

ReqSis: ferramenta para a gestão da engenharia de requisitos de software

Renato Souza de Oliveira, Fabrício de Sousa Pinto¹

¹Colegiado de Sistemas de Informação– Faculdade de Tecnologia e Ciências (FTC)
Caixa Postal 45.0020 -510 – Vitória da Conquista – BA – Brasil

reimortal@gmail.com, fabbricio7@yahoo.com.br

Abstract. Amidst the many challenges surrounding the process of software development, the requirements elicitation activity has proven to be one of the most complex tasks of software engineering. This article has focused primarily on the development of a system, with free distribution for the management of requirements engineering. The *ReqSis* possible to maintain version control for each elicited requirement, support bidirectional traceability and supports traceability matrix between requirements. It was developed in PHP (Hypertext Preprocessor). The database used was MySQL was developed taking into account the expected according the program improvement process of Brazilian software (MPS.BR) and that meets the principles of HMI (Human Machine Interface) results.

Resumo. Em meio a tantos desafios que cercam o processo de desenvolvimento de software, a atividade de levantamento de requisitos tem se mostrado como uma das tarefas mais complexas da engenharia de software. Este artigo teve como foco principal o desenvolvimento de um sistema, com distribuição gratuita, para a gestão da engenharia de requisitos. O *ReqSis* possibilita manter um controle de versão para cada requisito elicitado, suporte a rastreabilidade bidirecional e suporte a matriz de rastreabilidade entre requisitos. Foi desenvolvido em PHP (*Hypertext PreProcessor*). O banco de dados utilizado foi o *MySQL*. Foi desenvolvida levando em consideração os resultados esperados de acordo o programa de melhoria de processo do software brasileiro (MPS.BR) e que atende os princípios de IHM (Interface Homem Máquina). Foi validade num projeto acadêmico e os resultados mostraram-se satisfatórios.

1. Introdução

Diante do grande avanço tecnológico que a sociedade contemporânea vem sofrendo e da busca constante em se produzir mais com cada vez menos recursos, é a engenharia de software (ES) a área da computação que busca contribuir para a melhoria da qualidade do ciclo de vida do software nas fases de concepção do software, desenvolvimento, gerenciamento e manutenção.

Segundo Sommerville (2007) a ideia central da engenharia de software é o desenvolvimento de um produto dentro do orçamento adequado e com a qualidade aceitável. Não existem limitações físicas no potencial de software, há infinitas possibilidades no campo de desenvolvimento. Entretanto, quando se existem infinitas

possibilidades, há também o risco do nível de complexidade crescer exponencialmente, tornando-o quase impossível de ser compreendido.

O guia SWEBOK (2004) subdivide a engenharia de software em várias áreas de conhecimento. Dentre as áreas apontadas pelo guia existe a engenharia de requisitos. A engenharia de requisitos é apontada, pelos especialistas da área, como uma das tarefas mais complexas e desafiadoras para os engenheiros e analistas de sistemas. Isso se deve ao fato de ela estar diretamente relacionada com as definições do que o sistema deve fazer.

Conforme Pressman (2006), a tarefa de entender de forma precisa os requisitos de software é de fundamental importância para o sucesso do produto final. Uma boa codificação não é essencial para se garantir a qualidade do software se o mesmo não foi bem analisado e especificado. Conforme Rocha (2001), a engenharia de requisitos tem como foco principal tratar o processo de definição dos requisitos de software.

Em relação à engenharia de requisitos observou-se, após pesquisas e análise dos softwares disponíveis na web, grande escassez de ferramentas que atendam aos resultados esperados de acordo com o modelo de qualidade de software do MPS.BR. Uma grande parte das poucas ferramentas disponíveis utilizam arquitetura *desktop* e a outra parte não são ferramentas gratuitas.

O presente artigo teve como objetivo geral desenvolver uma ferramenta WEB gratuita que auxilie os profissionais da área de engenharia de software a gerirem os processos da engenharia de requisitos, levando em consideração os resultados esperados de acordo com o modelo MPS.BR no nível G de maturidade. Como objetivos específicos foram: pesquisar sobre a engenharia de requisitos buscando obter subsídios suficientes para o entendimento e desenvolvimento da ferramenta proposta; Pesquisar sobre custo/benefício das últimas tecnologias com plataforma WEB existentes, procurando selecionar as melhores opções de tecnologia que foram utilizadas no desenvolvimento da ferramenta; Realizar a modelagem do sistema possibilitando a criação de um modelo abstrato do mesmo; Planejamento e execução dos testes no sistema.

Este artigo está assim organizado: na seção 2 e 3 referem-se ao referencial teórico, tratando do conteúdo que dará embasamento para o entendimento do projeto. Na seção 4 refere-se à metodologia, onde mostra todos os métodos utilizados para a elaboração deste trabalho; A seção 5 descreve o desenvolvimento do *ReqSis*. A seção 6 trata-se dos trabalhos relacionados e a seção 7 tem as considerações finais.

2. Engenharia de Requisitos

Segundo Garcia (2013), Os requisitos de um sistema estabelecem, de forma clara e não ambígua, o que o sistema deve fazer, bem como seus serviços e suas restrições. A engenharia de requisitos é o processo de descobrir, analisar, documentar e verificar esses serviços e restrições.

Para os mais renomados especialistas da área computacional, a engenharia de requisitos é considerada como uma das áreas mais complexas de toda a engenharia de software. Tal consideração existe, pois os processos da engenharia de requisitos dependem diretamente da perfeita sincronia de percepção entre o engenheiro e o cliente/usuário.

Segundo Pressman (2006), apesar de em um primeiro momento a engenharia de requisitos não parecer tão difícil, por acreditarmos que o cliente já conhece o problema e consequentemente já sabe como resolvê-lo, o que se observa é que a realidade é completamente diferente. Na maioria das situações nem mesmo o cliente sabe o que ele precisa, e quando sabe não consegue expressar de forma clara. É neste momento que a engenharia de requisitos exerce papel fundamental no sucesso do projeto.

Koscianski e Soares (2007) dizem que os requisitos funcionais devem descrever as funcionalidades que o software vai fornecer quando o mesmo estiver pronto. Exemplos de tais funcionalidades são as entradas, saídas e as exceções. Eles informam que não deve existir vínculos entre os requisitos e as tecnologias de implementação, tais como linguagem de programação, bibliotecas e banco de dados.

De acordo SWEBOK (2004), os requisitos não funcionais na maioria das vezes estão diretamente relacionados com as exigências de qualidade, questões como desempenho, confidencialidade e com as restrições em relação à solução.

O documento de requisitos de software, muitas vezes chamada de especificação de requisitos de software. De acordo com Paula Filho (2009), a especificação dos requisitos do software refere-se ao documento oficial da descrição dos requisitos de software. Tal documento pode retratar um produto indivisível, ou um conjunto de componentes de software, que em conjunto formam um produto.

2.3 Processos de Engenharia de Requisitos

O principal objetivo da engenharia de requisitos é criar e manter uma base de dados denominada de requisito de sistemas. De um modo mais genérico, Sommerville (2007) define quatro subprocessos de alto nível de engenharia de requisitos, são eles: estudo da viabilidade (produz o relatório de viabilidade), elicitação e análise (produz o modelo de sistema), especificação (produz requisitos de usuário e de sistema) e validação (produz documento de requisitos). Esses subprocessos se relacionam entre si e produzem documentos em cada estágio do processo. É possível observar essa relação e a documentação gerada em cada estágio na ilustração da Figura 01.

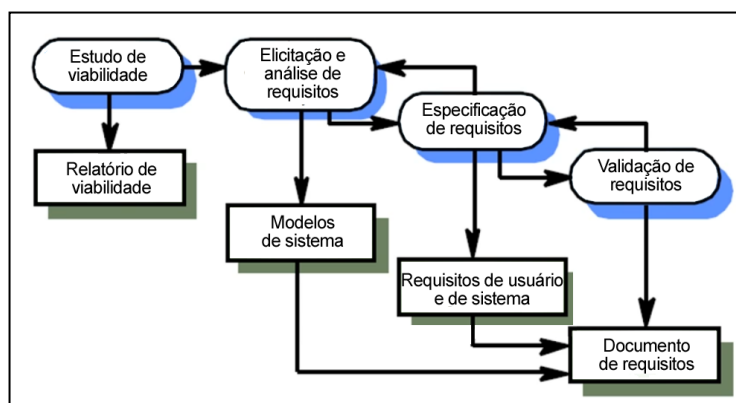


Figura 01: Processo de Engenharia de Requisitos
Fonte: Sommerville (p. 96, 2007)

Dentro do processo de gerenciamento de requisitos, o planejamento é o primeiro estágio. O nível de detalhamento necessário para o gerenciamento de requisitos é estabelecido no estágio de planejamento de cada projeto.

De acordo com Sommerville (2007), durante o estágio de gerenciamento de requisitos, deve-se decidir sobre: Identificação de requisitos: cada requisito deve ser identificado de forma única; Processo de gerenciamento de mudanças: processo de avaliação dos custos e dos impactos das mudanças; Políticas de rastreabilidade: refere-se ao relacionamento entre os requisitos e o projeto do sistema.

Apoio de ferramenta CASE (*Computer-Aided Software Engineering*). Gerenciar requisitos envolve o processamento de grandes números de informações sobre os requisitos. Várias ferramentas podem ser utilizadas, dentre sistemas especializados de gerenciamento até planilhas e sistemas simples.

De acordo com Sommerville (p. 108,2007), “A rastreabilidade é a propriedade de uma especificação de requisitos que reflete a facilidade de encontrar os requisitos relacionados”. Ainda segundo o autor, existem três tipos de informações de rastreabilidade que podem ser mantidos que são: Informações de rastreabilidade da origem (os *stakeholders* possuem ligações com os requisitos que eles propuseram); Informações de rastreabilidade de requisitos (ligam os requisitos que dependem de outros requisitos); Informações de rastreabilidade de projetos (ligam os requisitos aos módulos de projeto).

Na Figura 02 é possível visualizar uma matriz simples de rastreabilidade que demonstra as dependências entre os requisitos. A letra ‘D’ na intersecção linha/coluna ilustra que o requisito na linha depende do requisito identificado na coluna; a letra ‘R’ significa que existe algum outro relacionamento mais fraco entre os requisitos.

ID de requisito	1.1	1.2	1.3	2.1	2.2	2.3	3.1	3.2
1.1		D	R					
1.2			D			R		D
1.3	R			R				
2.1			R		D			D
2.2								D
2.3		R		D				
3.1								R
3.2							R	

Figura 02: Matriz de Rastreabilidade.
Fonte: Sommerville (2007, p. 109).

3 MPS.BR

No que se refere à qualidade dos processos de desenvolvimento do software, um modelo de referência, denominado de MPS.BR, tem sido amplamente usado pelos engenheiros de software brasileiros. O MPS.BR trata-se de um programa que tem como objetivo melhorar a capacidade de desenvolvimento de software nas empresas brasileiras e tem sua definição baseada nas normas técnicas w2x/IEC 12207:2008 e ISO/IEC 15504-2. O MPS.BR define níveis de maturidade que possibilita prever o desempenho futuro das

organizações, no âmbito de evolução dos processos. Os níveis de maturidade são: A (Em Otimização), B (Gerenciado Quantitativamente), C (Definido), D (Largamente Definido), E (Parcialmente Definido), F (Gerenciado) e G (Parcialmente Gerenciado). (SOFTEX, 2009).

4 Metodologia

Para o desenvolvimento da ferramenta proposta neste trabalho, foi inicialmente realizada uma revisão teórica sobre os aspectos envolvendo a engenharia de software e a engenharia de requisitos. Em paralelo, após a seleção de algumas ferramentas disponíveis no mercado, foi realizado um estudo sobre as mesmas, buscando analisar e melhorar as soluções já propostas em outras ferramentas.

Este trabalho é de caráter tecnológico, pois tem como objetivo desenvolver um produto/ferramenta de software.

Após a fase de estudo e definição dos requisitos do sistema, foi dado início a modelagem do mesmo. Para a modelagem do projeto foi utilizada a ferramenta JUDE. A linguagem de modelagem utilizada foi a UML (*Unified Modeling Language*).

Para a implementação (codificação) da ferramenta, foi utilizada a linguagem de programação PHP (*Hypertext PreProcessor*) e a linguagem de marcação HTML (*HyperText Markup Language*) e para a estilização do sistema foi usado o CSS (*Cascading Style Sheets*). A ferramenta utilizada para codificação foi: *Notepad++*.

Em relação ao banco de dados, foi utilizado o *Mysql 5.5* por se tratar de um sistema de gerenciamento de banco de dados (SGBD) livre e compatível com linguagem de programação PHP.

Durante o processo desenvolvimento foram realizados testes de unidade buscando validar os dados de entrada e saída em cada parte individual do software. Após o processo de desenvolvimento, foram realizados teste de sistema buscando promover uma investigação quanto à qualidade do software.

5 Desenvolvimento da Ferramenta *ReqSis*

Esta sessão tem como objetivo apresentar o processo de desenvolvimento da ferramenta *ReqSis*, bem como demonstrar os resultados obtidos, desde a modelagem da ferramenta até aos resultados finais.

Após a fase de implementação e teste, o código fonte do *ReqSis* foi hospedado em um repositório de código fonte e está disponível para *download* através do link: <http://sourceforge.net/p/reqsis>.

5.1 Apresentação do *Reqsis*

Esta seção tem como objetivo apresentar a ferramenta *ReqSis*. Serão apresentadas as principais telas do sistema que foram geradas a partir de um estudo de caso real. O sistema está acessível no link: www.reqproject.com.br e poderá ser acessado usando as seguintes credenciais: login: master e senha: master.

Dentre as principais características da ferramenta *ReqSis* destacam-se o fato da mesma dar suporte a: cadastro de requisitos, controle de versão dos requisitos, suporte a cadastro de caso de uso textual e cadastro de projetos e suporte a rastreabilidade bidirecional, matriz de rastreabilidade de requisito funcional x requisito funcional e requisito funcional x requisito não funcional, rastreabilidade bidirecional, emissão de documento de requisito para equipe técnica (verificação) e emissão de documento de requisito para cliente (validação), conforme ilustrado na Figura 03.

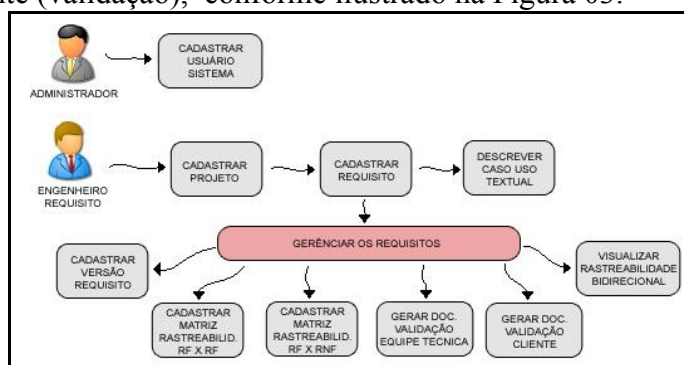


Figura 03: Fluxo de trabalho do ReqSis.

No processo de cadastro de um novo requisito, além do nome do requisito e da sua descrição, as informações relacionadas ao tipo, prioridade e status do requisito são solicitadas. Ao definir um requisito com o status de “proposto”, o mesmo aparecerá com um indicador em amarelo. Para o status de “aprovado” o indicador aparece na cor verde. E para o status de “rejeitado” o indicador aparece em vermelho. No momento que se cadastra um novo requisito, o mesmo recebe automaticamente uma numeração que determina a sua versão. A numeração inicia na versão 1.0. Ao editar um requisito o sistema pergunta se o usuário quer gerar uma nova versão para aquele requisito. Caso opte por gerar uma nova versão, a nova versão ficará registrada na aba de Versão de Requisito, conforme pode ser visto na Figura 04.

ID	NOME	VERSÃO ATUAL	TIPO	PRIORIDADE	QTD VERSÕES	ACÃO
1	Cadastro de Produto	1.3	RF	Baixa	3	🔍 ✕
3	Cadastro de Cliente	1.2	RF	Baixa	2	🔍 ✕
4	segurança	1.1	RNF	Media	1	🔍 ✕

Mostrando 3 do total de 3 registros

Figura 04: Controle de versões do ReqSis.

Conforme é ilustrado na Figura 05, ao clicar no botão [Rastreabilidade RF x RF] o sistema mostra, em uma janela modal, uma matriz contendo todos os requisitos funcionais cadastrados no sistema. Ao clicar no botão [Rastreabilidade RNF x RNF] o

o sistema também mostra uma matriz, entretanto a matriz é firmada entre os requisitos funcionais e os requisitos não funcionais (os requisitos funcionais são expostos na primeira coluna e os não funcionais na primeira linha).

	Cad Produto	Cad NotaFiscal	Cad Usuário	Relat Estoque
Cad Produto	-	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Cad NotaFiscal	<input checked="" type="checkbox"/>	-	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Cad Usuário	<input type="checkbox"/>	<input checked="" type="checkbox"/>	-	<input type="checkbox"/>
Relat Estoque	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	-

Showing 1 to 4 of 4 entries

Buttons: Editar, Fechar

Footer: (RF3) Cadastro de Nota Fiscal, Requisito Funcional

Figura 05: Matriz de Rastreabilidade RF x RF

É possível observar a dependência entre os requisitos nos pontos de intersecção dos eixos da matriz. No caso da matriz entre os requisitos funcionais e não funcionais, a lógica é a mesma. A dependência entre os requisitos funcionais e não funcionais acontecem na intersecção dos eixos da matriz. Outra função presente no módulo de engenharia de requisito é a rastreabilidade bidirecional entre os requisitos e os artefatos produzidos a partir de um determinado requisito. Tal recurso está acessível na aba Rastreabilidade Bidirecional, conforme pode ser visto na Figura 06.

A ferramenta *ReqSis* utilizou uma abordagem de árvore hierárquica para expor as dependências entre os requisitos e os artefatos. Na raiz existem os requisitos. A partir de um requisito surgem os artefatos que, ao serem criados, podem ser vinculados uns aos outros no decorrer do projeto. A rastreabilidade bidirecional pode ser acessada a qualquer momento e expandida automaticamente da raiz até a última folha da árvore. Isso possibilita a rastreabilidade bidirecional (da raiz para as folhas e das folhas para a raiz).

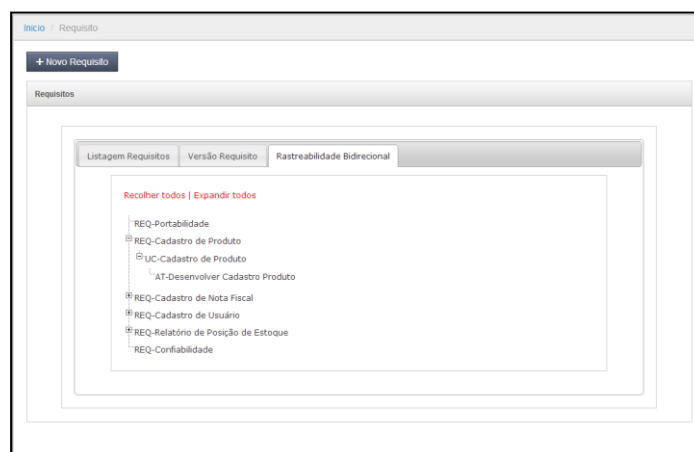


Figura 06: Rastreabilidade Bidirecional

A ferramenta *ReqSis* também oferece suporte ao cadastro do diagrama de caso de uso textual. Ao cadastrar um novo caso de uso textual, o usuário deverá informar o nome do caso de uso, a pré-condição e a pós-condição do caso de uso, os atores e o fluxo principal, alternativo e excepcional. Existe também o botão denominado de “Criar Vínculo”. É este botão que possibilita a vinculação do artefato caso de uso com

qualquer outro artefato já cadastrado. Geralmente um caso de uso está vinculado a um requisito, entretanto o sistema é muito flexível e possibilita o usuário a criar qualquer vínculo que o mesmo achar necessário.

Além da descrição do caso de uso textual, a ferramenta também possibilita a criação de dois tipos de documentos (documento de requisitos para equipe técnica e documento de requisitos para cliente) que podem ser utilizados para validação dos requisitos junto à equipe técnica ou aos clientes que solicitaram os requisitos do sistema.

5.2 Validação da Ferramenta

Com relação à validação da ferramenta foi utilizada na disciplina Engenharia de Software na Faculdade de Tecnologia e Ciências (FTC), campus de Vitória da Conquista-BA, num projeto prático, no mês de fevereiro de 2014. A turma foi composta de doze alunos divididos em três equipes, que fizeram a engenharia de requisitos do sistema proposto pelo professor da disciplina, utilizando essa ferramenta. Após uso, foi submetido um questionário para avaliação do mesmo e chegou ao seguinte resultado: 82% classificaram a ferramenta como ótima, 10% como boa e 8% como regular. Com relação aos princípios de IHM (Interface Homem-Máquina) foi avaliada com boa usabilidade pelos alunos que cursam à disciplina IHM da mesma faculdade.

6 Trabalhos Relacionados

Foi realizada uma análise entre as principais ferramentas já desenvolvidas e que são semelhantes à ferramenta *ReqSis*. Durante a pesquisa foram analisadas várias ferramentas e escolhidas três das quais conseguiram atender o maior número de resultados esperados de acordo o modelo MPS.BR. As ferramentas selecionadas foram: OSRMT (SMITH, 2013), CONTROLA (FRAGA FILHO, 2013) e OpenReq (GRINGS, 2013).

A metodologia de avaliação das ferramentas foi baseada no processo de gerência de requisitos do modelo MPS.BR. Os critérios de avaliação das ferramentas foram realizados com base nos resultados esperados pelo modelo MPS.BR no que se refere a gerência de requisitos no nível G de maturidade. São eles: GRE 1 - Os requisitos são entendidos, avaliados e aceitos junto aos fornecedores de requisitos, utilizando critérios objetivos; GRE 2 - O comprometimento da equipe técnica com os requisitos aprovados é obtido; GRE 3 - A rastreabilidade bidirecional entre os requisitos e os produtos de trabalho é estabelecida e mantida; GRE 4 - Revisões em planos e produtos de trabalho do projeto são realizadas visando identificar e corrigir inconsistências em relação aos requisitos; GRE 5 - Mudanças nos requisitos são gerenciadas ao longo do projeto.

A análise teve início com um estudo aprofundado das ferramentas avaliando os resultados esperados de acordo o MPS.BR, conforme citado anteriormente. Posteriormente foi feito um mapeamento na qual deu origem ao Quadro 01.

Quadro 01: Comparativo entre as Ferramentas Analisadas

Resultados Esperados	OSRMT	Controla	OpenReq	ReqSis
GRE 1 - Os requisitos são entendidos, avaliados e aceitos junto aos fornecedores de requisitos, utilizando critérios objetivos	x		X	x
GRE 2 - O comprometimento da equipe técnica com os requisitos aprovados é obtido				x
GRE 3 - A rastreabilidade bidirecional entre os requisitos e os produtos de trabalho é estabelecida e mantida	x			x
GRE 4 - Revisões em planos e produtos de trabalho do projeto são realizadas visando identificar e corrigir inconsistências em relação aos requisitos	x	x	X	x
GRE 5 - Mudanças nos requisitos são gerenciadas ao longo do projeto	x	x	X	x

O primeiro quesito analisado foi o GRE 1. As ferramentas *OSRMT*, *OpenReq* e *ReqSis* possuem funcionalidades de identificação do fornecedor de requisito na tela de cadastro de usuários. Tal funcionalidade associada à possibilidade de impressão de documento de requisito, que possibilita a certificação de que os requisitos foram entendidos, avaliados e aceitos junto aos fornecedores de requisitos, consegue contemplar os resultados esperados no primeiro critério de avaliação, o GRE 1. No quesito GRE 2, a funcionalidade de impressão de relatório contendo todas as informações dos requisitos e definição dos status de cada requisito, possibilita a certificação do comprometimento da equipe técnica com os requisitos aprovados. Tais funcionalidades foram encontradas apenas na ferramenta *ReqSis*. Quanto ao quesito GRE 3, referente à possibilidade de rastrear os requisitos bidirecionalmente, as ferramentas *OSRMT* e *ReqSis* possuem tal funcionalidade. No quesito GRE 4 todas as ferramentas dão possibilidade de edição para a correção de inconsistência na base de dados dos requisitos e dos artefatos. O quesito GRE 5, que trata do gerenciamento de mudanças nos requisitos, todas as ferramentas ofereceram suporte.

7 Considerações finais

Ao final deste artigo e após a análise comparativa feita entre a ferramenta *ReqSis* e outras ferramentas *freeware* disponíveis na WEB, conforme visto na seção 2 deste trabalho, chegou-se a conclusão de que a ferramenta *ReqSis* contempla todos os resultados esperados pelo programa de melhoria de processo do software brasileiro (MPS.BR). Funcionalidades como: controle de versão dos requisitos, matriz de rastreabilidade, rastreabilidade bidirecional, documento para validação junto à equipe técnica/cliente e cadastro de fluxos de caso de uso, associadas à flexibilidade da plataforma WEB na qual a *ReqSis* foi desenvolvida, fazem deste sistema uma potente ferramenta de auxílio aos profissionais da área de engenharia de software e requisitos.

A abordagem que a *Reqsis* deu para a rastreabilidade bidirecional, organizando as dependências em árvore hierárquica, despontou como um grande diferencial em relação as demais ferramentas freeware disponíveis na internet.

Como proposta de trabalhos futuros sugere-se a produção de novas funcionalidades que possam ser integradas junto a ferramenta *ReqSis*. E o aperfeiçoamento das funcionalidades já desenvolvidas. Dentre os possíveis aperfeiçoamentos estão: mecanismo de geração automática da matriz de rastreabilidade, mecanismos de buscas no módulo de rastreabilidade bidirecional e aperfeiçoamento da validação dos requisitos junto à equipe técnica e aos clientes.

Referencias

- GARCIA, Luís Fernando Fortes. **ENGENHARIA DE SOFTWARE I**. Canoas: Ulbra, 2013.
- KOSCIANSKI, André; SOARES, Michel dos Santos. **Qualidade de Software**: Aprenda as metodologias e técnicas mais modernas para o desenvolvimento de software. 2. ed. São Paulo: Novatec, 2007.
- PAULA FILHO, Wilson de Pádua. **Engenharia de Software**: Fundamentos, métodos e padrões. 3. ed. Rio de Janeiro: Ltc, 2009
- PRESSMAN, Roger S. Engenharia de software. São Paulo: Makron Books, 2006.
- SMITH, Aron. **Open Source Requirements Management Tool**. Disponível em: <<http://sourceforge.net/projects/osrmt/>>. Acesso em: 20 nov. 2013.
- SOFTEX, Sociedade para Promoção da Excelência do Software Brasileiro, MPS.BR – Melhoria de Processo de Software Brasileiro. Guia Geral, maio 2009
- SOMMERVILLE, I. Engenharia de Software. 8ª ed. São Paulo: Addison Wesley, 2007.
- SWEBOK 2004, Guide to the Software Engineering Body of Knowledge, 2004. Disponível em http://www.inf.ed.ac.uk/teaching/courses/seoc/2006_2007/resources/SWEBOK_Guide_2004.pdf. Acesso em 21/08/2007.