ser efetuadas através de correio eletrônico, compromisso eletrônico, *chat* e fórum.



**Figura 1 – Arquitetura da ferramenta proposta.**

#### 4.1. Os Serviços Gerais de Gerência de Conhecimento em CRETA

Seguindo a proposta de Natali & Falbo [10], os requisitos apresentados na seção 3 são tratados em CRETA principalmente através de serviços gerais de GC, que podem ser utilizados a qualquer momento. CRETA possibilita que o gerente de conhecimento cadastre o conhecimento formal sobre os processos de ER. A exclusão de um item de conhecimento formal pode ser feita pelo gerente de conhecimento, caso o item se torne obsoleto ou sem utilidade. CRETA permite também que o engenheiro de requisitos ou o gerente de projeto registrem os itens de conhecimento informal. No entanto, no caso das lições aprendidas, uma vez que nem todas são úteis à organização, antes que possam ser usadas e disseminadas CRETA possibilita que o gerente de conhecimento possa estruturá-la, adaptá-la e generalizá-la, bem como aprová-la.

Para facilitar o processo de recuperação dos diversos itens de conhecimento armazenados no repositório, esses itens são indexados em CRETA através de palavras-chave. Para apoiar a indexação automática do conhecimento, foi desenvolvido um *Agente Indexador de Conhecimento*. Quando um novo conhecimento formal for cadastrado ou alterado ou uma lição aprendida for aprovada, esse agente analisa seu conteúdo em busca de palavras-chave, previamente cadastradas, de forma a criar um índice do conteúdo da memória organizacional. Esse agente não se comunica com os outros agentes existentes em CRETA; apenas monitora o banco de dados à espera do cadastro de um novo item de conhecimento, para que este seja indexado.

CRETA permite a busca dos diversos itens de conhecimento armazenados na memória organizacional, através da utilização de filtros específicos. Essa busca é uma iniciativa do usuário, pois cabe a ele definir suas necessidades. Para auxiliar a manutenção da memória organizacional, se um item de conhecimento de ER for reutilizado, seu conteúdo deve ser avaliado. CRETA possibilita que o usuário efetue essa avaliação, atribuindo ao item uma classificação que varia de ruim a ótimo e registrando comentários, tais como problemas que enfrentou ao usá-lo e soluções adotadas. Finalmente, CRETA apóia o gerente de conhecimento na manutenção periódica da memória organizacional, tomando por base o feedback fornecido pelos usuários. Itens de conhecimento obsoletos ou não usados podem ser identificados e excluídos.

#### 4.2. Os Serviços Específicos de Gerência de Conhecimento em CRETA

Apesar da maioria dos serviços implementados em CRETA serem gerais, podendo estar presentes em qualquer ferramenta de GC, há uma funcionalidade que é fornecida de forma específica em CRETA: a disseminação de conhecimento, que é particularmente importante quando os usuários não estão motivados a

buscar ou não sabem da existência de conhecimento relevante à tarefa que estão executando [10].

Agentes específicos foram implementados em CRETA para desempenhar um papel ativo na disseminação do conhecimento armazenado. Esses agentes foram modelados utilizando-se a metodologia GAIA [14]. O *Agente Disseminador de Conhecimento* implementado é responsável por monitorar o repositório de conhecimento e quando perceber que um novo conhecimento cadastrado é de interesse de algum desenvolvedor, envia uma solicitação (através do *Agente Gerenciador de Serviço*) ao *Agente Mensageiro* para que este envie um correio eletrônico informando sobre o novo conhecimento disponível. O *Agente Disseminador* baseia-se nas informações das áreas de interesse cadastradas por cada participante para executar seu papel. O *Agente Gerenciador de Serviço* foi definido, em tempo de projeto de arquitetura, para ser responsável pelo controle, registro e manutenção das interações no sistema multiagente. Desse modo, as interações previstas entre os agentes são feitas via o *Agente Gerenciador de Serviço*, que trata de repassar a mensagem para o agente mais apropriado para respondê-la ou para toda a organização.

CRETA permite que o gerente de projeto registre um projeto e defina o processo de execução da ER, utilizando como apoio o conhecimento registrado na organização. Para apoiar essa definição, foi construído o *Agente Assistente do Processo de ER*, que monitora a ação do gerente de projeto durante a definição do processo de ER e, baseado nas informações fornecidas, compara os projetos já cadastrados no ambiente e seus respectivos processos, buscando características semelhantes. Esse agente pode enviar uma solicitação para o *Agente Atendente do Gerente do Projeto* (através do *Agente Gerenciador de Serviços*) fornecendo dados estatísticos e sugestões para prover conhecimento para o gerente de projeto.

## 5. Trabalhos Correlatos

Existem inúmeras pesquisas voltadas para a utilização da GC para apoiar as organizações de software e os processos envolvidos no seu negócio, dentre os quais está o processo da ER [5, 10]. Natali & Falbo [10] discutiram a importância da GC no desenvolvimento de software e apresentaram uma infra-estrutura de GC integrada ao ambiente ODE - *Ontology-based software Development Environment*, tendo por base ontologias.

CRETA possui alguns pontos em comum com esse trabalho. A base da infra-estrutura proposta é a memória organizacional, ao redor da qual os serviços de GC são organizados. No entanto, a memória organizacional de ODE é estruturada com base em ontologias, o que não ocorre com CRETA. ODE e CRETA apóiam os serviços gerais de GC, tais como armazenamento, busca, uso e disseminação de conhecimento e utilizam agentes nessa disseminação. Por outro lado, adotam uma abordagem de GC de forma a estar ativamente integrada ao processo diário de trabalho da equipe. ODE armazena conhecimento sobre o processo de software, enquanto CRETA trata especificamente do conhecimento relacionado à ER. Além disso, CRETA suporta o trabalho cooperativo e utiliza o conhecimento informal envolvido na troca de mensagens como um item de conhecimento do repositório, o que não acontece em ODE.

Dignum & Heinmannsfeld [5] apresentam a ferramenta *Wisdom*, que oferece uma linguagem de alto-nível para representação do conhecimento e possui facilidades para a coleta, codificação, apoio e processamento de conhecimento dentro e fora da organização. Em *Wisdom*, o conhecimento precisa ser representado como tabelas de decisões, sistemas de quadros ou através de Prolog, enquanto que, em CRETA, o conhecimento é representado na forma de objetos armazenados em tabelas de um sistema gerenciador de banco de dados relacional.

## 6. Conclusões

A ER, como um processo intensivo de conhecimento, deve se beneficiar dos recursos oferecidos pela GC para tornar-se mais eficiente, gerando produtos de maior qualidade em um tempo menor. Este artigo apresentou uma aplicação da Gerência de Conhecimento na Engenharia de Requisitos que utiliza agentes para automatizar esse apoio, facilitando a criação e utilização de uma memória organizacional. Em relação aos diversos requisitos apontados na seção 3, e que podem ser aplicados para apoiar a ER, o Quadro 1 apresenta a situação de atendimento desses requisitos na versão atual de CRETA.

No que tange a ER, CRETA permite registrar os principais itens de conhecimento formal sobre o processo

de ER (R1) (R8). Em relação aos itens de conhecimento informal (R2), CRETA trata de lições aprendidas e comunicações efetuadas (R11), que são tratadas separadamente. Seria interessante que CRETA permitisse a composição de pacotes de experiência usando tais comunicações, como sugere a abordagem da Fábrica de Experiência [2]. Com relação à manutenção da memória organizacional (R3) (R6), CRETA oferece mecanismos para tal. Contudo, a exclusão de itens de conhecimento (R3) pode ser aperfeiçoada, de forma a assumir uma postura mais pró-ativa. A indexação em CRETA (R4) (R12) é feita através de palavras-chave e pode ser aperfeiçoada com o uso de ontologias e, por conseguinte, a própria busca (R5) e disseminação (R7). CRETA apóia diversas atividades da ER (R10), dentre elas a definição do processo da ER (R9). Contudo, assistentes inteligentes podem aperfeiçoar esse apoio. Finalmente, no que se refere ao apoio ao trabalho em equipe (R13) (R14), CRETA oferece diversas facilidades, tais como mecanismos que possibilitam que um participante perceba quem são seus parceiros na execução de uma determinada atividade, por exemplo. No entanto, a percepção pode ser aperfeiçoada através da adoção de uma abordagem de feedback compartilhado.

**Quadro 1 – Atendimento dos requisitos de gerência de conhecimento em CRETA.**

Atendimento	Requisito													
	R1	R2	R3	R4	R5	R6	R7	R8	R9	R10	R11	R12	R13	R14
Parcial		x	x	x			x		x	x	x	x	x	x
Total	x				x	x		x						

Maiores experimentações são necessárias para validar a ferramenta, explorar suas limitações, confirmar as conclusões e apontar novas descobertas, pois, atualmente, CRETA está sendo utilizada somente com propósitos acadêmicos. A GC fornece os recursos básicos necessários para o desenvolvimento de uma cultura de trabalho na qual ocorrem constantes trocas de conhecimento e aprendizado. Apesar disso, apenas a existência de ferramentas de apoio à GC não é suficiente para o estabelecimento dessa cultura. Torna-se necessário que os participantes dos projetos estejam conscientes da responsabilidade de cada um em compartilhar o conhecimento que possuem, para enriquecer o conteúdo da memória organizacional.

## Referências

- [1] A. Abecker, A. Bernadi, K. Hinkelmann. "Toward a Technology for Organizational Memories". *IEEE Intelligent Systems*. Nova York, v. 13, n. 3, p. 40-48, mai./jun. 1998.
- [2] V. R. Basili, M. Lindvall, P. Costa. "Implementing the Experience Factory Concepts as a Set of Experience Bases". In: PROCEEDINGS OF THE INTERNATIONAL CONFERENCE ON SOFTWARE ENGINEERING & KNOWLEDGE ENGINEERING (SEKE'01), 13., 2001., Buenos Aires, Argentina.
- [3] D. Chaffey. *Groupware, workflow and intranets: reengineering the enterprise with collaborative software*. Boston: Digital Press, 1998.
- [4] T. Davenport, L. Prusak. *Working Knowledge: How Organizations Manage What They Know*. Boston, USA: Harvard Business School Press, 1998.
- [5] V. Dignum, K. Heinmannsfeld. "Knowledge Management for Requirements Engineering". In: PROCEEDINGS OF THE WORKSHOP ON KNOWLEDGE ACQUISITION (KAW'99), 12., 1999., Banff, Canada.
- [6] R. A. Falbo. *Integração de Conhecimento em um Ambiente de Desenvolvimento de Software*. 1998. 205 p. Tese (Doutorado em Ciências em Engenharia de Sistemas e Computação) - Programa de Pós-Graduação de Engenharia, Universidade Federal do Rio de Janeiro, Rio de Janeiro, 1998.
- [7] G. Fischer, J. Ostwald. "Knowledge management: problems, promises, realities, and challenges". *IEEE Intelligent Systems*. New York, v. 16, n. 1, p. 60-72, jan./fev. 2001.
- [8] G. Kotonya, I. Sommerville. *Requirements engineering: processes and techniques*. Chichester, Inglaterra: John Wiley, 1998.
- [9] T. Kouwenhoven. "Reengineering for learning". *SIGGROUP Bulletin*. [S. L.], v. 19, n. 1, p. 39-45, abr. 1998.
- [10] A. C. C. Natali, R. A. Falbo. "Gerência de Conhecimento em ODE". XVII SIMPÓSIO BRASILEIRO DE ENGENHARIA DE SOFTWARE (SBES2003), 17., 2003., Manaus, AM.
- [11] D. E. O'Leary. "Enterprise Knowledge Management". *IEEE Computer*. N. York, v. 31, n. 3, p.54-61, mar. 1998.
- [12] I. Rus, M. Lindvall, S. S. Sinha. *Knowledge management in software engineering: a state-of-the-art report*. Rome, NY, EUA: Departamento de Defesa Americano, 2001.
- [13] D. F. Togneri, R. A. Falbo, C. S. Menezes. "Supporting Cooperative Requirements Engineering with an Automated Tool". In: WORKSHOP ON REQUIREMENTS ENGINEERING(WER02), 5., 2002, Valência, Espanha.
- [14] F. Zambonelli, N. R. Jennings, M. Wooldridge. "Developing Multiagent Systems: The Gaia Methodology". *ACM Transactions on Software Engineering Methodology*, v. 12, n. 3, p. 317-370, julho 2003.