

# SIGERAR: Uma Ferramenta para Gerenciamento de Requisitos

José Inácio De Grande, Luiz Eduardo G. Martins  
Programa de Mestrado em Ciência da Computação  
Faculdade de Ciências Exatas e da Natureza  
UNIMEP - Universidade Metodista de Piracicaba  
{[ji-grande@uol.com.br](mailto:ji-grande@uol.com.br) [lgmartin@unimep.br](mailto:lgmartin@unimep.br)}

## Resumo

Atualmente, a utilização da Engenharia de Requisitos é um dos caminhos mais seguros para se obter qualidade no desenvolvimento e manutenção de sistemas de software, pois esta prática diminui sensivelmente os erros, falhas e ambigüidades do produto final a ser entregue. Com o passar do tempo, mudanças ocorrem nos requisitos devido a diversos fatores como erros, inconsistências, problemas organizacionais, evolução do conhecimento dos stakeholders<sup>1</sup>, alterações legais, etc., exigindo um grande esforço das empresas para o controle e gerenciamento dos mesmos. A proposta deste trabalho é apresentar uma ferramenta automatizada para gerenciamento de requisitos, chamada SIGERAR. A ferramenta coleta, armazena e mantém os requisitos, gerenciando as mudanças e promovendo rastreabilidade entre os requisitos e entre os documentos de requisitos. A contribuição deste trabalho é oferecer aos desenvolvedores de Software uma ferramenta de gerenciamento de requisitos, de uso livre e fácil instalação, configuração e operação, aderente a todo o ciclo de vida do software, proporcionando controle e análise de riscos, impactos e custos de mudanças nos requisitos.

**Palavras-Chave:** Engenharia de Requisitos, Gerenciamento de Requisitos, Ferramenta Automatizada, Rastreabilidade.

## 1. Introdução

O processo de Engenharia de Requisitos tem como principais objetivos a aquisição de conhecimentos das regras de negócios e verificação das necessidades do cliente, de forma a obter uma especificação não ambígua e completa dos requisitos de *software*, com o intuito de minimizar os erros, inadequações e falhas no produto final a ser entregue ao cliente.

O Gerenciamento de Requisitos é o processo de compreender e controlar as mudanças que ocorrem nos

requisitos, por força da evolução dos mesmos, refletindo as alterações que sofrem ao longo do tempo, no ambiente do sistema e nos objetivos da organização [7]. Além da Análise e Especificação, o Gerenciamento de Requisitos é de fundamental importância no processo da Engenharia de Requisitos, pois organiza o controle das mudanças, permitindo subsídios para a análise de impacto e custos em tempo e dinheiro, que estas trarão para a organização.

Este trabalho tem por objetivo apresentar uma ferramenta automatizada para documentação e gerenciamento de requisitos, durante todo o ciclo de vida do *software*. A ferramenta tem capacidade para coletar, armazenar e manter os requisitos acordados entre os *stakeholders*, gerenciar as mudanças ocorridas nos requisitos, em razão de sua natural evolução e rastrear os relacionamentos entre requisitos e entre documentos de requisitos. A motivação para a escolha do tema advém da escassez de ferramentas brasileiras disponíveis no mercado para gerenciamento de requisitos e das dificuldades que uma ferramenta internacional pode trazer, como: custos de aquisição, custos de treinamento, dificuldades dos *stakeholders* com línguas estrangeiras, falta de representantes no Brasil, falta de especialistas do produto no mercado, entre outros.

A contribuição deste trabalho é oferecer à comunidade de Engenharia de Software uma ferramenta de uso livre e fácil instalação, configuração e operação para documentação e gerenciamento de requisitos. A ferramenta apresentada conta com muitos dos recursos disponíveis nas ferramentas comerciais conhecidas, como controle de acesso e permissões, controle de versões (históricos), glossários, notificações e fórum de discussões, entre outros, e apresenta alguns pontos diferenciais. Um importante diferencial da ferramenta é o tratamento da rastreabilidade dos requisitos no processo de alteração, onde a ferramenta não somente demonstra os requisitos dependentes da modificação do requisito origem, mas exige que os responsáveis pelos requisitos (origem e dependentes) analisem e atribuam valores de risco, importância, impacto, prioridade e custo de cada um dos requisitos envolvidos. É importante ressaltar que um requisito dependente também tem sua própria matriz de dependência, assim sendo, temos que dar a ele o tratamento como se requisito origem fosse (efeito recursivo), tratando todos os seus dependentes até que o ciclo se feche. Desta

---

<sup>1</sup> “O termo *stakeholder* é utilizado para se referir a qualquer pessoa que terá alguma influência direta ou indireta sobre os requisitos do sistema” [6].

forma, a ferramenta garante que todos os requisitos envolvidos sejam rastreados, analisados e tratados, produzindo informações de vital importância ao Gerente do Projeto que poderá analisar todo o contexto do impacto e custos da alteração e com estes subsídios, tomará a decisão de aprovar ou rejeitar a proposta de alteração do requisito. Espera-se que esta ferramenta traga benefícios às organizações que venham adotá-la, pois além de não haver custos com aquisição da ferramenta e do banco de dados, irá proporcionar controle sobre as mudanças ocorridas nos requisitos e análise do risco, impacto e custos destas mudanças.

O restante deste trabalho está organizado da seguinte forma: a seção dois discorre sobre Gerenciamento de Requisitos que é a base do trabalho desenvolvido. Na seção três é registrada o desenvolvimento da ferramenta, composto dos processos de elicitação e especificação dos requisitos da ferramenta, modelagem lógica da ferramenta e funcionalidades da ferramenta. Na seção quatro é apresentada uma análise de viabilidade de uso da ferramenta, através de um estudo de caso. A seção cinco apresenta as conclusões do trabalho e propostas de trabalhos futuros.

## 2. Gerenciamento de Requisitos

O Gerenciamento de Requisitos é o processo de compreender e controlar as mudanças nos requisitos de sistemas e ocorre em conjunto com outros processos da Engenharia de Requisitos [7]. Os principais objetivos do Gerenciamento de Requisitos segundo Kotonya e Sommerville [3] são:

- Gerenciar mudanças nos requisitos acordados;
- Gerenciar o relacionamento entre requisitos;
- Gerenciar as dependências entre os documentos de requisitos e outros documentos produzidos no processo de Engenharia de *Software*.

Lam e outros [4] identificam três razões para gerenciamento de requisitos:

- Muitos sistemas são entregues incrementalmente. Entre cada entrega incremental, mudanças nos requisitos são estabelecidas e incorporadas no próximo incremento;
- Tipicamente requisitos mutáveis são os principais causadores de manutenções de *software* e atividades de reengenharia;
- Muitas organizações têm sistemas legados que são críticos e sustentam operações comerciais. Substituir totalmente ou recriar tais sistemas nem sempre é possível e necessitam evoluir para que a empresa sobreviva e permaneça competitiva.

Segundo Sommerville [7] os requisitos devem evoluir a fim de refletir as mudanças que ocorrem ao longo do tempo, no ambiente do sistema e nos objetivos

da empresa. Para tanto, a partir da perspectiva de evolução, divide os requisitos em duas classes: Requisitos permanentes ou estáveis e Requisitos voláteis.

Os fatores que mais contribuem para as mudanças de requisitos segundo Kotonya e Sommerville [3] são erros, conflitos e inconsistências nos requisitos, evolução do conhecimento dos clientes e usuários do sistema, problemas técnicos, de prazo e de custos, mudanças nas prioridades dos clientes, ambientais e organizacionais.

### 2.1. Gerenciamento de Mudanças

O gerenciamento de mudanças está relacionado à política de uso de procedimentos, processos e padrões que serão utilizados para gerenciar as mudanças nos requisitos do sistema [3]. Estas políticas incluem:

- O processo de solicitação de mudanças e as informações necessárias para processá-la;
- O processo usado para analisar o impacto e custo das mudanças e informações associadas à rastreabilidade;
- Definição dos membros da organização que formalmente consideram as solicitações de mudanças;
- O suporte de *software* necessário para o controle do processo de mudanças.

O Processo de Gerenciamento de Mudanças nos Requisitos consiste em um conjunto de atividades para documentar, reportar, analisar, definir custos e implementar mudanças de um conjunto de requisitos, conforme ilustra a Figura 1.



**Figura 1 – Processo de Gerenciamento de Mudanças nos Requisitos [3]**

Os três estágios representam:

- Algum problema de requisitos é identificado. Isto pode ser oriundo de uma análise do documento de requisitos, de novas necessidades dos *stakeholders*, ou problemas operacionais com o sistema. Os requisitos são analisados usando informações do problema e mudanças nos requisitos são propostas;
- As mudanças propostas são analisadas. Verificam-se quantos requisitos e, se necessário, os componentes do sistema, que serão afetados pelas mudanças, calculando-se de forma aproximada o custo em tempo e dinheiro;

- As mudanças são implementadas. Um conjunto de alterações ou uma nova versão do documento de requisitos são produzidos.

O Processo de Análise de Mudanças e Custo é composto de seis estágios, conforme apresenta a Figura 2 a seguir.

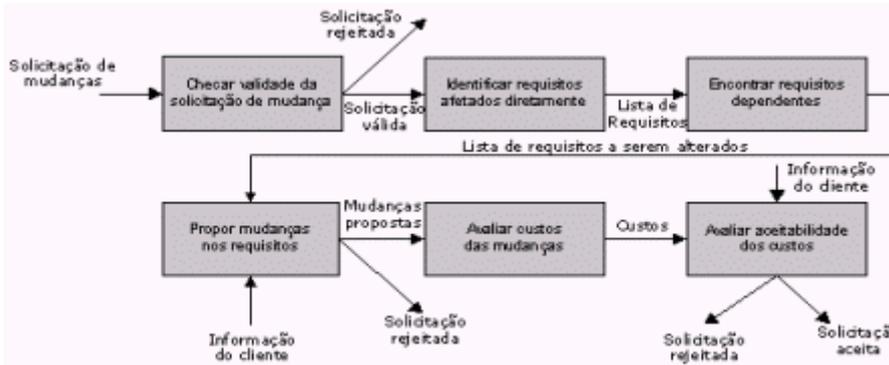


Figura 2 – Processo de Análise de Mudanças [3]

Os seis estágios desse processo contemplam:

- A requisição de mudanças é verificada quanto à sua validade, pois os *stakeholders* podem não entender os requisitos e sugerem mudanças desnecessárias;
- Os requisitos afetados diretamente pelas mudanças são descobertos;
- Informações de rastreabilidade são usadas para encontrar os requisitos dependentes que podem ser afetados pelas mudanças;
- As mudanças que podem ser feitas para os requisitos são propostas;
- Os custos das mudanças são estimados;
- São efetuadas negociações com os clientes para verificar se os custos das mudanças propostas são aceitáveis.

## 2.2. Rastreabilidade

Segundo Gotel e Finkelstein [2] “rastreabilidade de requisitos é a habilidade de descrever e acompanhar a vida de um requisito em ambas as direções do processo de *software* (do planejamento do negócio à especificação do projeto), idealmente durante todo o seu ciclo de vida”. Kotonya e Sommerville [3], afirmam “que um requisito pode ser rastreado se é possível determinar quem sugeriu o requisito, porque o requisito existe, a quais outros requisitos ele está relacionado e como ele está relacionado com outras informações, como artefatos de projeto, implementação e documentação de usuário. O rastreamento também

permite encontrar outros requisitos que podem ser afetados quando uma mudança é solicitada”. Pinheiro [5] entende rastreabilidade de requisitos como a habilidade de definir, capturar e acompanhar os rastros deixados pelos requisitos nos outros elementos do ambiente de desenvolvimento de *software* e os rastros deixados por esses elementos nos requisitos.

Ramesh [6] diz que a rastreabilidade de requisitos é usada para capturar o relacionamento entre requisitos, projeto e implementação de um sistema. Assim, todos os componentes do sistema (*hardware*, *software*, pessoas, manuais, políticas e procedimentos) criados nos vários estágios do processo de desenvolvimento, são ligados aos requisitos.

## 3. Desenvolvimento da Ferramenta

A ferramenta automatizada para gerenciamento de requisitos foi desenvolvida com o objetivo de coletar, armazenar e manter os requisitos acordados, durante todo o ciclo de vida do *software*, gerenciando as mudanças ocorridas nos requisitos, permitindo rastrear os relacionamentos entre os requisitos e entre os requisitos e documentos produzidos no processo de desenvolvimento do sistema.

Como um dos principais benefícios da ferramenta é sua livre distribuição e uso, seu desenvolvimento foi direcionado e orientado a ferramentas da mesma abordagem, de forma a propiciar mais facilmente melhorias futuras.

Desta forma, a ferramenta é operada via interface Web e foi desenvolvida em linguagem *Java* com JSP (*Java Server Pages*) e com o SGBD (Sistema Gerenciador de Banco de Dados) *Firebird* 1.5.

A Figura 3 a seguir representa a estrutura para aplicações Web em 3 camadas onde a ferramenta pode ser executada.

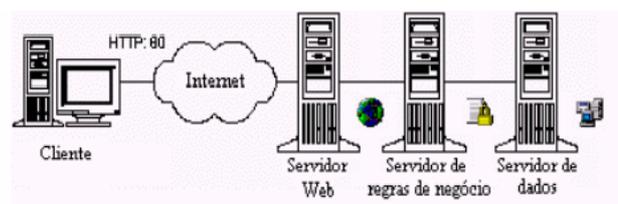


Figura 3 – Ambiente de Operação da Ferramenta em 3 Camadas

- 1ª camada – composta por Servidor Web (*Apache Server*) que gerencia as requisições vindas da Internet;
- 2ª camada – composta pelo Servidor de regras de negócio (*Jakarta Tomcat*) que gerencia o acesso às informações. As aplicações são *componentizadas* em classes *JAVA* que realizam todas as rotinas da ferramenta;
- 3ª camada – composta pelo Servidor de Dados (SGBD *Firebird*), que abriga a base de dados que alimenta todo o sistema.

### **Elicitação e Especificação dos Requisitos da Ferramenta**

Para elicitación dos requisitos da ferramenta foram efetuadas reuniões JAD (*Joint Application Design*) e entrevistas com a equipe de analistas, coordenadores e gerentes estabelecendo os limites do sistema, os Requisitos Funcionais e os Requisitos Não-Funcionais. Para especificar os requisitos da ferramenta foram utilizados vários diagramas da UML (*Unified Modeling Language*) [1].

### **Modelagem Lógica da Ferramenta**

A modelagem lógica da ferramenta foi especificada através do diagrama de classes (UML) representado na Figura 4, e permite uma visão geral das classes e relacionamentos que a compõe. A classe Projeto apresenta os dados relativos aos projetos que serão gerenciados na organização. Esta classe se relaciona com as classes Glossário e Termos, que contém os termos mais usuais da organização. A classe Usuário contém os atributos dos *stakeholders* do sistema e a Classe Departamento identifica o vínculo organizacional do usuário. A classe UsuárioProjeto identifica os usuários envolvidos nos projetos, com alçada pré-determinada pelo Gerente de cada Projeto. A classe Alçada determina os níveis de autorização/restricção dos usuários às determinadas funções da ferramenta.

A classe Requisito identifica os requisitos de cada projeto. A classe RequisitoDependente trata a rastreabilidade do requisito registrando a dependência entre os requisitos. O processo de gerenciamento está centrado na classe VersãoRequisito, que contém a descrição da alteração solicitada, motivo da alteração, solicitante, responsável, descrição, prioridade, risco, importância, custo, impacto e situação da versão. A situação representa o *status* atual da versão do requisito, que poderá ser ‘proposta’, ‘em análise’, ‘aprovada’, ‘em desenvolvimento’ e ‘implementada’. A classe AlteraçãoVersão, apóia e complementa a classe VersãoRequisito, com dados de controle das versões atuais, anterior e alteração origem. A classe Documentos permite ao analista relacionar todos os documentos envolvidos no processo de alteração, permitindo a rastreabilidade destes. As classes Motivo, Risco, Impacto, Prioridade, Importância e Volatilidade representam elementos de apoio ao Gerenciamento da Versão de Requisitos e contém atributos para mensurar os riscos envolvidos nas alterações dos requisitos e de seus requisitos dependentes.

### **Funcionalidades da Ferramenta**

No desenvolvimento da Ferramenta, buscou-se contemplar todas as funcionalidades elicitadas e acordadas com os envolvidos, através dos módulos abaixo descritos.

### **Módulo Administração do Sistema**

Expressa a fase inicial da ferramenta onde são criados e geridos os cadastros: Projetos, Usuários, Glossários, Volatilidade, Risco, Importância, Impacto, Motivo, Tipo e Departamento, que serão básicos para todos os projetos da organização. O Administrador será o responsável pela formatação e parametrização dos cadastros acima mencionados e, entre outras funções, executa a alocação dos Glossários aos Projetos e a atribuição de Gerentes aos Projetos.

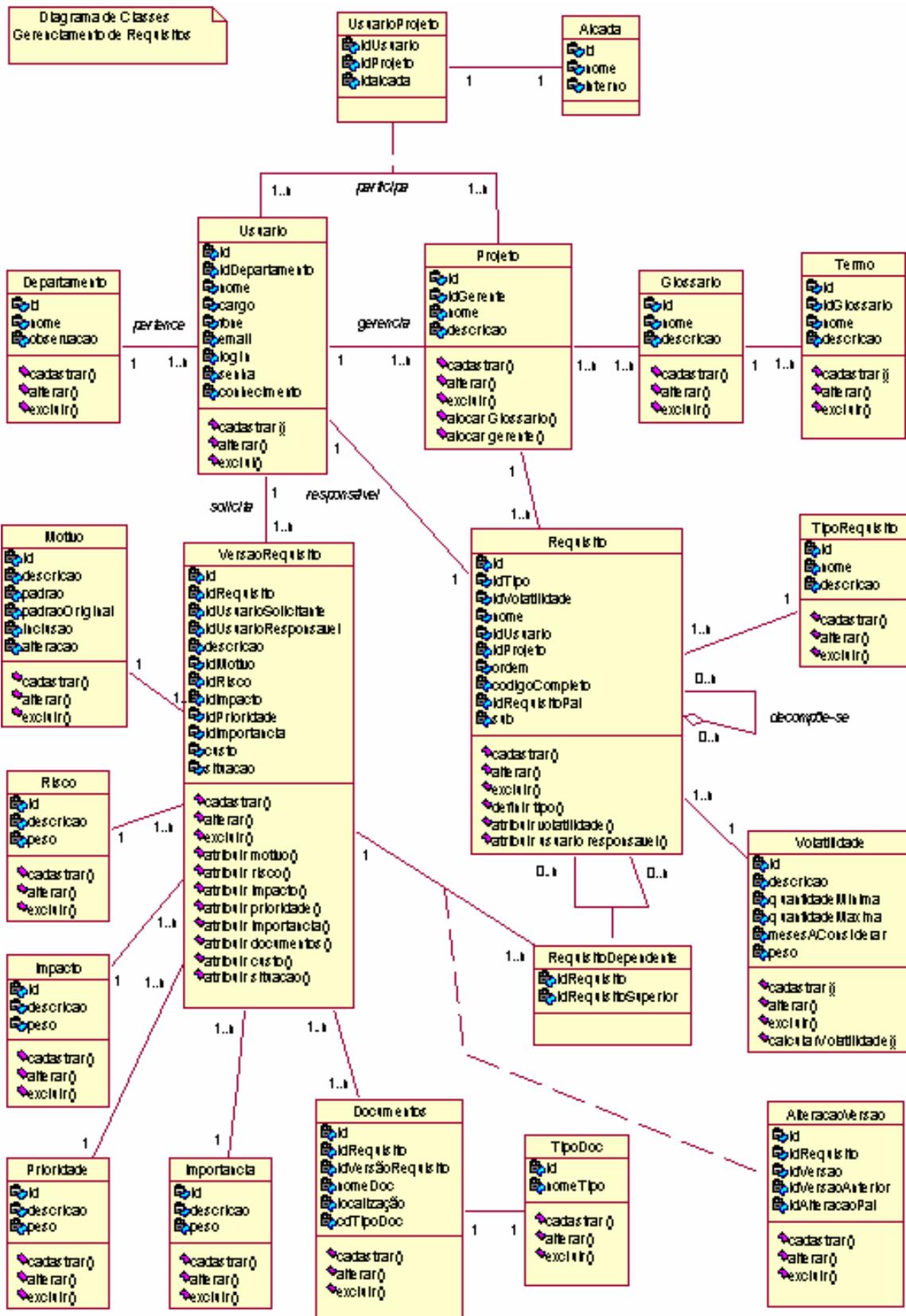


Figura 4 – Modelagem de Classes da Ferramenta

## Módulo Principal

Trata a alocação dos usuários aos projetos, mantém os requisitos e suas alterações (Versões de Requisitos), distribuídas em funções específicas, abaixo definidas:

- Aloca Usuários - permite ao Gerente do Projeto alocar usuários aos projetos, por níveis de alçada;
- Cadastra requisitos e dependências – permite cadastrar e alterar os requisitos do projeto, relacionando o requisito com seus requisitos dependentes, de forma a obter a rastreabilidade do requisito;
- Trata Alterações dos Requisitos – gera uma alteração no requisito do sistema implicando em criação de uma nova versão desse requisito, atribuindo impacto, risco, prioridade, motivo e custo, que é o número estimado de horas a ser consumida na implementação da versão do requisito. O sistema faz o tratamento do custo total da alteração proposta, que nada mais é que o somatório dos custos dos requisitos dependentes adicionado ao custo do requisito origem. O sistema usa a rastreabilidade para relacionar todos os requisitos dependentes (de forma recursiva) e a cada requisito dependente que gerar nova versão de requisito, o analista responsável deve atribuir valores conforme acima descrito, e assim sucessivamente até que todos os requisitos dependentes sejam analisados. A atribuição da situação das versões de requisitos é controlada internamente pelo sistema ou pelo Gerente do Projeto, de acordo com o estágio em que a versão do requisito origem e dependentes se encontram, e são definidas como "proposta", "em análise", "aprovada", "em desenvolvimento" e "implementada";
- Notifica envolvidos – são efetuadas após o sistema identificar e relacionar todos os requisitos dependentes, através da emissão de uma mensagem a cada analista responsável pelo requisito dependente, informando que uma nova versão de requisitos foi gerada e que há necessidade de intervenção no processo de análise (correio interno);
- Fórum de requisitos – o Fórum de requisitos é uma área de livre acesso a todos os envolvidos no projeto e é destinado a registrar críticas, sugestões, opiniões, alertas, entre outros, a respeito da alteração do requisito em análise;
- Relacionamento de documentos de especificação – o responsável relaciona os documentos produzidos na

especificação da nova versão como casos de uso, *layouts*, documentos textos, entre outros.

## 4. Análise de Viabilidade de Uso da Ferramenta

Após os testes iniciais partiu-se para a fase de análise de viabilidade de uso da ferramenta, utilizando-a em um estudo de caso real. O estudo de caso escolhido foi o Sistema de Gestão de Farmácias desenvolvido e implantado em Julho de 2002, por empresa fornecedora de sistemas de gestão na área de saúde, tendo seu escopo e documentação suficientes para registrar os requisitos iniciais e a evolução dos mesmos no decorrer dos períodos subsequentes de forma a aplicar a ferramenta na íntegra. Inicialmente foram levantados todos os documentos envolvidos no desenvolvimento (Lista dos requisitos Funcionais/Não-Funcionais e Lista de Dependências) e manutenção do sistema (Lista da Evolução dos Requisitos). A evolução dos requisitos foi registrada em ordem cronológica das alterações propostas nos requisitos e contém o histórico dos três anos seguintes à implantação do sistema.

### Utilização do Módulo Administração do Sistema

A primeira ação a ser tomada foi eleger um administrador que configurou o ambiente para o estudo de caso, cadastrando os Glossários, Departamentos, Usuários do sistema e dados dos cadastros de Volatilidade, Risco, Importância e Impacto, que receberam as atribuições de: “Baixo(a) – Peso 1”, “Médio(a) – Peso 5” e “Alto(a) – Peso 10”. No cadastro “Tipo de Requisito” estes foram classificados como Funcionais e Não-Funcionais e no cadastro “Tipo de Documento” foram cadastrados “Word”, “Excel” e “UML”. O cadastro “Motivo” recebeu os dados “Evolução”, “Legal” e “Correção”. A Figura 5 apresenta uma das telas do Módulo Administração do Sistema, onde foram cadastrados os usuários da ferramenta.

### Utilização do Módulo Principal

Após a configuração inicial do ambiente, o Gerente do Projeto e os demais usuários foram liberados para a utilização do “Menu Principal”. Inicialmente, o Gerente do Projeto alocou os envolvidos no projeto “Farmácias”, que então se tornaram aptos a trocarem mensagens através do correio interno. Em seguida, o Gerente do Projeto incluiu todos os requisitos iniciais (obtidos da Lista dos requisitos Funcionais/Não-Funcionais) com os respectivos tipos de requisitos,

responsável e volatilidade e as dependências entre os requisitos (obtidas da Lista de Dependências).

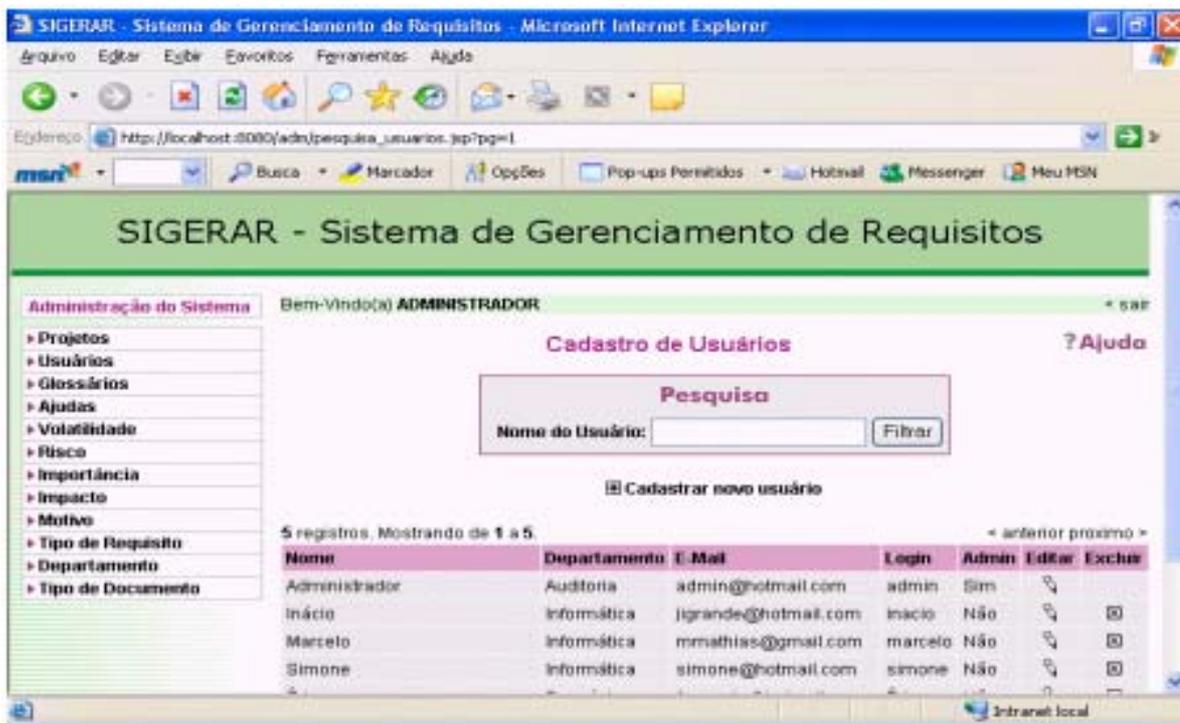


Figura 5 – Cadastro de Usuários da Ferramenta

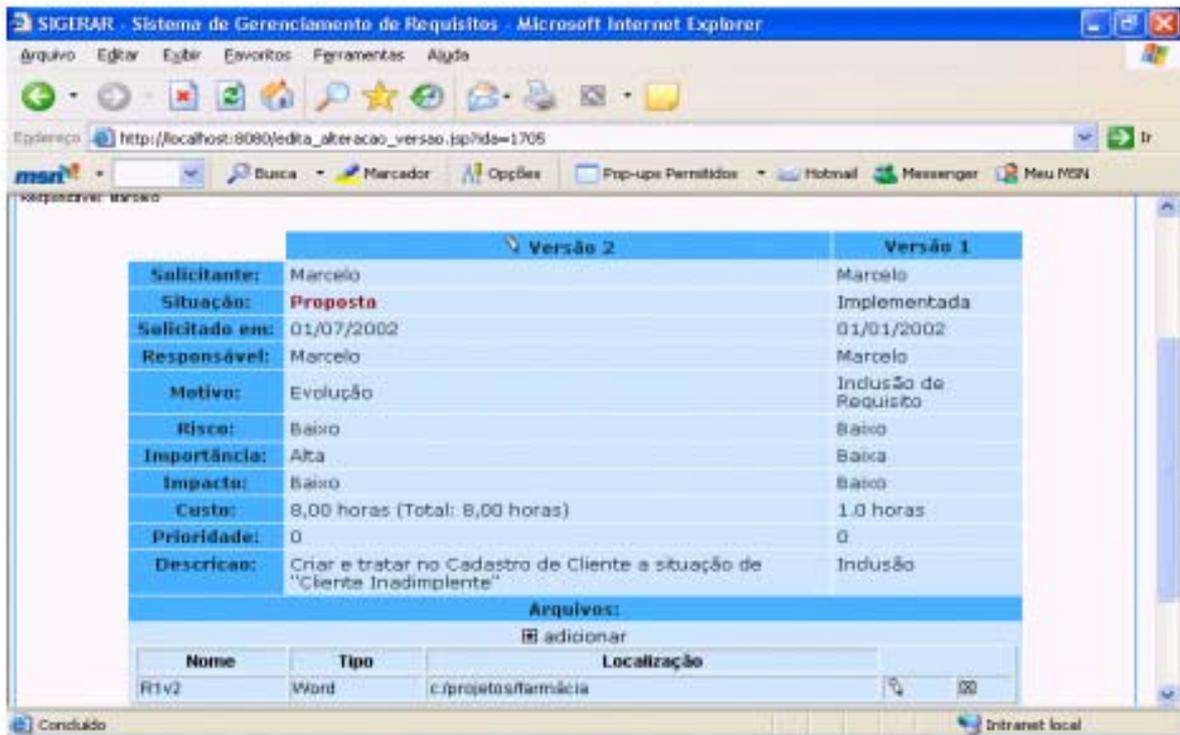


Figura 6 – Manutenção de Requisitos da Ferramenta

Em continuidade ao desenvolvimento do estudo de caso, a Lista da Evolução dos Requisitos foi submetida à ferramenta, seguindo a mesma ordem cronológica, a fim de observar o comportamento e potencialidade da ferramenta quanto ao tratamento da alteração dos requisitos (Versão de Requisitos). A Figura 6 apresenta uma das telas do Módulo Principal, onde é executada uma proposta de alteração de requisito.

### **Conclusões sobre o Uso da Ferramenta**

Na aplicação da ferramenta utilizando o estudo de caso “Sistema de Gestão de Farmácias” buscou-se traduzir com a maior fidelidade os acontecimentos desde o levantamento inicial dos requisitos e suas dependências, até a evolução cronológica dos requisitos. Estes dados foram sendo incluídos e testados na ferramenta, de forma que ao final desta atividade, concluímos o que segue.

### **Conclusões sobre o Módulo Administração**

- Todos os cadastros básicos necessários ao estudo de caso foram alimentados na ferramenta e nenhum problema foi constatado, mas sente-se necessidade de desenvolvimento de relatórios específicos.

### **Conclusões sobre o Módulo Principal**

- A alocação dos usuários no projeto com alçadas específicas foram efetuadas com sucesso;
- Todas as inclusões dos requisitos iniciais foram efetuadas e não houve nenhum impedimento ou problema constatado;
- As dependências iniciais listadas foram incluídas com sucesso;
- Nas gerações de versões, foram encontrados problemas no tratamento da alteração genérica “Ajustes nas telas do sistema para adequação ao Windows XP”, que envolve praticamente todos os requisitos e consiste em rever todas as telas do sistema. É necessário rever o conceito de alteração que envolva um grande número de requisitos, pois esta situação não foi contemplada na elicitação da ferramenta. Sentiu-se ainda, a necessidade de relatórios específicos que deverão ser implementados em próximas versões ou trabalhos futuros, como por exemplo, relatório de quantidade de horas consumidas no desenvolvimento e manutenção de requisitos, relatório de requisitos por situação (proposta / em análise / aprovada / em desenvolvimento / implementada), relatórios das dependências (rastreamento), etc.

Posto isto, pode-se afirmar que a ferramenta atende aos requisitos inicialmente traçados, embora careça de pequenas correções e melhorias, ressaltando que esta possui grande potencial de uso dentro da comunidade de Engenharia de *Software*, em particular entre gerentes e engenheiros de requisitos.

## **5. Conclusões**

Este trabalho contribui no sentido de propor uma ferramenta de uso livre, de fácil instalação, configuração e operação, para coletar, armazenar e manter os requisitos acordados entre os *stakeholders*, durante todo o ciclo de vida do *software*, de forma que equipes de gerentes, analistas e usuários de sistemas tenham controle sobre duas importantes questões do gerenciamento de requisitos: controle das versões de requisitos e rastreabilidade dos requisitos.

Além de contar com muitos dos recursos disponíveis nas ferramentas comerciais conhecidas, a ferramenta SIGERAR possui outras características importantes. Uma das principais características é o tratamento da rastreabilidade dos requisitos no processo de alteração, onde a mesma permite análise e atribuição de valores de risco, importância, impacto, prioridade e custo a todos os requisitos envolvidos (origem e dependentes), de forma a produzir informações ao Gerente do Projeto, que poderá analisar o contexto do impacto e custos da alteração.

Outro diferencial, é que embora a ferramenta contenha os textos de “ajuda” das telas previamente formatados, esta permite que cada organização possa customizá-los, de acordo com sua própria cultura ou características, de forma a obter maior adequação. Em trabalhos futuros, a ferramenta poderá ser aprimorada com inclusões de novas funcionalidades, como por exemplo, interoperabilidade com ferramentas eletrônicas utilizadas para documentação, como *Word*, *Excel*, *PowerPoint*, *Project*, *ferramentas CASE* que suportam UML entre outras. Outro item que merece destaque é o tratamento dos atributos prioridade, risco, importância e impacto dos requisitos origem e dependentes, pois cabe ao Gerente do Projeto avaliar estes dados e projetar mentalmente ou com a ajuda de outras ferramentas, os riscos e impactos potenciais da alteração.

Como cada dado tem um “peso” associado, a idéia é que a própria ferramenta possa obter o somatório dos “pesos” (de forma análoga ao praticado com o atributo custo), e compará-lo a valores previamente tabulados pelo administrador, para que forneçam informações sobre o risco e impacto que a mudança solicitada trará ao projeto, de forma a auxiliar os Gerentes de Projetos na tomada de decisão sobre a aceitação ou rejeição de alterações propostas.

## Referências

- [1] BOOCH, G., RUMBAUGH, J. e JACOBSON, I., “The Unified Modeling Language User Guide”, Addison Wesley, 1999.
- [2] GOTEL, O. e FINKELSTEIN, A., “An analysis of the Requirements Traceability Problem,” in Proceedings of the First International Conference on Requirements Engineering, (Colorado springs, CO), pp. 94-101, April 1994.
- [3] KOTONYA, G. e SOMMERVILLE, I., “Requirements Engineering: Processes and Techniques”, John Wiley and Sons, 1998.
- [4] LAM, W., LOOMES, M. e SHANKARARAMAN, V., “Managing Requirements Change Using Metrics and Action Planning”, Third European on Software Maintenance, mar. 1999, Amsterdam, Netherlands.
- [5] PINHEIRO, F. A. C., “Formal and Informal Aspects of Requirements Tracing”, III Workshop de Engenharia de Requisitos (WER 2000), 2000, Rio de Janeiro, Brasil.
- [6] RAMESH, B., POWERS, T. e STUBBS, C., “Implementing Requirements Traceability: A Case Study”, 2nd IEEE Symposium on Requirements Engineering, March 1995, York, England.
- [7] SOMMERVILLE, I., “Engenharia de Software”, 6ª edição. Pearson Education do Brasil, 2003.