

# ReqODE: Uma Ferramenta de Apoio à Engenharia de Requisitos Integrada ao Ambiente ODE

Aline Freitas Martins, Julio Cesar Nardi, Ricardo de Almeida Falbo

Departamento de Informática – Universidade Federal do Espírito Santo (UFES)  
Av. Fernando Ferrari s/n, Campus de Goiabeiras – 29.060-900 – Vitória – ES – Brasil

{alinefmart, julionardi}@yahoo.com.br, falbo@inf.ufes.br

**Abstract.** *One of the main measures of the success of a software product is the degree in which it reaches its goals and requirements. In general, Requirements Engineering (RE) is the process of identifying the requirements involved, and documenting and managing them in an appropriate way for later design and implementation. This paper presents ReqODE, a tool that partially supports the Requirements Engineering process, providing functionalities for controlling and tracing requirements. ReqODE is integrated to ODE, an ontology-based software development environment, and it was built based on a requirements ontology.*

**Resumo.** *Uma das principais medidas do sucesso de um produto de software é o grau no qual ele atende aos objetivos e requisitos para os quais foi construído. De forma geral, a Engenharia de Requisitos de Software é o processo de identificar todos os requisitos envolvidos, documentá-los e gerenciá-los de forma apropriada para posterior projeto e implementação. Este artigo apresenta uma ferramenta de apoio à Engenharia de Requisitos, que provê as funcionalidades de controle e rastreabilidade de requisitos, denominada ReqODE. ReqODE está integrada ao ambiente de desenvolvimento ODE e foi construída tendo por base uma ontologia de requisitos.*

## 1. Introdução

O processo de desenvolvimento de software tem início pela identificação dos requisitos, ou seja, dos serviços e das funções que o sistema deverá prover, bem como de suas restrições e propriedades importantes. A medida de sucesso de um software é dada em grande parte pelo grau que ele atende a esses requisitos. Para tanto, é importante não só um bom levantamento dos requisitos, mas também o acompanhamento das atividades relacionadas a estes durante todo o ciclo de desenvolvimento do software.

O processo de Engenharia de Requisitos (ER) envolve as atividades relativas ao tratamento dos requisitos de um sistema, incluindo: levantamento, modelagem, negociação, documentação, validação e gerência de requisitos. Dentre os problemas de um processo de ER ineficiente, podem-se citar [Kotonia e Sommerville 1998]: (i) requisitos inconsistentes, (ii) produto final com custo maior do que o esperado, (iii) software instável e com altos custos de manutenção e (iv) clientes insatisfeitos.

Por ser um processo complexo e de tamanha importância, é importante prover ferramentas automatizadas para apoiar a ER. Além disso, para melhor gerenciar os requisitos de um projeto, é importante que uma ferramenta de apoio à ER esteja integrada às demais ferramentas utilizadas no desenvolvimento de software. Uma forma de se conseguir isso é por meio de sua integração em um Ambiente de Desenvolvimento de Software (ADS).

Este trabalho apresenta ReqODE, a ferramenta de apoio à ER de ODE (*Ontology-based Development Environment*) [Falbo et al. 2003], um ADS construído tomando por base ontologias. Uma vez que ODE é baseado em ontologias, ReqODE foi desenvolvida com base em uma ontologia de requisitos de software [Nardi e Falbo 2006]. Uma ontologia procura capturar a conceituação básica acerca de um universo de discurso, descrevendo os conceitos e relações desse universo segundo o princípio do comprometimento ontológico mínimo. Assim, por tratar de aspectos básicos de requisitos, apontados consensualmente como importantes para a ER, espera-se que ReqODE possa ser útil para uma vasta gama de organizações. Por fim, o fato de estar integrada a ODE e, portanto, a outras ferramentas desse ambiente, facilita que ReqODE proveja funcionalidades de apoio à maior parte das atividades do processo de ER.

## **2. Requisitos de Software e Engenharia de Requisitos**

Requisitos de software são especificações de serviços que o sistema deve prover, restrições e conhecimentos necessários para desenvolvê-lo [Kotonia e Sommerville 1998]. Uma vez capturados, os requisitos de software devem ser modelados, documentados, validados e acompanhados. Neste processo, as propriedades de um requisito e os relacionamentos com outros elementos do processo de software são definidos e alterados. Portanto, definir e entender as propriedades e relações em torno de um requisito são aspectos essenciais na condução do processo de ER.

Requisitos, normalmente, são classificados segundo alguma categorização, definida de acordo com as práticas de cada organização. Ainda que essa categorização seja particular, há um consenso de que requisitos podem ser funcionais (representam o que o sistema deve fazer, suas funções) e não funcionais (representam os atributos do sistema enquanto software constituído) [Sommerville 2003].

Acerca de um requisito, é interessante saber: a sua origem, prioridade, os interessados (*stakeholders*) e responsáveis. De posse desses elementos, as atividades de análise, negociação e gerência de requisitos são facilitadas, pois é possível identificar os envolvidos e também o ponto de partida de um requisito [Kotonia e Sommerville 1998].

No centro da atividade de gerência de requisitos – além do controle de alteração e de versão de um requisito – está a rastreabilidade, isto é, a habilidade de se acompanhar a vida de um requisito em ambas as direções do processo de software e durante todo o ciclo de vida. A rastreabilidade de requisitos só é possível se houver ligações explícitas entre requisitos e outros elementos do processo de software. Dessa forma, a identificação da composição de requisitos, das dependências entre requisitos, de requisitos conflitantes, da origem dos requisitos e de seus interessados, além da identificação de em qual artefato (documento, diagrama etc) produzido durante o processo de software um requisito é tratado, é de fundamental importância para que a rastreabilidade seja implementada [Kotonia e Sommerville 1998].

Pela breve exposição anterior, é possível notar o quanto é complexo o processo da ER. Dada essa complexidade, construir ferramentas de apoio às atividades desse processo é fundamental para a efetiva execução do mesmo. Além disso, alguns aspectos da ER, tal como a gerência de requisitos, são praticamente inviáveis de serem tratados sem o apoio de ferramentas.

### **3. ReqODE: A Ferramenta de Apoio à Engenharia de Requisitos de ODE**

ReqODE é a ferramenta de apoio ao processo ER em ODE, permitindo tratar requisitos desde o levantamento, quando são capturados os requisitos, até a gerência de requisitos.

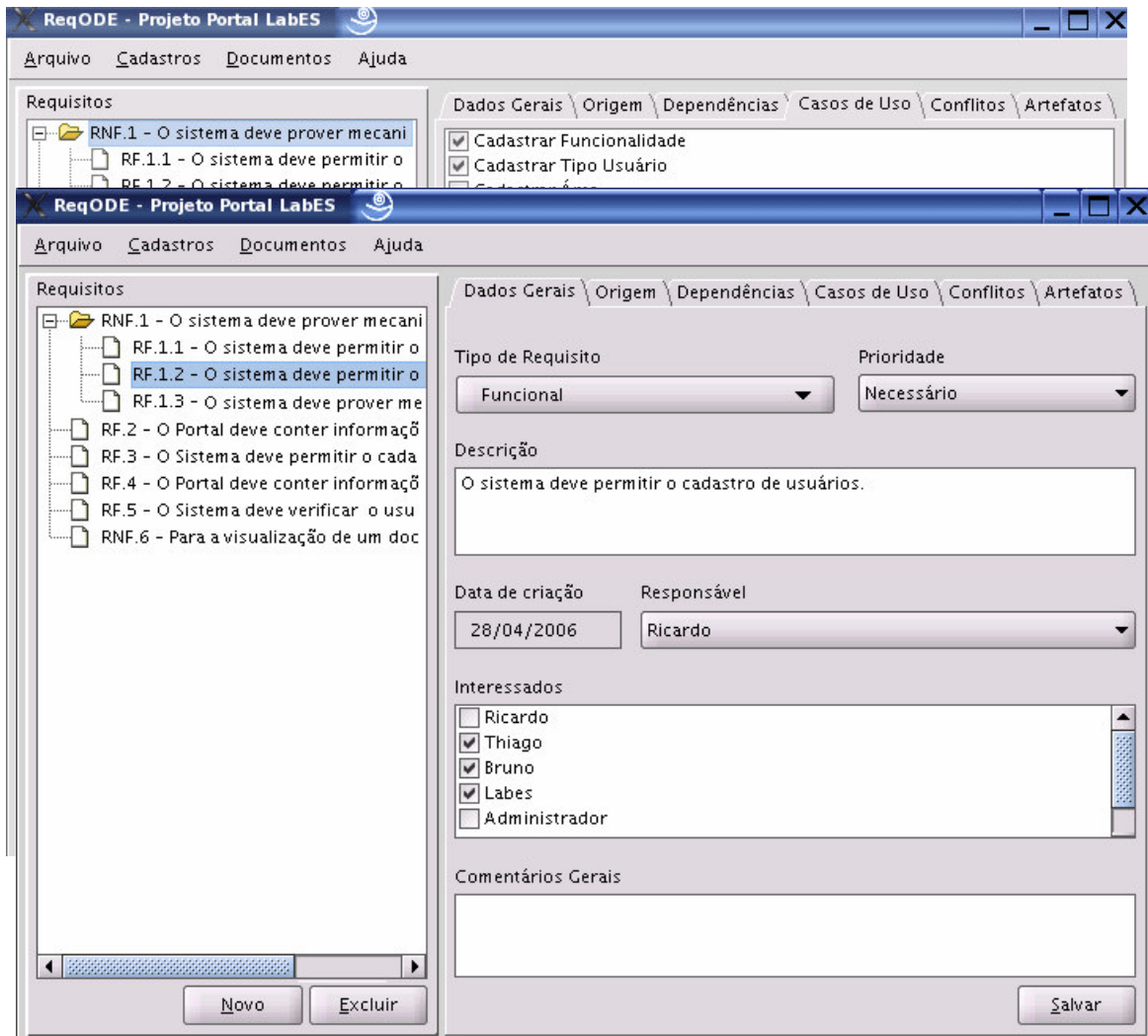
Conforme citado na seção 1, ReqODE foi construída tomando por base uma ontologia de requisitos e, portanto, grande parte de suas tarefas é realizada utilizando e respeitando os modelos e restrições impostos por essa ontologia. Além disso, um esquema de anotação conceitual é utilizado, com classes sendo definidas em três níveis. O nível ontológico (pacote *Ontologia*) é responsável pela descrição das ontologias em si. O meta-nível (pacote *Conhecimento*) abriga as classes que descrevem o conhecimento organizacional sobre um domínio de discurso. Já o nível base (pacote *Controle*) define as classes responsáveis por implementar as aplicações no contexto do ambiente [Falbo et al. 2005]. Todos esses pacotes fazem parte da camada de domínio do problema (CDP) da arquitetura de software de ODE. Essa arquitetura é composta, ainda, por mais quatro componentes, a saber: componente de interação humana (CIH), que, como o próprio nome indica, é responsável pelas classes de interface com o usuário; componente de gerência de tarefas (CGT), que é responsável pelo controle das funcionalidades (classes de aplicação), componente de gerência de dados (CGD), responsável pela persistência de objetos em um banco de dados relacional e componente de controle de interação (CCI), responsável por isolar as classes do CIH das classes de aplicação (CGT) e de domínio do problema (CDP), buscando compatibilidade com o padrão Modelo-Visão-Controlador (MVC) [Sun 2002].

Ao levantar um requisito e registrá-lo em ReqODE, o engenheiro de requisitos deve informar uma descrição e um tipo para o requisito. Este último é um objeto do meta-nível de ODE e é cadastrado no sistema através da funcionalidade de Cadastro de Conhecimento Organizacional de ODE [Natali e Falbo 2003], permitindo à organização manter seu próprio padrão de classificação dos requisitos.

No momento em que um requisito é criado, um identificador único é gerado para ele, sendo a criação desse identificador dependente do tipo escolhido para o requisito. Posteriormente, um requisito pode ser refinado, o que envolve a possibilidade de decompô-lo em outros requisitos e a definição de aspectos relacionados à sua origem, tais como razões para a sua criação e seu contexto de criação, dentre outros. Por fim, relações de dependência e de conflito entre requisitos podem ser informadas, permitindo apoiar atividades de gerência e negociação.

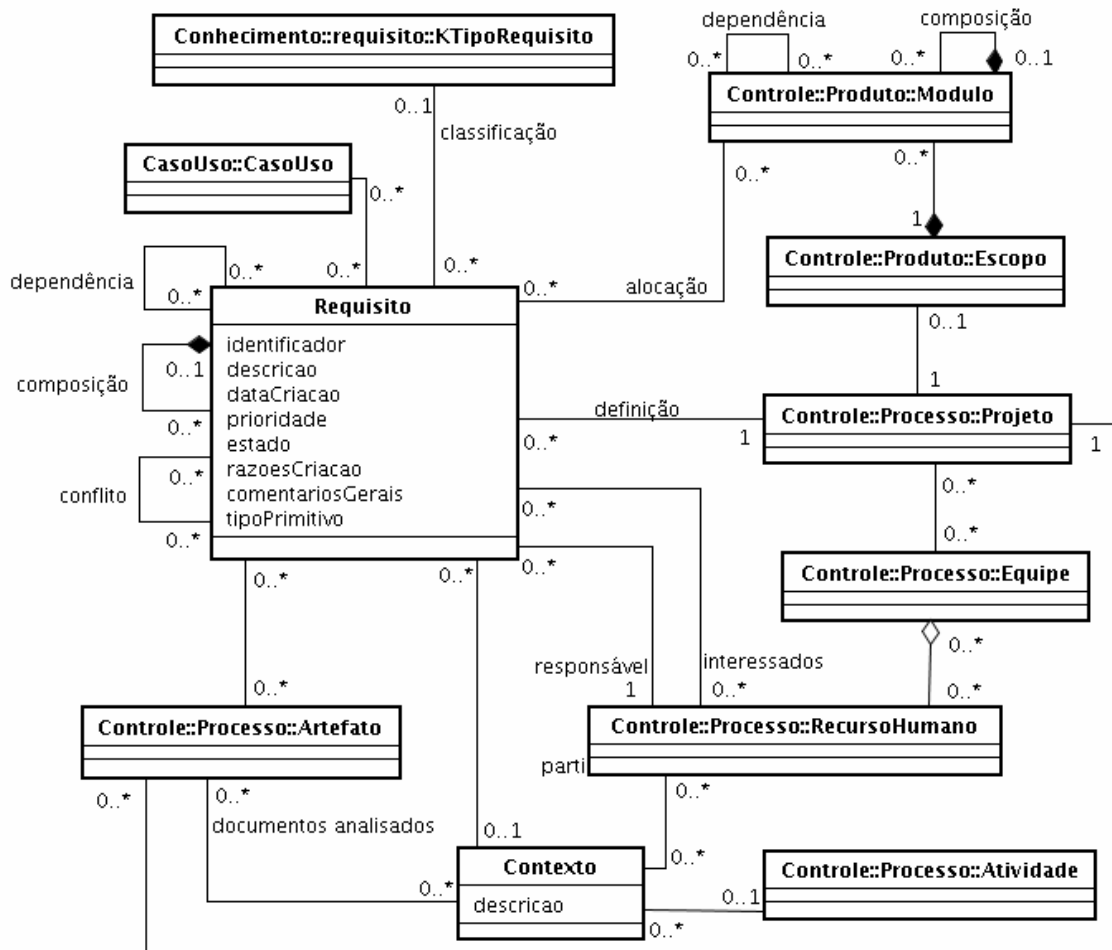
A Figura 1 ilustra uma utilização de ReqODE. Na árvore à esquerda, são apresentados os requisitos funcionais (RF) e não funcionais (RNF). Requisitos derivados de um requisito mais geral (decomposição de requisitos) são apresentados aninhados, como se pode perceber pelo RNF1 “O sistema deve prover mecanismos de controle de acesso”, decomposto em outros três. À direita, diversos painéis podem ser selecionados pela abas no alto. Na Figura 1 dois desses painéis são mostrados. O

primeiro, parcialmente oculto, permite associar requisitos a casos de uso. O segundo apresenta o painel de Dados Gerais de um requisito, no qual são informados o tipo do requisito, sua prioridade, descrição, data de criação, responsável, interessados e comentários gerais.



**Figura 1. Descrição de Requisitos em ReqODE.**

A Figura 2 mostra um diagrama de classes parcial do componente de domínio do problema de ReqODE. Esse diagrama aponta que requisitos podem ser decompostos em outros requisitos, bem como podem ser dependentes ou estar em conflito com outros requisitos. Visando à rastreabilidade, requisitos podem ser associados a casos de uso e artefatos produzidos no contexto do projeto. Além disso, o contexto no qual o mesmo foi levantado (o que inclui a atividade do processo de software, documentos analisados e participantes que atuaram em sua identificação) pode ser registrado. Por fim, para facilitar o gerenciamento do projeto, requisitos são alocados a módulos que indicam o escopo do projeto e recursos humanos são indicados como responsáveis e interessados. A partir desses elementos, um Documento de Rastreabilidade é gerado e pode ser utilizado para gerenciar mudanças. Isto é, se alguma alteração em um requisito for necessária, é possível identificar que elementos provavelmente também sofrerão mudanças, possibilitando a análise do impacto que essa alteração vai causar



**Figura 2. Diagrama de Classes (Parcial) do CDP de ReqODE**

Vale destacar algumas vantagens obtidas do fato de ReqODE estar integrada a ODE:

- É possível relacionar requisitos a interessados e a seus responsáveis, recursos humanos cadastrados no ambiente por meio das ferramentas de Gerência de Recursos Humanos e de Alocação de Recursos ao projeto em questão.
- Requisitos podem ser associados a casos de uso, registrados em ODE por meio da ferramenta OODE de modelagem de objetos usando UML.
- Requisitos podem ser associados a artefatos internos, produzidos pelas demais ferramentas de ODE, ou externos a ODE, cadastrados no ambiente por meio da ferramenta de apoio ao planejamento da documentação. Este é um aspecto chave para se tratar a rastreabilidade.

No que se refere à implementação, ReqODE foi desenvolvida utilizando-se a linguagem Java, priorizando a portabilidade, com interfaces *Swing*. A persistência dos objetos é feita no banco de dados relacional PostgreSQL, utilizando-se o *framework* de mapeamento objeto-relacional Hibernate. Foi uma premissa do desenvolvimento que as tecnologias utilizadas fossem livres, pois o grupo de desenvolvimento de ODE tem a intenção de disponibilizar o ambiente como software livre em um futuro próximo.

#### 4. Considerações Finais

Neste trabalho foi apresentada ReqODE, a ferramenta de ODE de apoio à ER. Ter esta ferramenta integrada a um ambiente de desenvolvimento trouxe diversas vantagens, sobretudo para a gerência de requisitos, pois é possível identificar outros elementos do processo de software associados aos requisitos. Assim, se permite uma análise de impacto mais eficiente das mudanças ocorridas nos requisitos.

ReqODE foi integrada à nova versão do ambiente ODE em março de 2006. Esta versão foi disponibilizada para uma organização de software parceira em abril de 2006 e assim, espera-se evoluí-la com base no feedback dado por essa organização. Como trabalho de melhoria já em andamento, está a oferta de serviços de gerência de conhecimento voltados especificamente para a ER, o que inclui o apoio à reutilização de modelos de domínio na forma de ontologias de domínio, padrões de análise e modelos de objetos desenvolvidos em outros projetos.

#### Agradecimentos

Este trabalho foi realizado com o apoio do CNPq e da CAPES, entidades do Governo Brasileiro dedicadas ao desenvolvimento científico e tecnológico, da FAPES, Fundação de Apoio à Ciência e Tecnologia do Espírito Santo, e das empresas VixTeam e Projeta, parceiras que têm financiado o projeto e dado feedback de sua aplicação a casos reais.

#### Referências

- Falbo, R. A., Natali, A. C. C., Mian, P.G., Bertollo, G., Ruy, F.B. (2003) “ODE: Ontology-based software Development Environment”, In: Memórias de IX Congreso Argentino de Ciencias de la Computación, p. 1124-1135, La Plata, Argentina.
- Falbo, R.A., Ruy, F.B., Dal Moro, R. (2005) “Using Ontologies to Add Semantics to a Software Engineering Environment” Proc. of the 17th International Conference on Software Engineering and Knowledge Engineering, Taipei, China.
- Kotonia, G. and Sommerville, I. (1998) *Requirements engineering: processes and techniques*. Chichester, England: John Wiley.
- Nardi, J.C., Falbo, R.A. (2006) “Uma Ontologia de Requisitos de Software”, In: Memórias de 9º Workshop Iberoamericano de Ingeniería de Requisitos y Ambientes de Software, La Plata, Argentina.
- Natali, A.C.C., Falbo, R.A. (2003) “Gerência de Conhecimento em ODE”, In: Anais do XVII Simpósio Brasileiro de Engenharia de Software, p. 270-285, Manaus, Brasil.
- Sommerville, I. (2003), *Engenharia de Software*, 6ª Edição, Addison Wesley.
- Sun Developer Network, “Model-view-controller”. Disponível em: <http://java.sun.com/blueprints/patterns/MVC-detailed.html>>. Acesso em: 11 jul. 2002.