

UNIVERSIDADE FEDERAL DO MARANHÃO
CENTRO DE CIÊNCIAS EXATAS E TECNOLOGIA
DEPARTAMENTO DE INFORMÁTICA
CURSO DE BACHARELADO EM CIÊNCIA DA COMPUTAÇÃO

JOSÉ LIMA FILHO

**PROCESSO DE DESENVOLVIMENTO DE SOFTWARE: aplicação de
rastreamento de requisitos no sistema WEBMET**

São Luís
2013

JOSÉ LIMA FILHO

PROCESSO DE DESENVOLVIMENTO DE SOFTWARE: aplicação de rastreabilidade de requisitos no sistema WEBMET

Monografia apresentada ao Curso de Ciência da Computação da Universidade Federal do Maranhão como parte dos requisitos necessários para obtenção do grau de Bacharel em Ciência da Computação.

Orientador: Prof.^a MSc. Maria Auxiliadora Freire

São Luís
2013

Lima Filho, José.

Processo de desenvolvimento de software: aplicação de rastreabilidade de requisitos no sistema WEBMET / José Lima Filho. – São Luís, 2013.

61f.

Impresso por computador (fotocópia).

Orientador: Maria Auxiliadora Freire.

Monografia (Graduação) – Universidade Federal do Maranhão, Curso de Ciência da Computação, 2013.

1. Gerencia de requisitos. 2. Matriz de rastreabilidade de requisitos. I Título.

CDU 004.451.2

JOSÉ LIMA FILHO

**PROCESSO DE DESENVOLVIMENTO DE SOFTWARE: aplicação
de rastreabilidade de requisitos no sistema WEBMET**

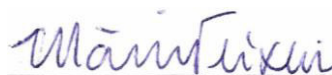
Monografia apresentada ao Curso de Ciência da Computação da Universidade Federal do Maranhão como parte dos requisitos necessários para obtenção do grau de Bacharel em Ciência da Computação.

Aprovada em: 07/03/2013

BANCA EXAMINADORA



Prof.^a MSc. Maria Auxiliadora Freire (Orientadora)
Universidade Federal do Maranhão



Prof. Dr. Mario Antonio Meireles Teixeira
Universidade Federal do Maranhão



Prof. MSc. Carlos Eduardo Portela Serra de Castro
Universidade Federal do Maranhão

À minha família, pela grande confiança e
companheirismo. Aos meus mestres. Aos meus
amigos.

AGRADECIMENTOS

Agradeço a Deus por sua graça em me conceder mais esta vitória.

Agradeço a minha esposa, Leandra Pereira Lima, por demonstrar sempre paciência e ternura durante esta fase de conclusão do meu curso e por me incentivar nas horas de desânimo.

Agradeço minhas filhas, Larysse Pereira Lima e Letícia Pereira Lima, por todo carinho e sorrisos que me encantam todos os dias como se fossem únicos.

Agradeço a minha família e em especial minha mãe, Eunice Rodrigues Lima, pela confiança que sempre depositaram em mim.

Agradeço aos colegas de curso, que sem dúvida possuem a maior participação para a conclusão deste curso, em especial a turma 2005.2 por estarem comigo desde o começo nesta jornada.

Agradeço a Prof.^a MSc. Maria Auxiliadora Freire, pelo apoio, atenção e contribuição dada na elaboração deste trabalho.

A todos os professores do curso de Ciência da Computação, pelos ensinamentos passados durante a minha vida acadêmica.

“A força não provém da capacidade física
e sim de uma vontade indomável.”

(Mahatma Gandhi)

RESUMO

A Gerência de Requisitos busca a identificação, controle e rastreamento dos requisitos de software. Entretanto, nas micro e pequenas empresas esse gerenciamento é pouco realizado, geralmente pela falta de recursos financeiros e humano e pelo uso de métodos informais no desenvolvimento. Este trabalho trata da utilização da Matriz de Rastreabilidade e sua importância no processo de desenvolvimento de software, mostrando que um bom rastreamento de requisitos proporciona qualidade e eficiência no desenvolvimento de softwares.

Foi realizada uma revisão sobre o tema Rastreabilidade de Requisitos e a aplicação da Matriz de Rastreabilidade nos projetos de desenvolvimento de software, além de apresentar as melhores práticas na construção das matrizes e as ferramentas que automatizam este processo.

Palavras-chave: Gerência de Requisitos, Matriz de Rastreabilidade de Requisitos, Requisitos, Rastreabilidade de Requisitos.

ABSTRACT

The Requirements Management seeks to identify, track and trace software requirements. However, in micro and small businesses that management is seldom performed, usually by a lack of human and financial resources and the use of informal methods in development. This paper deals with the use of Traceability Matrix and its importance in the process of software development, showing that a good requirements traceability provides quality and efficiency in software development.

We performed a literature review and application Requirements Traceability Matrix Traceability in software development projects, in addition to presenting the best practices in the construction of the matrices and tools that automate this process.

Keywords: Requirements Management, Requirements Traceability Matrix, Requirements, Requirements Traceability.

LISTA DE FIGURAS

| | |
|--|----|
| FIGURA 2.1 - Atividades da engenharia de requisitos..... | 19 |
| FIGURA 2.2 - Usuários de um documento de requisitos..... | 22 |
| FIGURA 2.3 - Leitores de diferentes tipos de especificação..... | 24 |
| FIGURA 2.4 - Classificação de requisitos não-funcionais..... | 25 |
| FIGURA 2.5 - Gerenciamento de mudança de requisitos..... | 27 |
| FIGURA 3.1 - Pré e Pós Rastreabilidade..... | 29 |
| FIGURA 3.2 - Metamodelo para rastreabilidade proposto por Ramesh&Jarke..... | 32 |
| FIGURA 3.3 - Metamodelo de rastreabilidade proposto por Toranzo..... | 34 |
| FIGURA 5.1 - Projeto WEBMET criado na ferramenta Controla..... | 48 |
| FIGURA 5.2 - Cadastro de um requisito na ferramenta Controla..... | 48 |
| FIGURA 5.3 - Cadastro dos requisitos na ferramenta Controla..... | 49 |
| FIGURA 5.4 - Cadastro de um caso de uso na ferramenta Controla..... | 49 |
| FIGURA 5.5 - Cadastro dos casos de uso na ferramenta Controla..... | 50 |
| FIGURA 5.6 - Matriz de dependência entre requisitos, sem elos criados, na ferramenta Controla..... | 50 |
| FIGURA 5.7 - Matriz de dependência entre requisitos, com elos criados, na ferramenta Controla..... | 51 |
| FIGURA 5.8 - Requisitos cadastrados com estado aprovado, na ferramenta Controla..... | 52 |
| FIGURA 5.9 - Matriz de rastreabilidade entre casos de uso e requisitos, com elos criados, na ferramenta Controla..... | 52 |

| | |
|---|----|
| FIGURA 5.10 - Histórico de mudança no estado do requisito, na ferramenta Controla..... | 53 |
| FIGURA 5.11 - Requisitos com mudança no estado, na ferramenta Controla..... | 54 |
| FIGURA 5.12 - Matriz de rastreabilidade entre casos de uso e requisitos, com rastreio de Itens modificados..... | 54 |
| FIGURA 5.13 - Diagrama de caso de uso macro do sistema WEBMET..... | 55 |
| FIGURA 5.14 - Diagrama de caso de uso..... | 56 |
| FIGURA 5.15 - Tela de Login do sistema WEBMET..... | 57 |
| FIGURA 5.16 - Tela troca de senha do sistema WEBMET..... | 57 |
| FIGURA 5.17 - Tela inicial após Login do sistema WEBMET..... | 58 |
| FIGURA 5.18 - Tela de observação meteorológicas do sistema WEBMET..... | 58 |

LISTA DE TABELAS

| | |
|--|----|
| TABELA 3.1 - Matriz de Rastreabilidade..... | 37 |
| TABELA 4.1 - Características da ferramentas de rastreabilidade..... | 44 |

LISTA DE ABREVIATURAS E SIGLAS

CASE - Computer-Aided Software Engineering

CSV - Comma-separated Values

EA - Enterprise Architect

ERS - Software Requirements Specification

GCS - Gerência de Configuração de Softwares

HTML – HyperText Markup Language

IBM - International Business Machines

IEEE - Institute of Electrical and Electronic Engineers

METAR – METeorological Aerodrome Report

OPMET – Operações Meteorológicas

PDF – Portable Document Format

PDL - Program Description Language

RTF - Rich Text Format

SPECI – Informação meteorológica especial para a aviação

SWEBOK - Guide to the Software Engineering Body of Knowledge

UML- Unified Modeling Language

WEBMET - Sistema automatizado de registros e gerenciamento das observações meteorológicas

XML - eXtensible Markup Language

SUMÁRIO

| | |
|---|-----------|
| 1 INTRODUÇÃO..... | 16 |
| 1.1 Objetivo geral..... | 16 |
| 1.2 Objetivos específicos..... | 16 |
| 1.3 Justificativa..... | 17 |
| 1.4 Organização do trabalho..... | 17 |
| 2 ENGENHARIA DE REQUISITOS..... | 18 |
| 2.1 Atividades da engenharia de requisitos..... | 18 |
| 2.1.1 Obtenção e análise de requisitos..... | 19 |
| 2.1.2 Especificação e documentação de requisitos..... | 21 |
| 2.1.3 Validação de Requisitos..... | 22 |
| 2.2 Tipos de requisitos..... | 23 |
| 2.3 Gerência de requisitos..... | 25 |
| 3 RASTREABILIDADE DE REQUISITOS..... | 28 |
| 3.1 Tipos de rastreabilidade..... | 28 |
| 3.2 Dificuldades relacionados à rastreabilidade..... | 29 |
| 3.3 Modelos de rastreabilidade..... | 31 |
| 3.3.1 Modelo Ramesh&Jarke..... | 31 |
| 3.3.2 Modelo Toranzo..... | 33 |

| | |
|---|-----------|
| 3.4 Matriz de rastreabilidade..... | 35 |
| 3.4.1 Tipos de elos de rastreabilidade..... | 36 |
| 4 FERRAMENTAS PARA CONSTRUÇÃO DA MATRIZ DE RASTREABILIDADE..... | 38 |
| 4.1 IBM Rational RequisitePro..... | 38 |
| 4.2 Enterprise Architect..... | 40 |
| 4.3 Controla..... | 42 |
| 4.4 Análise das ferramentas estudadas..... | 44 |
| 5 ESTUDO DE CASO - Aplicação de rastreabilidade de requisitos no sistema WEBMET..... | 46 |
| 5.1 Software WEBMET..... | 46 |
| 5.2 Principais funcionalidades do WEBMET..... | 46 |
| 5.3 Requisitos do sistema WEBMET..... | 47 |
| 5.3.1 Criando projeto WEBMET..... | 47 |
| 5.3.2 Cadastrando requisitos no projeto..... | 47 |
| 5.3.3 Cadastrando casos de uso no projeto..... | 49 |
| 5.3.4 Criando elos entre requisitos..... | 50 |
| 5.3.5 Matriz de rastreabilidade do projeto WEBMET com a ferramenta Controla..... | 51 |
| 5.3.6 Diagramas de casos de uso do sistema..... | 55 |
| 5.4 Implementação do sistema WEBMET..... | 56 |
| 6 CONCLUSÃO..... | 59 |
| 7 BIBLIOGRAFIA..... | 60 |

1. INTRODUÇÃO

Um dos problemas geralmente encontrados no desenvolvimento do software é a falta de controle dos requisitos, pois à medida que estes sofrem alterações, vários impactos ocorrem em artefatos e em outros requisitos relacionados aos mesmos. A falta de controle pode ocasionar aumento considerável de tempo de desenvolvimento e tornar o software instável. O gerenciamento de requisitos é um processo que deve ser realizado desde o início do projeto e se dedica a gerenciar as alterações nos requisitos.

Para Sommerville (2011), a Gerência de Requisitos (GR), consiste em compreender e controlar as mudanças nos requisitos dos sistemas, permitindo às organizações melhorar a qualidade de seus produtos e/ou serviços e aumentar sua capacidade de desenvolvimento.

Durante o processo de desenvolvimento de softwares vários artefatos são produzidos, tais como: requisitos, casos de testes e casos de uso. Uma informação bastante relevante para os envolvidos na produção de um software é a correlação existente entre esses vários artefatos, ou seja, se faz necessário manter continuamente uma rastreabilidade atualizada entre os vários artefatos produzidos.

A rastreabilidade traz vários benefícios para a qualidade e eficiência no desenvolvimento de softwares. Entretanto, devido à falta de recursos e pessoal, as pequenas empresas, na maioria, não realiza essa rastreabilidade. Outra barreira é que os artefatos são produzidos em ferramentas de apoio bastante distintas, fabricadas por diversas empresas. Muitas vezes as ferramentas não produzem artefatos que seguem um padrão aceito mundialmente que facilitasse a integração entre elas.

1.1 Objetivo geral

Apresentar a Matriz de Rastreabilidade de Requisitos e sua importância para no gerenciamento de requisitos de softwares e controle das modificações que venham a acontecer.

1.2 Objetivos Específicos

- a) Descrever o processo de Engenharia de Requisitos e suas atividades, dando ênfase à gerência de requisitos;
- b) Conceituação de Matriz de Rastreabilidade e seus benefícios;

- c) Caracterizar processos de desenvolvimentos nas Micros e Pequenas Empresas;
- d) Apresentar algumas ferramentas que automatizam a Rastreabilidade de Requisitos;
- e) Utilizar a ferramenta Controla com foco na Matriz de Rastreabilidade, através de um estudo de caso.

1.3 Justificativa

A proposta de realização deste trabalho monográfico de conclusão de curso é apresentar a importância da rastreabilidade no gerenciamento de requisitos, trazendo qualidade e eficiência no processo de desenvolvimento de softwares.

Este estudo contribuirá para a disseminação do conhecimento relacionado ao controle das inevitáveis mudanças que ocorrem durante o processo de desenvolvimento de software, acarretando alterações no orçamento e cronograma dos projetos, com a utilização da Matriz de Rastreabilidade de Requisitos.

1.4 Organização do trabalho

Esta monografia está dividida em seis capítulos. O primeiro capítulo corresponde à introdução do trabalho contextualizando os benefícios da rastreabilidade de requisitos de software, a justificativa e os objetivos deste trabalho.

No Capítulo 2, fez-se uma revisão da engenharia de requisitos e suas atividades no processo de desenvolvimento de software.

No Capítulo 3, trata-se da rastreabilidade de requisitos, descrevendo suas dificuldades, tipos de rastreabilidade, modelos e o uso da matriz de rastreabilidade.

No Capítulo 4, descreve-se as características principais das ferramentas para construir a matriz de rastreabilidade, com foco na ferramenta Controla.

No Capítulo 5, foi apresentado um estudo de caso aplicando a rastreabilidade de requisitos no sistema WEBMET através da ferramenta Controla.

No Capítulo 6, foi finalizado o trabalho mostrando as conclusões obtidas, as considerações finais.

2. ENGENHARIA DE REQUISITOS

A Engenharia de requisitos é uma atividade da Engenharia de Software que visa à produção (levantamento, registro, validação e verificação) e gerenciamento (controle de mudanças, configuração, rastreabilidade e qualidade) de requisitos de software.

Para Sommerville (2011) é um processo da Engenharia de Software que envolve as atividades relacionadas à produção e manutenção de requisitos de sistema. Contudo para desenvolver essas atividades, lista-se quatro subprocessos básicos de alto nível: viabilidade do sistema, a obtenção e a análise dos requisitos, a especificação de requisitos com sua documentação e a validação dos requisitos. Devido as mudanças inerentes ao requisitos, existe uma atividade que trata dessas alterações chamada gerência de requisitos e que será abordada neste capítulo.

Segundo Oliveira Espíndola (Revista Engenharia de Software Magazine, 2012), a busca por atender o que realmente o cliente deseja como produto de software, pode-se listar alguns objetivos da engenharia de requisito:

- buscar uma visão comum entre o cliente e os desenvolvedores do produto de software;
- fazer um acompanhamento dos requisitos ao longo do processo de desenvolvimento;
- fazer um controle e documentar os requisitos e estabelecer uma *baseline*¹ para uso gerencial;
- manter os requisitos consistentes com artefatos e atividades de software.

2.1 Atividades da engenharia de requisitos

Segundo Pressman (2006) para desenvolver suas funções a engenharia de requisitos necessita de sete atividades, que são: concepção, levantamento, elaboração, negociação, especificação, validação e gestão.

¹ Baseline é um conjunto de especificações ou produtos de trabalho que foram formalmente revisados e sobre os quais foi feito um acordo, que serve como base para desenvolvimento posterior e que pode ser modificado somente através dos procedimentos de controle de mudanças

Contudo, a classificação proposta por Sommerville (2011), vista anteriormente, que são: viabilidade do sistema, a obtenção e a análise dos requisitos, a especificação de requisitos com sua documentação e a validação dos requisitos será objeto desse estudo. A Figura 2.1 apresenta um modelo genérico das atividades (processo) da engenharia de requisitos

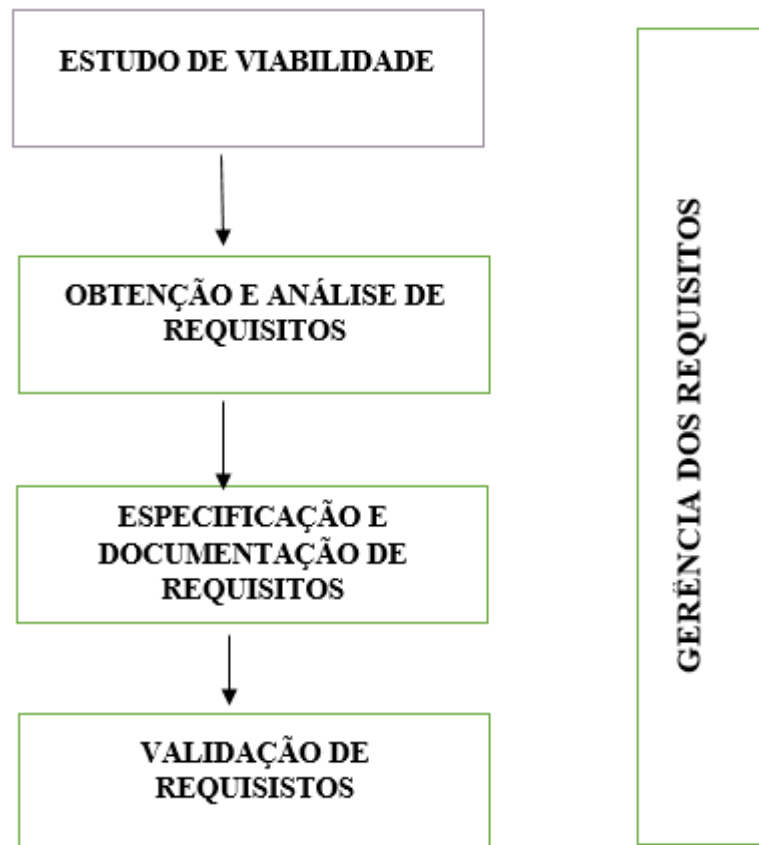


Figura 2.1: Atividades da engenharia de requisitos

2.1.1 Obtenção e análise de requisitos

Nessa atividade busca-se identificar junto aos interessados no projeto todas as informações inerentes ao sistema, tais como: objetivos do sistema, domínio da aplicação, restrições existentes, desempenho desejado, entre outros necessários para o projeto. Não é uma

etapa fácil devido aos problemas de entendimento dos requisitos pelos *stakeholders*² devido as várias razões, de acordo com Sommerville (2011).

1. Os *stakeholders* tem uma ideia muito subjetiva do que realmente querem do sistema, ignorando prazos e custos.

2. Os *stakeholders* expressam os requisitos de acordo com o conhecimento inerente a sua atividade, forçando um melhor entendimento por parte dos engenheiros.

3. Diferentes usuários expressam requisitos de forma diferentes. Os engenheiros precisam equalizar conflitos e consensos.

4. Fatores políticos podem influenciar requisitos, priorizando áreas específicas.

5. O dinamismo do ambiente pode alterar os requisitos, portanto novos podem surgir e outros perderem a importância.

A atividade de obtenção e análise de requisito deve ser feita de forma organizada, a fim de entender o ambiente, compreender as necessidades dos usuários e evitar transtornos durante a elicitação. Algumas técnicas são brevemente descritas a seguir:

- ✓ Técnicas tradicionais: são meios mais usuais de coleta de dados. Utiliza-se questionários e entrevistas, ambas elaboradas por um engenheiro com conhecimento do problema a ser questionado.
- ✓ Prototipação: parte do sistema é desenvolvido e tem por finalidade identificar mais requisitos ou possíveis falhas que antes não haviam sido detectadas. O cliente/usuários interage com o protótipo verificando se o projeto está caminhando de acordo com o esperado.
- ✓ Levantamento orientado a ponto de vista: usado para descobrir conflitos nos requisitos propostos por diferentes *stakeholders*.
- ✓ *Brainstorming*: é uma técnica em grupo para geração de ideias. Ela consiste em uma ou várias reuniões que permitem que as pessoas sugiram e explorem ideias.

² *Stakeholders* é qualquer pessoa ou grupo afetado pelo sistema, direta ou indiretamente

- ✓ Cenários: usa-se técnicas estruturadas como casos de uso e cenários de eventos com a finalidade de proporcionar uma visão mais prática do que realmente pretende os *stakeholders* do sistema.
- ✓ Etnografia: é utilizada para compreender os requisitos sociais e organizacionais, ou seja, entender a política organizacional, cultura de trabalho, objetivando familiarizar-se com o sistema e sua história.

2.1.2 Especificação e documentação de requisitos

Dependendo do contexto de desenvolvimento do projeto, a especificação de requisitos pode ter vários significados, desde um simples documento escrito, um modelo gráfico ou matemático formal, protótipo ou cenários, de acordo com PRESSMAN (2006).

A linguagem natural frequentemente usada para redigir especificações, se torna inviável quando se fala em requisitos de sistema devido a mesma ser ambígua e flexível pois é possível dizer a mesma coisa de maneira completamente distinta. No intuito de resolver esse problema, foram criadas alternativas para a especificação como o uso da linguagem natural estruturada, onde os requisitos são redigidos de forma padronizada.

Além da linguagem natural estruturada, pode-se utilizar a linguagem PDL (*Program Description Language*, Linguagem de Descrição de Programa) que possui a vantagem de gerar bons resultados de forma detalhada e sem ambiguidades. Utiliza-se também notações gráficas como a UML (*Unified Modeling Language*, Linguagem de Modelagem Unificada) usada para especificar, visualizar e documentar artefatos de software.

O ERS (*Software Requirements Specification*, especificação de requisitos de software) é o documento oficial que serve como guia para a implementação do software. Este documento deve conter todos os requisitos de usuários e uma especificação mais detalhada dos requisitos de sistema. O ERS deve ser elaborado para atender as exigências do sistema, além de contemplar seus diferentes tipos de usuários.

Sommerville (2011) mostra na Figura 2.2, como usuários diferentes utilizam o ERS.

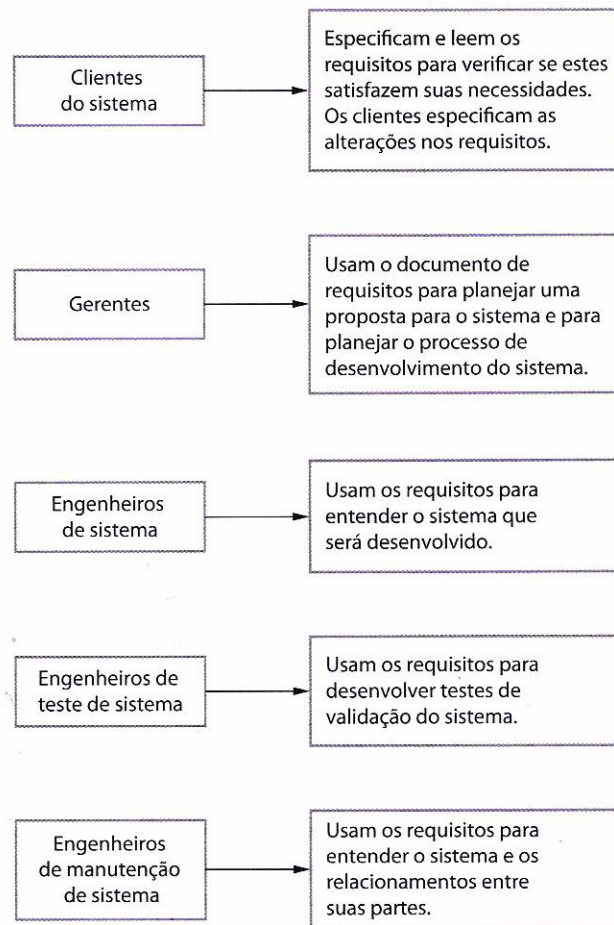


Figura 2.2: Usuários de um documento de requisitos
 Fonte: Sommerville, 2011

2.1.3 Validação de requisitos

É a forma de examinar os requisitos, buscando sua completeza e consistência junto aos *stakeholders*. Um documento de requisito mal elaborado pode ser dispendioso durante o processo quando se detecta requisitos com erros e inconsistentes. Clientes/Usuários e equipe de desenvolvedores devem revisar o documento e confirmar se ele está em conformidade com os padrões da empresa e do software em desenvolvimento, segundo *Guide to the Software Engineering Body of Knowledge - SWEBOK* (IEEE, 2004).

Durante a validação deve-se realizar algumas verificações de requisitos segundo Sommerville (2011).

- ✓ Verificações de validade: os *stakeholders* devem ter em mente que o documento de requisito foi confeccionado em conjunto, isto é, contempla todos os envolvidos e representa as funcionalidades exigidas por todos.
- ✓ Consistência: os requisitos não devem possuir nenhuma contradição ao que foi especificado.
- ✓ Completude: o documento de requisito deve conter todos os requisitos que definam funcionalidades e restrições do sistema.
- ✓ Realismos: o software deve estar de acordo com seu cronograma, orçamento e restrições que possam existir.
- ✓ Verificabilidade: os requisitos devem ser verificável, ou seja, permita que o cliente possa verificar se estão de acordo com o solicitado.

Além das revisões de requisitos, Swebok (IEEE,2004) propõe as técnicas de prototipagem e teste de aceitação:

- ✓ Prototipagem: busca validar os requisitos através de um protótipo executável em que o cliente verifica o funcionamento sistema. Isto facilita a interpretação dos requisitos pelo cliente e desenvolvedores.
- ✓ Teste de aceitação: os requisitos devem ser testados através de casos de testes antes de serem implementados, corrigindo assim possíveis falhas, erros e economizando tempo e dinheiro.

2.2 Tipos de requisitos

Paula Filho (2009, p.165) descreve que as seguintes definições de requisitos e que são aplicáveis e compatíveis com as terminologias e padrões de desenvolvimento vigente:

Condição ou potencialidade de que um usuário necessita para resolver um problema ou atingir um objetivo.

Condição ou potencialidade que um sistema, componente ou produto deve possuir para que seja aceito (isto é, satisfaça a um contrato, padrão, especificação ou outro documento formalmente imposto).

Expressão documentada dessa característica.

A indústria do software não trata os requisitos de software de forma consistente, considerando-o como uma declaração bem abstrata, de alto nível, como uma função ou restrição que o software deve atender. Devido essas diferentes percepções e descrições e no intuito de obter uma maior clareza e compreensão, Sommerville (2011) os distingue em:

1. Requisitos de usuário: declaração em linguagem natural juntamente com uso de diagramas com a finalidade de mostrar funcionalidades e restrições do sistema. É voltado para clientes/usuários com pouco conhecimento técnico.

2. Requisitos de sistema: possui maior detalhamento das funções e restrições operacionais do sistema, sendo chamado de documento de requisito de sistema. É direcionado para profissionais da área técnica e gerentes de projetos. Este documento define as implementações e servirá como um contrato entre cliente e desenvolvedor.

Os requisitos possuem diferentes níveis de detalhes com conhecimentos técnicos distintos. Por isso o documento de requisito deve contemplar a todos separando os requisitos do usuário dos de sistema de acordo com o tipo de leitor, como mostra a Figura 2.3, proposta por Sommerville (2011).

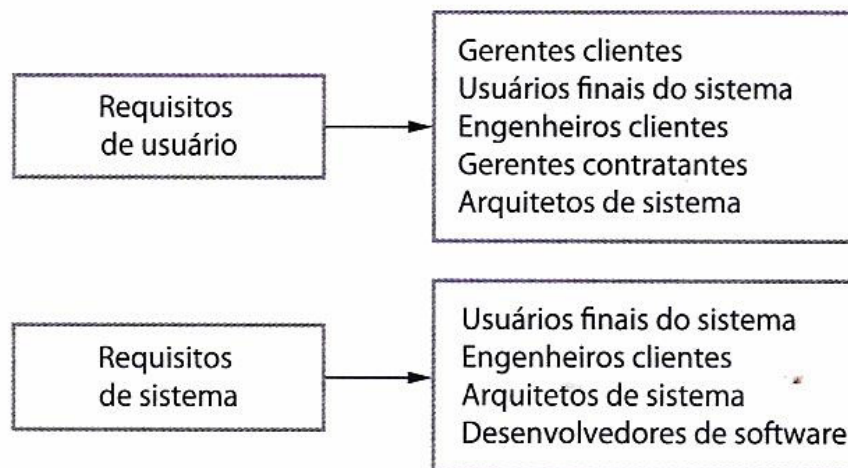


Figura 2.3: Leitores de diferentes tipos de especificação
Fonte: Sommerville, 2011

Os requisitos de sistema são ainda subdivididos em funcionais e não funcionais.

- Funcionais: descrevem como o sistema deve reagir a um estímulo de entrada e como deve se comportar em determinadas situações. Estabelece o que o sistema deverá fazer.

- Não-funcionais: descrevem as restrições que afetam as funções do sistema, podendo incluir algumas categorias como: segurança, confiabilidade, ambiente, padrões, entre outros.

Uma classificação de requisitos não-funcionais, figura 2.4, podem surgir devido alguma necessidade dos clientes/usuários, seja por questões orçamentária, nova função a ser desempenhada, questões de legislação ou alguma outra situação que necessite intervenção no projeto.

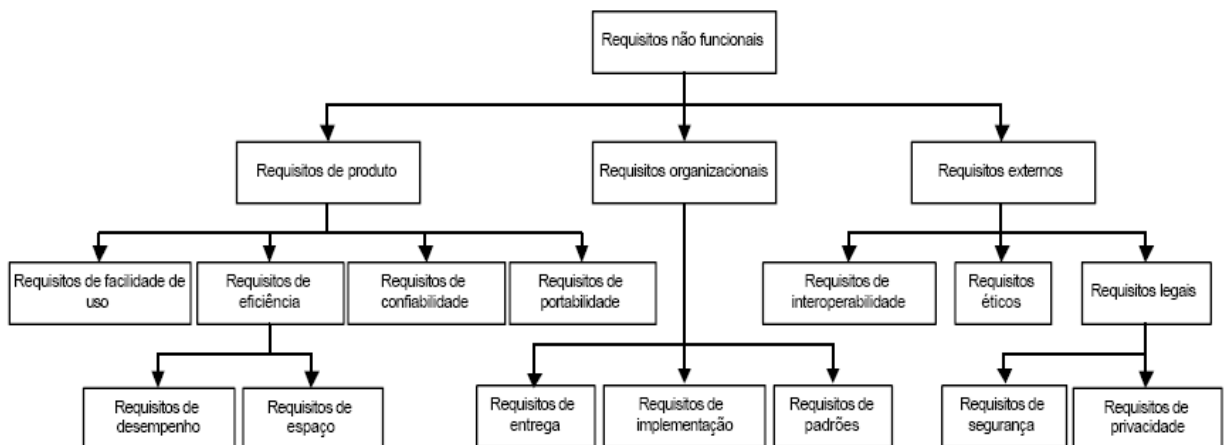


Figura 2.4: Classificação de requisitos não-funcionais
Fonte: Sommerville, 2011

Swebok (IEEE,2004, p.2-2) define ainda os requisitos como:

Requisitos de produto: Requisitos sobre o software a ser desenvolvido. Por exemplo “O software deve verificar se um aluno cumpriu os pré-requisitos antes de aceitar a sua matrícula numa disciplina”. **Requisitos de Processo:** são essencialmente restrições impostas ao desenvolvimento do software tais como linguagem de programação, processos e técnicas de desenvolvimento.

2.3 Gerência de requisitos

O gerenciamento de requisitos é uma atividade que tem por objetivo o controle, identificação, rastreamento e o gerenciamento das mudanças de requisitos do sistema. Além de controle versões de todos os artefatos utilizando a Gerência de Configuração de Softwares (GCS). Como os requisitos sofrem alterações, são voláteis, durante as fases do desenvolvimento, essa questão deve ser feita de forma contínua e controlada durante todo o

ciclo de vida do projeto. Em geral os requisitos sofrem mudanças devido a diversos fatores, tais como: erros, problemas de implantação, inconsistência, restrições orçamentárias, organizacionais ou ainda por um melhor entendimento dos mesmo pelos *stakeholders*.

De acordo com Swebok (2004) os requisitos possuem informações adicionais que auxiliam na interpretação por parte da gerência de requisitos:

- ✓ Resumo do produto;
- ✓ Fonte do requisito;
- ✓ Histórico das mudanças;
- ✓ Identificador: cada requisito deve ter uma única identificação, pois facilita a sua rastreabilidade. Um requisito é rastreável se for possível descobrir quem sugeriu, porque ele existe, seus relacionamentos e dependências com outros requisitos ou com o sistema.

Toda mudança de requisitos deve ser gerenciada, com a finalidade de avaliar o impacto e o custo das mudanças. Sommerville (2011) afirma que 65% da manutenção de sistema está relacionada à implementação de novos requisitos, 18% na alteração e 17% em correções de erros de sistema, considerando assim a manutenção parte do processo de desenvolvimento de software.

Esse processo deve ser feito de forma consistente e controlada durante o desenvolvimento. Segundo Sommerville (2011), existem 3 estágios nesse processo, que são:

- ✓ Análise do problema e especificação de mudanças: toda proposta de mudança de requisito deve ser verificada para analisar sua validade no sistema.
- ✓ Análise de mudança e custos: analisa-se os custos orçamentários que essas mudanças trarão ao projeto.
- ✓ Implementação de mudanças: o documento de requisitos deve estar estruturado de forma que as mudanças não afetem sua organização amplamente.

A Figura 2.5 mostra esses processos de gerencia de mudanças.



Figura 2.5: Gerenciamento de mudança de requisitos
Fonte: Sommerville, 2011

Pode-se ainda fazer uso de uma *baseline* de requisitos para organizar as mudanças pois permite diferenciar o requisito original, o que foi introduzido e o que descartado.

3.RASTREABILIDADE DE REQUISITOS

É uma técnica utilizada para relacionar os requisitos, projeto e implementação. Ou seja representa os relacionamentos entre os diversos artefatos³ produzidos durante o ciclo de vida do software, seja um relacionamento direto ou indireto. A rastreabilidade garante que os artefatos produzidos atendam ao requisitos propostos, sendo fundamental para o entendimento desses relacionamentos.

Gotel e Finkelstein (1994, p.97) definem rastreabilidade de requisitos como:

(...) habilidade de descrever e seguir a vida de um requisito a partir da sua origem (*backward*), como também através do seu desenvolvimento e especificação, até a sua distribuição e uso (*forward*); através de períodos de refinamento e iterações em qualquer uma dessas fases.

A rastreabilidade facilita a comunicação entre os *stakeholders*, além de permitir uma melhor avaliação do impacto das mudanças dos requisitos no projeto. Além disso, a rastreabilidade auxilia o desenvolvedor a: verificar alocação de requisitos a componentes de software; solucionar conflitos entre requisitos; verificar requisitos nos processos de testes; validar o sistema junto aos clientes; analisar o impacto na evolução do sistema; melhor avaliação de custos e prazos e gerenciamento de risco e reuso de componentes de software, de acordo com Sayão e Leite (2004).

3.1 Tipos de rastreabilidade

Gotel e Finkelstein (1994) definem dois tipos de rastreabilidade de requisitos: a pré-rastreabilidade – que é a habilidade de rastrear requisitos das origens ou fontes (clientes, usuários, normas, etc.) gerados no processo (são requisitos funcionais, não-funcionais, etc.). E a pós-rastreabilidade – que é a habilidade de rastrear requisitos com base no documento de requisito de softwares, através de vários artefatos de software nos quais os requisitos se relacionam.

³ Artefato é qualquer produto tangível resultado de várias etapas de fabricação de software.

Como a rastreabilidade é bidirecional, ou seja, é feita pra frente (*forward*) e pra trás (*backward*) existe ainda a seguinte classificação (FELICI,2004; BACKES,2008), conforme figura 3.1:



Figura 3.1: Pré e Pós Rastreabilidade.

✓ Na Pré-Rastreabilidade

- Rastreabilidade de frente-para (*forward-to tracebility*): rastreio de origem ou fontes (requisitos de cliente, requisito de sistema, etc.) para requisitos.
- Rastreabilidade de trás-de (*backward-from tracebility*): relaciona os requisitos às suas fontes (requisitos origem ou de cliente, requisito de sistema, etc.)

✓ Na Pós-Rastreabilidade

- Rastreabilidade de frente-de (*forward-forma tracebility*): relaciona os requisitos ao desenho, à implementação, etc. Permite avaliar o impacto de mudanças neste processo.
- Rastreabilidade de trás-para (*backward-to tracebility*): relaciona o desenho, às especificações, à implementação aos requisitos. Permite verificar se houve uma boa especificação de requisitos ou não.

3.2 Dificuldades relacionados à rastreabilidade

Embora esteja comprovado os benefícios do uso da rastreabilidade de requisitos, os custos com sua implantação são bastante elevados, o que dificulta essa prática em alguns

setores, principalmente nas pequenas empresas. Além disso, outros fatores dificultam o processo de rastreabilidade, tais como:

- ✓ Padronização – várias definições de rastreabilidade de requisitos causam diferentes entendimentos e conseqüentemente pode gerar uma má aplicação do rastreamento.
- ✓ Comprometimento organizacional – as organizações devem entender e reconhecer a importância do rastreamento, investir em tecnologia e instituir a sua prática no processo de desenvolvimento.
- ✓ Modelos de rastreabilidade – carência de modelos de rastreamento de requisitos que direcionem a elaborar um bom rastreamento. Falta orientações sobre o que rastrear, que informações devem constar em um modelo.
- ✓ Ferramentas de rastreio - existem diversas ferramentas que automatizam a rastreabilidade, porém cada uma possui um formato próprio para representar os artefatos, necessitando de profissionais com conhecimento específico para utilizá-las, acarretando gastos com treinamentos.

Com a finalidade de resolver alguns problemas relacionados com a rastreabilidade, um estudo (ARKLEY, MASON, RIDDLE, 2006) propôs as seguintes práticas para a rastreabilidade:

- Padronizar as representações dos artefatos, fins maior integração com as diversas ferramentas existentes no mercado.
- Repositório centralizado que facilite o desenvolvimento distribuído.
- Reuso das decisões tomadas em projetos anteriores.
- Ferramentas que alerte mudanças quando artefatos sofrerem alterações.
- Uso de técnicas de mineração de dados, fins descobrir quais artefatos estão potencialmente relacionados.

3.3 Modelos de rastreabilidade

Um modelo é um grupo coerente de elementos que descrevem algo construído através de alguma forma de análise para algum propósito particular, podendo ser expressado por uma linguagem (textual ou gráfica) que por sua vez possui um certo grau de abstração. No caso da rastreabilidade, os modelos são construídos com base nas informações sobre determinado domínio desse processo (usuário, práticas, metodologias, normas, etc.). Metamodelos representam um conjunto de conceitos em um determinado domínio e realçam as propriedades dos modelos originais.

3.3.1 Modelo Ramesh&Jarke

Ramesh&Jarke (2001, apud SAYÃO e LEITE, 2005, p.11 e 12) propõem um metamodelo para a rastreabilidade que possibilita a captura de informações relacionadas a agentes, fontes e objetos - as três dimensões dos modelos de rastreabilidade, conforme figura 3.2. Nesse metamodelo os interessados são ligados através de estruturas de contribuição aos objetos conceituais que eles influenciam e a documentos onde tais objetos são registrados. São apresentados objetos (relacionados ao produto sendo elaborado) e artefatos que são gerados pelo próprio processo de desenvolvimento. As três dimensões consideradas correspondem a:

- a) *Source* (fontes): documentos que remetem à origem dos requisitos (normas, padrões, legislação pertinente, atas de reuniões, etc.);
- b) *Stakeholders* (envolvidos): são as pessoas interessadas no processo de requisitos 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 gerados no processo de desenvolvimento.

Ramesh&Jarke ponderam que mesmo existindo uma grande variedade de elos de rastreabilidade, eles podem ser agrupados em duas categorias básicas, figura 3.2:

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

b) Relacionados ao processo: elos relacionados ao histórico de ações executadas no próprio processo; são subdivididos em elos de evolução e elos de *rationale*⁴.

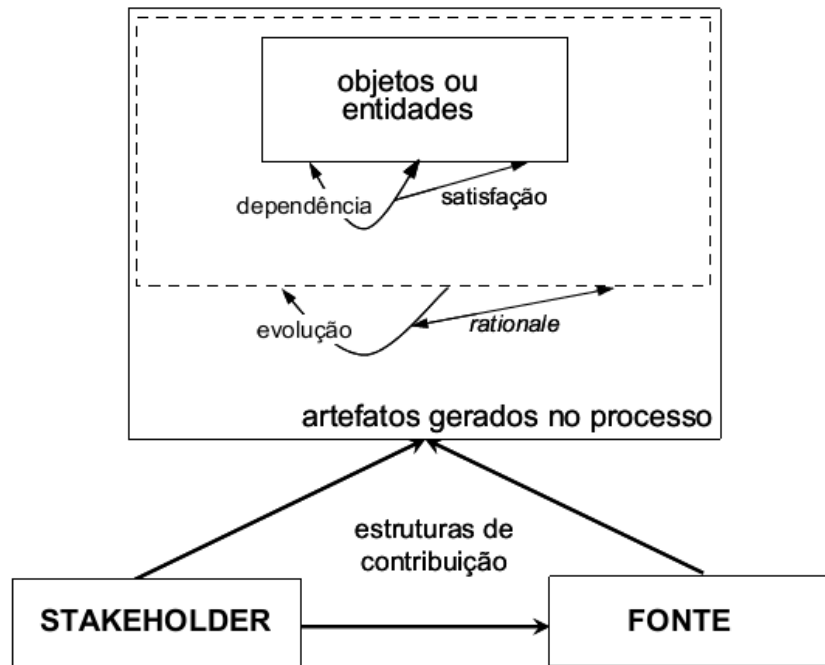


Figura 3.2 - Metamodelo para rastreabilidade proposto por Ramesh e Jarke (2001).

Fonte: Adaptado de Ramesh e Jarke (2001).

O propósito dos elos de satisfação é assegurar que os requisitos sejam atendidos pelo sistema, ou seja, a cada requisito foi associado um componente que deverá atendê-lo. Este tipo de elo é utilizado para registrar os desenhos criados para satisfazer requisitos e os componentes para os quais, os requisitos são alocados, para assegurar que todo componente satisfaça aos requisitos, registrar fatores críticos de sucesso associados a requisitos e assegurar consistência entre saídas das diferentes fases do ciclo de vida.

O propósito dos elos de evolução é registrar relacionamentos que levam de objetos existentes para objetos novos ou modificados. Este tipo de elo é útil para identificar as origens dos objetos para melhor compreensão dos requisitos e outros objetos (através de sua história) e para registrar as modificações e histórico de refinamentos dos vários objetos.

⁴ *Rationale* indica as razões, motivações ou intenções para uma determinada ação.

O propósito dos elos de *rationale* é representar as motivações subjacentes aos objetos existentes ou documentar as razões para evolução. Estes elos são utilizados para encontrar as justificativas para criação ou modificação de objetos, registrar suposições utilizadas no processo de decisão e identificar o contexto de criação de objetos. Estes elos possibilitam registrar aspectos do processo decisório, incluindo alternativas descartadas, de forma a providenciar clara compreensão da solução escolhida, facilitando a manutenção e o reuso. Em outras palavras, os elos de *rationale* auxiliam a gerenciar o desenvolvimento do sistema de acordo com as necessidades e objetivos organizacionais. Estes elos representam a área de atuação dos interessados, registrando as origens e o contexto no qual os objetos são desenvolvidos.

E finalmente, os elos de dependência têm por propósito apoiar o gerenciamento de dependências entre objetos, frequentemente impostas por restrições de recurso, de competência ou de compatibilidade, sendo úteis para registrar a composição e hierarquia dos objetos e apoiar o gerenciamento do impacto das alterações num objeto sobre os objetos que dele dependem.

3.3.2 Modelo Toranzo

Em seu modelo para a rastreabilidade Toranzo (2002, apud SAYÃO e LEITE, 2005, p.12 e 13) classifica as informações a serem rastreadas em quatro níveis:

- a) Ambiental: reúne informações referentes ao contexto ambiental onde a organização está inserida e que podem afetar o sistema a ser desenvolvido;
- b) Organizacional: reúne informações relacionadas à organização (missão, objetivos, metas e padrões) e que podem impactar os requisitos do sistema;
- c) Gerencial: agrega informações que permitem associar tarefas a requisitos, e que podem auxiliar a gerência do projeto;
- d) Desenvolvimento: congrega informações relacionadas aos diversos artefatos gerados no processo de desenvolvimento (documento de requisitos, diagramas, programas, casos de testes, etc.).

O metamodelo proposto, conforme figura 3.3, possui a metaclass base Elemento que possui as subclasses Elemento Generalizável, Relacionamento; instâncias de Elemento Generalizável se relacionam através de Generalização e Associação

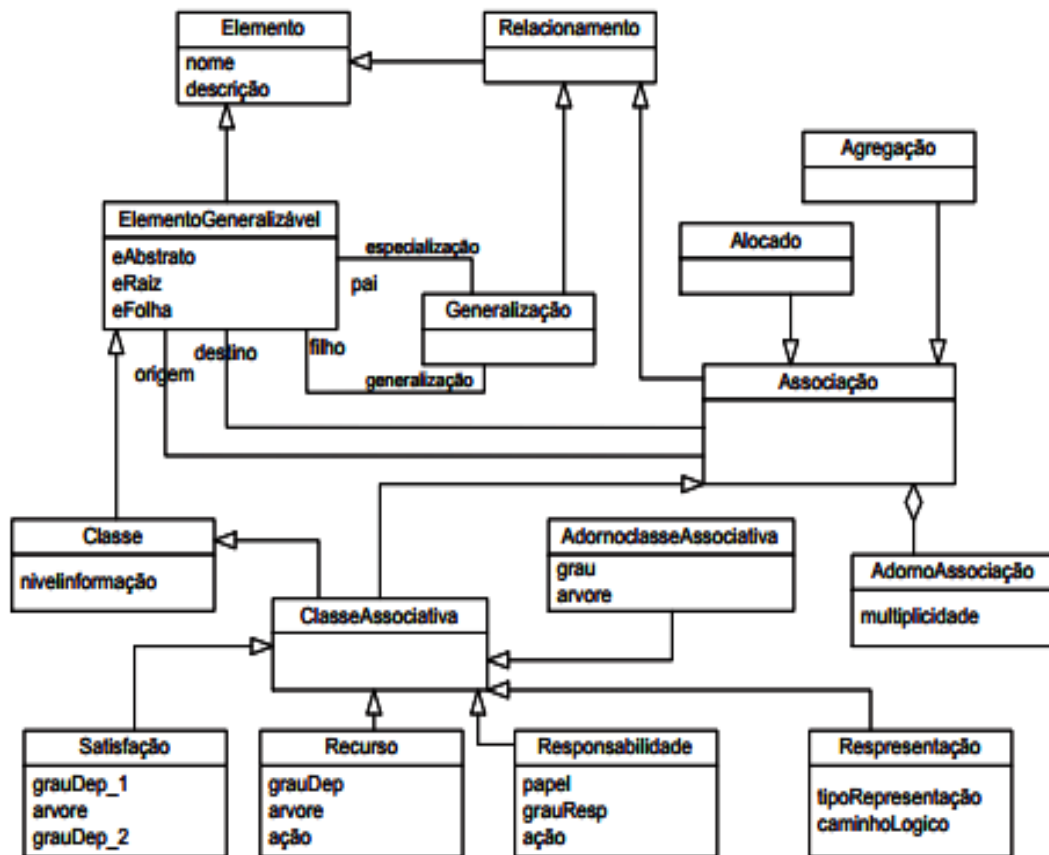


Figura 3.3 - Metamodelo de rastreabilidade proposto por Toranzo (2002).
 Fonte: Adaptado de Toranzo (2002).

Neste metamodelo, o suporte à rastreabilidade identifica os seguintes tipos de elos:

- satisfação: indica que a classe de origem tem dependência de satisfação com classe destino;
- recurso: indica que a classe de origem tem dependência de recurso com classe de destino;
- responsabilidade: registra a participação, responsabilidade e ação de pessoas sobre artefatos;
- representação: captura a representação ou modelagem dos requisitos em outras linguagens;
- alocado: classe de origem está relacionada à classe de destino, que representa um subsistema;
- agregação: indica composição de elementos.

Os metamodelos propostos permitem atender os aspectos básicos da rastreabilidade (registro das ligações entre requisitos, suas origens e artefatos de desenho e implementação) quanto a modelos mais sofisticados. O mérito da proposta de Ramesh&Jarke está na simplicidade destacando-se os aspectos evolutivos, registrando alterações sofridas pelos objetos rastreados, e as motivações para tais mudanças. Já Toranzo o foco dado aos aspectos gerenciais do projeto é uma de suas contribuições, dado que o gerenciamento de requisitos está intimamente ligado à gerência do projeto. Observa-se tanto nos elos de responsabilidade como na classificação proposta para as informações a serem rastreadas, enfatizando as influências dos contextos ambiental, organizacional e gerencial no projeto em desenvolvimento.

3.4 Matriz de rastreabilidade

É uma ferramenta para facilitar a visualização dos relacionamentos, ligação entre requisitos e outros artefatos ou objetos do projeto. Trata-se de uma tabela que liga os requisitos às suas origens e os rastreia durante todo o desenvolvimento do software. A utilização de uma matriz de rastreabilidade ajuda a garantir que cada requisito adiciona valor de negócio através de seus elos aos objetivos de negócio e aos objetivos do projeto. A matriz também fornece uma estrutura de gerenciamento das mudanças, permitindo assim que estimativas de custos e prazos

A matriz, qualquer que seja o tipo, sempre segue o mesmo modelo. Coloca-se os objetos a serem rastreados nos eixos de uma tabela e marca-se os pontos de intersecção dos mesmos. Na forma mais comum, a matriz de rastreabilidade entre requisitos ou de dependências, repete-se os requisitos nos eixos horizontal e vertical. Os atributos associados a cada requisito, isto é, informações adicionais que ajudam a interpretar e gerenciá-los, devem ser registrados na matriz de rastreabilidade. Os atributos típicos mais utilizados são:

- ✓ Identificador – requisitos deve ter uma identificação única e não ambígua;
- ✓ Descrição textual, ou seja, um resumo do requisito;
- ✓ Argumentos para a sua inclusão;
- ✓ Proprietário do requisito;
- ✓ Fonte do requisito, sua origem;
- ✓ Prioridade;

- ✓ Versão;
- ✓ Situação (ativo, cancelado, entre outros);
- ✓ Data conclusão;
- ✓ Critério de aceitação.

Pode-se manter a matriz de rastreabilidade manualmente em uma planilha, processadores de texto, banco de dados, mas essa prática se torna inviável para sistemas um pouco mais complexos, com uma grande quantidade de requisitos, sendo nesse caso aconselhável o uso de ferramentas que automatize esse rastreo, segundo Sommerville (2011).

3.4.1 Tipos de elos de rastreabilidade

É possível gerar matrizes de vários tipos. Sommerville (2011) sugere alguns tipos de elos entre requisitos e entre os requisitos e o projeto. Essas informações podem ser rastreadas através de matrizes com a finalidade de capturar 3 (três) características básicas como:

- Rastreamento da origem: são informações que ligam os requisitos à sua fonte, ou seja, os *stakeholders*.
- Rastreamento de requisitos: indica como os requisitos se relacionam entre si, isto é, características de dependência.
- Rastreamento de projeto: relaciona os requisitos ao projeto em que se encontra implementado.

A Tabela 3.1 exemplifica uma matriz de rastreabilidade. Nessa tabela as intersecções marcadas com a letra D especifica que requisitos da linha usa os de recursos dos requisitos da coluna, ou seja possui uma dependência e as marcadas com R significa um fraco relacionamento entre os requisitos.

Tabela 3.1- Matriz de Rastreabilidade

| 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 | |

Fonte: Sommerville (2011)

De acordo com Marquioni (2004, apud SANTOS, SCHUBERT e CÂMARA, 2008, p.3) existe seguintes tipos de elo de rastreabilidade:

- ✓ Requisitos – Fontes: elo do requisito às pessoas ou documentos que especificaram o requisito.
- ✓ Requisitos – Razão: elo do requisito com uma descrição do motivo da sua especificação.
- ✓ Requisitos – Requisitos: elo entre requisitos que sejam dependentes.
- ✓ Requisitos – Arquitetura: elo do requisito com os subsistemas onde estes requisitos estão implementados.
- ✓ Requisitos – Design: elo do requisito com hardware ou componentes de software específicos no sistema.
- ✓ Requisitos – Interface: elo do requisito com as interfaces externas.

4. FERRAMENTAS PARA CONSTRUÇÃO DA MATRIZ DE RASTREABILIDADE

Com objetivo de auxiliar a difícil atividade de gerência de requisitos, existem diversas ferramentas disponíveis no mercado, incluindo ferramentas CASE, seja ela paga ou gratuita. Essa ferramentas, na maioria, além de funcionalidade de gerenciamento com ênfase na rastreabilidade possuem também funções de modelagem UML.

Normalmente essas ferramentas permitem a criação de matriz de rastreabilidade, os relacionamentos entre os requisitos e posteriormente visualizar alguma alteração decorrente de mudanças nos requisitos.

Dentre as diversas ferramentas pode-se destacar as pagas *IBM Rational RequisitePro* da IBM e *EA (Enterprise Architect)* da SPARX e a gratuita *Controla*.

4.1 IBM Rational RequisitePro

O software *IBM Rational RequisitePro* é uma ferramenta de fácil uso para gerenciar requisitos e casos de usos, permitindo o compartilhamento e sincronização de requisito utilizando banco de dados e o Microsoft Word resultando numa melhor rastreabilidade, comunicação e análise de impacto com aumento da probabilidade de cumprir prazos, orçamentos, etc. Pode-se criar e compartilhar os seus requisitos usando métodos conhecidos baseados em documentos e, ao mesmo tempo, usar recursos de banco de dados, como rastreabilidade e análise de impacto. Isso pode melhorar a comunicação e o gerenciamento de requisitos, aumentar a qualidade e encurtar o tempo de entrada no mercado. Dentre suas características principais tem-se:

- ✓ Suporta o Microsoft Word para a criação e comunicação de requisitos;
- ✓ Complementa a entrada baseada em documentos com um banco de dados comercial para incluir recursos de organização, rastreamento e gerenciamento;
- ✓ Oferece integração com o Microsoft Word como uma opção para suportar equipes que preferem uma abordagem de banco de dados;
- ✓ Ajuda a assegurar que as equipes distribuídas tenham acesso de leitura e gravação aos requisitos a partir de qualquer local;

- ✓ Permite consultar esses relacionamentos para fazer uma análise de cobertura. Ajuda a assegurar a abrangência e permite evitar a perda de tempo ao construir uma parte do sistema que não é necessária;
- ✓ Fornece relatórios detalhados referentes à conformidade com as normas. É possível criar, visualizar e exportar matrizes de rastreabilidade filtráveis e relatórios de atributos para as necessidades de auditoria interna e externa;
- ✓ Permite que a administração do projeto seja realizada pela web;
- ✓ Permite configurar facilmente tipos de requisitos, atributos e documentos;
- ✓ Define consultas e filtros para localizar rapidamente informações de interesse;
- ✓ É adaptável ao seu processo. Os membros da equipe podem criar exibições facilmente para mostrar as informações de que eles precisam;
- ✓ Inclui a notificação automática por e-mail aos interessados quando os requisitos mudam;
- ✓ Cria uma linha de base dos requisitos do projeto baseada em XML. Pode-se usá-la para iniciar novos projetos ou compará-la com outras linhas de base de projeto, expondo mudanças de requisitos e omissões com vários níveis de detalhes;
- ✓ Suporta o desenvolvimento paralelo, no qual mais de um conjunto de requisitos deve coexistir;
- ✓ Determina quando e onde ocorrem as mudanças, para reduzir a confusão da equipe e esclarecer o efeito das decisões do projeto ao longo do tempo;
- ✓ Torna as informações sobre os requisitos uniformemente acessíveis e modificáveis por meio de produtos na plataforma de entrega de software IBM;
- ✓ Sincroniza toda a equipe e fornece rastreabilidade total ao longo do ciclo de vida integrando-se a esses produtos.

O software possui características de rastreabilidade bem acessíveis e fáceis de configurá-las com controle sistemáticos dos relacionamentos entre os requisitos e fornecendo uma análise

detalhada dessas relações. As consultas de rastreabilidade e análise podem ser salvas como visões e consultadas de forma rápida deixando claro o que poderá ainda ser feito.

4.2 Enterprise Architect (EA)

Essa ferramenta possui características multiusuário voltadas para a modelagem UML e integrando também à gerência de requisitos. O software auxilia todo o gerenciamento de requisitos de forma fácil e rápida, sobretudo quando se trata de rastreabilidade de requisitos, que é feita de forma completa e organizada.

Enterprise Architect (EA) representa um divisor de águas em relação às ferramentas de modelagem de processos, dados e sistemas, sendo de fácil utilização/aprendizado e que oferece o melhor custo-benefício do mercado, com algumas características, tais como:

- ✓ Suporte total ao ciclo de vida de modelagem de processos, dados e sistemas;
- ✓ Permite a utilização de técnicas de levantamento e documentação de Requisitos;
- ✓ Abordagem completa em Análise e Projeto de Sistemas conforme a UML;
- ✓ Automatização na engenharia de código (geração, reversa e sincronização), contemplando múltiplas linguagens de programação;
- ✓ Integração nativa com Visual Studio.NET e Eclipse;
- ✓ Geração de Documentação de apoio e Relatórios personalizados (em HTML e RTF);
- ✓ Rastreabilidade entre todos os elementos de (processos, regras, requisitos, casos de uso, classes, componentes, tabelas, etc.);
- ✓ *Workgroup*, possibilitando uso compartilhado e seguro pelos usuários;
- ✓ Exportação/troca de informações com outras ferramentas via XML;
- ✓ Permite criar protótipos de telas para validação e rastreabilidade de documentação.
- ✓ Cria e visualizar os requisitos como elementos diretamente no modelo;
- ✓ Grava meta-dados essenciais, incluindo uma breve descrição, o status, a dificuldade, a prioridade, tipo funcional, não funcional, e muito mais;

- ✓ Cria propriedades personalizadas, tais como estimativa de custo;
- ✓ Adiciona e vê requisitos em um formato de tabela hierárquica;
- ✓ Inter-relações modelo e dependências;
- ✓ Cor indicando o estado de cada elemento;
- ✓ Use a janela de rastreabilidade para analisar o impacto da mudança;
- ✓ Identifica e documenta as relações em um formato tabular com a Matriz de relacionamento;
- ✓ Grava contínuas mudanças no modelo de Requisitos usando auditoria;
- ✓ Cria um instantâneo do seu projeto usando linhas de base. Em seguida, compará-la com os estados futuros para identificar e reverter as alterações individuais;
- ✓ Mostra os casos de uso do modelo;
- ✓ Cria cenários para rapidamente obter exigências detalhadas;
- ✓ Modele cenários de uso com diagramas de sequência;
- ✓ Cria relatórios em formatos RTF, HTML e PDF documentação;
- ✓ Compartilhe um glossário de projeto comum para garantir um entendimento comum;
- ✓ Selecione as imagens alternadas para os elementos para melhor comunicar conceitos técnicos;
- ✓ Importe requisitos de um arquivo CSV externo;
- ✓ Arraste e solte os requisitos de documentos escritos em diagramas para criar automaticamente elementos exigência.

A matriz gerada é produto de relacionamento entre requisitos, geração de negócios e casos de usos fornecendo uma boa análise que garanta a integridade e a certeza que o sistema está respondendo ao que se espera dele. É uma ferramenta bastante intuitiva proporcionando resultado de rastreio tanto na forma de matriz com em estrutura de árvore.

4.3 Controla

É uma ferramenta gratuita que auxilia o gerenciamento de requisitos desenvolvida pela Faculdade Viçosa em Minas Gerais –MG. O software contempla diversas etapas do processo de qualidade de software e oferece diversas funcionalidades sendo as mais importantes segundo os desenvolvedores Fraga Filho e Maurício dos Reis (2005):

- ✓ Gerência de requisitos;
- ✓ Gerência de casos de uso;
- ✓ Gerência de casos de testes e erros;
- ✓ Gerência de implementações;
- ✓ Controle de dependência entre implementações;
- ✓ Priorização de requisitos;
- ✓ Matriz de rastreabilidade para os seguintes artefatos:
 - requisitos x casos de uso;
 - casos de uso x implementações;
 - casos de uso x casos de teste;
 - casos de testes x casos x erros;
 - liberações x casos de uso;
- ✓ Gerência de especificação de requisitos;
- ✓ Documento de casos de uso;
- ✓ Exportação de dados.

De acordo com Fraga Filho e Maurício dos Reis (2005) o gerenciamento de requisitos com a ferramenta Controla permite o gerenciamento dos requisitos, ou seja, regras e capacidades necessárias em um produto de software, utilizando como base descrição breve e completa da necessidade, suas restrições, data de criação, de finalização e responsáveis pela aprovação junto ao cliente. Outras informações essenciais a um requisito são prioridade, estabilidade e estado, que juntamente com as informações descritivas descrevem completamente o que deve contemplar o software solicitado pelo cliente. Finalmente, um

requisito pode ser proposto por mais de um *stakeholder*, sendo assim, na ferramenta é possível associar um ou mais proponentes a um mesmo requisito.

Na definição dos estados dos requisitos de um software, o Controla realiza validações, assim a ferramenta restringe operações inválidas por que podem ser executadas pelos analistas durante o gerenciamento destas informações, exibindo mensagens que possam o orientar. Os principais estados identificados para um requisito são:

- ✓ Proposto – indica que o requisito foi solicitado por uma fonte autorizada;
- ✓ Aprovado – seu impacto no projeto foi analisado, projetado e alocado para uma liberação específica. O usuário concordaram em incorporar o requisito, e o grupo de desenvolvimento de software já aceitou implementar tal requisito;
- ✓ Rejeitado – o requisito foi proposto mas não será incorporado em qualquer liberação;
- ✓ Em avaliação – o requisito foi proposto e está sendo avaliado por todos os usuários (*stakeholders*) e pela equipe de desenvolvimento;
- ✓ Implementado – a implementação do requisito foi projetada, realizada e testada. O requisito foi projetado e associado a um item de implementação;
- ✓ Verificado – o requisito foi testado para todo o produto de software, via teste de integração. O requisito foi rastreado aos casos de teste pertinentes. Neste estado o requisito é considerado completo;
- ✓ Atendido – o requisito atendeu às expectativas do cliente.

O controle de versão é um importante recurso oferecido nesta ferramenta, pois mantém documentadas as mudanças de cada requisito, possibilitando aos analistas consultarem e confrontarem os dados com a equipe de desenvolvimento e com os clientes, caso necessário. Esta funcionalidade é útil, pois reduz os desgastes provocados por discussões muitas vezes sem fundamento durante o processo de desenvolvimento de software.

A matriz de rastreabilidade é uma das principais funcionalidades da ferramenta Controla, pois permite visualizar de forma gráfica o relacionamento entre artefatos facilitando assim ao

analista a tomada de decisões e descobrir problemas na fase de implementação. Qualquer modificação no artefato é sinalizado, na cor amarela, como item modificado na matriz dando suporte ao analista na identificação de possíveis focos de problemas no desenvolvimento.

4.4 Análise das ferramentas estudadas

Tabela 4.1- Características da ferramentas de rastreabilidade

| Características | 1 | 2 | 3 |
|--|---|---|---|
| Oferece integração com Microsoft Word para criação e comunicação de requisitos; | S | N | N |
| Estabelecer dependências entre Requisitos Funcionais, Casos de Uso, Requisitos Não-Funcionais; | S | S | S |
| Permite configurar facilmente tipos de requisitos, atributos e documentos; | S | S | S |
| Cria relatório em formatos HTML, PDF; | S | S | N |
| Permite utilização de técnicas de levantamento e documentação de requisitos; | N | S | N |
| Definir requisitos e casos de uso na própria ferramenta; | S | S | S |
| Abordagem completa em Análise e Projeto de Sistemas conforme a UML; | N | S | N |
| Grava contínua mudança no modelo de Requisito; | S | S | S |
| Divulgar a Matriz de Rastreabilidade para todos os interessados; | S | S | N |
| Modela cenários de uso com diagramas de sequência; | N | S | N |
| Exportação de dados em XML, CVS; | S | S | S |
| Permite que a administração do projeto seja realizada pela web | S | N | N |
| Integração entre ferramentas Visual Studio.NET e Eclipse | N | S | N |

Legenda:

Ferramentas: (1) Requisite Pro; (2) Enterprise Architect; (3) Controla.

Avaliação: (S) Satisfaz; (N) Não satisfaz.

A ferramenta Enterprise Architect foi a que satisfaz a maioria dos requisitos citados, cobre todos os diagramas da UML permitindo o uso ao longo de todo o ciclo de vida do desenvolvimento, inclusive gerando código para Java, C#, HTML, etc. Possui um repositório que integra todos os artefatos de todas as fases do processo. Todavia, é uma ferramenta proprietária e sem acesso baseado na Web.

Para o estudo de caso proposto utilizar-se-á ferramenta Controla por se tratar de um software simples, gratuito e que atende as necessidades e objetivos do estudo abordado.

5 ESTUDO DE CASO – Aplicação de Rastreabilidade de Requisitos no Sistema WEBMET

5.1 Software WEBMET

É um sistema automatizado de registros e gerenciamento das observações meteorológicas. O sistema WEBMET tem como objetivos específicos:

- ✓ Envio de forma automatizada de dados meteorológicos de estações meteorológicas à superfície ao banco de dados climatológicos;
- ✓ Gerar códigos meteorológicos na forma METAR⁵ e SPECI⁶ e envia-los ao banco de dados meteorológicas (Banco OPMET) fins disponibilizar em toda rede de comunicação do Comando da Aeronáutica e também da internet. Esses códigos são de grande utilidade para segurança da navegação aérea pois informa os dados meteorológicos observados e previstos para determinadas localidades ou rotas aéreas e que podem ser acessados a qualquer momento via internet.

Possui interface Web e pode ser acessado via internet e intranet por meio de qualquer máquina (estação de trabalho, terminal, etc.) desde que previamente cadastrada ao sistema. É integrado externamente ao Banco OPMET e pode ser acessado através dos endereços eletrônicos: www.webmet.aer.mil e www.webmet.intraer.

5.2 Principais funcionalidades do WEBMET

O software WEBMET possui funcionalidades as quais pode-se destacar:

- ✓ Controle de usuários e estações meteorológicas de superfície;
- ✓ Cadastro de observações meteorológicas de superfície;
- ✓ Codificação de observações meteorológicas;
- ✓ Gerar os códigos METAR e SPECI através das observações meteorológicas inseridas;
- ✓ Enviar mensagem codificada ao banco de dados OPMET;

⁵ METAR reporte meteorológico regular de um aeródromo.

⁶ SPECI reporte especial de informação meteorológica de um aeródromo sempre que se justifique.

- ✓ Consultar as observações meteorológicas enviadas ao banco de dados;
- ✓ Emitir relatórios de dados enviados no formato de *Portable Document Format* (PDF).

5.3 Requisitos do sistema WEBMET

As fases de levantamento de requisitos e análise deve resultar numa especificação dos requisitos que sejam compreensível para os próprios usuários. Isso significa que deve ser utilizada a linguagem deles, que não é adaptada ao tipo de formalização que os profissionais de informática precisam.

Para encontrar os requisitos corretos o primeiro passo é conhecer o contexto do sistema. O modelo do negócio oferece uma solução para este problema, uma vez que descreve as operações dentro da “empresa” e de quem a responsabilidade de efetuá-las.

Para documento de requisitos e casos de uso utilizar-se à ferramenta Controla para o rastreio dos requisitos e casos de uso. Para isso criar-se-á um projeto e cadastrar-se-á todos os artefatos para a ferramenta Controla, que posteriormente serão rastreados.

5.3.1 Criando projeto WEBMET

Para o desenvolvimento de qualquer atividade no Controla, é necessário criar um projeto com o qual os requisitos vão estar atrelados. A figura 5.1 mostra o projeto criado e os seus atributos: dados do projeto, equipe que compõe o projeto e das expectativas dos usuários.

5.3.2 Cadastrando requisitos no projeto

Um requisito é cadastrado na ferramenta Controla através de um formulário, em que o usuário deve entrar com dados obrigatórios para o mesmo: nome, estado, estabilidade, prioridade, o tipo do requisito e o autor; e outros não obrigatórios: data de criação, data de início, data de finalização e descrição. A figura 5.2 mostra o requisito “fazer login no sistema” cadastrado no Controla. A figura 5.3 lista todos os requisitos do projeto WEBMET cadastrado na ferramenta.

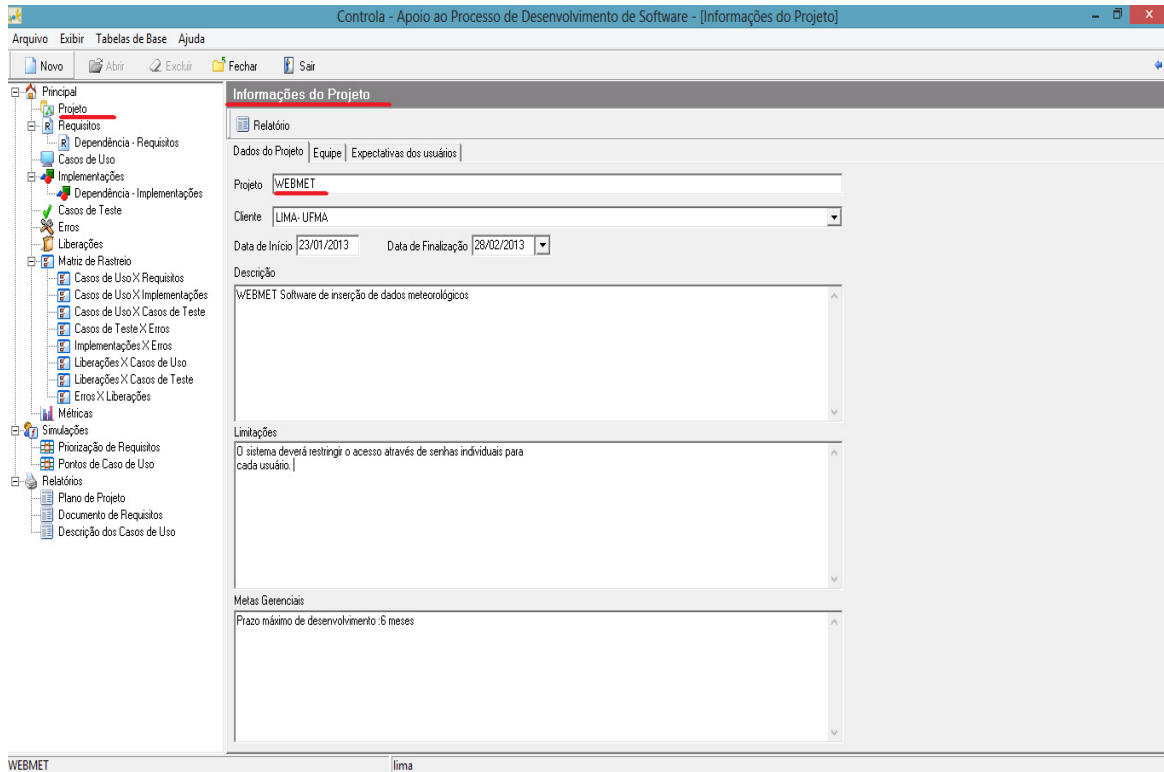


Figura 5.1 – Projeto WEBMET criado na ferramenta Controla.

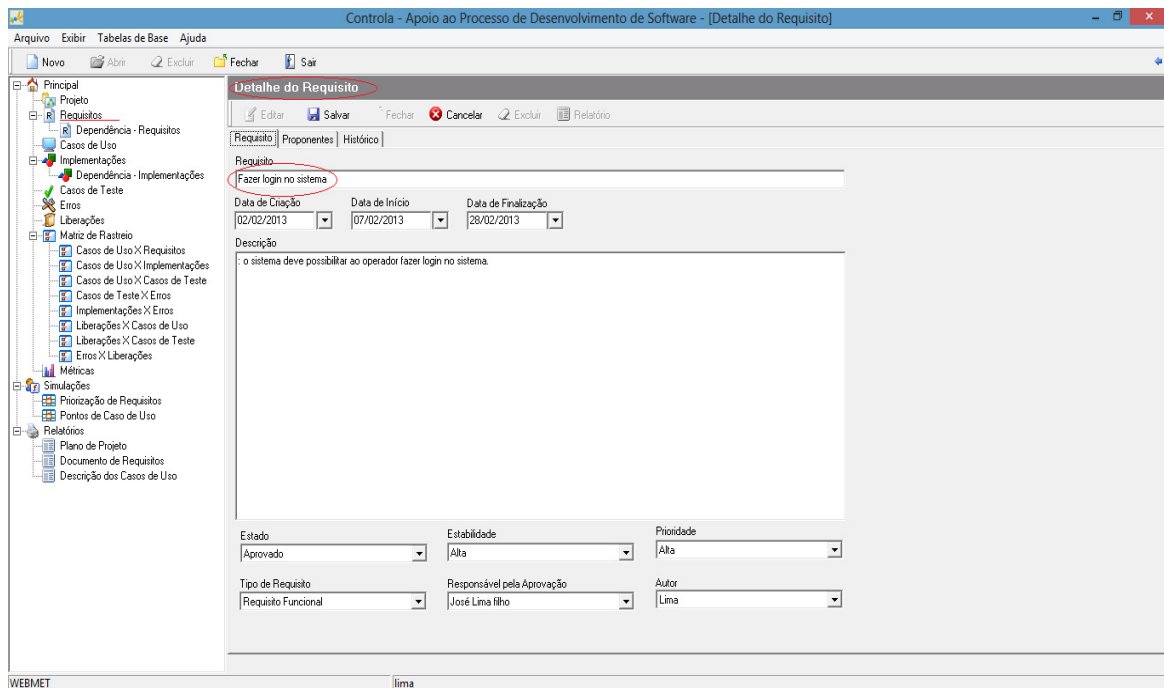


Figura 5.2 – Cadastro de um requisito na ferramenta Controla

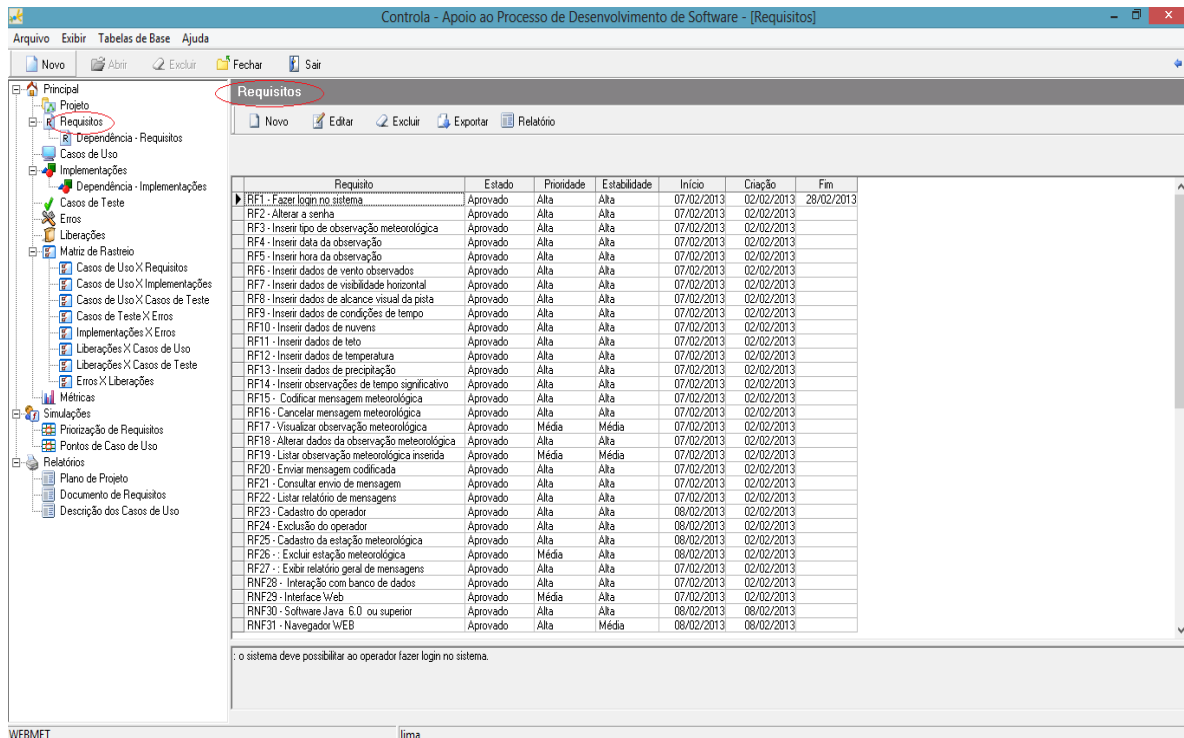


Figura 5.3 – Cadastro dos requisitos na ferramenta Controla

5.3.3 Cadastrando casos de uso no projeto

Um caso de uso é cadastrado na ferramenta Controla através de um formulário, em que o usuário deve entrar com dados obrigatórios para o mesmo: descrição, pré-condição, data, atores, responsável, tipo de caso de uso e estado e outros não obrigatórios: pós-condição, fluxos principais, fluxos alternativos e fluxos de erro. A figura 5.4 mostra o cadastro do caso de uso “fazer login” e a figura 5.5 mostra todos os casos de uso do projeto cadastrados no Controla.

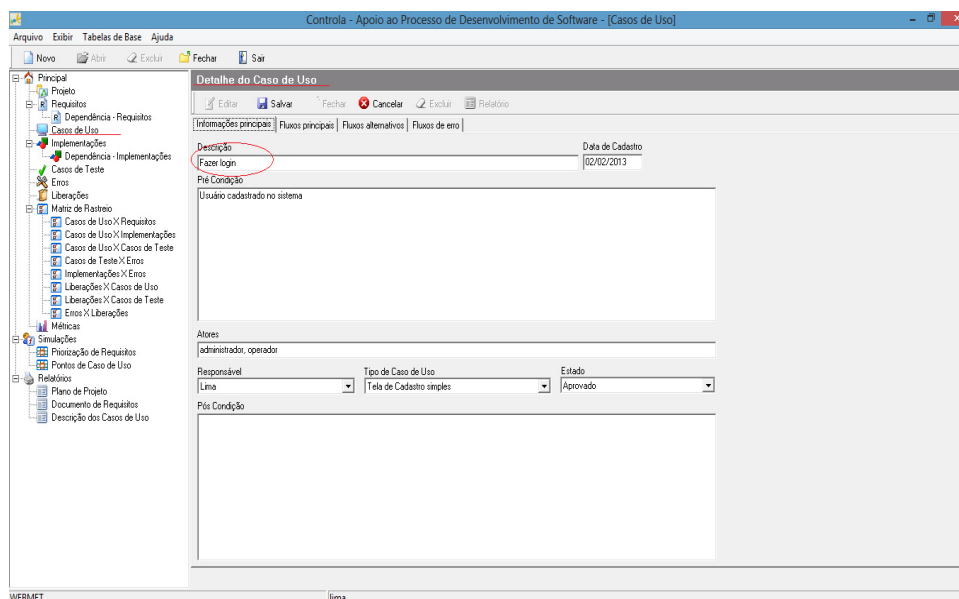


Figura 5.4 – Cadastro de um caso de uso na ferramenta Controla

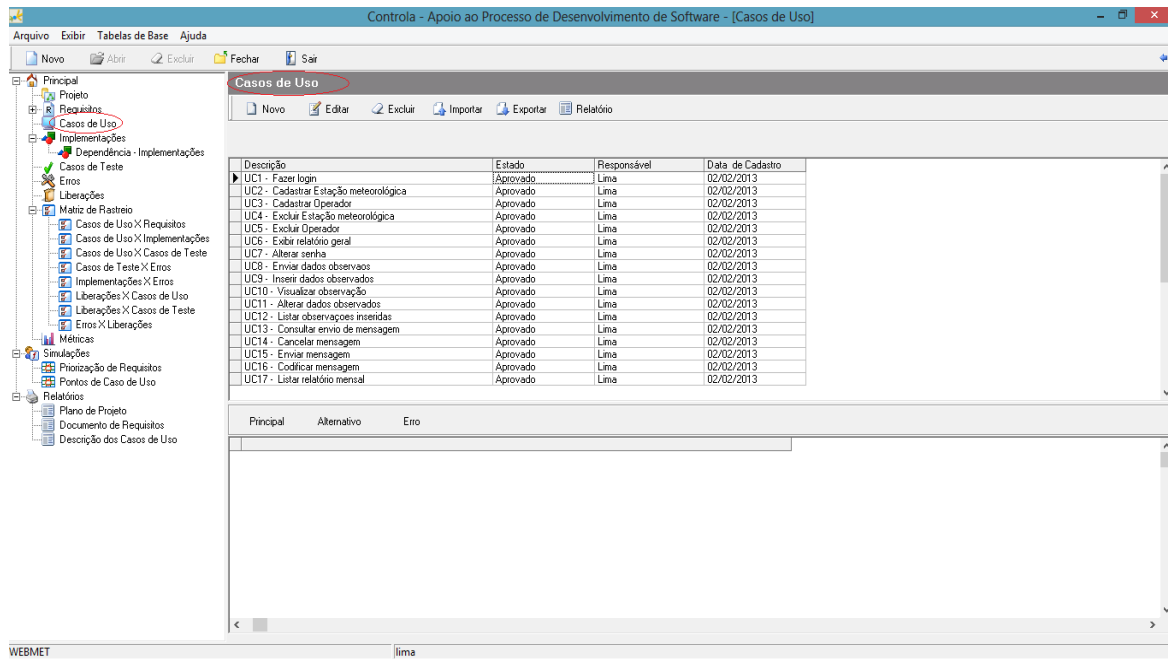


Figura 5.5 – Cadastro dos casos de uso na ferramenta Controla

5.3.4 Criando elos entre requisitos (rastreo)

Após o cadastro dos requisitos e casos de uso no Controla, é possível visualizar a matriz com os artefatos cadastrados e construir os elos entre os mesmo possibilitando a construção da matriz de rastreabilidade de requisitos. As figuras 5.6 e 5.7 mostra a matriz de dependência entre requisitos funcionais e não funcionais, primeiramente sem nenhum elo entre seus requisitos e após a criação dos elos.

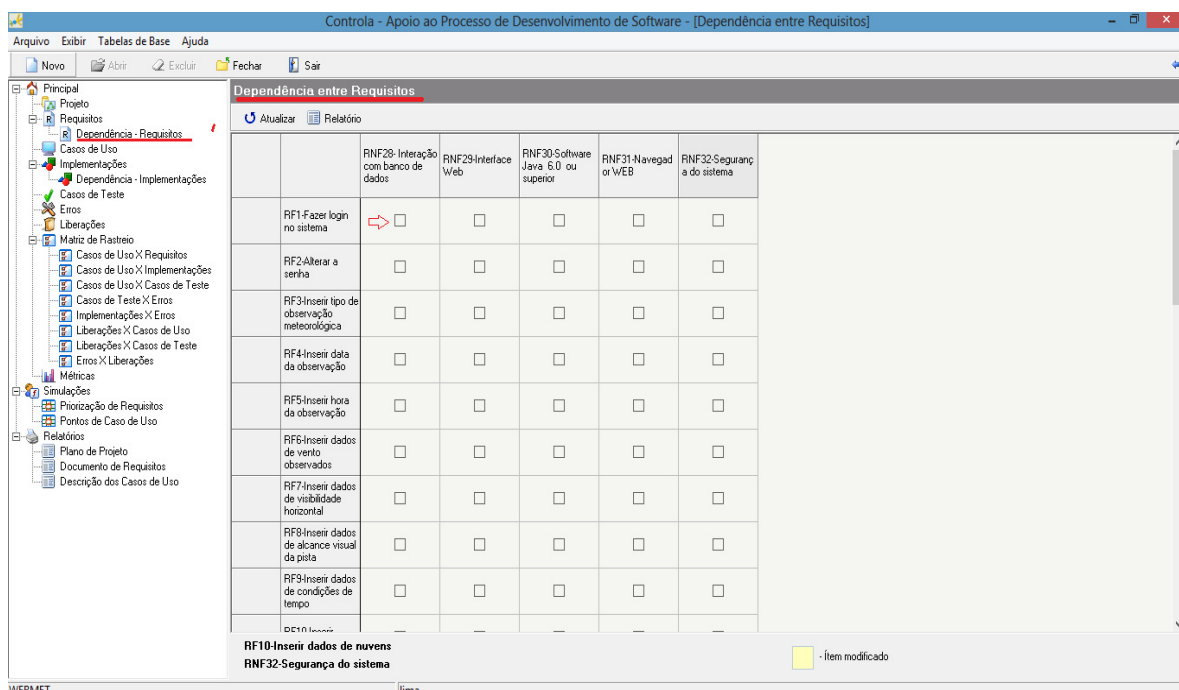


Figura 5.6 – Matriz de dependência entre requisitos, sem elos criados, na ferramenta Controla

Controla - Apoio ao Processo de Desenvolvimento de Software - [Dependência entre Requisitos]

Arquivo Exibir Tabelas de Base Ajuda

Novo Abrir Excluir Fechar Sair

Principal

- Projeto
- Requisitos
 - Dependência - Requisitos
 - Casos de Uso
 - Implementações
 - Dependência - Implementações
 - Casos de Teste
 - Erros
 - Liberações
 - Matriz de Rastreio
 - Casos de Uso X Requisitos
 - Casos de Uso X Implementações
 - Casos de Uso X Casos de Teste
 - Implementações X Erros
 - Liberações X Casos de Uso
 - Liberações X Casos de Teste
 - Erros X Liberações
- Métricas
- Simulações
 - Priorização de Requisitos
 - Pontos de Caso de Uso
- Relatórios
 - Plano de Projeto
 - Documento de Requisitos
 - Descrição dos Casos de Uso

Dependência entre Requisitos

Atualizar Relatório

| | RNF28- Interação com banco de dados | RNF29-Interfac e Web | RNF30-Softwar e Java 6.0 ou superior | RNF31-Navega dor WEB | RNF32-Seguran ça do sistema |
|---|---|---------------------------------------|--|---------------------------------------|--------------------------------|
| RF16-Cancelar mensagem meteorológica | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| RF17-Visualizar observação meteorológica | ⇒ <input checked="" type="checkbox"/> | ⇒ <input checked="" type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| RF18-Alterar dados da observação meteorológica | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| RF19-Listar observação meteorológica inserida | ⇒ <input checked="" type="checkbox"/> | ⇒ <input checked="" type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| RF20-Enviar mensagem codificada | ⇒ <input checked="" type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | ⇒ <input checked="" type="checkbox"/> | <input type="checkbox"/> |
| RF21-Consultar envio de mensagem | ⇒ <input checked="" type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| RF22-Listar relatório de mensagens | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| RF23-Cadastro do operador | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| RF24-Exclusão do operador | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| RF25-Cadastro | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |

RF19-Listar observação meteorológica inserida
RNF29-Interface Web

- Item modificado

WEBMET lima

Figura 5.7 – Matriz de dependência entre requisitos, com elos criados, na ferramenta Controla

Similarmente o que foi feito entre requisitos, pode-se fazer com outros artefatos, buscando assim registrar elos de dependências entre os mesmo.

5.3.5 Matriz de rastreabilidade do projeto WEBMET com a ferramenta Controla

Conforme citado anteriormente a ferramenta Controla tem suporte a rastreabilidade de requisitos, através de matriz de rastreio que oferece como grande facilitador a visualização dos casos de uso e requisitos do sistema em uma tabela de forma gráfica, dando suporte ao analista para tomar decisões e descobrir problemas e sua solução de forma mais rápida, muitas vezes antes da fase de implementação. Os artefatos modificados, ou seja, qualquer alteração no estado dos requisitos são sinalizados, na cor amarelo, oferecendo suporte visual ao analista na identificação de potenciais focos de problemas. As figuras 5.8 mostra os requisitos cadastrados na ferramenta Controla no estado inicial “aprovado”, ou seja, sem qualquer modificação nos mesmos. Em seguida a figura 5.9 mostra a matriz de rastreabilidade entre casos de uso e requisitos, com alguns elos registrados entre esses artefatos, porém sem qualquer alteração no estado inicialmente cadastrados.

| Requisito | Estado | Prioridade | Estabilidade | Início | Criação | Fim |
|---|----------|------------|--------------|------------|------------|-----|
| RF10 - Inserir dados de nuvens | Aprovado | Alta | Alta | 07/02/2013 | 02/02/2013 | |
| RF11 - Inserir dados de teto | Aprovado | Alta | Alta | 07/02/2013 | 02/02/2013 | |
| RF12 - Inserir dados de temperatura | Aprovado | Alta | Alta | 07/02/2013 | 02/02/2013 | |
| RF13 - Inserir dados de precipitação | Aprovado | Alta | Alta | 07/02/2013 | 02/02/2013 | |
| RF14 - Inserir observações de tempo significativo | Aprovado | Alta | Alta | 07/02/2013 | 02/02/2013 | |
| RF15 - Codificar mensagem meteorológica | Aprovado | Alta | Alta | 07/02/2013 | 02/02/2013 | |
| RF16 - Cancelar mensagem meteorológica | Aprovado | Alta | Alta | 07/02/2013 | 02/02/2013 | |
| RF17 - Visualizar observação meteorológica | Aprovado | Média | Média | 07/02/2013 | 02/02/2013 | |
| RF18 - Alterar dados da observação meteorológica | Aprovado | Alta | Alta | 07/02/2013 | 02/02/2013 | |
| RF19 - Listar observação meteorológica inserida | Aprovado | Média | Média | 07/02/2013 | 02/02/2013 | |
| RF20 - Enviar mensagem codificada | Aprovado | Alta | Alta | 07/02/2013 | 02/02/2013 | |
| RF21 - Consultar envio de mensagem | Aprovado | Alta | Alta | 07/02/2013 | 02/02/2013 | |
| RF22 - Listar relatório de mensagens | Aprovado | Alta | Alta | 07/02/2013 | 02/02/2013 | |
| RF23 - Cadastro do operador | Aprovado | Alta | Alta | 05/02/2013 | 02/02/2013 | |
| RF24 - Exclusão do operador | Aprovado | Alta | Alta | 05/02/2013 | 02/02/2013 | |
| RF25 - Cadastro da estação meteorológica | Aprovado | Alta | Alta | 07/02/2013 | 02/02/2013 | |
| RF26 - Excluir estação meteorológica | Aprovado | Média | Alta | 07/02/2013 | 02/02/2013 | |
| RF27 - Exibir relatório geral de mensagens | Aprovado | Alta | Alta | 07/02/2013 | 02/02/2013 | |
| RF28 - Interação com banco de dados | Aprovado | Alta | Alta | 07/02/2013 | 02/02/2013 | |

o sistema deve interagir com o banco de dados MySQL

Figura 5.8 – Requisitos cadastrados com estado aprovado, na ferramenta Controla

| | RF18-Alterar dados da observação meteorológica | RF19-Listar observação meteorológica inserida | RF20-Enviar mensagem codificada | RF21-Consultar envio de mensagem | RF22-Listar relatório de mensagens | RF23-Cadastro do operador | RF24-Exclusão do operador | RF25-Cadastro da estação meteorológica | RF26: Excluir estação meteorológica | RF27: Exibir relatório geral de mensagens |
|-------------------------------------|--|---|---------------------------------|----------------------------------|------------------------------------|-------------------------------------|-------------------------------------|--|-------------------------------------|---|
| UC1-Fazer login | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| UC2-Cadastrar Estação meteorológica | <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"/> | <input type="checkbox"/> |
| UC3-Cadastrar Operador | <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"/> |
| UC4-Excluir Estação meteorológica | <input 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"/> |
| UC5-Excluir Operador | <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"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| UC6-Exibir relatório geral | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| UC7-Alterar senha | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| UC8-Enviar dados observação | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| UC9-Inserir dados observados | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |

UC4-Excluir Estação meteorológica
RF26: Excluir estação meteorológica

• Item modificado

Figura 5.9 – Matriz de rastreabilidade entre casos de uso e requisitos, com elos criados, na ferramenta Controla

A ferramenta Controla possibilita fazer mudanças nos requisitos cadastrados e visualizar tais mudanças. Quando ocorre uma mudança no estado dos requisitos, ela é registrada no log de mudanças do requisito e pode ser visualizada acessando os menus: requisitos/editar/histórico. Esta funcionalidade permite que o gerente de requisitos acompanhe todas as mudanças ocorridas em um requisito e use tais informações para apoiar a tarefa de gerenciar a tomada de decisão a respeito de novas mudanças nesses requisitos. A figura 5.10 mostra esse histórico na ferramenta Controla.

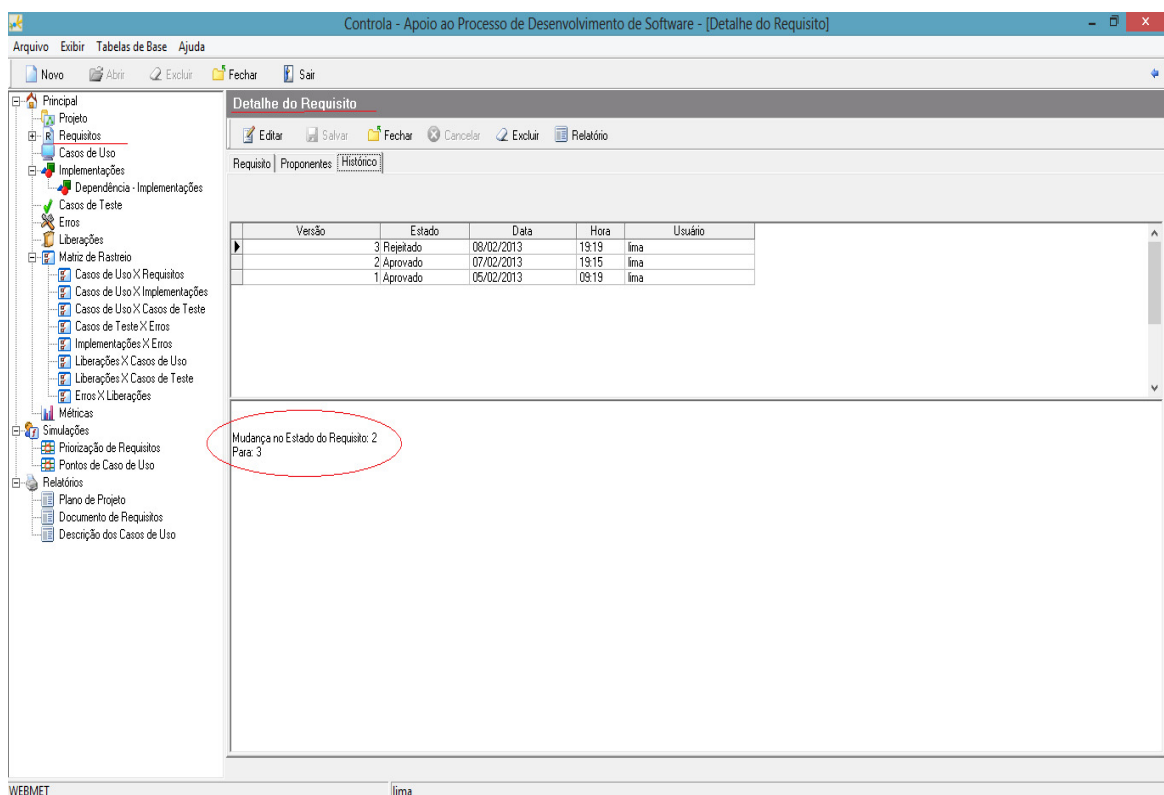


Figura 5.10 – Histórico de mudança no estado do requisito, na ferramenta Controla

Com a finalidade de verificar a funcionalidade da ferramenta Controla, fez-se mudanças de estado em alguns requisitos e atualizou-se o rastreo verificando assim que os artefatos modificados foram sinalizados, oferecendo suporte visual ao analista na identificação de potenciais focos de problemas, conforme mostra as figuras 5.11 e 5.12, respectivamente.

Controla - Apoio ao Processo de Desenvolvimento de Software - [Requisitos]

Arquivo Exibir Tabelas de Base Ajuda

Novo Abrir Excluir Fechar Sair

Requisitos

Novo Editar Excluir Exportar Relatório

| Requirito | Estado | Prioridade | Estabilidade | Início | Criação | Fm |
|---|--------------|------------|--------------|------------|------------|----|
| RF14 - Inserir observações de tempo significativo | Aprovado | Alta | Alta | 07/02/2013 | 02/02/2013 | |
| RF15 - Codificar mensagem meteorológica | Aprovado | Alta | Alta | 07/02/2013 | 02/02/2013 | |
| RF16 - Cancelar mensagem meteorológica | Aprovado | Alta | Alta | 07/02/2013 | 02/02/2013 | |
| RF17 - Visualizar observação meteorológica | Aprovado | Média | Média | 07/02/2013 | 02/02/2013 | |
| RF18 - Alterar dados da observação meteorológica | Aprovado | Alta | Alta | 07/02/2013 | 02/02/2013 | |
| RF19 - Listar observação meteorológica inserida | Aprovado | Média | Média | 07/02/2013 | 02/02/2013 | |
| RF20 - Enviar mensagem codificada | Aprovado | Alta | Alta | 07/02/2013 | 02/02/2013 | |
| RF21 - Consultar envio de mensagem | Aprovado | Alta | Alta | 07/02/2013 | 02/02/2013 | |
| RF22 - Listar relatório de mensagens | Aprovado | Alta | Alta | 07/02/2013 | 02/02/2013 | |
| RF23 - Cadastro do operador | Rejeitado | Alta | Alta | 05/02/2013 | 02/02/2013 | |
| RF24 - Exclusão do operador | Implementado | Alta | Alta | 05/02/2013 | 02/02/2013 | |
| RF25 - Cadastro da estação meteorológica | Alta | Alta | Alta | 07/02/2013 | 02/02/2013 | |
| RF26 - Excluir estação meteorológica | Em avaliação | Média | Média | 07/02/2013 | 02/02/2013 | |
| RF27 - Exibir relatório geral de mensagens | Aprovado | Alta | Alta | 07/02/2013 | 02/02/2013 | |
| RNF28 - Interação com banco de dados | Aprovado | Alta | Alta | 07/02/2013 | 02/02/2013 | |
| RNF29 - Interface Web | Aprovado | Média | Alta | 07/02/2013 | 02/02/2013 | |
| RNF30 - Software Java 6.0 ou superior | Aprovado | Alta | Alta | 08/02/2013 | 08/02/2013 | |
| RNF31 - Navegador WEB | Aprovado | Alta | Média | 08/02/2013 | 08/02/2013 | |
| RNF32 - Segurança do sistema | Aprovado | Alta | Alta | 08/02/2013 | 08/02/2013 | |

o sistema deve possuir interface Web

WEBMET lima

Figura 5.11 – Requisitos com mudança no estado, na ferramenta Controla

Controla - Apoio ao Processo de Desenvolvimento de Software - [Casos de Uso - Requisitos]

Arquivo Exibir Tabelas de Base Ajuda

Novo Abrir Excluir Fechar Sair

Rastreabilidade Casos de Uso -> Requisitos

Atualizar Relatório

| | RF19-Listar observação meteorológica inserida | RF20-Enviar mensagem codificada | RF21-Consultar envio de mensagem | RF22-Listar relatório de mensagens | RF23-Cadastro do operador | RF24-Exclusão do operador | RF25-Cadastro da estação meteorológica | RF26- Excluir estação meteorológica | RF27- Exibir relatório geral de mensagens |
|-------------------------------------|---|---------------------------------|----------------------------------|------------------------------------|-------------------------------------|-------------------------------------|--|-------------------------------------|---|
| UC1-Fazer login | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| UC2-Cadastrar Estação meteorológica | <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"/> |
| UC3-Cadastrar Operador | <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"/> |
| UC4-Excluir Estação meteorológica | <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"/> | <input type="checkbox"/> |
| UC5-Excluir Operador | <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"/> |
| UC6-Exibir relatório geral | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| UC7-Alterar senha | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| UC8-Enviar dados observaos | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| UC9-Inserir dados observados | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |

UC10-Visualizar observação
RF27- Exibir relatório geral de mensagens

Item modificado

WEBMET lima

Figura 5.12 - Matriz de rastreabilidade entre casos de uso e requisitos, com rastreo de itens modificados.

5.3.6 Diagramas de casos de uso do sistema

A Figura 5.13 demonstra de uma maneira simplificada os casos de uso macro do sistema WEBMET. Através desse diagrama é possível analisar as interações entre administrador, operador e o sistema.

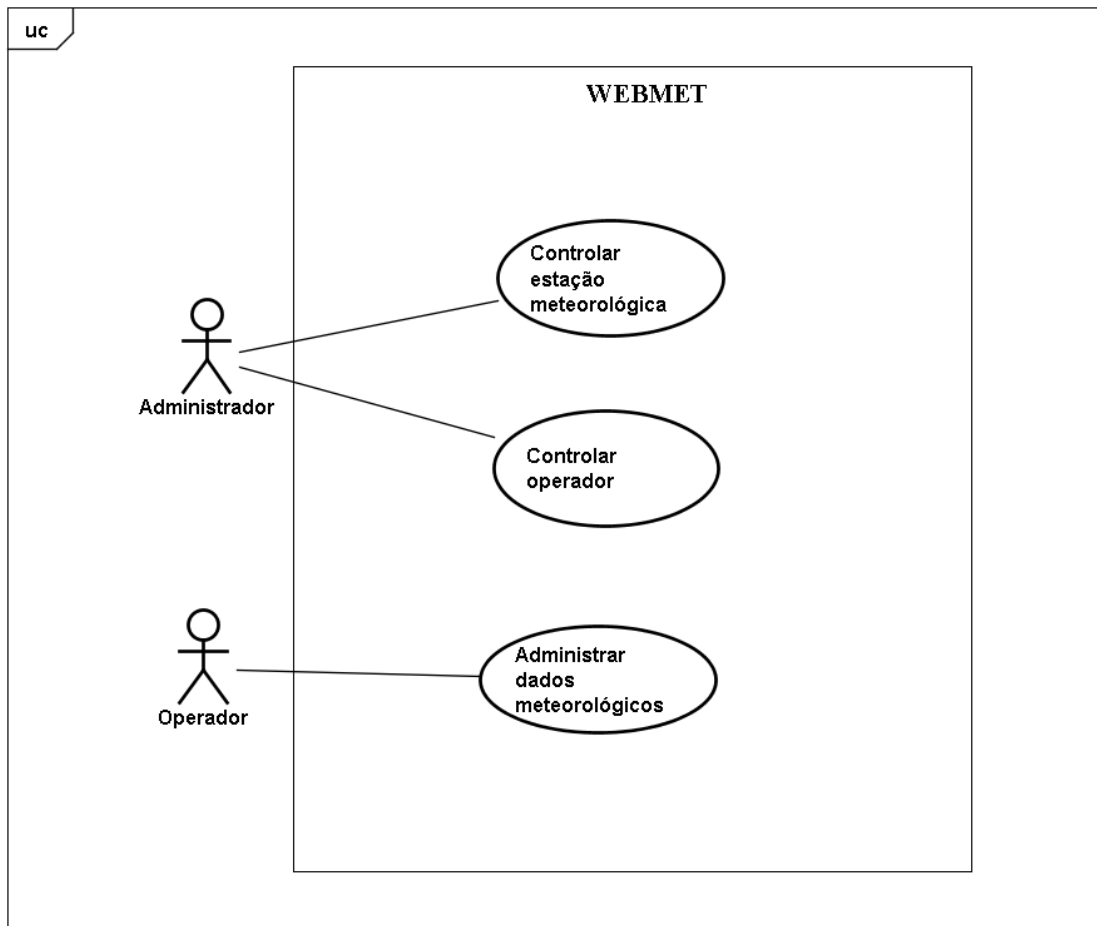


Figura 5.13 – Diagrama de caso de uso macro do sistema WEBMET

Através da Figura 5.14 pode-se verificar de forma simplificada as principais interações entre o operador e o sistema WEBMET. O operador administra todos os dados meteorológicos.

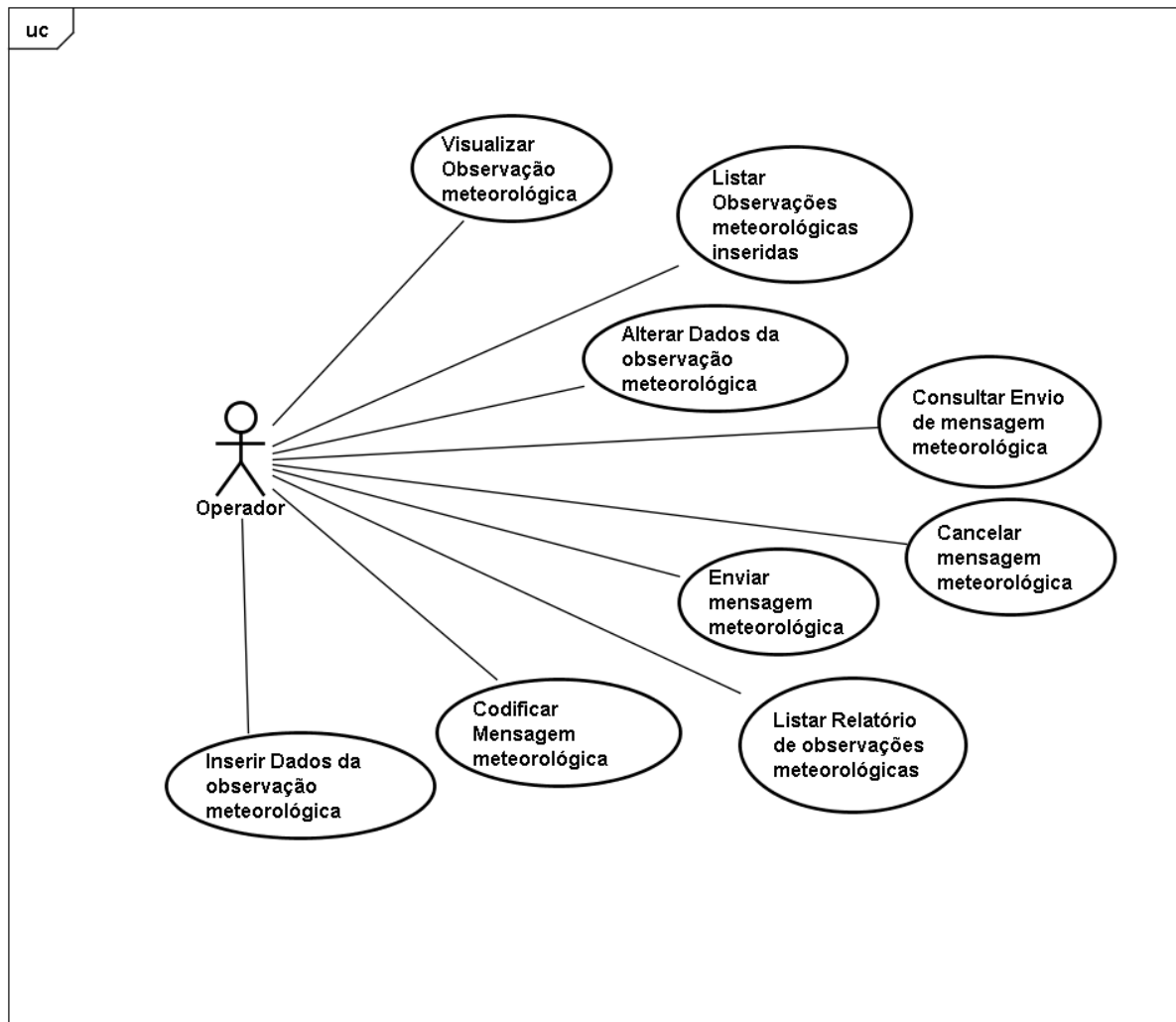


Figura 5.14 – Diagrama de caso de uso

5.4 Implementação do sistema WEBMET

Os dois principais usuários do software são: Administrador e Operador.

O Administrador tem acesso a todas as funcionalidades do sistema sendo o responsável pelo cadastramento das máquinas das estações meteorológicas de superfície e dos operadores destas.

O Operador é o profissional que realiza a atividade de observação meteorológica das estações meteorológicas do Comando da Aeronáutica, sendo o responsável por inserir e alterar os dados das observações no sistema WEBMET e por enviar as respectivas informações ao servidor WEBMET e ao banco OPMET.

Ao acessar o sistema através dos endereços eletrônicos já citados a tela de Login é apresentada. No primeiro acesso deve ser inserido um Login padrão com Login (nome de guerra do Operador/militar) e senha padrão (DCCO) que devem ser alterados na sequência conforme as figuras 5.15 e 5.16 abaixo:



Figura 5.15 – Tela de Login do sistema WEBMET

A imagem mostra a interface de troca de senha do sistema WEBMET. O título "Troca de Senha" está em um cabeçalho cinza. Abaixo, há três campos de entrada de texto rotulados "Senha Atual:", "Nova Senha:" e "Confirma Nova Senha:". Na base, há dois botões: "Gravar" e "Cancelar".

Figura 5.16 – Tela troca de senha do sistema WEBMET

A tela principal do WEBMET acessada pelo Operador/Usuário é mostrada na sequência com todas as funcionalidades que lhe são acessíveis ao clicar em OBS Meteorológica, figuras 5.17 e 5.18.

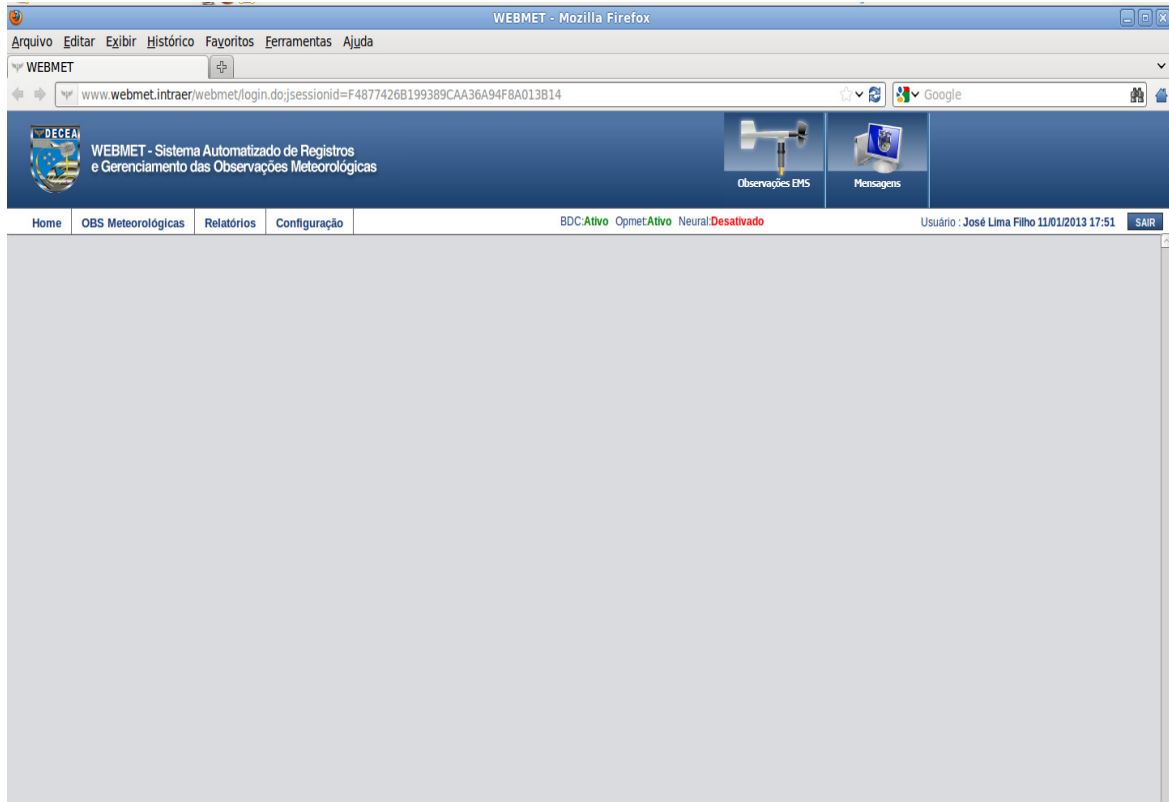


Figura 5.17 – Tela inicial após Login do sistema WEBMET

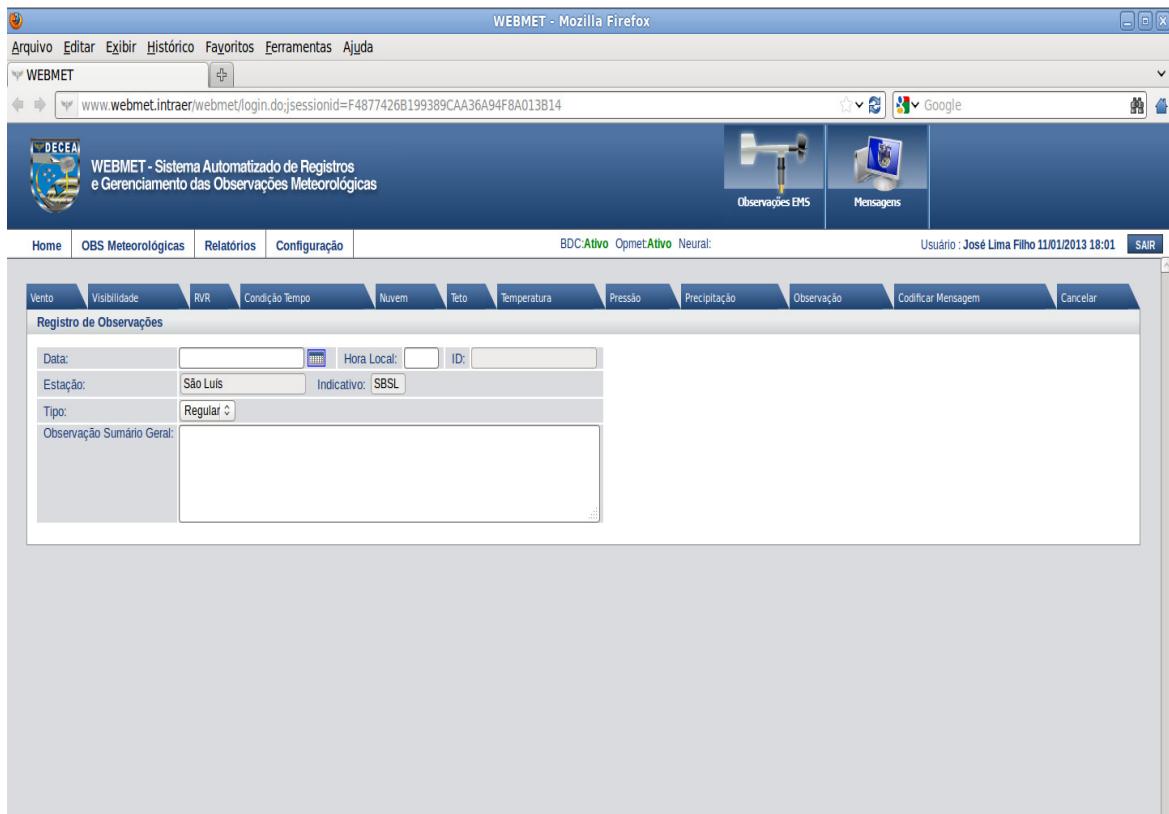


Figura 5.18 – Tela de observação meteorológicas do sistema WEBMET

6. CONCLUSÃO

Desenvolver de software de qualidade, dentro do prazo e orçamento é o grande estímulo para o desenvolvimento de software. A engenharia de requisitos busca entender os requisitos e como se relacionam com outros requisitos e itens de projeto, principalmente durante a mudança e o teste de requisitos. Isto é feito pela gerência de requisitos através da rastreabilidade de requisitos e uso da matriz de rastreabilidade como uma importante ferramenta para controle dos requisitos, seus relacionamentos e suas mudanças durante as diversas atividades do projeto, identificando facilmente os itens relacionados e quais impactos causados no projeto por alguma mudança no requisito.

Esse estudo apresentou uma revisão dos conceitos de engenharia de requisitos, deu enfoque na técnica de rastreabilidade de requisitos, seus tipos e dificuldades de uso no processo de desenvolvimento de software. Fez-se uma explanação das principais ferramentas automatizadas que auxiliam à gerência de requisitos, principalmente focando a rastreabilidade de requisitos. Além disso foi mostrado um estudo de caso aplicando a rastreabilidade de requisitos no sistema WEBMET através da ferramenta Controla, que é um software simples de grande utilidade para gerência de requisitos, pois atende ao que foi proposto. Nesse estudo de caso, notou-se que apesar de ser uma forma simples, fácil de representar os elos (*links*) entre os requisitos, pois usa somente uma ligação representando o relacionamento e que ainda existem grandes dificuldades em relação a manutenção dos elos de relacionamentos, assim como o investimento associado a seu uso seja ainda elevado, é fato que a rastreabilidade de requisitos é de grande importância, pois reduz custos e melhora a qualidade do software.

Como a automação das matrizes de rastreabilidade têm sido otimizadas, é preciso uma maior investimento por parte das empresas desenvolvedoras de software buscando treinar e aperfeiçoar suas equipes para utilização plena dessa atividade de forma a criar uma cultura dentro da empresa a fim de alcançar seus objetivos.

Uma possibilidade de trabalho futuro seria uma aplicação do modelo de rastreabilidade de requisitos proposto por Toranzo, além de estudar ferramentas que ofereçam suporte ao processo de rastreabilidade de requisitos com a finalidade de verificar a adequação destas ao modelo proposto por Toranzo.

7. BIBLIOGRAFIA

ARKLEY.P, MASON.P and RIDDLE.S, “Position Paper: Enabling Traceability,” Proceedings of the 1st International Workshop on Traceability in Emerging Forms of Software Engineering, Edinburgh, Scotland (Setembro 2002), pp. 61–65.

BACKES, Jerônimo. Rastreabilidade Semiautomática Através do Mapeamento de Entidades. 2008. 96 f. Dissertação – Universidade Federal do Rio Grande do Sul, Porto Alegre.

Disponível:

<http://www.lume.ufrgs.br/bitstream/handle/10183/15308/000677241.pdf?sequence=1>, (30 de novembro de 2012).

DEVMEDIA (2012) Revista Engenharia de Software Magazine, 2012. Disponível:

<http://www.devmedia.com.br/revista-engenharia-de-software-magazine>.

DEVMEDIA (2012). Controla – Ferramenta de Apoio ao Processo de Desenvolvimento de Software em Pequenas Empresas. Disponível:

<http://www.devmedia.com.br/articles/viewcomp.asp?comp=283>, (28 de dezembro de 2012).

GOTEL, O. C. Z. e FINKELSTEIN, A. C. W. (1994). An Analysis of the Requirements Traceability Problem. In: Proceedings of the First International Conference on Requirements Engineering, Colorado Springs, Co, 1994. Disponível: eprints.ucl.ac.uk/749/1/2.2_rtrprob.pdf, (04 de dezembro de 2012).

IBM (2012). Rational RequisitePro.

Disponível: <http://www-142.ibm.com/software/products/br/pt/reqpro/>, (23 dezembro de 2012).

PAULA FILHO, Wilson de Pádua. Engenharia de Software: Fundamentos, Métodos e Padrões. Rio de Janeiro: Mcgraw-hill Interamericana, 2009.

PRESSMAN, Roger S.. Engenharia de Software. 6.ed. São Paulo: McGraw-Hill, 2006.

RAMESH, Bala e JARKE, Matthias. (1999). Towards Reference Models for Requirements Traceability. IEEE Transactions on Software Engineering. Disponível: <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.40.6602&rep=rep1&type=pdf>, (17 de dezembro 2012).

SAYÃO, Miriam e LEITE, J. C. S. P. Rastreabilidade de Requisitos. 2005. 22 f. Monografia (Departamento de Informática) – PUC-RIO, Rio de Janeiro. Disponível: ftp://ftp.inf.puc-rio.br/pub/docs/techreports/05_20_sayao.pdf, (19 de dezembro de 2012).

SOMMERVILLE, Ian. Engenharia de Software.9.ed. São Paulo: Pearson, 2011.

SPARX SYSTEMS (2012). Enterprise Architect Model Driven UML Tools. Disponível: <http://www.sparxsystems.com.au/products/ea/10/index.html>, (20 de dezembro 2012).

SWEBOK (2004). Guide to the Software Engineering Body of Knowledge 2004 Version. (On-line). Disponível: <http://www.computer.org/portal/web/swebok/htmlformat>, (25 de novembro de 2012).

VILELA, D.(2008) Rastreabilidade de Requisitos INF2921.Disponível:<http://www.slideshare.net/transparenciadesoftware/rastreabilidade-de-requisitos>, (27 de novembro de 2012).

WEBMET Sistema Automatizado de Registro e Gerenciamento das Observações Meteorológicas.
<http://www.webmet.aer.mil.br/webmet/login.do>