

**INSTITUTO FEDERAL DE
EDUCAÇÃO, CIÊNCIA E TECNOLOGIA
FLUMINENSE**
Campus Campos-Centro

Secretaria de Educação
Profissional e Tecnológica

Ministério
da Educação



CURSO SUPERIOR DE TECNOLOGIA EM DESENVOLVIMENTO DE SOFTWARE

**DALINE GONÇALVES MORAES DE SOUZA
KAROLYNE ALMEIDA SIQUEIRA
RAFAEL LEITE DE FREITAS**

INTEGRAÇÃO DA GERÊNCIA DE REQUISITOS COM A PLATAFORMA REDMINE

Campos dos Goytacazes/RJ
2010

DALINE GONÇALVES MORAES DE SOUZA

KAROLYNE ALMEIDA SIQUEIRA

RAFAEL LEITE DE FREITAS

INTEGRAÇÃO DA GERÊNCIA DE REQUISITOS COM A
PLATAFORMA REDMINE

Monografia apresentada ao Instituto Federal de Educação
Ciência e Tecnologia Fluminense Campus Campos-Centro
como requisito parcial para a conclusão do Curso Superior de
Tecnologia em Desenvolvimento de Software.

Orientadora: Aline Pires Vieira de Vasconcelos

Campos dos Goytacazes/RJ
2010

DALINE GONÇALVES MORAES DE SOUZA

KAROLYNE ALMEIDA SIQUEIRA

RAFAEL LEITE DE FREITAS

INTEGRAÇÃO DA GERÊNCIA DE REQUISITOS COM A
PLATAFORMA REDMINE

Monografia apresentada ao Instituto Federal de Educação
Ciência e Tecnologia Fluminense Campus Campos-Centro
como requisito parcial para a conclusão do Curso Superior de
Tecnologia em Desenvolvimento de Software.

Aprovada em

Banca Avaliadora:

Profª Aline Pires de Vasconcelos (Orientadora)
Doutora em Engenharia de Sistemas e Computação pela COPPE/UFRJ
Instituto Federal de Educação, Ciência e Tecnologia Fluminense Campus Campos-Centro

Prof Jonivan Coutinho Lisboa
Mestre em Computação - Universidade Federal Fluminense
Instituto Federal de Educação Ciência, e Tecnologia Fluminense Campus Campos-Centro

Profª Michelle Maria Freitas Neto
Bacharel em Ciência da Computação – Universidade Federal Fluminense
Instituto Federal de Educação, Ciência e Tecnologia Fluminense Campus Itaperuna

AGRADECIMENTOS

Daline:

Agradeço primeiramente a Deus pela inteligência, sabedoria, persistência e saúde, e por nunca me deixar desistir dos meus sonhos. À minha família pelo sustento, educação e moral passados a mim através de exemplos de vida e conselhos, pelo apoio em todas as minhas decisões e vibração em todas as minhas conquistas. Ao meu namorado Fábio pelo amor, amizade, companheirismo e por me apoiar em tudo. À orientadora deste trabalho, Aline pelo conhecimento, dedicação e empenho em nos ajudar sempre. À professora Simone por me orientar no projeto Observatório EPT e pela confiança a mim prestada. A este Instituto Federal que se tornou minha segunda casa, e a todos os seus funcionários. Aos professores pelo conhecimento transmitido ao longo do curso. E aos colegas de classe pela companhia e ajuda.

Karolyne:

Agradeço primeiramente a Deus, pela força e por me conceder sabedoria para a realização desse trabalho. À minha família, pelo apoio, compreensão, ajuda e carinho em todos os momentos desta e de outras caminhadas. Especialmente a minha madrinha Maria Gracione Toledo Almeida, que esteve sempre ao lado e estará para sempre no meu coração. À minha orientadora Aline Vasconcelos, por todo o conhecimento passado, pela dedicação e orientação. Aos demais professores deste instituto que fizeram parte dessa jornada. Aos meus amigos, pela amizade e companheirismo.

Rafael:

Agradeço primeiramente a Deus por disponibilizar todos os caminhos para que eu pudesse alcançar meus objetivos. Aos meus familiares, pois foram eles que me apoiaram em todo momento para que eu estudasse e pudesse fazer meu curso superior. Agradeço também à minha namorada Bárbara, por ter me apoiado e estado comigo durante todo o decorrer do curso, sempre me animando quando o desânimo aparecia. Agradeço ainda à nossa orientadora Aline, que em nenhum momento deixou de nos ajudar no desenvolvimento do trabalho, e a todos os professores do curso que, em sua maioria, nos passaram o conhecimento necessário para que pudéssemos terminar o curso e consequentemente o trabalho de conclusão. Não podia deixar de lembrar dos meus companheiros do projeto SIGA-EPT que também contribuíram durante o curso e durante a elaboração da monografia.

RESUMO

O presente trabalho consiste na apresentação de uma ferramenta que se destina a apoiar o Gerenciamento de Requisitos de um projeto de software, fazendo com que esse processo seja realizado de forma mais qualitativa, eficiente e confiável, através da automatização de suas atividades em um ambiente de trabalho integrado. A ferramenta proposta visa apoiar as etapas de um processo de Gerência de Requisitos, automatizando o cadastro de requisitos, regras de negócio, glossário do domínio e casos de uso, e mantendo a rastreabilidade entre esses artefatos. Ela foi desenvolvida como um *plugin* para a plataforma Redmine, que se destina ao controle das tarefas de um projeto, apoiando a Gerência de Projetos. Dessa forma, a ferramenta proposta permite ao seu usuário, a otimização de tempo de trabalho, através da integração entre suas atividades de gerenciamento. O objetivo final é esta integração entre a Gerência de Requisitos e a plataforma Redmine. Ferramentas de apoio à Gerência de Requisitos foram pesquisadas, mas percebeu-se que nenhuma delas se encontra integrada a uma ferramenta de gerência de tarefas como o Redmine. O objetivo futuro é integrar as tarefas do Redmine com os requisitos e casos de uso que lhes dão origem, quando essas forem tarefas de desenvolvimento, permitindo ao Gerente de Projetos um controle integrado entre as diversas etapas e atividades do ciclo de vida de desenvolvimento de um sistema de software.

Palavras chave: Gerenciamento de Requisitos, Requisitos, Regras de Negócio, Casos de Uso, Glossário, Redmine e Rastreabilidade.

ABSTRACT

This work aims at presenting a tool for Requirements Management support in a software project, making this process more qualitative, efficient, and reliable through the automation of its activities in an integrated working environment. The proposed tool aims at supporting the steps of a Requirements Management process, making requirements, business rules, glossary, and use cases registration automated, and keeping the traceability among these artifacts. The tool was developed as a plugin for Redmine environment, that aims at controlling the tasks of a project, giving a support for Project Management. Therefore, the proposed tool allows its users to optimize working time, through the integration of his management activities. The final goal is this integration between Requirements Management and Redmine environment. Requirements Management tools were investigated, but it was observed that no tool was integrated in a tasks controlling environment as Redmine. The future goal is to integrate Redmine tasks with the requirements and use cases that originate them, whenever the tasks are development tasks, allowing the Project Manager to have an integrated control among the diverse of activities and steps that encompass the development life cycle of a software system.

Key words: Requirements Management, Requirements, Business Rules, Use Cases, Glossary, Redmine, and Traceability.

LISTA DE FIGURAS

| | |
|--|----|
| Figura 1 - Página do Redmine | 25 |
| Figura 2 - Portal da Renapi..... | 27 |
| Figura 3 - Diagrama de Casos de Uso | 35 |
| Figura 4 - Diagrama de Classes..... | 36 |
| Figura 5 - Tela inicial do Redmine..... | 39 |
| Figura 6 - Tela inicial da ferramenta | 40 |
| Figura 7 - Tela inicial de Glossário | 40 |
| Figura 8 - Tela de cadastro do glossário..... | 41 |
| Figura 9 - Tela de cadastro dos Requisitos..... | 41 |
| Figura 10 - Tela de exibição de um requisito | 42 |
| Figura 11 - Tela de rastreabilidade do Requisito RACAD-2.001 com termos de glossário | 42 |
| Figura 12 - Tela de cadastro de atores | 42 |
| Figura 13 - Tela de cadastro das Regras de Negócio | 43 |
| Figura 14 - Tela de cadastro do Caso de uso..... | 44 |
| Figura 15 - Tela de rastreabilidade do Caso de uso Manter Matriz com o termo do Glossário Matriz Curricular. | 45 |
| Figura 16 - Tela inicial da rastreabilidade com o glossário vinculado ao Caso de Uso..... | 45 |
| Figura 17 - <i>Links</i> para visualização das rastreabilidades..... | 46 |

LISTA DE TABELAS

| | |
|---|----|
| Tabela 1 - Descrição das classes da ferramenta | 37 |
| Tabela 2 - Comparação entre as ferramentas testadas | 46 |

LISTA DE SIGLAS

ASP - Active Server Pages

BPMN - Business Process Modeling Notation

CASE - *Computer-Aided Software Engineering*

CDU- Casos de Uso

CVS - *Concurrent Version System*

EPCT - Educação Profissional, Científica e Tecnológica

ERS – Especificação de Requisitos de Software

FAQ - *Frequently Asked Questions*

GPL - *General Public License*

GPS – Grupo de Processamento de Software

HTML - *HyperText Markup Language*

I/O - *Input/Output*

IFES – Instituto Federal do Espírito Santo

IFF – Instituto Federal Fluminense

LDAP - *Lightweight Directory Access Protocol*

MCA - Módulo de Controle de Acesso

OS – *Operating System*

PHP - *Hypertext Preprocessor*

PMBOK - *Project Management Body of Knowledge*

Renapi - Rede Nacional de Pesquisa e Inovação em Tecnologias Digitais

RUP - *Rational Unified Process*

SAMnet - Sistema de Administração de Materiais

SETEC/MEC - Secretaria de Educação Profissional e Tecnológica do Ministério da Educação

SIEP - Sistema de Informações da Educação Profissional e Tecnológica

SIGAD - Sistema Informatizado de Gestão Arquivística de Documentos

SIGA-EPT - Sistema Integrado de Gestão Acadêmica

SOS - Sistema de Ordem de Serviços

SVN - *Subversion*

UFAM – Universidade Federal do Amazonas

UFG – Universidade Federal de Goiás

UFMS – Universidade Federal do Mato Grosso do Sul

UML - *Unified Modeling Language*

URL - *Uniform Resource Locator*

USP – Universidade de São Paulo

SUMÁRIO

| | |
|---|----|
| RESUMO | 5 |
| ABSTRACT | 6 |
| LISTA DE FIGURAS | 7 |
| LISTA DE TABELAS | 8 |
| LISTA DE SIGLAS | 9 |
| | |
| CAPÍTULO 1: INTRODUÇÃO | 13 |
| 1.1 MOTIVAÇÕES | 14 |
| 1.2 OBJETIVOS | 14 |
| 1.3 ORGANIZAÇÃO | 15 |
| | |
| CAPÍTULO 2: REFERENCIAL TEÓRICO | 17 |
| 2.1 HISTÓRICO | 17 |
| 2.2 REQUISITOS | 17 |
| 2.3 GERÊNCIA DE REQUISITOS OU ENGENHARIA DE REQUISITOS | 19 |
| 2.3.1 RASTREABILIDADE | 23 |
| 2.4 REDMINE | 24 |
| 2.4.1 PROJETOS QUE UTILIZAM O REDMINE | 26 |
| 2.4.2 ALGUNS <i>PLUGINS</i> JÁ EXISTENTES | 30 |
| 2.5 FERRAMENTAS DE GERÊNCIA DE REQUISITOS DE SOFTWARE EXISTENTES | 31 |
| 2.6 CONSIDERAÇÕES FINAIS | 33 |
| | |
| CAPÍTULO 3: FERRAMENTA PROPOSTA DE GERÊNCIA DE REQUISITOS DE SOFTWARE INTEGRADA À FERRAMENTA REDMINE | 35 |
| 3.1 TECNOLOGIAS E ARQUITETURA | 35 |
| 3.1.1 DESCRIÇÃO DAS CLASSES | 37 |
| 3.1.2 RUBY ON RAILS | 37 |
| 3.2 EXEMPLO DE USO DA FERRAMENTA PROPOSTA | 38 |
| 3.2.1 DESCRIÇÃO DO CASO DE USO MANTER MATRIZ CURRICULAR | 38 |
| 3.2.2 DESCRIÇÃO DO REQUISITO RACAD-2.001 | 38 |
| 3.2.3 DESCRIÇÃO DA REGRA DE NEGÓCIO RACAD-2.001 | 38 |

| | |
|--|----|
| 3.3 UTILIZANDO A FERRAMENTA PROPOSTA | 39 |
| 3.3.1 CADASTRANDO GLOSSÁRIO | 40 |
| 3.3.2 CADASTRANDO REQUISITO | 41 |
| 3.3.3 DEFININDO A RASTREABILIDADE ENTRE REQUISITO E GLOSSÁRIO... | 41 |
| 3.3.4 CADASTRANDO ATORES | 42 |
| 3.3.5 CADASTRANDO REGRAS DE NEGÓCIO | 43 |
| 3.3.6 CADASTRANDO CASOS DE USO | 43 |
| 3.3.7 RASTREABILIDADES DOS CASOS DE USO | 44 |
| 3.3.7 DEFININDO RASTREABILIDADE DE CASOS DE USO X GLOSSÁRIO | 45 |
| 3.4 CONSIDERAÇÕES FINAIS | 46 |
| CAPÍTULO 4: CONCLUSÕES | 47 |
| 4.1 CONTRIBUIÇÕES | 47 |
| 4.2 APRIMORAMENTO DA FERRAMENTA | 47 |
| REFERÊNCIAS | 50 |
| APÊNDICE 1: Descrição dos casos de uso da ferramenta proposta..... | 56 |

CAPÍTULO 1: INTRODUÇÃO

Segundo *Sommerville* (2003), a Engenharia de Requisitos é um processo que envolve todas as atividades exigidas para criar e manter o documento de requisitos de sistema. Existem quatro atividades genéricas de processo de engenharia de requisitos que são de alto nível, ou seja, o estudo da viabilidade do sistema, a obtenção e a análise de requisitos, a especificação de requisitos e sua documentação, e a validação desses requisitos.

Um diferencial da Engenharia de Requisitos em relação às outras áreas da Engenharia de Software é seu estreito relacionamento com os usuários finais. Em um mundo com cada vez mais usuários e com a contínua inserção da computação no dia-a-dia das pessoas, a especificação dos requisitos, de maneira economicamente viável, torna-se cada vez mais uma atividade complexa e importante.

A Engenharia de Software é responsável pelo estudo e desenvolvimento de técnicas que auxiliem e guiem o processo de definição de requisitos durante toda a vida do projeto. Através do desenvolvimento de aplicação de novas técnicas, torna-se possível a detecção prematura de erros, implicando em menores custos e prazos e a satisfação das necessidades dos usuários do software.

O levantamento dos requisitos do sistema trata-se de um período em que as necessidades do cliente serão conhecidas, sendo assim uma fase de suma importância para o desenvolvimento de software.

Um requisito é uma expressão do comportamento pretendido para o sistema. Os requisitos focam as necessidades dos clientes e não a solução ou a implementação do sistema. Os requisitos são a definição da vontade do cliente, portanto representam a base para a satisfação do mesmo e para o sucesso de um sistema, daí sua importância e presença fundamental para a Engenharia de Software.

Segundo *Sommerville* (2003), os requisitos, em Engenharia de Software, são classificados em: requisitos funcionais e não funcionais. Os requisitos funcionais descrevem a interação entre o sistema e o seu ambiente, o que faz o sistema e quando faz, modos de operação, transformações nos dados, reações esperadas a estímulos externos, formato dos dados de I/O. Estes representam as funcionalidades que o sistema deve apresentar para o seu usuário final. Já os requisitos não funcionais descrevem qualidades/características que o software deve ter, como desempenho (tempo de resposta), usabilidade (tempo de aprendizagem para a utilização), segurança (acesso restrito), portabilidade (capacidade de

rodar em múltiplas plataformas), testabilidade (facilidade em detectar e isolar falhas), manutenibilidade (facilidade para corrigir erros, melhorar o sistema), etc.

1.1 MOTIVAÇÕES

Através de um estudo, foi possível perceber que existem ferramentas que fazem a Gerência de Requisitos, sendo a maioria de código fechado e de custo elevado. Além disso, essas ferramentas não são integradas a outras ferramentas de gerenciamento de tarefas. Neste trabalho, foi pesquisada a plataforma de trabalho Redmine (encontrado no site: www.redmine.org), que é uma plataforma de gerenciamento de tarefas, que apóia a Gerência de Projetos de software, e inclui várias sub-ferramentas, como: fórum, *wiki*, gerência de *bugs*, tarefas, geração de cronograma de Gantt, etc. Uma das motivações para o uso dessa plataforma é o fato dela vir sendo utilizada em diversos projetos do MEC, como os projetos da Renapi (Rede Nacional de Pesquisa e Inovação em Tecnologias Digitais), cujo desenvolvimento é apoiado por Núcleos de Pesquisa instituídos neste ambiente de ensino, i.e. no IFF.

O Redmine é um software livre, o qual não tem restrições quanto ao uso das tecnologias envolvidas e cujas licenças encorajam e protegem seu desenvolvimento. Ele é implementado em linguagem *Ruby on Rails*, *framework* de desenvolvimento *web*, gratuito e de código aberto que permite o desenvolvimento ágil, com alta produtividade, escrevendo poucas linhas de código e tendo muito resultado como consequência. O Redmine recebe constantes modificações no que diz respeito a novas funcionalidades, as quais visam, acima de tudo otimizar essa ferramenta que já se mostra tão completa para o que se propõe a fazer.

As outras ferramentas de requisitos existentes, como ferramentas CASE, *Enterprise Architect*, *Controla*, *Requisitemanager*, *SPRES*, *RequisitePro* e *Borland Caliber Analyst* não são integradas a uma ferramenta de gerenciamento de tarefas e apoio à Gerência de Projetos como o Redmine. Dessa forma, não permitem ao Gerente de Projetos um controle integrado das suas atividades e a visualização de boa parte do ciclo de vida do software em um único ambiente de trabalho.

1.2 OBJETIVOS

O presente projeto visa o desenvolvimento de uma ferramenta de apoio ao Gerenciamento de Requisitos de Software, visando apoiar as etapas de especificação,

documentação e validação dos requisitos. Dessa forma, ajuda a entender o funcionamento de todo o processo de levantamento e análise dos requisitos até o desenvolvimento do sistema, passando pela compreensão do domínio, a classificação, a definição das prioridades, a especificação dos requisitos e finalmente a documentação de requisitos. Todos esses objetivos visam ser atingidos através da automatização do cadastro dos artefatos da Gerência de Requisitos (i.e. requisitos, glossário do domínio, regras de negócio e casos de uso), em uma ferramenta de controle integrado, mantendo-se a rastreabilidade entre esses artefatos. A rastreabilidade é importante para a compreensão dos requisitos e para a análise de impacto de mudanças no software.

Desta forma, o presente trabalho apresenta uma ferramenta de Gerenciamento de Requisitos de software integrada ao Redmine, que visa garantir o desenvolvimento de software com elevado padrão de qualidade e reduzindo o custo e o tempo de gerenciamento e desenvolvimento do projeto.

A ferramenta tem como objetivo proporcionar ao usuário, e, principalmente, ao Gerente de Projetos, usufruir de um completo instrumento, que permite um controle integrado de tarefas e requisitos em um mesmo local, fazendo com que ocorra uma maior otimização de tempo e trabalho.

Outra finalidade da ferramenta é ser livre, uma vez que a plataforma Redmine é livre, e de código aberto, sendo esta uma característica que a diferencia de grande parte das ferramentas encontradas.

Dentre outras coisas, a ferramenta dará dinamismo ao processo de especificação de requisitos, permitindo que ele seja de forma mais eficiente, principalmente por estar automatizado, não havendo a necessidade de registro dos artefatos em diversos arquivos, e integrado em uma ferramenta de gerência de tarefas.

1.3 ORGANIZAÇÃO

Este trabalho está dividido em mais 3 Capítulos, além desta Introdução.

O Capítulo 2 descreve todo o referencial teórico pesquisado, envolvendo a Gerência de Requisitos e ferramentas estudadas, o que motivou a escolha da Gerência de Requisitos como tema deste projeto.

O Capítulo 3 apresenta a ferramenta desenvolvida e que é objeto desse trabalho, contendo uma explicação de como se deve desenvolver uma ferramenta como *plugin* para o Redmine, além dos modelos e o funcionamento da ferramenta.

O Capítulo 4 apresenta as conclusões e resultados obtidos com a realização do trabalho, e ainda trabalhos futuros que poderão ser desenvolvidos.

O Apêndice 1 apresenta a descrição dos casos de uso da ferramenta.

CAPÍTULO 2: REFERENCIAL TEÓRICO

2.1 HISTÓRICO

No final da década de oitenta surgiu a Engenharia de Requisitos. E desde então, esta área da Engenharia de Software vem ganhando novos aliados, como ferramentas que se destinam a tornar o Gerenciamento dos Requisitos uma tarefa cada vez mais aprimorada e facilitada para o desenvolvedor e o usuário de software.

2.2 REQUISITOS

O Gerenciamento de Requisitos é um modo sistemático para encontrar, documentar, organizar e rastrear os requisitos variáveis de um sistema.

Um requisito é definido como uma condição ou uma capacidade com a qual o sistema deve estar de acordo.

Segundo a *Rational Unified Process* (2001), o gerenciamento de requisitos trata-se de um modelo sistemático para:

- identificar, organizar e documentar os requisitos do sistema;
- estabelecer e manter acordo entre o cliente e a equipe do projeto nos requisitos do sistema.

Os principais itens para o gerenciamento eficiente de requisitos incluem manter uma declaração clara dos requisitos, juntamente com atributos aplicáveis para cada tipo de requisito e rastreabilidade para outros requisitos e outros artefatos do projeto.

Os atributos podem ser:

- Origem - Pessoa, documento ou outra fonte de um dado requisito;
- Prioridade - Declaração da importância do requisito, em relação ao sistema (obrigatório, crítico, opcional) ou aos outros requisitos (alta, média, baixa);
- Atribuído a - Quem na organização é responsável por certificar que o requisito foi satisfeito (nome da pessoa ou da organização);
- Comentários - Comentários do revisor ou autor sobre o requisito;
- Dificuldade - Uma indicação do nível de esforço necessário ou quão difícil será implementar o requisito (alta, média, baixa);
- Status - Grau de completude (completado, parcial, não iniciado);

- Risco - Medida de certeza sobre a satisfação (ou não) do requisito. Poderia ser alto, médio, baixo ou números inteiros numa escala de zero a dez;
- Data - Data em que o requisito deve ser fornecido;
- Método de Verificação - Tipo de qualificação a ser usado para verificar se um requisito foi satisfeito: análise, demonstração, inspeção, teste ou auditoria;
- Fase de Teste - Descreve a fase no ciclo de vida de verificação em que o requisito é dito satisfeito: teste unitário, componente, sistema ou produto;
- Alocação em Subsistema - Nome do sistema ou subsistema a que um requisito é atribuído (por exemplo, módulo de controle de voo, conjunto de asas, cabine de passageiros);
- Número do Teste - Identificação de um teste específico ou outro método de verificação.

Segundo *Sommerville* (2003), os requisitos podem ser separados em 3 níveis:

- Requisitos do usuário, que são declarações em linguagem natural ou em diagramas sobre as funções que o sistema deve conter e suas restrições;
- Requisitos de sistema, que estabelecem detalhadamente as funções e as restrições de sistema;
- Especificação de projeto de software, que é uma descrição abstrata do projeto de software, e serve como base para o mesmo e a implementação mais detalhados.

Por sua vez, os requisitos do sistema de software são frequentemente classificados como:

- Requisitos Funcionais: são declarações de funções que o sistema deve fornecer, como o sistema deve reagir a entradas específicas e como deve se comportar em determinadas situações. Em alguns casos, também pode explicitar o que o sistema não deve fazer.
- Requisitos Não Funcionais: são restrições sobre os serviços ou as funções oferecidas pelo sistema, entre eles, destacam-se restrições de tempo, restrições sobre o processo de desenvolvimento, padrões, entre outros.
- Requisitos de Domínio: se originam do domínio de aplicação do sistema e que refletem características desse domínio. Podem ser requisitos funcionais ou não funcionais.

Também segundo *Sommerville* (2003), no decorrer do processo de desenvolvimento de um software, os requisitos tendem a evoluir, a fim de refletir as mudanças que vão ocorrendo no sistema em construção. A partir dessa perspectiva de evolução, os requisitos são divididos em duas classes:

- Requisitos permanentes: são requisitos relativamente estáveis, que derivam da atividade principal da organização e que se relacionam diretamente com o domínio do sistema;
- Requisitos voláteis: são requisitos que provavelmente vão se modificar durante o desenvolvimento do sistema ou depois que o sistema estiver em operação.

2.3 GERÊNCIA DE REQUISITOS OU ENGENHARIA DE REQUISITOS

O Processo de Gerência de Requisitos de Software estabelece e mantém a concordância entre o cliente e os desenvolvedores durante as alterações dos requisitos do projeto, aproximando sistematicamente a elicitação, organização e documentação dos requisitos de software.

Esse processo é realizado em conjunto com outros processos da engenharia de requisitos. O planejamento inicia-se simultaneamente ao levantamento inicial de requisitos e o gerenciamento dos requisitos deve começar assim que um esboço da versão do documento estiver disponível.

Um Gerenciamento de Requisitos deficiente pode resultar em um software:

- que resolve incorretamente o problema;
- que não funciona como esperado;
- difícil de utilizar.

O Gerenciamento de Requisitos envolve a coleta, armazenamento e manutenção de grande quantidade de informação.

O processo de Gerenciamento de Requisitos nasceu da constatação que as mudanças normalmente ocorrem durante o processo de desenvolvimento de produtos. Estas mudanças são geradas por alterações de legislação, de objetivos de negócio, ou mesmo de um melhor entendimento das potencialidades oferecidas pelos sistemas de informações. Na maioria das vezes, tais mudanças implicam em renegociação de prazo, custo e de esforço para o projeto. Daí a importância em se ter os requisitos do software documentados durante todo o processo de desenvolvimento e atualizados constantemente mesmo com o software em produção. Uma plataforma que facilite essa atualização constante é de suma importância para uma adequada Gerência de Requisitos.

A Gerência de Requisitos deficiente é considerada o principal problema dos projetos de software, tendo como objetivo principal controlar a evolução dos requisitos de um sistema,

seja por constatação de novas necessidades, seja por constatação de deficiências nos requisitos registrados até um determinado momento.

Normalmente um sistema é particionado em uma hierarquia de elementos.

Cada requisito do sistema deve ser alocado a estes elementos (por exemplo: subsistemas, programas). Chama-se esta tarefa de alocação de requisitos, e é fundamental para realizar o acompanhamento e rastreabilidade dos requisitos.

Além disso, a Gerência de Requisitos conta com outros artefatos, como as regras de negócio, glossário do domínio e casos de uso.

Os sistemas de informação abrangem, com frequência, um grande volume de conhecimento organizacional formalizado. Tal conhecimento é especificado como regras de negócio, definidas por *Bellevue Golden* (1990) como sendo “declarações que apresentam a maneira de como o negócio está sendo feito, além das diretrizes e restrições com respeito a estados e processos em uma organização”. Tais regras de negócio têm sido amplamente utilizadas pelas organizações como forma de melhorar a qualidade de seus sistemas.

O glossário de domínio é definido como o conjunto de termos (nomes) onde cada qual possui o seu próprio significado (descrição). Domínios distintos empregam terminologias distintas, muitas vezes desconhecidas da equipe de desenvolvimento. Um glossário é o mecanismo empregado para registrar os termos de um domínio e, desta forma, facilitar o emprego deles.

Segundo *JACOBSON, I. et al.* (1999), um caso de uso é um pedaço de funcionalidade do sistema que dá ao usuário um resultado de valor. Casos de uso capturam requisitos funcionais e todos juntos resultam no modelo de casos de uso, o qual descreve a funcionalidade completa do sistema. Os casos de uso direcionam o processo de desenvolvimento, já que, baseados no modelo de casos de uso os desenvolvedores criam uma série de modelos de projeto e implementação que os realizam efetivamente.

Todos esses artefatos da Gerência de Requisitos, em separado, desempenham papel fundamental através de um ambiente de desenvolvimento, mas no decorrer do projeto faz-se necessária a integração destes através de um sistema de rastreabilidade, o que fará com que o impacto de mudanças dos requisitos seja estimado com maior eficiência, e dando integridade ao produto final.

Para apoiar um Gerenciamento efetivo de Requisitos funcionais, os indicadores de estabilidade (aferem o grau de mudanças para a *Baseline* dos requisitos de software. Eles fornecem o impacto da mudança no tamanho da *Baseline*, sob o ponto de vista funcional) e indicadores de rastreabilidade (medem o percentual de requisitos que podem ser rastreados

entre dois níveis adjacentes de especificação) são formas de representação quantificáveis de características de produtos e processos, sendo utilizados para acompanhar e melhorar os resultados ao longo do tempo [HAZAN & LEITE].

Segundo *Emam* [EMAM, 1997], os principais problemas relacionados à Gerência de Requisitos são os seguintes:

- Dificuldades de elicitar claramente as mudanças nos requisitos;
- Falta de habilidade para chegar a um consenso sobre as mudanças chave para os stakeholders;
- Falta de habilidade para manter o documento de requisitos consistente;
- Falta de habilidade para estimar adequadamente os recursos necessários para implementar as mudanças nos requisitos.

Segundo *Roger Oberg, et al* (2009), o Gerenciamento de Requisitos é um modelo sistemático para encontrar, documentar, organizar, rastrear, compreender e controlar as mudanças que ocorrem nos requisitos, por força da evolução dos mesmos, refletindo as alterações que ocorrem ao longo do tempo, no ambiente do sistema e nos objetivos da organização. Além da Análise e Especificação, o Gerenciamento de Requisitos é de fundamental importância no processo da Engenharia de Software, pois organiza o controle das mudanças, permitindo subsídios para a análise de impacto e custos em tempo e dinheiro, que estas mudanças trarão para a organização.

A Gerência de Requisitos tem como objetivo identificar e registrar alterações das necessidades que o produto de software terá que atender (requisitos do produto), além de orientar a avaliação de impactos destas mudanças nos planos do projeto.

Para manter os requisitos de forma consistente utiliza-se o processo de gerenciamento de requisitos, que segundo *Sommerville* (2003), é um processo de controle de alterações nos requisitos dos sistemas. Durante esse estágio é necessário decidir sobre os seguintes aspectos:

- identificação de requisitos: cada requisito precisa ser identificado de modo único, para que possa ser feita a referência cruzada deste com os outros requisitos e demais produtos de trabalho, e para que ele possa ser utilizado nas avaliações de facilidade de rastreamento;
- processo de gerenciamento de mudanças: trata-se do conjunto de atividades que avalia o impacto e o custo das mudanças, que devem ser avaliadas em função dos requisitos definidos inicialmente;

- políticas de facilidade de rastreamento: essas políticas definem as relações entre os requisitos e entre os requisitos e o projeto de sistema que devem ser registrados e também como esses registros devem ser mantidos;
- suporte de ferramentas: o gerenciamento de requisitos envolve o processo de uma grande quantidade de informações sobre os requisitos e, dessa forma, deve-se definir um ferramental de apoio.

Os requisitos retratam as intenções do cliente para com o software a ser desenvolvido, uma clara compreensão dos mesmos são de suma importância para que os desenvolvedores saibam exatamente o que deve ser construído para satisfazer as reais necessidades do cliente.

As mudanças dos requisitos ocorrem conforme os clientes desenvolvem um melhor entendimento de suas reais necessidades.

Segundo *Sommerville* (2003), novos requisitos surgem e há alterações nos requisitos em todos os estágios do processo de desenvolvimento do sistema. São comuns os casos em que mais de 50% dos requisitos são alterados antes que o sistema seja posto em operação, o que causa sérios problemas para os desenvolvedores.

Para minimizar dificuldades, os requisitos devem ser documentados e controlados.

Quando não há controle de alterações, mudanças de baixa prioridade podem ser implementadas antes daquelas de alta prioridade, além de mudanças com alto custo não serem necessariamente aprovadas. Além disso, pode ser difícil realizar uma análise de impacto de mudança, uma vez que os requisitos não estão adequadamente documentados e rastreados.

Os principais itens para o gerenciamento eficiente de requisitos incluem manter uma declaração clara dos requisitos, juntamente com atributos aplicáveis para cada tipo de requisito e rastreabilidade para outros requisitos e outros artefatos do projeto.

Os atributos dos requisitos dependem do tipo do requisito, e podem ser motivo, risco, esforço, *status*, percentual andamento, estabilidade, risco-tecnologia, etc.

A coleta de requisitos é uma tarefa dispendiosa porque:

- Nem sempre os requisitos são óbvios e podem vir de várias fontes.
- Nem sempre é fácil expressar os requisitos claramente em palavras.
- Existem diversos tipos de requisitos em diferentes níveis de detalhe.
- O número de requisitos poderá impossibilitar a gerência se não for controlado.
- Os requisitos estão relacionados uns com os outros, e também com o produto liberado do processo de engenharia do software.
- Os requisitos têm propriedades exclusivas ou valores de propriedade. Por exemplo, eles não são igualmente importantes nem igualmente fáceis de cumprir.

- Há várias partes interessadas, o que significa que os requisitos precisam ser gerenciados por grupos de pessoas de diferentes funções.

- Os requisitos são alterados.

As principais preocupações de gerenciamento de requisitos são:

- Gerenciar mudanças nos requisitos acordados.
- Gerenciar os relacionamentos entre os requisitos.
- Gerenciar as dependências entre o documento de requisitos e outros documentos produzidos ao longo do sistema e do processo de engenharia de software.

O processo de Engenharia de Requisitos tem como principal objetivo 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. Uma análise e especificação de requisitos com baixa qualidade podem levar a danos e prejuízos, que inviabilizam o projeto e que podem até mesmo colocar em risco vidas humanas, como por exemplo, em sistemas hospitalares, de tráfego aéreo, de controle de trânsito etc.

2.3.1 RASTREABILIDADE

O processo de desenvolvimento de software é bastante complexo e dinâmico, exigindo a produção de uma quantidade considerável de documentos. A cada etapa do desenvolvimento novos documentos são criados e documentos existentes podem ser alterados diversas vezes. Normalmente, uma modificação em um documento pode causar impacto em vários outros.

A rastreabilidade estima o impacto de uma mudança em um requisito, mostra se um requisito é derivado ou dependente de outro, ajuda a refletir a necessidade do cliente no produto final. Permite que os desenvolvedores avaliem o impacto das alterações dos requisitos e garante que o projeto atenda às expectativas. Para isso os vínculos da rastreabilidade devem ser compreendidos, documentados e mantidos. Permite também, verificar a origem dos requisitos, tanto dos tipos de Casos de Uso quanto das Especificações Suplementares, para a análise do impacto das mudanças de requisitos.

Segundo *Oberg* (2001), normalmente requisitos não estão isolados: solicitações de usuários estão relacionadas às funcionalidades do produto e estas por sua vez estão relacionadas aos requisitos funcionais e não-funcionais do sistema; casos de testes estão

relacionados a requisitos que os verificam e validam. Requisitos podem vincular-se entre si ou ser mutuamente exclusivos.

Segundo *Pinheiro* (1996), o rastreamento de requisitos é justificado porque:

- os requisitos evoluem durante a vida de um software;
- os requisitos são contextualizados e dependem de detalhes da situação particular do contexto em que surgem;
- o rastreamento de requisitos também é contextualizado. Isto implica que um rastreamento poderá produzir objetos significativos para determinada situação;
- os requisitos são parte intrínseca do processo de desenvolvimento e o rastreamento de artefatos do projeto é útil por todo o ciclo de vida dos requisitos e do desenvolvimento.

Manter a rastreabilidade entre os diversos documentos de um projeto é considerado imprescindível para obtenção de um produto de software de qualidade. Entre outras coisas, a rastreabilidade auxilia na avaliação do impacto no projeto que eventuais modificações em artefatos possam causar e também contribui para verificar se todas as partes envolvidas no projeto têm suas necessidades satisfeitas. Entende-se como “partes envolvidas”, os clientes, seus possíveis representantes, órgãos reguladores, desenvolvedores, testadores, enfim todos os envolvidos no projeto.

A rastreabilidade é complexa de implantar e cara de ser mantida, já que abrange o controle de um grande número de associações entre artefatos, de diversos tipos de informações e de várias partes envolvidas. Normalmente, ela vem sendo realizada de forma manual pelas equipes de desenvolvimento de software através de planilhas construídas no Excel.

2.4 REDMINE

O Redmine é uma plataforma especialmente destinada ao gerenciamento de tarefas e apoio ao gerenciamento de projetos baseada na *web*. Seu *framework* foi desenvolvido em linguagem *Ruby* (*Ruby on Rails*) e disponibilizado sob licença GPL. O Redmine é um software livre, multiplataforma e suporta diversos bancos de dados. Unifica uma variedade de SCM (Sistemas de Controle de Versão), *Subversion*, *CVS*, *Git*, *Mercurial*, *Bazaar* e *Darcs*.

A seguir tem-se a tela do Redmine.

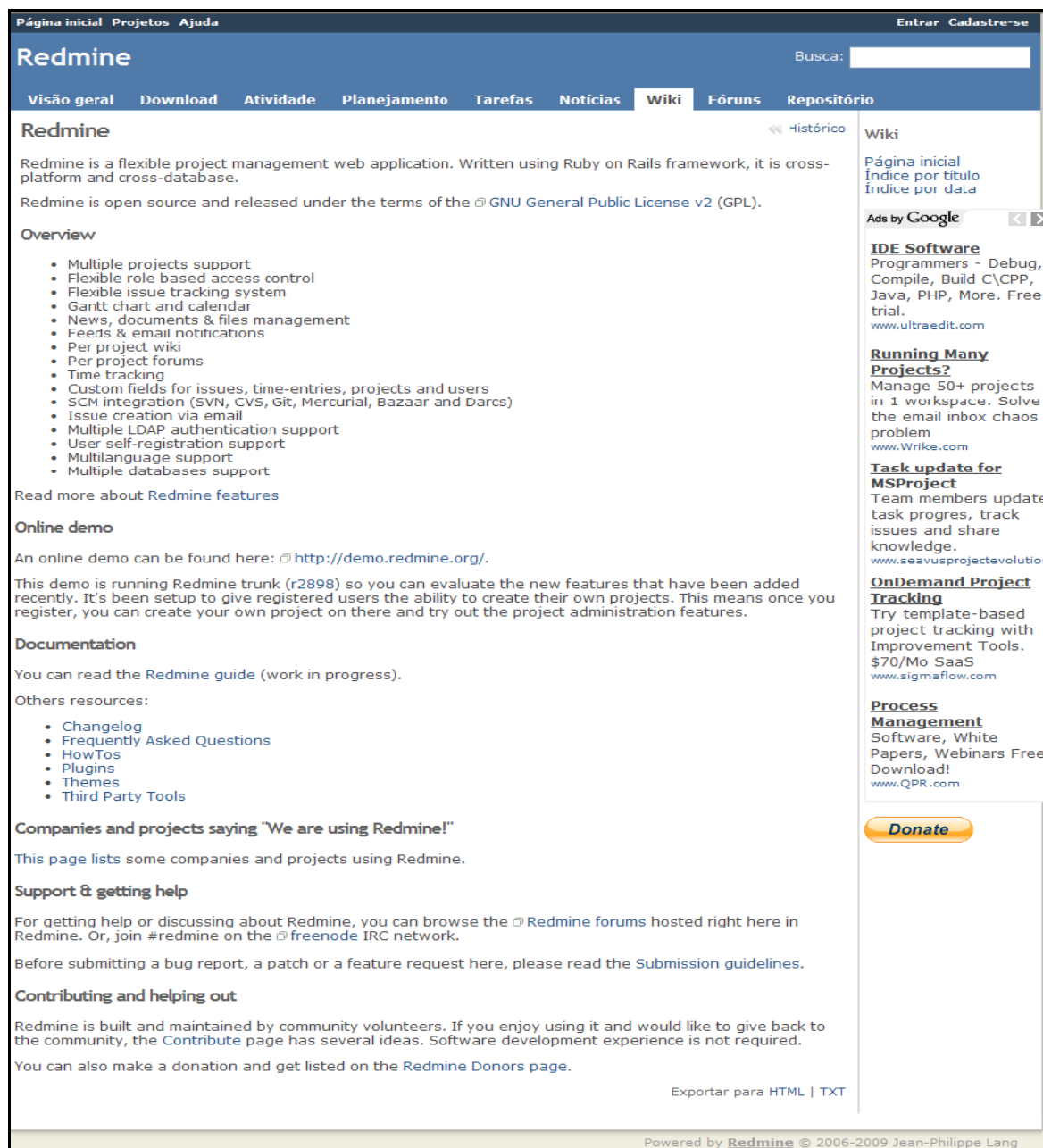


Figura 1 - Página do Redmine

O Redmine se destaca por ser uma ferramenta multiplataforma que suporta construção de *layouts* personalizados e *plugins* específicos.

Garante um melhor gerenciamento e a integração de todas as áreas de um projeto, possui como principais características a organização, suporte nativo a multi-projetos, níveis flexíveis de acesso e controle, suporte a atualizações e notificações por e-mail.

Tem como principal funcionalidade o Gerenciamento de Tarefas. Permitindo o desenvolvimento de projetos de maneira colaborativa, um único projeto pode ser construído por inúmeras pessoas, de lugares diferentes, através do designo de tarefas aos participantes, as

mesmas podem ser monitoradas pelos gerentes e alteradas a qualquer momento, o que faz com que o projeto se torne organizado, confiável e de qualidade.

O Redmine é implementado de forma muito mais completa que outros softwares que se dispõe ao mesmo objetivo e gera o caminho crítico do projeto. Permite o cadastro de fase, atividade, tarefa, esforço estimado, esforço real, data de início previsto, data de fim previsto, data de início real, data de fim real, recurso previsto e recurso real de um projeto.

Do ponto de vista do usuário final, o Redmine apresenta funcionalidades como uma *Wiki* e fórum disponíveis para cada projeto. E a estrutura de links no estilo *Wiki* ajuda bastante. A inclusão de novos projetos é feita na interface administrativa (através de roles específicas para isso), facilitando a inclusão de projetos.

Possui calendário das tarefas (com início e final previstos, além de número de horas previstas para cada atividade) e um gráfico de Gantt com o previsto e o real para cada atividade.

Através do Recurso de Planejamento obtém-se uma visão das versões previstas, incluindo a quantidade de dias restantes para finalização do projeto, a data da finalização, o percentual de término do projeto, a quantidade de tarefas concluídas e abertas com seus percentuais e as tarefas relacionadas. Além disso o gerente de projeto pode acompanhar o desenvolvimento através do Recurso de Atividades do Redmine. Toda alteração feita no projeto é apresentada em uma lista de ações classificadas por dia.

Cada projeto pode conter um repositório de dados, no qual tanto o código fonte como os documentos são versionados. Isso significa que é possível obter um documento ou código fonte em um estado anterior. Os projetos seguem um modelo padrão de árvore de diretório, com pastas para os documentos.

Com frequência, novas melhorias são disponibilizadas, por ser software livre, possibilita cada vez mais o crescimento da ferramenta com novas funcionalidades, pois possui uma equipe ativa de desenvolvimento e de melhoria do produto.

2.4.1 PROJETOS QUE UTILIZAM O REDMINE

O Redmine é utilizado nos projetos da Rede Nacional de Pesquisa e Inovação em Tecnologias Digitais – Renapi, como gerenciador, mas a utilização deste instrumento pode ser feita por qualquer desenvolvimento de software para além destes.

Abaixo temos a tela do Portal da Renapi.



Figura 2 - Portal da Renapi

Desde o início de 2006, a criação de sistemas de informação como estratégia de expansão, modernização e melhoria contínua da Educação Profissional, Científica e Tecnológica – EPCT, no âmbito da Secretaria de Educação Profissional e Tecnológica do Ministério da Educação – SETEC/MEC, tornou-se imprescindível e culminou na concepção do Sistema de Informações da Educação Profissional e Tecnológica – SIEP.

São inúmeros os benefícios colhidos pela SETEC e pela Rede Federal de EPCT durante o período de execução do SIEP, principalmente na consolidação de uma ação articulada e integrada voltada à pesquisa e inovação em tecnologias digitais.

O processo colaborativo, em rede, de forma descentralizada está no cerne do SIEP e representa uma condição indispensável para a consolidação do próprio SIEP e sua evolução, culminando na consolidação da Rede Nacional de Pesquisa e Inovação em Tecnologias Digitais - Renapi.

O SIEP possui Núcleos de desenvolvimento em diversos IF's (Institutos Federais) do país, inclusive no IF Fluminense, o que também representou uma das motivações para expandir o Redmine com a integração da Gerência de Requisitos.

Os projetos da Renapi que utilizam o Redmine são:

- Gerência Nacional: projeto de gestão das atividades administrativas da equipe da Gerência Nacional da Renapi.

- Acessibilidade Virtual: foi institucionalizado desde 2006 pelo Programa SIEP (Sistema de Informações da Educação Profissional e Tecnológica), hoje denominado Renapi (Rede Nacional de Pesquisa e Inovação) do Ministério da Educação. O objetivo desse trabalho é prover a acessibilidade virtual em *sites*, portais, objetos de aprendizagem, ambientes de educação à distância e sistemas *web*.

- Biblioteca Digital: tem como objetivo contribuir para a qualificação e a articulação das instituições de educação profissional, científica e tecnológica por meio de uma base de conhecimento que facilite a disseminação do material produzido pelos atores da Rede de EPCT, permitindo ainda a colaboração entre estes atores tendo esta base de conhecimento como alicerce. Este projeto é desenvolvido pelo Núcleo Campos (IFF - NCBD) e pelo Núcleo Teresina (IFPI - NTBD).

- Infraestrutura Renapi: suporte ao desenvolvimento de aplicações em todos os projetos da Renapi (Rede Nacional de Pesquisa e Inovação em Tecnologias Digitais). Este projeto tem como objetivo a manutenção dos serviços existentes para todos os outros projetos da Renapi bem como fazer prospecções tecnológicas, a fim de apontar soluções de infraestrutura para os gerentes de projeto.

- Observatório Nacional do Mundo do Trabalho e da EPCT: visa a estruturação uma rede de pesquisa, documentação e informações de referência que reflita de forma atualizada, permanente e contextualizada as várias dimensões do mundo do trabalho e de sua interação com a EPCT, utilizando um portal eletrônico como mecanismo de disseminação das informações e comunicação com o universo da EPCT e a sociedade em geral, assim como criar e gerenciar informações sobre pesquisa de realidades sócio-econômicas, de alunos/egressos e do mundo produtivo, gerando indicadores e análises.

- Portal da EPCT: projeto de confecção do Portal Nacional da EPT (temporariamente hospedado em <http://portal.renapi.org>).

- Qualidade (Quali): tem como objetivo principal a implantação da qualidade de software nos projetos da Renapi, segundo orientações do MPS.BR. Para isso, o Quali atua junto aos demais projetos, procurando interagir para que seja inserida a garantia da qualidade de software no fluxo de trabalho de cada um deles, respeitando as particularidades existentes.

- SIEP Gerencial: seu objetivo é prover a SETEC/MEC de instrumentos e ferramentas que possibilitem o exercício de sua função definidora de políticas, supervisora, estimulando um processo contínuo de avaliação, monitoramento, modernização e transparência da oferta e da expansão da Educação Profissional e Tecnológica no Brasil.

- SIGA-EPT: tem como objetivos desenvolver, implantar e dar suporte ao Sistema Integrado de Gestão Acadêmica – SIGA-EPT – como módulo do Sistema de Informações da Educação Profissional – SIEP – com código aberto, utilizando tecnologias de software livre, para prover as unidades acadêmicas supervisionadas pela SETEC/MEC de instrumentos e ferramentas que possibilitem sua gestão efetiva, tanto acadêmica quanto administrativa e garantir a integração das bases de dados locais com a SETEC/MEC.

De acordo com o site do Redmine, além dos projetos da Renapi, existem outros projetos em universidades que também utilizam o Redmine como gerenciador, dentre esses:

- Modelagem de Dados das IFES: tem por objetivo desenvolver e manter o modelo de referência de dados das IFES.

- Sistema de Biblioteca: projeto para estudo e possivelmente desenvolvimento de um Sistema Livre de Gestão de Biblioteca.

- Sistema de Saúde: visa à especificação de um Sistema de Saúde que possa ser adotado pelas áreas de saúde das IFES como, por exemplo, por hospitais universitários e por clínicas vinculadas a unidades acadêmicas e a órgãos administrativos.

- Extrator Lattes: desenvolvido pela USP e modificado pela UFAM para utilizar *PostgreSQL*.

- Estações Desktop Ubuntu: envolve a criação de rotinas para automatizar a configuração de estações de trabalho com Linux Ubuntu, bem como a disponibilização de materiais de apoio ao uso dessa distribuição Linux na universidade.

- GPS: Grupo de Processos de Software.

- SAMnet: Sistema de Administração de Materiais.

- SIGAD: Sistema Informatizado de Gestão Arquivística de Documentos.

- SOS: Sistema de Ordem de Serviços, desenvolvido pela equipe web da UFG.

- **THIS Gerenciador de Conteúdo:** Gerenciador de conteúdo multiportal, desenvolvido pela equipe *web* da UFG.
- **UNINET:** Novo portal e arcabouço de sistema de gestão universitária, planejado para integrar uma grande gama de módulos e ter uma interface *web* rica baseada em *Web OS*.
- **MCA:** Módulo de Controle de Acesso do UNINET.
- **Gerenciamento de Estações:** Sistema de gerenciamento de estações desenvolvido pela UFMS.

2.4.2 ALGUNS *PLUGINS* JÁ EXISTENTES

Alguns *plugins* já existentes para a plataforma Redmine se encontram a seguir. Essa informação foi retirada do site do Redmine.

- *BotsFilterPlugin*: previne *URLs* contra indexação de mecanismos de busca;
- *Bulk Time Entry*: adiciona múltiplas entradas de tempo de uma vez só;
- *Budget*: é um *plugin* para gerenciar o conjunto de resultados de cada projeto, automaticamente calcula alguns indicadores de desempenho;
- *Collapse*: transforma a barra lateral padrão do Redmine para uma barra expansível;
- *Customer*: gerencia informações básicas do cliente e analisa os projetos deste cliente;
- *Embedded*: permite enviar arquivos HTML (ex.: documentação auto-gerada, resultados de testes) em seu projeto;
- *Exception Handler*: habilita o Redmine a enviar *e-mails* quando algum erro ou exceção ocorrer;
- *ezFAQ*: gerenciador de *FAQ* para Redmine;
- *ezLibrarian*: cria um módulo que serve para a gestão de seus livros em sua empresa;
- *Google Analytics*: inclui de um modo simples o *Google Analytics* no Redmine;
- *Google Calendar*: integra o Redmine com o *Google Calendar*;
- *Project Scores*: permite ao usuário atribuir uma pontuação ao projeto, incluindo muitos critérios de análise;
- *RD formatter*: permite formatar texto no estilo *Ruby Doc*;
- *Question plugin*: que permite comunicação entre usuários nas tarefas;
- *Simple CI Stuff To Do*: permite o usuário atribuir a ordem das tarefas atribuídas a ele;

- *System Notification*: habilita o Administrador a enviar notificações por *e-mail* para usuários recém logados;
- *Timesheet plugin*: para análise de horas trabalhadas de cada usuário, por projeto;

2.5 FERRAMENTAS DE GERÊNCIA DE REQUISITOS DE SOFTWARE EXISTENTES

Algumas ferramentas Gerenciadoras de Requisitos de software foram testadas, com o objetivo de expor suas principais características e identificar pontos em aberto, em que a ferramenta proposta neste trabalho poderia atuar. As mesmas são descritas a seguir:

Enterprise Architect

É uma ferramenta de modelagem e construção baseada no UML 2.1 *standard*, de análises e design UML, cobrindo desenvolvimento de software, através dos estágios de análises, modelos de design, testes e manutenção. O software é uma ferramenta de gráfico, *Windows-based*, para multi- usuário, suporta o processo em um ambiente de forma fácil, rápida e flexível. O software proporciona rastreamento completo de requisitos e artefatos de design, através de organização e implementação, controle de versão com provedores CVS ou SCC. Gera matriz de rastreabilidade e cobre os artefatos referentes a requisitos, regras de negócio e casos de uso. Tem interface intuitiva com alta performance e visualização de projeto. Idioma: Inglês. Em relação à ferramenta proposta neste trabalho, apresenta os seguintes pontos em aberto: não é livre, não permite o cadastro do glossário do domínio e não está integrada a uma ferramenta de gerenciamento de tarefas, permitindo um relacionamento futuro do requisito com uma tarefa do projeto de forma automatizada.

Controla

Controla permite o Gerenciamento dos Requisitos, ou seja, regras e capacidades necessárias em um produto de software, utilizando como base descrição breve e completa da necessidade. Um requisito pode ser proposto por mais de um *stakeholder* (Cliente/Usuário), sendo assim, na ferramenta é possível associar um ou mais proponentes a um mesmo requisito. Realiza validações nos estados dos requisitos e possui controle de versão, rastreabilidade, e registro de métricas para todos os artefatos. Abrange apenas pequenas

empresas de desenvolvimento de software ou pequenas equipes, e não é integrada a nenhuma ferramenta de Gerência de Tarefas.

Requisite Manager

É uma ferramenta de Gerenciamento de Requisitos de software via *Web*, desenvolvida em PHP e utilizando banco de dados *MySQL*. As principais funcionalidades são: definição de tipos de requisitos; tipos de atributos; controle de requisitos; rastreabilidade; geração de relatórios; permite emitir o documento de requisitos de software obedecendo ao padrão IEEE/ANSI 830-1993, é voltada para o uso acadêmico.

Spres

É uma ferramenta CASE para especificação de requisitos de software, seguindo o modelo de documento IEEE-830-1998 e do Rational Unified Process (RUP). A ferramenta permite: criar atributos para identificar os requisitos; classificá-los por categoria; versões para o documento de Especificação de Requisitos de Software (ERS); matriz de rastreabilidade entre os elementos do ERS; cadastro do conteúdo das seções do documento de ERS; gerar o documento de ERS. Não é livre, e não é integrada a uma ferramenta de Gerência de Tarefas como o Redmine.

Raquest

É uma ferramenta de Gerenciamento de Requisitos desenvolvida por uma empresa filial da *Spark Systems* no Japão. Esta ferramenta é uma extensão do *Enterprise Architect*, sendo necessária a sua instalação do mesmo como requisito para utilização. Os principais recursos desta ferramenta são: as informações serão salvas no mesmo repositório do *Enterprise Architect*; os requisitos criados no *RaQuest* poderão ser alterados no *Enterprise Architect*; os requisitos poderão ser relacionados com os casos de uso criados no *Enterprise Architect*. Não é livre, e não é integrada a uma ferramenta de Gerência de Tarefas como o Redmine.

RequisitePro

É uma ferramenta desenvolvida pela IBM *Rational* Software, de Gerenciamento de Requisitos que possui os principais aspectos da Engenharia de Requisitos. Serve como um modelo para os requisitos; permite organizar requisitos por tipo em um nível mais alto; campo descritivo de informação associada ao requisito; Atributos podem ser definidos pelo Sistema ou pelo Usuário; Inclui a definição de formatação e texto pré-definido padrão. Não é livre e não é integrada a uma ferramenta de Gerência de Tarefas como o Redmine.

Borland Caliber Analyst

É uma ferramenta da *Borland*, que define e Gerência os Requisitos do Software durante todo o ciclo de vida da aplicação. Captura e especifica cenários visuais com facilidade, fornecendo uma linguagem comum a todos os participantes; Refina e valida os requisitos do software com uma execução de *storyboard*; Gera designs de UML, *test-case* e BPMN, aperfeiçoando a velocidade e precisão da engenharia de requisitos; gerencia e rastreia os requisitos do software por todo o ciclo de vida do projeto, avaliando o impacto e respondendo rapidamente a quaisquer mudanças nos requisitos. Não é livre, e não é integrada a uma ferramenta de Gerência de Tarefas como o Redmine.

2.6 CONSIDERAÇÕES FINAIS

Neste capítulo foi explicado o que é a Gerência de Requisitos, sua importância para o desenvolvimento de software e como funciona o processo de gerenciamento dos requisitos. Além disso, foram expostas características relevantes à Gerência de Requisitos de Software. Traz ainda um estudo realizado sobre ferramentas existentes para o Gerenciamento de Requisitos de Software, destacando suas características e uma comparação entre as mesmas, levando-se em consideração as características tidas como relevantes anteriormente. Dessa forma, percebe-se a necessidade de uma nova ferramenta desenvolvida para um ambiente integrado, livre, de código aberto, cobrindo características não encontradas nas ferramentas pesquisadas.

Concluiu-se que a ferramenta exposta como solução para os principais problemas relacionados à Gerência de Requisitos, realmente cumpre seu papel, pois supre a necessidade da elicitação clara das mudanças nos requisitos de software, e com a falta de habilidades

como as de saber fazer mudanças no sistema, quando isso se faz necessário, de estimar adequadamente os recursos para essas mudanças, e a consistência do documento de requisitos. Tudo isso deixa de ser um problema através da rastreabilidade dos requisitos, do glossário, e de todos os outros instrumentos de que essa ferramenta possui.

CAPÍTULO 3: FERRAMENTA PROPOSTA DE GERÊNCIA DE REQUISITOS DE SOFTWARE INTEGRADA À FERRAMENTA REDMINE

3.1 TECNOLOGIAS E ARQUITETURA

A ferramenta foi desenvolvida utilizando o framework *Ruby on Rails*, como um *plugin* do Redmine. Em todos os passos de seu desenvolvimento somente foram utilizados softwares livres, como o Ubuntu 9.04, o gEdit e o Aptana RadRails.

A ferramenta é totalmente baseada no padrão MVC de arquitetura de software, como pode ser visto na Figura 4. Este padrão possibilita a fácil manutenção e expansão das características do *plugin*.

Além disso, pode ser utilizada com vários tipos de Sistemas Gerenciadores de Banco de Dados, como o *MySQL*, o *PostgreSQL* e o *SQLite*, e com a maioria dos sistemas operacionais disponíveis no mercado. Esta flexibilidade só foi possível devido ao framework *Ruby on Rails*.

O *plugin* permite o cadastro de atores, termos de glossário, casos de uso, regras de negócio e requisitos de um software. Além disso, permite que seja feita a rastreabilidade entre essas entidades. Possibilita, também, a edição e a exclusão dos mesmos. Ele foi desenvolvido com o intuito de ser de fácil utilização, sem que seja necessário muito tempo de treinamento.

A ferramenta vincula a Gerência de Requisitos com os projetos cadastrados no Redmine, sem que um projeto tenha qualquer tipo de permissão em relação aos requisitos de outros projetos.

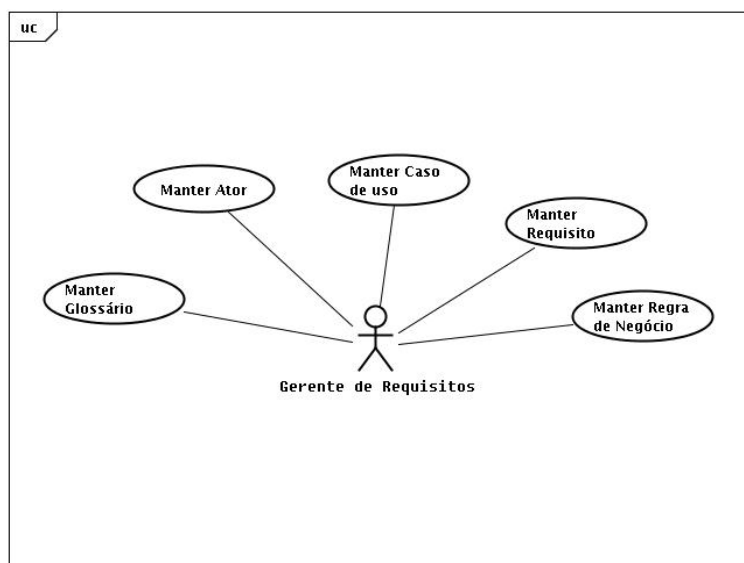


Figura 3 - Diagrama de Casos de Uso

As descrições dos casos de uso, demonstrados na Figura 3, podem ser encontradas no Apêndice 1.

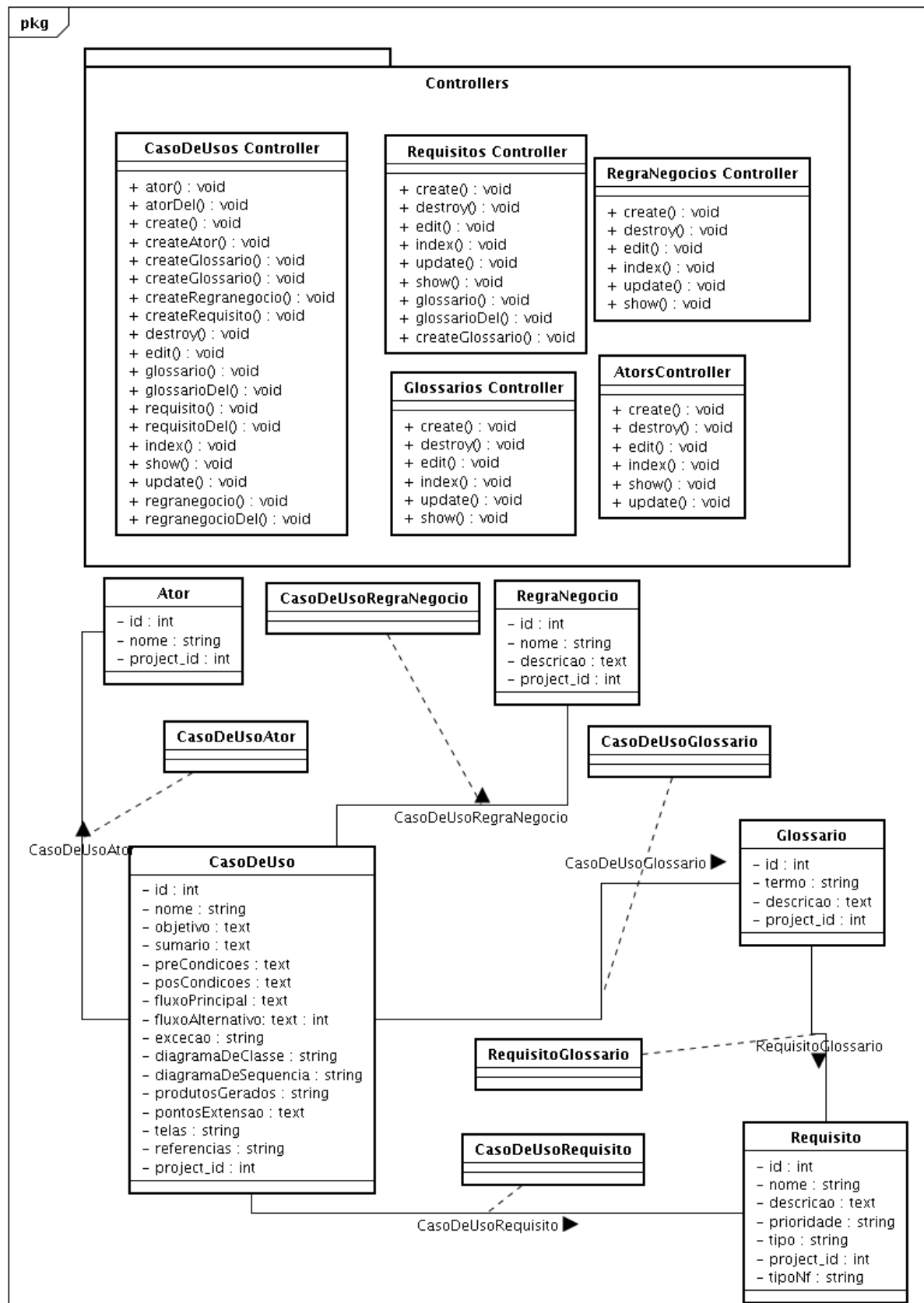


Figura 4 - Diagrama de Classes

3.1.1 DESCRIÇÃO DAS CLASSES

| Nome da classe | Descrição |
|-------------------------|--|
| CasoDeUsosController | Controla a ligação entre a classe CasoDeUso e as <i>views</i> relacionadas à mesma. |
| RequisitosController | Controla a ligação entre a classe Requisito e as <i>views</i> relacionadas à mesma. |
| RegraNegociosController | Controla a ligação entre a classe RegraNegocio e as <i>views</i> relacionadas à mesma. |
| AtorsController | Controla a ligação entre a classe Ator e as <i>views</i> relacionadas à mesma. |
| GlossariosController | Controla a ligação entre a classe Glossario e as <i>views</i> relacionadas à mesma. |
| Ator | Armazena os dados relacionados aos atores da ferramenta |
| CasoDeUso | Armazena os dados relacionados aos casos de uso da ferramenta |
| Glossario | Armazena os dados relacionados aos termos de glossário da ferramenta |
| Requisito | Armazena os dados relacionados aos requisitos da ferramenta |
| RegraNegocio | Armazena os dados relacionados às regras de negócio da ferramenta |
| CasoDeUsoAtor | Armazena a rastreabilidade entre ator e caso de uso |
| CasoDeUsoGlossario | Armazena a rastreabilidade entre glossário e caso de uso |
| CasoDeUsoRegraNegocio | Armazena a rastreabilidade entre regra de negócio e caso de uso |
| CasoDeUsoRequisito | Armazena a rastreabilidade entre requisito e caso de uso |
| RequisitoGlossario | Armazena a rastreabilidade entre glossário e requisito |

Tabela 1 - Descrição das classes da ferramenta

3.1.2 RUBY ON RAILS

Ruby on Rails é uma tecnologia que permite desenvolver sites dinâmicos orientados a banco de dados e aplicações para a web de forma semelhante a tantas outras linguagens de programação como PHP ou ASP.

É uma linguagem nova, porém tem chamado a atenção de desenvolvedores de todo o mundo, isso porque ele permite aumentar a velocidade e facilidade no desenvolvimento de projetos.

Rails é um *framework* criado na linguagem de programação *Ruby* (daí o nome *Ruby on Rails*).

O *Rails* foi criado com o intuito de permitir o desenvolvimento ágil, com alta produtividade, escrevendo poucas linhas de código e tendo muito resultado como consequência.

3.2 EXEMPLO DE USO DA FERRAMENTA PROPOSTA

Como exemplo, iremos utilizar o Caso de uso Manter Matriz Curricular, tendo como Ator o Responsável pelo Registro Acadêmico, o Requisito RACAD-2.001, a Regra de Negócio RACAD-2.001 e o termo de Glossário Matriz Curricular. Estes artefatos fazem parte do projeto SIGA.

3.2.1 DESCRIÇÃO DO CASO DE USO MANTER MATRIZ CURRICULAR

O objetivo desse caso de uso é descrever a manutenção de uma matriz curricular de um curso, bem como a manutenção de seus elementos curriculares. Uma matriz curricular corresponde à identificação do plano de curso. Já o plano de curso será composto pelos elementos curriculares associados à matriz curricular. Um elemento curricular poderá ter, dentro de uma matriz curricular, pré-requisitos e co-requisitos, que por sua vez são também elementos curriculares. Um curso poderá ter mais de uma matriz curricular, mas apenas uma matriz curricular vigente. Um elemento curricular de uma matriz curricular poderá ser equivalente a outro elemento curricular de outra matriz curricular de outro curso.

3.2.2 DESCRIÇÃO DO REQUISITO RACAD-2.001

O sistema deve permitir a manutenção (criação e atualização) de currículos para um curso, possivelmente com novas certificações ou habilitações, e possivelmente com outras matrizes curriculares associadas.

3.2.3 DESCRIÇÃO DA REGRA DE NEGÓCIO RACAD-2.001

Na matrícula, uma matriz curricular é associada a um aluno. Esta matriz curricular pode ser mudada desde que a instituição permita.

3.3 UTILIZANDO A FERRAMENTA PROPOSTA

Esta Seção trará explicações sobre o funcionamento da Ferramenta de Gerência de Requisitos, mostrando suas interfaces e funcionalidades. Logo após efetuar o *login* no Redmine e abrir o projeto desejado, o primeiro passo para abrir a ferramenta, é selecionar a aba Requisitos da página inicial do projeto no Redmine, conforme mostra a Figura 5:

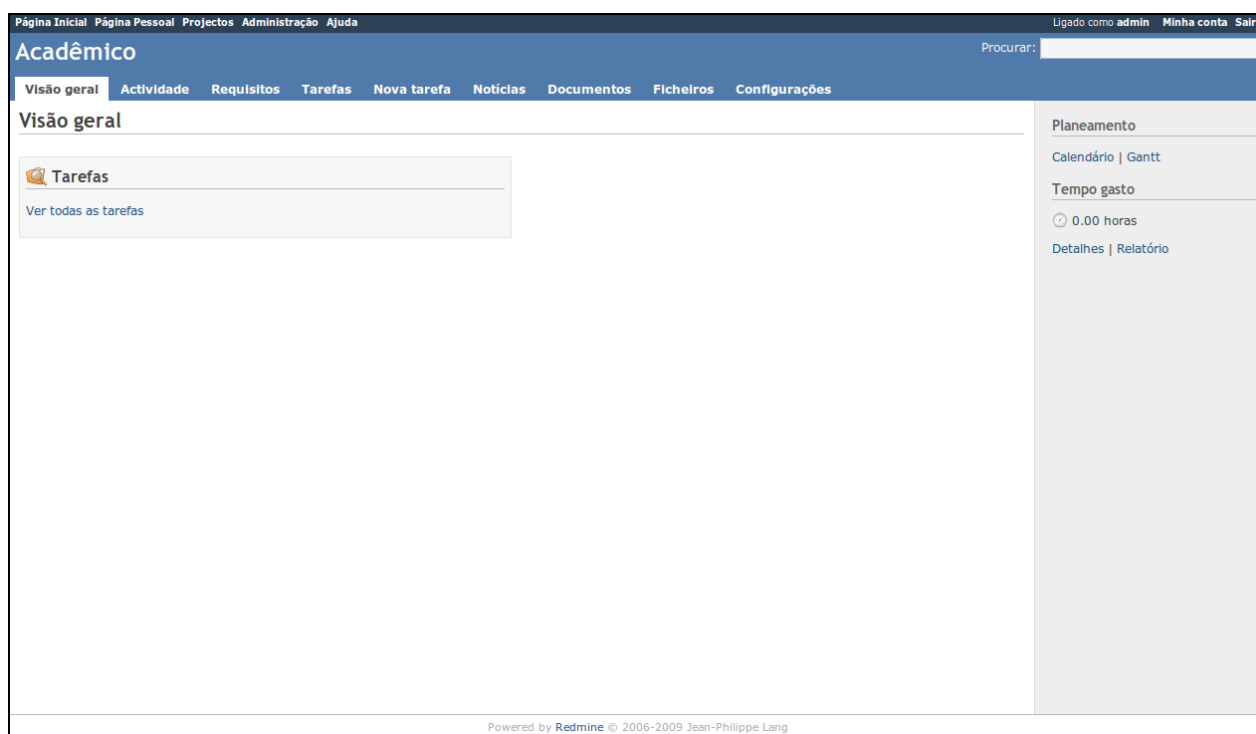


Figura 5 - Tela inicial do Redmine

Após selecionar a aba Requisitos, o sistema exibirá a tela inicial da ferramenta, como mostra a Figura 6. O *plugin* possui um menu com todos os itens para cadastro que fica à esquerda do mesmo, como pode ser visto na figura citada anteriormente. O próximo passo é cadastrar os artefatos no sistema.

Para que possa ser feita a rastreabilidade dos casos de uso com os demais itens, é necessário que os mesmos sejam cadastrados anteriormente. Esta afirmação também é válida em relação à rastreabilidade entre termos de glossário e requisitos, sendo necessário o cadastro de termos de glossário antes dos requisitos.

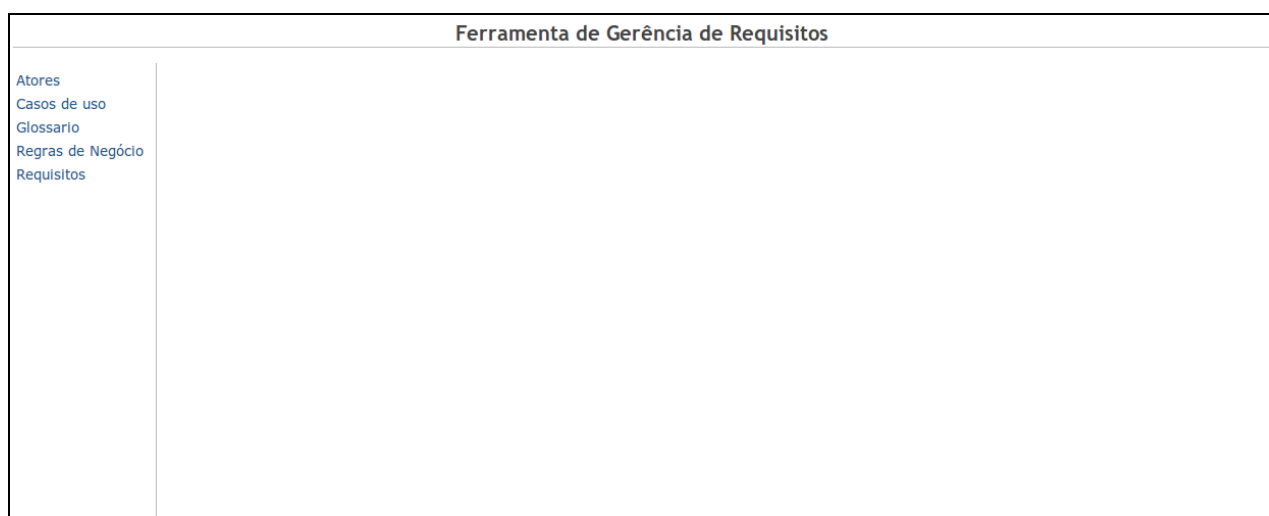


Figura 6 - Tela inicial da ferramenta

3.3.1 CADASTRANDO GLOSSÁRIO

Ao selecionar o *link* Glossário no menu da ferramenta, o sistema exibe a tela inicial de cadastro do mesmo, como mostra a Figura 7. Para inserir um novo item no glossário, deve-se clicar em inserir novo.

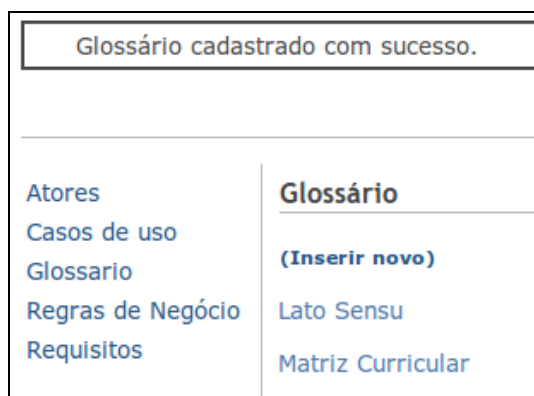


Figura 7 - Tela inicial de Glossário

Após clicar em inserir novo, o sistema exibirá a seguinte página, conforme mostra a Figura 8. O próximo passo é o usuário informar o termo e a descrição do novo item do glossário e clicar em cadastrar.

O campo termo deve ser preenchido com o nome do termo a ser descrito, neste caso temos o valor “Matriz Curricular”. Já o campo descrição, contém a descrição do mesmo, como num dicionário.

Este termo de glossário foi escolhido por estarmos exemplificando o caso de uso Manter Matriz Curricular, sendo de extrema importância a descrição correta de uma matriz curricular.

Figura 8 - Tela de cadastro do glossário

3.3.2 CADASTRANDO REQUISITO

Após cadastrar o glossário, é necessário que se cadastre os requisitos. Para o cadastro de requisitos deve-se seguir os mesmos passos descritos para cadastro de glossário.

O campo nome deve ser preenchido com o nome do requisito, neste caso utilizou-se “RACAD-2.001”. O campo descrição visa descrever o requisito. A prioridade contém os valores Alta, Média e Baixa e deve ser utilizado para definir o quão prioritário é o requisito. O campo tipo de requisito é utilizado para definir se o mesmo é funcional ou não funcional. O campo Tipo de Requisito Não-Funcional, só deve ser utilizado quando for escolhido um requisito não-funcional. Os valores possíveis para este campo são Desempenho, Usabilidade, Confiabilidade, Segurança, Disponibilidade, Portabilidade e Manutenibilidade.

Figura 9 - Tela de cadastro dos Requisitos

3.3.3 DEFININDO A RASTREABILIDADE ENTRE REQUISITO E GLOSSÁRIO

Como exemplo, iremos utilizar o requisito RACAD-2.001, que foi citado anteriormente, e o termo de glossário Matriz Curricular. Para fazer a rastreabilidade desse

requisito, basta clicar nele na página inicial de Requisitos. O sistema exibirá a tela, como mostra a Figura 10.

| | |
|--|--|
| Atores Casos de uso Glossario Regras de Negócio Requisitos | RACAD-2.001 Descrição: O sistema deve permitir a manutencao (criacao e atualizacao) de currículos para um curso, possivelmente com novas certificações ou habilitações, e possivelmente com outras matrizes curriculares associadas. Prioridade: Alta Tipo: Funcional Tipo Não Funcional: <div> Editar Excluir </div> Rastreabilidade: Glossário |
|--|--|

Figura 10 - Tela de exibição de um requisito

No rodapé da página da Figura 10, há um botão chamado Glossário, que permite a rastreabilidade entre o mesmo com o requisito selecionado anteriormente. Ao clicar nesse botão, o sistema exibirá a tela para vinculá-los, conforme a Figura 11.

Rastreabilidade do requisito RACAD-2.001

Inserir nova rastreabilidade

Matriz Curricular ▾

Cadastrar

Voltar

Figura 11 - Tela de rastreabilidade do Requisito RACAD-2.001 com termos de glossário

A caixa de seleção trará todos os glossários cadastrados no sistema. Selecione o termo de glossário Matriz Curricular para vincular ao requisito RACAD-2.001, e clique no botão cadastrar. Aparecerá na tela inicial da rastreabilidade o novo relacionamento.

3.3.4 CADASTRANDO ATORES

O próximo passo é inserir os atores no sistema, pois eles são necessários para a rastreabilidade dos casos de uso.

O campo nome deve conter o nome do ator a ser cadastrado no sistema.

| | |
|--|--|
| Atores Casos de uso Glossario Regras de Negócio Requisitos | Inserir novo ator Nome: <input type="text" value="Responsável pelo registro acadêmico"/> <div>Cadastrar</div> |
|--|--|

Figura 12 - Tela de cadastro de atores

3.3.5 CADASTRANDO REGRAS DE NEGÓCIO

O campo nome contém o nome da regra de negócio a ser cadastrado no sistema. O campo descrição deve ser preenchido com a descrição da mesma.

Figura 13 - Tela de cadastro das Regras de Negócio

3.3.6 CADASTRANDO CASOS DE USO

Segue a descrição de cada um dos campos do *template* de cadastro de caso de uso:

- Nome contém o nome do caso de uso.
- Objetivo serve para descrever resumidamente o objetivo do caso de uso, ou seja, a função do Sistema que será implementada.
- Sumário é um breve resumo do que será o caso de uso.
- Pré-condições é a condição essencial para que o Caso de Uso possa ser realizado.
- Fluxo Principal é o cenário do nosso caso de uso em que tudo se realizará normalmente.
- Fluxo Alternativo deve tratar de todos os fluxos que não obedecem à forma normal de execução, isto é, são eventos alternativos do caso de uso, configurando cenários diferentes ou tratando outras funcionalidades, como, por exemplo, uma alteração ou uma exclusão.
- Pós-condições é o que deve ser verdadeiro quando o processo terminar com sucesso, mostrando o resultado após a execução do caso de uso.
- Exceções contém situações que possam comprometer o bom funcionamento do sistema em questão.
- Diagrama de classe, Diagrama de Sequência e Telas devem conter o link para os mesmos.
- Produtos Gerados é uma descrição dos produtos gerados após a execução do caso de uso.
- Pontos de Extensão é uma indicação de que existe um relacionamento de extensão entre casos de uso.

- Referências contém a bibliografia em que se baseia o CDU.

Página Inicial | Página Pessoal | Projectos | Administração | Ajuda

Ligado como admin | Minha conta | Sair

Acadêmico

Procurar:

Visão geral | Actividade | Requisitos | Tarefas | Nova tarefa | Notícias | Documentos | Ficheiros | Configurações

Ferramenta de Gerência de Requisitos

Atores
Casos de uso
Glossário
Regras de Negócio
Requisitos

Inserir novo caso de uso

Nome:
Manter Matriz Curricular

Objetivo:
matriz curricular vigente. Um elemento curricular de uma matriz curricular poderá ser equivalente a outro elemento curricular de outra matriz curricular de outro curso. A manutenção de um elemento curricular é descrita no SIGA-EDU-CDU-RACAD-010 Manter Elemento Curricular.

Sumário:
N/A

Pré-condições:
Para manter uma matriz curricular, pelo menos um curso e um elemento curricular já devem ter sido cadastrados.

Fluxo Principal:
P15. O Responsável pelo registro acadêmico confirma a inclusão da matriz curricular clicando no botão gravar.
P16. O sistema valida os dados digitados.
P17. O sistema inclui a matriz curricular e exibe a mensagem No 01.

Fluxo Alternativo:
10. O sistema valida os dados informados.
11. O sistema exclui o elemento curricular para a matriz em questão e exibe a mensagem No 10.
A07[P01] Manutenção de elemento curricular equivalente da matriz curricular.

Exceções:
E01[P15] Dados obrigatórios não informados.
1. O sistema exibe a mensagem No 02 e retorna ao passo de origem.
E02[A01.9] A data fim de vigência informada não é válida

Pós-condições:
Uma instância de matriz curricular é mantida com seus respectivos elementos curriculares.

Diagrama de classe:
unk/fenix/docs/cdus/SIGA-EDU-CDU-RACAD-010/

Diagrama de sequência:
unk/fenix/docs/cdus/SIGA-EDU-CDU-RACAD-010/

Produtos Gerados:
N/A

Pontos de Extensão:
Nenhum até o momento

Telas:
EDU-CDU-RACAD-010-ManterMatrizCurricular.odt

Referências:
N/A

Cadastrar

Powered by Redmine © 2006-2009 Jean-Philippe Lang

Figura 14 - Tela de cadastro do Caso de uso

3.3.6 RASTREABILIDADES DOS CASOS DE USO

Os Casos de uso serão vinculados a atores, itens do glossário, regras de negócio e requisitos. Segue o exemplo do Caso de uso Manter Matriz Curricular com o ator

Responsável pelo Registro Acadêmico. Para fazer a rastreabilidade com os outros itens, deve-se realizar os passos descritos na próxima seção.

3.3.7 DEFININDO RASTREABILIDADE DE CASOS DE USO X GLOSSÁRIO

Para fazer a rastreabilidade do Caso de uso explicitado, basta abri-lo no sistema e clicar em Glossário, no final da página. O sistema exibirá a tela, como mostra a Figura 15.

Figura 15 - Tela de rastreabilidade do Caso de uso Manter Matriz com o termo do Glossário Matriz Curricular.

A caixa de seleção trará todos os itens do glossário cadastrados no sistema. Selecione o termo de glossário Matriz Curricular para vincular ao Caso de uso Manter Matriz Curricular, e clique no botão cadastrar. Aparecerá na tela inicial da rastreabilidade o novo relacionamento, como mostra a Figura 16:

Figura 16 - Tela inicial da rastreabilidade com o glossário vinculado ao Caso de Uso

Para definir as rastreabilidades de um caso de uso com atores, regras de negócio e requisitos, basta seguir os passos descritos para glossário. Para visualizar, basta abrir o caso

de uso e clicar no final da tela no tipo de rastreabilidade desejada, como pode ser visto na Figura 17.



Figura 17 - Links para visualização das rastreabilidades

3.4 CONSIDERAÇÕES FINAIS

Neste capítulo foi explicada a estrutura da ferramenta, sua arquitetura e os padrões seguidos. Além disso, foi feita uma breve descrição da linguagem utilizada. Também foram mostrados os diagramas de classe e de casos de uso, com suas devidas descrições. Por fim, uma breve explicação de como utilizar a ferramenta e quais benefícios a mesma pode trazer ao integrar o preenchimento de requisitos e casos de uso a um software consolidado como o Redmine. Podemos também mostrar que a geração de rastreabilidade pode tornar o desenvolvimento de softwares muito mais eficiente.

A Tabela 2 traz a comparação entre as ferramentas existentes, incluindo a Gerência de Requisitos, que foi apresentada neste trabalho. A coluna Glossário define se a ferramenta possui ou não o controle de glossário, já que a maioria das ferramentas não possuem esta característica. Na coluna posterior, identificamos quais ferramentas possuem integração com ferramentas de gerência de tarefas. Em Rastreabilidade, definimos quais ferramentas tem essa funcionalidade, de extrema importância para esta modalidade de ferramenta. As duas outras colunas foram escolhidas para demonstrar que só a nossa ferramenta é livre e somente outra ferramenta também é gratuita.

| | Glossário | Integração com Ferramenta de Gerência de Tarefas | Rastreabilidade | Software livre | Gratuita |
|-------------------------|-----------|--|-----------------|----------------|----------|
| Enterprise Architect | Não | Não | Sim | Não | Não |
| Controla | Não | Não | Sim | Não | Sim |
| Requisite Manager | Não | Não | Sim | Não | Não |
| SPRES | Não | Não | Sim | Não | Não |
| Raquest | Sim | Não | Sim | Não | Não |
| RequisitePro | Sim | Não | Sim | Não | Não |
| Borland Caliber Analyst | Não | Não | Sim | Não | Não |
| Gerência de Requisitos | Sim | Sim | Sim | Sim | Sim |

Tabela 2 - Comparação entre as ferramentas testadas

CAPÍTULO 4: CONCLUSÕES

4.1 CONTRIBUIÇÕES

Ao término deste trabalho, pode-se perceber que a Gerência de Requisitos pode ter um papel crucial na qualidade de um software, pois como os requisitos representam a base de todo o software, o seu controle automatizado traz uma série de benefícios, como facilidade de atualização, visualização e integração com outras atividades da Engenharia de Software. Dessa forma, é possível avaliar melhor o real funcionamento de sistemas de software desenvolvidos e os reais impactos causados por alterações de requisitos.

A maior contribuição deste trabalho consiste em possibilitar uma melhor integração da Gerência de Projetos com a Gerência de Requisitos em um ambiente integrado. Esta integração possibilita que os profissionais da área de desenvolvimento de software utilizem menos ferramentas durante o ciclo de vida do software, ganhando produtividade no desenvolvimento. Esta menor quantidade de ferramentas possibilita também a redução de custos do desenvolvimento de software, já que se evita retrabalho e perde-se menos tempo com o aprendizado de ferramentas distintas.

Além disso, a ferramenta contribui com um melhor controle de impactos de manutenções ou mudanças de requisitos, através da automatização dos cadastros e estabelecimento de rastreabilidade entre os itens.

Outro ponto a destacar neste trabalho de conclusão é que durante o desenvolvimento da ferramenta ficou claro o quanto pode ser complicado e trabalhoso utilizar tecnologias que até então não conhecíamos. Além disso, por serem tecnologias novas, torna-se dispendioso conseguir encontrar profissionais com conhecimento para prestar suporte quando encontram-se dificuldades maiores.

4.2 APRIMORAMENTO DA FERRAMENTA

A limitação da ferramenta está no fato de a mesma ainda não conseguir importar diagramas de casos de uso desenvolvidos em outras ferramentas CASE, como o JUDE ou o Rational Rose.

Além disso, a mesma não exporta a rastreabilidade em formatos como PDF, HTML e ODT.

Seguem os trabalhos futuros que podem ser desenvolvidos como prolongamento e enriquecimento do presente trabalho.

Incorporar a importação de casos de uso em XMI

É importante que a ferramenta possa fazer a importação do formato XMI, para que projetos que já estejam em estágios mais avançados possam rapidamente incorporar seus dados à ferramenta. Dessa forma, a descrição do caso de uso pode ser vinculada diretamente ao diagrama, além de ser possível expandir a rastreabilidade do requisito e caso de uso para classes e métodos.

Exportar a matriz de rastreabilidade em outros formatos

Para que os profissionais de informática possam interpretar melhor a rastreabilidade e realizar uma adequada análise de impacto de manutenções, seria interessante que os dados fossem exportados para formatos como PDF, HTML, ODT etc., facilitando a sua impressão e visualização.

Realizar buscas

Seria interessante que o sistema possibilitasse que o usuário fizesse buscas no mesmo para encontrar casos de uso ou itens em geral de acordo com palavras-chave.

Rastreabilidade automática de glossário

Uma possível nova funcionalidade a ser adicionada seria que, ao criar casos de uso, regras de negócio e requisitos, a ferramenta conseguisse fazer a ligação automática dos mesmos com os termos de glossário, através de hiperlinks para os itens cadastrados no glossário e que aparecem no texto do requisito, caso de uso e regra de negócio.

Relacionar tarefas com caso de uso

A ferramenta futuramente deve poder relacionar os casos de uso com as tarefas do Redmine, permitindo, assim, uma melhor integração entre as áreas da Engenharia de

Software, como Gerência de Projetos e Gerência de Requisitos. Atualmente, quando uma tarefa é cadastrada no Redmine é possível se fazer o *upload* de um arquivo referente ao caso de uso. No futuro, a ligação deverá ser via Redmine para o caso de uso cadastrado, permitindo que, ao clicar em uma tarefa, seja possível abrir o caso de uso sem precisar utilizar outro ambiente como um editor de texto e que, ainda, seja possível a partir do caso de uso navegar para os seus itens de rastreabilidade como requisito, regra de negócio e termos do glossário.

Incorporar a Gerência de Configuração à ferramenta

Futuramente, seria interessante que a ferramenta pudesse realizar a Gerência de Configuração, tornando possível que o usuário pudesse restaurar os artefatos para versões antigas, caso alguma alteração dos mesmos estivesse incorreta ou não fosse mais necessária. Além disso, poderia possibilitar ao mesmo uma melhor visualização da evolução dos requisitos durante o ciclo de vida do software.

REFERÊNCIAS

- [Pressman, 2002] PRESSMAN, R. S. “*Engenharia de Software*”, Trad. 5.ed. São Paulo: McGraw Hill, 2002.
- [SOMMERVILLE, 2003] SOMMERVILLE, Ian. “*Engenharia de Software*”. 6ª ed. São Paulo: Pearson Addison Wesley, 2003.
- [wikipedia, 2009] wikipedia. Disponível em: <<http://pt.wikipedia.org/wiki/CVS>>. Acessado em Setembro de 2009.
- [wikipedia, 2009] wikipedia. Disponível em: <<http://pt.wikipedia.org/wiki/Subversion>>. Acessado em Setembro de 2009.
- [wikipedia, 2009] wikipedia. Disponível em: <<http://pt.wikipedia.org/wiki/LDAP>>. Acessado em Setembro de 2009.
- [wikipedia, 2009] wikipedia. Disponível em: <http://pt.wikipedia.org/wiki/GNU_General_Public_License>. Acessado em Setembro de 2009.
- [wikipedia, 2009] wikipedia. Disponível em: <<http://pt.wikipedia.org/wiki/Git>>. Acessado em Setembro de 2009.
- [riopro, 2009] riopro. Disponível em: <<http://blog.riopro.com.br/2008/06/18/a-opcao-pelo-redmine/>>. Acessado em Setembro de 2009.
- [Redmine, 2009] Redmine. Disponível em: <http://www.redmine.org/wiki/redmine/Plugin_Tutorial>. Acessado em Setembro de 2009.
- [vebersol, 2009] vebersol. Disponível em: <<http://vebersol.net/2009/02/25/dicas-para-otimizar-o-redmine/>>. Acessado em Setembro de 2009.
- [wikipedia, 2009] wikipedia. Disponível em: <http://pt.wikipedia.org/wiki/Project_Management_Body_of_Knowledge>. Acessado em Setembro de 2009.
- [cosmoverbal, 2009] cosmoverbal. Disponível em: <<http://www.cosmoverbal.net/tag/redmine>>. Acessado em Setembro de 2009.
- [wikipedia, 2009] wikipedia. Disponível em: <<http://pt.wikipedia.org/wiki/Redmine>>. Acessado em Setembro de 2009.
- [xandrix, 2009] xandrix. Disponível em: <<http://www.xandrix.com.br/solucoes/redmine>>. Acessado em Setembro de 2009.

- [ufg, 2009] ufg. Disponível em: <<http://www.ufg.br/this2/uploads/files/17/redmine-article.pdf>>. Acessado em Setembro de 2009.
- [danilosbrasil, 2009] danilosbrasil. Disponível em: <<http://danilosbrasil.wordpress.com/>>. Acessado em Setembro de 2009.
- [wthreex, 2009] wthreex. Disponível em: <http://www.wthreex.com/rup/process/workflow/requirem/co_rm.htm>. Acessado em Outubro de 2009.
- [bfpug, 2009] bfpug. Disponível em: <<http://www.bfpug.com.br/islig-rio/Downloads/Ger%C3%A4ncia%20de%20Requisitos-o%20Principal%20Problema%20dos%20Projetos%20de%20SW.pdf>>. Acessado em Outubro de 2009.
- [usp, 2009] usp. Disponível em: <<http://www.poli.usp.br/pro/procsoft/tproepusp05.pdf>>. Acessado em Outubro de 2009.
- [puc-rio, 2009] puc-rio. Disponível em: <http://wer.inf.puc-rio.br/WERpapers/artigos/artigos_WER03/claudia_hazan.pdf>. Acessado em Outubro de 2009.
- [fapesp, 2009] fapesp. Disponível em: <<http://sysreq.incubadora.fapesp.br/portal/down/lite/gr>>. Acessado em Outubro de 2009.
- [puc-rio, 2009] puc-rio. Disponível em: <<http://www.inf.puc-rio.br/~wer01/Pro-Req-3.pdf>>. Acessado em Outubro de 2009.
- [bruno cunha, 2009] bruno cunha. Disponível em: <<http://pt.wikipedia.org/wiki/CVS>>. Acessado em Outubro de 2009.
- [agrosol, 2009] agrosol. Disponível em: <<http://www.agrosol.com.br/borland/caliberrm.shtml#Ferramenta%20utilizada>>. Acessado em Outubro de 2009.
- [borland, 2009] borland. Disponível em: <<http://www.borland.com/br/products/caliber/index.html>>. Acessado em Outubro de 2009.
- [borland, 2009] borland. Disponível em: <<http://www.borland.com/br/products/teamdefine/index.html>>. Acessado em Outubro de 2009.
- [via6, 2009] via6. Disponível em: <<http://www.via6.com/comunidade.php?cid=12006>>. Acessado em Outubro de 2009.

- [bruno cunha, 2009] bruno cunha. Disponível em: <<http://www.brunocunha.com/blog/?p=679>>. Acessado em Outubro de 2009.
- [bruno cunha, 2009] bruno cunha. Disponível em: <<http://www.brunocunha.com/blog/?p=284>>. Acessado em Outubro de 2009.
- [bruno cunha, 2009] bruno cunha. Disponível em: <http://www.brunocunha.com/anexos/Monografia_PUC-RIO_Bruno_Cunha_versao_final.pdf>. Acessado em Outubro de 2009.
- [borland, 2009] borland. Disponível em: <<http://www.borland.com/br/products/caliber/defineit.html>>. Acessado em Outubro de 2009.
- [administradores, 2009] administradores. Disponível em: <http://www.administradores.com.br/producao_academica/uma_proposta_de_escritorio_de_projetos_corporativo/2128/>. Acessado em Outubro de 2009.
- [usp, 2009] usp. Disponível em: <http://www.pcs.usp.br/~laa/html/rastreabilidade2005/leo_denise.pdf>. Acessado em Outubro de 2009.
- [sbc, 2009] sbc. Disponível em: <<http://www.sbc.org.br/bibliotecadigital/download.php?paper=983>>. Acessado em Outubro de 2009.
- [engenharia de software, 2009] engenharia de software. Disponível em: <<http://engenhariasoftware.wordpress.com/2009/05/01/a-rastreabilidade-dos-requisitos-como-importante-fator-da-garantia-de-qualidade/>>. Acessado em Outubro de 2009.
- [ines, 2009] ines. Disponível em: <<http://www.ines.org.br/?tag=rastreabilidade-de-requisitos>>. Acessado em Outubro de 2009.
- [simpros, 2009] simpros. Disponível em: <http://www.simpros.com.br/simpros2003/upload/arquivos_PDF/Artigos/ART_04.pdf>. Acessado em Outubro de 2009.
- [wikipedia, 2009] wikipedia. Disponível em: <<http://pt.wikipedia.org/wiki/Rastreabilidade>>. Acessado em Outubro de 2009.
- [slideshare, 2009] slideshare. Disponível em: <<http://www.slideshare.net/transparenciadesoftware/rastreabilidade-de-requisitos>>. Acessado em Outubro de 2009.

- [e-learningtec, 2009] e-learningtec. Disponível em: <http://www.e-learningtec.com.br/index.php?option=com_content&view=article&id=14:rastreabilidaderesquisitos&catid=1:engenhariasoftware&Itemid=2>. Acessado em Outubro de 2009.
- [e-learningtec, 2009] e-learningtec. Disponível em: <http://www.e-learningtec.com.br/index.php?option=com_content&view=article&id=14:rastreabilidaderesquisitos&catid=1:engenhariasoftware&Itemid=2>. Acessado em Outubro de 2009.
- [e-learningtec, 2009] e-learningtec. Disponível em: <http://www.e-learningtec.com.br/index.php?option=com_content&view=article&id=11:gerrequeamento&catid=1:engenhariasoftware&Itemid=2>. Acessado em Outubro de 2009.
- [e-learningtec, 2009] e-learningtec. Disponível em: <http://www.e-learningtec.com.br/index.php?option=com_content&view=article&id=14:rastreabilidaderesquisitos&catid=1:engenhariasoftware&Itemid=2>. Acessado em Outubro de 2009.
- [emagno, 2009] emagno. Disponível em: <<http://emagno.blogspot.com/>>. Acessado em Outubro de 2009.
- [concursospublicosonline, 2009] concursospublicosonline. Disponível em: <<http://www.concursospublicosonline.com/informacao/view/Apostilas/Metodologia-de-Desenvolvimento-de-Sistemas/Gerenciamento-de-Requisitos/>>. Acessado em Outubro de 2009.
- [renapi, 2009] renapi. Disponível em: <<http://www.renapi.org/qualidade/metodologia/gerencia-de-requisitos>>. Acessado em Outubro de 2009.
- [puc-rio, 2009] puc-rio. Disponível em: <http://wer.inf.puc-rio.br/WERpapers/artigos/artigos_WER03/claudia_hazan.pdf>. Acessado em Outubro de 2009.
- [ufpe, 2009] ufpe. Disponível em: <www.cin.ufpe.br/~ebs/flatpress/fp.../2-ferramenta-requisitepro.ppt>. Acessado em Outubro de 2009.
- [nix, 2009] nix. Disponível em: <<http://www.nix.com.br/modules/smartsection/item.php?itemid=18>>. Acessado em Outubro de 2009.
- [viva o linux, 2009] viva o linux. Disponível em: <<http://www.vivaolinux.com.br/artigo/SIGERAR-Sistema-de-Gerenciamento-de-Requisitos/>>. Acessado em Outubro de 2009.

- [viva o linux, 2009] viva o linux. Disponível em: <<http://www.vivaolinux.com.br/artigo/SIGERAR-Sistema-de-Gerenciamento-de-Requisitos/?pagina=2>>. Acessado em Outubro de 2009.
- [viva o linux, 2009] viva o linux. Disponível em: <<http://www.vivaolinux.com.br/artigo/SIGERAR-Sistema-de-Gerenciamento-de-Requisitos/?pagina=3>>. Acessado em Outubro de 2009.
- [wthreex, 2009] wthreex. Disponível em: <http://www.wthreex.com/rup/process/workflow/requirem/co_trace.htm>. Acessado em Outubro de 2009.
- [engenharia de software brasil, 2009] engenharia de software brasil. Disponível em: <<http://engenhariadesoftwarebrasil.blogspot.com/2007/11/ferramentas-de-apoio-engenharia-de.html>>. Acessado em Outubro de 2009.
- [qualitnews, 2009] qualitnews. Disponível em: <http://qualitnews.blogspot.com/2008/11/ferramenta-gratuita-para-gerencia-de_23.html>. Acessado em Outubro de 2009.
- [ibm, 2009] ibm. Disponível em: <<http://www-142.ibm.com/software/products/br/pt/reqpro>>. Acessado em Outubro de 2009.
- [ibm, 2009] ibm. Disponível em: <<http://www-142.ibm.com/software/products/br/pt/datamodeler>>. Acessado em Outubro de 2009.
- [ufpe, 2009] ufpe. Disponível em: <<http://www.cin.ufpe.br/~if119/aulas/cap5.PDF>>. Acessado em Outubro de 2009.
- [artigonal, 2009] artigonal. Disponível em: <<http://www.artigonal.com/programacao-artigos/a-gestao-de-requisitos-335358.html>>. Acessado em Outubro de 2009.
- [puc-rio, 2009] puc-rio. Disponível em: <http://wer.inf.puc-rio.br/WERpapers/artigos/artigos_WER00/zanlorenzi.pdf>. Acessado em Outubro de 2009.
- [scribd, 2009] scribd. Disponível em: <<http://www.scribd.com/doc/2152182/Aplicando-Gerenciamento-de-Requisitos-com-Casos-de-Uso>>. Acessado em Outubro de 2009.
- [200.169.53.89, 2009] 200.169.53.89. Disponível em: <http://200.169.53.89/download/CD%20congressos/2004/Sucesu/artigos_cientif/RequisitManager.PDF>. Acessado em Outubro de 2009.

- [baguete, 2009] baguete. Disponível em: <<http://www.baguete.com.br/artigosDetalhes.php?id=90>>. Acessado em Outubro de 2009.
- [campeche, 2009] campeche. Disponível em: <<http://campeche.inf.furb.br/tccs/2007-I/2007-1raphaelmarcosbatistavf.pdf>>. Acessado em Outubro de 2009.
- [superdownloads, 2009] superdownloads. Disponível em: <<http://superdownloads.uol.com.br/download/118/enterprise-architect/>>. Acessado em Outubro de 2009.
- [mhavila, 2009] mhavila. Disponível em: <www.mhavila.com.br/topicos/gestao/pmbok.html>. Acessado em Outubro de 2009.
- [ubuntu, 2010] ubuntu. Disponível em: <www.ubuntu.org>. Acessado em Janeiro de 2010.
- [gnome, 2010] gnome. Disponível em: <<http://projects.gnome.org/gedit/>>. Acessado em Janeiro de 2010.
- [aptana, 2010] aptana. Disponível em: <<http://www.radrails.org/>>. Acessado em Janeiro de 2010.
- [rubyonrails, 2010] rubyonrails. Disponível em: <rubyonrails.org/>. Acessado em Janeiro de 2010.

APÊNDICE 1: **DESCRIÇÃO DOS CASOS DE USO DA FERRAMENTA PROPOSTA**

Caso de Uso Manter Glossário

- **Descrição:** Descreve o processo de cadastro, alteração e exclusão de um glossário no sistema.

- Pré- Condição: Não Aplicado
- **Ator:** Gerenciador de requisitos
- Fluxo Principal:

P01. O Gerenciador de requisitos solicita o procedimento para cadastrar um glossário no sistema.

P02. O Sistema exibe a tela de cadastro de glossário.

P03. O Gerenciador de requisitos preenche os campos.

P04. O Sistema confirma o cadastro.

- Fluxo Alternativo:

A01. A operação solicitada é a alteração de um glossário

- O Gerenciador de requisitos seleciona o glossário que deseja alterar.
- O Gerenciador de requisitos seleciona a opção editar.
- O Sistema exibe a tela para a alteração.
- O Gerenciador de requisitos realiza as alterações necessárias.
- O Sistema exibe uma mensagem confirmando a alteração.

A02. A operação solicitada é a exclusão de um glossário

- O Gerenciador de requisitos seleciona o glossário que deseja excluir.
- O Gerenciador de requisitos seleciona a opção excluir.
- O Sistema exibe uma mensagem perguntando ao Gerenciador se realmente deseja excluir o glossário.
- O Gerenciador de requisitos confirma a exclusão.

- **Exceção:** Não aplicado
- **Pós- Condição:** Glossário cadastrado no sistema.
- Produtos Gerados: Não aplicado
- Pontos de Extensão: Não aplicado

Caso de Uso Manter Requisitos

- **Descrição:** Descreve o processo de cadastro, alteração e exclusão de um requisito no sistema.

- Pré- Condição: Não Aplicado
- **Ator:** Gerenciador de requisitos

- **Fluxo Principal:**

P01. O Gerenciador de requisitos solicita o procedimento para cadastrar um requisito no sistema.

P02. O Sistema exibe a tela de cadastro de requisitos.

P03. O Gerenciador de requisitos preenche os campos.

P04. O Sistema confirma o cadastro.

- **Fluxo Alternativo:**

A01. A operação solicitada é a alteração de um requisito

- O Gerenciador de requisitos seleciona o requisito que deseja alterar.
- O Gerenciador de requisitos seleciona a opção editar.
- O Sistema exibe a tela para a alteração.
- O Gerenciador de requisitos realiza as alterações necessárias.
- O Sistema exibe uma mensagem confirmando a alteração.

A02. A operação solicitada é a exclusão de um requisito

- O Gerenciador de requisitos seleciona o requisito que deseja excluir.
- O Gerenciador de requisitos seleciona a opção excluir.
- O Sistema exibe uma mensagem perguntando ao Gerenciador se realmente deseja excluir o requisito.
- O Gerenciador de requisitos confirma a exclusão.

- **Exceção:** Não aplicado
- **Pós- Condição:** Requisito cadastrado no sistema.
- **Produtos Gerados:** Não aplicado
- **Pontos de Extensão:** Não aplicado

Caso de Uso Manter Ator

- **Descrição:** Descreve o processo de cadastro, alteração e exclusão de um ator no sistema.

- **Pré- Condição:** Não Aplicado
- **Ator:** Gerenciador de requisitos
- **Fluxo Principal:**

P01. O Gerenciador de requisitos solicita o procedimento para cadastrar um ator no sistema.

P02. O Sistema exibe a tela de cadastro de atores.

P03. O Gerenciador de requisitos preenche os campos.

P04. O Sistema confirma o cadastro.

- Fluxo Alternativo:

A01. A operação solicitada é a alteração de um ator

- O Gerenciador de requisitos seleciona o ator que deseja alterar.
- O Gerenciador de requisitos seleciona a opção editar.
- O Sistema exibe a tela para a alteração.
- O Gerenciador de requisitos realiza as alterações necessárias.
- O Sistema exibe uma mensagem confirmando a alteração.

A02. A operação solicitada é a exclusão de um ator

- O Gerenciador de requisitos seleciona o ator que deseja excluir.
- O Gerenciador de requisitos seleciona a opção excluir.
- O Sistema exibe uma mensagem perguntando ao Gerenciador se realmente deseja excluir o ator.
- O Gerenciador de requisitos confirma a exclusão.

- **Exceção:** Não aplicado
- **Pós- Condição:** Ator cadastrado no sistema.
- Produtos Gerados: Não aplicado
- Pontos de Extensão: Não aplicado

Caso de Uso: Manter Regra de Negócio

• **Descrição:** Descreve o processo de cadastro, alteração e exclusão de uma Regra de Negócio no sistema.

- Pré- Condição: Não Aplicado
- **Ator:** Gerenciador de requisitos
- Fluxo Principal:

P01. O Gerenciador de requisitos solicita o procedimento para cadastrar uma Regra de Negócio no sistema.

P02. O Sistema exibe a tela de cadastro de Regras de Negócio.

P03. O Gerenciador de requisitos preenche os campos.

P04. O Sistema confirma o cadastro.

- Fluxo Alternativo:

A01. A operação solicitada é a alteração de uma Regra de Negócio

- O Gerenciador de requisitos seleciona a Regra de Negócio que deseja alterar.
- O Gerenciador de requisitos seleciona a opção editar.

- O Sistema exibe a tela para a alteração.
- O Gerenciador de requisitos realiza as alterações necessárias.
- O Sistema exibe uma mensagem confirmando a alteração.

A02. A operação solicitada é a exclusão de uma Regra de Negócio

- O Gerenciador de requisitos seleciona a Regra de Negócio que deseja excluir.
- O Gerenciador de requisitos seleciona a opção excluir.
- O Sistema exibe uma mensagem perguntando ao Gerenciador se realmente deseja excluir a regra de negócio.
- O Gerenciador de requisitos confirma a exclusão.
- **Exceção:** Não aplicado
- **Pós- Condição:** Regra de negócio cadastrada no sistema.
- Produtos Gerados: Não aplicado
- Pontos de Extensão: Não aplicado

Caso de Uso Manter Caso de uso

• **Descrição:** Descreve o processo de cadastro, alteração e exclusão de um Caso de uso no sistema.

- **Pré- Condição:** Termo de glossário cadastrado no sistema.

Requisito cadastrado no sistema.

Ator cadastrado no sistema.

Regra de Negócio cadastrada no sistema.

- **Ator:** Gerenciador de requisitos
- Fluxo Principal:

P01. O Gerenciador de requisitos solicita o procedimento para cadastrar um Caso de uso no sistema.

P02. O Sistema exibe a tela de cadastro de Casos de uso.

P03. O Gerenciador de requisitos preenche os campos.

P04. O Sistema confirma o cadastro.

- Fluxo Alternativo:

A01. A operação solicitada é a alteração de um Caso de uso

- O Gerenciador de requisitos seleciona o Caso de uso que deseja alterar.
- O Gerenciador de requisitos seleciona a opção editar.
- O Sistema exibe a tela para a alteração.
- O Gerenciador de requisitos realiza as alterações necessárias.

- O Sistema exibe uma mensagem confirmando a alteração.

A02. A operação solicitada é a exclusão de um Caso de uso

- O Gerenciador de requisitos seleciona o Caso de uso que deseja excluir.
- O Gerenciador de requisitos seleciona a opção excluir.
- O Sistema exibe uma mensagem perguntando ao Gerenciador se realmente deseja excluir o caso de uso.
- O Gerenciador de requisitos confirma a exclusão.
- **Exceção:** Não aplicado
- **Pós- Condição:** Caso de uso cadastrado no sistema.
- Produtos Gerados: Não aplicado
- Pontos de Extensão: Não aplicado