



UNIVERSIDADE ESTADUAL DO NORTE DO PARANÁ

CAMPUS LUIZ MENEGHEL

ROBERTO KOMESSU DE OLIVEIRA JUNIOR

**RASTREABILIDADE DE REQUISITOS: UM ESTUDO
PRÁTICO**

Bandeirantes
Novembro/2013

ROBERTO KOMESSU DE OLIVEIRA JUNIOR

**RASTREABILIDADE DE REQUISITOS: UM ESTUDO
PRÁTICO**

Trabalho de Conclusão de Curso apresentado ao Curso de Sistemas de Informação da Universidade Estadual do Norte do Paraná - *campus* Luiz Meneghel, como requisito para obtenção do grau de Bacharel em Sistemas de Informação, orientado pelo Prof^o Me. José Reinaldo Merlin.

Bandeirantes
Novembro/2013

ROBERTO KOMESSU DE OLIVEIRA JUNIOR

**RASTREABILIDADE DE REQUISITOS: UM ESTUDO
PRÁTICO**

Trabalho de Conclusão de Curso apresentado ao Curso de Sistemas de Informação da Universidade Estadual do Norte do Paraná - *campus* Luiz Meneghel, como requisito para obtenção do grau de Bacharel em Sistemas de Informação, orientado pelo Prof^o Me. José Reinaldo Merlin.

COMISSÃO EXAMINADORA

Prof. Me José Reinaldo Merlin.
Campus Luiz Meneghel

Prof.^a Me. Daniela de F. G. Trindade.
Campus Luiz Meneghel

Prof. Wellington Della Mura.
Campus Luiz Meneghel

Bandeirantes, 18 de Novembro de 2013.

"Continue com fome, continue bobo."

Steve Jobs

AGRADECIMENTOS

Agradeço primeiramente a Deus.

Agradeço aos meus pais Elisabete Komessu e Roberto Batista, sem eles não teria motivação pra continuar e nem ter chegado até aqui, por toda confiança investida em todos esses anos aqui em Bandeirantes. Espero retribuí-los com muito orgulho! Eu amo vocês!

Aos professores do curso de Sistemas de Informação, por todo conhecimento passado durante os quatro anos, em especial, agradeço ao professor José Reinaldo Merlin, meu orientador que não hesitou em me ajudar neste projeto e me incentivou e apoiou em todos os outros projetos e estágios durante a graduação.

A todos os amigos e companheiros de Empresa, juntos conseguimos fundar a primeira Empresa Júnior da UENP e eu não seria metade do profissional que sou hoje, obrigado por compartilhar essa experiência e essa amizade.

Aos meus amigos da XVIII Turma, por esses anos de amizade e quero sim que continuemos amigos. Também, aos meus amigos de república, por todas as festas, bagunças e a amizade construída nestes anos.

E de maneira geral por todos que durante a minha graduação fizeram parte da minha vida e me fizeram cada dia melhor, irei me lembrar de todos por toda a vida. Muito obrigado!

RESUMO

A gerência de requisitos, com as técnicas de Rastreabilidade de Requisitos é cada vez mais associada à qualidade de desenvolvimento de softwares e seus aspectos cada vez mais ligados à qualidade final do produto. Vê-se uma necessidade de estabelecer e manter uma validação alinhada com o cliente e se ter claramente definida as suas necessidades para a produção do software e com o passar do tempo controlar as mudanças que ocorrem nos requisitos devido a diversos fatores como erros, inconsistências, mudanças de ideias, etc., evitando que as empresas se prejudiquem para o controle e gerenciamento do mesmo. Por meio de formas visuais (Matrizes de Rastreabilidade) e modelos definidos, apoiados por uma ferramenta específica para a rastreabilidade, este trabalho tem como objetivo a gerência de requisitos, a fim de garantir a qualidade de implementação e manutenção de um sistema. Como objeto de estudo, um software produzido para a gestão de recursos e materiais do Centro de Ciência Tecnológicas da Universidade Estadual do Norte do Paraná – Campus Luiz Meneghel. A contribuição deste trabalho é a discussão da técnica de apoio à gerência e compreensão da necessidade de um modelo e também uma ferramenta que possibilite fazer de forma correta e mais precisa possível a Rastreabilidade de Requisitos.

Palavras – chave: Gerência de Requisitos; Rastreabilidade de Requisitos; Desenvolvimento de Software, Sistema de Reserva de Recursos.

ABSTRACT

The management of requirements with technical requirements traceability is increasingly associated with quality software development and aspects increasingly linked to the quality of the final product. Sees a need to establish and maintain a linear validation with the customer and have clearly defined needs for the production of software, and over time manage changes in requirements that occur due to various factors such errors, inconsistencies, change of ideas, etc., preventing companies from harming themselves for the control and management of the same. Through visual forms (traceability matrices) and defined models, supported by a specific tool for traceability, this work aims to requirements management, to ensure the quality of implementation and maintenance of a system. And as object of study, software produced for the management of resources and materials Science Technology Center of the Universidade Estadual do Norte do Paraná - Campus Luiz Meneghel. The contribution of this work is the discussion of technical management support and understanding of the need for a model and also a tool that allows to accurately and precisely as possible the traceability of requirements.

Keywords: Requirements Management, Requirements Traceability, Software Development, System Resource Reservation.

LISTA DE FIGURAS

Figura 1 - Pesquisa sobre projetos do Standish Group.....	15
Figura 2 - Elementos da Engenharia de Requisitos	20
Figura 3 - Requisição de mudança em requisito, proposto por Karl Wiegers.....	25
Figura 4 - Elementos de pré e pós rastreabilidade.....	28
Figura 5 - Tela principal do software Controla.....	33
Figura 6 - Tela Informações do Projeto do Software Controla.....	33
Figura 7 - Metamodelo de rastreabilidade.....	36
Figura 8 – Diagrama para o processo associado à rastreabilidade.....	37
Figura 9 - Matriz de rastreabilidade: Casos de uso X Requisitos Funcionais.....	38
Figura 10 - Matriz de rastreabilidade: Implementações X Casos de uso.	39
Figura 11 – Alteração da Matriz: Requisitos X Casos de uso	40
Figura 12 – Elementos alterados da Matriz: Requisitos X Casos de uso.	40
Figura 13 – Matriz: Implementações X Casos de uso	41
Figura 14 - Tela de consulta (admin).....	47
Figura 15 - Tela de consulta (usuário).....	48
Figura 16 - Tela de Administração	49
Figura 17 - Tela de Administração de Horários.....	49
Figura 18 – Matriz: Implementações X Casos de uso	51

LISTA DE SIGLAS E ABREVIATURAS

AJAX	<i>Asynchronous Javascript and XML</i>
C	Linguagem de programação C
CSS	<i>Cascading Style Sheets</i>
ECMA	<i>European Computer Manufacturers Association</i>
FDV	Faculdade de Viçosa
HTML	<i>HyperText Markup Language</i>
INCONSE	<i>International Council on Systems Engineering</i>
PHP	<i>Hypertext Preprocessor</i>
RF	Requisito Funcional
RFN	Requisito Não funcional
SGBD	Sistema Gerenciador de Banco de Dados
SQL	<i>Structured Query Language</i>
UENP	Universidade Estadual do Norte do Paraná
UC	Use Case

SUMÁRIO

1	INTRODUÇÃO	12
1.1	CONSIDERAÇÕES INICIAIS	12
1.2	FORMULAÇÃO DO PROBLEMA	12
1.3	JUSTIFICATIVA	14
1.4	OBJETIVO GERAL	15
1.4.1	<i>Objetivos Específicos</i>	15
1.5	ORGANIZAÇÃO DO TRABALHO	16
2	METODOLOGIA DE DESENVOLVIMENTO	17
3	FUNDAMENTAÇÃO TEÓRICA	18
3.1	ENGENHARIA DE SOFTWARE	18
3.2	ENGENHARIA DE REQUISITOS	19
3.2.1	<i>Identificação</i>	20
3.2.2	<i>Análise</i>	21
3.2.3	<i>Documentação</i>	21
3.2.4	<i>Validação</i>	22
3.3	GERÊNCIA DE REQUISITOS	23
3.3.1	<i>Gestão de mudanças</i>	24
3.4	RASTREABILIDADE DE REQUISITOS	26
3.4.1	<i>Classificação para a rastreabilidade</i>	27
3.4.2	<i>Problemas e aplicações da rastreabilidade</i>	28
3.4.3	<i>Matriz de rastreabilidade</i>	29
3.4.4	<i>Metamodelos</i>	30
3.4.5	<i>Ferramentas de Apoio a Rastreabilidade</i>	31
4	ESTUDO EMPIRICO CONDUZIDO	34
4.1	OBJETO DE ESTUDO	34
4.2	DESENVOLVIMENTO	35
4.2.1	<i>Escolha de um metamodelo</i>	36
4.2.2	<i>Matriz de Rastreabilidade</i>	38
4.2.3	<i>Simulação de alteração de requisito</i>	39
4.2.4	<i>Requisitos do sistema</i>	41
4.3	IMPLEMENTAÇÃO DO SISTEMA	44

4.3.1	<i>Grupos de Usuários e Tarefas</i>	44
4.3.2	<i>Banco de dados</i>	44
4.3.3	<i>Linguagem de programação</i>	44
4.3.4	<i>FRAMEWORKS</i>	45
4.4	FUNCIONAMENTO DO SISTEMA	46
4.4.1	<i>Tela de Consulta</i>	47
4.4.2	<i>Tela de Administração</i>	48
5	RESULTADOS OBTIDOS	50
5.1	ANÁLISE DAS MATRIZES	50
6	CONSIDERAÇÕES FINAIS	52
	REFERÊNCIAS	53

1 INTRODUÇÃO

1.1 CONSIDERAÇÕES INICIAIS

A utilização da Engenharia de requisitos é um dos caminhos mais seguros para se obter qualidade no desenvolvimento e manutenção de softwares, pois esta prática diminui sensivelmente os erros de levantamento de requisitos, falhas de implementação e ambiguidades do produto final a ser entregue de acordo com a necessidade de cada cliente. Para isso, métricas de padronização deste processo são comumente aplicadas para se obter um Gerenciamento de requisitos (TORANZO, 2002).

De acordo com Beck (2000), durante a realização de um projeto ocorrem eventos que mudam o seu planejamento. Não se pode evitar estes eventos, mas deve-se adaptar o planejamento a estas ocorrências.

O Gerenciamento de requisitos está relacionado ao processo de controlar todo o desenvolvimento apresentando como referência os requisitos identificados, analisados e documentados, visando manter os modelos e o conjunto de requisitos definidos para o software. A técnica de estudo deste trabalho é a Rastreabilidade de Requisitos.

O processo de Rastreabilidade de Requisitos leva a mapear um determinado software em um documento de requisitos, que pode ser gerado com o auxílio de uma ferramenta e fornecer uma visão rápida do projeto, diminuindo o impacto das mudanças quando necessárias e objetivando uma maior qualidade do produto de software.

Será desenvolvido um software respeitando os critérios e as técnicas estudadas, a fim de garantir o uso e a manutenção dos seus requisitos.

1.2 FORMULAÇÃO DO PROBLEMA

Identificar os requisitos de um software demanda esforço dos profissionais, para que seja compreendido o problema a ser solucionado, como também para definir estes requisitos e torná-los entendíveis a todos os envolvidos no processo.

Os erros cometidos nesta etapa podem implicar em alterações não apenas nos requisitos, mas também em muitas outras etapas do processo de desenvolvimento.

A correção de erros originados em requisitos implica em diversos problemas, tais como: o retrabalho, os custos excessivos para correção de erros se caso houvessem sido detectados nas etapas iniciais, ou até mesmo elicitados de forma que pudessem ser inseridos e tratados dentro do contexto, esses problemas possivelmente seriam reduzidos.

Sabe-se que durante o desenvolvimento surgirão imprevistos e de acordo com a complexidade do software e pelo contexto onde o software será utilizado serão mais difíceis de serem tratados. Modificações constantes, equipes distribuídas, problemas de comunicação entre membros da equipe, negociação mal feita com os clientes, cliente sem objetivo, são alguns dos exemplos que levam um requisito a ser modificado.

Segundo Leffingwell (2000), 40% a 60% de todos os problemas encontrados em um projeto de software são causados por falhas ocorridas na fase de levantamento de requisitos.

Desta forma, mesmo que se faça um bom trabalho de levantamento e também de gerência, à medida que aumenta o número de requisitos ou a quantidade que sofrem modificações, o esforço necessário para mantê-los e gerenciá-los tende a crescer.

Para Leite (2001) muitas das razões que levam a produção de softwares que não satisfazem as necessidades dos clientes são derivadas da falta de atenção para com a tarefa de definir e acompanhar as mudanças dos requisitos de software durante a execução de um projeto. Entre os principais problemas destacam-se:

- a) Ambiguidade: dificuldade de descrever os requisitos de uma forma objetiva e única, sem tornar a sua descrição longa ou de difícil compreensão;
- b) Confusão: dificuldade do cliente em saber diferenciar os requisitos funcionais ou não funcionais e descrever os objetivos do sistema; e
- c) Agrupamento de requisitos: dificuldade em separar claramente os requisitos, o que leva vários requisitos que sejam vistos como sendo apenas um.

1.3 JUSTIFICATIVA

A crescente importância da qualidade de desenvolvimento de software e exigência de qualidade no produto, associando os aspectos técnicos, gerenciais e a necessidade de um modelo de Rastreabilidade de Requisitos que garantirá a identificação de problemas e reduzirá em números aceitáveis as falhas durante o desenvolvimento de software. Também a Gestão de mudanças em uma manutenção do sistema já pronto é a motivação principal deste trabalho.

Com isso, vê-se a necessidade de um controle assíduo na especificação e na validação bilateral (cliente e desenvolvedor) do produto, que evite na identificação dos requisitos do software e do escopo do projeto como o todo, que haja falhas que se agravem durante a produção do software. Considerando que geralmente nem todas serão evitadas, a gerência desta mudança (com o uso da rastreabilidade), deve ser feita de forma a facilitar a sua visualização e documentar quais requisitos afetarão entre si, conseqüentemente reduzir o retrabalho, tempo e custos desnecessários por motivos de modificações ou descoberta de requisitos que afetarão todo o software durante o seu desenvolvimento.

Apesar de já ser estudada há décadas, a Rastreabilidade ainda é pouco e mal usada (Mader et al., 2009).

De acordo com uma pesquisa publicada no ano de 2002 pelo Standish Group, apenas 34% dos projetos em software são considerados como concluídos com sucesso, 15% são tidos como fracassados e 51% como não atingindo completamente os objetivos para os quais foram propostos.

Figura 1 - Pesquisa sobre projetos do Standish Group



Fonte: Miriam Sayão e Karin Koogan Breitman (2007).

Hammer aponta em algumas pesquisas em projetos de software, considerados estáveis esperam-se alterações da ordem de 1% ao mês, este que são visivelmente reduzidos com o uso das técnicas de rastreabilidade. E mesmo sabendo da sua importância, nem todas as organizações, envolvidas com o desenvolvimento de sistemas, aplicam a Rastreabilidade de Requisitos de forma adequada. (RAMESH; JARKE, 2001).

1.4 OBJETIVO GERAL

O objetivo deste trabalho é realizar um estudo empírico sobre a gerência de requisitos, mais especificamente, sobre a Rastreabilidade de Requisitos, mostrando a importância do seu bom uso durante o desenvolvimento de software.

1.4.1 Objetivos Específicos

- a) Documentar e controlar os requisitos para promover o seu gerenciamento;
- b) Analisar e selecionar ferramentas que irão apoiar a rastreabilidade e a gerência;
- c) Definir etapas na gerência com o apoio/uso de uma ferramenta;
- d) Aplicar o processo estudado no desenvolvimento de um software;

e) Mostrar a influência da Rastreabilidade de Requisitos na produção de software e os quais os impactos para a manutenção deste.

1.5 ORGANIZAÇÃO DO TRABALHO

O restante do trabalho está organizado da seguinte forma: Na Seção 2 é demonstrada a metodologia aplicada para que se atinjam os objetivos específicos propostos. Na Seção 3 são apresentados os principais conceitos de Engenharia de Software bem como suas principais áreas de conhecimento e a definição da gerência de requisitos e sua importância no desenvolvimento de software, aborda o que é a Engenharia de requisitos, para que serve e onde pode/deve ser aplicada, seguida dos conceitos de Gerência de Requisitos e por fim a definição do foco do trabalho a Rastreabilidade de Requisitos. Logo após na seção 4, é mostrado a definição do objeto de estudo, o software a ser desenvolvido e os métodos que serão utilizados para validar a pesquisa, como ocorrerá o desenvolvimento do trabalho com a aplicação das técnicas estudadas no desenvolvimento de um software, seguido da implementação do Sistema proposto como objeto de estudo na seção e o funcionamento do Sistema proposto e na Seção 5 os resultados obtidos com a pesquisa, considerando o estudo feito e aplicado. E por fim, considerações finais a respeito do trabalho enfatizando os resultados obtidos.

2 METODOLOGIA DE DESENVOLVIMENTO

O presente trabalho está caracterizado como uma pesquisa exploratória, mais especificamente como um estudo de caso, em que serão definidas as etapas metodológicas que comprovem a eficiência da aplicação das técnicas estudadas, sobre um caso específico.

É uma investigação que se assume como particularística, isto é, que se debruça deliberadamente sobre uma situação específica que se supõe ser única ou especial, pelo menos em certos aspectos, procurando descobrir a que há nela de mais essencial e característico e, desse modo, contribuir para a compreensão global de certo fenômeno de interesse (PONTE, 2006).

Para Yin (1994) o objetivo do estudo de caso é explorar, descrever ou explicar determinada situação real, adquirindo conhecimento e experiência que pode ser útil na tomada de decisão.

O procedimento de pesquisa a ser utilizado é do tipo empírico.

Uma pesquisa empírica tem como principal finalidade testar hipóteses que dizem respeito às relações de causa e efeito de uma circunstância. Envolvem: grupos de controle, seleção aleatória e manipulação de variáveis independentes.

Visto isso, serão necessários os seguintes passos metodológicos para o desenvolvimento do trabalho:

- a) Fundamentação teórica – Definição dos temas: Engenharia de Software, Engenharia de Requisitos, Gerência de Requisitos e Rastreabilidade de Requisitos;
- b) Estudar as ferramentas de apoio ao registro e gerência de requisitos e selecionar uma ferramenta para uso a aplicação;
- c) Desenvolver um sistema a partir dos requisitos já levantados e documentados e utilizar a rastreabilidade para documentação de uso e mudança destes; e
- d) Avaliar os principais impactos observados a partir do uso das técnicas propostas, simulando mudanças de requisitos em determinado fase do projeto e avaliando seu impacto no resultado final do projeto.

3 FUNDAMENTAÇÃO TEÓRICA

Neste capítulo serão abordadas as principais características da Engenharia de software no processo de desenvolvimento, a Engenharia de Requisitos e a técnica de Rastreabilidade de Requisitos de um projeto de software.

3.1 ENGENHARIA DE SOFTWARE

A Engenharia de Software surgiu em meados de 1970 numa tentativa de contornar a crise do software e dar um tratamento de engenharia e ao desenvolvimento de sistemas de software.

Segundo SOMMERVILLE (2005) a Engenharia de Software é uma disciplina da engenharia que aborda todos os aspectos da produção de software, desde os estágios iniciais de especificação, até a manutenção e treinamento de uso desse sistema, após sua entrega. A engenharia de software não está vinculada somente aos processos técnicos, mas também ao gerenciamento de projetos de software e o desenvolvimento de ferramentas, métodos e teorias para apoio ao desenvolvimento de software.

Segundo o SWEBOK (Corpo de Conhecimento da Engenharia de Software), as áreas de conhecimento da Engenharia de Software são:

- a) Requisitos de Software;
- b) Projeto de Software;
- c) Construção de Software;
- d) Teste de Software;
- e) Manutenção de software;
- f) Gerência de Configuração de Software;
- g) Gerência de Engenharia de Software;
- h) Processos de Engenharia de Software;
- i) Ferramentas e Métodos de Engenharia de Software; e
- j) Qualidade de Software;

Neste trabalho será abordada como grande área de conhecimento a Engenharia de requisitos, com estudo aplicado a área de Gerência de Requisitos.

3.2 ENGENHARIA DE REQUISITOS

A engenharia de requisitos é um processo de atividades que contribui para a produção de um documento, composto por todo o conjunto de requisitos do sistema e o apoio necessário para a manutenção de um sistema.

Segundo Dorfman e Thayer (1993) um requisito é definido como:

- a) Uma capacidade de software que o usuário necessita de modo a resolver um problema ou alcançar um objetivo;
- b) Uma capacidade de software que deve ser disponibilizada por um sistema ou componente de sistema de modo a atender um padrão, especificação ou outra formalidade imposta.

A importância dos requisitos para o desenvolvimento é uma parte tão crítica do trabalho, se caso não satisfaça, incapacita e prejudica tanto o sistema e gera problemas quanto for necessário corrigir os requisitos (BROOKS, 1995).

Existem duas fontes que formam os requisitos:

- a) Requisitos do utilizador (Cliente): Os requisitos do utilizador podem ser vistos de diferentes níveis hierárquicos da organização na qual o sistema será implantado, são descritos usando apenas linguagem natural, formulários e diagramas muito simples. Neste nível de especificação surgem dificuldades de falta de especificidade do requisito.
- b) Requisitos do sistema: têm um carácter mais técnico, consistindo numa descrição detalhada dos requisitos do utilizador, recorrendo ao uso, para além da linguagem natural, de linguagens estruturadas e notações gráficas. Estes requisitos destinam-se ainda aos utilizadores do sistema e também às equipas de especificação de arquitetura do sistema e de desenvolvimento.

Figura 2 - Elementos da Engenharia de Requisitos



Fonte: Adaptado de Michel dos Santos Soares (2007)

Segundo Soares (2005), o processo de engenharia de requisitos é composto por quatro atividades de alto nível: Identificação, Análise, Documentação e Validação.

3.2.1 Identificação

Na identificação ou também chamado de eliciação, é feita as boas práticas de declaração de visão do sistema, a fim de definir através de troca de informações dos engenheiros de software, analistas e os utilizadores, quais serão os procedimentos necessários para o entendimento geral do projeto. Nesta etapa também é especificada a viabilidade de aceite e continuidade da negociação, problemas como o cliente não saber ao certo o que irá querer são identificados nesta fase.

Para identificação dos requisitos podem ser citados alguns métodos:

a) Entrevistas e Questionários: é a técnica mais simples e utilizada, por ser bastante eficaz numa fase inicial de obtenção de dados e esclarecimento de algumas dúvidas;

b) Workshops de requisitos: a técnica usada através de uma reunião, da qual deve fazer parte um grupo de analistas, desenvolvedores e um representante do possível cliente;

c) Cenários: este tipo de técnica leva as pessoas a imaginarem o comportamento de um sistema através de exemplos práticos descritivos ou simulações; e

d) Prototipagem: pode ser acompanhado de outra técnica, mas esta exige um conhecimento prévio do projeto e não é só feito durante a identificação. O protótipo pode evoluir durante as outras etapas e mostra-se a forma mais fácil para que o cliente valide o projeto.

3.2.2 Análise

Elaboração de um contexto de onde será inserido e como será utilizado o sistema, documentação do escopo do software, priorização dos requisitos.

Entre as principais atividades desta etapa pode-se considerar:

a) Classificação: agrupar o sistema em módulos para facilitar a visão global do funcionamento pretendido;

b) Priorização: determinar a importância para cada requisito em diferentes visões, por isso é necessário um consenso entre analistas e utilizadores; e

c) Confirmação: aceitação e verificação com as partes interessadas a consistência e validade dos requisitos.

3.2.3 Documentação

É nesta fase que se dá a produção do documento de especificação de requisitos, de forma que todos os *stakeholders* (partes interessadas) possam entender o objetivo do sistema.

Importante ressaltar que em todos os tipos de especificação há dois tipos de requisitos a considerar:

a) Requisitos funcionais: descrevem as funcionalidades que se espera do sistema. É aquilo que o cliente quer que o sistema no final forneça e faça, atendendo aos propósitos para qual o sistema será desenvolvido.

b) Requisitos não funcionais: referem-se a aspectos técnicos do sistema, como controles e segurança, nas quais deve conter no sistema. Costumam

ser divididos em requisitos não funcionais de: Utilidade, Confiança, Desempenho, Suporte e Escalabilidade.

Este documento, normalmente inclui uma combinação dos requisitos do utilizador e do sistema e tem diferentes utilidades para diferentes pessoas (KOTONYA; SOMMERVILLE, 1998).

Uma estrutura proposta pelo IEEE é talvez a mais usada como padrão deste documento IEEE/ANSI 830-1993 (THAYER; DORFMAN, 1993).

3.2.4 Validação

Após os requisitos já terem sido documentados, é necessário que sejam cuidadosamente validados, principalmente quando a consistência e a complexidade ainda não foram inteiramente definidas. Uma vez que um requisito importante que não foi validado, pode acarretar em problemas significantes e esta falha influenciará no futuro do sistema.

Tendo que as alterações podem ser causadas pelos mais diversos fatores, desde inovações tecnológicas, a mudanças por parte do cliente, entre outras.

Segundo Soares (2005), na fase de validação dos requisitos, devem ser verificados os seguintes atributos dos requisitos:

- a) Validade: a especificação resulta da análise dos requisitos identificados junto das partes interessadas envolvidas.
- b) Consistência: não devem existir conflitos entre os requisitos identificados.
- c) Compreensibilidade / Ambiguidade: os requisitos devem poder ser compreendidos de forma evidente pelas partes interessadas.
- d) Completude: todas as funcionalidades pretendidas devem fazer parte da especificação do sistema.
- e) Realismo: as restrições do projeto (tecnologia, recursos e datas) o sistema especificado tem de ser viável.
- f) Rastreabilidade: identifica a origem dos requisitos e verifica se estão claramente identificados e conectados.

A validação trata de uma importante e complexa fase da engenharia de requisitos, da qual depende dos retornos transmitidos pelos clientes, por isso

bastante falível, deve ser feita de forma consistente a fim de corrigir e evitar erros que já foram detectados.

3.3 GERÊNCIA DE REQUISITOS

A Gerência de Requisitos é uma área da Engenharia de software que abrange uma grande parte dos problemas encontrados durante o desenvolvimento de um software. É o compromisso que procura equilibrar as necessidades das diferentes partes interessadas (clientes e desenvolvedores).

LENGWELL E WIDRIG (2000) definem o gerenciamento de requisitos como um esforço sistemático para elicitare, organizar, e documentar um processo de engenharia de requisitos a fim de estabelecer acordo entre clientes, usuários e o grupo de desenvolvimento que tange as mudanças dos requisitos de um sistema. Porém, a mudança dos requisitos, durante o desenvolvimento, é admitida como algo natural e, fato que se deve a própria natureza do software.

O Gerenciamento de requisitos pode ser dividido em três áreas específicas:

- a) Identificação;
- b) Gerenciamento de Mudanças; e
- c) Rastreabilidade;

A rastreabilidade é o foco principal deste trabalho e será explorada detalhadamente mais a frente.

Na prática, a gestão de requisitos é utilizada para a criação de novos requisitos ou gerenciar os que sofreram alterações.

Gerência de requisitos é o enfoque sistemático para a elicitare, organização e documentação dos requisitos do sistema e um processo que estabelece e mantém o acordo entre usuários e a equipe de projeto à medida que os requisitos se modificam. (Leffingwell-00)

Segundo Caper Jones os requisitos de software se modificam, em média, 2% ao mês durante as fases de análise e codificação. Durante a fase de manutenção esta taxa aumenta.

3.3.1 Gestão de mudanças

É comum inúmeras mudanças serem feitas durante a fase de projeto e mesmo ao decorrer do seu desenvolvimento, mas é necessário que o impacto dessas mudanças seja avaliado e entendido a fim de que qualquer mudança seja feita de maneira eficiente.

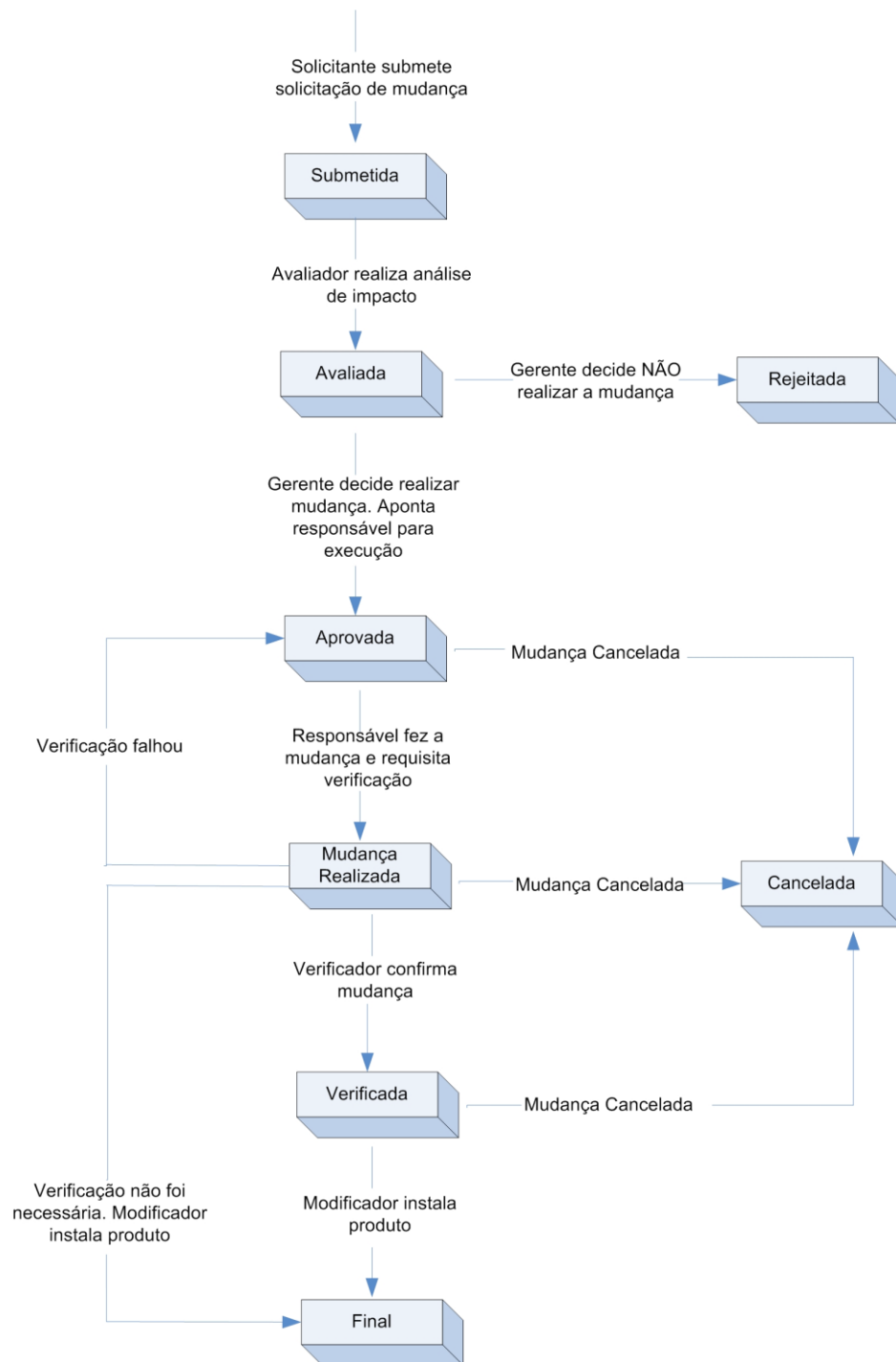
Para as mudanças serem feitas de um modo controlado, é importante que o processo de gestão de mudanças seja definido formalmente, sendo que deverá incluir as seguintes três fases:

- a) Análise do problema e especificação do que será modificado: identificação do problema existente nos requisitos atuais e propor a alteração para melhoria do sistema;
- b) Análise da alteração e medição do seu impacto: através das políticas de rastreabilidade definidas previamente pode se analisar quais impactos terão na alteração; e
- c) Implementação da alteração: alteração no documento de requisitos e conforme seja necessário, no design, na implementação e possivelmente no prazo de entrega do produto e no seu custo final.

Seguir um padrão para efetuar essas mudanças é o ponto chave para que as mudanças sejam facilmente identificadas e tratadas no futuro.

Na figura 3 apresenta-se um exemplo de processo de mudança de requisitos, proposto por Karl Wiegers.

Figura 3 - Requisição de mudança em requisito, proposto por Karl Wiegers.



Fonte: Miriam Sayão e Karin Koogan Breitman (2007).

3.4 RASTREABILIDADE DE REQUISITOS

Para acompanhar o ritmo dos ambientes de negócios, os projetos de software devem lidar com as mudanças e as necessidades do cliente, para isso é necessário ouvir o cliente durante todas as fases do desenvolvimento, para criação do documento, bem como a especificação de requisitos.

A atividade de “mapear”, criar um documento onde seja navegável a visão do escopo do projeto e ter a habilidade de acompanhar a vida de um requisito no processo de construção é definida como Rastreabilidade de Requisitos.

Rastreabilidade também pode ser identificada como a habilidade de localizar a história de toda característica do sistema, dado que os impactos de mudanças nos requisitos podem ser identificados (Hamilton, 1991);

Segundo Lee, (2003), a rastreabilidade é fator chave para proporcionar o rápido acesso à informação do sistema, diminuindo o impacto das mudanças quando necessário.

Em uma visão simples, rastreabilidade pode ser encarada como o conjunto de ligações entre as fontes dos requisitos, os requisitos propriamente ditos e outros artefatos como implementações e casos de teste.

Para Pinheiro (2004), tais ligações são essenciais no desenvolvimento de sistemas, uma vez que é grande o volume de informações produzidas e estas devem estar relacionadas, pois precisamos dela para procurar informações que nos ajudem a tomar decisões e a entender o impacto da mudança de um requisito.

De acordo com Pinheiro (2004) a rastreabilidade é uma tarefa auxiliar que serve tanto para propósitos técnicos quanto gerenciais. Rastrear significa seguir um caminho, como em um mapa, onde se pode ir de coordenada em coordenada, indo e voltando.

A rastreabilidade deve ser produzida de maneira a permitir ao interessado visualizar somente os rastros de seu interesse e atinja todos os interessados de forma que seja fácil a utilização do documento até mesmo para pessoas não envolvidas diretamente ao desenvolvimento do mesmo.

Um requisito é rastreável se é possível descobrir quem sugeriu o requisito, por que o requisito existe (chamado de *Rationale*), que outros requisitos estão

relacionados a ele e como o requisito se relaciona com outras informações tais como layout do sistema, implementação e documentação do usuário (SOMMERVILLE, 1998).

3.4.1 Classificação para a rastreabilidade

A capacidade de rastrear um requisito pode ser definida de duas maneiras principais, como rastrear até seus pequenos detalhes - para frente (*Forwards*), e a de rastrear um refinamento de detalhes até sua origem - rastrear para trás (*Backwards*).

Sobre os tipos de rastreabilidade existem basicamente duas classificações gerais (GOTEL, 1995):

- a) Rastreabilidade horizontal: Rastreabilidade horizontal é a rastreabilidade entre diferentes versões ou variações de requisitos em determinada fase do projeto.
- b) Rastreabilidade vertical é realizada entre requisitos produzidos pelo processo de desenvolvimento ao longo do ciclo de vida do projeto.

Podem ser denominados dois tipos de ligações de classificação dos requisitos: pré-rastreabilidade e de pós-rastreabilidade.

3.4.1.1 Pré-rastreabilidade

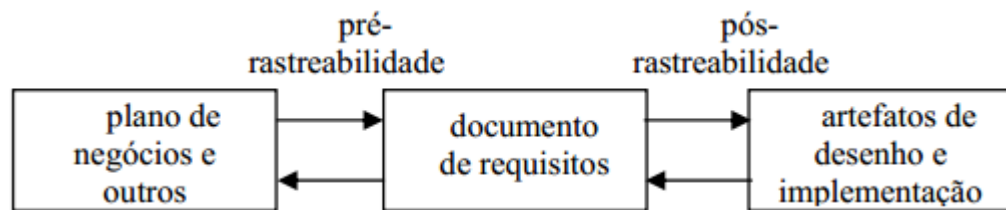
Estão relacionadas ao contexto do qual são retirados os requisitos. As ligações associadas à pré-rastreabilidade permitem identificar a origem de cada requisito e também quais os requisitos originados de uma determinada fonte.

A pré-rastreabilidade de Engenharia de Requisitos armazena informações a respeito de como os artefatos, os atores interessados ou as fontes de informação foram levantados, quais recursos e técnicas foram utilizados, quais atividades foram realizadas. Portanto, a pré-rastreabilidade é fundamental para questões futuras, tais como, auditabilidade do sistema e até para possíveis consultas. Em resumo, pré-rastreabilidade documenta o escopo a partir do qual surgem os requisitos (Davis, 1993).

3.4.1.2 Pós-rastreabilidade

As ligações associadas à pós-rastreabilidade permitem identificar quais componentes implementam um determinado requisito, e também possibilitam saber, quais os requisitos que ele deve atender. As ligações denominadas de pós-rastreabilidade estão relacionadas diretamente ao contexto técnico do processo de desenvolvimento. Em síntese, pós-rastreabilidade vincula os requisitos ao desenho do sistema e sua implementação (Davis, 1993).

Figura 4 - Elementos de pré e pós rastreabilidade



Fonte: Toranzo Castro, Melo (2002).

3.4.2 Problemas e aplicações da rastreabilidade

As principais fontes geradoras de alterações é o próprio conjunto de interessados, que inicialmente não tem muita nitidez nos objetivos e à medida que o desenvolvimento evolui, novas possibilidades têm de serem incluídas no projeto. Um gerenciamento de requisitos adequado necessita da rastreabilidade dos requisitos para controlar as modificações e se bem feito, pode auxiliar gerentes e desenvolvedores em várias situações:

- a) **Requisitos em conflito:** durante o processo de elicitação é comum que diferentes representantes do cliente troquem informações com a equipe de desenvolvimento expondo suas necessidades. Isso irá gerar requisitos, e a análise de requisitos pode gerar requisitos conflitantes.
- b) **Correção de defeitos:** após a identificação da origem do erro, a análise do problema pode indicar que a origem do defeito não está no código. Os elos de rastreabilidade indicarão quais artefatos deverão ser revistos para a identificação do erro;

- c) Validação: a etapa final de validação do sistema criado junto ao conjunto de clientes e usuários se beneficia da rastreabilidade, permitindo mostrar se o sistema entregue atende ao conjunto de requisitos acordados entre clientes e desenvolvedores;
- d) Análise de impacto na evolução do sistema: Facilidade de identificação dos componentes afetados por mudanças em um requisito ou mesmo por inclusão de novos requisitos, sem que haja necessidade de consultas a diferentes artefatos e caso um requisito alterado esteja ligado a outros requisitos estes também podem ser avaliados;
- e) Previsão de custos e prazos: quando uma nova funcionalidade deve ser incluída no sistema ou quando há uma mudança em um requisito já implementado, o gerente de projeto necessita de estimativas confiáveis para poder negociar custos e prazos junto ao cliente. (ANTONIO, 2002)

3.4.3 Matriz de rastreabilidade

De forma a facilitar o rastreamento de requisitos, normalmente são utilizadas tabelas de rastreabilidade, que criam ligações entre os requisitos. A matriz de rastreabilidade mostra as associações entre os requisitos, onde a linha é dependente da coluna e a coluna depende da linha. Trata-se, portanto, de uma forma de visualização gráfica do rastreamento de requisitos. Esta matriz demonstra de que forma um requisito influencia em outro, possibilitando uma análise do impacto de uma alteração do requisito.

	RF _ 01	RF _ 02	RF _ 03	RF _ 04
IMP_ 01	X			
IMP_ 02	X	X	X	
IMP_ 03				
IMP_ 04				X

Tabela 2: Exemplo de Matriz de Rastreabilidade

Pode se observar no exemplo da Tabela 2, que existem as ligações **X**, entre as colunas, esta que mostra o rastreio entre os eixos, por exemplo, se o RF_01 for alterado sabe-se que as IMP_01 e IMP_02, também irão sofrer alterações e conseqüentemente a RF_02 e RF_03, terá de ser revisada, por que também pode sofrer alguma alteração, devido somente a refatoração/evolução do requisito RF_01.

3.4.4 Metamodelos

Esta seção apresenta alguns do metamodelos de rastreabilidade que mostram as formas de definir as propriedades de rastros e elos juntos aos artefatos da Engenharia de software.

3.4.4.1 Metamodelo proposto por Ramesh e Jarke

Ramesh junto com sua equipe de pesquisadores foram um dos primeiros a definir um metamodelo de rastreabilidade, onde concluíram que são definidos três tipo de dimensões: os interessados, as fontes e os objetos:

- a) *Stakeholders* (interessados): são as pessoas envolvidas no Processo de Requisitos, pode-se incluir: Gerentes de projeto, analistas de sistema, design, etc..
- b) *Source* (fontes): documentos que remetem à origem dos requisitos (documentação de requisitos, normas, padrões, legislação pertinente, atas de reuniões) e que também possuem algum grau de interesse na rastreabilidade;
- c) *Objects* (objetos ou artefatos): correspondem a objetos conceituais relacionados ao produto ou a artefatos (Requisitos, design de interface, módulos de sistema, alternativas, etc..), gerados durante o ciclo de desenvolvimento.

Ramesh e Jarke (2001) classificam os elos de rastreabilidade e definem basicamente que são agrupados em duas categorias básicas:

- a) relacionados ao produto: elos que descrevem propriedades e relacionamentos dos objetos; são subdivididos em elos de satisfação e elos de dependência; e

b) relacionados ao processo: elos relacionados ao histórico de ações executadas no próprio processo; são subdivididos em elos de evolução e elos de *Rationale*.

3.4.4.2 Metamodelo proposto por Toranzo

A proposta de Toranzo para a rastreabilidade utiliza quatro estratégias de trabalho: apresenta uma classificação das informações a serem rastreadas, propõe um metamodelo para a rastreabilidade e um modelo intermediário para o rastreamento de requisitos e ainda inclui um processo para guiar o engenheiro de requisitos na aplicação das outras três estratégias.

3.4.5 Ferramentas de Apoio a Rastreabilidade

De acordo com SAYÃO a maioria das abordagens e ferramentas comercialmente disponíveis para o processo de requisitos não fornecem o suporte necessário ao registro dos elos de rastreabilidade, forçando os desenvolvedores a fazer manualmente o trabalho de registrar e manter atualizados todas as modificações de rastreabilidade.

Existe uma grande quantidade de ferramentas comerciais que variam em seu nível de suporte para as atividades do gerenciamento de requisitos. A maioria dessas ferramentas fornece uma visão orientada a texto no gerenciamento de requisitos (DOORS, RTM, Document Director, Requisite Pro e outros) e possuem pouca integração com o processo de desenvolvimento (TORANZO et al., 2002);

De 1998 até 2008 o INCONSE (International Council on Systems Engineering), apresenta uma comparação entre os mais importantes pacotes de software para sistemas de gerenciamento de requisitos.

Para parte do processo, foi escolhida uma ferramenta de uso acadêmico, chamada "Controla" dado ao fato que a ferramenta é de um Freeware (software gratuito), produzida também em um estudo de Rastreabilidade de Requisitos e observando outras ferramentas disponíveis, foi a que mais se adequou ao processo utilizado e ofereceu facilidade de uso.

3.4.5.1 Ferramenta: Controla

Desenvolvida como trabalho final de Curso Bacharelado em Sistemas de Informação da Faculdade de Viçosa (FDV) – MG, o Controla tem como objetivo apoiar as atividades inerentes ao processo de desenvolvimento de software. A definição e implantação de um processo de desenvolvimento de software, sobretudo em pequenas empresas não é tão comum, geralmente o foco está no desenvolvimento para o atendimento imediato da necessidade do cliente, ou seja, há pouco ou quase nenhum planejamento antes da codificação.

O Controla oferece importantes recursos e os mais importantes são relacionados a seguir:

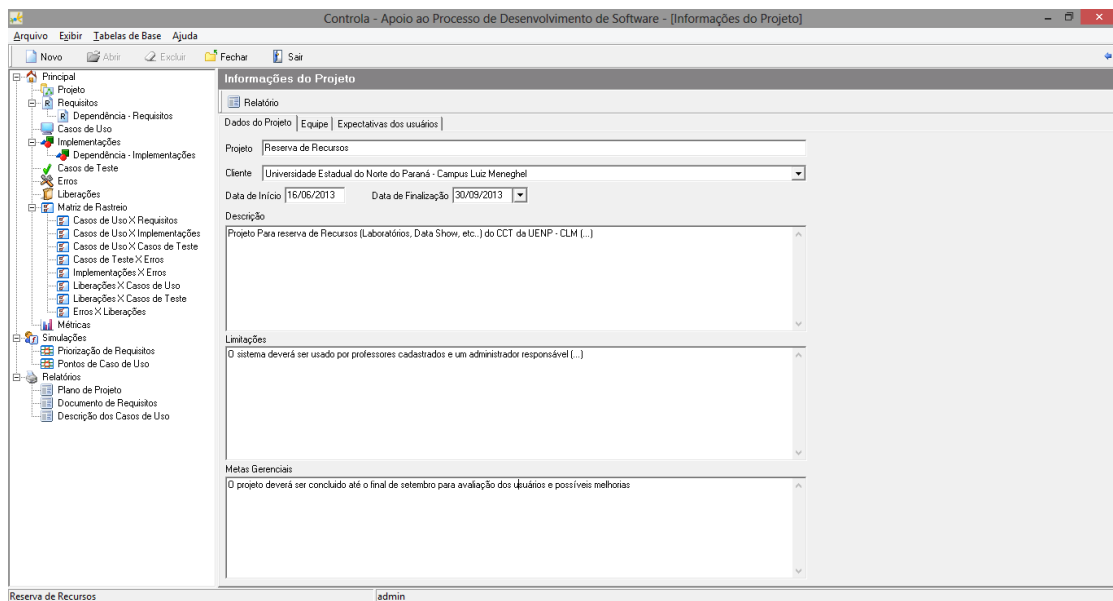
- a) Gerenciamento de Requisitos;
- b) Gerenciamento de Casos de Uso;
- c) Gerenciamento de Casos de Teste e Erros;
- d) Planejamento de Liberações;
- e) Gerenciamento de Implementações;
- f) Controle de Dependência entre implementações;
- g) Matriz de Rastreabilidade: - Rastreabilidade dos requisitos; Casos de Uso X Implementações; Casos de Uso X Casos de Teste; Casos de Teste X Erros; Implementações X Erros; Liberações X Casos de Uso; Liberações X Casos de Teste; Erros X Liberações;
- h) Registro de Métricas para todos os artefatos;
- i) Ferramenta para priorização de Requisitos;
- j) Documento de Plano de Projeto;
- k) Documento de Casos de Uso;
- l) Documento de Especificação de Requisitos;

Figura 5 - Tela principal do software Controla



Fonte: Software Controla

Figura 6 - Tela Informações do Projeto do Software Controla



Fonte: Software Controla

4 ESTUDO EMPIRICO CONDUZIDO

4.1 OBJETO DE ESTUDO

Para que a qualidade deste software seja mantida e sua manutenção seja feita de maneira que o sistema assegure que os requisitos essenciais estipulados do projeto sejam alcançados, é proposto que as técnicas estudadas de Rastreabilidade junto com o documento gerado (matriz de rastreabilidade, documento de requisitos), possibilitem que com sua aplicação obtenha-se a qualidade de desenvolvimento do sistema.

Em ambientes educacionais como universidades possuem-se recursos (notebooks, *datashows*, caixas de som, laboratórios de informática, salas de aula, auditórios, etc...) à disposição dos centros. O gerenciamento desses recursos envolve um controle de seu uso, de modo que não exista coincidência na data e horário de utilização e que todos os usuários possam ter acesso de maneira justa e organizada. A reserva de recursos requer atenção e na medida em que a demanda por reservas aumenta, torna-se mais complexo controlar um grande número de requisições. Além disso, como reservas são constantemente alteradas ou excluídas, gerenciá-las sem o auxílio de uma ferramenta computacional se torna mais complicado.

Visando melhorar o processo de reserva de recursos, neste trabalho é proposto o desenvolvimento do Sistema de Reserva de Recursos para o Centro de Ciências Tecnológicas da Universidade Estadual do Norte do Paraná – Campus Luiz Meneghel.

Esse sistema visa controlar as reservas de modo a facilitar o trabalho tanto dos responsáveis quanto dos usuários dos recursos. Por meio desse sistema o usuário terá o conhecimento de quais recursos estão à disposição para utilizar em suas atividades acadêmicas. Será possível visualizar os recursos disponíveis em determinada data bem como visualizar os já reservados por outros usuários.

4.2 DESENVOLVIMENTO

No desenvolvimento de um software onde a rastreabilidade é utilizada, vimos que as atividades associadas à rastreabilidade são vistas em quatro momentos: na fase inicial do trabalho são definidos os objetos e artefatos a serem rastreados (elicitação de requisitos). Durante todo o processo de desenvolvimento, os objetos e artefatos a serem rastreados deverão ser relacionados. Esse processo tanto pode ser realizado manualmente como contar com auxílio de ferramentas. A recuperação desses relacionamentos deverá ser realizada para atender necessidades dos interessados nas diferentes atividades: resolução de requisitos em conflito, identificação da origem de requisitos, alocação de requisitos a componentes, mudança espontânea de requisito. Além disso, espera-se que os casos de uso sofram evolução, portanto o processo de manutenção dos mesmos também deve ser realizado.

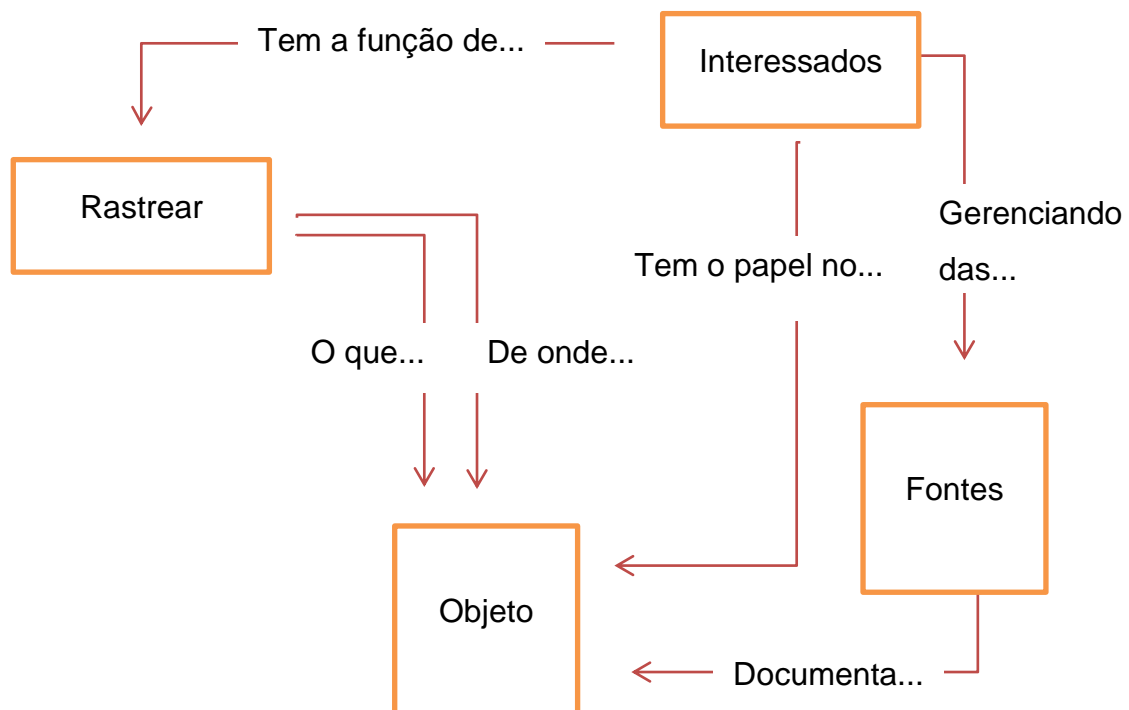
Pinheiro (2004) propõe a seguinte ordem para o desenvolvimento:

- a) Definição: envolve a definição das entidades a serem rastreados (artefatos, objetos), os tipos de relacionamentos a serem estabelecidos e sob que condições os elos deverão ser registrados. Esta definição pode ter como base um modelo de referência como os apresentados anteriormente, e deve considerar as necessidades da equipe e as restrições contextuais como prazos e custos;
- b) Registro dos elos: envolve o registro explícito dos relacionamentos, envolvendo artefatos e entidades, nas situações e condições definidas anteriormente. O uso de uma ferramenta é imprescindível devido ao volume de informações a serem manipuladas;
- c) Recuperação ou extração: dado um artefato rastreável implica em encontrar as entidades associadas, ou sua origem;
- d) Evolução: processo de atualização dos elos de rastreabilidade, conforme surjam modificações nas entidades relacionadas ou nos próprios relacionamentos estabelecidos entre elas, ao longo do processo de desenvolvimento e de evolução do software.

4.2.1 Escolha de um metamodelo

Alguns dos metamodelos estudados levam em consideração o tamanho do projeto e quantidade de informações que serão rastreadas, visto isso, dois metamodelos poderão ser analisados, metamodelos de Toranzo e Ramesh e Jarke,

Figura 7 - Metamodelo de rastreabilidade



Fonte: Imagem adaptada do metamodelo de Ramesh e Jarke, 2001.

Na visão de Ramesh e Jarke (2001), os rastros devem responder as seguintes perguntas:

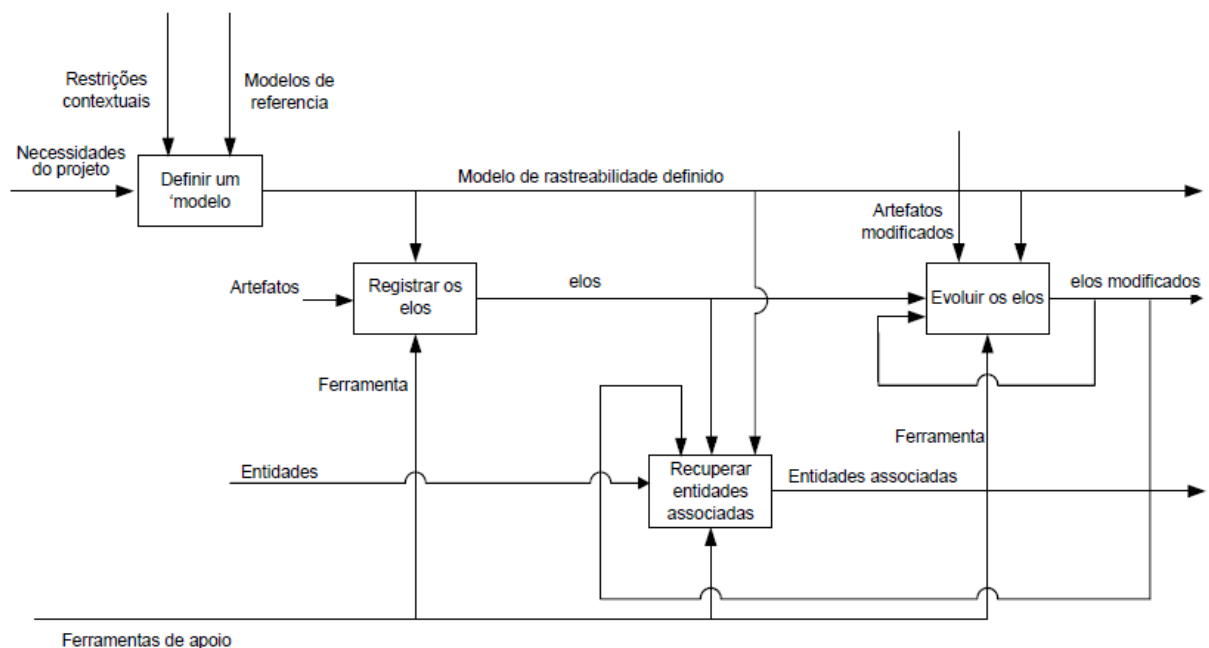
- Que informação é representada?
- Quem são as partes Interessadas que executam papéis na criação e manutenção dos objetos de rastreabilidade?
- Onde é representada?
- Como é representada?
- Quando a informação foi levantada, modificada ou evoluída?

Já Toranzo (2002), define um modelo mais rígido para a rastreabilidade, onde se mostra necessário seguir de alguns passos para que se tenha a rastreabilidade. No seu metamodelo não apenas é necessário definir um relacionamento, mas é proposto definir diferentes tipos de relacionamentos entre os artefatos rastreados. Os relacionamentos que ele utiliza para rastrear informação são: satisfação, recurso, responsabilidade, representação, alocado, agregação e generalização.

Portanto, Toranzo (2002) apresenta um modelo mais completo comparado ao de Ramesh, mas não é aplicável a projetos com poucos requisitos como é o objeto de estudo, por isso pode-se aproveitar de algumas das características principais dos dois estudos que basicamente é visto na figura 8.

Na figura 8 apresenta-se o processo básico de rastreabilidade, onde as atividades nomeadas nos retângulos possuem entradas representadas por setas à esquerda e saídas à direita. Controles são representados por setas descendentes e mecanismos são representados por setas ascendentes.

Figura 8 – Diagrama para o processo associado à rastreabilidade



Fonte: Sayao, 2000

4.2.2 Matriz de Rastreabilidade

Para primeira parte do processo proposto, segundo as técnicas estudadas e o metamodelo de Ramesh e Jarke, este que mais se mostrou indicado a um projeto de software pequeno e com pequeno número de requisitos, foi utilizado o software “Controla” para geração das Matrizes de Rastreabilidade, que pode ser visto nas Figuras 9 e 10.

No software foi necessário o cadastro completo do projeto, desde informações iniciais do projeto (Nome do projeto, Cliente, Limitações do projeto, Meta gerenciais, Datas de início e fim, etc.), até os registros de requisitos funcionais e especificações de cada um deles, também registro dos requisitos não funcionais, definição de casos de uso e especificação detalhada de implementações propostas,

Foi necessário uma análise manual das dependências na matriz, para mostrar quais elementos tem elos entre si, ou seja, é necessário um analista para que selecione os artefatos que apresentam características que interligam-se como elementos na matriz de rastreio. Com isso possibilitou a geração das duas matrizes principais, estes que também possibilitam uma visão detalhada do escopo do projeto e sua rastreabilidade.

Figura 9 - Matriz de rastreabilidade: Casos de uso X Requisitos Funcionais

The screenshot shows a web application window titled 'Controla - Apoio ao Processo de Desenvolvimento de Software - [Casos de Uso - Requisitos]'. The interface includes a menu bar (Arquivo, Exibir, Tabelas de Base, Ajuda) and a toolbar with buttons for 'Novo', 'Abrir', 'Excluir', 'Fechar', and 'Sair'. The main content area is titled 'Rastreabilidade Casos de Uso -> Requisitos' and contains a table with columns for various functional requirements (RF1-12) and rows for use cases (UC1-7). The table cells contain checkboxes indicating the relationship between each use case and requirement. A legend at the bottom indicates that a yellow square represents a modified item. The user 'admin' is logged in.

	RF1- Autenticar Usuário com Login e Senha	RF2- Gerenciar diferentes permissões de usuários	RF3- Reservar recurso	RF4- Reservar para terceiro	RF5- Cancelar sua própria reserva	RF6- Administrador pode cancelar todas reservas	RF7- Cadastrar novo usuário	RF8- Cadastrar nova reserva	RF9- Controlar horários para reserva	RF12- Cadastrar novo recurso
UC1- Manter Usuário	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
UC2- Manter Recurso	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
UC3- Manter Horário	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
UC4- Efetuar reserva	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
UC5- Cancelar Reserva	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
UC6- Reservar para terceiro	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
UC7- Consultar horários	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>

UC7- Consultar horários
RF12- Cadastrar novo recurso

Reserva de Recursos admin

Fonte: Software Controla

Figura 10 - Matriz de rastreabilidade: Implementações X Casos de uso.

	PR5-BD: Consultar Usuários	PR6-BD: Consultar Horário	PR7-BD: Consulta Recursos	PR8-BD: Consulta Reserva	PR11-Fn: Verifica Campos Vazios	PR13-BD: Consultar tipo de usuário	PR14-Fn: Verificar Reserva a ser cancelada	PR15-Interface: Mostrar Calendário	PR16-BD: Inserir Reserva	PR17-BD: Deletar Reserva	PR18-BD: Novo Recurso	PR19-BD: Novo Usuário
UC1-Manter Usuario	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
UC2-Manter Recurso	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
UC3-Manter Horário	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
UC4-Efetuar reserva	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
UC5-Cancelar Reserva	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
UC6-Reserva para terceiro	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
UC7-Consultar horários	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

UC7-Consultar horários
PR19-BD: Novo Usuario

Reserva de Recursos admin

Item modificado

Fonte: Software Controla

A visão das matrizes pode-se exemplificar uma possível alteração de um requisito e percorrer o caminho para trás (*Backward*), até a sua fonte e encontrar todos os requisitos que apresentam elos e também sofrerão alterações.

4.2.3 Simulação de alteração de requisito

O requisito principal do Sistema sofre alteração, RF3 – Reservar recurso, agora serão reservados mais recursos por reserva de usuário, onde antes apenas havia um recurso por reserva, agora necessita-se um ou mais recursos por reserva.

Na matriz de rastreabilidade “Casos de Uso -> Requisitos” é possível facilmente visualizar quais requisitos irão sofrer alterações (coluna em pé selecionada na figura 11) e quais possivelmente também terão de ser modificados a partir dessa alteração (linhas selecionadas na figura 11), mostrando visualmente a influencia de um artefato sobre o outro.

Na figura 12 foram selecionados apenas os artefatos modificados a partir da alteração do requisito RF3 – Reservar recurso.

Figura 11 – Alteração da Matriz: Requisitos X Casos de uso

	RF1- Autenticar Usuário com Login e Senha	RF2- Gerenciar diferentes permissões de usuários	RF3- Reservar recurso	RF4- Reserva para terceiro	RF5- Cancelar sua própria reserva	RF6- Administrador pode cancelar todas reservas	RF7- Cadastrar novo usuário	RF8- Efetuar nova reserva	RF9- Controlar horários para reserva	RF12- Cadastrar novo recurso
UC1- Manter Usuário	☒	☐	☐	☐	☐	☐	☒	☐	☐	☐
UC2- Manter Recurso	☒	☐	☒	☒	☐	☐	☐	☐	☐	☒
UC3- Manter Horário	☒	☐	☐	☐	☐	☐	☐	☐	☒	☐
UC4- Efetuar reserva	☒	☐	☒	☒	☐	☐	☐	☒	☐	☐
UC5- Cancelar Reserva	☒	☐	☐	☐	☒	☒	☐	☐	☐	☐
UC6- Reserva para terceiro	☒	☒	☒	☒	☐	☐	☐	☒	☐	☐
UC7- Consultar horários	☒	☐	☐	☐	☐	☐	☐	☐	☒	☐

Fonte: Software Controla

Figura 12 – Elementos alterados da Matriz: Requisitos X Casos de uso.

	RF3-Reservar recurso	RF4-Reserva para terceiro	RF8-Efetuar nova reserva	RF12-Cadastrar novo recurso
UC2-Manter Recurso	☒	☒	☐	☒
UC4-Efetuar reserva	☒	☒	☒	☐
UC6-Reserva para terceiro	☒	☒	☒	☐

Fonte: Software Controla

Os casos de uso: UC2 – Manter recurso, UC4 - Efetuar Reserva e UC6 – Reserva para terceiro. Terão certamente de serem revisadas por que também sofrerão alterações.

Seguindo as mesmas linhas das colunas, pode ser entender que os Requisitos RF4 – Reserva para terceiro, RF8 – Efetuar nova reserva e RF12 –

Cadastrar novo recurso, possivelmente e com grandes chances também terão de ser alterados, necessitando de uma análise detalhada da matriz.

Se formos mais a fundo, podemos pegar os casos de uso UC2, UC4 e UC6 e verificarmos na matriz de rastreio de (Implementações X Casos de Uso – Figura 13).

Figura 13 – Matriz: Implementações X Casos de uso
após alteração de requisito.

Rastreabilidade Casos de Uso -> Implementações											
Atualizar Relatório											
	PR5-BD: Consultar Usuários	PR6-BD: Consultar Horários	PR7-BD: Consulta Recursos	PR8-BD: Consulta Reserva	PR11-Fn: Verifica Campos Vazios	PR13-BD: Consultar tipo de usuário	PR14-Fn: Verificar Reserva a ser cancelada	PR15-Interface: Mostrar Calendário	PR16-BD: Inserir Reserva	PR17-BD: Deletar Reserva	PR18-BD: Novo Recurso
UC1-Manter Usuario	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
UC2-Manter Recurso	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
▶ UC3-Manter Horário	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
UC4-Efetuar reserva	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
UC5-Cancelar Reserva	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
UC6-Reserva para terceiro	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
UC7-Consultar horários	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

- Item modificado

Reserva de Recursos admin

Fonte: Software Controla

No momento da alteração do Requisito RF3 - Efetuar Reserva, o software “Controla” detectou alterações e na matriz de rastreio Casos de uso -> Implementações, ele identificou e sinalizou (em amarelo) todas as implementações alteradas: PR5 – Consultar usuário, PR6 – Consultar horários, PR7 – Consultar recursos, PR8 – Consultar reserva, PR11 – Verifica Campos Vazios, PR13 – Consultar tipo de usuário, PR16 – Inserir Reserva, PR18 – Novo recurso. Assim, facilita o entendimento completo da alteração do escopo e possibilita o seu rastreio e a diminuição do impacto da mudança.

4.2.4 Requisitos do sistema

Descrição detalhada de todos os requisitos do sistema, levantadas por meio das necessidades dos usuários e um estudo do ambiente em que será aplicado.

4.2.4.1 Requisitos funcionais

- a) O Sistema deve permitir que somente pessoas cadastradas possam consultar as reservas feitas através de uma área de acesso privado (login e senha);
- b) O Sistema deve possibilitar a exibição de maiores informações sobre uma reserva (hora da reserva e quem reservou);
- c) O Sistema deve classificar os usuários em categorias, com permissões diferenciadas (usuários e administrador);
- d) O Sistema deve conter um usuário do tipo Administrador, que será responsável pela administração de usuários, dos recursos e dos horários;
- e) O Sistema deve pedir que os seguintes dados sejam informados para a marcação de reservas eventuais: recurso a ser reservado, dia, mês, ano, horário da reserva;
- f) O Sistema deve permitir que sejam marcadas em uma mesma iteração quantas horários forem necessárias;
- g) O Sistema deve permitir que os usuários administradores faça cadastros para terceiros a partir do nome;
- h) O Sistema deve permitir que os usuários possam cancelar suas próprias reservas;
- i) O Sistema deve permitir que os usuários administradores possam limpar todas as reservas feitas por qualquer usuário;
- j) O Sistema não deve permitir que seja feito mais de uma reserva para o mesmo recurso;
- k) O Sistema deve pedir que no cadastramento de um novo usuário sejam informados: o nome completo, login e e-mail;
- l) O Sistema deve pedir que no cadastramento de um novo recurso sejam informados: o nome do recurso a ser exibido, descrição breve;
- m) O Sistema deve gerar senhas automáticas ao fazer o cadastro de um novo usuário;
- n) O Sistema deve verificar que os novos usuários e os novos recursos não devam apresentar o mesmo nome;
- o) O Sistema não deve permitir que os horários contenham o mesmo intervalo;

4.2.4.2 Requisitos Não Funcionais

- p) O Sistema deve exibir um calendário para visualização rápida, para escolha da data;
- q) O Sistema deve ser acessado com subdomínio da Universidade. (uenp.edu.br);
- r) O Sistema deve ser implementado em plataforma *Web e Linguagem PHP*.
- s) O Sistema deve ter alta portabilidade, devendo ser visualizado por qualquer *browser* de dispositivos móveis;
- t) O Sistema deve ter uma alta usabilidade, uma vez que será utilizado por vários funcionários, com diferentes graus de conhecimentos em informática;
- u) O Sistema deve ter uma alta manutenibilidade, para possíveis mudanças de requisitos;
- v) O sistema deve ser compatível com as versões mais recentes dos navegadores Google Chrome, Firefox e Internet Explorer;
- w) O controle de data e horário serão feitos via servidor;
- x) O Sistema deve apresentar interface com carregamento Assíncrono;

4.2.5 Casos de uso

Aqui será especificado um dos artefatos utilizado para geração da matrizes de rastreabilidade, os casos de uso do sistema Reserva.

- a) Manter Usuário (Criar usuário, Editar usuário, remover usuário);
- b) Manter Recurso (Criar Recurso, Editar recurso, remover recurso);
- c) Manter Horário (Criar Horário, Editar horário, remover horário);
- d) Efetuar reserva;
- e) Cancelar reserva;
- f) Reserva para terceiro;
- g) Consultar horário;
- h) Reserva por período;

4.3 IMPLEMENTAÇÃO DO SISTEMA

Nesta seção é descrito os passos que levaram a produção do software que serviu como objeto de estudo deste trabalho.

4.3.1 Grupos de Usuários e Tarefas

Ao projetar o sistema é necessário definir seus grupos de usuários. Os grupos de usuários existentes e as possíveis tarefas de cada um deles são:

No sistema proposto foi definido dois tipos de usuários:

- a) Administrador: terá controle total do sistema, podendo cadastrar ou remover usuários, cadastrar ou remover recursos e definir permissões de outros usuários. Além disso, poderá efetuar a reserva de um recurso, alterar e também cancelar a reserva de quaisquer usuários.
- b) Usuário (Professores/Técnicos): poderão visualizar reservas, recursos, reservar determinado recurso, bem como cancelar a reserva de um recurso previamente efetuado por ele.

4.3.2 Banco de dados

Neste sistema o SGBD utilizado foi o MySQL, atualmente um dos mais populares sistemas Open Source de gerenciamento de banco de dados. Essa escolha foi feita pela portabilidade desse SGBD que possui integração com qualquer plataforma, é compatível com a maioria das linguagens de programação atuais e por ter bom desempenho e estabilidade.

4.3.3 Linguagem de programação

Para o desenvolvimento do software foi necessário conhecimento de algumas linguagens de programação e ferramentas. A seguir a descrição das linguagens adotadas e as principais ferramentas que foram utilizadas.

4.3.3.1 PHP

Para a implementação das regras de negócio do sistema foi utilizada uma linguagem de programação para desenvolvimento web denominada PHP.

PHP é uma linguagem de script multiuso ideal para desenvolvimento para Web com integração dentro do HTML. Muito da sintaxe PHP é emprestada de linguagens como C e Java com alguns recursos específicos. O objetivo da linguagem é permitir a escrita dinâmica aos desenvolvedores através da geração de páginas dinâmicas (Dall'Oglio, 2007).

A opção por utilizar a linguagem PHP se dá pelo fato de ela ter o poder de programar uma solução simples e eficiente para o desenvolvimento web, com agilidade de desenvolvimento.

4.3.3.2 Javascript

Javascript é uma linguagem de programação baseada em *scripts* e padronizada pela ECMA *International* (associação especializada na padronização de sistemas de informação).

Foi criada por Brendan Eich (Netscape) e surgiu em 1995 como linguagem de *script client-side* de páginas web. É uma linguagem dinâmica, orientada a objetos e criada com sintaxe similar à linguagem C.

Apesar de conter Java no nome, a linguagem Javascript é distinta da linguagem Java e apresenta recursos não disponibilizados em Java ou C++.

Os *scripts* desenvolvidos em Javascript são muito populares e amplamente integrados em páginas web.

Foi escolhido devido ao fato da necessidade de uma linguagem associada à tecnologia de carregamento assíncrono Ajax e necessidade de recursos para implementação Web para auxílio do PHP.

4.3.4 FRAMEWORKS

4.3.4.1 AJAX

AJAX é acrônimo em língua inglesa de "*Asynchronous Javascript and XML*", que em português significa "Javascript e XML Assíncronos". Designa um conjunto de técnicas para programação e desenvolvimento web que utiliza tecnologias como Javascript e XML para carregar informações de forma assíncrona.

Desde que o modelo AJAX começou a ser utilizado, é visto como uma ferramenta revolucionária, que torna mais agradável a navegação e utilização de serviços diversos na internet.

O desenvolvimento em AJAX permite a criação de páginas mais criativas e com maior interação com o usuário. A sua utilização foi importante para a criação do conceito Web 2.0, que tornou a experiência do usuário mais interessante e proveitosa.

O uso do AJAX no desenvolvimento possibilitou que a informação fosse carregada de maneira mais simples e precisa. O usuário não precisa aguardar que a página seja totalmente recarregada quando faz uma requisição, pois o servidor só irá retornar os dados relevantes, reduzindo o tráfego de dados pela rede.

4.3.4.2 Bootstrap

Originalmente criado por um designer e um programador no Twitter, Bootstrap tornou-se um dos quadros de *front-end* mais populares e projetos de código aberto do mundo.

Bootstrap foi criado no Twitter, em meados de 2010. Poucos meses em desenvolvimento, o Twitter realizou sua primeira “Semana Hack” e o projeto explodiu como desenvolvedores de todos os níveis.

Já passou por três atualizações, sua versão mais recente propõe uma ótima responsividade às versões de websites para dispositivos móveis.

Foi escolhido com intuito de propor uma interface amigável e com boa usabilidade, além é claro de utilizar todos os recursos que o deixou popular (responsividade) portabilidade que o Framework oferece para uso em múltiplos navegadores de múltiplas resoluções.

4.4 FUNCIONAMENTO DO SISTEMA

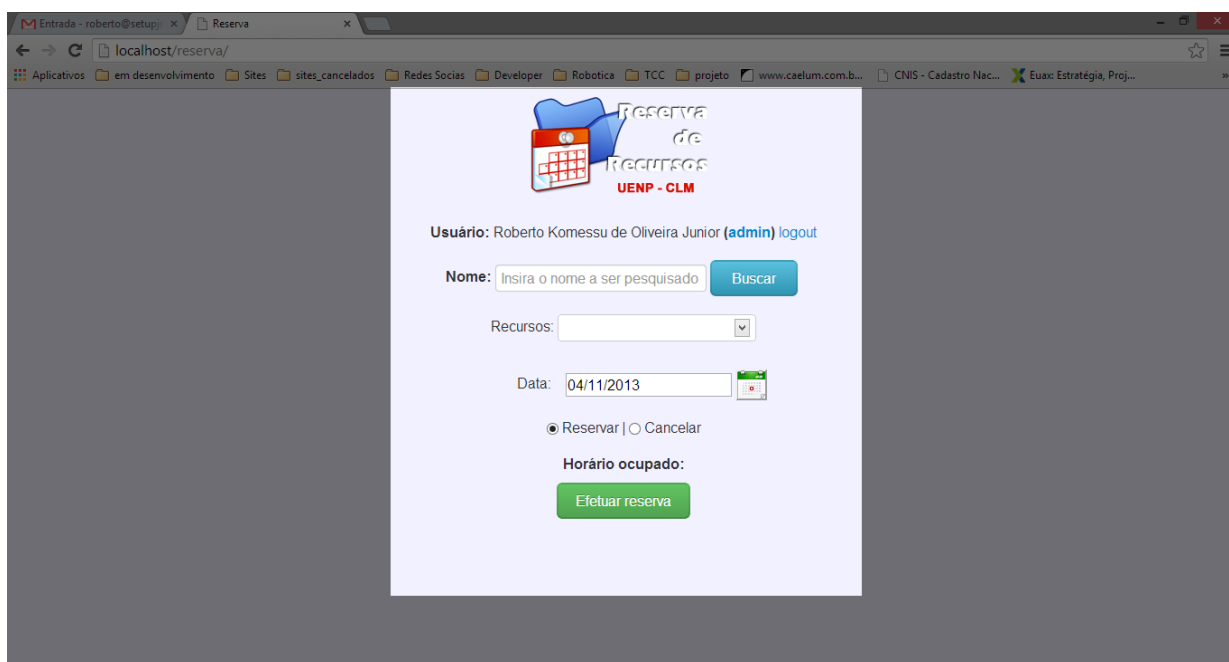
Para objeto de estudo foi necessário o desenvolvimento de um sistema, nesta seção será mostrado o seu funcionamento e suas principais funcionalidades.

O sistema desenvolvido será de uso do CCT, da Universidade apresenta somente duas telas principais, Tela de Consulta de reservas e Tela de Administração.

4.4.1 Tela de Consulta

Após autenticar-se por Login e Senha, o Usuário será direcionado para a primeira tela (Figura 14), identificado pelo seu nome completo, caso seja administrador, o usuário terá opções de acesso para painel “admin”, opção de reserva para outra pessoa efetuando uma busca pelo nome, ou cancelando qualquer reserva feita. Caso o usuário seja não tenha “admin”, terá as opções de reservas pessoais e consultas para qualquer recurso (figura 16).

Figura 14 - Tela de consulta (admin)

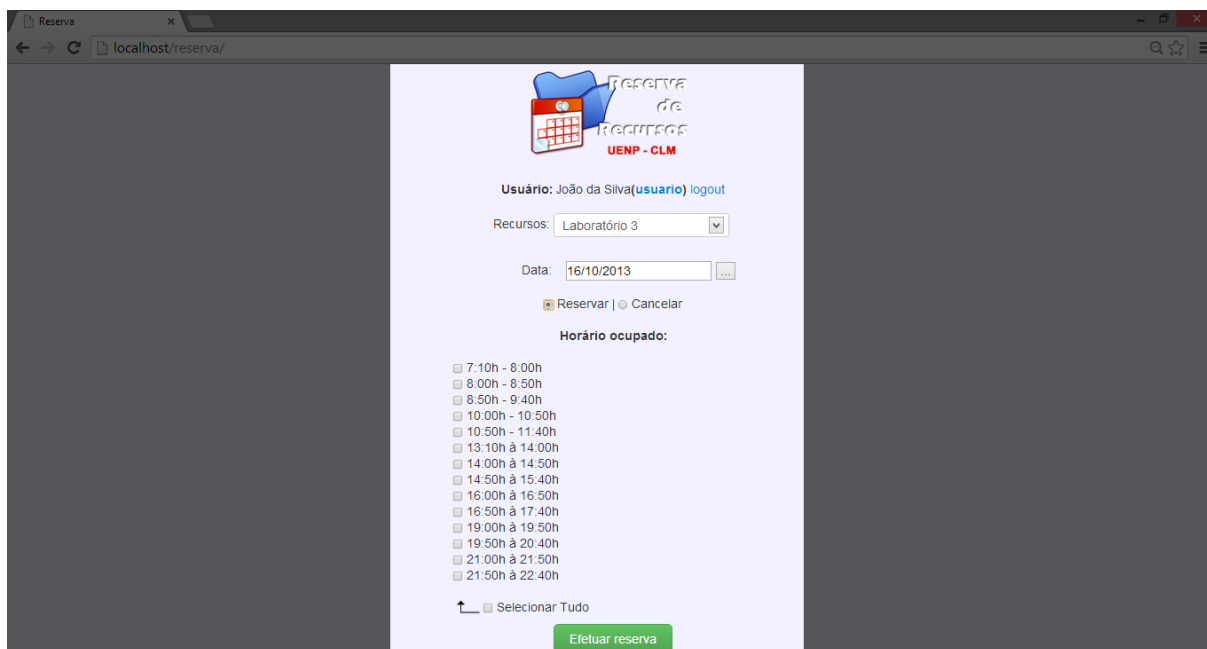


Fonte: Software Reserva, o autor

Os passos para efetuar reserva são simples:

1. Insira o nome/sobrenome do Usuário que será reservado (caso seja admin);
2. Selecione o recurso que será reservado;
3. Insira uma data manual no campo, ou clique no calendário e selecione;
Serão mostrados todos os horários do recurso para data selecionada.
4. Selecione os horários que deseja e clique em Efetuar reserva.

Figura 15 - Tela de consulta (usuário)



Fonte: Software Reserva, o autor.

Para reserva com permissões de Usuário são seguidos os mesmos passos.

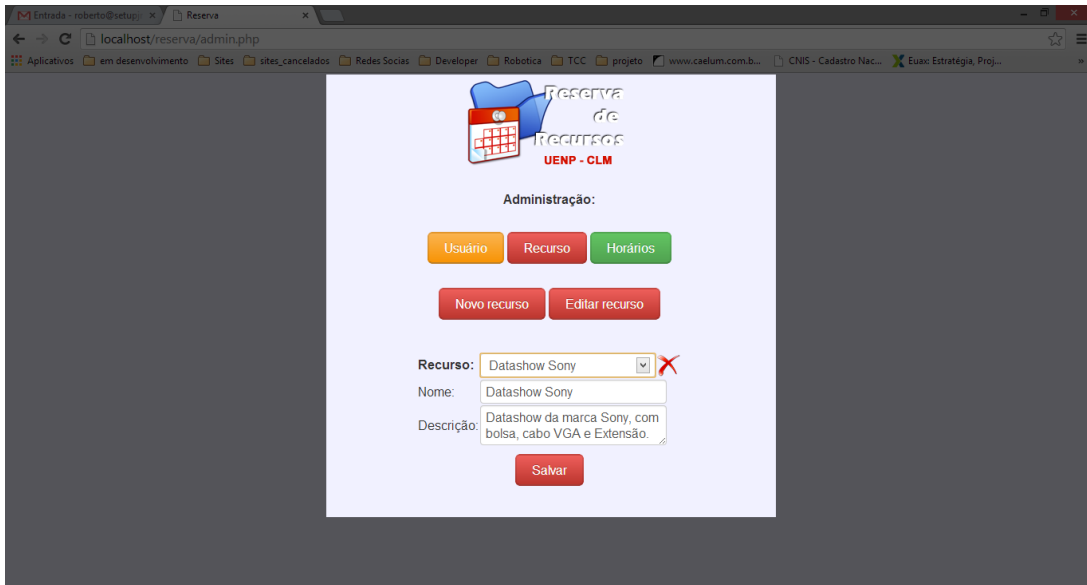
E para cancelar algum horário, siga os mesmos passos mas selecione a opção Cancelar, automaticamente o botão “Efetuar reserva” passará a ter a cor vermelha e se chamará “Cancelar reserva”.

4.4.2 Tela de Administração

No software foi criada uma administração simples para manter as opções de Usuário, Recursos e Horários. As telas são carregadas por meio Javascript do navegador e as consultas de banco de dados são feitas por meio de carregamento assíncrono (Ajax).

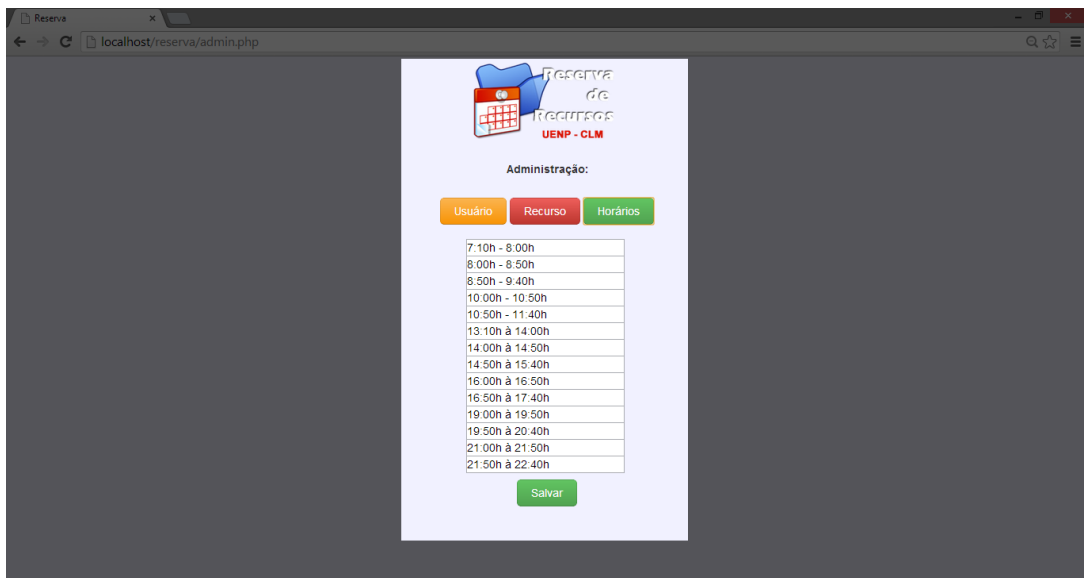
Nesta tela de administração, os usuários com permissões de administrador tem a opção de criar novos usuários, recursos ou alterar qualquer um desses, inclusive os horários pré-definidos para as reservas (Figura 17).

Figura 16 - Tela de Administração



Fonte: Software Reserva, o autor

Figura 17 - Tela de Administração de Horários



Fonte: Software Reserva, o autor

5 RESULTADOS OBTIDOS

Os resultados com a aplicação da técnica na produção de software evidenciou-se que o uso da Rastreabilidade de Requisitos na gerência e em uma possível manutenção de software, pode reduzir o esforço aplicado para a gestão de mudanças e uma clara visualização dos elos que mostram a ligação entre os artefatos elicitados do sistema, proporcionando aos responsáveis pelo desenvolvimento um fácil acesso a qualquer mudança e/ou evolução que possa vir a ocorrer.

O que pode se observar também do ponto de vista da gerência do desenvolvimento, é que a rastreabilidade possibilita identificar rapidamente artefatos (casos de uso, requisitos de projeto e de implementação), quando afetados por uma solicitação de mudança, possibilitando avaliar o impacto das alterações necessárias e medir alterações o seu custo no desenvolvimento. A Rastreabilidade auxilia a validação do software junto ao cliente, evidenciando a entrega da implementação em relação aos requisitos estabelecidos.

Após a busca de ferramentas, o software “Controla”, foi o que possibilitou a facilidade de uso e viabilidade para apoio da rastreabilidade em projetos pequenos, como foi o sistema “Reserva”. O software “Controla” foi utilizado como ferramenta para a geração de Matrizes de Rastreio, este que forneceu uma visualização das ligações dos artefatos e uma melhor visão do impacto de uma possível mudança. E de acordo com os metamodelos estudados, Ramesh e Jarke citam uma necessidade de definição de tipos de elos (motivo/causa que os artefatos são ligados ou dependentes), o “Controla” não supriu essa necessidade de definição, mas por outro lado, forneceu um rastreio automático e preciso de Casos de Uso X Implementações X Requisitos, portanto, com a ferramenta foi possível o rastreio sem a definição de elos, esta que não modificou o resultado na visualização do impacto da mudança.

5.1 ANÁLISE DAS MATRIZES

Na primeira matriz gerada (Figura 9), observou-se que no caso do sistema “Reserva”, os casos de usos atendem necessariamente dois ou mais requisitos,

garantindo assim que em qualquer alteração ou evolução de algum requisito, exista uma dependência entre eles, reforçando a necessidade do modelo de rastreabilidade. Na segunda matriz (Figura 18), pode-se percorrer o caminho até as implementações do sistema, observando a ligação que existe agora entre Requisito X Caso de Uso X Implementação do Sistema. Essa observação revela que é visível a necessidade de um modelo de rastreabilidade e com a Matriz de rastreio, se feita de acordo com o processo, torna-se um trabalho menos árduo para análise de um sistema, seja ele complexo ou não.

Figura 18 – Matriz: Implementações X Casos de uso
após alteração de requisito.

Rastreabilidade Casos de Uso -> Implementações											
Atualizar Relatório											
	PR5-BD: Consultar Usuários	PR6-BD: Consultar Horarios	PR7-BD: Consulta Recursos	PR8-BD: Consulta Reserva	PR11-Fn: Verifica Campos Vazios	PR13-BD: Consultar tipo de usuário	PR14-Fn: Verificar Reserva a ser cancelada	PR15-Interface: Mostrar Calendario	PR16-BD: Inserir Reserva	PR17-BD: Deletar Reserva	PR18-BD: Novo Recurso
UC1-Manter Usuario	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
UC2-Manter Recurso	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
▶ UC3-Manter Horário	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
UC4-Efetuar reserva	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
UC5-Cancelar Reserva	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
UC6-Reserva para terceiro	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
UC7-Consultar horários	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
											<input type="checkbox"/> - ítem modificado
Reserva de Recursos					admin						

Fonte: Software Controla

6 CONSIDERAÇÕES FINAIS

O uso adequado da rastreabilidade pode facilitar uma série de atividades do processo de desenvolvimento, podendo também auxiliar na melhoria da qualidade, tanto para o processo quanto para o produto de software.

A engenharia de requisitos define um dos mais importantes conjuntos de atividades a serem realizadas em projetos de desenvolvimento de software. Este trabalho apresentou a área de conhecimento de Rastreabilidade de Requisitos, parte importante que embora não defina a total qualidade dos produtos gerados, é mostrada como um requisito influencia para que se obtenha sucesso no desenvolvimento de projeto de software. É mostrado também que gerenciar os requisitos durante todas as fases em um projeto e aplicar as métricas especificadas pela Rastreabilidade, irá auxiliar a equipe na tarefa de estabelecer processos de desenvolvimento e melhorar a visão do escopo. No processo de rastreabilidade foram gerados as matrizes de rastreio, que possibilitaram visualizar a evolução de elos de rastreabilidade, assim as ligações de cada requisito no processo e no conjunto de tarefas, buscando com isso mostrar a importância do registro da rastreabilidade e sua evolução durante o processo de desenvolvimento.

Sobre os modelos, ficou evidente a perspectiva de que uma melhor compreensão das relações entre os artefatos, já que é uma tarefa complexa, entender o artefato pela sua ligação, origem e dependência.

Como resultado, viu-se que na prática as técnicas em Rastreabilidade utilizadas para a implementação de software não são simples. Também foi observado que os interessados na Rastreabilidade têm foco em itens específicos e necessitam de ferramentas para auxiliar o acompanhamento e evolução dos requisitos, reduzindo acentuadamente o trabalho da equipe de desenvolvimento.

REFERÊNCIAS

BECK, Kent. FOWLER, Martin. Planning Extreme Programming. 160 p. Addison Wesley Professional; 1st edition, October 2000.

BROOKS, F. P. J. The mythical man-month: essays on software engineering. Reading { MA: Addison Wesley, 1995. 2.ed.

DALL'OGGIO, Pablo. PHP Programando com Orientação a Objetos: Inclui Design Patterns. 1 ed. São Paulo: Novatec, 2007. 576 p. ISBN 978-85-7522-137-2

GOTEL, O. C. Z. Contribution structures for requirements traceability. 354 p. Submitted for the Diploma of Imperial College of Science, Technology and Medicine. Thesis (Doctor of Philosophy) | Faculty of Engineering of the University of London, London, 1995.

GOTEL, O. C. Z.; FINKELSTEIN, A. An analysis of the requirements traceability problem. In: FIRST INTERNATIONAL CONFERENCE REQUIREMENTS

LEFFINGWELL, D; Calculating the Return on Investment from More Effective Requirements Management; *American Programmer* 10(4); 13-16; 1997.

LEITE, J. C. S. P. "Engenharia de Requisitos", Notas de aula, PUC-Rio, Rio de Janeiro, 1994.

LEITE, J.C.S.P. "Gerenciando a Qualidade de Software com Base em requisitos", Prentice Hall , 2001, pp. 238-246.

LEE, CHRISTOPHER. GUADAGNO, LUIGI. JIA, XIAOPING. "An Agile Approach to Capturing Requirements and Traceability", Outubro 2003, Montreal, Canada, p. 1723.

JONES, CAPER, "Defense software development in evolution," CrossTalk, November, 2002, pp. 26-29.

KOTONYA, Gerald; SOMMERVILLE, Ian. Requirements Engineering: Processes and Techniques. 1998.

LEFFINGWELL, Dean; WIDRIG, Don. Managing software requirements: a unified approach. Addison-Wesley Professional, 2000.

PINHEIRO, F. A. C., "Formal and Informal Aspects of Requirements Tracing", III Workshop de Engenharia de Requisitos Rio de Janeiro, Brasil. (WER 2000), 2000.

RAMESH, B., JARKE, M., "Towards reference Models for Requirements Traceability", IEEE Trans. Software Eng., vol. 27(1), 2001. pp. 58-93.

SAYÃO, M. & LEITE, J. C. S. P. – “Rastreabilidade de Requisitos – relatório técnico 20/05”, série Monografias em Ciência da Computação, DI/PUC-Rio, 2005.

SOARES A. L. “Introdução, Identificação e Análise em Engenharia de Requisitos”. Antônio Lucas Soares. 2005.

SOMMERVILLE, “Software Engineering. Ian Sommerville - 6ª edição”. Addison Wesley, 2001.

SCHNEIDER, G. E, WINTERS, J., "Applying Use Cases: a practical guide", Addison Wesley, 1998.

THAYER E DORFMAN, “Software Requirements Engineering”. IEEE Computer Society Press. 1993.

TORANZO, MARCO, J. CASTRO, ELTON MELLO. "Uma proposta para melhorar o rastreamento de requisitos." Proceedings of the V Workshop on Requirements Engineering. 2002.

WIEGERS K. E.; “Software Requirements, Microsoft Press”, 1999

YIN, R. K. *Estudo de Caso: planejamento e métodos*. 2. ed. Tradução Daniel Grassi. Porto Alegre: Bookman, 2001.