

UNIVERSIDADE DE CAXIAS DO SUL
Centro de Ciências Exatas e Tecnologia
Curso de Bacharelado em Sistemas de Informação

Rodrigo Debastiani Martins

**PROPOSTA DE UM PROCESSO DE ENGENHARIA DE REQUISITOS
PARA O NUSIS**

Caxias do Sul

2013

Rodrigo Debastiani Martins

**PROPOSTA DE UM PROCESSO DE ENGENHARIA DE REQUISITOS
PARA O NUSIS**

Trabalho de Conclusão de Curso para
obtenção do grau de Bacharel em
Sistemas de Informação da Universidade
de Caxias do Sul

Iraci Silveira da Cristina De Carli
Orientador

Caxias do Sul

2013

RESUMO

A engenharia de requisitos tem sido reconhecida como um dos maiores desafios do processo de desenvolvimento de *software*. Os requisitos compreendem a essência de um sistema de *software*, e a sua determinação de forma assertiva está diretamente relacionada à qualidade final do produto. São eles os responsáveis por definir as funcionalidades que o sistema deve fornecer e sob quais condições o sistema deve operar.

Na Universidade de Caxias do Sul, o NUSIS – Núcleo de Sistemas de Informação conta com um projeto que visa o desenvolvimento de sistemas de informação para uso acadêmico e empresarial. Constituído a partir de uma equipe de trabalho formada pelos próprios acadêmicos e docentes da universidade, tal projeto objetiva integrar as diferentes áreas de negócio de uma organização em um único produto, através da construção de um ERP - Sistema Integrado de Gestão Empresarial.

O presente trabalho se propõe a estabelecer um modelo de processo para as atividades da engenharia de requisitos do ambiente de desenvolvimento de *software* do NUSIS, e pretende definir as atividades que devem ser desenvolvidas, as técnicas que devem ser utilizadas, os artefatos que devem ser produzidos e a ferramenta *CASE* que deve ser empregada nas atividades da engenharia de requisitos, visando garantir a qualidade do produto resultante das mesmas.

Palavras-chave: Engenharia de Software, Engenharia de Requisitos, Processo Unificado

ABSTRACT

The requirement engineering has been recognized as one of the biggest development software process challenges. The requirements comprise the essence of a software system, and its determination assertively is directly related to the quality of the final product. They are responsible for defining the functionality that the system should provide and under what conditions the system should operate.

At the University of Caxias do Sul, the NUSIS - Information Systems Center has a project that aims to develop information systems for business and academic use. Composed from a work team formed by the academics and university professor, such project aims to integrate the different business areas of an organization into a single product, by the development of an ERP - Enterprise Resource Planning.

This work proposes to define a process model for the requirement engineering activities of the development environment software existing in NUSIS, and want to set the activities to be developed, the techniques to be used, the artifacts to be produced and the tool *CASE* that should be employed in the requirement engineering activities in order to ensure the product quality resulting.

Key-words: Software Engineering, Requirements Engineering, Unified Process

LISTA DE FIGURAS

| | |
|--|-----|
| Figura 1 - Fases do Processo Unificado | 40 |
| Figura 2 - <i>Workflow</i> de Atividades da Disciplina de Requisitos | 42 |
| Figura 3 - Representação do subfluxo Analisar o Problema | 43 |
| Figura 4 - Representação do subfluxo Compreender as Necessidades dos Envolvidos..... | 44 |
| Figura 5 - Representação do subfluxo Definir o Sistema..... | 45 |
| Figura 6 - Representação do subfluxo Gerenciar o Escopo do Sistema..... | 46 |
| Figura 7 - Representação do subfluxo Refinar a Definição do Sistema..... | 47 |
| Figura 8 - Representação do subfluxo Gerenciar Requisitos Mutáveis | 48 |
| Figura 9 - Exemplo de relação entre Papel, Atividade e Artefato..... | 50 |
| Figura 10 - Papéis e Artefatos construídos na Disciplina de Requisitos | 51 |
| Figura 11 - Caso de Uso resumido: “Processar Venda”..... | 53 |
| Figura 12 - Caso de Uso informal: “Tratar Devoluções” | 54 |
| Figura 13 - Modelo para casos de uso completos de uma coluna | 55 |
| Figura 14 - Caso de Uso “inteiramente” completo de uma coluna | 55 |
| Figura 15 - <i>Stakeholders</i> e Interesses para o Caso de Uso: “Processar Venda”..... | 58 |
| Figura 16 – Exemplos da seção Pré-Condição para diferentes Casos de Uso..... | 58 |
| Figura 17 – Exemplos da seção Garantias Mínimas para diferentes Casos de Uso | 59 |
| Figura 18 – Exemplo da seção Garantia de Sucesso para diferentes Casos de Uso..... | 59 |
| Figura 19 - Acionador para o Caso de Uso “Processar Venda” | 60 |
| Figura 20 - Cenário de Sucesso Principal para o Caso de Uso "Registrar Perda" | 61 |
| Figura 21 - Cenário de Sucesso Principal e Extensões para o Caso de Uso “Verificar Ortografia” | 62 |
| Figura 22 - Lista de Variantes Tecnológicas e de Dados para o Caso de Uso “Reembolsar Cliente” .. | 63 |
| Figura 23 - Narrativa de Uso: “Pegando Dinheiro Rápido” | 63 |
| Figura 24 – Diagrama de Casos de Uso: “Videolocadora” | 64 |
| Figura 25 - Processo de Engenharia de Requisitos Proposto | 71 |
| Figura 26 – Exemplo de Modelo de Domínio..... | 77 |
| Figura 27 - Modelo proposto para as Narrativas de Uso..... | 78 |
| Figura 28 - Modelo proposto para os Casos de Uso..... | 81 |
| Figura 29 - Seções para validação dos requisitos no modelo proposto para os Casos de Uso..... | 83 |
| Figura 30 - Exemplo de Rastreabilidade no entre os artefatos e seus requisitos..... | 84 |
| Figura 31 - Processo de Avaliação (ISO/IEC 14598-1)..... | 88 |
| Figura 32 - Tela principal da ferramenta OSRMT com requisitos da categoria “ <i>Requirement</i> ” | 94 |
| Figura 33- Interface para preenchimento dos atributos de um requisito da categoria " <i>Requirement</i> " .. | 95 |
| Figura 34 - Seções para especificação do caso de uso na ferramenta OSRMT..... | 96 |
| Figura 35 - Matriz de Rastreabilidade entre requisitos na ferramenta OSRMT..... | 97 |
| Figura 36 - Aba de suplementos da ferramenta RequisitePro no Microsoft Word | 98 |
| Figura 37 - Interface para criação de requisito na ferramenta RequisitePro | 99 |
| Figura 38 - Interface principal da ferramenta RequisitePro | 99 |
| Figura 39 - Matriz de rastreabilidade na ferramenta RequisitePro..... | 100 |
| Figura 40 - Trecho da seção de Necessidades dos Stakeholders extraída do documento de Visão | 106 |
| Figura 41 - Requisitos com as Necessidades dos Stakeholders visualizados através do RequisitePro | 107 |
| Figura 42 - Árvore de diretórios do projeto no RequisitePro..... | 108 |
| Figura 43 - Diagrama de casos de uso do processo de Requisição de Compra..... | 110 |

| | |
|--|-----|
| Figura 44 - Modelo de domínio do processo de Requisição de Compra..... | 111 |
| Figura 45 - Narrativa de uso "Requisitar notebook para funcionário" | 112 |
| Figura 46 - Requisitos funcionais do processo de Requisição de Compra..... | 113 |
| Figura 47 - Requisitos do sistema e atributos da atividade de negociação de requisitos | 115 |
| Figura 48 - Seção "Fluxo de Eventos" do caso de uso "Efetuar Requisição de Compra" | 116 |
| Figura 49 - Protótipo de tela do caso de uso "Efetuar Requisição de Compra" | 117 |
| Figura 50 - Estrutura de diretórios dos casos de uso do processo de Requisição de Compra | 118 |
| Figura 51 - Seções do caso de uso "Efetuar Requisição de Compra" visualizadas no RequisitePro .. | 118 |
| Figura 52 - Requisitos não funcionais extraídos das especificações suplementares | 119 |
| Figura 53 - Casos de uso do processo de Requisição de Compra e seus atributos de validação..... | 121 |
| Figura 54 - Matriz de rastreabilidade entre Necessidades dos <i>Stakeholders</i> e Requisitos do Sistema | 122 |
| Figura 55 - Matriz de rastreabilidade entre Requisitos do Sistema e Narrativas de Uso | 122 |
| Figura 56 - Matriz de rastreabilidade entre Requisitos do Sistema e Casos de Uso | 123 |
| Figura 57 - Matriz de rastreabilidade entre Narrativas de Uso e Casos de Uso | 124 |
| Figura 58 - Matriz de rastreabilidade entre Casos de Uso | 124 |
| Figura 59 - Formas de rastreabilidade possíveis na gerência de requisitos..... | 125 |

LISTA DE TABELAS

| | |
|---|-----|
| Tabela 1 - Características e Subcaracterísticas da ISO/IEC 9126..... | 24 |
| Tabela 2 - Técnicas para Validação de Requisitos..... | 36 |
| Tabela 3 - Atributos e Características de Qualidade desejadas para o NUSIS..... | 70 |
| Tabela 4 - Papéis do Processo de Engenharia de Requisitos Proposto | 71 |
| Tabela 5 - Estrutura do Documento Visão proposto | 74 |
| Tabela 6 - Modelo e Exemplificação da Lista de Requisitos | 78 |
| Tabela 7 - Ferramentas gratuitas avaliadas (Alves, 2007) | 90 |
| Tabela 8 - Sistema de Métricas | 90 |
| Tabela 9 - Resultado da avaliação das ferramentas CASE gratuitas | 91 |
| Tabela 10 - Requisitos e pesos para avaliação das ferramentas OSRMT e RequisitePro | 93 |
| Tabela 11 - Resultados obtidos a partir da avaliação das ferramentas OSRMT e RequisitePro | 101 |
| Tabela 12 - Questionamentos elaborados para condução de entrevista | 109 |
| Tabela 13 – Considerações sobre os atributos e características de qualidade..... | 127 |

SUMÁRIO

| | | |
|----------|--|-----------|
| 1 | INTRODUÇÃO | 10 |
| 1.1 | PROBLEMÁTICA DE PESQUISA..... | 12 |
| 1.2 | QUESTÃO DE PESQUISA | 13 |
| 1.3 | OBJETIVOS | 13 |
| 1.4 | METODOLOGIA E ORGANIZAÇÃO DO TRABALHO | 14 |
| 2 | NORMAS, PADRÕES E MODELOS DE PROCESSO..... | 16 |
| 2.1 | ISO/IEC 12207..... | 17 |
| 2.2 | ISO/IEC 15504..... | 18 |
| 2.3 | CMMI – CAPABILITY MATURITY MODEL INTEGRATION..... | 19 |
| 2.4 | MPS.BR – MODELO DE PROCESSO DE SOFTWARE BRASILEIRO | 21 |
| 2.5 | IEEE 830 | 22 |
| 2.6 | ISO/IEC 9126..... | 23 |
| 2.7 | CONSIDERAÇÕES FINAIS..... | 24 |
| 3 | ENGENHARIA DE REQUISITOS..... | 27 |
| 3.1 | PROCESSO DE ENGENHARIA DE REQUISITOS..... | 27 |
| 3.1.1 | <i>Concepção</i> | 28 |
| 3.1.2 | <i>Levantamento</i> | 28 |
| 3.1.3 | <i>Elaboração</i> | 32 |
| 3.1.4 | <i>Negociação</i> | 33 |
| 3.1.5 | <i>Especificação</i> | 34 |
| 3.1.6 | <i>Validação</i> | 35 |
| 3.1.7 | <i>Gestão</i> | 36 |
| 3.2 | PROCESSO UNIFICADO E A DISCIPLINA DE REQUISITOS | 37 |
| 3.2.1 | <i>Fases do Processo Unificado</i> | 39 |
| 3.2.2 | <i>Disciplina de Requisitos</i> | 41 |
| 3.2.2.1 | <i>Analisar o Problema</i> | 43 |
| 3.2.2.2 | <i>Compreender as Necessidades dos Envolvidos</i> | 44 |
| 3.2.2.3 | <i>Definir o Sistema</i> | 45 |
| 3.2.2.4 | <i>Gerenciar o Escopo do Sistema</i> | 46 |
| 3.2.2.5 | <i>Refinar a Definição do Sistema</i> | 47 |
| 3.2.2.6 | <i>Gerenciar Requisitos Mutáveis</i> | 48 |
| 3.2.3 | <i>Papéis da Disciplina de Requisitos</i> | 49 |
| 3.2.4 | <i>Artefatos da Disciplina de Requisitos</i> | 51 |
| 3.3 | MODELO DE CASOS DE USO..... | 53 |
| 3.3.1 | <i>Principais Elementos de um Caso de Uso Completo</i> | 56 |
| 3.3.1.1 | <i>Ator Primário</i> | 57 |
| 3.3.1.2 | <i>Nível</i> | 57 |
| 3.3.1.3 | <i>Stakeholders e Interesses</i> | 57 |
| 3.3.1.4 | <i>Pré-Condições</i> | 58 |
| 3.3.1.5 | <i>Garantias Mínimas</i> | 58 |
| 3.3.1.6 | <i>Garantias de Sucesso</i> | 59 |
| 3.3.1.7 | <i>Acionistas</i> | 59 |
| 3.3.1.8 | <i>Cenário de Sucesso Principal</i> | 60 |
| 3.3.1.9 | <i>Extensões</i> | 61 |
| 3.3.1.10 | <i>Lista de Variáveis Tecnológicas e de Dados</i> | 62 |
| 3.3.2 | <i>Narrativas de Uso</i> | 63 |

| | | |
|----------|--|------------|
| 3.3.3 | <i>Diagrama de Casos de Uso</i> | 63 |
| 3.4 | CONSIDERAÇÕES FINAIS | 64 |
| 4 | PROCESSO DE ENGENHARIA DE REQUISITOS PARA O NUSIS | 66 |
| 4.1 | CARACTERÍSTICAS DO NUSIS | 67 |
| 4.2 | ATRIBUTOS DE QUALIDADE DESEJÁVEIS | 68 |
| 4.3 | PROCESSO PROPOSTO | 70 |
| 4.3.1 | <i>Concepção e Análise do Problema</i> | 72 |
| 4.3.2 | <i>Levantamento e Compreensão dos Requisitos</i> | 75 |
| 4.3.3 | <i>Negociação de Requisitos</i> | 79 |
| 4.3.4 | <i>Especificação e Modelagem dos Requisitos</i> | 80 |
| 4.3.5 | <i>Validação dos Requisitos</i> | 82 |
| 4.3.6 | <i>Gerência de Requisitos</i> | 83 |
| 4.4 | CONSIDERAÇÕES FINAIS | 85 |
| 5 | ANÁLISE DAS FERRAMENTAS CASE | 87 |
| 5.1 | PROCESSO DE AVALIAÇÃO DA FERRAMENTA CASE | 87 |
| 5.2 | OSRMT | 94 |
| 5.3 | REQUISITEPRO | 97 |
| 5.4 | RESULTADOS OBTIDOS | 101 |
| 5.5 | CONSIDERAÇÕES FINAIS | 103 |
| 6 | APLICAÇÃO DO MODELO DE PROCESSO PROPOSTO | 104 |
| 6.1 | CONCEPÇÃO E ANÁLISE DO PROBLEMA | 105 |
| 6.2 | LEVANTAMENTO E COMPREENSÃO DOS REQUISITOS | 108 |
| 6.3 | NEGOCIAÇÃO DE REQUISITOS | 114 |
| 6.4 | ESPECIFICAÇÃO E MODELAGEM DOS REQUISITOS | 115 |
| 6.5 | VALIDAÇÃO DOS REQUISITOS | 120 |
| 6.6 | GERÊNCIA DE REQUISITOS | 121 |
| 6.7 | CONSIDERAÇÕES FINAIS | 125 |
| 7 | CONSIDERAÇÕES FINAIS | 128 |
| 7.1 | TRABALHOS FUTUROS | 129 |
| 8 | REFERÊNCIAS | 130 |
| 9 | ANEXOS | 130 |

INTRODUÇÃO

Diante das constantes mudanças e dos intensos avanços tecnológicos que vem ocorrendo nas últimas décadas, a informação passou a ser um recurso estratégico das empresas. O *software* se tornou então, a força motora desta nova era. A integridade das informações oferecidas por um *software* pode diferenciar uma empresa de suas concorrentes. Ele é capaz de manipular o bem mais importante para uma empresa – a informação. A fim de evitar que o desenvolvimento de *software* seja realizado com baixa qualidade e pouca previsibilidade de custos e recursos, surgiram técnicas de engenharia de *software*.

Entende-se por engenharia de *software* uma disciplina que integra processo, métodos e ferramentas no desenvolvimento, operação e manutenção de *software*. Seu objetivo é auxiliar no processo de produção do *software*, de forma que o processo dê origem a produtos de alta qualidade, produzidos mais rapidamente e a um custo cada vez menor. Para poder aplicar esses princípios na construção de sistemas de *software*, os envolvidos devem estar equipados com as metodologias apropriadas e com os métodos e as ferramentas específicas que o auxiliarão a incorporar as propriedades desejadas aos processos e produtos (Carvalho, 2005).

A engenharia de *software* está diretamente relacionada a todos os aspectos da produção de software, desde os estágios iniciais de especificação do sistema até sua manutenção, depois de entrar em operação. Um dos maiores desafios enfrentados ao longo do desenvolvimento de *software* seja o processo de requisitos, que é apontado como crítico dentre as atividades de engenharia de *software* (Sommerville, 2007).

Os requisitos compreendem a essência de um sistema de *software*. Eles definem as funcionalidades que o sistema deve fornecer e sob quais condições o sistema deve operar. Em outras palavras, podemos dizer que são os requisitos os responsáveis por estabelecer as funções que o sistema deve possuir e as restrições que deve satisfazer.

Requisitos de *software*, de modo geral, podem ser classificados em dois grandes grupos: requisitos funcionais e requisitos não funcionais. Os requisitos funcionais descrevem as funcionalidades desejadas do sistema, seu comportamento e “o que” se espera que o *software* faça. As questões que se referem a requisitos funcionais têm respostas independentes da implementação de uma solução para o problema do cliente. (Pfleeger, 2004). Já os requisitos não funcionais são aqueles não diretamente relacionados às funções fornecidas pelo sistema. Eles podem especificar desempenho, proteção, disponibilidade e outras propriedades

emergentes do sistema (Sommerville, 2011). Expressam “como” deve ser feito. Um conceito complementar é que os requisitos não funcionais descrevem restrições para o sistema, restrições essas que limitam as opções para criar-se uma solução para o problema. Geralmente estão ligadas a questões inerentes à linguagem, plataforma, técnicas ou ferramentas de implementação (Pfleeger, 2004).

Em um artigo sobre os avanços tecnológicos na construção de *software*, Brooks (1987) comenta que “a parte mais árdua na construção de um sistema de *software* é decidir o que construir. Nenhuma outra parte do trabalho compromete mais o sistema se for feito de forma imprópria”. Para que essa decisão seja tomada de maneira adequada visando satisfazer as expectativas existentes, é necessário entender os requisitos antes que o projeto e a construção de um sistema baseado em computador possam começar. Para conseguir isso, um conjunto de tarefas de engenharia de requisitos é conduzido (Pressman, 2006).

Independente do projeto, a engenharia de requisitos é uma das fases mais importantes no desenvolvimento de *software*, pois nela são identificados, analisados e definidos os propósitos, funcionalidades e o escopo do *software* (Lopes, 2004).

A engenharia de requisitos pode ser vista como uma subárea da engenharia de *software*. Segundo (Pressman, 2006) ela “fornece o mecanismo apropriado para entender o que o cliente deseja, analisando as necessidades, avaliando a exequibilidade, negociando uma condição razoável, especificando a solução de modo não ambíguo, validando a especificação e gerindo os requisitos à medida que eles são transformados em um sistema operacional”.

A engenharia de requisitos ocorre durante a comunicação com o cliente, e compreende as atividades relacionadas à produção de *software* – concepção, levantamento, elaboração, negociação, especificação, validação e gestão (Pressman, 2006).

O *software*, bem como qualquer outro sistema complexo, evolui com o passar do tempo. À medida que o desenvolvimento ocorre, requisitos de negócio e de produto podem ser alterados (Pressman, 2006). Para desenvolver um *software*, um processo ou metodologia são fundamentais, independentemente das técnicas, ferramentas ou notações de *software*. O processo unificado é um modelo de processo que tem como características principais a adoção de casos de uso como ponto de partida para a etapa de especificação de requisitos; ser iterativo, ou seja, possuir flexibilidade incremental com ciclos de projeto menores gerando as evoluções do produto; e por último, ser centrado na arquitetura, procurando modelá-la de acordo com os processos estáticos e dinâmicos do projeto (Rezende, 2005).

No processo unificado, a vida de um sistema de *software* pode ser dividida em uma série de ciclos, onde são definidas disciplinas que compreendem as diferentes fases desses

ciclos (Scott, 2003). Cada disciplina representa um conjunto de atividades executadas por vários membros do projeto e abrange o que deve ser realizado para que um determinado conjunto de artefatos seja produzido (Rational, 2002). Dentre estas, a disciplina de requisitos estabelece um *workflow* de atividades que orienta como as necessidades devem ser coletadas junto aos *stakeholders* e transformadas em um conjunto de requisitos capaz de satisfazer seus interesses.

A maneira com que são desempenhadas as atividades da engenharia de requisitos é fator determinante para obtenção de um produto final de qualidade. Diversas normas, padrões e modelos de qualidade para o processo de desenvolvimento de *software* surgiram em paralelo à evolução da engenharia de *software*. Na sua maioria, apresentam um conjunto de práticas tidas como recomendáveis, com o objetivo de assegurar a qualidade do produto de trabalho resultante das tarefas realizadas. Considerando que o *software* é um produto que visa prover valor a quem o utiliza, a sua qualidade nada mais é que o atendimento das necessidades desse alguém. Um *software* de qualidade é aquele que faz, de maneira correta, o que o cliente precisa que ele faça.

1.1 PROBLEMÁTICA DE PESQUISA

Na Universidade de Caxias do Sul, o NUSIS – Núcleo de Sistemas de Informação conta com um projeto para o desenvolvimento de sistemas de informação para uso acadêmico e empresarial. Tal projeto encontra-se em fase inicial, e tem como propósito a geração de um *software* de domínio público, construído a partir de uma equipe de trabalho formada pelos próprios acadêmicos e docentes da universidade.

Os sistemas a serem desenvolvidos visam à integração de diversas áreas de negócio em um único produto, de modo a promover a integração de alunos e professores de diferentes cursos da universidade ao longo do projeto. Tendo relação a isso, o foco inicial do projeto visa à construção de um ERP - Sistema Integrado de Gestão Empresarial voltado para a plataforma Web, com o objetivo de integrar todos os dados e processos de uma organização em um único sistema.

Para desenvolver um *software*, um processo ou metodologia são fundamentais, independentemente das técnicas, ferramentas ou notações de *software*. A adoção do processo unificado, em conjunto a UML – Linguagem Unificada de Modelagem e ao paradigma de orientação a objetos e a UML, é uma exigência do NUSIS perante o ambiente de desenvolvimento proposto, e visa tornar clara a necessidade de atribuições de tarefas aos

indivíduos envolvidos diretamente no projeto. Outra premissa estabelecida é que as ferramentas utilizadas, além de atenderem aos requisitos do processo, devem ser compatíveis com ferramentas já utilizadas na Universidade, não acarretando geração de investimentos.

Por se tratar de um sistema desenvolvido a partir de um ambiente experimental no NUSIS, as necessidades existentes não partem de um cliente/contratante do serviço. A engenharia de requisitos deve observar a característica da alta rotatividade e inexperiência da equipe de trabalho, uma vez que as tarefas resultantes da análise de requisitos serão desempenhadas pelos próprios alunos dos cursos do Centro de Computação e Tecnologia da Informação através das disciplinas de Estágio Curricular e TCC. Possivelmente alguns alunos possuirão experiência profissional nas áreas em que irão atuar, no entanto, para grande parte dos envolvidos os conceitos e o negócio serão novidade e acarretarão em incertezas. Desta forma, o acompanhamento dos professores ao longo do projeto será crucial para garantir que as atividades sejam corretamente executadas. Os alunos, no papel de engenheiro de requisitos, poderão fazer uso de sistemas similares existentes no mercado de *software*, buscando características de negócio que colaborem nas tarefas de concepção e levantamento de requisitos.

1.2 QUESTÃO DE PESQUISA

Com base na contextualização apresentada na seção anterior, o objetivo deste trabalho consiste em responder à seguinte questão de pesquisa: *Como garantir a qualidade das atividades da engenharia de requisitos no ambiente de desenvolvimento de software proposto para o NUSIS?*

1.3 OBJETIVOS

O objetivo geral deste trabalho é propor um modelo de processo de engenharia de requisitos destinado ao ambiente experimental de desenvolvimento de *software* proposto para o NUSIS. Esse processo pretende definir as atividades que deverão ser desenvolvidas, as técnicas que serão utilizadas, os artefatos que serão produzidos e a ferramenta de gestão de requisitos que será empregada nas atividades da engenharia de requisitos, visando garantir a qualidade do produto resultante das mesmas.

A avaliação do modelo proposto será realizada através da aplicação em um estudo de caso.

Para atender a este objetivo geral emergem os seguintes objetivos específicos:

- Análise das diferentes técnicas e métodos existentes no processo de engenharia de requisitos por meio de pesquisa bibliográfica;
- Identificação e apontamento dos atributos e características de qualidade desejáveis para o projeto do NUSIS;
- Elaboração de uma proposta preliminar do processo de engenharia de requisitos para o ambiente de desenvolvimento de *software* proposto para o NUSIS;
- Análise e seleção da ferramenta *CASE* adequada às diferentes atividades do processo de engenharia de requisitos proposto, visando satisfazer as particularidades apresentadas;
- Avaliação do protótipo e ajustes no processo proposto.

1.4 METODOLOGIA E ORGANIZAÇÃO DO TRABALHO

Este trabalho será desenvolvido através da abordagem qualitativa utilizando o método de pesquisa exploratória.

Para a realização deste trabalho foram definidas as seguintes atividades:

- Estudar, levantar e estabelecer os atributos e características de qualidade desejáveis para o processo de engenharia de requisitos de acordo com o problemática de pesquisa apresentada, com base em estudos bibliográficos do padrão IEEE 830, das normas ISO/IEC 9126, ISO/IEC 12207 e ISO/IEC 15504, e dos modelos de maturidade CMMI e MPS.BR;
- Aprofundar os estudos bibliográficos sobre a engenharia de requisitos e a disciplina de requisitos do processo unificado;
- Propor um esboço preliminar do processo de engenharia de requisitos que se adéque ao NUSIS;
- Avaliar as ferramentas a serem utilizadas ao longo das atividades, sendo essas de grande importância nas tarefas de construção dos artefatos da engenharia de requisitos;
- Aplicar um estudo de caso em um conjunto de funcionalidades do ERP a ser definido utilizando o modelo proposto;

- Avaliar o modelo proposto através de um estudo de caso em ambiente de protótipo. Tal avaliação não contará com indicadores mensuráveis, uma vez que estes só são possíveis de obtenção através da aplicação em um ambiente real;
- Aprimorar o processo de engenharia de requisitos proposto;
- Ajustes no processo proposto com base na avaliação realizada.

O trabalho está dividido em capítulos. No capítulo 2, são abordadas as normas, padrões e modelos de qualidade relacionados às atividades da engenharia de requisitos e ao produto final de *software* produzido resultante do processo de desenvolvimento. Este estudo possibilitou a identificação dos atributos e características de qualidade desejáveis para o ambiente do NUSIS.

No capítulo 3 são apresentados os estudos bibliográficos sobre a engenharia de requisitos e suas atividades, sobre o processo unificado e a disciplina de requisitos, e também sobre o modelo de casos de uso utilizado na especificação de requisitos de *software*. Este estudo compõe o embasamento teórico para a criação da proposta do processo de engenharia de requisitos a qual esse trabalho se propõe.

Tal processo é apresentado e descrito ao longo do capítulo 4. Inicialmente é caracterizado o ambiente a qual o processo destina-se e quais são os atributos de qualidade desejáveis. A contextualização do processo é feita considerando esses aspectos e com base nos estudos realizados, visando garantir a qualidade das atividades da engenharia de requisitos para o ambiente do NUSIS.

No capítulo 5 foi realizada a análise e avaliação das ferramentas *CASE* destinadas às atividades da engenharia de requisitos com base nas normas ISO/IEC 14598 e ISO/IEC 9126, visando identificar àquela que melhor satisfaz às necessidades do NUSIS de acordo com o modelo de processo proposto.

O capítulo 6 tem por objetivo aplicar o modelo de processo de engenharia de requisitos proposto no capítulo 4 em um estudo de caso utilizando a ferramenta de gerenciamento de requisitos RequisitePro, para desta forma avaliar a aferição do modelo e da ferramenta ao ambiente experimental de desenvolvimento de *software* do NUSIS.

Por fim, no capítulo 7 são trazidas as considerações finais do trabalho, descrevendo os objetivos alcançados, os problemas enfrentados e deixando sugestões para trabalhos futuros.

2 NORMAS, PADRÕES E MODELOS DE PROCESSO

Qualidade de *software* é definida pelo IEEE (*The Institute of Electrical and Electronics Engineers*) como "o grau com que um sistema, componente ou processo atende (1) aos requisitos especificados e (2) às expectativas ou necessidades de clientes ou usuários". Já a ISO (*The International Standards Organization*) define qualidade como "a totalidade de características de um produto ou serviço que comprovam sua capacidade de satisfazer necessidades especificadas ou implícitas". Estas duas definições mostram que a qualidade de um produto está estritamente ligada ao atendimento de seus requisitos.

Durante o processo de *software* os requisitos mudam constantemente, seja em virtude do melhor entendimento do problema, com o amadurecimento da compreensão dos envolvidos acerca do que desejam que o *software* faça, ou ainda de alterações de hardware, *software* ou ambiente organizacional no qual o sistema está inserido. Além disso, novos requisitos surgem à medida que novas necessidades e prioridades são descobertas por parte dos *stakeholders* ao longo do processo. Pode-se dizer desta forma, que os requisitos devem evoluir visando refletir as novas visões do problema (Sommerville, 2007).

A preocupação com a qualidade na produção de *software* inicialmente centrou foco na qualidade do produto; esta visão evoluiu e atualmente a preocupação com qualidade envolve tanto o processo de produção (o ciclo de desenvolvimento do *software*) quanto o resultado final - o produto ou *software* gerado (Sayão, 2003).

A busca pelo aprimoramento das atividades no processo de desenvolvimento de *software* deu origem as normas ISO/IEC 12207 e ISO/IEC 15504, e aos modelos de processo CMMI (*Capability Maturity Model Integration*) e MPS.BR (Modelo de Processo do *Software* Brasileiro). Essas normas e modelos representam o processo de *software* sob a ótica de seu funcionamento, mas não apresentam como estes processos devem ser definidos, sendo esta uma responsabilidade de atribuição das fabricantes de *software*. Nelas é apresentado um conjunto de resultados esperados em termos de qualidade para a engenharia de requisitos (Lahoz, 2003).

Por sua vez, o IEEE apresenta um documento que orienta os desenvolvedores quanto às atividades de especificação de requisitos de *software* (IEEE, 1998).

E já relacionado ao *software* como produto final, a ISO/IEC 9126 define um modelo de qualidade com um conjunto de características e subcaracterísticas de que aferem as capacidades do *software* no atendimento aos atributos de qualidade (ISO/IEC, 2003). Para que

se possa determinar um processo de engenharia de requisitos deve-se ter claro o que é qualidade para um produto de *software*.

2.1 ISO/IEC 12207

A norma internacional ISO/IEC 12207 é usada como referência por empresas produtoras e prestadoras de serviços de *software* em muitos países, com o objetivo de auxiliar os envolvidos na construção de *software* a definir seus papéis, e por meio de processos bem definidos, proporcionar a obtenção de um entendimento das atividades a serem executadas (ROCHA, 2011).

A norma estabelece uma arquitetura de alto nível para o ciclo de vida do *software*, que abrange desde a concepção até a descontinuidade do mesmo. Os processos que envolvem o ciclo de vida do *software* são agrupados em classes, dentre elas a classe de processos fundamentais, onde está contido o processo de desenvolvimento (Rocha, 2011).

Neste processo, segundo a norma ISO/IEC 12207, os requisitos são divididos em requisitos de sistema e requisitos de *software* (ISO/IEC, 2008).

A especificação dos requisitos de sistema deve ser devidamente documentada e deve descrever as funções e capacidades do *software* através de requisitos funcionais; requisitos de negócio, organizacionais e de usuários; requisitos não funcionais; requisitos de proteção, de segurança, de ergonomia, de interface, de operações e de manutenção; além de restrições de projeto e requisitos de qualificação (ISO/IEC, 2008).

Na análise dos requisitos de *software*, o desenvolvedor deve estabelecer e documentar os requisitos do *software*, incluindo as especificações das características de qualidade, que além das aplicáveis aos requisitos de sistema, deve contar com a definição de dados e requisitos de bases de dados, requisitos de instalação e aceitação do produto, documentação do usuário, requisitos do usuário para execução, operação e manutenção do *software* (ISO/IEC, 2008).

De acordo com a ISO/IEC 12207, a avaliação dos requisitos deve ser realizada e ter seus resultados documentados, considerando os critérios abaixo listados (ISO/IEC, 2008):

- Rastreabilidade para os requisitos do sistema e projeto do sistema;
- Consistência externa com os requisitos do sistema;
- Consistência interna;
- Testabilidade;

- Viabilidade do projeto do *software*;
- Viabilidade da operação e manutenção.

2.2 ISO/IEC 15504

O padrão, ou modelo de referência ISO/IEC 15504, também conhecido como SPICE, é uma evolução da norma ISO/IEC 12207 para definição do processo de desenvolvimento de *software*. Neste padrão os processos são divididos em categorias e em níveis de capacidade (Rocha, 2001).

Segundo (Rouiller, 2008), a ISO/IEC 15504 é uma norma que representa um padrão internacional emergente que estabelece um *framework* para construção de processos de avaliação e melhoria do processo de *software*. Este *framework* pode ser utilizado pelas empresas envolvidas em planejar, gerenciar, monitorar, controlar e melhorar a aquisição, fornecimento, desenvolvimento, operação, evolução e suporte do *software*. A ISO/IEC 15504 não constitui um método isolado para avaliação, e sua característica genérica possibilita a utilização em conjunto a uma variedade de métodos, técnicas e ferramentas (Lahoz, 2003).

A Gerência de Requisitos é tratada na ISO/IEC 15504 dentro do Processo de Engenharia, mais especificamente nos Processos de Desenvolvimento (ENG.1). De acordo com a norma, a finalidade dos processos de desenvolvimento é transformar um conjunto de exigências (requisitos) em um produto (*software*), produto este que deve atender a todo o conjunto de exigências especificadas (ISO/IEC, 2003).

A seguir são detalhados os processos (ENG.1.1 e ENG.1.2) que englobam a Gerência de Requisitos na norma ISO/IEC 15504 (ISO/IEC, 2003):

- **ENG.1.1 - Análise e Projeto de Requisitos de Sistema:** o objetivo deste processo é estabelecer requisitos funcionais e não funcionais, bem como a arquitetura do sistema, identificando quais requisitos devem ser alocados a quais elementos do sistema. Quando a execução deste processo for bem sucedida, obtêm-se as seguintes saídas:
 - Os requisitos do sistema serão identificados de acordo com a necessidade do cliente;
 - Uma solução será proposta, identificando os principais elementos do sistema;
 - Os requisitos serão alocados de acordo com os elementos principais do sistema;

- Um projeto será desenvolvido, através da definição de prioridade de implementação de cada requisito do sistema;
 - Todos os requisitos serão aprovados e passíveis de alteração quando necessário;
 - A solução proposta, os requisitos levantados e seus relacionamentos, serão comunicados a todas as partes interessadas.
- **ENG.1.2 - Análise de Requisitos do *Software*:** esse processo tem como objetivo estabelecer os requisitos dos componentes de *software* do sistema. A execução bem sucedida deste processo resultará nas seguintes saídas:
- Os requisitos dos componentes do *software* serão definidos;
 - Os requisitos analisados, aprovados e testados serão implementados;
 - O impacto dos requisitos no ambiente de utilização será estudado e entendido;
 - Um cronograma de implementação e disponibilização do *software* será desenvolvido, considerando as prioridades dos requisitos;
 - Todos os requisitos serão aprovados e passíveis de alteração quando necessário;
 - Será estabelecida a consistência entre os requisitos do sistema e os requisitos do projeto de *software*;
 - Os requisitos do *software* serão comunicados a todas as partes interessadas.

2.3 CMMI – *CAPABILITY MATURITY MODEL INTEGRATION*

O CMMI - *Capability Maturity Model Integration* é um modelo de processo, criado e mantido pelo SEI (*Software Engineering Institute*), que recomenda um conjunto de “melhores práticas” para os processos de desenvolvimento de *software*. Atualmente, o CMMI é o modelo mais recomendado para empresas que pretendem exportar *software* ou serviços de TI, devido a sua ampla aceitação no mercado comprador internacional (Taurion, 2005).

O modelo CMMI não define como um processo deve ser implementado, mas prescreve a estrutura, objetivos e graus de qualidade com que o trabalho deve ser realizado, através de uma divisão em níveis de maturidade (SEI, 2010).

Dentro do modelo CMMI, a gerência de requisitos é uma área de processo que compõe o Nível 2 de Maturidade, conhecido como “Gerenciado”. Neste nível, os processos da

empresa devem ser realizados por pessoas habilitadas seguindo um padrão, de acordo com os controles definidos (SEI, 2010).

Os processos de gerenciamento de requisitos gerenciam todos os requisitos recebidos ou gerados pelo projeto, incluindo requisitos técnicos e não técnicos, bem como os requisitos impostos no projeto. Seu objetivo é gerenciar os requisitos dos produtos e componentes de produtos do projeto e identificar as inconsistências entre estes requisitos e os planos e os produtos de trabalho do projeto (SEI, 2010).

De acordo com o CMMI, são estabelecidas as seguintes atividades (*SP-Specific Practices*) para a gerência de requisitos (SEI, 2010):

- **SP 1.1-1 - Obter um entendimento dos requisitos:** deve-se buscar a compreensão dos requisitos junto aos seus fornecedores, estabelecendo critérios para sua determinação, e assegurando que os significados dos requisitos obtidos são de entendimento compartilhado e compatível.
- **SP 1.2-2 - Obter compromissos com os requisitos:** essa prática assegura que os participantes do projeto se comprometem com os requisitos atuais estabelecidos e com as mudanças resultantes nos planos de projeto, atividades e produtos de trabalho.
- **SP 1.3-1 - Gerenciar mudanças de requisitos:** com o decorrer do projeto novos requisitos são derivados, bem como mudanças nos requisitos já existentes podem ser necessárias. É essencial que esses acréscimos e mudanças sejam gerenciadas de forma eficaz e eficiente. Para possibilitar a correta análise dessas mudanças, faz-se necessário conhecer a fonte de cada requisito e ter documentada toda e qualquer mudança nos requisitos.
- **SP 1.4-2 - Manter a rastreabilidade bidirecional de requisitos:** esta prática tem como intenção manter a rastreabilidade bidirecional dos requisitos, ou seja, de um requisito fonte até seus requisitos de mais baixo nível, e destes de volta para o seu requisito fonte. Tal forma de rastreabilidade auxilia a determinar se a totalidade dos requisitos fonte foi tratada e se todos os requisitos de mais baixo nível podem ser rastreados a partir de uma fonte válida. Adicional a isso, a rastreabilidade de requisitos pode cobrir também relacionamentos com outras entidades e é necessário para análise do impacto de mudanças de requisitos nos planos do projeto, atividades e produtos de trabalho.

- **SP 1.5-2 - Identificar inconsistências entre o trabalho do projeto e os requisitos:** essa prática visa identificar as inconsistências entre os requisitos e os planos do projeto e produtos de trabalho e iniciar as ações de correção quando necessário.

2.4 MPS.BR – MODELO DE PROCESSO DE *SOFTWARE* BRASILEIRO

O projeto MPS.BR (Modelo de Processo do *Software* Brasileiro) apresenta um modelo de referência brasileiro para o processo de *software*, adequado ao perfil das empresas nacionais e compatível com os padrões de qualidade aceitos internacionalmente pela comunidade de *software*, mas a custos acessíveis para a realidade da grande maioria das empresas do país, uma vez que o modelo CMMI apresenta um grau de investimento bastante elevado (Taurion, 2005).

A gerência de requisitos encontra-se no nível G do modelo MPS.BR, sendo este o primeiro de sete níveis crescentes de maturidade. Neste nível, “o propósito do processo de gerência de requisitos é gerenciar os requisitos do produto e dos componentes do produto do projeto e identificar inconsistências entre os requisitos, os planos do projeto e os produtos de trabalho do projeto” (SOFTEX, 2011).

Com a aplicação do nível G do modelo MPS.BR no processo de *software* são esperados os seguintes resultados para a gerência de requisitos (GRE), onde pode ser observada a similaridade com o Nível 2 de maturidade do modelo CMMI: (SEI, 2010)(SOFTEX, 2011)

- **GRE 1.** O entendimento dos requisitos é obtido junto aos fornecedores de requisitos;
- **GRE 2.** Os requisitos são avaliados com base em critérios objetivos e um comprometimento da equipe técnica com estes requisitos é obtido;
- **GRE 3.** A rastreabilidade bidirecional entre os requisitos e os produtos de trabalho é estabelecida e mantida;
- **GRE 4.** Revisões em planos e produtos de trabalho do projeto são realizadas visando identificar e corrigir inconsistências em relação aos requisitos;
- **GRE 5.** Mudanças nos requisitos são gerenciadas ao longo do projeto.

2.5 IEEE 830

O IEEE 830 é uma recomendação prática para escrever especificações de requisitos de *software*. Seu objetivo é descrever o conteúdo e as qualidades fundamentais para a elaboração de um bom documento de especificação de requisitos de *software*. Tal recomendação não identifica nenhum método, nomenclatura ou ferramenta específica a serem utilizadas na preparação do documento de especificação (IEEE, 1998).

De acordo com a recomendação IEEE 830, um conjunto de requisitos deve considerar os atributos de qualidade apresentados a seguir:

- **Correção:** um conjunto de requisitos é considerado correto se todos os requisitos representam algo que deve estar presente no sistema que está sendo desenvolvido, ou seja, os requisitos reais do usuário devem coincidir com os requisitos identificados.
- **Não-ambigüidade:** um conjunto de requisitos é considerado não ambíguo quando pode ser interpretado por todos os envolvidos em um projeto de uma única maneira.
- **Completo:** um conjunto de requisitos é dito completo quando descreve todas as demandas de interesse dos usuários. Estas demandas incluem requisitos funcionais, de desempenho, restrições, atributos e interfaces externas.
- **Consistência:** um conjunto de requisitos é dito consistente se nenhum subconjunto destes requisitos entra em conflito com os demais requisitos do sistema.
- **Classificabilidade:** um conjunto de requisitos é considerado classificável quando pode ser classificado por grau de importância e/ou estabilidade.
- **Verificabilidade:** um requisito é verificável se existe uma forma efetiva, em termos de tempo e custo, para que pessoas ou ferramentas indiquem se um sistema cumpre o requisito.
- **Modificabilidade:** um conjunto de requisitos é modificável quando seu estilo e estrutura é tal que as alterações podem ser realizadas de forma simples e consistente com os demais requisitos.
- **Rastreabilidade:** um requisito é considerado rastreável quando se pode acompanhar a vida de um requisito em ambas as direções do processo de *software* e durante todo o seu ciclo de vida.

2.6 ISO/IEC 9126

A norma ISO/IEC 9126-1 apresenta um conjunto de características que definem um modelo de qualidade e podem ser aplicadas a qualquer produto de *software*. Nela são propostas seis características essenciais para determinar a qualidade de produtos de *software*, sendo que cada característica é refinada em um conjunto de subcaracterísticas, e cada subcaracterística é avaliada por um conjunto de métricas (ISO/IEC, 2003).

As características de qualidade definidas pela norma ISO/IEC 9126 são (ISO/IEC, 2003):

- **Funcionalidade:** conjunto de atributos que evidenciam a existência de funções que satisfazem as necessidades explícitas e implícitas;
- **Usabilidade:** conjunto de atributos que evidenciam o esforço necessário para se poder utilizar o *software*;
- **Confiabilidade:** conjunto de atributos que evidenciam a capacidade do *software* de manter seu nível de desempenho sob condições estabelecidas durante um período de tempo estabelecido;
- **Eficiência:** conjunto de atributos que evidenciam o relacionamento entre o nível de desempenho do *software* e a quantidade de recursos usados, sob condições estabelecidas;
- **Manutenibilidade:** conjunto de atributos que evidenciam o esforço necessário para fazer modificações especificadas no *software*;
- **Portabilidade:** conjunto de atributos que evidenciam a capacidade do *software* de ser transferido de um ambiente para outro

As características e subcaracterísticas definidas pela norma ISO 9126 são apresentadas na tabela 1. Para cada subcaracterística é elencada uma pergunta chave que remete ao objetivo ao qual a mesma se propõe.

Tabela 1 - Características e Subcaracterísticas da ISO/IEC 9126

| Característica | Subcaracterística | Pergunta chave para a subcaracterística |
|--|---------------------|---|
| Funcionalidade (Satisfaz as necessidades?) | Adequação | Propõe-se a fazer o que é apropriado? |
| | Acurácia | Faz o que foi proposta de forma correta? |
| | Interoperabilidade | Interage com os sistemas especificados? |
| | Conformidade | Está de acordo com as normas, leis, etc? |
| | Segurança de acesso | Evita acesso não autorizado aos dados? |
| Confiabilidade (É imune a falhas?) | Maturidade | Com que frequência apresenta falhas? |
| | Tolerância a falhas | Ocorrendo falhas como ele reage? |
| | Recuperabilidade | É capaz de recuperar dados em caso de falhas? |
| Usabilidade (É fácil de usar?) | Inteligibilidade | É fácil de entender o conceito e a aplicação? |
| | Apreensibilidade | É fácil de aprender a usar? |
| | Operacionalidade | É fácil de operar e controlar? |
| Eficiência (É fácil, rápido e “enxuto”?) | Tempo | Qual é o tempo de resposta, a velocidade de execução? |
| | Recursos | Quanto recurso usa? Durante quanto tempo? |
| Manutenibilidade (É fácil de modificar?) | Analisabilidade | É fácil de encontrar uma falha quando ocorre? |
| | Modificabilidade | É fácil modificar e adaptar? |
| | Estabilidade | Há grande risco quando se faz alterações? |
| | Testabilidade | É fácil testar quando se faz alterações? |
| Portabilidade (É fácil de usar em outro ambiente?) | Adaptabilidade | É fácil adaptar em outros ambientes? |
| | Instabilidade | É fácil instalar em outros ambientes? |
| | Conformidade | Está de acordo com padrões de portabilidade? |
| | Substituibilidade | É fácil usar para substituir outro? |

Fonte: adaptada de (ISO/IEC, 2003).

2.7 CONSIDERAÇÕES FINAIS

No desenvolvimento de *software*, a qualidade do produto está diretamente relacionada à qualidade do processo de desenvolvimento. Pode-se definir ainda, que a produção de um *software* com qualidade possui direta relação à conformidade para com seus requisitos. Neste capítulo foram estudados diferentes modelos, padrões e normas de qualidade ligadas ao processo de gerência de requisitos e também ao produto final de *software*, visando estabelecer os atributos de qualidade desejáveis ao processo de engenharia de requisitos a ser proposto para o ambiente experimental do NUSIS.

A norma ISO/IEC 12207 é usada como referência por empresas desenvolvedoras de *software* na definição de papéis e responsabilidades aos envolvidos, e objetiva tornar clara as

atividades que devem ser executadas dentro de um processo de *software*. Ela define uma estrutura para os processos de ciclo de vida do *software*, podendo ser adaptada a qualquer empresa (Rocha, 2011).

A norma ISO/IEC 15504 é uma evolução da ISO/IEC 12207, e estabelece através de sua característica genérica um *framework* que pode ser utilizado para construção de processos de avaliação e melhoria do processo de *software*. Seu objetivo é garantir que as empresas consigam cobrir pontos vulneráveis do processo produtivo (Lahoz, 2003).

O modelo CMMI recomenda um conjunto de “melhores práticas” para os processos de desenvolvimento de *software*. Enquanto que o MPS.BR pode ser tido como uma adaptação brasileira do CMMI, uma vez que este apresenta um grau de investimento considerado elevado para ser aplicado por pequenas e médias empresas (Taurion, 2005). Ambos trazem a gerência de requisitos com resultados esperados em comum (Hirama, 2012):

- O entendimento dos requisitos deve ser obtido junto ao cliente;
- Os requisitos devem ser avaliados e deve haver o comprometimento com os mesmos;
- Deve ser mantida rastreabilidade bidirecional entre os requisitos e mais produtos de trabalho;
- Devem ocorrer revisões visando identificar inconsistências entre o produto de trabalho e os requisitos elicitados;
- Mudanças nos requisitos ao longo do ciclo de vida devem ser gerenciadas.

Os modelos CMMI e MPS.BR, assim como a norma ISO/IEC 15504, não estabelecem um processo único para as atividades que envolvem a definição e o gerenciamento de requisitos, elas apenas prescrevem recomendações de documentação e informações necessárias para as atividades (ISO/IEC 2003)(SEI, 2010)(SOFTEX, 2011). Em outras palavras, as abordagens indicam o que fazer e não como fazer.

Atrelado ao isso, o IEEE elaborou um documento com recomendações práticas que orientam os desenvolvedores quanto às atividades de especificação de requisitos de *software*, no qual são trazidos os atributos de qualidade desejáveis para um conjunto de requisitos (IEEE, 1999).

Já a ISO/IEC 9126 destina-se a avaliação da qualidade de um produto de *software*, e não do seu processo de desenvolvimento. Nesta norma são estabelecidas características tidas

como importantes para a aferição da qualidade de um produto de *software* e as métricas utilizadas para esse fim (ISO/IEC 2003).

Por fim, através do estudo realizado neste capítulo percebeu-se a importância da adoção de boas práticas ao longo das atividades da engenharia de requisitos como forma de garantir a qualidade nos produtos de trabalho gerados, e conseqüentemente no produto final construído através do processo de desenvolvimento de *software*.

3 ENGENHARIA DE REQUISITOS

A principal forma de se medir o sucesso no desenvolvimento de um *software* se dá através da forma como ele cumpre com as tarefas para as quais foi proposto. Para que esse sucesso possa ser atingido, é fundamental identificar e documentar as necessidades e os propósitos do *software*, o que exige a compreensão do ambiente onde o *software* será inserido, considerando as características do negócio, as necessidades reais existentes, e a possibilidade de modificações futuras (Lopes, 2004).

Segundo (Pressman, 2006), não existe uma forma incontestável de assegurar que a especificação construída pela engenharia de requisitos estará em total conformidade com as necessidades dos *stakeholders*, e que irão satisfazer suas necessidades. Este é justamente o maior desafio encontrado na engenharia de requisitos, e para isso um conjunto de atividades deve ser estabelecido e conduzido ao longo de um projeto de *software*.

A Engenharia de Requisitos é a disciplina que procura sistematizar o processo de definição dos requisitos, propondo um conjunto de métodos, técnicas e ferramentas que o suportem (Rocco, 2001). Ela fornece o mecanismo apropriado para entender as necessidades do cliente e transformar suas ideias em um *software* que as satisfaçam. As atividades da engenharia de requisitos vão desde a concepção do sistema até a modelagem conceitual do que vai ser construído (Rocco, 2001).

3.1 PROCESSO DE ENGENHARIA DE REQUISITOS

Os requisitos de um sistema devem expressar o que se espera que *software* faça, os serviços que oferece e as restrições impostas ao seu funcionamento. O processo de engenharia de requisitos objetiva descobrir, analisar, documentar e verificar esses serviços e restrições (Sommerville, 2009).

Não existe um processo único para a engenharia de requisitos. Cada organização deve buscar desenvolver seu próprio processo, de acordo com o conjunto de características envolvidas (Lopes, 2004). Segundo (Pressman, 2006), a engenharia de requisitos ocorre durante a comunicação com o cliente, e compreende as atividades relacionadas à produção de *software* – concepção, levantamento, elaboração, negociação, especificação, validação e gestão, podendo algumas dessas ocorrer em paralelo, e sendo todas adaptáveis às necessidades do projeto.

3.1.1 Concepção

A etapa de concepção serve de ponto de partida para as demais atividades da engenharia de requisitos. Seu objetivo é estabelecer um entendimento básico do problema e dos fatores a ele relacionados, quais os *stakeholders* e a natureza da solução desejada. (Sommerville, 2007) aborda a etapa de concepção como um estudo de viabilidade inicial para o processo de engenharia de requisitos, onde são obtidos um conjunto preliminar de requisitos de negócio, um breve esboço da descrição do sistema, e de como este poderá contribuir com os processos dentro das áreas fim.

Nesta fase inicial são identificados os primeiros *stakeholders* do projeto, sendo que à medida em que o mesmo avança novos interessados vão sendo naturalmente descobertos. Busca-se explorar os diferentes pontos de vista desses *stakeholders* com relação aos requisitos do sistema, categorizando todas as informações de modo a permitir que os tomadores de decisão optem por um conjunto de requisitos consistente. Identificar áreas de concordância; requisitos com os quais todos *stakeholders* concordam, e áreas de conflito; requisitos desejados por um ou mais *stakeholder*, mas que conflitam com as necessidades de outro *stakeholder*, talvez seja o maior desafio dessa etapa (Pressman, 2006).

Trabalha-se com a formulação das primeiras questões para com os *stakeholders*, focando nos objetivos globais e seus benefícios, questões essas “livres de contexto” (Pressman, 2006). Algumas das perguntas que poderiam ser feitas: (Pressman, 2006):

- Quem está por trás da solicitação deste trabalho?
- Quem vai usar a solução?
- Há outra fonte para a solução que você necessita?
- Que problema(s) essa solução enfrentaria?
- Você pode me mostrar (ou descrever) o ambiente de negócios no qual a solução será usada?
- Alguém mais pode fornecer informações adicionais?

3.1.2 Levantamento

Enquanto na etapa de concepção o foco está nos objetivos macros do projeto, durante o levantamento este foco é dirigido para as necessidades de negócio existentes, procurando

entende-las e apresentá-las de forma detalhada, por meio de diferentes abordagens (Pressman, 2006).

Elicitar e compreender os requisitos são uma tarefa difícil de ser conduzida, devido a uma série de fatores. (Pressman, 2006) citando (Christel, 1992) identifica os tipos de problemas que podem dificultar o levantamento de requisitos:

- Problemas de escopo: quando o limite do sistema é mal definido ou então o *stakeholder* especifica detalhes técnicos supérfluos que podem levar a confusão dos objetivos, ao invés do seu esclarecimento.
- Problemas de entendimento: quando o cliente não apresenta clareza quanto as suas necessidades, e detém pouca compreensão acerca do ambiente e do domínio do problema, com dificuldade em comunicar-se, muitas vezes omitindo informações importantes.
- Problemas de volatilidade: requisitos podem mudar ao longo do ciclo de vida de um *software*, o que pode ser difícil de ser gerenciado.

Fora estes problemas, frequentemente existiram conflitos de ideias, além de questões do âmbito político e econômico, as quais deverão ser sempre consideradas pelo engenheiro de requisitos (Sommerville, 2007).

A atividade de levantamento junto ao cliente requer um ambiente favorável à coleta dos requisitos. Para que tal ambiente possa ser propiciado podemos fazer uso de diferentes técnicas: questionários, entrevistas, reuniões, observações in-loco, verificação de documentos e prototipagem (Rezende, 2005).

a) Questionário

Normalmente preparado através de um formulário, é utilizado como instrumento para levantamento das informações desejadas. Pode ser utilizado como um roteiro estruturado de entrevista. Embora traga agilidade ao processo de elicitação, apresenta como desvantagem a possibilidade de que as informações sejam manipuladas antes de seu fornecimento, e também que as respostas sigam uma tendência de concentração em alternativas padrão (Rezende, 2005).

b) Entrevistas

Fazem parte da maioria dos processos de engenharia de requisitos. Nessa técnica, os *stakeholders* são questionados sobre o sistema que eles usam e o sistema a ser desenvolvido, derivando de suas respostas os requisitos (Sommerville, 2011). Recomenda-se a utilização de entrevistas para levantamento de informações passíveis de reflexão, uma vez que elas induzem o entrevistado a fornecer informações sobre determinado assunto (Rezende, 2005). Por meio de entrevistas pode-se verificar posicionamentos pessoais; omissões, medos, desvios, acerca das questões envolvidas, encontrando pistas de como atacar determinada questão os aspectos com os quais deve-se tomar maior precaução, tendo em vista as pessoas e os interesses envolvidos (Tonsig, 2003).

c) Reuniões

Podem ser vistas como uma extensão da técnica de entrevista, onde busca-se ouvir diversos usuários em um mesmo momento. Essa técnica propicia uma interação natural entre os participantes, sendo possível observar diferentes pontos de vista a respeito do problema (Carvalho, 2005) (Rocco, 2001).

O levantamento de requisitos por meio de reuniões ocorre de uma maneira participativa, uma vez que diversos usuários são envolvidos, além dos analistas da equipe de desenvolvimento. Esse encontro possibilita o contraste de opiniões e o enriquecimento do conhecimento elicitado. Deve-se no entanto, ter cuidado para que o foco da reunião seja mantido, não deixando que aspectos pessoais dos participantes se sobressaiam.

Existem dois métodos principais que enfatizam a participação coletiva na especificação do *software*. A técnica *Joint Application Development (JAD)* utiliza-se de sessões estruturadas e disciplinadas, onde os envolvidos se reúnem para desenvolver juntos o sistema de *software* (Rocco, 2001). Enquanto que a técnica *Participatory Design (PD)* tem seu foco voltado ao envolvimento dos usuários, por facilitar o processo de aprendizado entre analistas e usuários através situações de trabalho simuladas e vivenciadas em conjunto (Rocco, 2001).

A técnica de *Joint Application Development* pode ser vista ainda sob a ótica de um *workshop* de requisitos. Através dela, usuários e analistas projetam o sistema juntos em sessões de grupo bem estruturadas e guiadas por um líder, estimulando a criatividade e o trabalho em equipe através de discussões de *brainstorming* sobre as funcionalidades do sistema. Os *workshops* de requisitos diferem das reuniões quanto a sua característica de intensidade, pois além de servirem para identificar as necessidades, já visam também direcionar para a solução através de requisitos específicos (IBBA, 2011).

d) Observações in-loco

Seu objetivo é permitir que uma situação cotidiana possa ser vivenciada pelo engenheiro de requisitos. Por meio desta, pode-se obter a confirmação para informações recebidas e levantar e avaliar os procedimentos realizados pelos usuários. A observação permite que sejam buscados detalhes operacionais não considerados ou então tidos como improdutivos para serem abordados em uma entrevista. Tal técnica pode servir ainda como método de aferição da qualidade dos resultados obtidos através de entrevistas, evitando informações contraditórias ou dúbias. Ao invés de se promover acareações ou refazer entrevistas, pode-se buscar a resposta no exato local onde as atividades se desenvolvem, através da observação da realidade (Rezende, 2005).

Como vantagem, essa técnica não requer a interrupção das atividades e a disponibilização exclusiva de tempo por parte dos usuários. Por outro lado, sua utilização pode gerar um mal-estar na equipe sob observação, comprometendo os resultados. Outra característica negativa da aplicação desta técnica é o fato da mesma não produzir evidências formais (Rezende, 2005).

e) Verificação de Documentos

A verificação dos documentos manipulados pelos processos é uma atividade necessária, porém pode ser considerada complementar. A análise de documentos leva à constatação sobre os dados que são utilizados ou não nos processos existentes. É comum identificar por meio desta técnica a existência de documentos prevendo o preenchimento de campos que são comumente ignorados. Além disso, pode se averiguar situações inversas, nas quais o usuário envolvido em um processo negligência a existência de determinados dados, que só vêm à tona através da inspeção dos documentos (Tonsig, 2003).

A informação de documentos pode ser recuperada por meio da utilização da engenharia reversa. Essa técnica tem como princípio a desmontagem das caixas pretas do *software*, de trás pra frente, ou seja, o processo de recuperação do projeto com especificação e documentação procedimental, arquitetural e de dados. A normalização é essencial neste processo, sendo o principal meio para detecção e eliminação das redundâncias de dados presentes nos arquivos (Rezende, 2005).

f) Prototipagem

Essa técnica tem como objetivo construir partes dos requisitos propostos no intuito de determinar a necessidade e a viabilidade dos requisitos, ou o quanto estes são desejáveis e agregarão valor ao sistema (Pfleeger, 2004).

Durante a atividade de levantamento de requisitos na maioria das vezes os *stakeholders* não possuem convicção sobre aquilo que querem ou necessitam (Pfleeger, 2004). Nessas situações, a comparação com um produto de *software* que sirva de referência pode auxiliar os usuários a entender e expressar melhor as suas necessidades. Quando esse produto não existe, a prototipagem pode ser usada na criação de um esboço de produto que ilustre as características consideradas relevantes, permitindo aos usuários rever de forma mais clara quais são suas reais necessidades para com o sistema (Carvalho, 2005).

Cabe ressaltar que a prototipagem só traz benefícios se o protótipo puder ser construído mais rápido que o sistema real, caso contrário a aplicação desta técnica não traz benefício algum (Carvalho, 2005).

3.1.3 Elaboração

As informações obtidas através das atividades de concepção e levantamento são expandidas e refinadas durante a elaboração. Essa atividade expande ainda mais os requisitos através da criação de um modelo de análise, com a primeira representação técnica do sistema. Para elaboração do modelo de análise uma série de tarefas de modelagem é executada, e levam à especificação dos requisitos de um modo consistente, e conseqüentemente, mais inteligível (Pressman, 2006).

O objetivo do modelo de análise é criar um conjunto de representações capaz de fornecer uma descrição dos requisitos quanto à informação, função e comportamento (Pressman, 2006). Este modelo é composto por diferentes elementos, para os quais utiliza-se a UML – Linguagem de Modelagem Unificada para modelá-los. A UML por sua vez, é uma linguagem padrão para a elaboração da estrutura de projetos de *software*, podendo ser aplicada para a visualização, a especificação, a construção e a documentação de artefatos (Booch, 2000).

De acordo com (Pressman, 2006), o modelo de análise é composto por quatro elementos de modelagem:

- Modelos baseados em cenário: representam os requisitos de *software* sob o ponto de vista do usuário, tendo como principal elemento da modelagem o caso de uso

(a construção de um caso de uso será abordada em capítulos posteriores). Pode-se utilizar ainda diagramas de atividades, que são representações gráficas semelhantes a fluxogramas, para mostrar o fluxo de processamento dentro de um cenário específico, ou ainda diagramas de raia, a fim de ilustrar como esse fluxo é alocado entre diferentes atores ou classes.

- Modelos de fluxo: focalizam o fluxo de objetos de dados à medida em que esses são transformados por funções de processamento. Embora o diagrama de fluxo de dados não seja parte formal da UML, ele pode ser utilizado para complementar os diagramas propostos por esse padrão, fornecendo uma visão adicional dos requisitos e fluxo do sistema, demonstrando como uma entrada é transformada em saída através das funcionalidades do sistema.
- Modelos baseados em classe: esse modelo utiliza a informação derivada da modelagem baseada em cenário e orientada a fluxo para identificar as classes de análise e o relacionamento entre elas. Além do diagrama de classes que objetiva-se construir através desse modelo, uma variedade de outras notações da modelagem UML pode ser aplicada na definição de hierarquias, relacionamentos, associações, agregações e dependências entre as classes.
- Modelos comportamentais: enquanto os três primeiros modelos fornecem uma visão estática do *software*, a modelagem comportamental mostra seu comportamento dinâmico. Esse modelo deve indicar como o *software* responderá a eventos ou estímulos externos. Partindo da análise dos casos de uso e por meio da notação UML, são criados diagramas de estado e diagramas de sequência para a modelagem comportamental.

3.1.4 Negociação

À medida que os requisitos são identificados e o modelo de análise é criado, negociações devem ocorrer envolvendo os *stakeholders* do projeto. Elas servem para conciliar conflitos, definir prioridades, analisar riscos, custos e prazos (Pressman, 2006).

A negociação de requisitos é o processo de discussão dos conflitos de requisitos e a busca de um compromisso no qual todos *stakeholders* estejam de acordo. Essa atividade pode ser dividida em três estágios (Luna, 2008):

1. Discussão dos requisitos: os requisitos que foram identificados como problemáticos são discutidos junto aos *stakeholders*, que por sua vez apresentam seus pontos de vista a respeito;
2. Priorização dos requisitos: os requisitos são priorizados visando identificar aqueles considerados críticos e auxiliar no processo de tomada de decisão;
3. Concordância dos requisitos: soluções para os problemas de requisitos são identificadas e um acordo sobre o conjunto de requisitos é firmado. Mudanças em alguns dos requisitos podem ser necessárias para se chegar a um acordo.

3.1.5 Especificação

De acordo com (Pressman, 2006), a especificação é o produto de trabalho final produzido pelo engenheiro de requisitos, e pode ser um documento escrito, um modelo gráfico, um modelo matemático formal, uma coleção de cenários de uso, um protótipo ou qualquer combinação desses elementos.

A forma de especificação mais utilizada para descrever os requisitos funcionais de um sistema são os casos de uso textuais (Cockburn, 2005). Na seção 3.3 o modelo de representação de requisitos através de casos de uso é descrito em detalhes.

Os requisitos não funcionais são especificados através do artefato de Especificações Suplementares. Nele são captados e identificados outros tipos de requisitos, tais como: relatórios, documentação, empacotamento, capacidade de suporte, licenciamento, entre outros. As especificações suplementares são descritas textualmente através de um documento dividido em seções. Essas seções podem ser categorizadas de acordo com o modelo FURPS+ (Funcional, Usabilidade, Confiabilidade, Desempenho, Suportabilidade + Restrições) (Larman, 2007).

Juntamente, Casos de Uso e Especificações Suplementares definem todos os requisitos do sistema. Os casos de uso descrevem os requisitos comportamentais do sistema, enquanto as Especificações Suplementares descrevem todos os outros requisitos que não são capturados durante a elaboração dos casos de uso (Larman, 2007).

3.1.6 Validação

Segundo (Pfleeger, 2004) “a validação dos requisitos é o processo que determina se a especificação é consistente com a definição dos requisitos”, garantindo que os mesmos atenderão às necessidades dos *stakeholders*.

A atividade de validação consiste na execução de ações com o objetivo de ratificar o conhecimento obtido, aferindo se o que foi especificado realmente retrata as necessidades dos *stakeholders*. Sob essa ótica, a validação vem a ser considerada também uma atividade de negociação, com o contato constante entre o cliente e o engenheiro de requisitos (Rocco, 2001).

Uma forma simples de verificar os requisitos é revisá-los conjuntamente, representantes da equipe de desenvolvimento e *stakeholders*. Essa tarefa de revisão deve envolver as atividades (Pfleeger, 2004):

- Revisão de metas e objetivos traçados para o sistema.
- Comparação dos requisitos com tais metas e objetivos, a fim de verificar que todos os requisitos são necessários.
- Descrição do ambiente onde o sistema irá operar. Deverão ser analisadas todas as interfaces envolvidas; se estas estão corretas e completas, o fluxo e a estrutura das informações do sistema; se os requisitos refletem com precisão a intenção e o entendimento do usuário, e por fim, avaliar novamente os requisitos na procura por possíveis omissões, imperfeições e inconsistências.
- Verificação, avaliação e documentação dos riscos envolvendo o desenvolvimento ou o funcionamento do sistema. Uma vez identificado um risco, o mesmo deve ser discutido e avaliado, buscando alternativas para minimizá-lo e mantendo-o documentado.
- Organizar a forma como irão ocorrer os testes do sistema e quem fornecerá os dados para os testes.

Diversas técnicas, manuais ou automatizadas, podem ser usadas na validação de requisitos. A escolha da técnica a ser utilizada em cada projeto está diretamente relacionada à experiência, preferência e adequação com a maneira de elicitação dos requisitos. Algumas dessas técnicas são descritas na tabela 2.

Tabela 2 - Técnicas para Validação de Requisitos

| | |
|-------------------------------|--|
| Técnicas Manuais | Leitura |
| | Referência Cruzada Manual |
| | Entrevistas |
| | Revisões |
| | Listas de Verificação |
| | Modelos manuais para verificação de funções e relações |
| | Cenários |
| | Provas Matemáticas |
| Técnicas Automatizadas | Referência Cruzada Automatizada |
| | Modelos automatizados para ativar funções |
| | Protótipos |

Fonte: (Pfleeger, 2004)

Com a conclusão da validação dos requisitos, os engenheiros de requisitos e os usuários do sistema deverão estar convictos de que a especificação de requisitos está correta, fazendo com que o projeto possa avançar para as próximas fases (Pfleeger, 2004).

3.1.7 Gestão

A atividade de gestão ou gerência de requisitos é utilizada para compreender e controlar as mudanças dos requisitos de *software* (Sommerville, 2007). Tem como objetivo mapear os requisitos identificados na fase inicial de um projeto e rastreá-los até os artefatos finais, permitindo a validação da consistência entre os mesmos e das requisições originalmente vindas dos solicitantes (SEI, 2010).

De acordo com (Oberg, Probasco e Ericsson, 2002) o gerenciamento de requisitos é:

- uma abordagem sistemática de fazer levantamento, organizar e documentar os requisitos do sistema e
- um processo que estabelece e mantém concordância entre o cliente e a equipe do projeto na alteração de requisitos do sistema.

É importante que os requisitos de um projeto sejam acompanhados individualmente, mantendo ligações entre os requisitos dependentes, de forma que seja possível avaliar o impacto de possíveis mudanças. Para isso, um processo de gerenciamento deve ser estabelecido e mantido (Sommerville, 2011).

O planejamento é um estágio essencial no processo de gestão dos requisitos. Nesse momento deve-se decidir sobre (Sommerville, 2011):

1. Identificação de requisitos: forma como cada requisito será identificado unicamente de modo que possa ser feita referência cruzada entre este e outros requisitos e para que ele possa ser utilizado nas avaliações de rastreabilidade.
2. Processo de gerenciamento de mudanças: conjunto de atividades que avaliam o impacto e o custo das mudanças.
3. Políticas de rastreabilidade: regras que definem os relacionamentos entre os requisitos e entre os requisitos e o projeto do sistema, que devem ser registrados, e como esses registros serão mantidos.
4. Apoio de ferramentas *CASE*: uma ferramenta automatizada pode ser adotada como forma de apoio para a atividade de gerenciamento. Essas ferramentas trazem benefícios quanto ao armazenamento dos requisitos, o gerenciamento de mudanças e o gerenciamento de rastreabilidade.

As informações de rastreabilidade são geralmente representadas através de matrizes de rastreabilidade, podendo relacionar os requisitos aos *stakeholders* que os propuseram, a outros requisitos numa relação de dependência, ou aos módulos de um projeto, nos quais esses requisitos são implementados (Sommerville, 2011). A utilização de matrizes de rastreabilidade podem ser usadas quando um pequeno número de requisitos está envolvido. Para sistemas com muitos requisitos, a utilização e manutenção de matrizes pode ser uma tarefa complexa e dispendiosa. Nestes casos aconselha-se a automatização do processo por meio de uma ferramenta de gerenciamento de requisitos (Sommerville, 2011).

3.2 PROCESSO UNIFICADO E A DISCIPLINA DE REQUISITOS

O processo unificado de desenvolvimento de *software* caracteriza-se como um conjunto de atividades necessárias para transformar requisitos do cliente em um sistema de *software*. Ele é visto ainda como uma estrutura genérica de processo, que pode ser customizada adicionando-se ou removendo-se atividades com base nas necessidades específicas existentes (Scott, 2003).

O modelo estabelecido pelo processo unificado identifica três princípios chave: direcionado a casos de usos, centrado na arquitetura e iterativo e incremental (Scott, 2003). Esses princípios são abordados na sequência.

✓ **Direcionado a Casos de Uso**

Um caso de uso é uma espécie de “história” linear que detalha uma maneira comum de utilizar uma aplicação (Braude, 2004). Ele deve ser capaz de captar as interações que ocorrem entre produtores de *software* e consumidores de informação, além do sistema entre si (Pressman, 2006). Os casos de uso capturam requisitos funcionais, e quando agrupados resultam no modelo de casos de uso, o qual descreve a funcionalidade completa do sistema (Larman, 2007).

Com base no modelo de casos de uso os desenvolvedores guiam suas atividades, criando uma série de modelos de projeto e implementação que os realizam efetivamente. É a partir do modelo de casos de uso também, que os responsáveis pelos testes realizam seu trabalho, visando assegurar que os componentes do modelo de implementação cumpram adequadamente os objetivos traçados nos casos de uso. Os casos de uso servem ainda como balizadores para as atividades dos usuários chaves do negócio, analistas de processo e projetistas, ao longo do processo de desenvolvimento do *software*, permitindo a estes uma melhor assimilação do processo e de suas peculiaridades. Enquanto que o gerente de projeto faz uso do modelo de casos de uso no planejamento das iterações durante a modelagem de negócio e também no rastreamento do progresso do projeto (Larman, 2007).

Desta forma, caracterizar o processo unificado como sendo direcionado a casos de uso nada mais é, que dizer que o processo de desenvolvimento executa uma sequência de tarefas derivadas dos casos de uso, sendo que estes não somente iniciam o processo de desenvolvimento, mas também o mantêm coeso.

✓ **Centrado na Arquitetura**

Embora sejam os casos de uso os responsáveis por dirigir o processo, eles não são selecionados isoladamente, mas sim em conjunto à arquitetura do sistema. Os casos de uso direcionam a arquitetura do sistema, que por sua vez influencia na seleção dos casos de uso, sendo que ambos amadurecem no decorrer do ciclo de vida do sistema.

A arquitetura de um sistema de *software* é descrita através de diferentes visões do sistema em construção. Representa a organização fundamental do sistema como um todo, na qual as características de maior relevância são colocadas em destaque, deixando detalhes de lado (Torossi, 2005). Nela são englobados os elementos estáticos e dinâmicos, a maneira como estes trabalham juntos e o estilo de arquitetura que orienta a organização do sistema. Questões como desempenho, escalabilidade, reuso e restrições econômicas e tecnológicas também estão implicitamente relacionadas à arquitetura do *software* (Scott, 2003).

✓ Iterativo e Incremental

No processo unificado, o desenvolvimento é organizado em uma série de mini-projetos de duração fixa, chamadas iterações. Cada iteração resulta em uma versão do sistema liberada interna ou externamente, devidamente testada, integrada e capaz de ser executada. Presume-se que essa nova versão ofereça uma melhora incremental sobre a anterior, motivo pelo qual a iteração é chamada de incremento (Scott, 2003) (Larman, 2007).

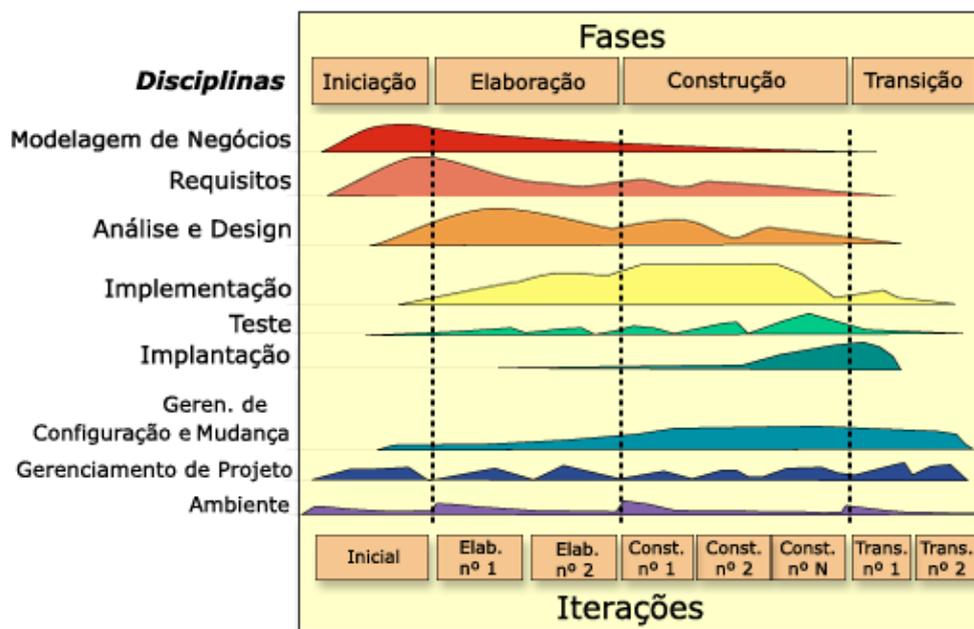
No desenvolvimento iterativo e incremental o sistema cresce incrementalmente ao longo do tempo, ou seja, a cada nova iteração disponibilizada. (Larman, 2007). Cada iteração inclui suas próprias atividades de análise de requisitos, projeto, implementação e teste (Larman, 2007). Em cada iteração, os desenvolvedores identificam e especificam os casos de uso relevantes, criam um projeto utilizando a arquitetura escolhida, implementam o projeto em componentes e verificam se esses componentes satisfazem os casos de uso. Se uma iteração atinge seus objetivos, o desenvolvimento prossegue com a próxima iteração, caso contrário, os desenvolvedores tem de rever suas decisões e tentar uma nova abordagem.

A adoção de um processo iterativo gera uma série de benefícios, permitindo um melhor controle de riscos sobre prazos e custos, a otimização do tempo de desenvolvimento do projeto como um todo, além da facilidade para realizar-se adaptações no projeto caso venham ocorrer alterações nos requisitos (Scott, 2003).

3.2.1 Fases do Processo Unificado

A vida de um sistema de *software* pode ser dividida em uma série de ciclos. Um ciclo é encerrado com a disponibilização de uma versão do sistema ao cliente, e é encarado como uma sequência de quatro fases: iniciação, elaboração, construção e transição. Cada fase é delimitada por um marco, e pode conter uma ou mais iterações. Ao atingir o marco, os resultados obtidos na fase são analisados em comparação aos objetivos principais, e decisões importantes são tomadas sobre a continuidade do projeto (Scott, 2003). Na figura 1 podemos ter uma visão mais detalhada das fases, iterações e marcos.

Figura 1 - Fases do Processo Unificado



Fonte: (Rational, 2002)

- 1. Conceção:** abrange atividades de comunicação com o cliente e planejamento, e estabelece a viabilidade e o escopo do sistema proposto. Nessa fase os requisitos de negócio são identificados, um esboço da arquitetura do sistema é proposto e um plano para a natureza iterativa e incremental do projeto é desenvolvido. Através de casos de uso preliminares são descritos os requisitos fundamentais do negócio. São definidos ainda os critérios de sucesso, a análise de risco, os recursos necessários e um plano para as fases seguintes (Pressman, 2006)(Scott, 2003).
- 2. Elaboração:** a preocupação central da fase de elaboração é analisar o domínio do problema, estabelecer uma arquitetura sólida para o sistema, refinar e expandir os casos de uso preliminares e monitorar e eliminar os riscos. Espera-se como resultado para esta fase: o modelo de casos de uso; com a identificação de todos os atores e descrição de suas interações, a descrição da arquitetura do *software* e um plano de desenvolvimento para todo o projeto (Sommerville, 2007).
- 3. Construção:** o objetivo dessa fase é efetuar o desenvolvimento e integração dos componentes no produto final, além de testar todas as funcionalidades existentes (Silva, 2001). Ela está essencialmente relacionada ao projeto, programação e teste de sistema. Ao término da fase de construção, o *software* deve estar em funcionamento e sua documentação associada concluída (Sommerville, 2007). De acordo com (Silva, 2001), o objetivo desta fase é disponibilizar o *software* para o usuário. Para isso

deve-se assegurar que o sistema produzido está em conformidade com os requisitos inicialmente definidos.

4. Transição: a última fase do processo unificado tem por objetivo disponibilizar o *software* para o usuário. Entretanto, para que isso aconteça, é necessário garantir que as funcionalidades identificadas tenham sido desenvolvidas e que o seu todo constitua um sistema funcional e em atendimento aos requisitos; é necessário ainda ter o sistema de *software* documentado e operando corretamente em seu ambiente operacional (Silva, 2001) (Sommerville, 2007).

3.2.2 Disciplina de Requisitos

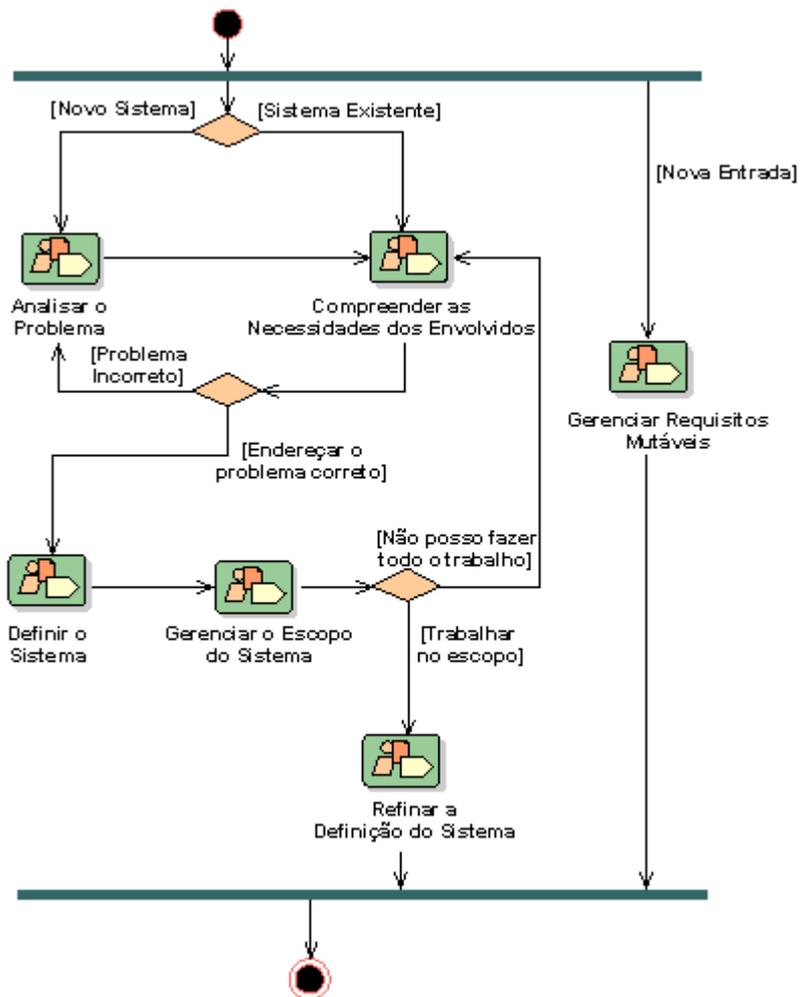
Dentro do Processo Unificado, são definidas nove disciplinas que compreendem as diferentes fases do ciclo de vida de um *software*. Cada disciplina representa um conjunto de atividades executadas por vários membros do projeto e abrange o que deve ser realizado para que um determinado conjunto de artefatos seja produzido (Rational, 2002).

Dentre as disciplinas estabelecidas pelo processo unificado temos a disciplina de Requisitos, cujas atividades tem como finalidade (Rational, 2002):

- Estabelecer e manter concordância com os clientes e outros envolvidos sobre o que o sistema deve fazer.
- Oferecer aos desenvolvedores do sistema uma compreensão melhor dos requisitos do sistema.
- Definir as fronteiras do sistema (ou delimitar o sistema).
- Fornecer uma base para planejar o conteúdo técnico das iterações.
- Fornecer uma base para estimar o custo e o tempo de desenvolvimento do sistema.
- Definir uma interface de usuário para o sistema, focando nas necessidades e metas dos usuários.

Para que estes objetivos sejam alcançados, a disciplina de requisitos estabelece um *workflow* demonstrado através da Figura 2.

Figura 2 - *Workflow* de Atividades da Disciplina de Requisitos

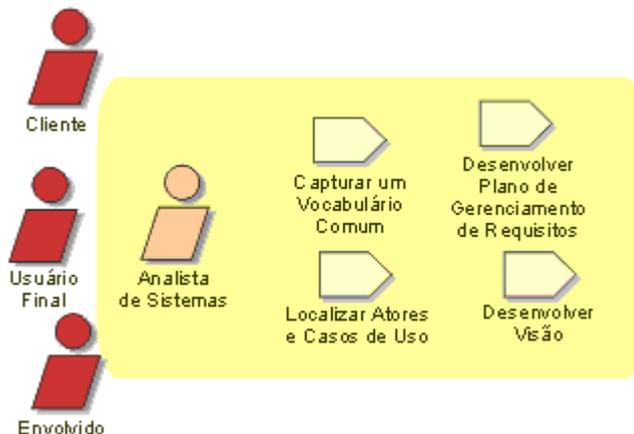


Fonte: (Rational, 2002)

“O princípio fundamental do *workflow* de Requisitos é trabalhar para construir o sistema certo” (Scott, 2003). Tal fluxo de trabalho pode ser dividido em subfluxos ou detalhamentos, cada um com os seus objetivos específicos e o conjunto de atividades que devem ser executadas para que esses sejam atingidos. As subseções 3.2.2.1 a 3.2.2.6 foram elaboradas com base no processo unificado descrito pela (Rational, 2002).

3.2.2.1 Analisar o Problema

Figura 3 - Representação do subfluxo Analisar o Problema



Fonte: adaptada de (Rational, 2002)

Os objetivos do subfluxo *Analisar o Problema* consistem em estabelecer uma visão uniforme do sistema por todos os envolvidos e uma descrição clara e precisa desta visão. Nesse momento busca-se estabelecer um acordo sobre o problema a ser resolvido, certificando-se que todas as partes envolvidas estão em comum acordo quanto ao problema a ser resolvido. Para isso, deve-se ter claro quem são os *stakeholders* e quais são os limites do sistema, ou seja, a fronteira entre o que faz parte da solução e o que está ao seu redor, as entidades externas que interagem com o sistema. Nessa etapa também devem ser identificadas as restrições impostas ao sistema, restrições essas que podem limitar a liberdade que se tem no desenvolvimento de uma solução. Essas restrições podem ser de cunho econômico, técnico, político ou ambiental, e podem estar relacionadas a prazos, recursos ou orçamento.

As atividades necessárias para alcançar esses objetivos são:

Capturar um Vocabulário Comum: definir um vocabulário comum que possa ser utilizado em todas as descrições textuais do sistema, principalmente nas descrições dos casos de uso.

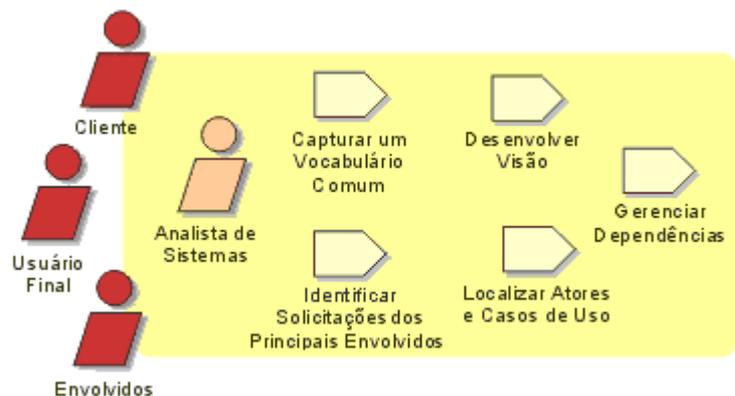
Localizar Atores e Casos de Uso: descrever a funcionalidade do sistema por meio da modelagem de casos de uso; delinear o que será tratado pelo sistema e o que será tratado fora dele; definir quem irá interagir com o sistema (atores); dividir o modelo em pacotes com atores e casos de uso, construir diagramas do modelo de casos de uso; desenvolver uma primeira descrição dos casos de uso.

Desenvolver o Plano de Gerenciamento dos Requisitos: desenvolver um plano para que os requisitos sejam documentados, com seus atributos e diretrizes de rastreabilidade e gerência dos requisitos de produto.

Desenvolver a Visão: obter um acordo sobre os problemas que serão tratados; identificar os *stakeholders*, estabelecer os limites do sistema; descrever as principais características do sistema.

3.2.2.2 Compreender as Necessidades dos Envolvidos

Figura 4 - Representação do subfluxo Compreender as Necessidades dos Envolvidos



Fonte: adaptada de (Rational, 2002)

A finalidade deste detalhamento é levantar informações de todas as partes interessadas no projeto, de forma a compreender as necessidades existentes, para com base nelas definir as características do sistema e documentá-las no documento de Visão. As necessidades dos *stakeholders* podem ser identificadas por diversos meios, como a realização de entrevistas, *brainstorming*, *workshops* de requisitos, reuniões, e outras técnicas de elicitação, e sua representação pode ser feita através de regras de negócio, requisitos de melhoria, listas de requisitos. É importante salientar que, embora esse subfluxo seja realizado principalmente nas fases de Iniciação e Elaboração, suas atividades podem ocorrer ao longo de todo o projeto, com o surgimento de novas necessidades e alterações no escopo do sistema.

As atividades necessárias para alcançar os objetivos do subfluxo *Compreender as Necessidades dos Envolvidos* são:

Capturar um Vocabulário Comum: continuar a definição de um vocabulário comum que possa ser utilizado nas descrições textuais do sistema, adicionando novos conceitos aprendidos.

Identificar Solicitações dos Principais Envolvidos: compreender quem são os *stakeholders* do projeto; elicitare os requisitos e as necessidades dos envolvidos que o sistema deverá atender; priorizar requisitos dos principais *stakeholders*.

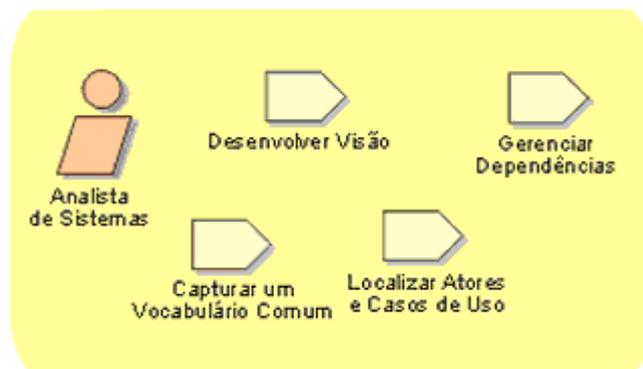
Desenvolver a Visão: complementar e refinar as principais características do sistema, anteriormente definidas no subfluxo *Analisar o Problema*.

Localizar Atores e Casos de Uso: continuar a procura por atores e casos de uso; complementar e refinar os modelos de casos de uso já criados, buscando melhorar as descrições dos atores e dos fluxos de execução; ajustar as interações entre atores e casos de uso com base na compreensão atual do sistema.

Gerenciar Dependências: gerenciar o escopo do projeto a mudança nos requisitos por meio da rastreabilidade e também com base em critérios como importância, complexidade e tamanho.

3.2.2.3 Definir o Sistema

Figura 5 - Representação do subfluxo Definir o Sistema



Fonte: adaptada de (Rational, 2002)

Os objetivos deste subfluxo consistem no alinhamento do conhecimento da equipe acerca das funcionalidades que o sistema deve contemplar. Desta forma, as atividades presentes na definição do sistema devem converter e organizar o entendimento dos envolvidos acerca do sistema a ser desenvolvido; refinar o documento de Visão com os requisitos do sistema e seus atributos; descrever o fluxo de execução de alguns casos de uso, e avaliar os resultados documentando-os. Na Definição do Sistema, os atores e casos de uso deverão ser identificados inteiramente, e os requisitos não funcionais globais expandidos. O foco principal deste detalhamento é descrever os fluxos de execução dos casos de uso.

As atividades necessárias para alcançar os objetivos desse subfluxo são:

Desenvolver a Visão: complementar e refinar as características do sistema documentadas até o momento.

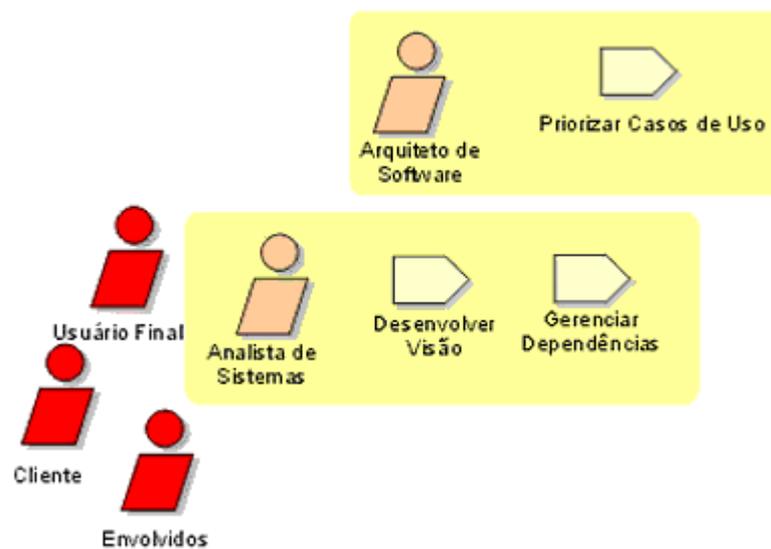
Gerenciar Dependências: da mesma forma como realizado anteriormente, o escopo do projeto e a mudança nos requisitos devem ser gerenciadas através da rastreabilidade e também por critérios de relevância dos requisitos.

Capturar um Vocabulário Comum: dar continuidade na elaboração do vocabulário ou glossário, incluindo conceitos captados no decorrer das demais atividades do projeto.

Localizar Atores e Casos de Uso: dar seguimento na procura por atores e casos de uso; refinar os modelos já criados melhorando descrições de atores e casos de uso; ajustar a interação entre estes elementos de acordo com o entendimento atual do sistema;

3.2.2.4 Gerenciar o Escopo do Sistema

Figura 6 - Representação do subfluxo Gerenciar o Escopo do Sistema



Fonte: adaptada de (Rational, 2002)

O objetivo desse detalhamento do fluxo de trabalho é assegurar que os objetivos do projeto serão alcançados. Os requisitos a serem desenvolvidos em cada iteração são escolhidos com base nas informações recebidas de todos os *stakeholders* e da priorização de acordo com seus atributos (risco, complexidade, impacto), considerando também questões como tempo de desenvolvimento, recursos e orçamento. O refinamento dos atributos e das rastreabilidades, e a melhor compressão sobre estes também são objetivos do gerenciamento do escopo do sistema.

São necessárias as seguintes atividades para atingir os objetivos desse subfluxo:

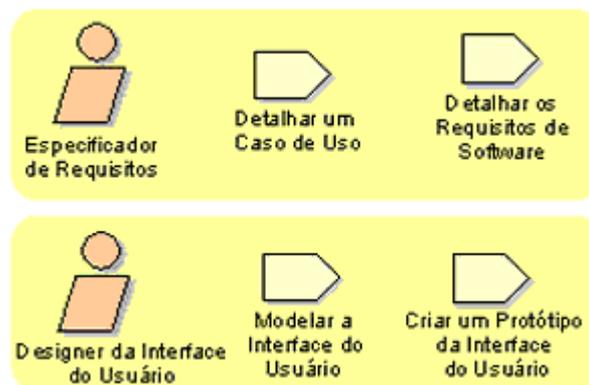
Priorizar Casos de Uso: definir os casos de uso e cenários que serão analisados e projetados a cada iteração, com base no ponto de vista de sua importância perante as funcionalidades desejadas para o sistema, no impacto arquitetural e em aspectos de restritivos, considerando também questões como a disponibilidade de pessoal, ferramentas e produtos de terceiros.

Desenvolver a Visão: refinar e complementar o documento de Visão com as características conhecidas até o momento. Deve-se salientar que mudanças realizadas fora da fase de Iniciação devem ser tratadas como requisições de mudanças.

Gerenciar Dependências: o escopo do projeto e a mudança dos requisitos devem ser gerenciados por meio da rastreabilidade e também através de atributos especiais dos requisitos, da mesma forma como em subfluxos anteriores.

3.2.2.5 Refinar a Definição do Sistema

Figura 7 - Representação do subfluxo Refinar a Definição do Sistema



Fonte: adaptada de (Rational, 2002)

Os objetivos deste subfluxo consistem em refinar a solução descrita para o sistema, descrevendo detalhadamente os casos de uso, e documentando formalmente as especificações dos requisitos não funcionais. Nesta etapa deve-se trabalhar conjuntamente aos usuários a fim de modelar e criar um protótipo da interface, que servirá para tratar a usabilidade do sistema, além de ajudar a identificar qualquer requisito que por ventura não tenha sido descoberto.

Para que esses objetivos sejam alcançados fazem necessárias as seguintes atividades:

Detalhar um Caso de Uso: aprofundar a descrição de um caso de uso previamente identificado nas etapas anteriores, dando detalhes com base na descrição resumida e no passo a passo do fluxo de eventos já existentes. Esta descrição deve deixar claro o fluxo de execução, pré e pós-condições, fluxos de erro e fluxos alternativos, além de demais

informações relativas aos caso de uso. Essa descrição servirá para averiguar se a equipe e o cliente compreendem o funcionamento proposto para o sistema.

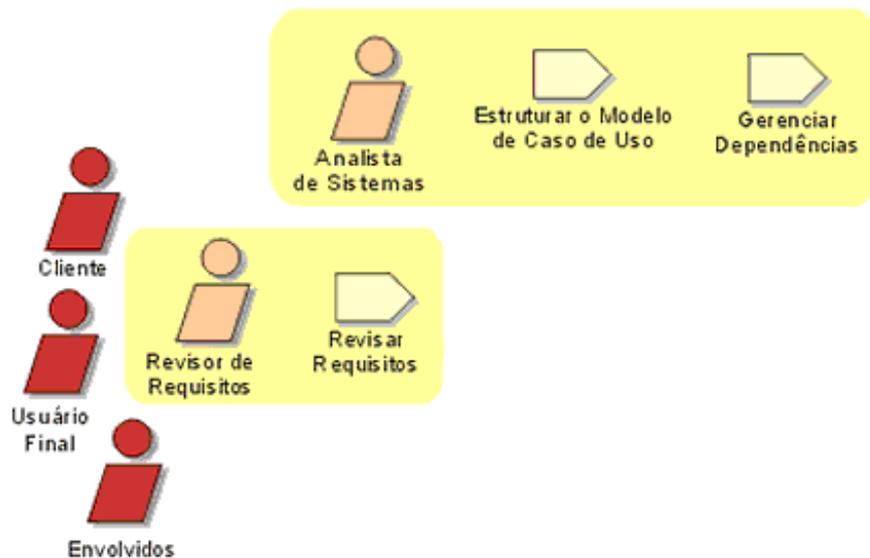
Detalhar os Requisitos de *Software*: coletar, detalhar e organizar o conjunto de artefatos que descrevem, de maneira completa, os requisitos do sistema. Dentre os principais artefatos destacam-se os atributos dos requisitos, a especificação de requisitos de *software* (listando todos os requisitos do sistema) e as especificações suplementares (listando os requisitos não funcionais).

Modelar a Interface com o Usuário: criar um esboço da interface do sistema com o usuário, considerando os atores e os fluxos de eventos dos casos de uso; capturar requisitos de usabilidade através da encenação do caso de uso; identificar classes de fronteira entre o sistema e seus usuários (requisitos para a composição da interface); descrever a interação entre essas classes e complementar os demais modelos com as novas definições.

Criar um Protótipo da Interface do Usuário: projetar e implementação um protótipo da interface do sistema com o usuário, tendo como base a modelagem realizada e os elementos da interface (janelas, botões); obter *feedback* do protótipo e avaliar os resultados.

3.2.2.6 Gerenciar Requisitos Mutáveis

Figura 8 - Representação do subfluxo Gerenciar Requisitos Mutáveis



Fonte: adaptada de (Rational, 2002)

Ao longo do ciclo de vida de um projeto, mudanças nos requisitos são esperadas. Tais mudanças podem ser fruto de novas necessidades de negócio, da melhor compreensão dos requisitos pelos *stakeholders*, e ainda de soluções para problemas identificados nas

primeiras iterações do *software*. A gerência destas mudanças deve avaliar cada solicitação de mudança recebida, definindo seu impacto para o conjunto de requisitos já existente; adaptar o modelo de casos de uso com as mudanças aceitas; definir e ajustar os atributos de rastreabilidade perante as mudanças; verificar formalmente se os resultados gerais da disciplina de Requisitos estão de acordo com a visão do cliente sobre o sistema.

Para alcançar os objetivos do gerenciamento de requisitos mutáveis, emergem as seguintes atividades:

Estruturar o Modelo de Casos de Uso: refinar o modelo de casos de uso e extrair deles os comportamentos que precisam ser considerados casos de uso abstratos (que não são vistos pelos atores), definindo suas relações de inclusão (*include*) e extensão (*extend*); procurar novos atores abstratos que definam papéis compartilhados por diversos atores.

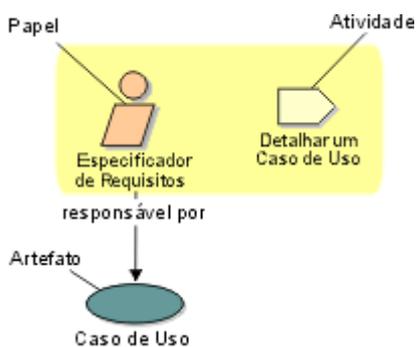
Gerenciar Dependências: da mesma maneira como em outros subfluxos, o escopo do projeto e a mudança nos requisitos devem ser gerenciadas por meio da rastreabilidade e também com base e uma série de critérios como complexidade, risco e impacto dos requisitos.

Revisar Requisitos: verificar de modo formal se os resultados gerados pela disciplina de Requisitos como um todo estão em conformidade com as expectativas do cliente; revisar o estado atual dos requisitos e as requisições de mudança submetidas.

3.2.3 Papéis da Disciplina de Requisitos

De acordo com o processo unificado definido pela (Rational, 2002), um papel é “uma definição abstrata de um conjunto de atividades executadas e dos respectivos artefatos”. Os papéis têm como função descrever o comportamento das pessoas no negócio e suas respectivas responsabilidades, podendo ser desempenhados por uma pessoa ou um grupo de pessoas trabalhando em equipe. Cabe ressaltar que um membro da equipe pode e normalmente exerce múltiplos papéis ao longo de um projeto. A figura 9 apresenta um exemplo da relação entre papel, atividade e artefato.

Figura 9 - Exemplo de relação entre Papel, Atividade e Artefato



Fonte: (Rational, 2002)

Os papéis têm associados a eles um conjunto de atividades que devem ser executadas, e estas são agrupadas de acordo com suas funcionalidades. Recomenda-se que atividades semelhantes em termos de funcionalidade sejam executadas por uma mesma pessoa. As atividades estão fortemente ligadas aos artefatos, que fornecem a entrada e a saída para as atividades.

Dentro do fluxo de trabalho da disciplina de Requisitos são elencados os seguintes papéis para as atividades:

Analista de Sistemas: responsável por coordenar a identificação de requisitos, a localização dos atores e casos de uso, e as demais atividades até a modelagem dos mesmos, delimitando o sistema e definindo suas funcionalidades. Também é geralmente o principal responsável pela elaboração do glossário.

Especificador de Requisitos: encarregado por detalhar a especificação das funcionalidades do sistema, descrevendo de maneira minuciosa o fluxo de eventos dos casos de uso de modo que o cliente e os usuários possam compreendê-los.

Designer da Interface do Usuário: responsável por coordenar a construção do protótipo e o design da interface do usuário, coletando os requisitos de interface junto ao usuário, atestando pela usabilidade do sistema, e fornecendo o *feedback* apropriado a equipe de trabalho.

Revisor de Requisitos: encarregado de planejar e conduzir a revisão formal do modelo de casos de uso, e validar se o trabalho resultante das atividades da disciplina de Requisitos está em conformidade com às expectativas do cliente.

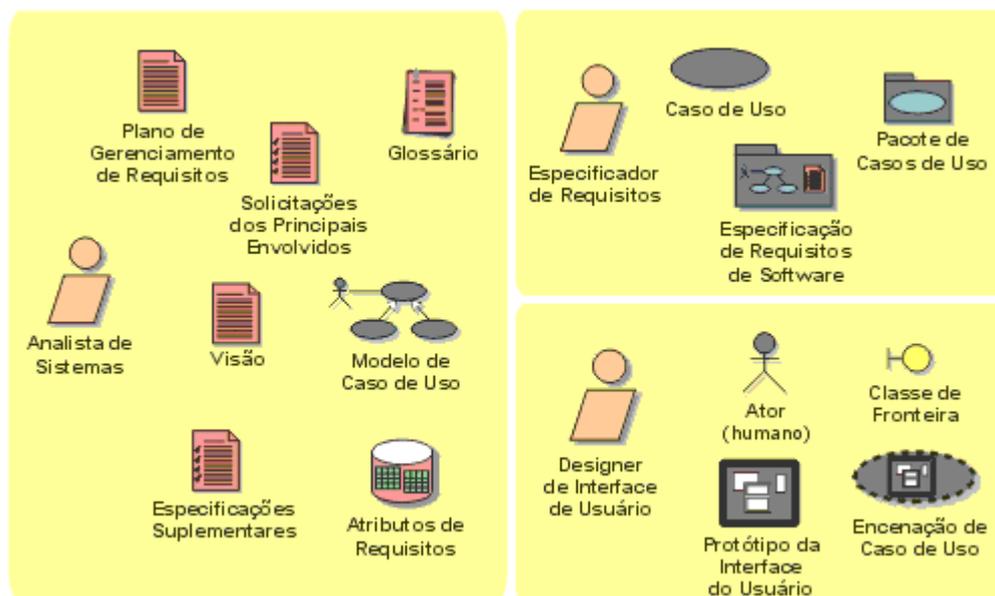
Arquiteto de *Software*: responsável pela priorização dos casos de uso, definindo quais serão os casos de uso e cenários analisados e projetados a cada iteração, tendo como base suas funcionalidades, o impacto arquitetural, aspectos restritivos e também questões como a disponibilidade de pessoal, ferramentas e produtos de terceiros.

3.2.4 Artefatos da Disciplina de Requisitos

(Rational, 2002) define artefatos como sendo “produtos de trabalhos finais ou intermediários produzidos e usados durante os projetos”, sendo utilizados para capturar e transmitir informações no decorrer do mesmo.

A disciplina de Requisitos apresenta diversos artefatos ao longo do seu fluxo de trabalho. Segundo (Larman, 2005), assim como todos os artefatos do processo unificado, eles são opcionais. Este conjunto de artefatos é ilustrado através da figura 10.

Figura 10 - Papéis e Artefatos construídos na Disciplina de Requisitos



Fonte: (Rational, 2002)

(Larman, 2007) define alguns dos artefatos como artefatos-chave para o processo de requisitos:

Modelo de Casos de Uso: apresenta as funções pretendidas para o sistema através de cenários típicos de utilização do mesmo, e serve como um contrato entre o cliente e os desenvolvedores. Abrange principalmente os requisitos funcionais. (Larman, 2007) (Rational, 2002), sendo utilizado como fonte de informações para as atividades de análise, design e teste (Rational, 2002).

Especificações Suplementares: contemplam basicamente tudo o que não está nos casos de uso (Larman, 2007). Destinam-se principalmente para requisitos não funcionais, que tratam questões como *performance*, *segurança*, *backup* ou ainda restrições vindas de fora do sistema (Scott, 2003).

Glossário: registra o vocabulário comum do projeto, na linguagem do cliente, e visa fornecer um conjunto consistente de definições que evitem diferentes interpretações de um mesmo termo. É criado na fase de iniciação do projeto, e evolui gradativamente com ao longo do seu desenvolvimento, servindo como referência a todos participantes (Rational, 2002).

Visão: define uma visão geral das ideias que os *stakeholders* têm perante o produto a ser desenvolvido, trazendo um resumo de alto nível dos requisitos e do modelo de negócio. A descrição dos requisitos contida no documento de Visão serve como base para o detalhamento dos mesmos no modelo de casos de uso e nas especificações suplementares (Larman, 2007) (Rational, 2002).

Regras de Negócio: (Larman, 2007) sugere a utilização deste artefato (também chamadas de regras de domínio) para descrever requisitos ou políticas que influenciem diretamente no produto a ser desenvolvido, e que vão além do projeto de *software*. É citado como exemplo o conjunto de leis sobre impostos governamentais, que impactam diretamente no contexto de negócio de um sistema financeiro. Ainda segundo (Larman, 2007), tais regras de negócio podem ser registradas através do artefato de Especificações Suplementares, embora o mesmo recomende a distinção destas regras por meio do artefato sugerido.

Uma série de outros artefatos também pode ser gerada ao longo do *workflow* de requisitos. Abaixo são listados alguns desses demais artefatos:

Modelo de Domínio: este artefato construído na etapa de iniciação, captura os principais conceitos e entidades acerca do mundo real no qual o problema está inserido. Utiliza-se o diagrama de classes da UML como elemento para representar o conteúdo deste modelo, sempre em alto nível, demonstrando basicamente o número de classes e os relacionamentos de maior relevância para o processo (Scott, 2003).

Solicitações dos Principais Envolvidos: tem como objetivo capturar toda e qualquer solicitação recebida em relação ao projeto oriunda dos principais *stakeholders*. Embora seja de responsabilidade do analista de sistemas, várias pessoas podem contribuir com este artefato (Rational, 2002).

Protótipo da Interface do Usuário: apresenta um modelo da interface do sistema com o usuário, podendo ser criado através de esboços de tela, desenhos à mão livre, protótipos estáticos ou funcionais do sistema. Auxiliam os desenvolvedores a moldar a forma de interação do sistema com o usuário (Scott, 2003).

Atributos de Requisitos: este artefato contempla um repositório de dependências, atributos e requisitos, que são recolhidos ao longo de todo o processo de análise de requisitos.

Serve como base para as atividades de gerenciamento dos requisitos, e deve ser acessível a todos os participantes incluídos nas etapas de desenvolvimento (Rational, 2002).

3.3 MODELO DE CASOS DE USO

O modelo de casos de uso é um dos principais, se não o principal artefato construído pelo *workflow* de Requisitos. Nele estão contidas as funções pretendidas para o sistema, servindo como uma espécie de contrato entre o cliente e os desenvolvedores, e como mecanismo de comunicação do comportamento do sistema a todos os envolvidos (Rational, 2002).

Na sua essência, casos de uso possuem uma forma textual, e contam uma história de como um usuário interage com o sistema sob um conjunto de circunstâncias específicas para atingir um objetivo (Pressman, 2006). Outras maneiras de representação para os casos de uso podem ser adotadas, como fluxogramas, diagramas de sequência, redes de Petri ou linguagens de programação, entretanto, devido aos casos de uso servirem como meio de comunicação entre pessoas, e muitas vezes estas pessoas não possuem o conhecimento necessário para poder interpretar tais representações, geralmente é aconselhada a utilização da forma textual para sua representação (Cockburn, 2005). A figura 11 apresenta um exemplo de caso de uso no formato resumido, que normalmente é formado por apenas um parágrafo representando o cenário de sucesso principal (Larman, 2007).

Figura 11 - Caso de Uso resumido: “Processar Venda”

| |
|--|
| <p>Processar Venda: um cliente chega em um ponto de pagamento com itens que deseja adquirir. O caixa usa o sistema para registrar cada item comprado. O sistema vai apresentando um total parcial e uma linha de detalhes à medida que registra cada item. O cliente entra os dados sobre o pagamento, que são validados e, em seguida, registrados pelo sistema. O sistema atualiza o estoque. O cliente recebe um recibo do sistema e sai com os itens comprados.</p> |
|--|

Fonte: (Larman, 2007)

A essência dos casos de uso consiste em descobrir e registrar os requisitos funcionais de um sistema. Normalmente, um caso de uso necessita ser melhor elaborado do que no formato resumido (Larman, 2007).

No formato informal, os casos de uso apresentam uma série de cenários relacionados de sucesso e fracasso, com suas sequências de ações e interações entre atores e o sistema. Esse modelo visa descrever portanto, um ator (identificado por seu papel) utilizando o sistema

como meio de atingir um objetivo em específico, apresentando o cenário de sucesso desejado, e os cenários alternativos possíveis (Larman, 2007). A figura 12 descreve um exemplo de caso de uso informal.

Figura 12 - Caso de Uso informal: “Tratar Devoluções”

Tratar Devoluções

Cenário de sucesso principal: um cliente chega a um poso de pagamento com itens a serem devolvidos. O caixa usa o sistema PDV para registrar cada item devolvido...

Cenários alternativos:

Se o cliente pagou a crédito e a transação de reembolso para estorno em sua conta de crédito é rejeitada, informa o cliente e o reembolse com dinheiro.

Se o identificador do item não for encontrado no sistema, este notifica o caixa e sugere que entre manualmente o código do produto (talvez ele esteja corrompido).

Se o sistema detecta uma falha para se comunicar com o sistema externo de contabilidade...

Fonte: (Larman, 2005)

De acordo com (Larman, 2007), além dos formatos resumido e informal, os casos de uso podem ser escritos em um terceiro formato, o completo, para casos de uso que necessitam ser melhor detalhados.

Os casos de uso no formato completo permitem uma compreensão aprofundada dos objetivos, tarefas e requisitos (Larman, 2007). Neste modelo, todos os passos e variantes são escritos de forma detalhada e estruturada, contando ainda com seções de suporte, como pré-condições e garantias de sucesso (Cockburn, 2007).

Para que um bom caso de uso seja escrito e possa ser lido com facilidade pelos interessados, faz-se necessária a compreensão das partes que o compõem. (Cockburn, 2005), aconselha em seu livro *Escrevendo Casos de Uso Eficazes* a utilização do formato completo, e apresenta as definições resumidas das partes e das características básicas que o compõem por meio de um modelo. Este mesmo modelo também é referenciado por (Larman, 2007), que por sua vez estabelece um gabarito para casos de uso completos baseados no mesmo. A figura 13 apresenta o modelo para construção de casos de uso completos proposto por (Cockburn, 2005), enquanto a figura 14 exemplifica a utilização deste.

Figura 13 - Modelo para casos de uso completos de uma coluna

| |
|---|
| <p><o nome deveria ser o objetivo como uma pequena frase de verbo ativo> Contexto de Uso: <uma sentença maior do objetivo, se necessária, suas condições de ocorrência normal> Escopo: <escopo de desenvolvimento, qual sistema está sendo considerado caixa-preta sob desenvolvimento> Nível: <um dentre estes: resumo, objetivo do usuário, subfunção> Ator Primário: <um nome de papel para o ator primário ou descrição> Stakeholders e Interesses: <lista de <i>stakeholders</i> e interesses chaves no caso de uso> Pré-condição: <o que esperamos que já seja o estado do mundo> Garantias Mínimas: <como os interesses são protegidos sob todas saídas> Garantias de Sucesso: <o estado do mundo se o objetivo tem sucesso> Acionador: <o que inicia o caso de uso, pode ser evento de tempo> Cenário de Sucesso Principal: <coloque aqui os passos do cenário do acionamento a entrega do objetivo e qualquer limpeza posterior> <# do passo><descrição da ação> Extensões: <coloque aqui as extensões, uma por vez, cada uma referindo-se ao passo do cenário principal> <passo alterado><condição>: <ação do sub caso de uso> <passo alterado><condição>: <ação do sub caso de uso> Lista de Variações Tecnológicas & de Dados: <coloque aqui as variações que causarão eventuais bifurcações no cenários> <# do passo ou variação> <lista de variações> <# do passo ou variação> <lista de variações> Informação Relacionada: <tudo que seu projeto precisa de informação adicional></p> |
|---|

Fonte: (Cockburn, 2005)

Figura 14 - Caso de Uso “inteiramente” completo de uma coluna

| |
|---|
| <p>Ator Primário: Requisitante Objetivo no Contexto: Requisitante compra algo através do sistema, recebe-o. Não inclui pagamento por ele Escopo: Negócio – o mecanismo geral de compra, eletrônico e não eletrônico, como visto pelas pessoas na companhia Nível: Resumo Stakeholders e Interesses: Requisitante: Deseja o que pediu, uma maneira fácil de fazer isso. Companhia: Deseja controlar gastos, mas permite as compras necessárias. Fornecedor: Deseja receber por qualquer mercadoria entregue. Pré-condição: nenhuma Garantia Mínima: toda ordem enviada foi aprovada por um autorizador válido. Ordem foi rastreada de forma que a companhia só pode ser faturada para mercadorias válidas recebidas. Garantia de Sucesso: Requisitante com mercadorias, orçamento correto pronto para ser debitado.. Acionador: Requisitante decide comprar algo. Cenário de Sucesso Principal: 1. <i>Requisitante:</i> <u>inicia o pedido.</u> 2. <i>Aprovador:</i> verifica dinheiro no orçamento, verifica preço das mercadorias, <u>completa pedido para submissão.</u> 3. <i>Comprador:</i> verifica o conteúdo do depósito, encontra o melhor fornecedor para as mercadorias. 4. <i>Autorizador:</i> <u>valida assinatura do Aprovador.</u> 5. <i>Comprador:</i> <u>completa o pedido para ordem, inicia a OC com o Fornecedor.</u> 6. <i>Fornecedor:</i> entrega as mercadorias para a Recepção, pega o recibo pela entrega (fora do escopo do sistema em desenvolvimento).</p> |
|---|

7. *Receptor*: registra a entrega; envia as mercadorias para o Requisitante.

8. *Requisitante*: marca pedido entregue.

Extensões:

1a. Requisitante não conhece fornecedor nem preço: Deixar essas partes em branco e continuar.

1b. A qualquer momento, antes do recebimento das mercadorias, o Requisitante pode mudar ou cancelar o pedido:

Cancelando-o, remove-o do processo ativo (Excluí-lo do sistema?).

Reduzindo o preço, envia-o de volta ao Aprovador.

2a. Aprovador não conhece fornecedor nem preço: Deixar em branco e deixar o Comprador preencher ou retornar.

2b. Aprovador não é gerente do Requisitante: Permanece OK, contanto que o Aprovador assinie.

2c. Aprovador recusa: Enviar de volta ao Requisitante para alteração ou exclusão.

3a. Comprador encontra mercadorias no depósito: Pegá-las, diminuir a quantidade do pedido, e continuar.

3b. Comprador preenche o Fornecedor e preço, que estavam faltando: Pedido é reenviado ao Aprovador.

4a. Autorizador recusa Aprovador: Enviar de volta ao Requisitante e remover do processo ativo. (O que isto significa?)

5a. Pedido envolve múltiplos Fornecedores: Comprador gera múltiplas OCs.

5b. Comprador funde vários pedidos: Mesmo processo, mas marca OC com os pedidos fundidos.

6a. Fornecedor não entrega a tempo: Sistema alerta a não entrega.

7a. Entrega parcial: Receptor marca entrega parcial na OC e continua.

7b. Entrega parcial de múltiplos pedidos OC: Receptor atribui quantidades aos pedidos e continua.

8a. Mercadorias são incorretas ou de má qualidade: Requisitante recusa mercadorias entregues. (O que isto significa?)

8b. Requisitante deixou a companhia: Comprador verifica com o gerente do Requisitante: um dos dois designa novo Requisitante ou devolve as mercadorias e cancela o pedido.

Lista de Variações Tecnológicas e de Dados: Nenhuma

Prioridade: Diversa

Versões: Várias

Tempo de Resposta: Diverso

Frequência de Uso: 3/dia

Canal para Ator Primário: Navegador de Internet, sistema de correio eletrônico ou equivalente

Atores secundários: Fornecedor

Canal para Atores Secundários: Fax, telefone, carro

Aspectos abertos:

Quando um pedido cancelado é excluído do sistema?

Que autorização é necessária para cancelar um pedido?

Quem pode alterar o conteúdo de um pedido?

Que histórico de alteração deve ser mantido nos pedidos?

O que acontece quando o Requisitante recusa a mercadoria entregue?

Como uma requisição funciona de maneira diferente de uma ordem?

Como o pedido faz referência e uso do armazenamento interno?

Fonte: (Cockburn, 2005)

3.3.1 Principais Elementos de um Caso de Uso Completo

Nos subcapítulos a seguir são explicados e exemplificados os principais elementos que compõem um caso de uso no formato completo.

3.3.1.1 Ator Primário

O ator primário ou principal de um caso de uso é o *stakeholder* que chama os serviços do sistema para atingir um objetivo (Larman, 2007). Esse objetivo com relação ao sistema pode ser ou não satisfeito por meio de sua operação (Cockburn, 2005).

Geralmente, um caso de uso tem início com a ação de um ator primário, entretanto, o mesmo pode ser iniciado por meio de outra pessoa na função do ator primário, ou ainda por um evento programado pelo próprio sistema (Cockburn, 2005).

É importante ter conhecido os atores primários no início da produção dos casos de uso, estes auxiliam na compreensão dos objetivos e permitem o particionamento de um grande conjunto de casos de uso em pacotes, que podem ser atribuídos à equipes de desenvolvimento diferentes.

3.3.1.2 Nível

Os níveis de objetivos servem para classificar os casos de uso, (Larman, 2007) citando o sistema de níveis de objetivos estabelecido por (Cockburn, 2005), apresenta a classificação dos casos de uso como estando no nível de objetivo de usuário, sendo este mais comum; que descreve os cenários para que os objetivos de um ator principal sejam atingidos, ou no nível de subfunção; que descreve os subpassos necessários para apoiar um objetivo de usuário. Além destes, (Cockburn, 2005) expõe ainda o nível de resumo, quando múltiplos objetivos do usuário são envolvidos, em nível mais alto, não tão específico. Dentre estes, (Cockburn, 2005) enfatiza a necessidade de voltar esforços na identificação dos objetivos de nível de usuário, sendo esses os de maior importância para o sistema.

3.3.1.3 Stakeholders e Interesses

Um *stakeholder* é algo ou alguém com interesse no comportamento do caso de uso. Naturalmente, todo ator primário é também um *stakeholder*, mas os casos de uso podem contar com outros interessados, sem que estes interajam diretamente com o sistema. Como exemplo, podemos citar os sócios da empresa para qual o sistema está sendo desenvolvido, seus gerentes, diretores, auditores de qualidade, entre outros (Cockburn, 2005).

Nessa seção devem ser listados os *stakeholders* e seus interesses junto ao caso de uso. Essas informações são de extrema importância para determinar o que o caso de uso deve

conter. Os comportamentos do sistema que não estão relacionados com a satisfação dos *stakeholders* não devem ser tratados pelo caso de uso (Larman, 2007). A figura 15 apresenta os *stakeholders* e seus respectivos interesses para o caso de uso Processar Venda.

Figura 15 - *Stakeholders* e Interesses para o Caso de Uso: “Processar Venda”

Stakeholders e Interesses:

Caixa: deseja entrada rápida, precisa e sem erros de pagamento, tal como falta de dinheiro na gaveta do caixa, pois esta será deduzida do seu salário.

Vendedor: deseja comissões sobre vendas atualizadas.

...

Fonte: (Larman, 2007)

3.3.1.4 Pré-Condições

“Pré-condições declaram o que deve sempre ser verdadeiro antes de iniciar um cenário do caso de uso” (Larman, 2007). De modo geral, uma pré-condição é estabelecida mediante a execução com sucesso de outro caso de uso. Ao descrever pré-condições, entendeu-se que essas serão verdadeiras, não havendo preocupação em testá-las. Elas comunicam pressupostos importantes e subentendeu-se que os leitores já as conheçam (Larman, 2007). Um erro comum na construção de casos de uso é descrever pré-condições que nem sempre serão verdadeiras, o que pode vir a impactar diretamente na execução dos cenários (Cockburn, 2005). Na figura 16 são apresentados exemplos da seção pré-condição para diferentes casos de uso.

Figura 16 – Exemplos da seção Pré-Condição para diferentes Casos de Uso

Pré-condição: O usuário fez login no sistema.

Pré-condição: O cliente foi validado.

Pré-condição: O sistema já localizou a informação da apólice do cliente.

Fonte: (Cockburn, 2005)

3.3.1.5 Garantias Mínimas

As garantias mínimas são de interesse quando o cenário de sucesso não é alcançado. Referem-se as menores promessas feitas pelo sistema aos *stakeholders*, ou seja o mínimo que se espera do sistema caso o objetivo do ator principal não seja atingido. Elas são escritas como um conjunto de simples asserções que ao final da execução do caso de uso serão verdadeiras, independente do fluxo seguido. É importante que os *stakeholders* estejam de acordo com as garantias mínimas oferecidas (Cockburn, 2005). A figura 17 trás exemplos da seção de garantias mínimas para diferentes casos de uso.

Figura 17 – Exemplos da seção Garantias Mínimas para diferentes Casos de Uso

Garantias Mínimas: Pedido será iniciado somente se pagamento for recebido.

Garantias Mínimas: Se a informação mínima não foi capturada, a reivindicação parcial foi descartada e nenhum registro da chamada foi feito. Se a informação mínima foi capturada (veja Regras De Negócio), então a reivindicação parcial foi salva e registrada.

Fonte: (Cockburn, 2005)

3.3.1.6 *Garantias de Sucesso*

As garantidas de sucesso ou pós-condições estabelecem o que deve ser verdadeiro quando da conclusão com êxito do caso de uso, seja ao término do cenário de sucesso principal ou ao término de algum outro caminho de sucesso alternativo (Larman, 2007). Assim como as garantias mínimas, as garantias de sucesso são escritas são escritas comum conjunto de asserções que se aplicam ao final de uma bem-sucedida execução do caso de uso e visa atender às necessidades de todos os *stakeholders* (Cockburn, 2005). Exemplos de garantias de sucesso para diferentes casos de uso podem ser vistos na figura 18.

Figura 18 – Exemplo da seção Garantia de Sucesso para diferentes Casos de Uso

Garantia de Sucesso: O Requerente receberá a quantia combinada, a reivindicação é fechada, o pagamento é registrado.

Garantia de Sucesso: O arquivo será salvo.

Garantia de Sucesso: O sistema iniciará uma ordem para o cliente, terá recebido informação de pagamento e registrado o pedido pela ordem.

Fonte: (Cockburn, 2005)

3.3.1.7 *Acionadores*

O acionador serve como gatilho de disparo do caso de uso. Ele especifica o evento que faz o caso de uso começar. O acionador pode anteceder o primeiro passo de um caso de uso, ou às vezes o próprio acionador por ser o primeiro passo, não existindo uma regra específica (Cockburn, 2005). A figura 19 representa um exemplo de acionador para o caso de uso “Processar Venda”.

Figura 19 - Acionador para o Caso de Uso “Processar Venda”

Acionador: cliente insere cartão.

1. Cliente insere cartão com ID do banco, conta bancária e senha
2. Sistema valida...

Fonte: (Cockburn, 2005)

3.3.1.8 Cenário de Sucesso Principal

Este cenário também chamado por (Larman, 2007) de cenário “caminho feliz”, descreve o caminho típico de sucesso que satisfaz o objetivo dos *stakeholders* (Larman, 2007). Este cenário nada mais é que uma explanação a respeito dos passos necessários para se chegar ao resultado desejado.

No cenário de sucesso principal, elabora-se uma descrição de um cenário de fácil compreensão e bastante característico, no qual a sequência de ações a serem seguidas são enumeradas de cima para baixo, onde ao seu término o objetivo do ator primário é atingido e todos os interesses dos *stakeholders* são satisfeitos. Todas as demais maneiras do caso de uso ter sucesso, bem como o tratamento de todas as falhas, são descritos nas extensões do cenário (Cockburn, 2005).

O corpo de um cenário pode registrar três tipos de passos (Cockburn, 2005)(Larman, 2007):

- Uma interação entre atores (“Cliente entra com endereço”).
- Uma validação, geralmente feita pelo sistema (“Sistema valida a senha”).
- Uma mudança interna (“Sistema deduz quantia de saldo”).

A figura 20 apresenta o cenário de sucesso principal para o caso de uso “Registrar Perda”.

Figura 20 - Cenário de Sucesso Principal para o Caso de Uso "Registrar Perda"

Cenário de Sucesso Principal:

1. Atendente entra com o número da apólice do segurado ou então com nome e data do incidente.
 2. O sistema preenche informação disponível de apólice e indica que a reivindicação corresponde à apólice.
 3. Atendente entra com informação básica de perda.
 4. Sistema confirma que não há reivindicações que competem e atribui um número de reivindicação.
 5. Atendente continua entrando com informação específica de perda para linha de reivindicação
 6. Atendente tem Sistema extraindo outra informação de cobertura de outros sistemas computacionais.
 7. Atendente seleciona e designa um Organizador.
 8. Atendente confirma que ele terminou.
- Sistema salva e dispara (gatilho) confirmação para ser enviada ao Agente.

Fonte: (Cockburn, 2005)

3.3.1.9 Extensões

As extensões descrevem os pontos onde o comportamento do cenário de sucesso principal pode desviar do seu fluxo de execução por causa de uma condição em particular (Cockburn, 2005). Indicam todos os outros cenários de sucesso e fracasso possíveis para um caso de uso (Larman, 2007).

Uma extensão é uma ramificação de um passo do cenário de sucesso principal. Ela começa com uma condição, e apresenta uma sequência de passos de ação que descrevem o que pode acontecer sob tal condição, e termina com a realização ou o abandono do objetivo da extensão (Cockburn, 2005). Extensões são divididas em duas partes, condição e tratamento.

Condições de extensão são as condições sob as quais o sistema apresenta um comportamento diferente do esperado (Cockburn, 2005). Estas devem ser retratadas como algo que possa ser detectado pelo sistema ou por um ator (Larman, 2007).

Tratamento de extensões, de um modo simples, representa uma sequência básica de passos que lidam com a condição apresentada. Uma extensão pode ser vista ainda como um caso de uso miniatura, cujo objetivo é completar a execução com sucesso do caso de uso ou então recuperá-lo de qualquer falha que possa ocorrer. Uma extensão pode terminar com a realização ou o abandono de seu objetivo, da mesma forma que em um caso de uso. Ao fim do tratamento de uma extensão deve ser alcançado um ponto que permita o reingresso ao cenário de sucesso principal, ou então um ponto que indique a falha do caso de uso (Cockburn, 2005)(Larman, 2007).

Como forma de identificar as possíveis maneiras que o cenário pode falhar e os cursos alternativos em que o mesmo pode ter sucesso, (Cockburn, 2005) aconselha a realização de exaustivos *brainstorming* junto aos *stakeholders*.

Na figura 21 pode ser observado o cenário de sucesso principal e suas extensões para o caso de uso “Verificar Ortografia”.

Figura 21 - Cenário de Sucesso Principal e Extensões para o Caso de Uso “Verificar Ortografia”

| |
|---|
| <p>Cenário de Sucesso Principal:</p> <p>...</p> <p>3. O sistema percorre o documento, verificando cada palavra com seu dicionário de ortografia.</p> <p>4. O sistema detecta um erro ortográfico, destaca a palavra e apresenta alternativas para o usuário.</p> <p>5. O usuário seleciona uma das opções para substituição. O sistema substitui a palavra destacada com a opção de substituição do usuário.</p> <p>...</p> <p>Extensões:</p> <p>...</p> <p>4a. O sistema não detecta outro erro de ortografia até o final do documento:</p> <p>4a1. O sistema notifica o usuário, termina o caso de uso.</p> <p>5a. Usuário escolhe manter a ortografia original:</p> <p>5a1. O sistema ignora a palavra e continua.</p> <p>5b. Usuário digita uma nova ortografia que não está na lista:</p> <p>5b1. O sistema revalida a nova ortografia, volta ao passo 3.</p> <p>...</p> |
|---|

Fonte: (Cockburn, 2005)

3.3.1.10 Lista de Variáveis Tecnológicas e de Dados

As extensões de um caso de uso expressam o que o sistema é capaz de fazer. Entretanto, podem existir diferentes maneiras de se fazer algo dentro do sistema. Tais variações não podem ser consideradas extensões de algum caso de uso, pois este estaria em um nível baixíssimo, o que impossibilitaria sua escrita. Para isso, essas variantes técnicas não sobre *o que*, mas sim sobre *como* deve ser feito, são descritas na lista de variações tecnológicas & de dados. Em seu livro, (Cockburn, 2005) apresenta um exemplo onde o sistema deve fornecer crédito a um cliente pelas mercadorias devolvidas, podendo essa devolução ocorrer através de cheque, transferência eletrônica ou por um crédito sobre a próxima compra. Para este caso, seria adicionada uma restrição à lista de variações tecnológicas e de dados, conforme demonstrado na figura 22.

Figura 22 - Lista de Variantes Tecnológicas e de Dados para o Caso de Uso “Reembolsar Cliente”

| |
|---|
| <p>Cenário de Sucesso Principal:</p> <p>...</p> <p>2. Reembolsar cliente pelas mercadorias devolvidas.</p> <p>...</p> <p>Lista de Variantes Tecnológicas e de Dados:</p> <p>2a. Reembolsar por cheque, TED, ou crédito sobre futuras compras.</p> |
|---|

Fonte: (Cockburn, 2005)

3.3.2 Narrativas de Uso

Uma narrativa de uso nada mais é que um exemplo de um caso de uso em ação, apresentando de forma específica a utilização do sistema por um ator. Uma narrativa não chega a ser considerada um caso de uso, entretanto, pode servir como um dispositivo útil para antevê-lo (Cockburn, 2005).

(Cockburn, 2005) aconselha a utilização de narrativas de uso nas etapas iniciais de um projeto, como forma de pessoas com pouca experiência na escrita de casos de uso terem um melhor entendimento do negócio para assim poder detalhá-lo de forma correta e concisa.

Redigir uma narrativa não deve levar muito tempo. Uma narrativa requer um ator, que pode ser fictício, mas específico. Deve-se captar o estado mental desse ator – por que ele quer, o que ele quer, ou quais condições o levam a agir como age, captando a ação do início ao fim, de modo que o leitor possa ser conduzido facilmente no próprio caso de uso. De acordo com (Cockburn, 2005), “*a narrativa não é requisito; particularmente, ela prepara o terreno para descrições mais detalhadas e generalizadas dos requisitos*”. Na figura 23 abaixo é apresentado um exemplo de narrativa de uso.

Figura 23 - Narrativa de Uso: “Pegando Dinheiro Rápido”

| |
|--|
| <p>Maria, levando suas duas filhas à creche no caminho do trabalho, dirige até o caixa eletrônico, passa seu cartão no leitor de cartões, digita sua senha, seleciona Dinheiro Rápido e entra com a quantidade de R\$ 35. O caixa eletrônico libera uma nota de R\$ 20 e três de R\$ 5, e mais um recibo mostrando seu saldo da conta depois que os R\$ 35 foram debitados. O terminal reseta sua tela depois de cada transação de Dinheiro Rápido, assim Maria pode sair e não se preocupar que o próximo motorista terá acesso a sua conta. Maria gosta do Dinheiro Rápido porque ele evita muitas questões que deixam a interação lenta. Ela veio a este caixa eletrônico em particular porque ele tem notas de R\$ 5, que ela usa para pagar a mantenedora da creche, e ela não precisa sair do seu carro para usá-lo.</p> |
|--|

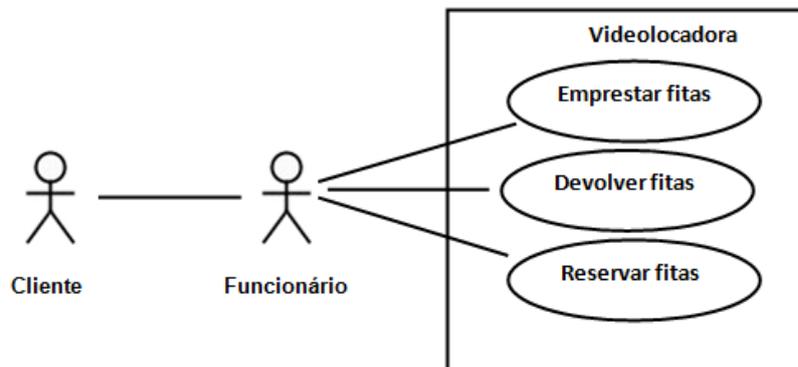
Fonte: (Cockburn, 2005)

3.3.3 Diagrama de Casos de Uso

De acordo com (Silva, 2001), um diagrama de caso de uso descreve a relação entre atores e casos de uso de um sistema. Isso permite dar uma visão global e de alto nível do sistema, descrevendo sua visão externa e as interações com o mundo exterior. Estes diagramas são utilizados preferencialmente na fase de especificação de requisitos, podendo ser utilizados também durante a modelagem dos processos de negócio (Silva, 2001).

Na figura 24 pode-se visualizar um exemplo de diagrama de caso de uso para um sistema de videolocadora. As elipses representam casos de uso, os bonecos representam atores e o retângulo representa a fronteira do sistema (Wazlawick, 2004).

Figura 24 – Diagrama de Casos de Uso: “Videolocadora”



Fonte: (Wazlawick, 2004).

3.4 CONSIDERAÇÕES FINAIS

Neste capítulo foram apresentados os referenciais teóricos a respeito da engenharia de requisitos, disciplina de requisitos dentro do processo unificado e também sobre o modelo de casos de uso. O objetivo deste estudo é possibilitar a melhor compreensão dos conceitos que envolvem o processo de requisitos para assim subsidiar a definição de uma proposta de modelo de processo de engenharia de requisitos para o NUSIS.

Este estudo permitiu reconhecer a importância das atividades da engenharia de requisitos dentro de um processo de *software*. É preciso buscar o completo entendimento dos requisitos envolvidos em um projeto antes que a etapa de construção tenha início. Este entendimento acerca do problema e das necessidades dos *stakeholders* do projeto é justamente o maior desafio da engenharia de requisitos.

Não existe um processo único que possa ser utilizado por todas as empresas que desenvolvem *software*. Cada organização deve estipular seu próprio processo, visando a adequação do mesmo em consideração aos tipos de sistema em desenvolvimento, cultura organizacional, experiência dos envolvidos, entre outros fatores relevantes.

A engenharia de requisitos é abordada por autores como (Pressman, 2006) e (Sommerville, 2011) com uma representação genérica, onde de forma abstrata pode-se dividir o processo em etapas não necessariamente sequenciais. (Pressman, 2006) divide a engenharia de requisitos em fases: concepção, levantamento, elaboração, negociação, especificação, validação e gestão. Em cada etapa são apresentadas técnicas para as atividades desencadeadas, podendo estas variar de acordo com o cenário a qual se destinam. Ao longo das atividades também surgem artefatos, que são criados como maneira de formalizar e documentar os resultados das tarefas realizadas.

No processo unificado, a disciplina de requisitos estabelece um fluxo de atividades, que vai desde a análise do problema até o gerenciamento das mudanças nos requisitos ao longo do ciclo de vida de um *software*. Fica evidente a preocupação do processo unificado com a geração de evidências ao longo das atividades. Relacionado a isso uma série de artefatos é criada. É notória também a preocupação do processo unificado com o estabelecimento de papéis e a atribuição de responsabilidades a estes ao longo das atividades.

Todo o estudo realizado neste capítulo, seja nas fases da engenharia de requisito ou no *workflow* da disciplina de requisitos do processo unificado, trouxe a construção do modelo de casos de uso como premissa básica de sucesso não somente para a tarefa de especificação dos requisitos funcionais, mas para o processo de engenharia de requisitos como um todo. A modelagem dos requisitos através de casos de uso é uma das principais características do processo unificado. Pode-se dizer que neste processo os casos de uso são responsáveis por direcionar a arquitetura do sistema, que por sua vez influencia diretamente na seleção dos casos de uso.

O modelo de casos de uso é tido como o principal artefato produzido na engenharia de requisitos. Os casos de uso possuem sua essência na forma textual, e consistem em descobrir e registrar os requisitos funcionais de um sistema. Foram estudados três formatos para construção dos casos de uso: resumido, informal e completo.

No formato completo todos os passos e variantes são escritos de forma detalhada e estruturada, permitindo uma maior compreensão dos objetivos, tarefas e requisitos (Larman, 2007). No entanto, para que um caso de uso neste formato possa ser escrito e entendido com

facilidade, é necessário conhecer as partes que o compõem. Para isso, buscou-se aprofundar o estudo das principais seções contempladas em um caso de uso completo.

No capítulo 4 é apresentada uma proposta para o processo de engenharia de requisitos do NUSIS elaborada com base na contextualização do ambiente de desenvolvimento e nos atributos e características de qualidade desejáveis. Tal proposta é apoiada pelos estudos deste capítulo.

Este capítulo descreve um modelo de processo de engenharia de requisitos, elaborado em consideração às características apontadas na problemática da pesquisa. Tal modelo foi desenvolvido com base nos conceitos de engenharia de requisitos e no *workflow* da disciplina de requisitos do processo unificado. Visa assegurar a qualidade das atividades da engenharia de requisitos no ambiente de desenvolvimento de *software* proposto para o NUSIS com base nos atributos de qualidade desejáveis do processo.

4.1 CARACTERÍSTICAS DO NUSIS

Para que os objetivos do modelo de processo aqui proposto possam ser compreendidos é preciso situar-se no contexto do cenário de onde a motivação para este trabalho teve origem.

Ainda em fase inicial, o projeto para o desenvolvimento de sistemas do NUSIS tem como propósito a geração de *software* construído a partir de uma equipe de trabalho formada pelos próprios acadêmicos e docentes da universidade. A criação de um ERP voltado para a plataforma *Web* pode ser tido como o principal objetivo a qual destina-se o projeto.

Com a finalidade de integrar todos os dados e processos de uma organização em um único sistema, a construção de um ERP requer a integração de diversas áreas de negócio, não ficando restritas somente às áreas relacionadas ao Centro de Computação e Tecnologia da Informação. Para isso faz-se necessário o envolvimento de alunos e professores de diferentes cursos da universidade ao longo do projeto.

A adoção de um processo é fundamental para guiar as atividades do desenvolvimento de *software*. No ambiente proposto para o NUSIS, a construção do sistema será baseada em iterações com entregas incrementais. Cada iteração pode ser vista como um mini-projeto de duração fixa, com suas próprias atividades de análise de requisitos, projeto, implementação e teste (Larman, 2007). A arquitetura do *software* também exerce forte influência sobre o projeto. Questões relacionadas à linguagem de programação, sistema operacional, banco de dados, *frameworks* de desenvolvimento, e aos requisitos não funcionais devem ser levadas em consideração ao definir-se um processo de *software* (Larman, 2007). Justifica-se contudo a escolha do processo unificado, em conjunto a UML e ao paradigma de orientação a objetos, como uma exigência do NUSIS perante as características apresentadas, além de tornar clara a necessidade de atribuição de tarefas aos indivíduos envolvidos diretamente no projeto.

Neste projeto piloto as necessidades não irão partir de um cliente/contratante do serviço, o que vêm de encontro diretamente com a forma de obtenção e validação dos requisitos do projeto. Aliado a isso, a engenharia de requisitos deve se valer de aspectos como a provável inexperiência e a alta rotatividade da equipe de trabalho, uma vez que as tarefas resultantes serão desempenhadas pelos próprios alunos dos cursos do Centro de Computação e Tecnologia da Informação através das disciplinas de Estágio Curricular e TCC. O desconhecimento acerca do negócio também é um fator relevante, e em conjunto as demais características fazem com que o acompanhamento do professor no papel de orientador do aluno ao longo das tarefas seja crucial para garantir que as atividades da engenharia de requisitos sejam realizadas da maneira correta.

Espera-se que os artefatos gerados ao longo das atividades sirvam de instrumento facilitador na compreensão dos requisitos e possibilitem aos envolvidos nas demais atividades do ciclo de vida do desenvolvimento do *software* uma melhor visibilidade do negócio e do problema.

4.2 ATRIBUTOS DE QUALIDADE DESEJÁVEIS

Para garantir a qualidade das atividades da engenharia de requisitos no ambiente de desenvolvimento de *software* proposto para o NUSIS, é necessário estabelecer quais serão os atributos de qualidade desejáveis para o processo. Para que isso seja possível é essencial levar em consideração as características elencadas para o processo, com as particularidades do projeto.

O primeiro ponto, e também o de maior criticidade, é o fato de não existir um cliente contratante do serviço. Como as necessidades não partirão de um cliente propriamente dito, as atividades de identificação, obtenção de compromisso e também aprovação dos requisitos terão de ser tratadas especificamente, num trabalho conjunto entre aluno e professor, tendo como base a documentação de negócio existente.

A alta rotatividade aliada a pouca experiência da equipe de trabalho, requerem a capacidade de que as mudanças nos requisitos sejam gerenciadas de forma eficaz e eficiente. Como não existe um prazo para término, ou até mesmo um cronograma completo de atividades para o projeto, somado ao fato da constante mudança nos envolvidos nas atividades e também a não exigência de experiência para desempenhar as tarefas, ter documentada toda e qualquer mudança nos requisitos, bem como conhecer a origem dos mesmos, pode ser considerada uma prática mais que fundamental para o ambiente do NUSIS.

As especificações geradas devem fornecer a base necessária para que os requisitos sejam testados. Uma vez construídas, as especificações devem ser capazes de comunicar as necessidades do cliente a todos os envolvidos. Todo requisito antes de especificado deve passar por uma análise criteriosa durante as etapas de levantamento e elaboração, sendo que a sua especificação só deve ter início após ser assumido compromisso de modo a corroborar o comprometimento dos participantes. Após especificados, todos os requisitos devem ser verificados e aprovados. Espera-se também que não existam ambiguidades entre requisitos. Desta forma, as especificações geradas devem ser avaliadas eliminando ambiguidades existentes.

Voltado ao cenário exposto pelo NUSIS surge também a necessidade de manter a rastreabilidade dos requisitos. Sendo um ERP um sistema modular, caracterizado pelo seu grande porte e alta complexidade, uma vez que engloba diversas áreas de negócio, conseguir rastrear um requisito de mais baixo nível até chegar na sua fonte, e vice-versa, serve como garantia para determinar se todos os requisitos foram devidamente tratados e permite aos envolvidos realizar a análise sobre o impacto que uma mudança pode gerar no contexto do projeto.

Relacionadas às características do *software* como produto, é imprescindível assegurar que os requisitos especificados satisfaçam as funções almejadas pelo cliente, e que estas sejam disponibilizadas por meio de uma boa usabilidade do *software*. Atrelado a isso, pode-se afirmar que o conjunto de requisitos deve ser completo e consistente.

Características que asseverem a viabilidade de manutenção também são desejadas, uma vez que modificações e adaptações, ou até mesmo a correção de falhas, é algo visto como uma prática comum em um produto de *software*.

O conjunto de atributos e características de qualidade desejadas para o processo de engenharia de requisitos do NUSIS é representado na tabela 3, fazendo referência aos modelos, normas e padrões em que são citados, conforme pesquisa apresentada no capítulo 2 deste trabalho.

Tabela 3 - Atributos e Características de Qualidade desejadas para o NUSIS

| Atributos / Características | ISO/IEC 12207 | ISO/IEC 15504 | CMMI | MPS.BR | IEEE 830 |
|-----------------------------|---------------|---------------|------|--------|----------|
| Comunicação | | X | | | |
| Compromisso / Aprovação | | X | X | X | |
| Capacidade de Gerenciamento | | | X | X | |
| Testabilidade | X | X | | | X |
| Não-ambiguidade | | | | | X |
| Completeness | | | | | X |
| Consistência | X | X | X | X | X |
| Verificabilidade | | | X | X | X |
| Modificabilidade | | X | X | X | X |
| Rastreabilidade | X | X | X | X | X |

4.3 PROCESSO PROPOSTO

O processo proposto possui as seguintes atividades: concepção e análise do problema, levantamento e compreensão dos requisitos, negociação dos requisitos, especificação e modelagem dos requisitos, validação dos requisitos e gerência de requisitos.

Para o processo foram definidos os papéis de acordo com a abordagem do processo unificado para a disciplina de requisitos visto na seção 3.2.3 deste trabalho. Os papéis do processo são: analista de sistemas, gerente de projeto, revisor de requisitos, arquiteto de *software*, além do cliente/usuário.

A figura 25 abaixo ilustrada apresenta o processo de engenharia de requisitos proposto para o NUSIS, enquanto que os papéis definidos para o processo são listados na tabela 4. Nos subcapítulos a seguir serão detalhadas as etapas deste processo e a interação com os papéis estabelecidos.

Figura 25 - Processo de Engenharia de Requisitos Proposto

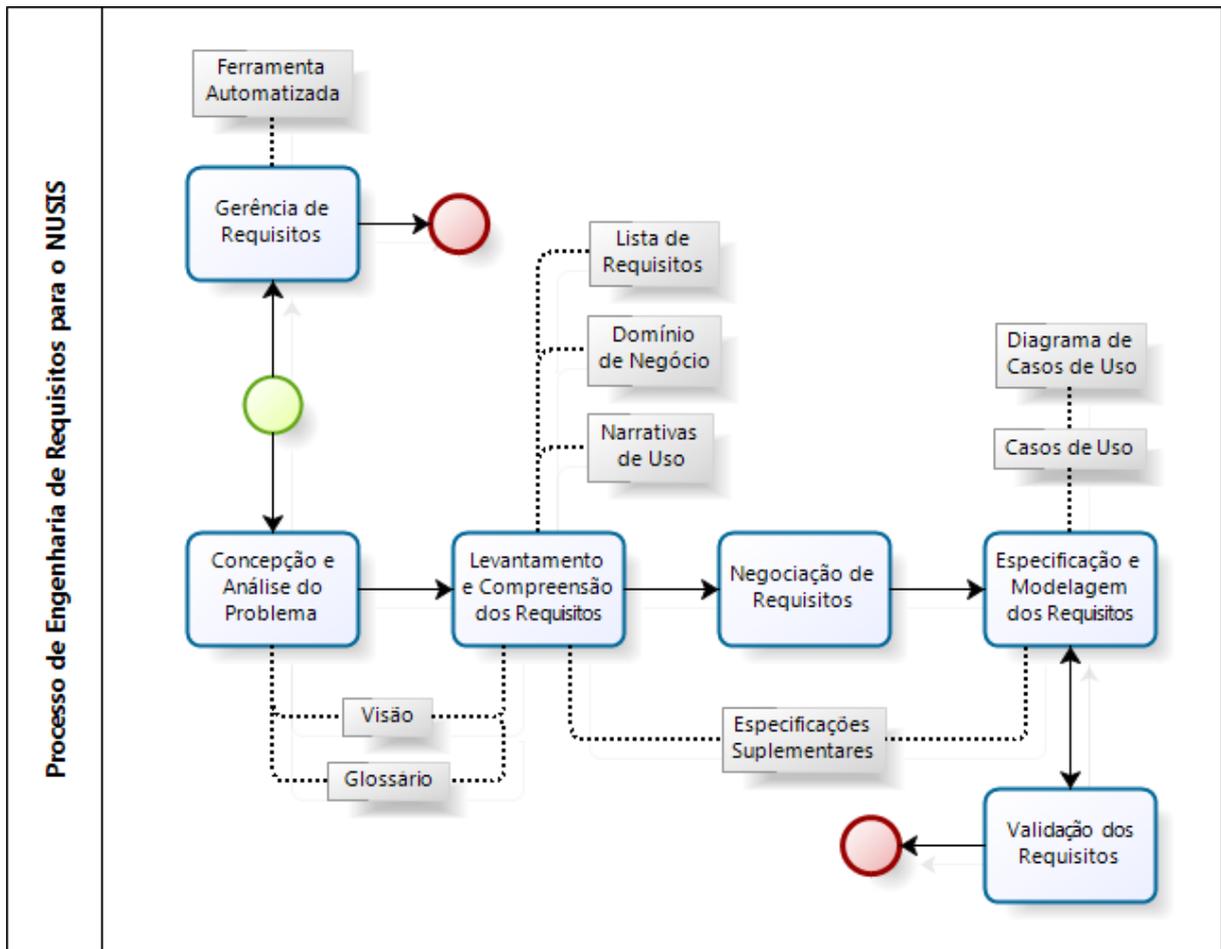


Tabela 4 - Papéis do Processo de Engenharia de Requisitos Proposto

| Indivíduo | Papéis | Atividades |
|--|---|--|
| Aluno | Analista de Sistemas | O principal responsável pelas atividades de engenharia de requisitos: concepção e análise do problema, levantamento e compreensão dos requisitos, especificação e modelagem dos requisitos e gerência de requisitos. |
| Professor Orientador | Gerente de Projeto Revisor de Requisitos | Responsável pelo bom andamento do projeto, acompanhando o Engenheiro de Requisitos em suas atividades, além de participar de forma direta da Validação dos Requisitos. |
| Professores de outras áreas e funcionários da universidade | Cliente / Usuário | Representam os <i>stakeholders</i> do sistema, atuando como fornecedores de requisitos do projeto. |

4.3.1 Concepção e Análise do Problema

Como visto anteriormente, a etapa de concepção serve como ponto de partida para as demais atividades da engenharia de requisitos (Pressman, 2006). Nessa etapa, o acadêmico no papel de analista de sistemas deve estabelecer um entendimento básico sobre o problema e os fatores a ele relacionados.

Possivelmente o acadêmico não possuirá conhecimento sobre o domínio de negócio em questão. No entanto, através da compreensão progressiva sobre este domínio, deve-se ter claro o escopo do projeto, seus critérios de sucesso e os riscos existentes.

Vem à tona desde este primeiro momento a característica mais relevante do ambiente do NUSIS, a ausência da figura do cliente. Como o ambiente de desenvolvimento é considerado experimental, uma vez que não existe um contratante para o serviço, o cliente não estará presente ao longo das atividades do processo de engenharia de requisitos. Desta forma, o professor orientador do aluno assume papel fundamental, representando os interesses dos *stakeholders*.

Cabe ressaltar que no ambiente do NUSIS a engenharia de requisitos busca resolver um problema genérico de uma organização. Um novo produto está sendo criado. O foco do projeto está voltado ao desenvolvimento de sistemas com base no conhecimento.

Nesta etapa de concepção, o aluno deve primeiramente buscar informações básicas sobre o funcionamento de um ERP. Através da análise de sistemas correlatos bem considerados perante o negócio no qual o projeto estará inserido, uma base preliminar de conhecimento deve ser formada, auxiliando na compreensão de conceitos e regras e facilitando a comunicação inicial.

Num segundo momento, o aluno deve formular um conjunto de perguntas que o auxiliem a compreender as características macro do projeto, identificando quem são os primeiros *stakeholders*, os objetivos globais e os benefícios do projeto (Pressman, 2006). As questões elaboradas devem ser respondidas pelo professor, que será o primeiro contato do aluno para esse fim. Abaixo são demonstradas algumas das questões que podem ser feitas:

- Quem vai usar a solução que será desenvolvida?
- Quem poderá ser envolvido e prestar auxílio ao longo das atividades?
- Quais os fatores cruciais de sucesso para esta solução?
- Quais seriam os resultados considerados “bons” gerados através da utilização desta solução?
- O que não pode ser deixado de lado na solução que iremos desenvolver?

- Você poderia exemplificar o ambiente de negócios no qual a solução poderá ser usada?
- Quais as restrições existentes quanto ao projeto?
- Quais as dificuldades que o projeto pode enfrentar?
- Quais os limites do projeto? Até onde irá o trabalho?
- Alguém mais pode fornecer informações sobre o processo?
- Onde posso procurar documentação sobre o processo?
- Devo perguntar-lhe mais alguma coisa?

Uma vez que se tenha essas respostas e, obtendo-se um melhor entendimento do problema e da percepção de solução, o documento visão deve ser criado. Neste artefato são descritas as principais características desejadas para o sistema (Larman, 2007). A ideia de utilizar esse artefato é fazer com que o conhecimento obtido sobre o modelo de negócio e sobre os requisitos, embora que ainda em alto nível, seja registrado e sirva como base para o detalhamento a ser realizado nas próximas atividades.

O modelo de documento visão proposto é simples, e contém a estrutura demonstrada na tabela 5.

Tabela 5 - Estrutura do Documento Visão proposto

| DOCUMENTO VISÃO | |
|---|--|
| 1. Introdução | Deve proporcionar uma abordagem inicial ao documento de visão, apresentando as necessidades e desafios levantados para o projeto. |
| 2. Escopo | Identificar qual o propósito da solução. Delimitar o que será feito e o que não será feito pelo sistema. |
| 3. Referências | Conjunto de referências para o projeto, podendo estas serem pessoas, livros, artigos, páginas da <i>WEB</i> , estudos de mercado, etc. |
| 4. Stakeholders | Deve conter a descrição e as responsabilidades dos stakeholders identificados para o projeto. |
| 5. Necessidades dos Stakeholders | Descrição das necessidades identificadas com uma breve proposta de solução identificada. |
| 6. Prioridades | Apresentar a ordem de priorização dada as necessidades identificadas. |
| 7. Recursos do sistema | Elencar, através de uma lista, o que o sistema será capaz de realizar após sua implementação. |
| 8. Outros Requisitos | Elencar, através de uma lista, requisitos não funcionais identificados. |
| 9. Informações Complementares | Destinado a descrever outros aspectos de relevância para o projeto. |
| 10. Visão Geral UML | Apresentar através da utilização de casos de uso como o sistema tende a ser implementado e como é pretendido que este venha a ser utilizado, evitando-se detalhes. |

O documento de visão não é concluído com o término dessa etapa do processo de engenharia de requisitos. Ele deverá ser incrementado durante as atividades de levantamento de requisitos e definição do sistema.

Ainda na etapa de concepção, será criado o glossário de terminologias do sistema. Esse documento, cujo objetivo é registrar o vocabulário comum do projeto na linguagem do cliente, deve ser constantemente alimentado no decorrer das atividades à medida que novos termos são identificados (Rational, 2002). Tal artefato servirá de referência a todos os participantes ao longo do desenvolvimento do projeto. Como os estágios de construção e transição do processo de *software* ocorrerão em um segundo momento, e não necessariamente na sequência das fases de concepção e elaboração que estamos tratando aqui, o glossário virá

a servir como uma importante ferramenta capaz de municiar de conhecimento as futuras equipes participantes do projeto.

4.3.2 Levantamento e Compreensão dos Requisitos

Conforme descrito na seção 3.1.2, a atividade de levantamento de requisitos transfere o foco inicial dos objetivos macros do projeto para as necessidades específicas de negócio existentes (Sommerville, 2011). Salva a verificação de documentos, todas as demais técnicas de coleta de requisitos abordadas nesta seção requerem a presença do cliente, o que vem em contraposição ao cenário apresentado pelo NUSIS.

Embora não haja a figura do cliente, o sistema de ERP a ser desenvolvido visa à integração de diversas áreas do conhecimento em um único produto, não ficando somente restrito às áreas ligadas ao Centro de Computação e Tecnologia da Informação da universidade. Desta forma, faz-se importante o envolvimento e participação de profissionais com conhecimento sobre as áreas fim a qual o incremento em questão destina-se atender, podendo estes profissionais serem professores de outros centros ou até mesmo funcionários da universidade.

Mediante isso, as técnicas de entrevista e *workshop* de requisitos podem ser utilizadas na identificação dos requisitos de negócio desejáveis para o projeto. De acordo com a área, até mesmo observações *in loco* podem ser realizadas, na qual o aluno acompanharia o processo realizado por um funcionário da universidade em sua rotina de trabalho, buscando reconhecer nesse aspectos que agreguem valor à atividade de levantamento realizado.

A busca por estes profissionais deve sempre ter o consentimento do professor no papel de orientador do aluno, que deverá guiar o aluno sobre os melhores caminhos a serem seguidos durante o levantamento.

Levando em consideração a ideia que professores e/ou funcionários com conhecimento na área foram localizados e participarão do processo de elicitação dos requisitos, o aluno deverá agendar uma reunião inicial. A mesma deverá ser previamente comunicada, deixando claro aos participantes seu objetivo e quais os tópicos serão tratados. Poderá ser apresentado ainda o documento de Visão com as informações identificadas para o projeto até então, o que servirá de base para situar os participantes no contexto do projeto.

A ideia desta reunião é firmar a colaboração dos envolvidos e dar início a coleta dos requisitos. Em um segundo momento, após esse contato inicial, o aluno poderá agendar entrevistas com os participantes, aprofundando o nível de detalhamento dos requisitos. Essas

entrevistas devem ocorrer de forma semi-estruturada. Algumas perguntas padrão devem ser seguidas para nortear a sua condução, enquanto outros questionamentos surgirão naturalmente de acordo com o ensejo e com o contexto do requisito.

Em um processo normal de levantamento uma ata seria criada e assinada pelos participantes como forma de corroborar o seu comprometimento com os requisitos. No entanto, devido às circunstâncias do projeto, a participação destes é tida como voluntária, e não deseja-se portanto fazer com que os mesmos assumam responsabilidades. Todavia, isso não exime a importância de ter registrado o conteúdo das reuniões e entrevistas. O aluno deverá deste modo, manter documentado através de uma ata simplificada os assuntos e regras discutidas, os resultados esperados e as exceções existentes, sem a necessidade de colher assinaturas dos participantes. É importante que esta ata apresente também os participantes envolvidos e a data de ocorrência.

É possível que não seja localizado nenhum professor ou funcionário da universidade com disposição em participar da etapa de levantamento, bem como pode ocorrer de mais profissionais estarem envolvidos. Essas questões vão variar de projeto a projeto, incremento a incremento.

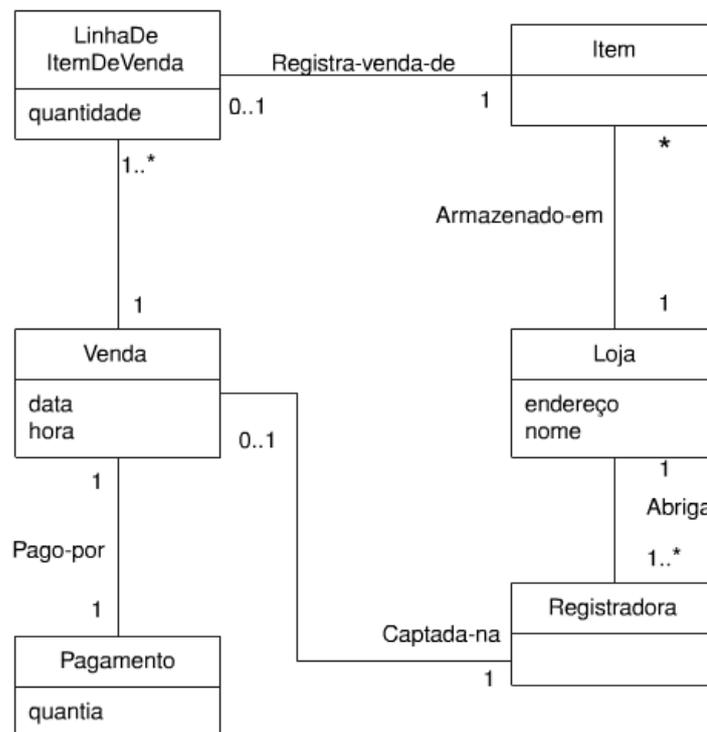
Ainda de acordo com essa disponibilidade, sendo viável acompanhar um funcionário em suas atividades do dia a dia, o aluno através da observação in loco buscaria por detalhes operacionais que por ventura tenham sido esquecidos. Tal técnica permitiria ainda avaliar se os requisitos levantados até o momento são de fato condizentes com a realidade de um usuário de ERP. No entanto, essa técnica de observação in-loco somente poderia ser aplicada para áreas de negócio específicas, existentes dentro da universidade, como finanças, contabilidade e recursos humanos.

Outra importante forma de identificar requisitos será a verificação de documentos. Esta por sua vez não necessita da personalidade do cliente. O aluno deve se valer das fontes levantadas durante a concepção e citadas no documento de visão através do item 3. Referências. Tais fontes podem ser artigos da área de negócio, livros, slides, manuais de processos, páginas da *WEB*, entre outros, que auxiliem o aluno a aprofundar seu entendimento sobre os requisitos necessários para implementação do incremento.

No decorrer da atividade de levantamento, o documento de Visão deve ser atualizado com as necessidades coletadas, suas exceções e também possíveis soluções, além de refinar os recursos desejados para o sistema, apontar os novos *stakeholders* e usuários identificados, novos requisitos não funcionais e outras informações adicionais. O glossário de termos também deve ser incrementado e refinado durante esta etapa da engenharia de requisitos.

Outro artefato criado no decorrer da atividade de levantamento será o modelo de domínio. O objetivo desse artefato é capturar e representar os principais conceitos e entidades que envolvem o mundo real com o qual o incremento de *software* a ser desenvolvido se relaciona (Pressman, 2006). A construção desse modelo deve ser feita utilizando o diagrama de classes da UML. A forma de representação deve ser simples, demonstrando apenas as classes em alto nível, com seus nomes e os relacionamentos e associações de maior relevância para o processo. O modelo de domínio permitirá tanto ao aluno que o estiver modelando quanto ao analista de sistemas que posteriormente irá projetar o desenvolvimento do *software* ter uma percepção inicial simplificada do contexto no qual o projeto está inserido. A figura 26 apresenta um exemplo de modelo de domínio. Nele estão representados ainda a multiplicidade entre as classes.

Figura 26 – Exemplo de Modelo de Domínio



Fonte: adaptada de (Larman, 2007)

A conclusão dessa etapa se dá com a elaboração de uma lista oficial de requisitos funcionais do sistema e suas narrativas de uso. A criação da lista de requisitos objetiva evidenciar todas as funcionalidades desejadas para o *software* (Wazlawick, 2004), e permitir que os artefatos gerados ao longo das atividades da engenharia de requisitos sejam rastreados

até os requisitos que lhes originaram. Já utilização de narrativas de uso, conforme visto na seção 3.3.2, é aconselhada para que pessoas com pouca experiência na escrita de casos de uso possam assimilar melhor o domínio do negócio, podendo assim detalhá-lo de forma correta (Cockburn, 2005).

A tabela 6 demonstra o modelo proposto para construção da lista de requisitos e a exemplificação de seu preenchimento com requisitos de um sistema de leilões eletrônicos, enquanto que a figura 27 apresenta o modelo para construção das narrativas de uso.

Tabela 6 - Modelo e Exemplificação da Lista de Requisitos

| ID Requisito | Funcionalidade | Descrição | Atores |
|--------------|-------------------|--|------------------------|
| R1 | Cadastrar Usuário | Permite ao usuário se cadastrar ou alterar seu cadastro no sistema para criar leilões, dar lances em leilões abertos ou abrir seu próprio leilão. | Consumidor, Fornecedor |
| R2 | Cancelar Cadastro | Permite ao usuário cancelar seu cadastro no sistema a qualquer momento | Consumidor, Fornecedor |
| R3 | Criar Leilão | Permite que o consumidor crie um leilão para determinado produto. <RN o usuário deverá informar a forma de pagamento e o término do leilão e o local de entrega> | Consumidor |
| R3 | Efetuar Lance | Permite ao usuário efetuar um lance em leilões em aberto. | Fornecedor |

Fonte: adaptada de (Wazlawick, 2004).

Figura 27 - Modelo proposto para as Narrativas de Uso

| |
|---|
| <p>ID Requisito: <código identificador do requisito></p> <p>ID Narrativa de Uso: <código identificador da narrativa de uso></p> <p>Versão: <versão do artefato></p> <p>Data da Última Alteração: <data de última alteração do artefato></p> <p>Analista de Alteração: <persona responsável pela última alteração no artefato></p> <p>Analista Responsável: <aluno responsável pelo levantamento e construção do artefato></p> <p>Professor Orientador: <professor responsável por acompanhar as atividades do aluno></p> |
| <p><u>Narrativa de Uso</u></p> <p>Fluxo Normal: <texto livre, com uma descrição do fluxo normal de eventos do cenário></p> <p>O que pode dar errado: <uma descrição do que pode dar errado e como isso pode ser tratado></p> |

A pouca experiência é um dos fatores críticos para o sucesso das atividades da engenharia de requisitos identificados na problemática do NUSIS. Junto a isso, a utilização de casos de uso será o mecanismo proposto como forma de documentação dos requisitos nas

seções seguintes. Logo, a ideia é fazer com que o aluno exerça a tarefa de construção das narrativas com o apoio dos envolvidos durante um *workshop* de requisitos, como forma de desenvolver uma percepção mais aguçada acerca dos requisitos elicitados, forçando a sua compreensão para desta maneira poder criar os cenários para as narrativas.

Por meio da construção das narrativas de uso o aluno conduziria professores e funcionários a discutir sobre as necessidades encontradas, levando a uma convergência de pontos de vista sobre as funcionalidades desejadas para o sistema. Através dessa forma dinâmica de troca de ideias são elaboradas as narrativas de uso, e conseqüentemente a lista de requisitos oficiais da etapa de levantamento, culminando com o término desta etapa.

4.3.3 Negociação de Requisitos

Segundo (Pressman, 2006), ao longo de toda a etapa de levantamento devem ocorrer negociações com os *stakeholders*. É de conhecimento que no projeto do NUSIS não existe um cliente diretamente relacionado, no entanto, os próprios professores da universidade representam os *stakeholders* e seus interesses.

Durante todas as atividades de elicitação, seja nas entrevistas, *workshops* de requisitos e até mesmo na análise de documentos, o aluno deve concentrar o foco de seu trabalho em assegurar que os objetivos do projeto sejam alcançados. Num primeiro momento, enquanto ainda estiver empenhado em coletar os requisitos, é importante assimilar os riscos, a complexidade e o impacto de cada funcionalidade desejada para o sistema com base nos pontos de vista de cada *stakeholder*. Será a partir dessas informações que a priorização dos requisitos ocorrerá.

O professor orientador terá papel fundamental na atividade de priorização dos requisitos. Como será este professor o responsável por assessorar o aluno em suas atividades ao longo do projeto, e por entender que o mesmo deverá estar integrado com o assunto e os objetivos traçados, e também com o intuito de eximir os demais professores e funcionários de maiores responsabilidades, a priorização dos requisitos será realizada entre o aluno e seu orientador.

Para isso, todas as necessidades dos *stakeholders* deverão estar registradas no documento visão, o glossário de termos alimentado, e as narrativas de uso que descrevem a exemplificação do comportamento desejado para o sistema a ser construído. Por meio destes artefatos, e de acordo com a sua percepção sobre os requisitos, o aluno trará ao conhecimento do orientador o conjunto de requisitos identificados na atividade de levantamento.

Conjuntamente os dois trabalharão na detecção dos requisitos tidos como críticos para o processo, chegando num comum acordo sobre quais requisitos devem entrar no escopo do projeto.

Em um projeto normal, fatores como prazos, custos e recursos para desenvolvimento também teriam de ser levados em consideração ao definir-se a prioridade dos requisitos (Pressman, 2006). No entanto, tais aspectos não precisam ser levados em conta devido ao ambiente experimental do NUSIS.

Por fim, a concordância sobre o conjunto de requisitos deve ser ratificada junto aos funcionários e professores que participaram da atividade de levantamento. Nesse momento discordâncias podem surgir. Buscar conciliar esses conflitos é um dos objetivos da etapa de negociação, para isso, pode ser que sejam necessárias adaptações ou mudanças nos requisitos (Luna, 2008).

4.3.4 Especificação e Modelagem dos Requisitos

As informações coletadas ao longo do levantamento de requisitos e registradas através do documento de visão e também através das narrativas de usos serão as entradas para a especificação e modelagem do requisito.

A técnica de modelagem adotada será a de casos de uso, em seu formato completo, descrita na seção 3.3. A escolha por esta técnica se justifica pela adoção do processo unificado para o ambiente de desenvolvimento de *software* proposto para o NUSIS. Uma das características do processo unificado é o seu direcionamento a casos de uso, ou seja, todo o processo de desenvolvimento executa suas tarefas tendo como base os casos de uso descritos para o sistema (Scott, 2003). Em um ambiente como o NUSIS, onde o ciclo de desenvolvimento será iterativo e as entregas do *software* incremental, utilizar casos de uso para descrever e registrar os requisitos é sem dúvida a técnica mais adequada.

Na atividade e especificação e modelagem um requisito poderá gerar múltiplos casos de uso, assim como um caso de uso poderá representar mais de um requisito. Já as narrativas de uso, embora desejáveis, são opcionais. Um requisito poderá possuir diversas narrativas, que facilitem o entendimento do cenário e colaborem para a criação dos casos de uso. No entanto, requisitos considerados básicos podem não contar com narrativas.

Após documentar os requisitos por meio de casos de uso descritivos, deverá ser criada uma representação do conjunto de casos de uso do sistema através da notação de diagrama de casos de uso da UML. O objetivo da utilização desse diagrama é facilitar o

entendimento futuro dos requisitos por parte da equipe de desenvolvimento, uma vez que tal diagrama é especialmente útil para evidenciar os recursos que o sistema deve ter, descrevendo suas funcionalidades de uma forma visual e de fácil compreensão (Booch, 2000).

O modelo para casos de uso completos proposto possui a estrutura apresentada na figura 28.

Figura 28 - Modelo proposto para os Casos de Uso

| |
|---|
| <p>ID Requisito: <código identificador do requisito></p> <p>ID Caso de Uso: <código identificador do caso de uso></p> <p>Versão: <versão do artefato></p> <p>Data da Última Alteração: <data de última alteração do artefato></p> <p>Analista de Alteração: <pessoa responsável pela última alteração no artefato></p> <p>Analista Responsável: <aluno responsável pelo levantamento e construção do artefato></p> <p>Professor Orientador: <professor responsável por acompanhar as atividades do aluno></p> |
| <p>Caso de Uso: <nome do caso de uso, composto pelo objetivo e iniciado por um verbo ativo></p> <p>Objetivo no contexto: <uma sentença maior do objetivo, se necessária, suas condições de ocorrência normal></p> <p>Ator Principal: <nome de papel para o ator principal que procura os serviços para atingir um objetivo></p> <p>Stakeholders e Interesses: <lista de <i>stakeholders</i> e seus interesses chaves no caso de uso></p> <p>Pré-Condições: <o que se espera que seja verdadeiro antes do início do fluxo do caso de uso></p> <p>Garantias Mínimas: <as garantias mínimas oferecidas pelo sistema para a execução de qualquer fluxo do caso de uso></p> <p>Garantias de Sucesso: <o que deve ser verdadeiro quando da bem-sucedida execução do caso de uso, seja pelo fluxo do cenário de sucesso principal ou outro fluxo alternativo></p> <p>Acionador: <quem dá início ao caso de uso, podendo ser um ator ou um evento></p> <p>Cenário de Sucesso Principal: <descreve o caminho típico de sucesso que satisfaz os interesses dos <i>stakeholders</i>></p> <p>Extensões: <descrevem todos os outros cenários possíveis de sucesso ou fracasso para o caso de uso></p> <p>Lista de Variações Tecnológicas e de Dados: <variantes técnicas sobre como algo deve ser feito, mas não sobre o que deve ser feito, e que vale a pena estar registrado no caso de uso></p> <p>Casos de Uso Relacionados: <lista de outros casos de uso relacionados ao longo do fluxo do cenário de sucesso principal ou outro fluxo alternativo></p> |
| <p>Aceite do Caso de Uso: <utilizado posteriormente para indicar a validação do requisito></p> <p><input type="checkbox"/> Aprovado</p> <p><input type="checkbox"/> Reprovado</p> <p>Prioridade do Caso de Uso: <define a prioridade de implementação do requisito após sua validação></p> <p><input type="checkbox"/> Baixa</p> <p><input type="checkbox"/> Normal</p> <p><input type="checkbox"/> Alta</p> <p><input type="checkbox"/> Crítica</p> |

Relacionado aos requisitos não funcionais, os mesmos devem ser documentos através do artefato de Especificações Suplementares. O modelo adotado para este será o mesmo definido pela (Rational, 2002) para o processo unificado, onde são elencados requisitos de usabilidade, confiabilidade, desempenho, suportabilidade, documentação, além de restrições

de design, interface e legislação. É importante salientar que este artefato não é finalizado neste momento. O mesmo evolui em conjunto com a atividade de especificação e modelagem dos requisitos.

Ao longo das atividades de especificação e modelagem dos requisitos deve-se aprimorar o modelo de domínio criado. Devem ainda ser refinados o glossário e o documento de visão do sistema com base no atual entendimento acerca do projeto e seus requisitos.

4.3.5 Validação dos Requisitos

Conforme visto na seção 3.1.6, a validação de requisitos é o processo que determina se a especificação realizada é consistente com a definição dos requisitos, assegurando que as necessidades dos *stakeholders* serão atendidas (Pfleeger, 2004).

Em um processo normal, um representante ou usuário chave dentre os *stakeholders* ouvidos durante a fase de levantamento do requisito seria convidado a participar da etapa de validação. No caso do NUSIS, é possível que até tenhamos o envolvimento de outras pessoas na atividade de elicitação, entretanto, conforme já argumentado anteriormente, optou-se pela isenção destes nas demais tarefas do processo de engenharia de requisitos aqui proposto.

Após o levantamento, todas as informações passarão por uma etapa de negociação junto ao professor orientador no papel de revisor de requisitos, onde os requisitos elicitados serão devidamente analisados, escolhidos e priorizados de acordo com o projeto. Por conseguinte, acredita-se que este professor estará apto a se posicionar com relação a especificação efetuada, uma vez que esteve envolvido diretamente no processo desde o seu princípio. A ideia com isso, é efetuar a validação dos requisitos junto ao professor que estará orientando o aluno durante suas atividades. Conjuntamente, aluno e professor conseguirão aferir se a maneira como os requisitos foram especificados através dos casos de uso permitirá com que o sistema seja implementado durante a fase de construção, e também se o incremento de *software* atenderá a necessidade de negócio demandada após sua implantação.

Desta forma, a validação do requisito será realizada de forma manual através de sua revisão, considerando pontos importantes que influenciam na definição do requisito como por exemplo: a comparação da definição com as metas e objetivos traçados para o sistema, descrição do ambiente onde o requisito irá operar e verificação dos riscos que envolvem a definição (Pfleeger, 2004).

O aceite formal do requisito será feito através do próprio caso de uso. Para isso, uma seção exclusiva será destinada à validação do requisito. Por meio do item “*Aceite do Caso de*

Uso” apresentado ao final do caso de uso, deverá ser indicado se o mesmo está ou não de acordo com as especificações descritas. Junto a essa informação será adicionado uma classificação de prioridade do caso de uso para a etapa de desenvolvimento, com o objetivo de auxiliar a equipe que atuará na construção do sistema a identificar a criticidade do requisito perante o negócio do cliente. A figura 29 demonstra as seções relativas à validação de requisitos no modelo para construção dos casos de uso.

Figura 29 - Seções para validação dos requisitos no modelo proposto para os Casos de Uso

Aceite do Caso de Uso:

Reprovado

Aprovado

Prioridade do Caso de Uso:

Baixa

Normal

Alta

Crítica

4.3.6 Gerência de Requisitos

Como trazido no capítulo 3, a gerência de requisitos engloba um conjunto de atividades que auxiliam na identificação, controle, rastreamento e administração de mudanças nos requisitos, sendo importante o acompanhamento individual dos requisitos e suas ligações de dependências, de forma que seja possível avaliar o impacto de possíveis mudanças (Sommerville, 2011).

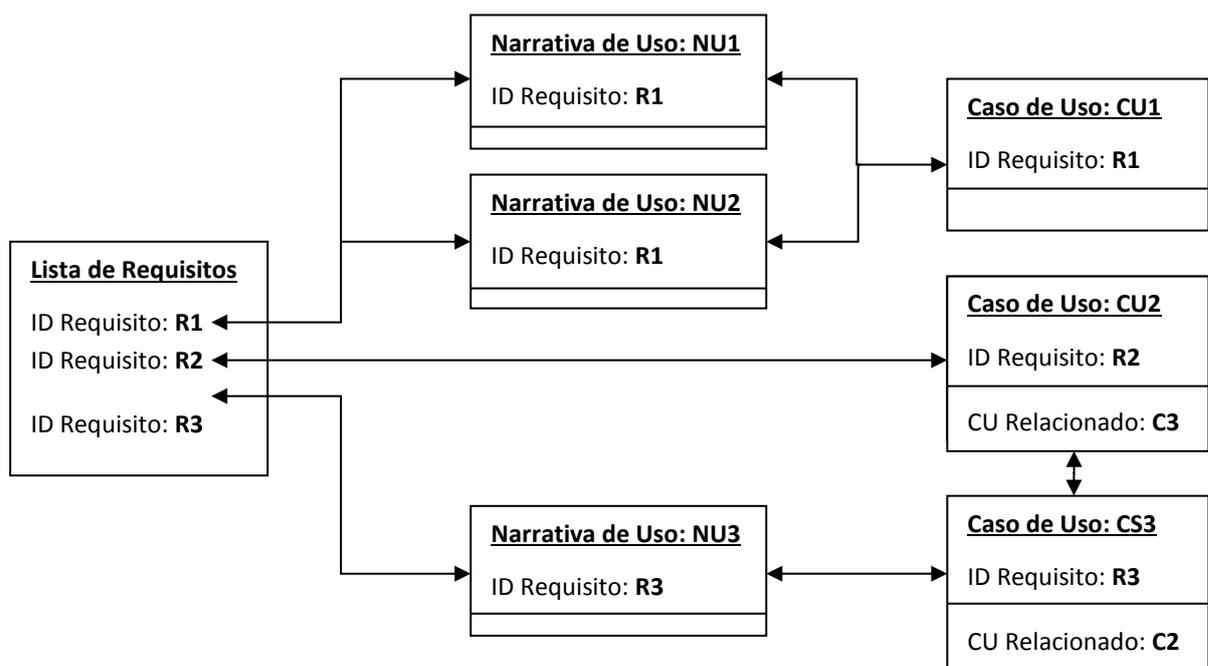
Durante o projeto as necessidades não partirão diretamente de um cliente, e conseqüentemente não virão formalizadas através de um documento. No entanto, todos os requisitos deverão estar contemplados na lista de requisitos oficial do sistema. Nesta lista, cada requisito possuirá um código identificador (ID Requisito), destinado a identificá-lo de forma única dentro de todo o projeto.

No modelo proposto, os requisitos são representados através dos casos de uso e suas respectivas narrativas. Os modelos desses artefatos foram concebidos de forma a possibilitar o relacionamento para com os requisitos, por meio da seção “ID Requisito”. Estes mesmos modelos contam ainda com a seção “Versão”, cujo objetivo é permitir o versionamento dos requisitos através de uma numeração incremental e sequencial, e as seções “Data da última alteração” e “Analista de Alteração”, responsáveis por registrar quando e quem realizou a última alteração no caso de uso.

A rastreabilidade bidirecional entre requisitos é imprescindível. Uma ligação entre requisitos é representada dentro de um caso de uso quando o nome de outro caso de uso é trazido de forma sublinhada no texto que descreve as ações do acionador, do cenário de sucesso principal ou das extensões (Cockburn, 2005). Todas essas ligações entre casos de uso devem ser listadas na seção “Casos de Uso Relacionados”. É comum relacionar casos de uso no estabelecimento de condições e ações a serem realizadas pelo sistema. Permitir o rastreamento de um requisito fonte até seus requisitos de mais baixo nível, e destes de volta para o seu requisito fonte é essencial no impacto de mudanças ao longo de todo o processo de desenvolvimento (SEI, 2010).

A figura 30 ilustra um exemplo de rastreabilidade entre os artefatos do modelo de casos de uso e seus requisitos.

Figura 30 - Exemplo de Rastreabilidade no entre os artefatos e seus requisitos



Outras formas de rastreabilidade citadas por (Pressman, 2006) como de interface e subsistemas, também podem ser consideradas dispensáveis nesse momento para processo de requisitos do NUSIS.

Tendo em vista que o foco desta pesquisa é propor um processo de engenharia de requisitos para a construção de um ERP, que por sua vez é caracterizado pela modularidade, grande porte e alta complexidade, ter o apoio de uma ferramenta automatizada para as tarefas de gerenciamento de requisitos é algo mais que recomendado.

Como visto no capítulo 3.1.7, utilizar matrizes de rastreabilidade em sistemas com muitos requisitos pode se tornar algo extremamente complexo e dispendioso. Além de facilitar a atividade de gerenciamento de rastreabilidade, as ferramentas automatizadas trazem uma série de benefícios quanto ao armazenamento dos requisitos e o gerenciamento de mudanças (Sommerville, 2011).

Mediante a esses fatores, e de acordo com o ambiente a qual destina-se o modelo que aqui propõe-se, fica implícito que a utilização de uma ferramenta *CASE* é algo fundamental para uma boa gerência dos requisitos. No mercado muitas são as ferramentas que podem ser encontradas para esse fim. Dentre elas podemos destacar a RequisitePro da IBM, a OSRMT e a Jeremia, ferramentas de código aberto. Fica estabelecido que durante o seguimento das atividades desta pesquisa será aprofundado o estudo de ferramentas para gestão de requisitos que adéquem-se ao ambiente do NUSIS.

4.4 CONSIDERAÇÕES FINAIS

A partir do modelo de processo proposto neste capítulo acredita-se ser possível assegurar a qualidade das atividades da engenharia de requisitos no ambiente de desenvolvimento de *software* proposto para o NUSIS.

Na tabela 3 apresentada na seção 4.2 deste mesmo capítulo, foram elencados os atributos e características de qualidade desejáveis para as atividades da engenharia de requisitos. A seleção destes atributos e características levou em consideração o contexto da problemática que envolve o NUSIS, e foi realizada com base no estudo dos diferentes modelos, normas e padrões de qualidade vistos no capítulo 2, extraído dos resultados esperados e dos aspectos de qualidade estipulados por cada padrão o que entendeu-se necessário e desejável para o ambiente da universidade.

A principal particularidade deste modelo em comparação a realidade normalmente vivenciada dentro das empresas desenvolvedoras de *software* é sem dúvida a ausência da figura do cliente. A análise de sistemas correlatos e o estudo de documentação acerca do negócio, bem como o envolvimento de professores de outras áreas da universidade, e até mesmo de funcionários, foi a forma encontrada para sanar o problema da inexistência de um cliente. Além deste fator, buscou-se levar em conta a pouca experiência do analista de sistemas, papel este conferido ao aluno, seu provável desconhecimento do negócio, e a alta rotatividade da equipe de trabalho, na elaboração do modelo de processo apresentado neste capítulo.

Assim como no processo unificado, o modelo proposto busca deixar clara as responsabilidades de cada envolvido através dos papéis definidos (Rezende, 2005). Preconiza a geração de uma série de artefatos e documentos: visão, glossário, modelo de domínio, lista de requisitos, narrativas de uso, casos de uso, especificações suplementares e diagrama de casos de uso do sistema. Todo esse material evidencia a necessidade de ter registrado da melhor forma possível todo o trabalho ao longo das atividades. Em um processo de desenvolvimento de *software* iterativo-incremental como é o caso do NUSIS, a adoção dessa prática pode ser tida como essencial para o sucesso do projeto num longo prazo.

Dos atributos e características extraídas das normas e tidas como desejáveis para o processo, pode-se justificar a comunicação através das especificações dos requisitos por meio dos casos de uso e das especificações suplementares. Entende-se que estes artefatos serão capazes de comunicar a todos os interessados no projeto quais são os seus requisitos. Os casos de uso produzidos devem assegurar também que a capacidade de que os requisitos sejam testados.

O compromisso para os requisitos normalmente é firmado junto aos *stakeholders*. No processo proposto, busca-se eximir os professores e funcionários da universidade, que por sua vez representam os interesses do cliente e dos usuários do sistema, de responsabilidades no projeto. Desta forma, o comprometimento será acordado apenas entre o professor orientador, que neste caso representaria um gerente de projeto com propriedade no assunto, e o aluno, no papel de analista de sistemas.

As especificações geradas devem ser revisadas e aprovadas, avaliando e eliminando ambiguidades existentes. Através da gerência de requisitos, para a qual ficou definida a necessidade de realizar um estudo das ferramentas *CASE* existentes e a sua aderência ao NUSIS, será possível rastrear para os casos de uso produzidos, quais foram as narrativas de uso descritas (caso existam), e qual os requisitos que os originaram. A atividade de gerenciamento dos requisitos permite ainda avaliar se todos os requisitos foram especificados e se estão consistentes com a necessidade coletada.

Como ao longo de um ciclo de vida de um *software* mudanças surgem constantemente (Sommerville, 2007), a modificabilidade é obtida em conjunto a atividade de gerência, que permite a avaliação do impacto de uma mudança nos requisitos.

Por fim, entende-se que o modelo proposto é aderente as premissas apresentadas pelo ambiente do NUSIS e garante a qualidade das atividades da engenharia de requisitos. No entanto, a avaliação deste modelo somente será possível através do seu instaciamento para um estudo de caso, a ser realizado no seguimento deste trabalho.

5 ANÁLISE DAS FERRAMENTAS CASE

Conforme abordado na atividade de gerência de requisitos do processo proposto, vide capítulo 4.3.6, ficou implícita a necessidade de utilização de uma ferramenta *CASE* ao longo das tarefas da engenharia de requisitos estabelecidas, tendo em vista o conjunto de características identificadas para o processo, o ambiente a qual destina-se o projeto, e os benefícios que a utilização de uma ferramenta automatizada podem trazer.

Gerir a evolução dos requisitos com o desenrolar de um projeto é uma tarefa difícil de ser realizada de forma manual. Uma ferramenta apropriada pode colaborar significativamente com muitos dos desafios que a gerência de requisitos precisa responder. Uma ferramenta *CASE* pode ser descrita como um assistente automático que acompanha o ciclo de vida dos requisitos (Rodrigues, 2008). Nesse sentido, podem ser encontradas diversas ferramentas destinadas à gerência de requisitos no mercado.

5.1 PROCESSO DE AVALIAÇÃO DA FERRAMENTA CASE

A principal medida para avaliar o sucesso de um sistema informatizado reside no simples fato deste servir ou não para os fins com o qual é destinado. Desta forma, pode-se dizer que toda escolha de *software* num âmbito empresarial deve seguir um processo de avaliação.

A série de normas ISO/IEC 14598, que deve ser aplicada em conjunto à ISO/IEC 9126, tem como objetivo a definição de um processo para avaliação de *softwares*, fornecendo também orientação quanto à avaliação prática dos mesmos. São recomendações que visam assegurar o cumprimento das necessidades do utilizador e do cliente, sendo direcionadas tanto para *softwares* já existentes quanto para os e desenvolvimento.

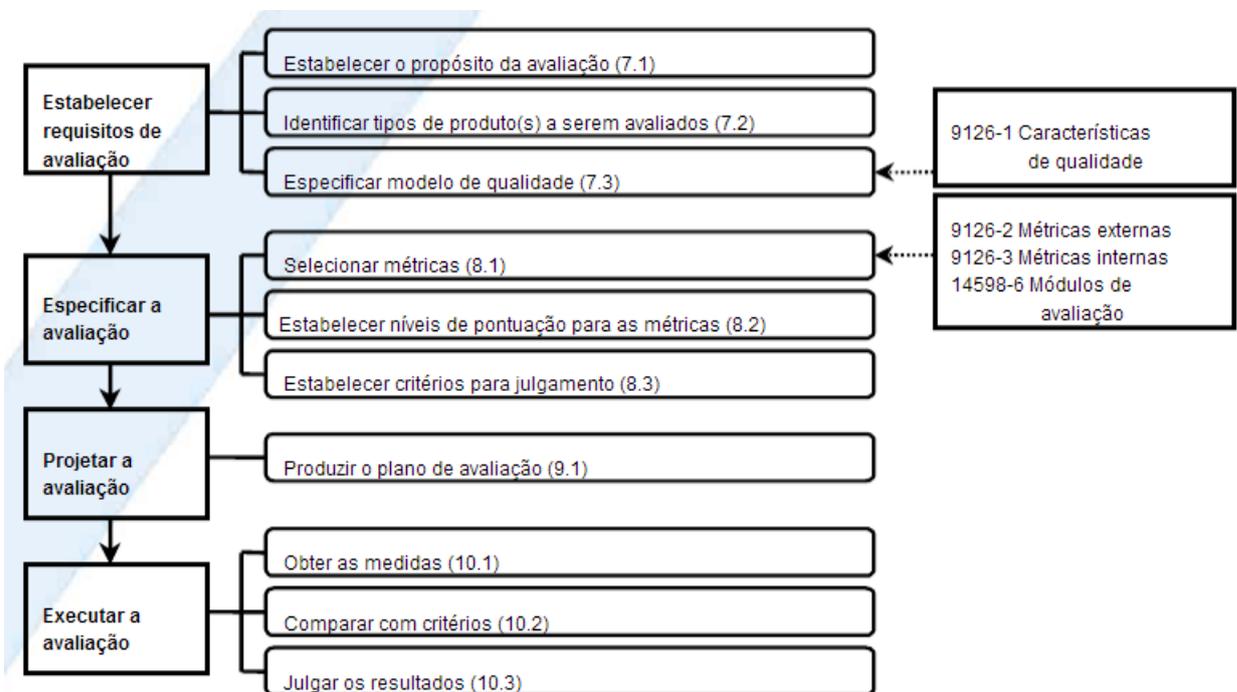
Dentre os documentos da norma, a ISO/IEC 14598-5 sugere a realização de cinco etapas durante o processo de avaliação de um produto de *software*. São elas (ISO/IEC, 2000):

1. Estabelecer Requisitos de Avaliação: definição dos objetivos da avaliação e dos requisitos a serem avaliados.
2. Especificar a Avaliação: definição do escopo e das métricas da avaliação, especificando as medições as quais o produto de *software* será submetido.
3. Projetar a Avaliação: elaboração de um plano de avaliação no qual estejam documentados os procedimentos a serem utilizados pelo avaliador na execução das medições no produto de *software* especificadas na etapa anterior.

4. Executar a Avaliação: consiste na inspeção, medição e teste dos produtos de *software* e seus componentes de acordo com o plano de avaliação. Ao final dessa etapa obtém-se um rascunho do relatório e dos registros da avaliação.
5. Conclusão da Avaliação: revisão do relatório da avaliação e disponibilização dos dados resultantes.

Uma visão geral deste processo pode ser visualizada na figura 31.

Figura 31 - Processo de Avaliação (ISO/IEC, 2000)



De acordo com a ISO/IEC 14598-4, a seleção de *softwares* para o processo de avaliação pode ser realizada com base em *feedbacks* do usuário final do produto, através de revisões de documentações, pesquisas em literatura e testes de produto. Como já explanado anteriormente, o projeto em questão possui caráter experimental e não conta com usuários em sua etapa atual de desenvolvimento. Desta forma, os *softwares* elencados se valeram da opinião especializada de professores do CCTI ministrantes de disciplinas ligadas à Engenharia de *Software*, sites conceituados de conteúdo e discussão acerca do assunto, além de pesquisas bibliográficas em livros e teses.

Uma das premissas do projeto que visa o desenvolvimento de sistemas de informação do NUSIS é a utilização de ferramentas que não acarretem custos financeiros à

universidade ou a qualquer um dos envolvidos. Com base nesse fator, a seleção das ferramentas a serem avaliadas procurou alternativas nos *softwares* considerados de domínio público, que não requerem a aquisição de licenças para sua utilização. Apenas uma ferramenta que não se encaixa nesse padrão e que é comercializada no mercado foi selecionada para o processo de avaliação: a RequisitePro da IBM, devido a uma parceria existente entre a empresa detentora dos direitos do *software* e a Universidade de Caxias do Sul, que torna viável a sua utilização para fins acadêmicos. Esta ferramenta mundialmente conhecida tem participação expressiva no mercado de ferramentas *CASE*, sendo uma das aplicações mais utilizadas na gestão de requisitos por empresas desenvolvedoras de *software* (Rodrigues, 2008).

Para que o modelo de processo de engenharia de requisitos proposto possa ser validado através de sua aplicação no ambiente do NUSIS torna-se imprescindível a utilização de uma ferramenta *CASE*. Nesse sentido, são muitos os *softwares* gratuitos encontrados que destinam-se às atividades da engenharia de requisitos, e uma avaliação completa dessas diferentes ferramentas passa pelas etapas de instalação, configuração, aprendizagem e utilização de cada *software* individualmente, para somente assim poder avaliar a sua conformidade para com os requisitos estabelecidos.

Diante disso, o processo de avaliação das ferramentas *CASE* foi dividido em duas etapas. Em um primeiro momento realizou-se a avaliação somente das ferramentas gratuitas. Tal avaliação teve como base um relatório técnico elaborado pela Universidade Federal de Pernambuco (Alves, 2007), com o objetivo de mostrar que as ferramentas gratuitas para gerenciamento de requisitos podem ser uma boa alternativa para as empresas brasileiras devido ao seu baixo custo de implantação e flexibilidade de adaptação.

A tabela 7 lista as ferramentas *CASE* gratuitas analisadas e indica o endereço onde as mesmas estão disponíveis na internet.

Tabela 7 - Ferramentas gratuitas avaliadas (Alves, 2007)

| Ferramenta | |
|-------------------|---|
| Nome | ATRequid |
| Endereço | http://sourceforge.net/project/showfiles.php?group_id=43008 |
| Nome | OSRMT - Open Source Requirements Management Tool |
| Endereço | http://sourceforge.net/projects/osrmt/ |
| Nome | TIGER PRO - Tool to InGest and Elucidate Requirements PROfessional |
| Endereço | http://www.therightrequirement.com/TigerPro/TigerPro.html |
| Nome | TRUC - Tracking Requirements & Use Cases |
| Endereço | http://sourceforge.net/project/showfiles.php?group_id=180360 |
| Nome | WIBNI |
| Endereço | http://www.johnrichards.pwp.blueyonder.co.uk/pt/ |
| Nome | Xuse |
| Endereço | http://sourceforge.net/project/showfiles.php?group_id=154928 |

Uma vez selecionados os produtos de *software*, tem início a etapa de definição dos requisitos a serem avaliados, com as características de qualidade relevantes ao processo. Fundamentado nas características de qualidade definidas pela norma ISO/IEC 9126 e ainda sob a ótica do relatório técnico que analisou as ferramentas *CASE* gratuitas, os requisitos avaliados foram especificados em categorias, sendo extraídos destas apenas os requisitos que interessam ao processo de engenharia de requisitos do NUSIS em seu estágio atual de desenvolvimento. Com base nas características de qualidade de

Toda avaliação precisa de métricas. A precisão de uma avaliação de qualidade depende em grande parte das métricas escolhidas (Manera, 2007). Partindo desse pensamento e baseado nos requisitos de avaliação das normas ISO/IEC 14598 e ISO/IEC 9126, foram atribuídas notas de acordo com o grau de conformidade das ferramentas com cada um dos requisitos, como descrito na tabela 8.

Tabela 8 - Sistema de Métricas

| Conformidade | Nota |
|-------------------------------------|-------------|
| Não atende o critério | 0 |
| Atende com sérias restrições | 1 |
| Atende parcialmente | 2 |
| Atende | 3 |

Os resultados obtidos para as ferramentas *CASE* gratuitas podem ser visualizados através da tabela 9.

Tabela 9 - Resultado da avaliação das ferramentas *CASE* gratuitas

| Requisito | ATRequid | OSRMT | TIGER PRO | TRUC | WIBNI | Xuse |
|---|----------|-------|-----------|------|-------|------|
| Documento | | | | | | |
| Dados estruturados | 3 | 3 | 3 | 1 | 3 | 1 |
| Atributos customizáveis | 0 | 3 | 0 | 0 | 3 | 0 |
| Suporte a figuras, tabelas e gráficos | 0 | 2 | 0 | 3 | 0 | 0 |
| Rastreabilidade | | | | | | |
| Relacionamentos verticais | 0 | 3 | 2 | 0 | 3 | 3 |
| Relacionamentos horizontais | 0 | 3 | 0 | 0 | 3 | 3 |
| Criação de semântica para rastreabilidade | 0 | 3 | 1 | 0 | 0 | 0 |
| Matriz de rastreabilidade | 0 | 3 | 0 | 0 | 0 | 0 |
| Relatórios | | | | | | |
| Criação de relatórios | 3 | 2 | 3 | 0 | 3 | 3 |
| Exportação de relatório | 0 | 3 | 2 | 0 | 2 | 0 |
| Busca de requisitos | 0 | 3 | 0 | 3 | 3 | 0 |
| Ordenação de requisitos | 0 | 3 | 0 | 0 | 3 | 0 |
| Controle de versão | | | | | | |
| Controle de mudanças | 0 | 3 | 0 | 2 | 3 | 1 |
| Recuperação de versão | 0 | 3 | 0 | 1 | 0 | 1 |
| Aprovação de documentos | 0 | 0 | 0 | 1 | 3 | 0 |
| Trabalho colaborativo | | | | | | |
| Trabalho concorrente | 3 | 3 | 3 | 3 | 2 | 3 |
| Controle de acesso multi-nível | 0 | 3 | 0 | 1 | 1 | 2 |
| Ambiente | | | | | | |
| Grátis | 3 | 3 | 3 | 3 | 3 | 3 |
| Open source | 3 | 3 | 0 | 3 | 0 | 3 |
| Manual de uso / help | 0 | 3 | 3 | 1 | 3 | 0 |
| Integração com outras ferramentas | 0 | 0 | 1 | 0 | 3 | 0 |
| TOTAL | 15 | 55 | 21 | 22 | 41 | 23 |

Dentre as ferramentas gratuitas avaliadas, a que alcançou os melhores resultados foi a OSRMT - Open Source Requirements Management Tool. Além de ser a mais completa dentre as ferramentas avaliadas, seu projeto ainda está em andamento, fazendo com que o *software* esteja em constante atualização. Tem como ponto forte a capacidade de garantir uma completa rastreabilidade entre requisitos durante todo o ciclo de vida do desenvolvimento do *software*, além de possuir um bom mecanismo para controle de versão dos artefatos, geração de relatórios, visualização de gráficos com dependências entre requisitos, entre outras funcionalidades (Alves, 2007).

Uma vez concluída a etapa de avaliação das ferramentas gratuitas, tem início a segunda etapa do processo de avaliação das ferramentas de gerenciamento de requisitos para o NUSIS.

Sabendo do alto nível de recomendação da RequisitePro para as atividades da engenharia de requisitos, sendo este embasado não somente nas pesquisas bibliográficas realizadas, mas também na opinião técnica dos professores do NUSIS, a segunda etapa da avaliação tem como propósito avaliar comparativamente a ferramenta RequisitePro com a melhor ferramenta avaliada dentre as gratuitas, a OSRMT.

Para esta segunda etapa da avaliação, ambas as ferramentas foram instaladas e configuradas. Os requisitos avaliados também foram diferentes. No atual estágio do projeto, o principal foco da avaliação de uma ferramenta *CASE* para as atividades da engenharia de requisitos consiste em julgar o como e quão bem a mesma atende aos artefatos e funcionalidades propostos para o processo.

Com finalidade de fazer uma avaliação consistente e não tendenciosa, analisou-se criteriosamente os requisitos identificados, pontuando-os de acordo com a sua criticidade para o projeto. Para cada *software* avaliado foi definida uma pontuação máxima de 105 pontos, onde pesos diferentes foram atribuídos aos requisitos. Os pesos estabelecidos para o processo de avaliação foram definidos como:

- 3 – Indispensável
- 2 – Dispensável com restrições
- 1 – Desejável

O sistema de métricas adotado nesta etapa manteve os mesmos critérios estabelecidos anteriormente.

A tabela 10 apresenta os requisitos, sua descrição, e os pesos adotados para cada requisito no modelo de avaliação.

Tabela 10 - Requisitos e pesos para avaliação das ferramentas OSRMT e RequisitePro

| Requisito | Descrição do Requisito | Pesos |
|---|--|-------|
| Aporte aos artefatos: | | |
| Documento de Visão | Dá suporte a criação do artefato de Visão | 2 |
| Glossário | Dá suporte a criação do artefato de Visão | 2 |
| Domínio de Negócio | Dá suporte a criação do artefato Domínio de Negócio | 1 |
| Lista de Requisitos | Dá suporte a criação do artefato Lista de Requisitos | 2 |
| Narrativa de Uso | Dá suporte a criação do artefato Narrativa de Uso | 2 |
| Caso de Uso | Dá suporte a criação do artefato Caso de Uso | 3 |
| Criação de semântica para rastreabilidade | Permite que exista vários tipos de relacionamento, com semântica definida pelo usuário | 3 |
| Rastreabilidade bi-direcional | Possibilita a rastreabilidade bi-direcional entre requisitos | 3 |
| Matriz de rastreabilidade | Exibe uma matriz contendo os relacionamentos existentes de forma clara | 2 |
| Geração de relatórios | Gera e exporta relatórios em forma de documento de requisitos para um formato padrão (.pdf, .doc, .html) | 2 |
| Controle de mudanças | Registra todas as mudanças feitas nos requisitos, indicando quem, o que, quando, onde e porquê houve a alteração | 3 |
| Aprovação de documentos | Identifica se os requisitos foram aprovados ou não | 3 |
| Fórum de discussão | Possibilita a troca de mensagens sobre os requisitos entre os usuários envolvidos | 1 |
| Documentação de utilização do sistema | Existe documentação de utilização do sistema (FAQs, Manuais, Ajuda) | 2 |
| Facilidade de aprendizagem | Possui boa usabilidade, com processos bem definidos que permitam que a forma correta de utilização seja facilmente assimilada pelo usuário | 2 |
| Integração com outras ferramentas | Permite a integração de dados com outras ferramentas utilizadas em atividades das demais fases do ciclo de vida | 2 |

Nas sessões seguintes são apresentadas as avaliações individuais realizadas com as ferramentas com as ferramentas OSRMT e RequisitePro.

5.2 OSRMT

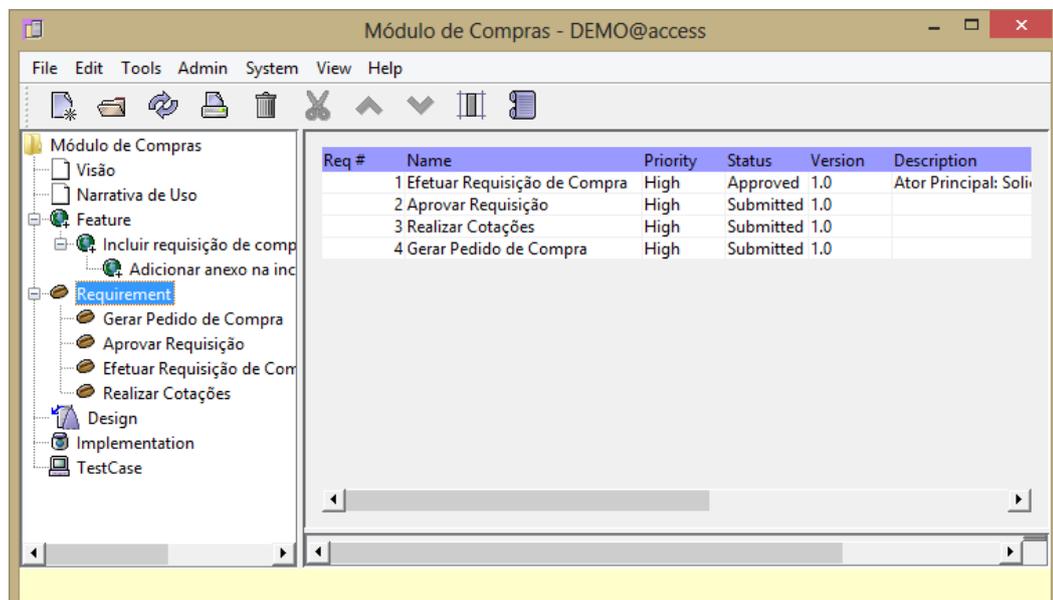
A ferramenta de gerência de requisitos OSRMT (Open Source Requirements Management Tools) é um *software* licenciado sobre os termos da GPL (General Public License), sendo, portanto, uma ferramenta de código aberto.

A avaliação da ferramenta foi realizada utilizando a versão 1.5 do *software*, disponível para download através do endereço <http://sourceforge.net/projects/osrmt>.

Para o processo de engenharia de requisitos, a aplicação possibilita a criação de apenas duas categorias de requisitos: funcionalidades (*features*) e requisitos (*requirements*). Para o seguimento do ciclo de vida do desenvolvimento, existe ainda a possibilidade de criação de requisitos para os artefatos de design, caso de teste e implementação.

Sabe-se que a customização de outros requisitos é viável, no entanto, bastante complexa. Sem experiência na utilização da ferramenta não foi possível instanciar outros tipos de artefato, uma vez que a documentação instalada com o produto é vaga. Cada requisito conta com atributos previamente definidos pela ferramenta, e permitem controles como prioridade, aprovação e complexidade. A figura 32 trás a tela principal da ferramenta OSRMT listando os requisitos criados dentro da categoria “*requirement*” com alguns de seus respectivos atributos.

Figura 32 - Tela principal da ferramenta OSRMT com requisitos da categoria “*Requirement*”



Requisitos desta categoria permitem a inclusão dos passos do cenário de sucesso principal e extensões de um caso de uso diretamente na interface da ferramenta. Outros campos de texto livre podem ser adaptados para descrição de outras seções de um caso de uso. Conta-se ainda com a possibilidade da adição de anexos ao requisito, o que permite a utilização de um modelo externo criado através de um editor de texto e seu posterior anexo junto ao requisito. A figura 33, apresenta a interface para preenchimento dos atributos do requisito, enquanto que a figura 34 demonstra os passos descritos na especificação do caso de uso atrelado ao mesmo.

Figura 33- Interface para preenchimento dos atributos de um requisito da categoria "Requirement"

The screenshot shows a software interface window titled "Requirement Efetuar Requisição de Compra". The window has a tabbed interface with tabs for "Requirement", "Details", "Background", "Use Case", "Dependencies", and "History". The "Requirement" tab is active. The interface contains several input fields and dropdown menus for defining a requirement. The "Requirement" field is filled with "Efetuar Requisição de Compra". Other fields include "Version" (1.0), "Priority" (High), "Requirement Nbr" (1), "Status" (Approved), "Complexity" (Medium), "Assigned" (empty), "Effort" (0.0), and "Category" (Use Case). Below these fields is an "Attachments" table with two entries: "NU01 - Efetuar Requisição de Compra.docx" and "NU02 - Efetuar Requisição de Compra.docx". A "Description" field contains the following text: "Objetivo no Contexto: Solicitante requisita compra de material", "Ator Principal: Solicitante", and "Stakeholders e Interesses: - Solicitante: Deseja que o material requisitado esteja disponível para utilização. - Aprovador: Deseja receber para avaliação somente as requisições que não fore". At the bottom of the window, there is a "Requirement title" label, an information icon, and four buttons: "< Back", "Next >", "Apply", and "Cancel".

| Descripti... | File name | Source |
|--------------|--|--------|
| | NU01 - Efetuar Requisição de Compra.docx | |
| | NU02 - Efetuar Requisição de Compra.docx | |

Figura 34 - Seções para especificação do caso de uso na ferramenta OSRMT

The screenshot shows the 'Requirement Efetuar Requisição de Compra' window in the OSRMT tool. The 'Use Case' tab is selected. The fields are filled with the following text:

- Goal: Efetuar Requisição de Compra
- Context: Solicitante requisita compra de material
- Precondition: Solicitante deve estar logado no sistema e possuir permissão para efetuar requisição de

The Main Flow section contains three steps:

1. Solicitante: preenche formulário para requisição de
2. Sistema: valida dados informados em tela e pede c
3. Solicitante: confirma a inclusão da requisição.

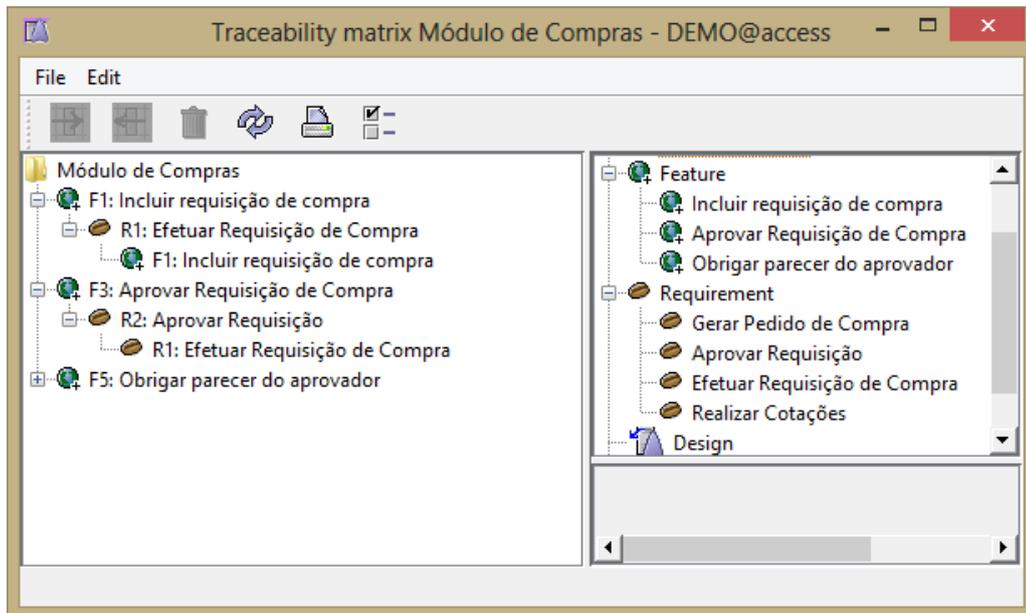
The Alt Flow section contains three steps:

- 2a. Sistema identifica que campos obrigatórios da re
- 3a. Solicitante cancela inclusão de requisição: limpar
- 4a. Sistema identifica regra de aprovação através do

At the bottom left, there is a 'Use case goal' icon. At the bottom right, there are buttons for '< Back', 'Next >', 'Apply', and 'Cancel'.

Segundo (Rodrigues, 2008), a OSRMT é uma ferramenta de gestão de requisitos desenhada para garantir uma completa rastreabilidade durante o ciclo de vida de desenvolvimento do *software* para funcionalidades, requisitos, interfaces, implementação e casos de testes. E isso de fato é algo que pôde ser constatado. A matriz de rastreabilidade é de fácil utilização, possibilitando rastrear os requisitos bidirecionalmente. No entanto, a criação da rastreabilidade através do relacionamento entre requisitos tem de ser realizada dentro de cada um dos requisitos, não possuindo funcionalidade que permita o estabelecimento das relações dentro da própria matriz. Através da figura 35 pode ser observada a matriz de rastreabilidade para os requisitos criados durante a avaliação da ferramenta.

Figura 35 - Matriz de Rastreabilidade entre requisitos na ferramenta OSRMT



Outras características e funcionalidades importantes identificadas durante a avaliação da ferramenta foram:

- Controle de versão dos requisitos;
- Geração de alguns relatórios de forma automática
- Administração de grupos, usuários, acessos e projetos;
- Fórum de discussões inexistente;
- Inexistência de forma padrão de integração com aplicações externas.

A usabilidade da ferramenta deixa um pouco a desejar, bem como a documentação instalada juntamente com a ferramenta. Não foi localizado outro material que suportasse o aprendizado da aplicação.

5.3 REQUISITEPRO

O RequisitePro é uma ferramenta para gerência de requisitos desenvolvida pela Rational, de propriedade da IBM. A ferramenta faz parte de uma suíte de aplicativos da Rational para apoio ao ciclo de vida do desenvolvimento de *software*, tendo como base o processo unificado, abordado no capítulo 3.2 deste trabalho.

A avaliação da ferramenta foi realizada utilizando a versão 7.1.2.0 do produto encontrada para *download* diretamente no site da IBM em www.ibm.com/software/products/br/pt/reqpro.

Assim como no processo unificado, as atividades no RequisitePro são suportadas pela geração de uma série de artefatos e modelos para a construção dos mesmos. Uma das características evidenciadas através da avaliação da ferramenta foi sua forte integração com o editor de textos Microsoft Word.

O RequisitePro permite manter os requisitos diretamente nos documentos, associando estes ao banco de dados da aplicação. Desta forma, torna-se prático e trivial definir os requisitos diretamente no editor de textos, e utilizar uma aba de suplementos incorporada ao Microsoft Word em função do RequisitePro para gerenciá-los. Desta forma, no processo ideal de utilização da ferramenta, na base de dados da aplicação ficam armazenados apenas os atributos e propriedades do requisito, enquanto que a descrição textual fica contida no próprio documento. As figuras 36 e 37 apresentam na sequência a aba de suplementos do editor de texto com as opções agregadas pelo RequisitePro, e ação da ferramenta após acionar a opção para criação do requisito “Realizar Cotações”.

Figura 36 - Aba de suplementos da ferramenta RequisitePro no Microsoft Word

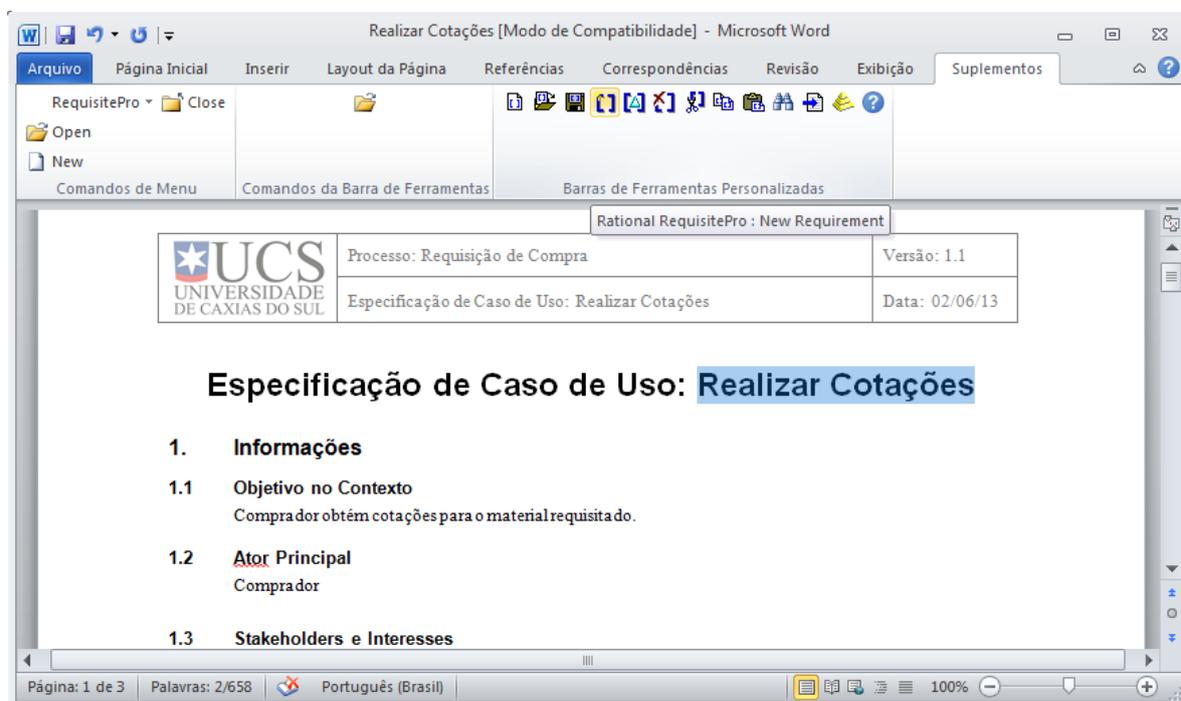
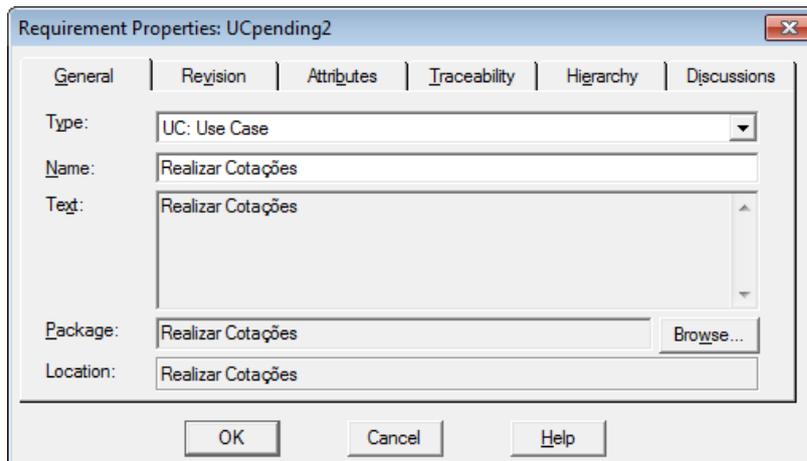


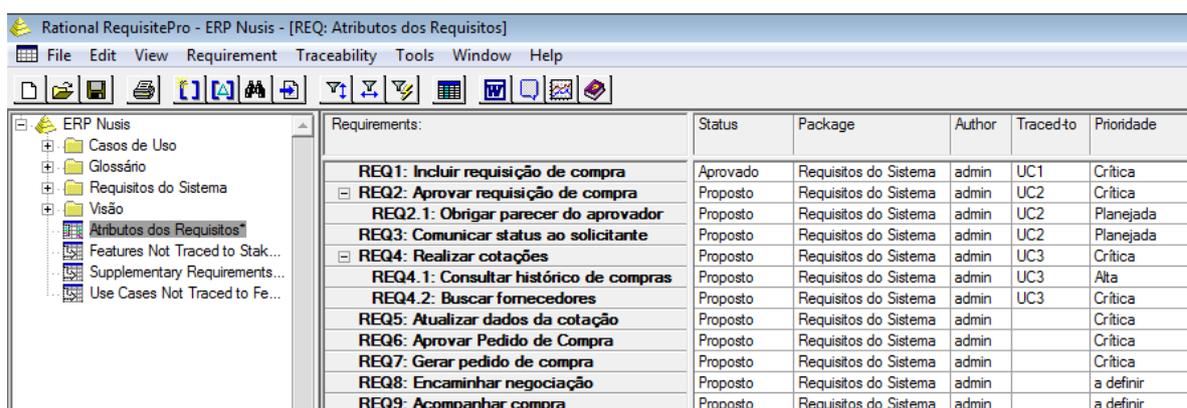
Figura 37 - Interface para criação de requisito na ferramenta RequisitePro



Para suprir os diferentes tipos de artefatos que são essenciais a um projeto, a ferramenta oferece a possibilidade de customizar tipos de requisitos e documentos. Desta forma, a aplicação não fica engessada somente aos artefatos por ela sugeridos, o que faz com que qualquer tipo de artefato possa ser suportado.

Cada tipo de requisito pode possuir atributos também customizáveis, de acordo com as informações que se deseja gerir. Isto torna viável a criação de atributos de classificação, como prioridade, dificuldade, status de aprovação, entre outros. Na figura 38 é possível visualizar a interface principal do RequisitePro e os atributos dos requisitos.

Figura 38 - Interface principal da ferramenta RequisitePro



Através da criação de requisitos para os diferentes tipos de artefatos, posteriormente pode-se estabelecer uma relação entre requisitos. O estabelecimento dessa relação através da rastreabilidade é algo essencial em qualquer projeto e demasiadamente simples de ser realizado no RequisitePro. Para isso utiliza-se a matriz de rastreabilidade, que pode ocorrer bi-

direcionalmente entre quaisquer tipos de requisitos. A figura 39 apresenta a matriz de rastreabilidade elaborada entre requisitos funcionais do sistema e seus casos de uso durante a presente avaliação

Figura 39 - Matriz de rastreabilidade na ferramenta RequisitePro

| Relationships: - direct only | REQ1: Incluir requisição de... | REQ2: Aprovar requisição de... | REQ2.1: Obrigar parecer do... | REQ3: Comunicar status ao... | REQ4: Realizar cotações | REQ4.1: Consultar histórico de.. | REQ4.2: Buscar fornecedores | REQ5: Atualizar dados da... | REQ6: Aprovar Pedido de... | REQ7: Gerar pedido de compra | REQ8: Encaminhar negociação | REQ9: Acompanhar compra |
|-----------------------------------|--------------------------------|--------------------------------|-------------------------------|------------------------------|-------------------------|----------------------------------|-----------------------------|-----------------------------|----------------------------|------------------------------|-----------------------------|-------------------------|
| UC1: Efetuar Requisição de Compra | | | | | | | | | | | | |
| UC2: Aprovar Requisição de Compra | | | | | | | | | | | | |
| UC3: Realizar Cotações | | | | | | | | | | | | |

Algumas das principais características e funcionalidades identificadas para a ferramenta:

- Controle de versões dos requisitos;
- Fácil integração com outras ferramentas do mesmo fornecedor;
- Completa possibilidade de customização de atributos, documentos e requisitos;
- Permite definir relacionamentos entre requisitos diretamente na matriz de rastreabilidade;
- Controle sobre as alterações - quando alterado determinado requisito, todos os requisitos atrelados ao mesmo são sinalizados;
- Fórum de discussões de requisitos;
- Possibilidade de geração de relatórios através do filtro dos dados.

(Theilacker, 2008) cita mais algumas funcionalidades apresentadas pela ferramenta:

- Permite visualização do impacto das alterações nos requisitos;
- Disponibiliza informações estatísticas dos requisitos;
- Mantém visões que permitem uma análise criteriosa dos requisitos especificados;
- Agrupamento dos artefatos em diretórios;

- Permite relacionamento entre requisitos e grupos de segurança;
- Permite acesso distribuído aos requisitos.

Como referência para o aprendizado da ferramenta utilizou-se a documentação instalada junto ao produto em conjunto a outros tutoriais facilmente encontrados na internet. A própria documentação do produto por sua vez já se mostrou completa, suprimindo praticamente todas as dúvidas surgidas.

5.4 RESULTADOS OBTIDOS

Os resultados obtidos a partir da avaliação individual das ferramentas OSRMT e RequisitePro quanto aos requisitos estabelecidos para avaliação são apresentados na tabela 11. A pontuação atribuída é expressa considerando os pesos dos requisitos e o sistema de métricas anteriormente estabelecido.

Tabela 11 - Resultados obtidos a partir da avaliação das ferramentas OSRMT e RequisitePro

| Requisito | RequisitePro | OSRMT |
|---|---------------------|--------------|
| Aporte aos artefatos: | | |
| Documento de Visão | 3 (2) = 6 | 0 (2) = 0 |
| Glossário | 3 (2) = 6 | 0 (2) = 0 |
| Domínio de Negócio | 0 (1) = 0 | 0 (1) = 0 |
| Lista de Requisitos | 3 (2) = 6 | 2 (2) = 4 |
| Narrativa de Uso | 3 (2) = 6 | 2 (2) = 4 |
| Caso de Uso | 3 (3) = 9 | 2 (3) = 6 |
| Criação de semântica para rastreabilidade | 3 (3) = 9 | 3 (3) = 9 |
| Rastreabilidade bi-direcional | 3 (3) = 9 | 3 (3) = 9 |
| Matriz de rastreabilidade | 3 (2) = 6 | 1 (2) = 2 |
| Geração de relatórios | 3 (2) = 6 | 3 (2) = 6 |
| Controle de mudanças | 3 (3) = 9 | 3 (3) = 9 |
| Aprovação de documentos | 3 (3) = 9 | 3 (3) = 9 |
| Fórum de discussão | 3 (1) = 3 | 0 (1) = 3 |
| Documentação de utilização do sistema | 3 (2) = 6 | 1 (2) = 2 |
| Facilidade de aprendizagem | 2 (2) = 4 | 2 (2) = 4 |
| Integração com outras ferramentas | 3 (2) = 6 | 0 (2) = 0 |
| TOTAL | 100 | 67 |

Perante a tais resultados, pode-se afirmar que a ferramenta *CASE* que melhor satisfaz os requisitos levantados de acordo com as atividades da engenharia de requisitos proposta para o NUSIS é a RequisitePro. Além de se demonstrar uma ferramenta mais sólida e completa, outro atrativo é a facilidade de se encontrar conteúdo relacionado na internet, com tutoriais, aplicações em situações reais e experiências de utilização compartilhadas por usuários.

A RequisitePro apresenta uma série de vantagens sobre a ferramenta OSRMT. Muitos são os pontos que podem ser destacados. Na RequisitePro existe a possibilidade de criação de requisitos e artefatos, tantos quantos se fizerem necessários. Pode-se estabelecer uma semântica para a rastreabilidade, bem como o relacionamento entre requisitos pode ser realizado através da própria matriz, o que torna o trabalho de manter a correta rastreabilidade muito mais fácil. Por ser uma ferramenta pertencente a uma suíte de aplicativos para o desenvolvimento de *software*, a RequisitePro permite integração ágil com outras aplicações que podem ser utilizadas nas atividades da engenharia de requisitos, como a Rational Rose que permite a criação de uma série de diagramas UML, e a Rational ClearCase, utilizada para gerenciamento de versões e processos de auditoria.

Enquanto que a OSRMT não possui integração com nenhum editor de texto, a RequisitePro por sua vez integra-se com o Microsoft Word. Através dele é possível definir modelos para as especificações de caso de uso, narrativas de uso, especificações suplementares, glossário, visão, e qualquer outro artefato que se faça necessário. Embora não seja de cunho obrigatório, a utilização de tal editor de texto é altamente recomendada, chegando ao ponto de poder declarar que a ferramenta foi concebida para trabalhar em conjunto ao editor.

Ainda sobre as duas ferramentas, a usabilidade, forma de organização e visualização das informações, possibilidade de parametrização, comunicação, entre outros fatores, fez com que a experiência de utilização da RequisitePro tenha sido muito mais agradável. Nela, é possível visualizar os requisitos e a rastreabilidade de diferentes maneiras. A criação de atributos é fácil e a comunicação da ferramenta com o usuário é detalhada. Pode-se parametrizar os campos que serão mostrados em tela. A RequisitePro conta ainda com um fórum para discussão dos requisitos, algo que inexiste na OSRMT.

5.5 CONSIDERAÇÕES FINAIS

Neste capítulo foi realizada a análise e avaliação das ferramentas *CASE* destinadas às atividades da engenharia de requisitos com base nas normas ISO/IEC 14598 e ISO/IEC 9126, visando identificar àquela que melhor satisfaz às necessidades do NUSIS de acordo com o modelo de processo proposto.

O processo de avaliação considerou a premissa estabelecida para o projeto, onde devem ser utilizados *softwares* que não acarretem custos aos envolvidos. Seguindo este critério, duas ferramentas foram avaliadas individualmente através da aplicação de uma parte do estudo de caso que será visto na sequência deste trabalho. A OSRMT, melhor ferramenta gratuita de acordo com (Alves, 2007), e a RequisitePro, proveniente de uma parceria da Universidade de Caxias do Sul que isenta de custos sua utilização para fins acadêmicos.

Através dos resultados obtidos para as avaliações realizadas, constou-se a superioridade da RequisitePro sobre a OSRMT ao considerar os requisitos elecandos de acordo com o modelo de processo proposto. Tal ferramenta se mostrou capaz de atender satisfatoriamente todos quesitos. A única particularidade que deve evidenciada é a integração existente entre a RequisitePro e o editor de textos Microsoft Word. Embora a integração entre as aplicações não seja uma exigência para o funcionamento da ferramenta *CASE*, a utilização do editor de texto está relacionada a muitas das funcionalidades e praticidades apresentadas pela solução, onde a não utilização da mesma limita algumas das funções existentes, principalmente quanto aos artefatos que são construídos ao longo das atividades da engenharia de requisitos. Esse ressalvo cabe aqui uma vez que o Microsoft Word é uma ferramenta proprietária, necessitando de licença paga para sua utilização, o que vem em confronto ao princípio do emprego de soluções livres de custos.

Por fim, valendo-se da avaliação que decretou o RequisitePro como a melhor ferramenta *CASE* para gerência de requisitos do NUSIS, na sequência deste trabalho será apresentada a aplicação do modelo de processo proposto no capítulo 4 utilizando a respectiva ferramenta.

6 APLICAÇÃO DO MODELO DE PROCESSO PROPOSTO

O presente capítulo tem por objetivo aplicar o modelo de processo de engenharia de requisitos proposto no capítulo 4 em um estudo de caso utilizando a ferramenta de gerenciamento de requisitos RequisitePro, para desta forma avaliar a aferição do modelo e da ferramenta ao ambiente experimental de desenvolvimento de *software* do NUSIS.

O estudo de caso contou inicialmente com a definição de um conjunto de funcionalidades do ERP para sua aplicação. Após a análise das diferentes áreas de negócio e processos abrangidos por um ERP, foi selecionado o processo de requisição de compra de materiais de consumo e expediente para que se realizasse a instanciação do modelo de processo.

O referido processo contempla a requisição inicial de uma compra, que pode ser oriunda de diversos setores e pessoas, a aprovação dessa requisição por um responsável, a obtenção de cotações de preços junto aos fornecedores cadastrados, a análise e escolha da melhor cotação e a geração do pedido de compra e seu encaminhamento para o fornecedor, além do acompanhamento da compra até que a mesma seja recebida pela empresa. A compra de matéria-prima para o setor de produção e de material para os setores de vendas e distribuição não está englobada no contexto do processo elencado para a aplicação do estudo de caso. Essa categoria de compra é atribuída ao módulo de Planejamento de Necessidade de Material (MRP), responsável por suprir esse tipo de demanda e assegurar a disponibilidade desses recursos através do monitoramento de estoques. As necessidades e características relacionadas ao cadastro, manutenção e gerenciamento das informações inerentes aos fornecedores também é algo enquadrado fora do escopo do processo de requisição de compra e conseqüentemente não está compreendido no presente estudo de caso.

As próximas seções descrevem a adoção do processo proposto no estudo de caso, passando pelas atividades de Concepção e Análise do Problema, Levantamento e Compreensão dos Requisitos, Negociação de Requisitos, Especificação e Modelagem dos Requisitos, Validação dos Requisitos e Gerência de Requisitos. Ao longo dessas atividades foi utilizada a versão 7.1.2.0 da ferramenta RequisitePro, versão esta também empregada durante a análise das ferramentas *CASE* vista no capítulo 5, e disponível para *download* diretamente no endereço da IBM na internet.

6.1 CONCEPÇÃO E ANÁLISE DO PROBLEMA

Desde o primeiro momento, onde buscou-se a definição da área de atuação para o estudo de caso, o contato com a bibliografia especializada demonstrou ser essencial para o entendimento do problema e suas atividades subsequentes. Através da leitura de publicações da área de Administração (Arnold, 1999)(Baily, 2000)(Francischini, 2010)(Gonçalves, 2010), mais precisamente voltadas à administração de suprimentos e materiais, pôde-se formar uma base preliminar de conhecimento perante o negócio, que por sua vez será o objeto de estudo das atividades desempenhadas.

Compreendendo ainda que superficialmente os elementos envolvidos no processo de Requisição de Compras de uma empresa, e antes mesmo do primeiro contato oficial com o professor orientador para o início da atividade de concepção e análise do problema, procurou-se analisar sistemas correlatos bem conceituados perante o negócio. Além desta análise propiciar um maior entendimento sobre o fluxo de atividades desencadeadas em um sistema por conta da inclusão de uma nova requisição de compra, que vai desde a solicitação inicial de compra até a geração do pedido e seu devido acompanhamento, tal avaliação permitiu também a assimilação de outros conceitos importantes acerca do funcionamento de um ERP.

Após inserção no contexto do projeto e tendo uma base de conhecimento do negócio envolvido, o conjunto de perguntas genérico formulado para esta etapa trazido na seção 4.3.1 deste trabalho, foi levado para o primeiro encontro junto ao professor orientador. Tais perguntas serviram de roteiro para o encontro e demonstraram-se extremamente úteis, auxiliando a fixar o escopo do projeto e estabelecer os limites do trabalho. Cabe ressaltar neste caso que as características macro do projeto que visa o desenvolvimento de *software* dentro do NUSIS já eram conhecidas antes mesmo do início das atividades do estudo de caso, uma vez que estas tiveram de ser consideradas para a construção da modelo de processo de engenharia de requisitos proposto neste trabalho. No entanto, ao pensar que o aluno no papel de analista de sistemas estará ingressando neste momento no projeto, as perguntas e suas respostas servirão de mecanismo para o esclarecimento dúvidas, compreensão de características, identificação dos objetivos globais e benefícios do projeto.

Com base na conversa mantida com o professor orientador no primeiro encontro, ainda que sem maiores formalidades, e nas respostas obtidas para os questionamentos, o documento de visão pôde ser criado e ter suas primeiras informações adicionadas. Tal artefato será único para todo o projeto do NUSIS, desta forma, crescerá de forma gradativa de acordo com o andamento das atividades.

Por se tratar de um sistema ERP, caracterizado por sua complexidade e amplitude, e também considerando o fato que o *software* crescerá de maneira incremental de acordo com os ciclos de desenvolvimento, decidiu-se por subdividir o documento de visão em seções específicas por processo. Desta forma, cada processo do ERP que tenha passado pelas atividades da engenharia de requisitos possuirá uma seção própria com as informações pertinentes aos seus objetivos específicos, conjunto de necessidades, *stakeholders*, usuários e recursos do sistema. Levando em conta o seguimento das atividades do desenvolvimento do *software*, essa alteração no documento de visão possibilitará uma melhor visão individual de cada um dos processos, facilitando as atividades futuras relacionadas aos mesmos. A figura 40 apresenta um trecho da seção extraída do documento de visão onde estão contidas as necessidades dos *stakeholders* para o processo de requisição. Tal documento pode ser encontrado na íntegra nos anexos deste trabalho.

Figura 40 - Trecho da seção de Necessidades dos *Stakeholders* extraída do documento de Visão

| Necessidade | Descrição | Proposta de Solução | Prioridade |
|--------------------------------|---|---|------------|
| Incluir requisição de compra | Possibilitar a inclusão de requisições de materiais. | A função para inclusão da requisição de compra permitirá que qualquer funcionário com permissão requisite material. | Crítica |
| Manter o solicitante informado | Manter o solicitante informado sobre o andamento de sua requisição de compra. | Comunicar o solicitante do andamento de sua requisição através de e-mails disparados automaticamente pelo sistema. | Planejada |
| Aprovar requisição de compra | Possibilitar que as requisições passem por etapa de aprovação antes de chegar ao setor de compras. | Para toda requisição será verificada a necessidade de aprovação, fazendo com que uma requisição só caia para o setor de Compras após ser aprovada, ou então na inexistência de regras de aprovação. | Crítica |
| Aprovação por níveis | Permitir que a aprovação das requisições ocorra em diferentes níveis, uma vez que uma requisição pode ter de ser aprovada por mais de um responsável. | Sistema verificará regras de negócio para aprovação da requisição, criando uma espécie de trilha de aprovação pela qual a requisição obrigatoriamente tem que passar antes de chegar ao setor de Compras. Cada uma das regras estabelecidas será observada pelo sistema, seguindo a ordem estabelecida, onde primeiro serão verificadas as regras por código/tipo de material, em seguida as regras por centro de custo, e por fim as regras por faixa de valor. | Crítica |

Outro refinamento realizado no documento de visão está relacionado à forma de priorização das necessidades. Conforme pode ser observado na figura 40, as necessidades são

classificadas de acordo com sua prioridade na mesma tabela em que são descritas, tornando a existência de uma seção exclusiva para priorização das necessidades supérflua.

Para cada uma das necessidades dos *stakeholders* elencadas no documento de visão foi criado um requisito diretamente na ferramenta RequisitePro. Posteriormente tais requisitos permitirão estabelecer a rastreabilidade das necessidades com os requisitos funcionais do sistema. A figura 41 demonstra tais requisitos consultados através da ferramenta CASE.

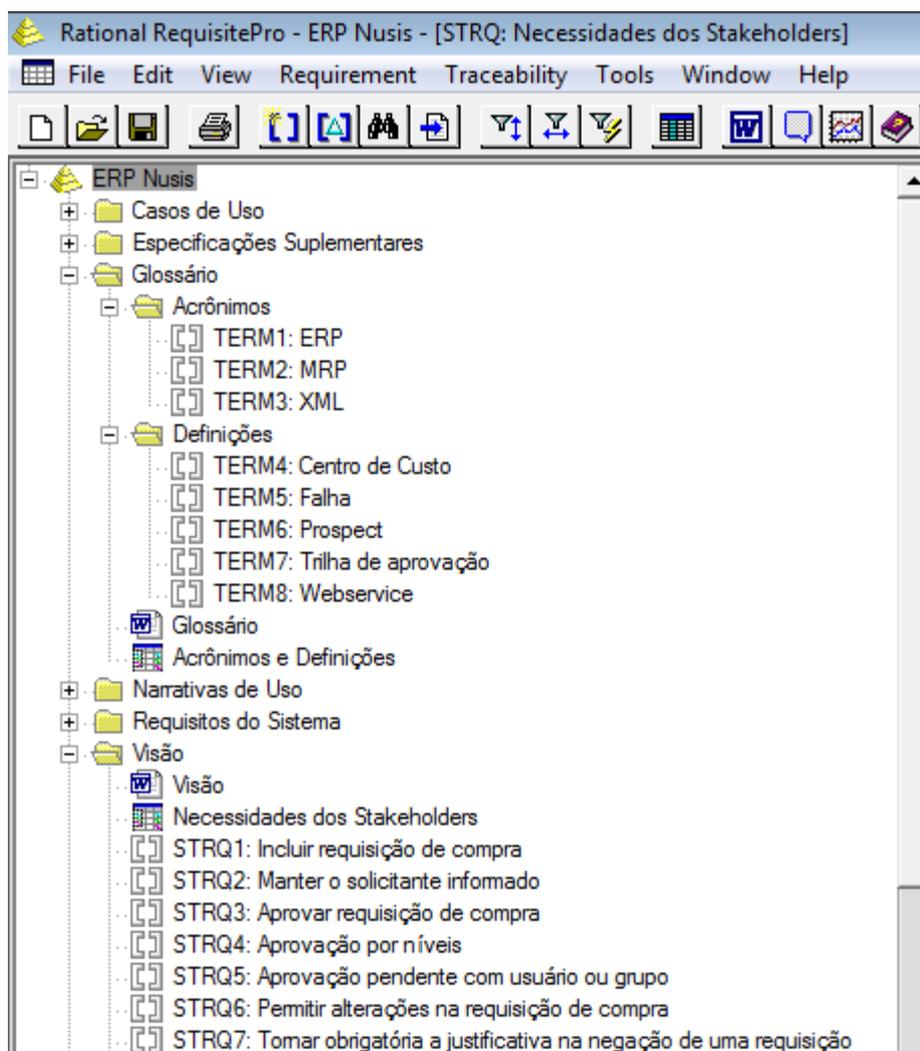
Figura 41 - Requisitos com as Necessidades dos *Stakeholders* visualizados através do RequisitePro

| Requirements: | Proposta de Solução | Criticidade | Author | Revision |
|---|--|-------------|--------|----------|
| STRQ1: Incluir requisição de compra Possibilitar a inclusão de requisições de materiais. | A função para inclusão da requisição de compra permitirá que qualquer funcionário com permissão requisite material. | Crítica | admin | 1.0003 |
| STRQ2: Manter o solicitante informado Manter o solicitante informado sobre o andamento de sua requisição de compra. | Comunicar o solicitante do andamento de sua requisição através de e-mails disparados automaticamente pelo sistema. | Planejada | admin | 1.0002 |
| STRQ3: Aprovar requisição de compra Possibilitar que as requisições passem por etapa de aprovação antes de chegar ao setor de compras. | Para toda requisição será verificada a necessidade de aprovação, fazendo com que uma requisição só caia para o setor de Compras | Crítica | admin | 1.0003 |
| STRQ4: Aprovação por níveis Permitir que a aprovação das requisições ocorra em diferentes níveis, uma vez que uma requisição pode ter... | Será possível parametrizar na regra de aprovação quem será o responsável, usuário ou grupo. | Crítica | admin | 1.0003 |
| STRQ5: Aprovação pendente com usuário ou... Permitir que as requisições fiquem pendentes de aprovação com um grupo de usuários ou então com... | Na etapa de aprovação, o usuário responsável poderá alterar informações da requisição. | Planejada | admin | 1.0002 |
| STRQ6: Permitir alterações na requisição de... Permitir que o responsável de aprovação altere e/ou acrescente informações à requisição de compra. | Ao negar uma requisição, obrigatoriamente o usuário aprovador terá de incluir um parecer/justificativa para concluir a operação. | Alta | admin | 1.0003 |
| STRQ7: Tomar obrigatória a justificativa na... Tomar obrigatória a justificativa com um parecer por parte do responsável de aprovação em caso da... | A realização de cotações ocorrerá em dois momentos distintos no sistema. Primeiramente serão selecionados os fornecedores para os | Planejada | admin | 1.0002 |
| STRQ8: Permitir a realização de cotações Permitir a realização de cotações para a requisição de compra. | Haverá integração com o Módulo de Gerenciamento de Fornecedor que permitirá a busca dos fornecedores cadastrados | Crítica | admin | 1.0003 |
| STRQ9: Possibilitar a busca pelos... Possibilitar ao Comprador na etapa de realização das cotações a busca pelos fornecedores cadastrados... | Através da integração com o Módulo de Gerenciamento de Fornecedor serão retomadas as informações dos fornecedores | Crítica | admin | 1.0003 |
| STRQ10: Apresentar os fornecedores... Apresentar ao Comprador os fornecedores cadastrados para o material requisitado de acordo com o nível de... | Na etapa de cotações, será disponibilizada opção que permitirá ao Comprador consultar o histórico de compras do material, com os últimos | Alta | admin | 1.0003 |

A criação das necessidades dos *stakeholders* no RequisitePro deu início a atividade de gerência de requisitos, que ocorre paralelamente as demais atividades da engenharia de requisitos.

Simultaneamente ao documento de visão foi criado o glossário de terminologias do sistema, com os primeiros termos na linguagem do cliente identificados para o negócio e relevantes para o desenvolvimento do *software*. Na ferramenta RequisitePro, os mesmos foram concebidos diretamente abaixo da estrutura criada para o projeto ERP NUSIS. A figura 42 demonstra a árvore de diretórios do projeto, destacando os artefatos visão e glossário, e seus requisitos.

Figura 42 - Árvore de diretórios do projeto no RequisitePro



6.2 LEVANTAMENTO E COMPREENSÃO DOS REQUISITOS

Com a conclusão da etapa de concepção e análise do problema, o foco que até então recaía sobre os objetivos macros do projeto é transferido para as necessidades específicas de negócio existentes.

No modelo de processo proposto enfatizou-se a importância do envolvimento e participação de profissionais com conhecimento sobre as áreas fim na atividade de levantamento e compreensão dos requisitos, podendo estes serem professores de outros centros e até mesmo funcionários da universidade. Embora tal ideia seja válida e com certeza benéfica ao processo, não pôde-se contar com este tipo de participação na execução da atividade no presente estudo de caso.

Dentre as técnicas de levantamento de requisitos propostas, a verificação de documentos se mostrou determinante para o sucesso da atividade. A utilização de bibliografia relacionada à área de compras, a busca por material de estudo em páginas da *WEB* e a análise de documentos do processo de requisição de compra de sistemas já existentes no mercado possibilitaram um melhor entendimento acerca das necessidades de usuários e *stakeholders* em relação ao processo, e possibilitaram que as mesmas fossem transcritas através de requisitos funcionais e não funcionais para o sistema. Foi com base nesse conhecimento do negócio adquirido através da técnica de verificação de documentos que os *workshops* de requisitos e as entrevistas com o professor orientador foram conduzidos.

Para o primeiro contato com o professor orientador dentro da atividade de levantamento e compreensão dos requisitos, foi elaborada uma série de perguntas focadas na área de negócio para que fosse possível assim exercer uma entrevista a fim de se coletar requisitos. Tal entrevista abriu os caminhos para o seguimento das atividades do levantamento, servindo para que as dúvidas surgidas através da verificação de documentos pudessem ser esclarecidas. Algumas das perguntas construídas são apresentadas na tabela 12.

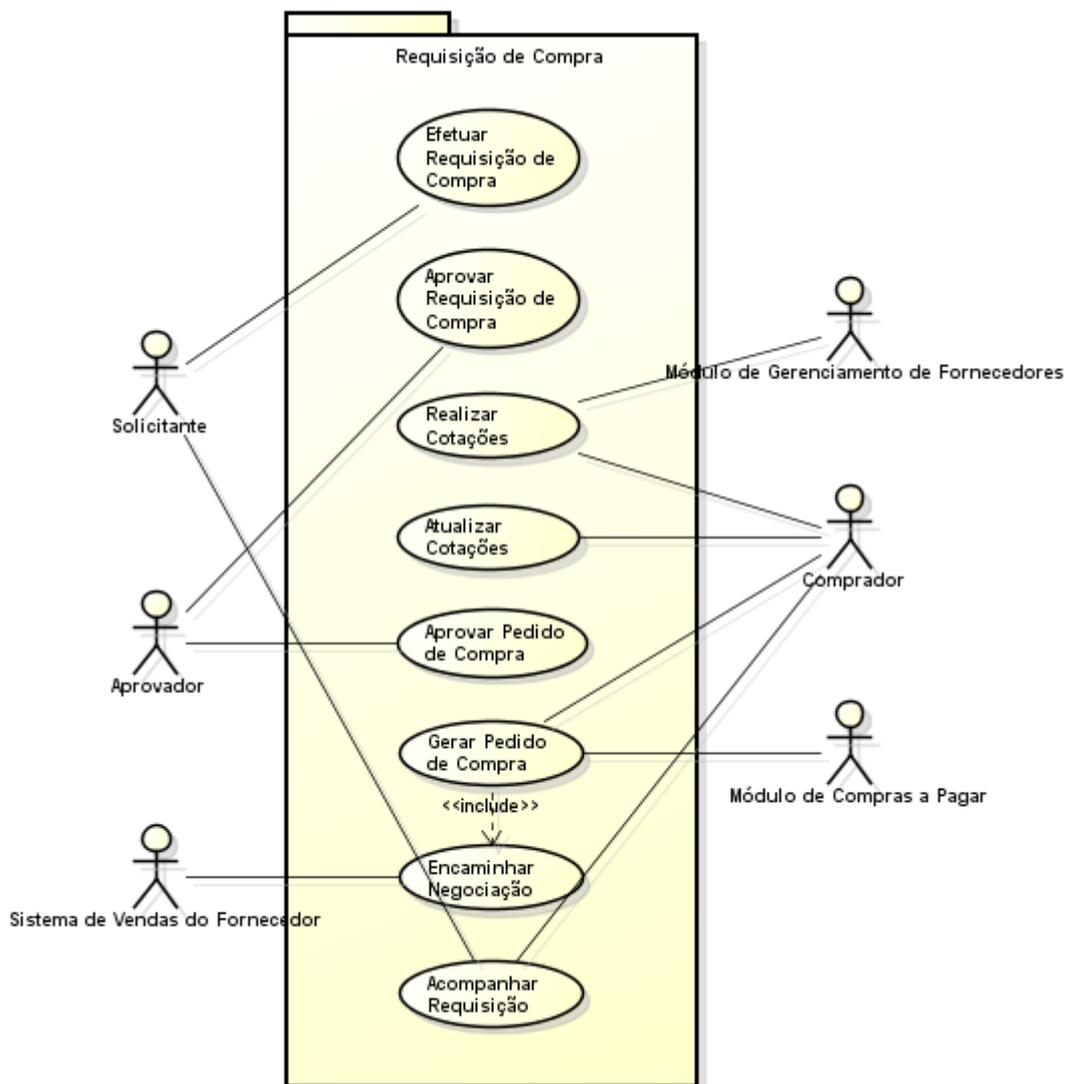
Tabela 12 - Questionamentos elaborados para condução de entrevista

| |
|---|
| Até onde vai o processo de requisição de compra? |
| Quais as atividades englobadas pelo processo? |
| Que tipo de material pode ser solicitado em uma requisição? |
| A requisição de compra pode ser realizada por quais setores dentro da empresa? |
| Quais funcionários podem requisitar uma compra? |
| As requisições de compra têm de passar por processo de aprovação? |
| Neste caso, quem são esses aprovadores? Existem regras para as aprovações? Uma aprovação pode ocorrer em N níveis, com responsáveis diferentes aprovando? |

Após essa primeira entrevista, um esboço do diagrama de casos de uso foi criado utilizando a notação UML. A criação do mesmo teve como objetivo facilitar o entendimento acerca dos atores envolvidos e etapas do processo de requisição de compra, já servindo como base para que posteriormente se construíssem os casos de uso. No processo proposto, o diagrama de casos de uso só seria criado após a conclusão da etapa de especificação e

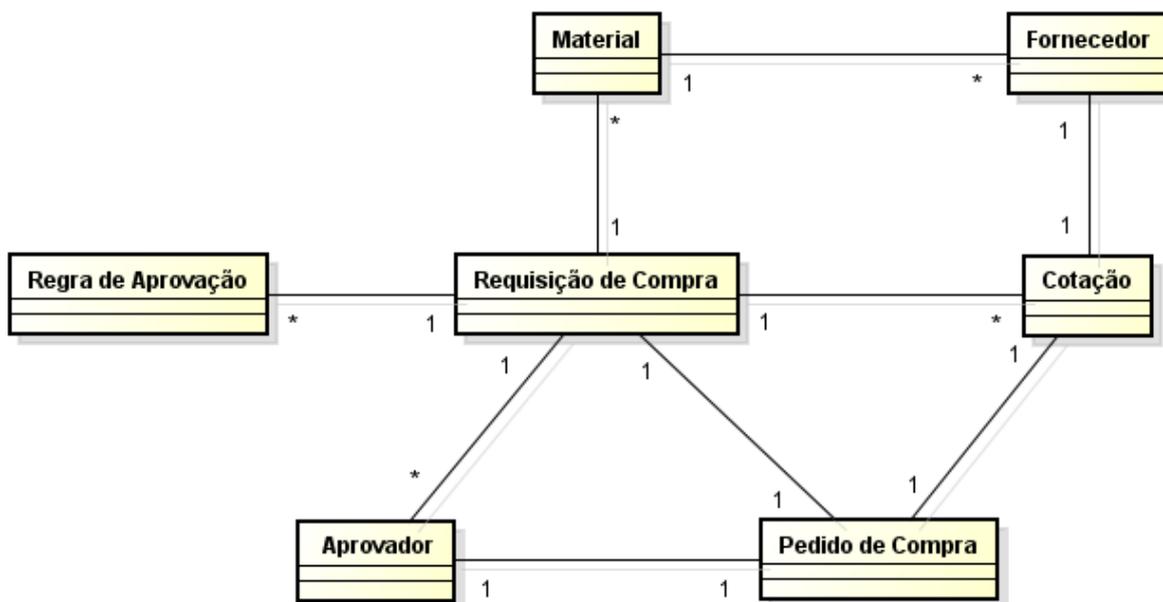
modelagem dos requisitos, no entanto, a sua antecipação para que já começasse a ser esboçado ainda na atividade de levantamento e compreensão dos requisitos se demonstrou extremamente eficaz e útil para as atividades. Cabe se ressaltar que neste momento apenas foi criado um esboço para o diagrama de casos de uso, sendo que a sua versão final só foi obtida após concluída a especificação dos requisitos. A figura 43 demonstra o diagrama de casos de uso do processo de requisição de compra.

Figura 43 - Diagrama de casos de uso do processo de Requisição de Compra



Com o conhecimento adquirido a esta altura, outro artefato que já pôde ser criado neste momento foi o modelo de domínio. Através de um diagrama de classes da UML foram capturados e representados os principais conceitos e entidades relacionados ao negócio no qual o processo de requisição de compra está inserido. Através da sua simples representação, apresentando apenas as classes em alto nível, com nomes e relacionamentos de maior relevância, o modelo de domínio permitiu uma melhor percepção do contexto de negócio em que se trabalhava. A figura 44 apresenta o modelo de domínio criado e a multiplicidade entre as classes.

Figura 44 - Modelo de domínio do processo de Requisição de Compra



Para o próximo encontro com o professor orientador, além do esboço do diagrama de casos de uso e do modelo de domínio, também foram rascunhadas as primeiras narrativas de uso. Devido ao pouco tempo dos encontros semanais, ficou impraticável conceber as narrativas durante os *workshops* de requisitos. Desta forma, as narrativas foram produzidas e levadas para que fossem discutidas conjuntamente, para assim realizar os ajustes necessários. As narrativas de uso se demonstraram úteis à medida que possibilitaram uma percepção mais aguçada sobre os requisitos, com situações reais de utilização do sistema e funcionalidades pretendidas. Com o avanço da convergência de ideias entre o cliente, representado pelo orientador, e o analista de sistemas, configurado pelo aluno, as narrativas de uso propuseram uma forma dinâmica de captação de requisitos, que foi muito bem aceita por ambas as partes.

Ao modelo proposto para as narrativas uma seção intitulada “Descrição” foi adicionada, para que nela pudessem ser sintetizados os objetivos macro da narrativa, de modo a descomplicar sua interpretação por parte do leitor. A figura 45 demonstra o conteúdo da narrativa de uso “Requisitar notebook para funcionário”, criada após os encontros realizados.

Figura 45 - Narrativa de uso "Requisitar *notebook* para funcionário"

| | | |
|--|---|----------------|
| | Processo: Requisição de Compra | Versão: 1.0 |
| | Narrativa de Uso: Requisição de notebook para funcionário | Data: 25/05/13 |

Narrativa de Uso: Requisição de notebook para funcionário

- 1. Narrativa de Uso**
 - 1.1 Descrição**

Coordenador da equipe de vendas que deseja requisitar a aquisição de um notebook para um novo funcionário no setor.
 - 1.2 Fluxo Normal**

Recentemente um novo funcionário foi contratado para o setor de vendas. Este novo funcionário atuará em campo, visitando *prospects* e tirando pedidos em clientes da região. Pelos próximos trinta dias ele permanecerá em treinamento junto a outros funcionários da empresa. Passado esse período ele passará a exercer as atividades que competem a sua função. Para isso, o coordenador da equipe de vendas necessita requisitar um notebook novo para seu funcionário. O coordenador não possui qualquer conhecimento de informática para poder detalhar aspectos de *hardware* e *software* para a nova aquisição, e entende que tal avaliação tenha de ficar sob responsabilidade de um funcionário especializado do setor de TI. Após algumas trocas de e-mail com funcionários da TI, o coordenador de vendas preenche o formulário de requisição de compra no sistema, detalhando no campo de “Justificativa” os objetivos de utilização deste novo equipamento, indicando o funcionário que o utilizará, e quais os requisitos de *hardware* e *software* necessários de acordo com o parecer técnico obtido da TI. Após submeter a requisição de compra, lhe é retornado um número identificador e o status que a requisição encontra-se pendente aprovação com o responsável pelo seu centro de custo.
 - 1.3 O que pode dar errado**

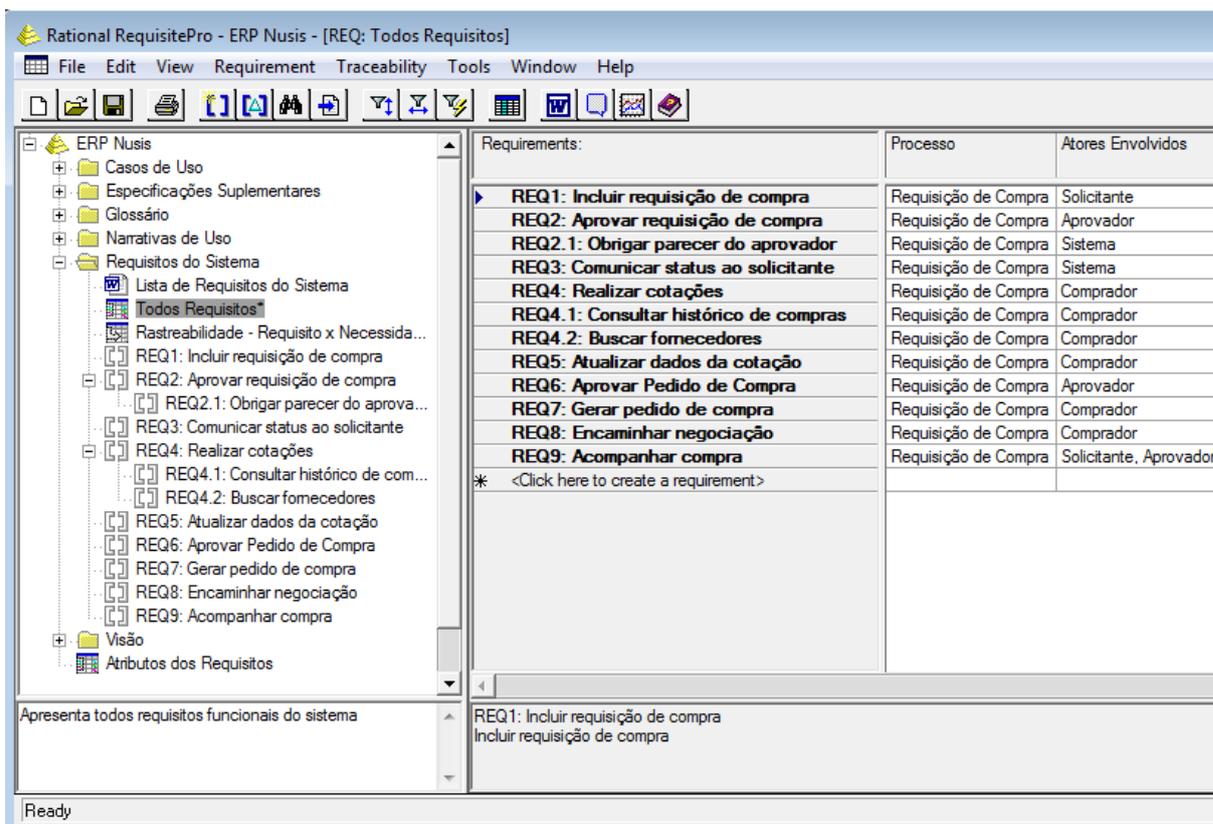
A requisição que deveria cair para aprovação do centro de custo foi autorizada prontamente pelo sistema, caindo direto na etapa de Cotações. Neste caso, devem ser revistas as regras de aprovação parametrizadas.

Com o seguimento das tarefas, o documento de visão e o glossário foram refinados e enriquecidos com novas informações. No documento de visão as necessidades coletadas e suas possíveis soluções foram sendo adicionadas, bem como novos *stakeholders* e usuários identificados para o processo, além de outras informações complementares. Os requisitos não funcionais, por serem comuns aos diferentes processos do ERP, foram adicionados na seção

“Outros Requisitos” que é compartilhada por todos os processos descritos no documento de visão. Já o glossário foi incrementado conforme surgiam termos desconhecidos, e refinado à medida que se aprofundava o conhecimento sobre o negócio.

Foi com base nas necessidades dos *stakeholders* identificadas e apresentadas no documento de visão que a lista oficial de requisitos funcionais do sistema foi elaborada. Neste artefato, as funcionalidades elicitadas ao longo da atividade de levantamento e compreensão foram descritas e posteriormente relacionadas como requisitos na ferramenta RequisitePro. No documento criado para a lista de requisitos, além das categorias previamente definidas no modelo de processo proposto, uma nova seção foi adicionada para possibilitar a classificação dos requisitos de acordo com o processo a qual pertencem, com o objetivo de facilitar a identificação das funcionalidades à medida outros processos passarem pela etapa de levantamento. A figura 46 expõe os requisitos elicitados diretamente no RequisitePro.

Figura 46 - Requisitos funcionais do processo de Requisição de Compra



6.3 NEGOCIAÇÃO DE REQUISITOS

A negociação de requisitos ocorreu paralelamente à etapa de levantamento. À medida que as tarefas de elicitação progrediam, negociações constantes entre cliente e analista foram acontecendo.

Durante as atividades para coleta dos requisitos, por meio da análise de documentos, entrevistas e *workshops*, manteve-se o foco na compreensão dos riscos, complexidade e impacto de cada uma das funcionalidades demandadas pelo processo de requisição de compra. Com base no conhecimento adquirido, pôde-se ter o embasamento necessário para que a negociação dos requisitos fosse conduzida junto ao professor orientador, podendo assim argumentar pontos de vista e discordâncias surgidas. Por muitas vezes os requisitos tiveram de ser modificados em decorrência das negociações e do acordo ratificado sobre as funcionalidades pretendidas.

Da mesma forma como o professor orientador se posicionou frente aos interesses dos *stakeholders* durante o levantamento dos requisitos do sistema, o mesmo teve papel fundamental também frente à priorização dos requisitos. Cada um dos requisitos identificados para o sistema foi discutido de forma individual. Conjuntamente, aluno e orientador pontuaram os requisitos de acordo com a sua criticidade para o projeto, dando ênfase àqueles que imprescindivelmente não poderiam ser deixados de lado. Na ferramenta RequisitePro, os requisitos tiveram sua prioridade adicionada utilizando um dos valores: crítica, alta, planejada e baixa. Da mesma forma que uma categoria de *status* foi criada para permitir identificar os requisitos já aprovados, os ainda em negociação (propostos) e os tidos como obsoletos, requisitos esses antigos ou que já não agreguem funcionalidade ao sistema. A figura 47 apresenta uma *view*¹ criada no RequisitePro para a etapa de negociação de requisitos, apresentando as informações de *status* e prioridade de cada um dos requisitos durante a negociação dos mesmos.

¹ Consulta composta dinamicamente apresentando atributos previamente definidos.

Figura 47 - Requisitos do sistema e atributos da atividade de negociação de requisitos

| Requirements: | Processo | Atores Envolvi | Status | Prionidade | Author | Revision | Date |
|---|---------------------|-----------------|----------|------------|--------|----------|--------------|
| ▶ REQ1: Incluir requisição de compra | Requisição de Compr | Solicitante | Aprovado | Crítica | admin | 1.0010 | 17/06/2013 2 |
| ▣ REQ2: Aprovar requisição de compra | Requisição de Compr | Aprovador | Proposto | Crítica | admin | 1.0010 | 17/06/2013 2 |
| REQ2.1: Obrigar parecer do aprovador | Requisição de Compr | Sistema | Proposto | Planejada | admin | 1.0009 | 17/06/2013 2 |
| REQ3: Comunicar status ao solicitante | Requisição de Compr | Sistema | Proposto | Planejada | admin | 1.0007 | 17/06/2013 2 |
| ▣ REQ4: Realizar cotações | Requisição de Compr | Comprador | Proposto | Crítica | admin | 1.0007 | 17/06/2013 2 |
| REQ4.1: Consultar histórico de compras | Requisição de Compr | Comprador | Proposto | Alta | admin | 1.0008 | 17/06/2013 2 |
| REQ4.2: Buscar fornecedores | Requisição de Compr | Comprador | Proposto | Crítica | admin | 1.0008 | 17/06/2013 2 |
| REQ5: Atualizar dados da cotação | Requisição de Compr | Comprador | Proposto | Crítica | admin | 1.0006 | 17/06/2013 2 |
| REQ6: Aprovar Pedido de Compra | Requisição de Compr | Aprovador | Proposto | Crítica | admin | 1.0007 | 17/06/2013 2 |
| REQ7: Gerar pedido de compra | Requisição de Compr | Comprador | Proposto | Crítica | admin | 1.0006 | 17/06/2013 2 |
| REQ8: Encaminhar negociação | Requisição de Compr | Comprador | Proposto | a definir | admin | 1.0008 | 15/06/2013 2 |
| REQ9: Acompanhar compra | Requisição de Compr | Solicitante, Ap | Proposto | a definir | admin | 1.0008 | 15/06/2013 2 |
| * <Click here to create a requirement> | | | Proposto | | admin | 1.0000 | 20/06/2013 0 |

6.4 ESPECIFICAÇÃO E MODELAGEM DOS REQUISITOS

A etapa de especificação e modelagem dos requisitos teve início imediatamente após a finalização da lista oficial de requisitos do sistema. Serviram de entrada para esta atividade todas as informações coletadas e registradas por meio dos diferentes artefatos construídos até então.

Seguindo a premissa estabelecida no modelo de processo de engenharia de requisitos proposto para o NUSIS, e em concordância a um dos princípios básicos do processo unificado, todos os requisitos funcionais do sistema foram especificados através de casos de uso no formato completo.

Por meio das narrativas de uso criadas com cenários de utilização do sistema pôde-se projetar a interação entre atores e sistema. Tais artefatos favoreceram muito à atividade, ao passo de tornar mais perspicaz a compreensão dos requisitos para assim conseguir especificá-los. O diagrama de casos de uso do processo de requisição de compra (figura 43) esboçado durante a etapa de levantamento serviu de base para a criação das especificações de casos de uso.

Os casos de uso foram construídos seguindo a estrutura definida para o artefato no modelo de processo proposto. Além das seções previamente estipuladas para os casos de uso, viu-se a necessidade da criação de uma nova seção destinada a prototipagem de telas. A utilização de protótipos de tela é algo presente na fundamentação teórica realizada, inclusive pode ser encontrada em diferentes subfluxos da disciplina de requisitos do processo unificado no capítulo 3.2.2. Durante a elaboração do modelo de processo proposto a importância da criação de protótipos de tela passou despercebida, sendo deixada de lado. No entanto, ao aplicar o modelo através do presente estudo de caso percebeu-se a substancialidade da

utilização de protótipos na modelagem dos requisitos. Os esboços das telas criadas para os requisitos permitiram aferir a conformidade das interações entre atores e sistema, descritas nos passos do cenário de sucesso principal e suas extensões, e demonstraram-se essenciais à tarefa de especificação das funcionalidades desejadas pelo cliente. A figura 48 apresenta o conteúdo da seção “Fluxo de Eventos” da especificação do caso de uso “Efetuar Requisição de Compra”, enquanto que a figura 49 traz o protótipo de tela criado para o caso de uso. A especificação completa deste e dos demais caso de uso construídos podem ser encontradas em anexo a este trabalho.

Figura 48 - Seção "Fluxo de Eventos" do caso de uso "Efetuar Requisição de Compra"

4. Fluxo de Eventos

4.1 Acionador
Solicitante que necessita requisitar material.

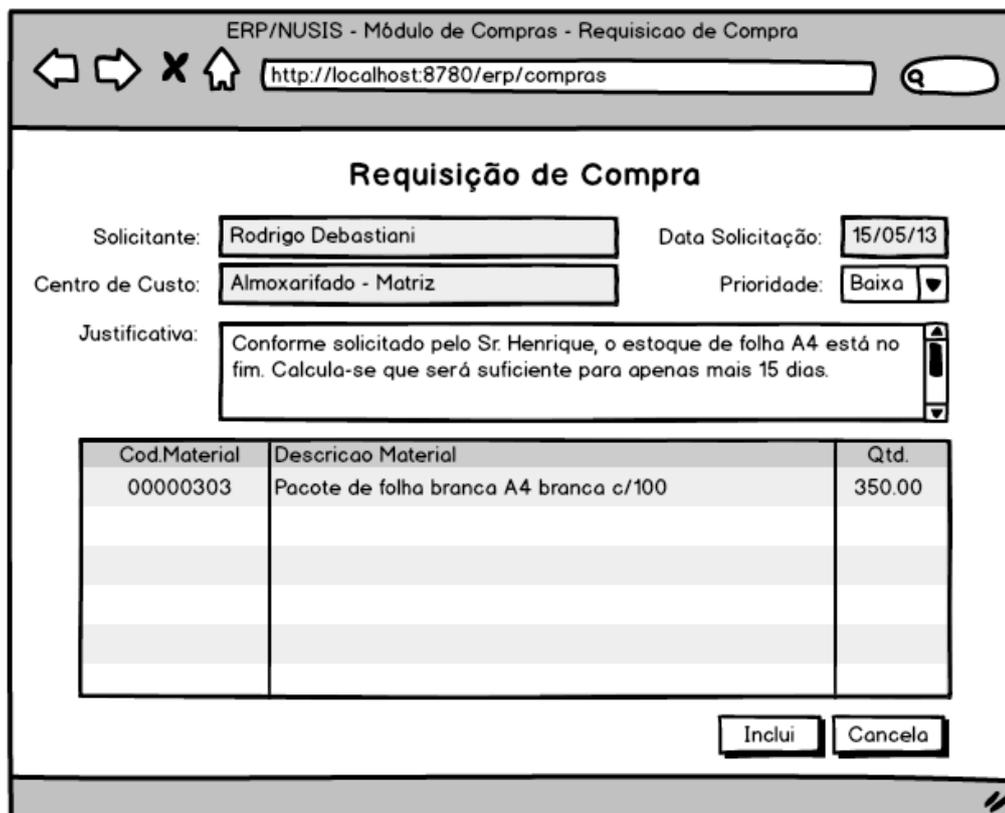
4.2 Cenário de Sucesso Principal

1. *Solicitante*: preenche formulário para requisição de compra.
2. *Sistema*: valida dados informados em tela e pede confirmação ao Solicitante para inclusão da requisição.
3. *Solicitante*: confirma a inclusão da requisição.
4. *Sistema*: verifica regras de aprovação através do código/tipo de material requisitado.
5. *Sistema*: verifica regras de aprovação através do centro de custo a qual o Solicitante pertence.
6. *Sistema*: verifica regras de aprovação através da faixa de valor do material requisitado.
7. *Sistema*: cria trilha de aprovação da requisição a partir das regras de aprovação encontradas, com o caminho de aprovação que a requisição terá de seguir, podendo este caminho possuir até três níveis de aprovação.
8. *Sistema*: aprova requisição ou a deixa pendente de aprovação.
9. *Sistema*: informa ao Solicitante o número da requisição e o seu *status*.

4.3 Extensões

- 2a. Sistema identifica que campos obrigatórios da requisição não foram preenchidos: retomar erro ao Solicitante alertando sobre a obrigatoriedade do preenchimento dos campos.
- 3a. Solicitante cancela inclusão de requisição: limpar dados do formulário e permanecer na tela de inclusão.
- 8a. De acordo com a trilha de aprovação, o Sistema pode:
 - 8a1. Aprovar automaticamente a requisição, caso não haja a necessidade de aprovação.
 - 8a2. Encaminhar a requisição para o primeiro nível de aprovação, atribuindo-a a um grupo de usuários aprovadores ou a um único usuário dentro de um grupo.

Figura 49 - Protótipo de tela do caso de uso "Efetuar Requisição de Compra"



ERP/NUSIS - Módulo de Compras - Requisicao de Compra

http://localhost:8780/erp/compras

Requisição de Compra

Solicitante: Data Solicitação:

Centro de Custo: Prioridade:

Justificativa:

| Cod.Material | Descricao Material | Qty. |
|--------------|--|--------|
| 00000303 | Pacote de folha branca A4 branca c/100 | 350.00 |
| | | |
| | | |
| | | |

Considerando o tamanho que um sistema ERP pode atingir e a quantidade de casos de uso que pode conter, na ferramenta RequisitePro optou-se por subdividir as especificações de casos de uso por processo. Desta forma, uma nova estrutura de diretórios deverá ser criada a cada novo processo que passe pela atividade de especificação e modelagem de requisitos. Após construir os casos de uso através do documento de especificação, suas seções foram trazidas para a ferramenta de gestão de requisitos e categorizadas. A figura 50 apresenta a estrutura de diretórios criada para os casos de uso do processo de requisição de compra. Na sequência, a figura 51 demonstra as seções do caso de uso “Efetuar Requisição de Compra” visualizadas a partir do RequisitePro.

Figura 50 - Estrutura de diretórios dos casos de uso do processo de Requisição de Compra

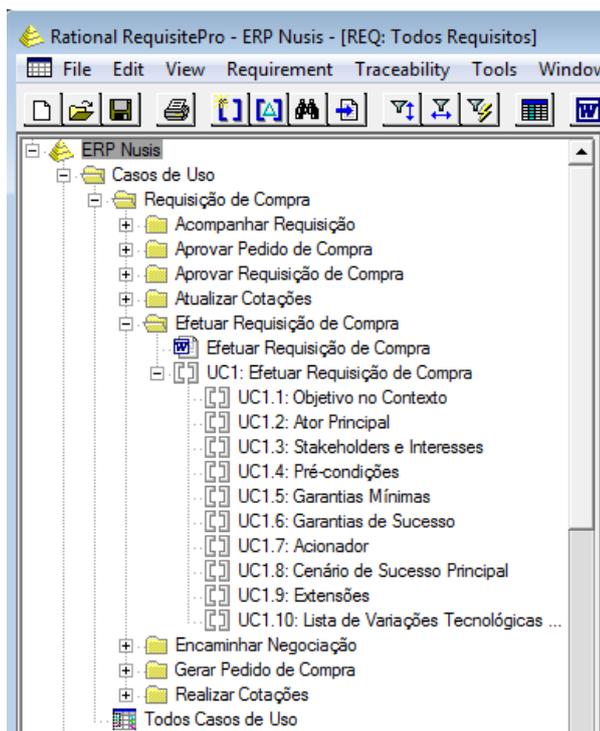


Figura 51 - Seções do caso de uso "Efetuar Requisição de Compra" visualizadas no RequisitePro

| Requirements: | Propriedades | Status | Dificuldade | Prioridade | Package | Author | Revision | Date |
|---|------------------------------|----------|-------------|------------|------------------------------|--------|----------|------------------|
| <ul style="list-style-type: none"> [-] UC1: Efetuar Requisição de Compra Efetuar Requisição de Compra | Nome do Caso de Uso | Aprovado | Alta | Planejada | Efetuar Requisição de Compra | admin | 1.0012 | 22/06/2013 20:47 |
| <ul style="list-style-type: none"> UC1.1: Objetivo no Contexto Solicitante requisita compra de material. | Objetivo no Contexto | | | | Efetuar Requisição de Compra | admin | 1.0008 | 22/06/2013 20:47 |
| <ul style="list-style-type: none"> UC1.2: Ator Principal Solicitante | Ator Principal | | | | Efetuar Requisição de Compra | admin | 1.0011 | 22/06/2013 20:47 |
| <ul style="list-style-type: none"> UC1.3: Stakeholders e Interesses Solicitante: Deseja que o material requisitado esteja disponível para utilização. | Stakeholders e Interesses | | | | Efetuar Requisição de Compra | admin | 1.0008 | 22/06/2013 20:47 |
| <ul style="list-style-type: none"> UC1.4: Pré-condições Solicitante deve estar logado no sistema e possuir permissão para efetuar requisição de compra. | Pré-Condições | | | | Efetuar Requisição de Compra | admin | 1.0009 | 22/06/2013 20:47 |
| <ul style="list-style-type: none"> UC1.5: Garantias Mínimas Toda requisição de compra submetida será analisada de acordo com as políticas de compra de materiais da empresa, podendo ser aprovada automaticamente pelo... | Garantias Mínimas | | | | Efetuar Requisição de Compra | admin | 1.0009 | 22/06/2013 20:47 |
| <ul style="list-style-type: none"> UC1.6: Garantias de Sucesso Requisição de compra incluída com sucesso, com status de aprovada ou então pendente de aprovação. | Garantias de Sucesso | | | | Efetuar Requisição de Compra | admin | 1.0010 | 22/06/2013 20:47 |
| <ul style="list-style-type: none"> UC1.7: Acionador Solicitante | Acionador | | | | Efetuar Requisição de Compra | admin | 1.0009 | 22/06/2013 20:47 |
| <ul style="list-style-type: none"> UC1.8: Cenário de Sucesso Principal 1.Solicitante: preenche formulário para requisição de compra. 2.Sistema: valida dados informados em tela e pede... | Cenário de Sucesso Principal | | | | Efetuar Requisição de Compra | admin | 1.0008 | 22/06/2013 20:47 |
| <ul style="list-style-type: none"> UC1.9: Extensões 2a. Sistema identifica que campos obrigatórios da requisição não foram preenchidos: retomar erro ao Solicitante alertando sobre a obrigatoriedade do... | Extensões | Aprovado | Normal | Planejada | Efetuar Requisição de Compra | admin | 1.0009 | 22/06/2013 20:47 |
| <ul style="list-style-type: none"> UC1.10: Lista de Variações Tecnológicas e de... Nenhuma | Variações Tecn. e de Dados | Aprovado | Normal | Planejada | Efetuar Requisição de Compra | admin | 1.0007 | 22/06/2013 20:47 |

Enquanto que os requisitos funcionais foram descritos através dos casos de uso, os requisitos não funcionais foram registrados no artefato de especificações suplementares. Tal artefato foi construído seguindo o modelo adotado pelo processo unificado. Neste momento apenas alguns requisitos não funcionais para o projeto puderam ser identificados, no entanto, estes já permitiram que a estrutura do artefato e a forma de criação dos requisitos na ferramenta de gerência fossem validadas. A figura 52 apresenta um trecho do artefato de especificações suplementares, com alguns dos requisitos não funcionais identificados. Estes mesmos requisitos foram criados também no RequisitePro, de igual forma aos requisitos funcionais do sistema.

Figura 52 - Requisitos não funcionais extraídos das especificações suplementares

| | |
|------------|--|
| 2. | Funcionalidade A serem definidos. |
| 3. | Usabilidade |
| 3.1 | Facilidade de Uso É necessário que a interface com o usuário do sistema ERP NUSIS seja intuitiva e fácil de usar, conceitos de usabilidade deverão ser implementados. |
| 4. | Confiabilidade A confiabilidade será um fator de extrema importância por se tratar de um sistema de tempo real. |
| 5. | Desempenho |
| 5.1 | Acessos Simultâneos É necessário que o ERP NUSIS seja capaz de suportar acessos simultâneos. |
| 6. | Suportabilidade A serem definidos. |
| 7. | Restrições de Design A serem definidos. |
| 8. | Requisitos de Sistema de Ajuda e de Documentação de Usuário On-line É necessário disponibilizar para o usuário, através da Web, a documentação dos processos do ERP existentes no sistema. |

No decorrer das atividades de especificação e modelagem dos requisitos o documento de visão e o glossário foram refinados à medida que o entendimento acerca do processo de requisição de compra e seus requisitos se desenvolveu. Da mesma maneira, o

modelo de domínio esboçado na etapa de levantamento pôde ser ratificado, e o diagrama de casos de uso finalizado, após passar por ajustes visando adequar-se aos artefatos construídos.

6.5 VALIDAÇÃO DOS REQUISITOS

Seguindo o modelo de processo proposto, uma vez especificados e modelados, os requisitos têm de ser validados. Nessa atividade, as especificações geradas são avaliadas quanto a sua consistência e conformidade para com a definição dos requisitos, assegurando que as necessidades dos *stakeholders* serão atendidas.

Conforme pré-estabelecido, a validação dos requisitos contou com o professor orientador no papel de revisor de requisitos. Este mesmo professor participou das atividades de levantamento e negociação dos requisitos e, portanto, apresentou-se apto a posicionar-se de maneira crítica quanto às especificações geradas.

Um a um os requisitos foram confrontados com as suas especificações. Cada caso de uso foi analisado individualmente, e neste momento, teve-se fundamentado novamente o porquê da necessidade de se construir protótipos de tela. Estes artefatos se demonstraram essenciais para a atividade de validação. Ao avaliar a interação dos atores com o sistema através do cenário de sucesso principal e suas extensões, aluno e professor puderam ratificar se os passos descritos se apresentaram condizentes com as funcionalidades pretendidas para o sistema através de seus requisitos.

Com a revisão manual considerando o escopo do processo de requisição de compra e as necessidades dos *skakeholders*, algumas das especificações tiveram de ter seu conteúdo refinado. Após realizados os devidos ajustes e passando por nova revisão, as especificações dos casos de uso puderam ser aprovadas. No RequisitePro o aceite formal do revisor de requisitos para a especificação é adicionado através do atributo “*Status*” estabelecido para o caso de uso. Outro atributo adicionado ao caso de uso de acordo com o modelo de processo aplicado neste estudo de caso foi a sua classificação de prioridade. Com o objetivo de pontuar a criticidade das funcionalidades descritas nos casos de uso perante o negócio do cliente, a classificação de prioridade é mais um instrumento de auxílio para o seguimento do projeto. Além destes dois atributos, *Status* e Prioridade, um terceiro foi criado com o propósito de fixar o grau de dificuldade encontrado ao especificar o caso de uso. O atributo “Dificuldade”, com os valores “alta”, “normal” e “baixa”, indica a complexidade do caso de uso em relação ao processo. A figura 53 apresenta os casos de uso criados para o processo de requisição de

compra com os atributos da etapa de validação devidamente preenchidos após a conclusão da atividade.

Figura 53 - Casos de uso do processo de Requisição de Compra e seus atributos de validação

| Requirements: | Status | Dificuldade | Prioridade | Package | Author | Revision | Date |
|--|----------|-------------|------------|------------------------------|--------|----------|------------------|
| ▶ ⊕ UC1: Efetuar Requisição de Compra Efetuar Requisição de Compra | Aprovado | Normal | Crítica | Efetuar Requisição de Compra | admin | 1.0014 | 24/06/2013 02:10 |
| ⊕ UC2: Aprovar Requisição de Compra Aprovar Requisição de Compra | Aprovado | Normal | Crítica | Aprovar Requisição de Compra | admin | 1.0011 | 24/06/2013 02:10 |
| ⊕ UC3: Realizar Cotações Realizar Cotações | Aprovado | Alta | Crítica | Realizar Cotações | admin | 1.0006 | 24/06/2013 02:10 |
| ⊕ UC4: Atualizar Cotações Atualizar Cotações | Aprovado | Alta | Crítica | Atualizar Cotações | admin | 1.0003 | 24/06/2013 02:10 |
| ⊕ UC5: Aprovar Pedido de Compra Aprovar Pedido de Compra | Aprovado | Normal | Crítica | Aprovar Pedido de Compra | admin | 1.0003 | 24/06/2013 02:10 |
| ⊕ UC6: Gerar Pedido de Compra Gerar Pedido de Compra | Aprovado | Alta | Crítica | Gerar Pedido de Compra | admin | 1.0003 | 24/06/2013 02:10 |
| ⊕ UC7: Encaminhar Negociação Encaminhar Negociação | Aprovado | Normal | Planejada | Encaminhar Negociação | admin | 1.0004 | 24/06/2013 02:12 |
| ⊕ UC8: Acompanhar Requisição Acompanhar Requisição | Aprovado | Normal | Planejada | Acompanhar Requisição | admin | 1.0004 | 24/06/2013 02:12 |
| * <Click here to create a requirement> | Proposto | Normal | Planejada | None | admin | 1.0000 | 24/06/2013 02:01 |

6.6 GERÊNCIA DE REQUISITOS

Ao propor a atividade de gerência de requisitos para o NUSIS, ficou implícita a necessidade da utilização de uma ferramenta *CASE*. O capítulo 5 deste trabalho trouxe a análise dessas ferramentas, considerando as particularidades e exigências identificadas para o projeto em função do ambiente a qual é destinado. Ainda de acordo com análise realizada, e através do estudo comparativo entre as ferramentas disponíveis, chegou-se a conclusão sobre a ferramenta *CASE* mais indicada para o NUSIS: a RequisitePro da IBM.

A referida ferramenta foi utilizada desde a primeira etapa da aplicação deste estudo de caso, e acompanhou todo o ciclo de atividades da engenharia de requisitos realizado. Nela os artefatos construídos puderam ser organizados já pensando na evolução do projeto, sabendo das dimensões que um sistema ERP pode atingir. Por meio da ferramenta de gestão de requisitos, a tarefa de administração dos requisitos que poderia ser encarada como um obstáculo para o projeto se viu facilitada.

Através da ferramenta foram criados diferentes níveis de rastreabilidade. Primeiramente, os requisitos criados a partir da lista de requisitos oficial do sistema foram

Os casos de uso são especificações das funcionalidades pretendidas para o sistema e identificadas através dos requisitos. Um requisito pode dar origem a múltiplos casos de uso, assim como um caso de uso pode representar mais de um requisito. Este talvez seja o relacionamento mais importante da engenharia de requisitos. A figura 56 traz a matriz de rastreabilidade gerada entre requisitos e casos de uso a partir da aplicação do estudo de caso.

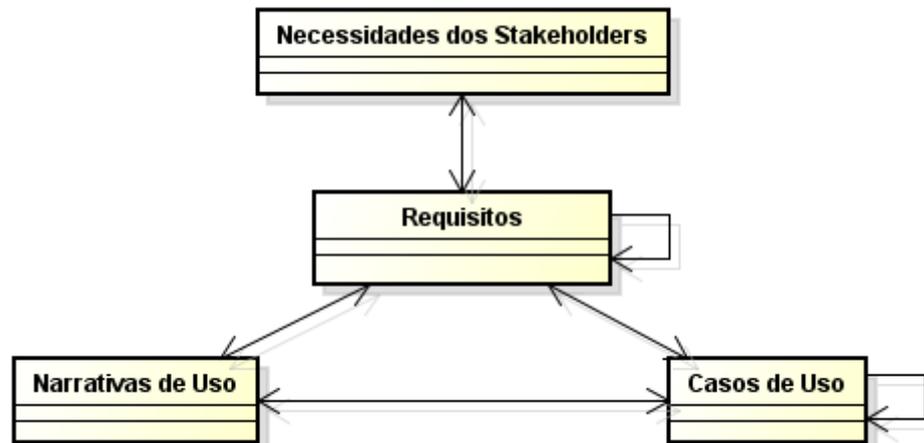
Figura 56 - Matriz de rastreabilidade entre Requisitos do Sistema e Casos de Uso

| Relationships: - direct only | REQ1: Incluir requisição de... Incluir requisição de compra | REQ2: Aprovar requisição... Aprovar requisição de compra | REQ2.1: Obrigar parecer do... Obrigar parecer do aprovador | REQ3: Comunicar status ao... Comunicar status ao solicitante | REQ4: Realizar cotações Realizar cotações | REQ4.1: Consultar histórico... Consultar histórico de compras | REQ4.2: Buscar fornecedores Buscar fornecedores | REQ5: Atualizar dados da... Atualizar dados da cotação | REQ6: Aprovar Pedido de... Aprovar Pedido de Compra | REQ7: Gerar pedido de... Gerar pedido de compra | REQ8: Encaminhar... Encaminhar negociação | REQ9: Acompanhar compra Acompanhar compra |
|---|--|---|---|---|--|--|--|---|--|--|--|--|
| ⊕ UC1: Efetuar Requisição de Compra Efetuar Requisição de Compra | | | | | | | | | | | | |
| ⊕ UC2: Aprovar Requisição de Compra Aprovar Requisição de Compra | | | | | | | | | | | | |
| ⊕ UC3: Realizar Cotações Realizar Cotações | | | | | | | | | | | | |
| ⊕ UC4: Atualizar Cotações Atualizar Cotações | | | | | | | | | | | | |
| ⊕ UC5: Aprovar Pedido de Compra Aprovar Pedido de Compra | | | | | | | | | | | | |
| ⊕ UC6: Gerar Pedido de Compra Gerar Pedido de Compra | | | | | | | | | | | | |
| ⊕ UC7: Encaminhar Negociação Encaminhar Negociação | | | | | | | | | | | | |
| ⊕ UC8: Acompanhar Requisição Acompanhar Requisição | | | | | | | | | | | | |

Considerando a ideia da utilização das narrativas de uso como forma de antever os casos de uso que terão de ser especificados, faz-se necessário também rastrear um caso de uso a partir de sua narrativa e vice-versa. Desta forma, uma matriz de rastreabilidade teve de ser criada para manter a associação entre esses dois tipos de artefatos. Essa relação é apresentada através da matriz de rastreabilidade na figura 57.

Ao término da aplicação do estudo de caso, diferentes formas de rastreabilidade foram mantidas. As rastreabilidades possíveis são sintetizadas através do diagrama de classes demonstrado na figura 59.

Figura 59 - Formas de rastreabilidade possíveis na gerência de requisitos



6.7 CONSIDERAÇÕES FINAIS

A instanciação do modelo de processo proposto no capítulo 4 deste trabalho, em conjunto à ferramenta *CASE* para gerência de requisitos RequitePro resultante da avaliação apresentada no capítulo 5, objetivou aferir a aplicabilidade do modelo proposto em relação ao processo de engenharia de requisitos do NUSIS, através de um estudo de caso para o qual foram selecionadas funcionalidades do processo de requisição de compra existente em um ERP.

Por meio da aplicação do modelo de processo foi possível apurar que o mesmo pode sim ser adequado à engenharia de requisitos do NUSIS. O estudo de caso possibilitou colocar em prática os estudos realizados ao longo deste trabalho e vivenciar as situações reais de utilização do processo.

No decorrer do estudo de caso algumas alterações e melhorias no modelo proposto foram necessárias, visando aprimorar e moldar o modelo à realidade enfrentada. A técnica de verificação de documentos teve início ainda na etapa de concepção e análise do problema. O documento de visão foi subdividido em seções específicas por processo, já prevendo o aumento de informações com o andamento do projeto. O diagrama de casos de uso, que

somente seria concebido ao término da etapa de especificação e modelagem dos requisitos, teve seu início antecipado para a atividade de levantamento e compreensão dos requisitos. A versão final de tal diagrama continua a ser obtida após a conclusão das especificações, no entanto, o diagrama começa a ser esboçado mais cedo, paralelamente as tarefas de coleta de requisitos.

Para a negociação de requisitos, nenhum atributo que permitisse classificar os requisitos elicitados havia sido instituído. No decorrer da aplicação do processo proposto sentiu-se a necessidade de identificar os requisitos já aprovados, bem como a sua criticidade perante o projeto. Para isso os atributos “prioridade” e “status” foram criados junto aos requisitos.

Outro refinamento realizado junto ao modelo proposto estabeleceu a construção de protótipos de tela para as especificações dos casos de uso. Quando da elaboração das atividades e dos artefatos para a engenharia de requisitos do NUSIS, não deu-se conta da importância em possuir esboços para as telas pretendidas para o sistema, que dentre outras facilidades auxiliam a aferir a conformidade das interações descritas no caso de uso.

Por meio da utilização da ferramenta de gestão de requisitos a tarefa de manter a rastreabilidade entre requisitos e artefatos se mostrou algo simples de ser realizado. A partir disso, viu-se a conveniência da criação de uma nova forma de rastreabilidade além das previamente estabelecidas, permitindo relacionar as necessidades dos *stakeholders* descritas no documento de visão com os requisitos elicitados para o projeto.

Diante disso, pode-se dizer que sem a instanciação do modelo de processo através do estudo de caso, a necessidade de tais adaptações e melhorias não seria identificada, o que vem a reconhecer a forma utilizada para avaliação, que embora sem contar com indicadores mensuráveis pôde através da sua aproximação com a realidade e com as tarefas normalmente executadas no cotidiano dos envolvidos, contribuir para o refinamento do processo.

Ao propor o modelo de processo para a engenharia de requisitos do NUSIS, no capítulo 4.2 foram elencados os atributos e características de qualidade que o mesmo deveria atender. Ao fim da aplicação deste estudo de caso podemos concluir que tais atributos e características são assegurados pelo processo.

A tabela 13 resgata os atributos e características tidos como desejáveis ao NUSIS, de acordo com o estudo das normas, padrões e modelos de qualidade realizado ao longo do capítulo 2 deste trabalho. Nesta tabela são descritos como cada um dos requisitos é atendido pelo modelo de processo. Tal avaliação somente foi possível após a aplicação do estudo de caso.

Tabela 13 – Considerações sobre os atributos e características de qualidade

| Atributos / Características | Considerações após aplicação do modelo através do estudo de caso |
|-----------------------------|---|
| Comunicação | Temos a comunicação evidenciada através dos diferentes artefatos produzidos ao longo das atividades. Estes mesmos artefatos serviram para que as tarefas da engenharia de requisitos fossem conduzidas em contato com o cliente, e também asseguram a continuidade do ciclo de desenvolvimento por meio das especificações produzidas, que serão utilizadas nas próximas etapas do desenvolvimento do <i>software</i> . |
| Compromisso / Aprovação | Através do atributo de “Status” dos requisitos funcionais e também dos casos de uso, pode-se obter o compromisso do cliente com os requisitos coletados e com as especificações produzidas para os mesmos. |
| Capacidade de Gerenciamento | A capacidade de gerenciamento foi garantida através da ferramenta <i>CASE</i> utilizada ao longo das atividades, que permitiu administrar adequadamente requisitos e artefatos. |
| Testabilidade | A testabilidade dos requisitos é viabilizada por meio das especificações produzidas. A necessidade da criação de protótipos de tela para os casos de uso colaborou para o atendimento desta característica de qualidade. |
| Não ambiguidade | Uma vez adicionados à ferramenta <i>CASE</i> , o controle de ambiguidade pode ser facilmente realizado com o seguimento do projeto. |
| Completeness | Por meio das matrizes de rastreabilidade, é possível rastrear partindo dos requisitos do sistema, aqueles que porventura não estejam relacionado a nenhuma especificação. Tal rastreabilidade possibilita assegurar que todo requisito levantado junto ao cliente tenha sua funcionalidade especificada por meio de uma especificação. |
| Consistência | Amparado pelas formas de rastreabilidade possíveis, que permitem se chegar nas especificações dos casos de uso a partir das necessidades dos <i>stakeholders</i> , é possível confrontar se as necessidades identificadas são consistentes com a sua forma de especificação. |
| Verificabilidade | Este atributo, que muito se assemelha as características de testabilidade e consistência, está relacionado à capacidade de se verificar em termos de tempo e custo, se o sistema cumpre o requisito. Tal verificação é possível através das especificações produzidas, com o auxílio da ferramenta de gerência dos requisitos e das formas de rastreabilidade existentes. |
| Modificabilidade | Este atributo pode ser evidenciado através da ferramenta <i>CASE</i> , que permite avaliar o impacto da mudança de um requisito perante aos demais requisitos atrelados a ele e seus respectivos artefatos. |
| Rastreabilidade | As diferentes matrizes de rastreabilidade construídas e demonstradas na seção 6.6 garantem a rastreabilidade bi-direcional de requisitos e artefatos. |

7 CONSIDERAÇÕES FINAIS

O presente trabalho buscou propor um processo de engenharia de requisitos para o ambiente do NUSIS capaz de assegurar a qualidade de suas atividades.

Para propor o processo, primeiramente foi necessário compreender o que se espera de qualidade dentro da engenharia de requisitos. O aprofundamento do estudo das normas, padrões e modelos de processo que possuem relação direta com a engenharia de requisitos possibilitou relacionar, através dos atributos e características desejáveis, o que se entende como necessário para as atividades perante o ambiente do NUSIS. A elaboração do modelo proposto considerou além das características apontadas na problemática da pesquisa, os conceitos de engenharia de requisitos e *workflow* de requisitos do processo unificado.

As particularidades apresentadas pelo NUSIS, em comparação ao cotidiano de outras empresas desenvolvedoras de *software*, foram sem dúvida os maiores desafios encontrados. O fato de não haver um cliente, a alta rotatividade da equipe de trabalho, a inexperiência e a falta de conhecimento acerca do negócio, fizeram com que o processo proposto fosse moldado na tentativa de amenizar os problemas que se pode ter relacionado a cada um desses aspectos. Atentando a esses fatores, no modelo proposto pode-se destacar a técnica de verificação de documentos instituída desde a etapa de concepção e análise do problema, a construção das narrativas de uso com cenários de utilização do sistema, e a análise de sistemas correlatos como base para fundamentação do negócio.

Através da instanciação do modelo proposto para um estudo de caso, buscou-se aferir a sua coerência e conformidade para com as atividades do processo de engenharia de requisitos do ambiente experimental do NUSIS. No decorrer da aplicação, algumas alterações e melhorias foram realizadas visando o aprimoramento do modelo de acordo com a realidade enfrentada. Tendo em vista que o foco deste trabalho é voltado para a construção de um ERP, caracterizado pela sua modularidade, grande porte e alta complexidade, e considerando também o ambiente a qual é destinado, teve-se implícita a necessidade da utilização de uma ferramenta *CASE* para a atividade de gerenciamento dos requisitos. Por sua vez, para que uma ferramenta pudesse ser designada, um processo de avaliação teve de ser conduzido. Neste, por meio de análises comparativas, a ferramenta RequisitePro da IBM foi selecionada para ser utilizada conjuntamente ao modelo de processo proposto, apoiando as atividades na geração dos diferentes artefatos estabelecidos e na administração dos requisitos do projeto.

Por fim, os resultados obtidos através da aplicação do modelo de processo no estudo de caso permitiram concluir que o mesmo é adequado à engenharia de requisitos do NUSIS. A avaliação individual de cada um dos atributos e características de qualidade vista na tabela 13 permitiu avaliar sua conformidade em relação ao processo, e atestar a garantia da qualidade nas atividades demonstradas. O modelo de processo proposto neste trabalho não é encerrado aqui. Ele continua a evoluir conjuntamente ao *software*.

7.1 TRABALHOS FUTUROS

A gerência de requisitos estabelecida para o processo de engenharia de requisitos do NUSIS é suportada com o apoio de uma ferramenta *CASE*, a qual foi determinada através de um processo de avaliação das soluções existentes no mercado. Nele, buscou-se a ferramenta que melhor se adaptasse ao modelo de processo proposto. No entanto, as particularidades do ambiente de desenvolvimento de *software* do NUSIS são muitas. Isso faz emergir a ideia do desenvolvimento de uma ferramenta própria para a gestão de requisitos do NUSIS, construída e moldada com base nas atividades e artefatos propostos neste trabalho.

O modelo de processo aqui proposto pode ainda ser avaliado através de sua experimentação por outro analista, uma vez que durante sua instanciação para um estudo de caso o mesmo foi aplicado pelo próprio autor.

8 REFERÊNCIAS

ALVES, Carina, DELMAS, Cesar A., PENAFORTE, Daniel A., OLIVEIRA, Hector P. de L., PIMENTEL, João H. C. **Um Estudo Empírico sobre Práticas de Engenharia de Requisitos junto a Empresas de Pacotes de Software** . Relatório técnico. Pernambuco, 2007.

ARNOLD, J.R. Tony. **Administração de materiais: uma introdução**. São Paulo: Atlas, 1999.

BAILY, Peter. **Compras: princípios e administração**. São Paulo: Atlas, 2000.

BOOCH, Grady; RUMBAUGH, James; JACOBSON, Ivar. **UML Guia do Usuário**. Rio de Janeiro: Campus, 2000.

BRAUDE, Eric. **Projeto de Software. Da programação à arquitetura: uma abordagem baseada em Java**. Porto Alegre: Bookman, 2005.

BROOKS, Frederick P. “**No Silver Bullet: Essence and Accidents of Software Engineering**”. Disponível em:
<http://www.cs.unibo.it/~cianca/wwwpages/ids/lecture/Brooks.pdf>. Acesso em: 18 ago. 2012.

CARVALHO, Ariadne M. B. R.; CHIOSSI, Thelma C. dos S. **Introdução à engenharia de software**. Campinas: Editora da Unicamp, 2005.

COCKBURN, Alistair. **Escrevendo casos de uso eficazes**. Porto Alegre: Bookman, 2005.

FRANCISCHINI, G. Paulino. **Administração de materiais e do patrimônio**. São Paulo: Cengage Learning, 2010.

GONÇALVES, Paulo Sérgio. **Administração de materiais**. Rio de Janeiro: Elsevier, 2010.

HIRAMA, Kechi. **Engenharia de software [recurso eletrônico]: qualidade e produtividade com tecnologia**. Rio de Janeiro: Elsevier, 2012.

IEEE. **IEEE Standard 830: recommended practice for software requirements specifications**. New York, 1998.

INTERNATIONAL INSTITUTE OF BUSINESS ANALYSIS. **Um Guia Para o Corpo de Conhecimento de Análise de Negócios (Guia BABOK)**, versão 2.0, Canadá, 2011.

INTERNATIONAL ORGANIZATION FOR STANDARDIZATION AND INTERNATIONAL ELECTROTECHINICAL COMISSION. **ISO/IEC 9126-1: 2000**. Engenharia de Software – Qualidade de produto de Software – Parte 1: Modelo de Qualidade, Genebra, 2003.

INTERNATIONAL ORGANIZATION FOR STANDARDIZATION AND INTERNATIONAL ELECTROTECHINICAL COMISSION. **ISO/IEC 12207:2008** - Tecnologia de Informação - Processos de ciclo de vida de Software. Genebra, 2008.

INTERNATIONAL ORGANIZATION FOR STANDARDIZATION AND INTERNATIONAL ELECTROTECHINICAL COMISSION. **ISO/IEC 12598:2000** – Information Technology - Software Product Evaluation. Genebra, 2000.

INTERNATIONAL ORGANIZATION FOR STANDARDIZATION AND INTERNATIONAL ELECTROTECHINICAL COMISSION. **ISO/IEC 15504**: Information Technology - Process Assessment – Part 2 - Performing an Assessment, Genebra, 2003.

LUNA, Alexandre J. H. de O. **Abordagem da Engenharia de Requisitos em Projetos de Desenvolvimento de Software para Telesaúde/Telemedicina**. 2008. Monografia (Pós-graduação) – Universidade Federal de Pernambuco. Recife. 2008.

LAHOZ, Carlos H. N.; SANTANNA, Nilson. **Os Padrões ISO/IEC 12207 e 15504 e a Modelagem de Processos da Qualidade de Software**. In: III Workshop dos Cursos de Computação Aplicada do INPE, 2003, São José dos Campos. Anais do III WORKCAP, 2003. p. 43-48.

LARMAN, Craig. **Utilizando UML e Padrões: Uma introdução à análise e ao projeto orientados a objetos e ao desenvolvimento iterativo, 3ª edição**. Porto Alegre: Bookman, 2007.

LOPES, Leandro T.. **Um Modelo de Processo de Engenharia de Requisitos para Ambientes de Desenvolvimento Distribuído de Software**. Dissertação (Mestrado) – Pontifícia Universidade Católica do Rio Grande do Sul. Porto Alegre. 2004.

LUNA, Alexandre J. H. de O. **Abordagem da Engenharia de Requisitos em Projetos de Desenvolvimento de Software para Telesaúde/Telemedicina**. 2008. Monografia (Pós-graduação) – Universidade Federal de Pernambuco. Recife. 2008.

OBERG, Roger; PROBASCO, Leslee; ERICSSON, Maria. **Applying requirements management with use cases**. Califórnia, 1998. Disponível em: http://www.uml.org.cn/requirementproject/pdf/wp-applying_requirements_management_with_use_cases.pdf. Acesso em: 15 set. 2012.

PFLEEGER, Shari L.. **Engenharia de Software: teoria e prática 2ª edição**. São Paulo: Prentice Hall, 2004.

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

REZENDE, Denis Alcides. **Engenharia de Software e Sistemas de Informação 3ª edição**. Rio de Janeiro: Brasport, 2005.

RATIONAL – IBM Rational Software. **Rational Unified Process, versão 2002.05.00**, 2002. Disponível em: <<http://www.wthree.com/rup/>>. Acessado em: 27 de outubro, 2012.

ROCCO, Giovanni Ely. **Um modelo de Estruturação de Requisitos para o Método Fusion**. Dissertação (Mestrado) – Universidade Federal do Rio Grande do Sul. Porto Alegre. 2001.

ROCHA, Ana. R. C.; MALDONADO, José. C.; WEBER, Kival. C. **Qualidade de Software : teoria e prática**. São Paulo: Prentice-Hall, 2001.

RODRIGUES, Vasco M.. **Gestão de Requisitos: aplicação colaborativa para a gestão de requisitos**. Dissertação (Mestrado) – Universidade Técnica de Lisboa. Lisboa. 2008.

ROUILLER, Ana Cristina. **Gerência de Projetos de Software/Ana Cristina Rouiller**. Lavras: UFLA/FAEPE, 2008.

SAYÃO, Miriam; STAA Arnd V.; LEITE, Julio C. S. de P. **Qualidade em Requisitos**. Rio de Janeiro: Departamento de Informática, PUC-Rio, 2003 (Monografia).

SCOTT, Kendall. **O Processo Unificado Simplificado**. Porto Alegre: Bookman, 2003.

SEI, Software Engineering Institute. **CMMI for Development: Version 1.2: CMMIDEV**. USA: SEI, 2010.

SILVA, Alberto M. R.; VIDEIRA, Carlos A. E. **UML - Metodologias e Ferramentas CASE**. Lisboa: Centro Atlântico, 2001.

SOFTEX – Associação para Promoção da Excelência do Software Brasileiro. **MPS.BR – Guia Geral:2011**, agosto de 2011. Disponível em: <<http://www.softex.br>>. Acessado em: 13 de outubro, 2012.

SOMMERVILLE, Ian. **Engenharia de Software, 8ª edição**. São Paulo: Pearson Addison-Wesley, 2007.

SOMMERVILLE, Ian. **Engenharia de Software, 9ª edição**. São Paulo: Pearson Prentice Hall, 2011.

TAURION, Cezar. **Software embarcado: oportunidades e potencial de mercado**. Rio de Janeiro: Brasport, 2005.

TONSIG, Sérgio Luiz. **Engenharia de Software: Análise e Projeto de Sistemas**. São Paulo: Futura, 2003.

WAZLAWICK, Raul Sidnei. **Análise e Projeto de Sistemas de Informação Orientados a Objeto**. Rio de Janeiro: Elsevier, 2004.

Universidade de Caxias do Sul – NUSIS

ERP NUSIS Documento de Visão

Versão 1.1

| | | |
|---|---------------------------------------|----------------|
|  | Universidade de Caxias do Sul – NUSIS | Versão: 1.1 |
| | ERP NUSIS: Documento de Visão | Data: 24/06/13 |

Histórico de Revisão

| Data | Versão | Descrição | Autor |
|-------------|---------------|--|--------------------|
| 01/05/13 | 1.0 | Criado documento | Rodrigo Debastiani |
| 24/06/13 | 1.1 | Ajudadas Necessidades dos Stakeholders | Rodrigo Debastiani |
| | | | |
| | | | |

| | | |
|---|---------------------------------------|----------------|
|  | Universidade de Caxias do Sul – NUSIS | Versão: 1.1 |
| | ERP NUSIS: Documento de Visão | Data: 24/06/13 |

Índice Analítico

| | | |
|-------|--|---|
| 1. | Objetivos Gerais | 4 |
| 2. | Estruturação do Documento | 4 |
| 3. | Processos | 5 |
| 3.1 | Requisição de Compra | 5 |
| 3.1.1 | Escopo | 5 |
| 3.1.2 | Referências | 5 |
| 3.1.3 | <i>Stakeholders</i> | 6 |
| 3.1.4 | Necessidades dos <i>Stakeholders</i> | 6 |
| 3.1.4 | Recursos do Sistema | 8 |
| 3.1.5 | Informações Complementares | 8 |
| 3.1.6 | Visão Geral UML – Diagrama de Casos de Uso | 9 |
| 4. | Outros Requisitos | 9 |

| | | |
|---|---------------------------------------|----------------|
|  | Universidade de Caxias do Sul – NUSIS | Versão: 1.1 |
| | ERP NUSIS: Documento de Visão | Data: 24/06/13 |

Visão

1. Objetivos Gerais

A finalidade deste documento é coletar, analisar e definir as necessidades e características do sistema ERP em desenvolvimento pelo Núcleo de Sistemas de Informação da Universidade de Caxias do Sul. Ele se concentra nos recursos necessários aos *stakeholders* e usuários-alvo, e no por que dessas necessidades existirem. Os detalhes de como os processos do ERP atenderão a essas necessidades estão descritos nos casos de uso e nas especificações suplementares.

2. Estruturação do Documento

Para melhor estruturar este documento, seu conteúdo foi dividido de acordo com os processos levantados para o ERP. Este documento será incrementado de acordo com o andamento do projeto, agregando as informações pertinentes a cada um dos processos trabalhados. Apenas os requisitos não funcionais, por serem comuns aos diferentes processos do ERP, são apresentados em uma seção a parte, através dos “Outros Requisitos”.

| | | |
|--|---------------------------------------|----------------|
| | Universidade de Caxias do Sul – NUSIS | Versão: 1.1 |
| | ERP NUSIS: Documento de Visão | Data: 24/06/13 |

3. Processos

3.1 Requisição de Compra

3.1.1 Escopo

As informações trazidas nas próximas subseções deste documento limitam-se ao processo de compra de materiais de consumo e expediente desencadeado na área de Compras de uma empresa.

É contemplado por este processo a requisição inicial de uma compra, que pode ser oriunda de diversos setores e pessoas, a aprovação dessa requisição por um responsável, a obtenção de cotações de preços junto aos fornecedores cadastrados, a análise e escolha da melhor cotação e a geração do pedido de compra e seu encaminhamento para o fornecedor, além do acompanhamento da compra até que a mesma seja recebida pela empresa.

A compra de matéria-prima para o setor de produção e de material para os setores de vendas e distribuição não está englobada no contexto do projeto e portanto não é abrangida por este documento. Esta é uma atribuição do módulo de Planejamento de Necessidade de Material (MRP), responsável por suprir esse tipo de demanda e assegurar a disponibilidade desses recursos através do monitoramento de estoques.

As necessidades e características relacionadas ao cadastro, manutenção e gerenciamento das informações inerentes aos fornecedores também é algo que está fora do escopo do projeto e consequentemente não é compreendido neste documento.

3.1.2 Referências

| Referência | |
|--------------------|--|
| Professor do NUSIS | Iraci Cristina da Silveira de Carli |
| Bibliografia/Livro | ARNOLD, J.R. Tony. Administração de materiais: uma introdução . São Paulo: Atlas, 1999. |
| Bibliografia/Livro | BAILY, Peter. Compras: princípios e administração . São Paulo: Atlas, 2000. |
| Bibliografia/Livro | FRANCISCHINI, G. Paulino. Administração de materiais e do patrimônio . São Paulo: Cengage Learning, 2010. |
| Bibliografia/Livro | GONÇALVES, Paulo Sérgio. Administração de materiais . Rio de Janeiro: Elsevier, 2010. |

| | | |
|--|---------------------------------------|----------------|
| | Universidade de Caxias do Sul – NUSIS | Versão: 1.1 |
| | ERP NUSIS: Documento de Visão | Data: 24/06/13 |

3.1.3 Stakeholders

| Stakeholders | Responsabilidades |
|--------------|---|
| Solicitante | Responsável pela inclusão da requisição de compra. Quem de fato faz uso dos materiais de consumo e expediente. |
| Aprovador | Responsável pelos interesses financeiros da empresa através do controle de gastos. Responsável por analisar e aprovar as requisições de compra antes da etapa de cotação. Também é o responsável pela escolha da cotação com melhor custo-benefício à empresa para geração do pedido de compra junto ao fornecedor. |
| Comprador | Responsável por selecionar os fornecedores para realização das cotações, e responsável por atualizar os registros de acordo com os retornos obtidos junto ao fornecedor. Também é o responsável pela geração do pedido de compra e encaminhamento da negociação junto ao fornecedor. |

3.1.4 Necessidades dos Stakeholders

| Necessidade | Descrição | Proposta de Solução | Prioridade |
|---|---|--|------------|
| Incluir requisição de compra | Possibilitar a inclusão de requisições de materiais. | A função para inclusão da requisição de compra permitirá que qualquer funcionário com permissão requisite material. | Crítica |
| Manter o solicitante informado | Manter o solicitante informado sobre o andamento de sua requisição de compra. | Comunicar o solicitante do andamento de sua requisição através de e-mails disparados automaticamente pelo sistema. | Planejada |
| Aprovar requisição de compra | Possibilitar que as requisições passem por etapa de aprovação antes de chegar ao setor de compras. | Para toda requisição será verificada a necessidade de aprovação, fazendo com que uma requisição só caia para o setor de Compras após ser aprovada, ou então na inexistência de regras de aprovação. | Crítica |
| Aprovação por níveis | Permitir que a aprovação das requisições ocorra em diferentes níveis, uma vez que uma requisição pode ter de ser aprovada por mais de um responsável. | Sistema verificará regras de negócio para aprovação da requisição, criando uma espécie de trilha de aprovação pela qual a requisição obrigatoriamente tem que passar antes de chegar ao setor de Compras. Cada uma das regras estabelecidas será observada pelo sistema, seguindo a ordem estabelecida, onde primeiro serão verificadas as regras por código/tipo de material, em seguida a regras por centro de custo, e por fim as regras por faixa de valor. | Crítica |
| Aprovação pendente com usuário ou grupo | Permitir que as requisições fiquem pendentes de aprovação com um grupo de usuários ou então com | Será possível parametrizar na regra de aprovação quem será o responsável, usuário ou grupo. | Planejada |

| | | |
|--|---------------------------------------|----------------|
| | Universidade de Caxias do Sul – NUSIS | Versão: 1.1 |
| | ERP NUSIS: Documento de Visão | Data: 24/06/13 |

| | | | |
|--|--|---|-----------|
| | um único usuário dentro de um grupo. | | |
| Permitir alterações na requisição de compra | Permitir que o responsável de aprovação altere e/ou acrescente informações à requisição de compra. | Na etapa de aprovação, o usuário responsável poderá alterar informações da requisição. | Alta |
| Tornar obrigatória a justificativa na negação de uma requisição | Tornar obrigatória a justificativa com um parecer por parte do responsável de aprovação em caso da requisição de compra ser negada. | Ao negar uma requisição, obrigatoriamente o usuário aprovador terá de incluir um parecer/justificativa para concluir a operação. | Planejada |
| Permitir a realização de cotações | Permitir a realização de cotações para a requisição de compra | A realização de cotações ocorrerá em dois momentos distintos no sistema. Primeiramente serão selecionados os fornecedores para os quais serão solicitadas cotações. Após receber o retorno para as mesmas os dados terão de ser atualizados no sistema. | Crítica |
| Possibilitar a busca pelos fornecedores cadastrados | Possibilitar ao Comprador na etapa de realização das cotações a busca pelos fornecedores cadastrados para o material em questão. | Haverá integração com o Módulo de Gerenciamento de Fornecedores que permitirá a busca dos fornecedores cadastrados. | Crítica |
| Apresentar os fornecedores cadastrados de acordo com a forma de ranqueamento | Apresentar ao Comprador os fornecedores cadastrados para o material requisitado de acordo com o nível de excelência destes através das políticas de ranqueamento de fornecedores da empresa. | Através da integração com o Módulo de Gerenciamento de Fornecedores serão retornadas as informações dos fornecedores baseadas nos critérios de ranqueamento estabelecidos pela empresa. | Alta |
| Apresentar o fornecedor da última compra | Apresentar ao Comprador o fornecedor da última compra realizada para o material requisitado. | Na etapa de cotações, será disponibilizada opção que permitirá ao Comprador consultar o histórico de compras do material, com os últimos fornecedores do mesmo. | Planejada |
| Diferenciar os materiais com contrato de fornecimento | Diferenciar para o Comprador os materiais que possuem contrato de fornecimento daqueles que não possuem. | Será destacado ao Comprador na etapa de cotações a existência de contrato de fornecimento para o material. | Planejada |
| Possibilitar o estabelecer de um número mínimo de cotações | Possibilitar o estabelecimento de um número mínimo de cotações necessárias de acordo com o material/tipo requisitado. | Será possível parametrizar um número mínimo de cotações para o material/tipo, que por sua vez ser considerado pelo sistema na etapa de cotações. | Alta |
| Diferenciar as cotações no aguardo do | Diferenciar as cotações para as quais já foi recebido retorno do fornecedor | Haverão <i>status</i> que permitirão diferenciar as cotações ainda sem retorno daquelas que já tenham sido atualizadas com as informações | Alta |

| | | |
|--|---------------------------------------|----------------|
| | Universidade de Caxias do Sul – NUSIS | Versão: 1.1 |
| | ERP NUSIS: Documento de Visão | Data: 24/06/13 |

| | | | |
|--|---|--|-----------|
| fornecedor | daquelas que ainda se estão no aguardo de um retorno. | retornadas pelo fornecedor. | |
| Aprovar pedido de compra | Antes de gerar um pedido de compra, as cotações deverão ser analisadas por um responsável, normalmente representado pelo gerente financeiro ou um comprador mais experiente. | Após atualizar as cotações com as informações retornadas pelos fornecedores, a requisição de compra passará por nova etapa de aprovação, agora para geração do pedido de compra. | Crítica |
| Integrar pedido de compra com o contas a pagar | Todo pedido de compra deverá ser comunicado ao setor de contas a pagar, pois implica em uma despesa para a empresa. | Durante o processo de geração do pedido de compra, o mesmo será integrado com o Módulo de Contas a Pagar, gerando uma previsão de despesa no sistema. | Alta |
| Encaminhar negociação com o fornecedor automaticamente | Ao gerar o pedido de compra deve existir a possibilidade de encaminhar a negociação automaticamente ao fornecedor, através de e-mail ou então pela integração direta com o sistema do fornecedor. | Após concluir a geração do pedido, o Comprador será questionado sobre o interesse em encaminhar a negociação de compra. Nesse caso o sistema verificará a parametrização existente para o fornecedor, podendo esta ser por e-mail ou então <i>WebService</i> , e disparará automaticamente a negociação do pedido baseado nos dados da cotação. | Planejada |
| Possibilitar o acompanhamento da requisição | Possibilitar o acompanhamento da requisição a qualquer momento, por parte do Solicitante e também do Comprador | Será disponibilizada função de acompanhamento da requisição, diferenciando os usuários. O sistema sempre retornará o histórico da requisição, com todas as etapas passadas e seus respectivos usuários responsáveis. Quando a consulta estiver sendo realizada pelo Comprador e a negociação de compra tiver sido encaminhada para o fornecedor através de <i>WebService</i> , será disparada uma consulta direta ao sistema do fornecedor para saber o <i>status</i> da negociação. | Alta |

3.1.4 Recursos do Sistema

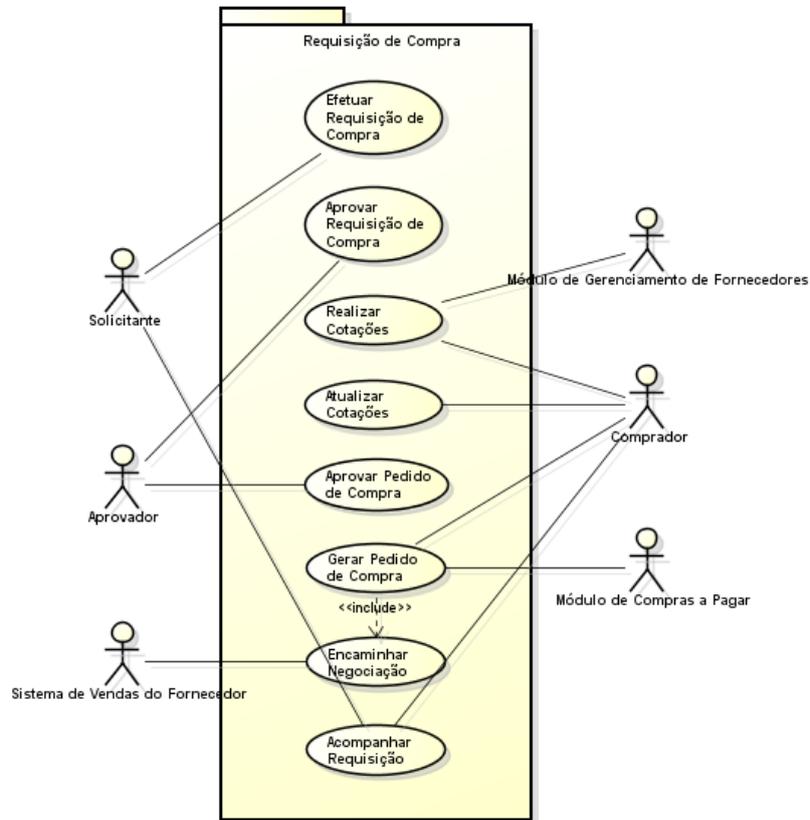
Após implementação das funcionalidades pretendidas para o processo, o sistema deverá ser capaz de:

- Incluir requisição de compra;
- Aprovar requisição de compra;
- Realizar cotações para as requisições;
- Encaminhar a negociação do pedido de compra ao fornecedor;
- Acompanhar o andamento das requisições.

3.1.5 Informações Complementares

O processo de Requisição de Compra possui integração com o Módulo de Gerenciamento de Fornecedores e também com o Módulo de Contas a Pagar.

3.1.6 Visão Geral UML – Diagrama de Casos de Uso



4 Outros Requisitos

4.1 Requisitos de Operação

- O sistema deverá operar em plataforma WEB e ser desenvolvido utilizando uma das linguagens utilizadas nas disciplinas de programação dos cursos de tecnologia da Universidade de Caxias do Sul.

4.2 Requisitos de Segurança

- O sistema terá um procedimento de autenticação de usuários, os quais deverão que se identificar usando um *login* e uma senha. O controle de permissões de acesso será realizado com base neste *login*.

4.3 Requisitos de Processo

- As atividades da engenharia de requisitos deverão contar com o apoio de uma ferramenta CASE apropriada.

**ERP NUSIS
Glossário**

Versão 1.0

| | | |
|---|---------------------------------------|----------------|
|  UCS UNIVERSIDADE DE CAXIAS DO SUL | Universidade de Caxias do Sul – NUSIS | Versão: 1.0 |
| | ERP NUSIS: Glossário | Data: 13/06/13 |

Histórico de Revisão

| Date | Version | Description | Autor |
|-------------|----------------|--------------------|--------------------|
| 11/06/2013 | 1.0 | Criado documento | Rodrigo Debastiani |
| 13/06/2013 | 1.0 | Adicionados termos | Rodrigo Debastiani |
| | | | |
| | | | |

| | | |
|---|---------------------------------------|----------------|
|  | Universidade de Caxias do Sul – NUSIS | Versão: 1.0 |
| | ERP NUSIS: Glossário | Data: 13/06/13 |

Glossário

1. Introdução

Este documento define a terminologia utilizada nos diferentes documentos e especificações do sistema ERP desenvolvido pelo NUSIS

1.1 Finalidade

Este documento é utilizado para definir a terminologia específica do domínio do problema, explicando termos, que poderão ser desconhecidos para o leitor (como siglas, abreviaturas e definições), das descrições de caso de uso ou de outros documentos do projeto. Este documento lista, em ordem alfabética, os termos utilizados para descrever os diferentes processos do ERP desenvolvido pelo NUSIS e pode ser usado como um dicionário de dados informal, capturando definições de dados para que as descrições de casos de uso e outros documentos do projeto possam se concentrar no que o sistema deve fazer com as informações.

1.2 Escopo

Este glossário trata de todos os termos que possuem significados específicos no sistema ERP do NUSIS.

Os atores não se encontram listados nele porque serão descritos, de forma mais detalhada, nas especificações dos casos de uso.

Todas as siglas ou termos técnicos utilizados nos documentos produzidos para o projeto estão listados neste glossário.

2. Acrônimos

ERP – Planejamento de Recursos Empresariais (Enterprise Resource Planning)

MRP – Planejamento da Necessidade de Material (Material Requirement Planning)

XML – Extensible Markup Language (Linguagem Extensível de Marcação Genérica)

3. Definições

Centro de Custo – unidade organizacional que representa uma localização e serve para delimitar a origem dos custos de uma empresa ou para onde estão indo suas despesas.

Falha – uma manifestação de um erro, que causa uma operação defeituosa no sistema.

Prospect – condição daquilo que é provável; um cliente ou fornecedor em potencial.

Trilha de aprovação – caminho de autorização pelo qual uma requisição de compra deve seguir até que seja aprovada.

Webservice – solução utilizada na integração de sistemas e na comunicação entre aplicações diferentes. Permitem às aplicações enviar e receber dados em formato XML.

Universidade de Caxias do Sul – NUSIS

ERP NUSIS
Especificações Suplementares

Versão 1.0

| | | |
|---|---|----------------|
|  | Universidade de Caxias do Sul – NUSIS | Versão: 1.0 |
| | ERP NUSIS: Especificações Suplementares | Data: 20/06/13 |

Histórico de Revisão

| Data | Versão | Descrição | Autor |
|-------------|---------------|-------------------|--------------------|
| 20/06/2013 | 1.0 | Criado documento. | Rodrigo Debastiani |
| | | | |
| | | | |
| | | | |

| | | |
|---|---|----------------|
|  UCS UNIVERSIDADE DE CAXIAS DO SUL | Universidade de Caxias do Sul – NUSIS | Versão: 1.0 |
| | ERP NUSIS: Especificações Suplementares | Data: 20/06/13 |

Índice Analítico

| | | |
|-----|---|---|
| 1. | Introdução | 4 |
| 1.1 | Finalidade | 4 |
| 1.2 | Escopo | 4 |
| 1.3 | Definições, Acrônimos e Abreviações | 4 |
| 1.4 | Referências | 4 |
| 2. | Funcionalidade | 4 |
| 3. | Usabilidade | 4 |
| 4. | Confiabilidade | 4 |
| 5. | Desempenho | 4 |
| 6. | Suportabilidade | 4 |
| 7. | Restrições de Design | 5 |
| 8. | Requisitos de Sistema de Ajuda e de Documentação de Usuário On-line | 5 |
| 9. | Componentes Comprados | 5 |
| 10. | Interfaces | 5 |
| 11. | Requisitos de Licenciamento | 5 |
| 12. | Observações Legais, de Direitos Autorais etc | 5 |

| | | |
|--|---|----------------|
| | Universidade de Caxias do Sul – NUSIS | Versão: 1.0 |
| | ERP NUSIS: Especificações Suplementares | Data: 20/06/13 |

Especificações Suplementares

4. Introdução

Este documento de Especificações Suplementares captura os requisitos de sistema que não foram identificados imediatamente nas especificações de casos de uso. Entre estes requisitos estão incluídos:

- Requisitos legais e reguladores, incluindo padrões de aplicativo;
- Atributos de qualidade do sistema a ser criado, incluindo requisitos de usabilidade, confiabilidade, desempenho e suportabilidade;
- Outros requisitos, como sistemas operacionais e ambientes, requisitos de compatibilidade e restrições de design.

4.1 Finalidade

A finalidade deste documento é definir os requisitos não funcionais para o sistema ERP desenvolvido pelo NUSIS. Esta Especificação Suplementar lista os requisitos que não foram imediatamente identificados nas Especificações de Casos de Uso. As Especificações Suplementares e as Especificações de Casos de Uso, juntos, originam o conjunto completo de requisitos do sistema ERP do NUSIS.

4.2 Escopo

Este documento de Especificações Suplementares aplica-se ao projeto do sistema ERP desenvolvido pelo NUSIS, e define requisitos não funcionais como: requisitos de usabilidade, confiabilidade, desempenho e suportabilidade, bem como alguns requisitos de funcionalidade, comuns a vários casos de uso, entre outros requisitos.

4.3 Definições, Acrônimos e Abreviações

Vide documento Glossário.

4.4 Referências

Glossário

5. Funcionalidade

A serem definidos.

6. Usabilidade

6.1 Facilidade de Uso

É necessário que a interface com o usuário do sistema ERP NUSIS seja intuitiva e fácil de usar, conceitos de usabilidade deverão ser implementados.

7. Confiabilidade

A confiabilidade será um fator de extrema importância por se tratar de um sistema de tempo real.

8. Desempenho

8.1 Acessos Simultâneos

É necessário que o ERP NUSIS capaz de suportar acessos simultâneos.

9. Suportabilidade

A serem definidos.

| | | |
|---|---|----------------|
|  | Universidade de Caxias do Sul – NUSIS | Versão: 1.0 |
| | ERP NUSIS: Especificações Suplementares | Data: 20/06/13 |

10. Restrições de Design

A serem definidos.

11. Requisitos de Sistema de Ajuda e de Documentação de Usuário On-line

É necessário disponibilizar para o usuário, através da Web, a documentação dos processos do ERP existentes no sistema.

12. Componentes Comprados

A serem definidos.

13. Interfaces

A serem definidas.

14. Requisitos de Licenciamento

A serem definidos.

15. Observações Legais, de Direitos Autorais e Outras

A serem definidas.

| | | |
|---|--|----------------|
|  | Processo: Requisição de Compra | Versão: 1.0 |
| | Narrativa de Uso: Requisição de café para a matriz | Data: 25/05/13 |

Narrativa de Uso: Requisição de café para a matriz

Narrativa de Uso

Descrição

Setor de almoxarifado requisita café para a matriz, e requisição é aprovada automaticamente pelo sistema.

Fluxo Normal

A empresa terceirizada responsável pela limpeza e também pelo preparo de café para os diferentes setores da matriz verificou que a quantidade de café disponível será suficiente para apenas mais uma semana. Na matriz, várias marcas de café foram experimentadas ao longo dos dois últimos anos, sendo que apenas a última marca adquirida não recebeu reclamações. Ao ser comunicado pela empresa terceirizada, o setor de almoxarifado entra com a requisição de compra no sistema, indicando o código do material, no caso o café da marca "A". Devido ao grande consumo de café na empresa, e não se tratando de um item perecível, foi solicitada uma quantidade suficiente para os próximos 60 dias. Após preencher e submeter o formulário de requisição de compra no sistema, o usuário do almoxarifado prontamente recebeu um aviso em tela contendo o status da requisição, que neste caso fora aprovada, e um número identificador gerado automaticamente pelo sistema.

O que pode dar errado

Seção não considerada nesta narrativa.

| | | |
|---|---|----------------|
|  | Processo: Requisição de Compra | Versão: 1.0 |
| | Narrativa de Uso: Requisição de notebook para funcionário | Data: 25/05/13 |

Narrativa de Uso: Requisição de notebook para funcionário

Narrativa de Uso

Descrição

Coordenador da equipe de vendas que deseja requisitar a aquisição de um notebook para um novo funcionário no setor.

Fluxo Normal

Recentemente um novo funcionário foi contratado para o setor de vendas. Este novo funcionário atuará em campo, visitando *prospects* e tirando pedidos em clientes da região. Pelos próximos trinta dias ele permanecerá em treinamento junto a outros funcionários da empresa. Passado esse período ele passará a exercer as atividades que competem a sua função. Para isso, o coordenador da equipe de vendas necessita requisitar um notebook novo para seu funcionário. O coordenador não possui qualquer conhecimento de informática para poder detalhar aspectos de *hardware* e *software* para a nova aquisição, e entende que tal avaliação tenha de ficar sob responsabilidade de um funcionário especializado do setor de TI. Após algumas trocas de e-mail com funcionários da TI, o coordenador de vendas preenche o formulário de requisição de compra no sistema, detalhando no campo de “Justificativa” os objetivos de utilização deste novo equipamento, indicando o funcionário que o utilizará, e quais os requisitos de *hardware* e *software* necessários de acordo com o parecer técnico obtido da TI. Após submeter a requisição de compra, lhe é retornado um número identificador e o status que a requisição encontra-se pendente aprovação com o responsável pelo seu centro de custo.

O que pode dar errado

A requisição que deveria cair para aprovação do centro de custo foi autorizada prontamente pelo sistema, caindo direto na etapa de Cotações. Neste caso, devem ser revistas as regras de aprovação parametrizadas.

| | | |
|---|---|----------------|
|  | Processo: Requisição de Compra | Versão: 1.0 |
| | Narrativa de Uso: Requisição de aparelho celular com contrato de fornecimento | Data: 25/05/13 |

Narrativa de Uso: Requisição de aparelho celular com contrato de fornecimento

Narrativa de Uso

Descrição

Empregado que necessita solicitar aparelho celular, que por sua vez possui contato de fornecimento.

Fluxo Normal

Recentemente Fabiano, consultor da empresa na região, teve seu aparelho celular extraviado. Tal aparelho era cedido pela empresa sem nenhum custo, para fins exclusivamente profissionais, uma vez que a atribuição do colaborador necessita de constante contato com a fábrica. Após registrar o furto junto à polícia, Fabiano decidiu solicitar um novo aparelho para a empresa, pois sentiu que suas tarefas diárias estavam sendo prejudicadas pela ausência de um celular. Antes de abrir a requisição, ele conversou com seu superior, que o instruiu a incluir a requisição. Nos detalhes da inclusão da requisição, foi justificada a necessidade do aparelho, bem como o fato de seu responsável direto estar ciente do ocorrido. Após aberta, a requisição caiu inicialmente para aprovação do próprio coordenador de Fabiano, que prontamente autorizou a requisição. Em um segundo momento, devido ao tipo de material solicitado, a requisição teve de ser autorizada também pelo gerente responsável pelo centro de custo a qual pertence Fabiano. Após aprovada, a requisição chegou ao setor de compras, e neste caso caiu diretamente com um funcionário específico, encarregado por esse tipo de compra. Ao verificar no sistema os dados da requisição, é destacado para o funcionário de compras a existência de contrato para esse material. Devido à quantidade de consultores, vendedores e outros funcionários que necessitam de telefonia móvel, a empresa firmou um contrato de fornecimento com uma operadora, possuindo ela total exclusividade. Desta forma, não faz-se necessário cotar aparelhos celulares com outros fornecedores. Ainda no sistema, o funcionário seleciona a opção existente que lhe permite verificar quem é o fornecedor do material. Junto às informações do fornecedor são trazidos também os detalhes do contrato. O funcionário analisa o contrato onde está disposto o modelo de aparelho que é fornecido e seu respectivo valor, e não encontrando nada que reprove a compra, encaminha a requisição para aprovação do pedido de compra.

O que pode dar errado

Seção não considerada nesta narrativa.

| | | |
|---|---------------------------------------|----------------|
|  | Processo: Requisição de Compra | Versão: 1.0 |
| | Narrativa de Uso: Cotação de folha A4 | Data: 26/05/13 |

Narrativa de Uso: Cotação de folha A4

Narrativa de Uso

Descrição

Setor de compras recebe uma requisição do almoxarifado, que após ser aprovada chega a etapa de realização das cotações. Após escolha dos fornecedores as cotações são realizadas e passadas para aprovação, que num primeiro momento rejeita as cotações obtidas, para somente depois de uma nova cotação aprovar o pedido de compra.

Fluxo Normal

O setor de compras recebeu uma requisição para compra de material de escritório. Tal ordem partiu de um funcionário do setor de almoxarifado, que averiguou que estava para acabar o estoque de folha A4 da unidade. É grande o consumo deste tipo de material pelos setores administrativos da empresa, logo, foi solicitada quantidade suficiente para no mínimo seis meses. Após ser aprovada prontamente pelo coordenador do almoxarifado, a requisição foi encaminhada para o setor de compras. Fábio, é o funcionário do setor de compras que ficou responsável pela requisição. Ele analisa a solicitação e opta por consultar no sistema o histórico de compras desse material, identificando os últimos valores negociados e seus respectivos fornecedores. Após verificar o histórico, Fábio consulta a relação de todos os fornecedores do material. Sabendo que tal item deve possuir ao mínimo três cotações, Fábio seleciona na lista de fornecedores retornada pelo sistema o último fornecedor, e além deste, mais dois fornecedores locais. A seleção destes outros dois fornecedores é feita com base em critérios de ranqueamento estabelecidos pela empresa para seus fornecedores, que considera dentre outros fatores o comprometimento com os prazos de entrega, a flexibilidade nas formas de pagamento e a facilidade de relacionamento. Uma vez selecionados os fornecedores, Fábio realiza o contato para solicitar a cotação com cada um deles. Prontamente os retornos para as cotações são obtidos, as informações recebidas são agregadas junto à requisição de compra. Após analisar os retornos, Fábio pondera que a melhor cotação ainda pertence ao fornecedor atual do material, mas como não é sua atribuição dentro do setor de Compras escolher quem será o fornecedor, ele apenas acrescenta seu parecer e transfere a requisição e suas cotações para avaliação do responsável por escolher e aprovar a melhor cotação, aquela que irá resultar no pedido de compra.

O que pode dar errado

Seção não considerada nesta narrativa.

| | | |
|---|---|----------------|
|  | Processo: Requisição de Compra | Versão: 1.0 |
| | Narrativa de Uso: Cotação e aprovação de pedido para o setor de Marketing | Data: 27/05/13 |

Narrativa de Uso: Cotação e aprovação de pedido para o setor de Marketing

Narrativa de Uso

Descrição

Setor de compras recebe uma requisição de material oriunda da área de marketing. Após aprovada pelo centro de custo, a requisição chega à etapa de cotações. Se tratando de um produto novo, é realizado o contato com o setor responsável pelo cadastramento de fornecedores para prospecção de fornecedores. Após realizar as cotações, as mesmas foram primeiramente rejeitadas pelo gerente responsável, para somente após nova negociação com um dos fornecedores ter o pedido de compra aprovado.

Fluxo Normal

A área de marketing em sua nova campanha publicitária decidiu por distribuir *squeezes* como brinde aos clientes. Com isso, a responsável pelo projeto inclui no sistema uma requisição de duas mil unidades do item. Uma vez aprovada por parte da gerente do centro de custo, a requisição cai para o setor de compras, que após contato com os responsáveis pela prospecção de fornecedores recebe retorno de três cotações para o material solicitado. Dentre essas, a melhor cotação pertence a um fornecedor da cidade vizinha que já fornece materiais de divulgação da marca da empresa como canecas, canetas e agendas. João, o funcionário responsável pelas cotações as analisa individualmente, e adiciona seu parecer, indicando a que no seu entendimento é a melhor cotação. Transferida para aprovação do pedido de compra, o gerente responsável rejeita as cotações, e insatisfeito com os valores apresentados solicita que o setor de compras busque outro fornecedor com preço mais em conta. Após nova pesquisa de mercado, outro fornecedor para o material é encontrado. Após ser cadastrado e aprovado, tal fornecedor passa a estar “ativo” no sistema, podendo ser candidato no processo de cotação. Com isso, João envia uma solicitação de cotação para o novo fornecedor, cuja resposta apresenta o mesmo valor do atual fornecedor desse tipo de material para a empresa. Com tal retorno, João decide negociar com o fornecedor já conhecido, uma vez que a empresa está satisfeita com o serviço de entrega e também com as condições de pagamento, nunca tendo tido qualquer tipo de problema. Após alguns e-mails, o fornecedor cede, e envia nova proposta com 10% de desconto sobre o valor anteriormente repassado. João então encaminha novamente a requisição para aprovação do pedido de compra contendo a nova melhor cotação, satisfazendo desta vez seu gestor, que rapidamente aprova o pedido.

O que pode dar errado

Na inexistência de fornecedores para o material, o setor responsável pelo cadastramento e prospecção de fornecedores deverá ser contatado.

| | | |
|---|--|----------------|
|  | Processo: Requisição de Compra | Versão: 1.0 |
| | Narrativa de Uso: Geração e encaminhamento de pedido de copos descartáveis | Data: 25/05/13 |

Narrativa de Uso: Geração e encaminhamento de pedido de copos descartáveis

Narrativa de Uso

Descrição

Setor de compras gera pedido de compra após o mesmo ter sido aprovado, e encaminha negociação para o fornecedor.

Fluxo Normal

Após ser aprovada, a requisição com o pedido de compra de copos descartáveis oriundo do setor de almoxarifado retorna ao setor de compras. Henrique, o comprador encarregado desta requisição a seleciona em tela e aciona opção para geração do pedido. Automaticamente o sistema envia os dados do pedido para a área de contas a pagar, gerando uma despesa para a empresa. Após ser informado do sucesso na geração do pedido de compra, Henrique decide por encaminhar a negociação ao fornecedor. Neste momento, após verificar as parametrizações de negociação com o fornecedor, o sistema dispara um e-mail para o endereço eletrônico parametrizado com os dados do pedido de compra gerado através da cotação obtida.

O que pode dar errado

Fornecedor pode não possuir parametrização de negociação, neste caso, não deve ser habilitada opção para encaminhamento da negociação.

| | | |
|---|--|----------------|
|  | Processo: Requisição de Compra | Versão: 1.1 |
| | Especificação do Caso de Uso: Efetuar Requisição de Compra | Data: 02/06/13 |

Especificação de Caso de Uso: Efetuar Requisição de Compra

Informações

Objetivo no Contexto

Solicitante requisita compra de material.

Ator Principal

Solicitante

Stakeholders e Interesses:

Solicitante: Deseja que o material requisitado esteja disponível para utilização.

Aprovador: Deseja controlar gastos e adquirir somente os materiais realmente necessários. Deseja receber para análise somente as requisições que não forem aprovadas automaticamente pelo sistema.

Comprador: Deseja receber como entrada para o processo de cotação somente as requisições aprovadas automaticamente pelo sistema.

Pré-condições

Solicitante deve estar logado no sistema e possuir permissão para efetuar requisição de compra.

Garantias

Garantias Mínimas

Toda requisição de compra submetida será analisada de acordo com as políticas de compra de materiais da empresa, podendo ser aprovada automaticamente pelo sistema ou então cair como pendente de aprovação.

Garantias de Sucesso

Requisição de compra incluída com sucesso, com status de aprovada ou então pendente de aprovação.

Fluxo de Eventos

Acionador

Solicitante que necessita requisitar material.

Cenário de Sucesso Principal

1. *Solicitante*: preenche formulário para requisição de compra.
2. *Sistema*: valida dados informados em tela e pede confirmação ao Solicitante para inclusão da requisição.
3. *Solicitante*: confirma a inclusão da requisição.
4. *Sistema*: verifica regras de aprovação através do código/tipo de material requisitado.
5. *Sistema*: verifica regras de aprovação através do centro de custo a qual o Solicitante pertence.
6. *Sistema*: verifica regras de aprovação através da faixa de valor do material requisitado.
7. *Sistema*: cria trilha de aprovação da requisição a partir das regras de aprovação encontradas, com o caminho de aprovação que a requisição terá de seguir, podendo este caminho possuir até três níveis

| | | |
|--|--|----------------|
| | Processo: Requisição de Compra | Versão: 1.1 |
| | Especificação do Caso de Uso: Efetuar Requisição de Compra | Data: 02/06/13 |

de aprovação.

8. *Sistema:* aprova requisição ou a deixa pendente de aprovação.
9. *Sistema:* informa ao Solicitante o número da requisição e o seu status.

Extensões

- 2a. Sistema identifica que campos obrigatórios da requisição não foram preenchidos: retornar erro ao Solicitante alertando sobre a obrigatoriedade do preenchimento dos campos.
- 3a. Solicitante cancela inclusão de requisição: limpar dados do formulário e permanecer na tela de inclusão.
- 8a. De acordo com a trilha de aprovação, o Sistema pode:
 - 8a1. Aprovar automaticamente a requisição, caso não haja a necessidade de aprovação.
 - 8a2. Encaminhar a requisição para o primeiro nível de aprovação, atribuindo-a a um grupo de usuários aprovadores ou a um único usuário dentro de um grupo.

Lista de Variações Tecnológicas e de Dados

Nenhuma.

Protótipos de Tela

ERP/NUSIS - Módulo de Compras - Requisicao de Compra

http://localhost:8780/erp/compras

Requisição de Compra

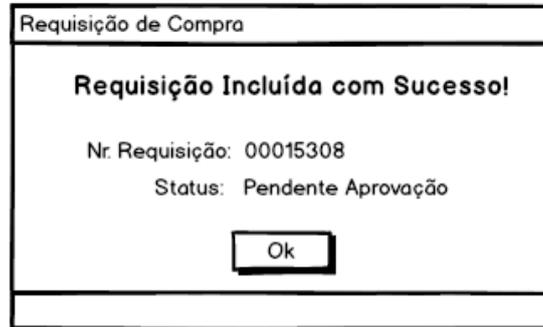
Solicitante: Data Solicitação:

Centro de Custo: Prioridade:

Justificativa:

| Cod.Material | Descricao Material | Qtd. |
|--------------|--|--------|
| 00000303 | Pacote de folha branca A4 branca c/100 | 350.00 |
| | | |
| | | |
| | | |

| | | |
|--|--|----------------|
| | Processo: Requisição de Compra | Versão: 1.1 |
| | Especificação do Caso de Uso: Efetuar Requisição de Compra | Data: 02/06/13 |



Casos de Uso Relacionados

Aprovar Pedido de Requisição

Histórico de Revisão

| Data | Versão | Descrição | Autor |
|----------|--------|---|--------------------|
| 01/06/13 | 1.0 | Criado documento de especificação de caso de uso. | Rodrigo Debastiani |
| 03/06/13 | 1.1 | Adicionado protótipo de tela. | Rodrigo Debastiani |
| | | | |
| | | | |

| | | |
|---|--|----------------|
|  | Processo: Requisição de Compra | Versão: 1.1 |
| | Especificação de Caso de Uso: Aprovar Requisição de Compra | Data: 02/06/13 |

Especificação de Caso de Uso: Aprovar Requisição de Compra

Informações

Objetivo no Contexto

Aprovador autoriza ou nega requisição de compra.

Ator Principal

Aprovador.

Stakeholders e Interesses

Solicitante: Deseja ser comunicado sobre o status de sua Requisição.

Aprovador: Deseja controlar gastos e adquirir somente os materiais realmente necessários. Deseja receber para análise somente as requisições que não forem aprovadas automaticamente pelo sistema.

Comprador: Deseja receber como entrada para o processo de cotação somente as requisições aprovadas.

Pré-condições

Aprovador deve estar logado no sistema e possuir permissão para efetuar aprovação de requisição de compra. Requisições devem estar com status pendente de aprovação.

Garantias

Garantias Mínimas

Toda requisição de compra com status pendente de aprovação será analisada individualmente por um Aprovador, podendo ser negada, autorizada e repassada para o próximo nível de aprovação, ou então, não havendo mais níveis de aprovação pendentes, autorizada e encaminhada aos responsáveis pelo processo de cotação.

Garantias de Sucesso

Requisição de compra negada, autorizada e repassada para próximo nível de aprovação, ou então autorizada e encaminhada para cotação.

Fluxo de Eventos

Acionador

Aprovador que tem de analisar requisições de compra.

Cenário de Sucesso Principal

1. *Aprovador*: filtra em tela as requisições pendentes de aprovação para o seu usuário ou grupo e seleciona a requisição a ser analisada.
2. *Sistema*: apresenta as informações da requisição de compra selecionada.
3. *Aprovador*: analisa requisição, acrescenta e/ou modifica informações, e adiciona parecer/justificativa.
4. *Sistema*: valida dados alterados e também a obrigatoriedade da informação do parecer/justificativa.

| | | |
|--|--|----------------|
| | Processo: Requisição de Compra | Versão: 1.1 |
| | Especificação de Caso de Uso: Aprovar Requisição de Compra | Data: 02/06/13 |

5. *Aprovador*: autoriza requisição de compra.
6. *Sistema*: pede confirmação ao Aprovador para autorização da requisição.
7. *Aprovador*: confirma a autorização.
8. *Sistema*: verifica trilha de aprovação da requisição. Caso não hajam níveis de aprovação pendentes então encaminha a requisição para processo de cotação e altera seu status para pendente cotação, senão a mantém como pendente aprovação e a repassa para o próximo nível de aprovação.
9. *Sistema*: informa ao Aprovador o status da requisição e dispara e-mail comunicando Solicitante.

Extensões

- 1a. Aprovador filtra requisições através do número da requisição, usuário solicitante, data de inclusão, código/tipo de material, centro de custo ou faixa de valor.
- 4a. Sistema identifica que campos obrigatórios não foram preenchidos: retornar erro ao Aprovador alertando sobre a obrigatoriedade do preenchimento dos campos.
- 5a. Aprovador nega requisição: Sistema retira requisição das pendências de aprovação e atribui status de Requisição negada à mesma. Sistema dispara e-mail comunicando Solicitante.
- 7a. Aprovador cancela a autorização: Sistema retorna a tela anterior com as informações da requisição de compra selecionada.

Lista de Variações Tecnológicas e de Dados

Nenhuma.

Protótipos de Tela

ERP/NUSIS - Módulo de Compras - Aprovação de Requisição

http://localhost:8780/erp/compras

Aprovar Requisição de Compra

Listar Requisições Pendentes c/: Usuário Grupo

Filtrar por: Data de Solicitação ▼

Dt.Solicitação: 15/05/13

| Dt.Solic. | Nr. Requisição | Solicitante | Centro de Custo |
|-----------|----------------|--------------------------|-----------------------|
| 15/05/13 | 000015305 | 00124-Fernando Silva | Recursos Humanos |
| 15/05/13 | 000015308 | 00037-Rodrigo Debastiani | Almoxarifado-Matriz |
| 15/05/13 | 000015310 | 00085-Felipe Silveira | Vendas-Sorocaba |
| 15/05/13 | 000015311 | 00085-Felipe Silveira | Vendas-Sorocaba |
| 15/05/13 | 000015313 | 00113-Maria Rubens | Almoxarifado-Sorocaba |

Selecionar Cancelar

ERP/NUSIS - Modulo de Compras - Aprovacao de Requisicao

http://localhost:8780/erp/compras

Dados da Requisição | Materiais Requisitados

Nr Requisição: 000015308

Solicitante: 00037-Rodrigo Debastiani Data da Solicitação: 15/05/13

Centro de Custo: Almoarifado - Matriz Prioridade: Baixa

Justificativa: Conforme solicitado pelo Sr. Henrique, o estoque de folha A4 está no fim. Calcula-se que será suficiente para apenas mais 15 dias.

Parecer / Justificativa do Aprovador:

Ação:

Autorizar

Negar

Voltar

ERP/NUSIS - Modulo de Compras - Aprovacao de Requisicao

http://localhost:8780/erp/compras

Dados da Requisição | Materiais Requisitados

| Cod.Material | Descricao Material | Qtd. | Vi.Sistema |
|--------------|--|--------|------------|
| 00000303 | Pacote de folha branca A4 branca c/100 | 250.00 | R\$8.45 |
| | | | |
| | | | |
| | | | |

Alterar Material

Parecer / Justificativa do Aprovador:

Após contato com o Sr. Henrique, supervisor do setor de Almoarifado, foi acordado que neste mês serão solicitadas apenas 250.00 unidades, as quais acreditamos ser suficiente para os próximos 60 dias. Uma vez averiguada a necessidade de mais material, uma nova requisição terá de ser aprovada. A diminuição da quantidade leva em

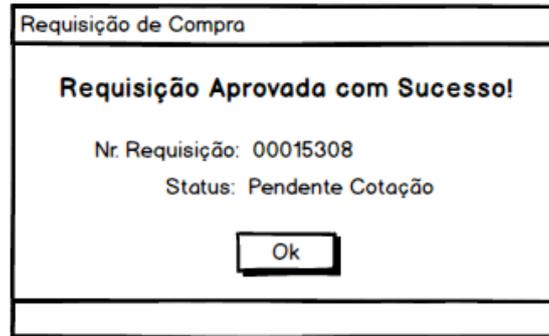
Ação:

Autorizar

Negar

Voltar

| | | |
|---|--|----------------|
|  | Processo: Requisição de Compra | Versão: 1.1 |
| | Especificação de Caso de Uso: Aprovar Requisição de Compra | Data: 02/06/13 |



Casos de Uso Relacionados

Efetuar Requisição de Compra.
Realizar Cotações.

Histórico de Revisão

| Data | Versão | Descrição | Autor |
|----------|--------|---|--------------------|
| 01/06/13 | 1.0 | Criado documento de especificação de caso de uso. | Rodrigo Debastiani |
| 02/06/13 | 1.1 | Adicionada extensão 7a. | Rodrigo Debastiani |
| | | | |
| | | | |

| | | |
|--|---|----------------|
| | Processo: Requisição de Compra | Versão: 1.1 |
| | Especificação de Caso de Uso: Realizar Cotações | Data: 11/06/13 |

Especificação de Caso de Uso: Realizar Cotações

Informações

Objetivo no Contexto

Comprador obtém cotações para o material requisitado.

Ator Principal

Comprador

Stakeholders e Interesses

Solicitante: Deseja que o material requisitado esteja disponível para utilização.

Aprovador: Deseja receber a requisição para aprovação do pedido de compra com o mínimo de cotações estabelecidas ou então com o contrato de fornecimento analisado.

Comprador: Deseja identificar os fornecedores melhor ranqueados para realizar o mínimo de cotações estabelecidas ou então receber a informação sobre a existência de contrato de fornecimento.

Pré-condições

Requisição de compra deve ter sido devidamente autorizada por todos aprovadores.

Garantias

Garantias Mínimas

Toda requisição de compra será analisada e terá suas cotações criadas baseadas nas informações retornadas pelo Módulo de Gerenciamento de Fornecedores, ou, na existência de contrato de fornecimento para o material, o mesmo será avaliado antes da requisição ser transferida para aprovação do pedido de compra.

Garantias de Sucesso

Requisição com cotações realizadas aguardando retorno por parte dos fornecedores ou então requisição encaminhada para aprovação do pedido de compra com seu contrato de fornecimento analisado.

Fluxo de Eventos

Acionador

Comprador que necessita obter cotações para o material requisitado.

Cenário de Sucesso Principal

1. *Comprador*: filtra em tela as requisições pendentes de cotação ainda sem usuário responsável.
2. *Sistema*: retorna através de uma lista as requisições de acordo com o filtro aplicado.
3. *Comprador*: seleciona a requisição a ser analisada.
4. *Sistema*: apresenta as informações da requisição de compra selecionada e seus aprovadores, e os materiais requisitados com a quantidade mínima de cotações necessárias e também a informação sobre a existência de contrato de fornecimento para o material.
5. *Comprador*: através de uma opção existente assume responsabilidade sobre a requisição.
6. *Sistema*: atribui Comprador como usuário responsável pela atividade de obtenção de cotações da requisição.

| | | |
|---|---|----------------|
|  | Processo: Requisição de Compra | Versão: 1.1 |
| | Especificação de Caso de Uso: Realizar Cotações | Data: 11/06/13 |

7. *Comprador*: aciona a opção que permite visualizar o histórico de últimas compras do material.
8. *Sistema*: retorna em um *grid* as informações das últimas compras do material, apresentando as quantidades negociadas, os fornecedores, datas, valores e a formas de pagamento.
9. *Comprador*: seleciona opção de busca de fornecedores, ainda na tela com as informações da requisição.
10. *Sistema*: verifica relação material x fornecedor junto ao Módulo de Gerenciamento de Fornecedores.
11. *Módulo de Gerenciamento de Fornecedores*: retorna ao Sistema uma lista com os fornecedores ativos melhor ranqueados de acordo com parametrizações vigentes, destacando entre estes o fornecedor da última compra.
12. *Sistema*: apresenta em uma nova tela as informações retornadas pelo Módulo de Gerenciamento de Fornecedores.
13. *Comprador*: utiliza das informações retornadas sobre os fornecedores, e com base na quantidade mínima de cotações necessárias seleciona os fornecedores para os quais serão solicitadas cotações.
14. *Sistema*: gera um registro dentro da requisição de compra com o status de cotação pendente para cada um dos fornecedores selecionados pelo Comprador, indicando a pendência do retorno do fornecedor para a cotação solicitada. Informa o Comprador sobre o êxito na criação da(s) cotação(ões).

Extensões

- 1a. Comprador filtra requisições através do número da requisição, usuário solicitante, data de inclusão, código/tipo de material, centro de custo ou faixa de valor.
- 11a. Módulo de Gerenciamento de Fornecedores não localiza nenhum registro na relação material x fornecedor: retornar erro ao Sistema indicando a ausência de fornecedor para o material requisitado.
- 11b. Módulo de Gerenciamento de Fornecedores identifica que material possui contrato de fornecimento: retorna ao Sistema apenas o fornecedor do contrato em vigor.
- 13a. Retornado erro pelo Sistema indicando a ausência de fornecedor para o material: contatar responsáveis pela manutenção e prospecção de fornecedores.
- 13b. Retornada a existência de contrato fornecimento, examina informações e estando de acordo encaminha requisição para a aprovação do pedido de compra.
- 13c. Retornada a existência de contrato de fornecimento, e não satisfazendo interesses atuais: Comprador mantém requisição como pendente de cotação enquanto entra em contato com responsáveis pelo contrato.

Lista de Variações Tecnológicas e de Dados

Nenhuma.

Protótipos de Tela

A serem adicionados ao documento.

Casos de Uso Relacionados

Atualizar Cotações.
Aprovar Pedido de Compra.

| | | |
|---|---|----------------|
|  | Processo: Requisição de Compra | Versão: 1.1 |
| | Especificação de Caso de Uso: Realizar Cotações | Data: 11/06/13 |

Histórico de Revisão

| Data | Versão | Descrição | Autor |
|-------------|---------------|--|--------------------|
| 01/06/13 | 1.0 | Criado documento de especificação de caso de uso. | Rodrigo Debastiani |
| 01/06/13 | 1.1 | Alterações no Cenário de Sucesso Principal relacionadas as informações das últimas compras do material. Comprador passa a dispor de opção em tela para consulta dessas informações. | Rodrigo Debastiani |
| 11/06/13 | 1.2 | Alterações no Cenário de Sucesso Principal relacionadas ao status que a requisição após realizadas as cotações. Até então a transação passava para Pendente Avaliação, agora passa para Pendente Aprovação de Cotação. | Rodrigo Debastiani |
| | | | |

| | | |
|---|--|----------------|
|  | Processo: Requisição de Compra | Versão: 1.1 |
| | Especificação de Caso de Uso: Atualizar Cotações | Data: 11/06/13 |

Especificação de Caso de Uso: Atualizar Cotações

Informações

Objetivo no Contexto

Comprador atualiza cotações com os retornos obtidos dos fornecedores.

Ator Principal

Comprador

Stakeholders e Interesses

Solicitante: Deseja que o material requisitado esteja disponível para utilização.

Aprovador: Deseja receber a requisição para aprovação do pedido de compra com o mínimo de cotações estabelecidas.

Comprador: Deseja atualizar os registros das cotações criadas com os dados retornados pelo fornecedor.

Pré-condições

Requisição de compra deve ter passado pela etapa de realização das cotações e possuir um ou mais registros de cotações criados.

Garantias

Garantias Mínimas

Toda requisição de compra será transferida para a etapa de aprovação do pedido de compra contendo o número mínimo de cotações estabelecidas de acordo com o Módulo de Gerenciamento de Fornecedores, e com todos dados obrigatórios preenchidos.

Garantias de Sucesso

Requisição encaminhada para aprovação do pedido de compra contendo o mínimo de cotações estabelecidas com todos dados obrigatórios preenchidos.

Fluxo de Eventos

Acionador

Comprador que necessita obter cotações para o material requisitado.

Cenário de Sucesso Principal

1. *Comprador*: filtra em tela as requisições pendentes de cotação para o seu usuário.
2. *Sistema*: retorna através de uma lista as requisições de acordo com o filtro aplicado.
3. *Comprador*: seleciona a requisição para a qual serão atualizadas as cotações.
4. *Sistema*: apresenta as informações da requisição de compra selecionada e seus aprovadores, e os materiais requisitados com a quantidade mínima de cotações necessárias e também a informação sobre a existência de contrato de fornecimento para o material.
5. *Comprador*: aciona opção para poder visualizar as cotações geradas para a requisição.
6. *Sistema*: valida a existência de cotações, e apresenta em uma nova tela as cotações existentes para

| | | |
|---|--|----------------|
|  | Processo: Requisição de Compra | Versão: 1.1 |
| | Especificação de Caso de Uso: Atualizar Cotações | Data: 11/06/13 |

a requisição.

7. *Comprador*: seleciona a cotação que terá seus dados atualizados de acordo com o retorno obtido junto ao fornecedor.
8. *Sistema*: verifica o status da cotação selecionada, e estando como cotação pendente então habilita para preenchimento os campos da cotação.
9. *Comprador*: preenche o formulário da cotação com os dados recebidos com o retorno do fornecedor para a cotação solicitada. Uma vez concluída a atualização aciona a opção de salvamento.
10. *Sistema*: valida se os dados obrigatórios foram preenchidos, e altera o status do registro da cotação passando-a de cotação pendente para cotação recebida.
11. *Sistema*: a cada nova interação do Comprador junto às cotações, verifica se todos os registros de cotação já receberam retorno por parte dos fornecedores. Uma vez recebidos todos os retornos e estando todas as cotações com status de cotação recebida, informa ao usuário que a etapa de cotações foi concluída, e pede confirmação sobre a transferência da requisição de compra para a etapa de aprovação do pedido, onde será feita a análise e escolha da cotação que irá gerar o pedido de compra. Nesse momento, o Comprador poderá adicionar um parecer geral para as cotações, além dos pareceres individuais de cada cotação que puderam ser acrescentados ao longo do processo.
12. *Comprador*: confirma transferência da requisição de compra para a etapa de aprovação do pedido.
13. *Sistema*: solicita usuário ou grupo para o qual será transferida a requisição.
14. *Comprador*: informa o dado solicitado e finaliza a transferência.
15. *Sistema*: atribui responsabilidade sobre a requisição ao usuário ou grupo informado, altera o status da requisição para pendente aprovação de pedido.
16. *Sistema*: informa ao Comprador o novo status da requisição e dispara e-mail comunicando Solicitante.

Extensões

- 1a. Comprador filtra requisições através do número da requisição, usuário solicitante, data de inclusão, código/tipo de material, centro de custo ou faixa de valor.
- 6a. Sistema não encontra nenhum registro de cotação para a requisição: retornar alerta ao Comprador informando que requisição ainda não possui cotação.
- 8 a. Sistema verifica que cotação selecionada possui status de cotação recebida: alertar o Comprador sobre o status e solicitar confirmação para seguir com a alteração dos dados.
 - 8 a1. Comprador cancela alteração da cotação: Sistema retorna a tela anterior com as cotações existentes para a requisição.
- 10a. Sistema identifica que campos obrigatórios da cotação não foram preenchidos: retornar alerta ao Comprador informando sobre a obrigatoriedade do preenchimento dos campos e manter status como cotação pendente.
- 11a. Sistema verifica que ainda existem cotações com status de cotação pendente: manter requisição de compra como pendente de cotação.
- 12a. Comprador cancela transferência da requisição de compra para a etapa de análise e escolha da melhor cotação: Sistema deve manter requisição de compra como pendente de cotação.

Lista de Variações Tecnológicas e de Dados

Nenhuma.

Protótipos de Tela

A serem adicionados ao documento.

| | | |
|---|--|----------------|
|  | Processo: Requisição de Compra | Versão: 1.1 |
| | Especificação de Caso de Uso: Atualizar Cotações | Data: 11/06/13 |

Casos de Uso Relacionados

Realizar Cotações
Aprovar Pedido de Compra.

Histórico de Revisão

| Data | Versão | Descrição | Autor |
|----------|--------|--|--------------------|
| 01/06/13 | 1.0 | Criado documento de especificação de caso de uso. | Rodrigo Debastiani |
| 01/06/13 | 1.1 | Alterações no Cenário de Sucesso Principal relacionadas as informações das últimas compras do material. Comprador passa a dispor de opção em tela para consulta dessas informações. | Rodrigo Debastiani |
| 11/06/13 | 1.2 | Alterações no Cenário de Sucesso Principal relacionadas ao status que a requisição após realizadas as cotações. Até então a transação passava para Pendente Avaliação, agora passa para Pendente Aprovação de Cotação. | Rodrigo Debastiani |
| | | | |

| | | |
|---|--|----------------|
|  | Processo: Requisição de Compra | Versão: 1.1 |
| | Especificação de Caso de Uso: Aprovar Pedido de Compra | Data: 03/06/13 |

Especificação de Caso de Uso: Aprovar Pedido de Compra

Informações

Objetivo no Contexto

Aprovador avalia cotações e aprova a melhor cotação para geração do pedido de compra.

Ator Principal

Aprovador.

Stakeholders e Interesses

Solicitante: Deseja que o material requisitado esteja disponível para utilização.

Aprovador: Deseja avaliar as cotações e escolher aquela que apresente melhor custo-benefício para a empresa, ou então, avaliar o contrato de fornecimento antes de aprovar o pedido.

Comprador: Deseja que o pedido de compra seja aprovado para dar seguimento ao processo de geração do pedido.

Pré-condições

Requisição de compra deve estar com status pendente aprovação de pedido. Requisição de compra deve possuir o número mínimo de cotações de acordo com o material.

Garantias

Garantias Mínimas

Toda requisição com status pedido de compra aprovado terá sido analisada por um Aprovador.

Garantias de Sucesso

Requisição de compra analisada e aprovada para geração de pedido de compra.

Fluxo de Eventos

Acionador

Aprovador que tem de avaliar cotações e aprovar o pedido de compra.

Cenário de Sucesso Principal

1. *Aprovador*: filtra em tela as requisições pendentes de aprovação de pedido para o seu usuário ou grupo e seleciona a requisição a ser analisada.
2. *Sistema*: apresenta as informações da requisição de compra selecionada, com as cotações realizadas e o parecer geral do Comprador de forma resumida.
3. *Aprovador*: avalia a requisição e utiliza a opção para detalhar cada uma das cotações.
4. *Sistema*: ao ser selecionada a opção de detalhamento da cotação, apresenta os dados informados pelo Comprador com o retorno da solicitação de cotação obtida junto ao fornecedor. Apresenta também um indicador sobre a existência de anexos.
5. *Aprovador*: dentro do detalhamento da cotação, aciona a função para visualizar o anexo existente.

| | | |
|---|--|----------------|
|  | Processo: Requisição de Compra | Versão: 1.1 |
| | Especificação de Caso de Uso: Aprovar Pedido de Compra | Data: 03/06/13 |

6. *Sistema*: busca o anexo no diretório onde o mesmo foi armazenado e o abre em tela.
7. *Aprovador*: após visualizar cada uma das cotações, volta para a tela com os dados da requisição, e aciona a opção para detalhar o parecer geral do Comprador sobre a requisição e suas cotações.
8. *Sistema*: apresenta em uma nova tela de forma completa o parecer geral adicionado pelo Comprador na etapa de Realizar Cotações.
9. *Aprovador*: avalia parecer do Comprador e retorna a tela principal com os dados da requisição.
10. *Sistema*: volta para a tela aonde são apresentadas as informações da requisição de compra selecionada, com as cotações realizadas e o parecer geral do Comprador de forma resumida.
11. *Aprovador*: após analisar os dados da requisição, as cotações individualmente, e também o parecer geral do Comprador, escolhe a melhor cotação, selecionando-a em tela e acionando a função para aprovar pedido de compra.
12. *Sistema*: pede confirmação ao Aprovador para a aprovação do pedido de compra.
13. *Aprovador*: confirma a aprovação.
14. *Sistema*: atribui Aprovador como usuário responsável pela aprovação do pedido de compra e altera status da requisição para pedido de compra aprovado. Atribui o mesmo Comprador da atividade de Realizar Cotações como responsável pela etapa de Gerar Pedido de Compra.
15. *Sistema*: informa ao Aprovador o novo status da requisição e dispara e-mail comunicando Solicitante.

Extensões

- 1a. Aprovador filtra requisições através do número da requisição, usuário solicitante, data de inclusão, código/tipo de material, centro de custo ou faixa de valor.
- 2a. Sistema identifica que material requisitado possui contrato de fornecimento e neste caso não requer cotação: apresentar informação indicando a existência de contrato e quem é o fornecedor do mesmo.
- 3a. Para material com contrato de fornecimento, Aprovador seleciona opção para visualizar contrato: Sistema detalha dados do contrato e indica a existência de anexo.
 - 3a1 Aprovador aciona opção para visualizar anexo: Sistema busca o anexo no diretório onde o mesmo foi armazenado e o abre em tela.
 - 3a1a. Sistema não consegue localizar anexo: apresenta erro informando da impossibilidade de localizar anexo.
- 6a. Sistema não consegue localizar anexo: apresenta erro informando da impossibilidade de localizar anexo.
- 11a. Aprovador não satisfeito com as cotações obtidas rejeita requisição: Sistema solicita justificativa para rejeição.
 - 11a1. Aprovador informa justificativa e confirma rejeição da requisição: Sistema atribui novamente status de pendente cotação e retorna requisição para a etapa de Realizar Cotações.
- 11b. Para material com contrato de fornecimento, após analisar os dados da requisição e também o parecer geral do Comprador, aciona a função para aprovar pedido.
- 13a. Aprovador cancela a aprovação do pedido de compra: Sistema retorna a tela anterior com as informações da requisição de compra.

Lista de Variações Tecnológicas e de Dados

Nenhuma.

Protótipos de Tela

A serem adicionados ao documento.

| | | |
|---|--|----------------|
|  | Processo: Requisição de Compra | Versão: 1.1 |
| | Especificação de Caso de Uso: Aprovar Pedido de Compra | Data: 03/06/13 |

Casos de Uso Relacionados

Atualizar Cotações.
Gerar Pedido de Compra.

Histórico de Revisão

| Data | Versão | Descrição | Autor |
|----------|--------|---|---------------------|
| 01/06/13 | 1.0 | Criado documento de especificação de caso de uso. | Rodrigo Debastiani |
| 03/06/13 | 1.1 | Alterado nome do caso de uso de “Analisar e Escolher Cotação” para “Aprovar Pedido de Compra” | Rodrigo Debastiani. |
| | | | |
| | | | |

| | | |
|---|--|----------------|
|  | Processo: Requisição de Compra | Versão: 1.0 |
| | Especificação de Caso de Uso: Gerar Pedido de Compra | Data: 06/06/13 |

Especificação de Caso de Uso: Gerar Pedido de Compra

Informações

Objetivo no Contexto

Comprador gera pedido de compra para o material requisitado.

Ator Principal

Comprador

Stakeholders e Interesses

Solicitante: Deseja que o material requisitado esteja disponível para utilização.

Comprador: Deseja gerar o pedido de compra para a requisição após aprovação a mesma ter sido devidamente aprovada.

Pré-condições

Garantias

Garantias Mínimas

Todo pedido de compra gerado a partir de uma requisição será integrado com o Módulo de Contas a Pagar. Um pedido de compra de material sem contrato de fornecimento só será gerado se a cotação escolhida estiver com data de validade válida.

Garantias de Sucesso

Pedido de compra incluído com sucesso e integrado com o Módulo de Contas a Receber. Requisição com status pedido de compra gerado.

Fluxo de Eventos

Acionador

Comprador que deseja gerar o pedido de compra para a requisição.

Cenário de Sucesso Principal

1. *Comprador*: filtra em tela as requisições com status pedido de compra aprovado cujo responsável seja o seu usuário e seleciona a requisição a ser analisada.
2. *Sistema*: apresenta as informações da requisição de compra selecionada, trazendo as cotações realizadas e ressaltando dentre estas a cotação aprovada para geração do pedido de compra e o usuário responsável pela aprovação.
3. *Comprador*: aciona a função de geração do pedido de compra.
4. *Sistema*: monta através de uma nova tela o formulário para geração do pedido de compra, trazendo como *default* as informações já existentes no registro da requisição e também da cotação aprovada.
5. *Comprador*: avalia os dados preenchidos e conclui geração do pedido.
6. *Sistema*: valida se todos os campos obrigatórios foram preenchidos e se a data de validade da cotação é válida, e pede confirmação para geração do pedido de compra.

| | | |
|--|--|----------------|
| | Processo: Requisição de Compra | Versão: 1.0 |
| | Especificação de Caso de Uso: Gerar Pedido de Compra | Data: 06/06/13 |

7. *Comprador*: confirma geração do pedido.
8. *Sistema*: efetua integração do pedido de compra com o Módulo de Contas a Pagar.
9. *Módulo de Contas a Pagar*: gera uma previsão de despesa através do pedido de compra integrado e retorna para o Sistema um identificador com o número da despesa gerada.
10. *Sistema*: armazena o identificador com o número da despesa retornado pelo Módulo Contas a Pagar junto ao pedido de compra gerado.
11. *Sistema*: altera o status da requisição para pedido de compra gerado e atribui Comprador como usuário responsável pela geração do pedido.
12. *Sistema*: informa ao Comprador o novo status da requisição e dispara e-mail comunicando Solicitante.
13. *Sistema*: encaminha a negociação do pedido.

Extensões

- 1a. Comprador filtra requisições através do número da requisição, usuário solicitante, data de inclusão, código/tipo de material, centro de custo ou faixa de valor.
- 2a. Sistema identifica que material requisitado possui contrato de fornecimento e neste caso não possui cotação: apresentar informação indicando a existência de contrato e quem é o fornecedor do mesmo.
- 6a. Sistema identifica que campos obrigatórios da geração de pedido não foram preenchidos: retornar erro ao Comprador alertando sobre a obrigatoriedade do preenchimento dos campos.
- 6b. Sistema identifica que cotação indicada possui data de validade expirada: retornar erro ao Comprador alertando sobre a expiração da cotação.
- 7a. Comprador cancela a geração do pedido: retornar a tela com o formulário do pedido com os dados já preenchidos.
- 10a. Sistema recebe retorno da integração do pedido com erro por parte do Módulo de Contas a Pagar: apresentar erro alertando para a falha na integração do pedido e retornar a tela com o formulário do pedido com os dados já preenchidos.

Lista de Variações Tecnológicas e de Dados

Nenhuma.

Protótipos de Tela

A serem adicionados ao documento.

Casos de Uso Relacionados

Encaminhar Negociação.

Histórico de Revisão

| Data | Versão | Descrição | Autor |
|----------|--------|---|--------------------|
| 02/06/13 | 1.0 | Criado documento de especificação de caso de uso. | Rodrigo Debastiani |
| | | | |
| | | | |

| | | |
|---|---|----------------|
|  | Processo: Requisição de Compra | Versão: 1.0 |
| | Especificação de Caso de Uso: Encaminhar Negociação | Data: 11/06/13 |

Especificação de Caso de Uso: Encaminhar Negociação

Informações

Objetivo no Contexto

Comprador encaminha negociação para o fornecedor.

Ator Principal

Comprador

Stakeholders e Interesses

Solicitante: Deseja que o material requisitado esteja disponível para utilização.

Comprador: Deseja encaminhar ao fornecedor o pedido de compra gerado com base na cotação obtida.

Pré-condições

Requisição com status de pedido de compra gerado e forma de encaminhamento da negociação parametrizada.

Garantias

Garantias Mínimas

Os pedidos de compra cujos fornecedores possuem parametrização para encaminhamento da negociação poderão ser encaminhados.

Garantias de Sucesso

Todo pedido de compra que for encaminhado ao fornecedor será identificado com a forma de encaminhamento, data e hora de ocorrência,

Fluxo de Eventos

Acionador

Comprador que deseja encaminhar negociação do pedido de compra gerado.

Cenário de Sucesso Principal

1. *Sistema*: após informar ao Comprador o novo status da requisição e ter disparado e-mail para comunicar o Solicitante, questiona em tela ao Comprador se o mesmo deseja encaminhar a negociação de compra.
2. *Comprador*: confirma o interesse em encaminhar a negociação.
3. *Sistema*: verifica parametrização de negociação do fornecedor junto ao Módulo de Gerenciamento de Fornecedores.
4. *Módulo de Gerenciamento de Fornecedores*: retorna indicador com a existência de parametrização para encaminhamento da negociação, e junto a ela os dados pertinentes à forma de encaminhamento: e-mail ou *webservice*.
5. *Sistema*: valida indicador de parametrização retornado pelo Módulo de Gerenciamento de Fornecedores, monta negociação de acordo com a forma parametrizada e dispara encaminhamento.

| | | |
|--|---|----------------|
| | Processo: Requisição de Compra | Versão: 1.0 |
| | Especificação de Caso de Uso: Encaminhar Negociação | Data: 11/06/13 |

6. *Sistema*: aguarda retorno do encaminhamento, marca o pedido de compra como negociação encaminhada atribuindo a forma de encaminhamento, a data e a hora de ocorrência, e informa ao Comprador sobre o sucesso no envio da negociação.

Extensões

- 2a. Comprador opta por não encaminhar negociação: Sistema retorna a tela de inicial de geração dos pedidos de compra.
- 5a. Sistema verifica que não existe forma de encaminhamento parametrizada para o fornecedor de acordo com o indicador retornado pelo Módulo de Gerenciamento de Fornecedores: Sistema informa ao Comprador sobre a inexistência de parametrização para encaminhamento da negociação, e retorna a tela inicial de geração dos pedidos de compra.
- 6a. Sistema identifica falha ao encaminhar negociação: informa ao Comprador que não foi possível encaminhar negociação e aponta o erro ocorrido

Lista de Variações Tecnológicas e de Dados

Nenhuma.

Protótipos de Tela

A serem adicionados ao documento.

Casos de Uso Relacionados

Gerar Pedido de Compra.

Histórico de Revisão

| Data | Versão | Descrição | Autor |
|----------|--------|---|--------------------|
| 06/06/13 | 1.0 | Criado documento de especificação de caso de uso. | Rodrigo Debastiani |
| | | | |
| | | | |
| | | | |

| | | |
|---|---|----------------|
|  | Processo: Requisição de Compra | Versão: 1.0 |
| | Especificação de Caso de Uso: Acompanhar Requisição | Data: 11/06/13 |

Especificação de Caso de Uso: Acompanhar Requisição

Informações

Objetivo no Contexto

Solicitante e/ou Comprador desejam acompanhar requisição de compra.

Ator Principal

Solicitante/Comprador.

Stakeholders e Interesses

Solicitante: Deseja acompanhar andamento da requisição de compra.

Comprador: Deseja verificar situação da negociação de compra encaminhada ao fornecedor.

Pré-condições

Requisição de compra incluída no sistema.

Garantias

Garantias Mínimas

Será retornado o histórico da requisição de compra, com os usuários responsáveis pelas atividades. Quando o perfil do usuário for o Comprador serão retornadas informações pertinentes ao status atual da negociação.

Garantias de Sucesso

Retornadas informações da requisição de compra com base em seu histórico.

Fluxo de Eventos

Acionador

Solicitante que deseja saber o andamento de sua requisição.

Comprador que deseja verificar o andamento da negociação encaminhada ao fornecedor.

Cenário de Sucesso Principal

1. *Solicitante/Comprador*: seleciona uma das opções possíveis para seleção da requisição a ser consultada.
2. *Sistema*: retorna através de uma lista as requisições de acordo com a seleção adotada.
3. *Solicitante/Comprador*: seleciona a requisição a ser consultada.
4. *Sistema*: apresenta o histórico da requisição, com todas as etapas pela qual a mesma já passou e o usuário responsável por cada uma delas. Identificando o perfil do usuário como Comprador, caso a requisição esteja com status pedido de compra gerado, verifica se foi encaminhada negociação para o fornecedor. Caso tenha sido encaminhada negociação, e a forma de encaminhamento tenha sido através de *WebService*, habilita opção para verificar status da negociação. Caso o encaminhamento tenha ocorrido via e-mail, indica em tela tal informação. Habilita também opção que permite o detalhamento dos dados da requisição.

| | | |
|---|---|----------------|
|  | Processo: Requisição de Compra | Versão: 1.0 |
| | Especificação de Caso de Uso: Acompanhar Requisição | Data: 11/06/13 |

5. *Comprador*: aciona opção para verificar status da negociação.
6. *Sistema*: monta transação para consulta do status da negociação e dispara para o *WebService* disponibilizado pelo fornecedor.
7. *Sistema*: aguarda retorno da consulta, com uma mensagem de processamento através de uma nova tela. Uma vez obtido apresenta em tela o status retornado pelo sistema do fornecedor.

Extensões

- 1a. Comprador filtra requisições através do número da requisição, usuário solicitante, data de inclusão, código/tipo de material, centro de custo ou faixa de valor.
- 7a. Sistema identifica falha ao encaminhar a consulta do status da negociação: informa ao Comprador que não foi possível obter o status da negociação e aponta o erro ocorrido

Lista de Variações Tecnológicas e de Dados

Nenhuma.

Protótipos de Tela

A serem adicionados ao documento.

Casos de Uso Relacionados

Nenhum.

Histórico de Revisão

| Data | Versão | Descrição | Autor |
|-------------|---------------|---|--------------------|
| 11/06/13 | 1.0 | Criado documento de especificação de caso de uso. | Rodrigo Debastiani |
| | | | |
| | | | |
| | | | |