



MINISTÉRIO DA CIÊNCIA, TECNOLOGIA, INOVAÇÕES E COMUNICAÇÕES
INSTITUTO NACIONAL DE PESQUISAS ESPACIAIS

sid.inpe.br/mtc-m21b/2017/05.29.21.16-TDI

DEFINIÇÃO E APLICAÇÃO DE UM PROCESSO PARA GERENCIAMENTO DE REQUISITOS A SISTEMAS AEROESPACIAIS

Fernando Luiz Ferreira de Azevedo

Dissertação de Mestrado do Curso de Pós-Graduação em Engenharia e Tecnologia Espaciais/Engenharia e Gerenciamento de Sistemas Espaciais, orientada pelo Dr. Marcelo Lopes de Oliveira e Souza, aprovada em 26 de maio de 2017.

URL do documento original:

<http://urlib.net/8JMKD3MGP3W34P/3P27HKH>

INPE
São José dos Campos
2017

PUBLICADO POR:

Instituto Nacional de Pesquisas Espaciais - INPE

Gabinete do Diretor (GB)

Serviço de Informação e Documentação (SID)

Caixa Postal 515 - CEP 12.245-970

São José dos Campos - SP - Brasil

Tel.:(012) 3208-6923/6921

E-mail: pubtc@inpe.br

**COMISSÃO DO CONSELHO DE EDITORAÇÃO E PRESERVAÇÃO
DA PRODUÇÃO INTELECTUAL DO INPE (DE/DIR-544):****Presidente:**

Maria do Carmo de Andrade Nono - Conselho de Pós-Graduação (CPG)

Membros:

Dr. Plínio Carlos Alvalá - Centro de Ciência do Sistema Terrestre (CST)

Dr. André de Castro Milone - Coordenação de Ciências Espaciais e Atmosféricas (CEA)

Dra. Carina de Barros Melo - Coordenação de Laboratórios Associados (CTE)

Dr. Evandro Marconi Rocco - Coordenação de Engenharia e Tecnologia Espacial (ETE)

Dr. Hermann Johann Heinrich Kux - Coordenação de Observação da Terra (OBT)

Dr. Marley Cavalcante de Lima Moscati - Centro de Previsão de Tempo e Estudos Climáticos (CPT)

Silvia Castro Marcelino - Serviço de Informação e Documentação (SID)

BIBLIOTECA DIGITAL:

Dr. Gerald Jean Francis Banon

Clayton Martins Pereira - Serviço de Informação e Documentação (SID)

REVISÃO E NORMALIZAÇÃO DOCUMENTÁRIA:

Simone Angélica Del Duca Barbedo - Serviço de Informação e Documentação (SID)

Yolanda Ribeiro da Silva Souza - Serviço de Informação e Documentação (SID)

EDITORAÇÃO ELETRÔNICA:

Marcelo de Castro Pazos - Serviço de Informação e Documentação (SID)

André Luis Dias Fernandes - Serviço de Informação e Documentação (SID)



MINISTÉRIO DA CIÊNCIA, TECNOLOGIA, INOVAÇÕES E COMUNICAÇÕES
INSTITUTO NACIONAL DE PESQUISAS ESPACIAIS

sid.inpe.br/mtc-m21b/2017/05.29.21.16-TDI

DEFINIÇÃO E APLICAÇÃO DE UM PROCESSO PARA GERENCIAMENTO DE REQUISITOS A SISTEMAS AEROESPACIAIS

Fernando Luiz Ferreira de Azevedo

Dissertação de Mestrado do Curso de Pós-Graduação em Engenharia e Tecnologia Espaciais/Engenharia e Gerenciamento de Sistemas Espaciais, orientada pelo Dr. Marcelo Lopes de Oliveira e Souza, aprovada em 26 de maio de 2017.

URL do documento original:

<<http://urlib.net/8JMKD3MGP3W34P/3P27HKH>>

INPE
São José dos Campos
2017

Dados Internacionais de Catalogação na Publicação (CIP)

Azevedo, Fernando Luiz Ferreira de.

Az25d Definição e aplicação de um processo para gerenciamento de requisitos a sistemas aeroespaciais / Fernando Luiz Ferreira de Azevedo. – São José dos Campos : INPE, 2017.

xxvi + 224 p. ; (sid.inpe.br/mtc-m21b/2017/05.29.21.16-TDI)

Dissertação (Mestrado em Engenharia e Tecnologia Espaciais/Engenharia e Gerenciamento de Sistemas Espaciais) – Instituto Nacional de Pesquisas Espaciais, São José dos Campos, 2017.

Orientador : Dr. Marcelo Lopes de Oliveira e Souza.

1. Requisitos. 2. Engenharia de requisitos. 3. Gerenciamento de requisitos. 4. Controle de alterações. 5. Engenharia de sistemas. I.Título.

CDU 629.7.01



Esta obra foi licenciada sob uma Licença [Creative Commons Atribuição-NãoComercial 3.0 Não Adaptada](https://creativecommons.org/licenses/by-nc/3.0/).

This work is licensed under a [Creative Commons Attribution-NonCommercial 3.0 Unported License](https://creativecommons.org/licenses/by-nc/3.0/).

Aluno (a): **Fernando Luiz Ferreira de Azevedo**

Título: "DEFINIÇÃO E APLICAÇÃO DE UM PROCESSO PARA GERENCIAMENTO DE REQUISITOS A SISTEMAS AEROESPACIAIS".

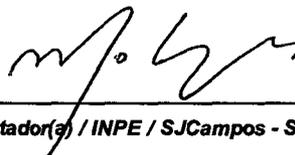
Aprovado (a) pela Banca Examinadora em cumprimento ao requisito exigido para obtenção do Título de **Mestre** em **Engenharia e Tecnologia Espaciais/Eng. Gerenc. de Sistemas Espaciais**

Dr. **Maurício Gonçalves Vieira Ferreira**



Presidente / INPE / SJCampos - SP

Dr. **Marcelo Lopes de Oliveira e Souza**



Orientador(a) / INPE / SJCampos - SP

Dr. **Otávio Luiz Bogossian**



Membro da Banca / INPE / São José dos Campos - SP

Dr. **Fernando José de Oliveira Moreira**



Convidado(a) / EMBRAER / SJCampos - SP

Este trabalho foi aprovado por:

maioria simples

unanimidade

São José dos Campos, 26 de maio de 2017

“Success consists of going from failure to failure without loss of enthusiasm”.

Winston Churchill

*À minha bela e obstinada esposa
pelo seu apoio, paciência e
companheirismo, as minhas filhas
e, sobretudo, em memória dos meus
pais, pela educação que me foi
dedicada, apesar de todas as
adversidades.*

AGRADECIMENTOS

Ao termo deste trabalho gostaria de externar meus agradecimentos:

A Deus por ter-me permitido superar as dificuldades encontradas ao longo desta jornada, sem perder o entusiasmo.

Ao INPE pela oportunidade que me foi dada de aprimorar-me profissionalmente.

Aos professores do Curso de Engenharia e Tecnologia Espaciais, da Área de Engenharia e Gerenciamento de Sistemas Espaciais, pelo tempo dedicado a um ensino de altíssimo padrão.

Em especial agradeço ao Professor Dr. Marcelo Lopes de Oliveira e Souza, por sua valiosa orientação, sem a qual seria impossível realizar esta pesquisa, e ainda, por sua impressionante dedicação à missão de compartilhar, e incentivar a busca do conhecimento, e pelo apoio, singular e contagiante, que coloca à disposição dos seus alunos.

Aos membros da Banca Examinadora Prof. Dr. Mauricio Gonçalves Vieira Ferreira, Prof. Dr. Otávio Luiz Bogossian, Dr. Fernando José de Oliveira Moreira.

Ao Instituto de Aeronáutica e Espaço (IAE), do Comando da Aeronáutica, pela liberação do tempo para que eu pudesse frequentar as aulas do curso de mestrado.

Ao Tcel Sandro Rondon Rett, do IAE, pelo incentivo e apoio administrativo para iniciar o curso.

Ao Luiz Aldo Leite das Neves, adjunto de minha função no IAE e amigo, por seu apoio constante, substituindo-me durante minhas ausências no trabalho, o que foi imprescindível para que eu pudesse realizar o curso de mestrado.

Ao Tcel Waldecy Oliveira da Silva, do IAE, pelo apoio na fase final deste trabalho.

Aos engenheiros do INPE, Mario Celso Padovan de Almeida, Ana Paula de Sá Santos Rabello, Inaldo Soares de Albuquerque e Hadler Egydio da Silva, pelas informações fornecidas e ajuda inestimável.

Ao engenheiro do INPE Marcio Silva Alves Branco pelo apoio inestimável ao disponibilizar a base de dados para demonstração de aplicabilidade do processo proposto e apoio na utilização do software PTC Integrity.

Aos servidores do INPE: David Cristiano dos Santos pelo apoio para tornar disponível equipamento e acesso ao software PTC Integrity e Edgar Varela por ajuda na utilização do software PTC Integrity.

À Valdirene Moreira de Paula e à Edleusa Aparecida Ferreira pelo apoio nos processos administrativos relativos ao curso.

À Yolanda Ribeiro da Silva Souza e ao André Luis Dias Fernandes, da Biblioteca do INPE, pelo trabalho de revisão deste trabalho.

Enfim, a todos que de forma direta, ou não, contribuíram para a elaboração desta pesquisa, o meu muito obrigado!

RESUMO

A presente pesquisa tem por objetivo principal propor e aplicar um processo, para o gerenciamento de requisitos de sistemas aeroespaciais, com aplicabilidade geral à Engenharia Aeroespacial, abrangendo o controle das alterações ocorridas ao longo do desenvolvimento de um sistema/produto. Para tanto, é realizada uma análise e comparação de quatro abordagens de gerenciamento de requisitos, apresentadas em documentos de engenharia de sistemas da NASA, ECSS, INCOSE e IEEE. Nestas abordagens são apresentados, de forma geral, os processos, atividades e recursos que devem estar presentes para o tratamento de requisitos, durante todo o ciclo de vida do sistema, deixando, no entanto, de explicitar a forma como isto deverá ser realizado. Para isto, propôs-se um processo que agregou os principais aspectos e as boas práticas presentes na citada documentação, com instruções detalhadas do processo de execução do controle de mudanças, apresentando fluxos e etapas necessárias, para a condução de um processo, organizado, para o gerenciamento de requisitos, em face das mudanças que, naturalmente, ocorrem durante o desenvolvimento de um produto/sistema. O processo proposto está baseado em quatro macroprocessos: o tratamento das mudanças, o monitoramento, a avaliação e a elaboração de uma base de conhecimento formada a partir de lições aprendidas e foi testado através de aplicação em um estudo de caso do INPE, com o uso de uma ferramenta de software da instituição, para gerenciamento de requisitos, como suporte computacional para o método. Os resultados obtidos, com a aplicação prática do processo proposto, mostraram-se úteis para a melhoria do processo de tratamento de requisitos do projeto; e mostram que o processo serve como instrumento de coleta, e armazenamento de conhecimentos a serem utilizados para subsidiar o processo decisório, em projetos futuros.

Palavras-chave: Requisitos. Engenharia de Requisitos. Gerenciamento de Requisitos. Controle das Alterações. Engenharia de Sistemas.

DEFINITION AND APPLICATION OF A REQUIREMENTS MANAGEMENT PROCESS FOR AEROSPACE SYSTEMS

ABSTRACT

The objective of this research is to define and apply a process for requirements management of aerospace systems, with general applicability to Aerospace Engineering, covering the control of changes happened along the development of a system/product. To do so, an analysis and comparison of four requirements management approaches presented in NASA, ECSS, INCOSE and IEEE system engineering documents, about the suggested process, is performed. These approaches proposes, in general, the processes, activities and resources that must be present for the treatment of requirements throughout the system lifecycle, but not explaining, however, how this should be accomplished. To do that, we defined a process that joined the main aspects and good practices, recommended by the analyzed documentation, with detailed instructions on how to carry out the execution processes of control changes, presenting the necessary flow and steps, for the conduction of an organized process for the management, of changes that naturally occur during the development of a product/system. The proposed process is based on four macro processes: the change management, monitoring, evaluation and the development of a knowledge base generated from lessons learned, it was tested through its application in a case study of INPE, with the use of an institution software tool, for requirements management, as a computational support for the method. The obtained results with the practical application of the proposed process have proved useful in improving the process of managing project requirements and show that the process serves as a tool for collecting and storing knowledge to be used to support the decision-making process, on other projects.

Keywords: Requirements. Requirements Engineering, Requirements Management. Change Management. Systems Engineering

LISTA DE FIGURAS

	<u>Pág.</u>
Figura 1.1 - Resultados das 10 principais práticas de gerenciamento de requisitos em organizações de alto desempenho.	7
Figura 2.1 - Desenvolvimento de pequenos programas.	10
Figura 2.2 - Desenvolvimento de grandes programas.	10
Figura 2.3 - Diagrama IDEF0	15
Figura 2.4 - Rastreabilidade bidirecional.	22
Figura 2.5 - Alguns possíveis elos de rastreamento de requisitos.	23
Figura 2.6 - Subdisciplinas de Engenharia de Requisitos de software.	32
Figura 2.7 - Mudanças nos requisitos.	34
Figura 2.8 - Limites entre o desenvolvimento e o gerenciamento de requisitos.	37
Figura 3.1 - Modelo da Engenharia de Sistemas.	43
Figura 3.2 - Processo de gerenciamento de requisitos – NASA.	44
Figura 3.3 - Processo de gerenciamento de mudanças – NASA.	46
Figura 3.4 - Funções e fronteiras da Engenharia de Sistemas- ECSS.	50
Figura 3.5 - Funções da Engenharia de Sistemas e relacionamentos-ECSS. .	51
Figura 3.6 - Gerenciamento de Configuração-ECSS.	54
Figura 3.7 - Implementação do Gerenciamento de Configuração-ECSS.	56
Figura 3.8 - Fluxo, alocação e derivação de requisitos - INCOSE.	61
Figura 3.9 - Diagrama de contexto para o processo de Gerenciamento de Configuração-INCOSE.	62
Figura 3.10 - Decomposição de tópicos para Requisitos de Software – IEEE.	69
Figura 3.11 - Fluxo do Processo de Controle de Mudanças - IEEE.	71
Figura 3.12 – Workflow do processo de mudanças de engenharia no INPE. ..	72
Figura 3.13 - Processo para mudanças de engenharia - INPE.	73
Figura 4.1 - Processo proposto.	98
Figura 4.2 - Página inicial para acesso aos formulários.	102
Figura 4.3 - Página com dados de identificação e detalhes da mudança proposta.	103

Figura 4.4 - Página com dados da análise preliminar da mudança proposta.	109
Figura 4.5 - Página com dados para análise de impacto da mudança proposta.	117
Figura 4.6 - Roteiro para auxílio à análise de impacto da mudança proposta.	118
Figura 4.7 - Roteiro para identificar artefatos afetados pela mudança proposta.	119
Figura 4.8 - Página com dados de verificação da mudança proposta.....	123
Figura 4.9 - Página com dados de validação da mudança proposta.....	126
Figura 4.10 - Página com aprovação/rejeição pelo CCB.....	128
Figura 4.11 - Página da base de conhecimento proposta.	156
Figura 4.12 - Processos do processo proposto.	158
Figura 4.13 - Processo 4.1 - Tratar mudanças.	159
Figura 4.14 - Processo 4.1.1 - Analisar solicitação.	160
Figura 4.15 - Processo 4.1.1.1 - Realizar análise preliminar.	161
Figura 4.16 - Processo 4.1.1.2 - Analisar impactos.	162
Figura 4.17 - Processo 4.1.2 - Alterar contrato.....	163
Figura 4.18 - Processo 4.1.4 – Dar manutenção à <i>Baseline</i>	164
Figura 4.19 - Processo 4.2 - Monitorar.....	165
Figura 4.20 - Processo 4.2.2 - Auditar.....	166
Figura 4.21 - Processo 4.3 - Avaliar.	167
Figura 4.22 - Processo 4.4 - Gerenciar base de conhecimento.	168
Figura 5.1 - Solicitação de mudança em requisito.....	172
Figura 5.2 - Análise preliminar.....	174
Figura 5.3 – Análise de impacto.	176
Figura 5.4 - Documento de especificação de requisitos.	177
Figura 5.5 – Detalhe de tela de proposta de mudança de requisito.	177
Figura 5.6 - Tela de módulo de análise de impacto.....	178
Figura 5.7 - Detalhe da tela de análise de impacto.	179
Figura 5.8 - Detalhe da tela de análise de impacto.	180
Figura 5.9 - Formulário de verificação.....	182
Figura 5.10 - Formulário de validação.....	182
Figura 5.11 - Formulário de aprovação do CCB.....	183

Figura 5.12 - Formulário de resultado de avaliação.	186
Figura 5.13 - Formulário de manutenção da base de conhecimentos.....	188

LISTA DE TABELAS

	<u>Pág.</u>
Tabela 2.1 - Tabela de rastreabilidade entre requisitos.	26
Tabela 2.2 - Características individuais de requisitos.	28
Tabela 2.3 - Características do conjunto de requisitos.....	29
Tabela 3.1 - Comparação de pontos-chaves nas abordagens de: NASA, INCOSE, IEEE, ECSS.....	76
Tabela 4-1 - Modelo para suporte de decisão baseado em respostas.....	101
Tabela 4.2 - Atributos da solicitação de mudança.....	108
Tabela 4.3 - Roteiro para auxiliar nas atividades do processo de análise de impacto das mudanças dos requisitos solicitadas.....	111
Tabela 4.4 - Roteiro para identificar artefatos que serão afetados com a realização da mudança solicitada.	112
Tabela 4.5 - Formulário para a estimativa do esforço a ser empreendido para uma mudança de requisito.	112
Tabela 4.6 - Indicadores.....	114
Tabela 4.7 - Verificação de volatilidade dos requisitos.....	116
Tabela 4.8 - Classificação de defeitos para avaliação de requisitos e suas alterações.....	137
Tabela 4.9 - Classes de defeitos e características sob inspeção.....	139
Tabela 4.10 - Categorização dos defeitos resultantes dos trabalhos mencionados.	140
Tabela 4.11. - Classes dos defeitos constantes nos trabalhos mencionados.	142

LISTA DE SIGLAS E ABREVIATURAS

CCB	<i>Configuration Control Board</i>
CI	<i>Configuration Item</i>
CM	<i>Change Management</i>
CR	<i>Change Request</i>
CMMI	<i>Capability Maturity Model Integration</i>
CMP	<i>Configuration Management Plan</i>
CONOPS	<i>Concept of Operations</i>
ECP	<i>Engineering Change Proposal</i>
ECR	<i>Engineering Change Request</i>
ECSS	<i>European Cooperation for Space Standardization</i>
EN	<i>Engineering Notice</i>
ESA	<i>European Space Agency</i>
IAE	<i>Instituto de Aeronáutica e Espaço</i>
IDM	<i>Information Documentation Management</i>
IEC	<i>International Electrotechnical Commission</i>
IEEE	<i>Institute of Electrical and Electronics Engineers</i>
INCOSE	<i>International Council on Systems Engineering</i>
INPE	<i>Instituto Nacional de Pesquisas Espaciais</i>
IS	<i>Information Systems</i>
ISO	<i>International Standardization Organization</i>
KA	<i>Knowledge Area</i>
KPA	<i>Key Process Area</i>
NASA	<i>National Aeronautics and Space Administration</i>
PHP	<i>PHP Hypertext Preprocessor</i>
RVTM	<i>Requirements Verification and Traceability Matrix</i>
SCN	<i>Specification Change Notice</i>
SEI	<i>Software Engineering Institute</i>
SQL	<i>Structured Query Language</i>
TBD	<i>To Be Defined/Determined</i>
TBR	<i>To Be Reviewed/Revised</i>
TBS	<i>To Be Specified/Supplied</i>
TPM	<i>Technical Performance Measures</i>

TS *Technical Requirements Specification*
WBS *Work Breakdown Structure*

SUMÁRIO

	<u>Pág.</u>
1 INTRODUÇÃO.....	1
1.1. Contexto	1
1.2. Objetivo do trabalho.....	2
1.3. Metodologia	3
1.4. Motivação e justificativa	4
1.5. Organização da pesquisa	8
2 REVISÃO BIBLIOGRÁFICA - CONCEITOS BÁSICOS	9
2.1. Engenharia de Requisitos.....	9
2.1.1. Definições básicas	11
2.1.1.1. Requisito	12
2.1.1.2. Processo	13
2.1.1.3. Diagrama IDF0.....	14
2.1.1.4. Modelo de referência CMMI	15
2.1.1.5. Procedimento	16
2.1.1.6. Atividades.....	16
2.1.1.7. Sistema	16
2.1.1.8. Stakeholder	17
2.1.1.9. Engenharia de requisitos.....	18
2.1.1.10. Rastreabilidade de requisitos	20
2.1.1.11. Baseline	26
2.2. Características de requisitos.....	27
2.2.1. Classificação dos requisitos	30
2.3. Atividades de engenharia de requisitos	31
2.3.1. Desenvolvimento de requisitos.....	32
2.3.1.1. Elicitação de requisitos.....	32
2.3.1.2. Análise	32
2.3.1.3. Especificação	33
2.3.1.4. Validação.....	33
2.3.2. Gerenciamento de requisitos.....	33

2.4. Ferramentas.....	40
2.5. Sistemas aeroespaciais	41
3 ESTUDO DOS MODELOS	43
3.1. Abordagem NASA.....	43
3.2. Abordagem ECSS.....	48
3.2.1. ECSS-E-ST-10C	50
3.2.2. ECSS-M-ST-40C Rev. 1.....	52
3.3. Abordagem INCOSE.....	57
3.4. Abordagem IEEE	67
3.5. Considerações sobre o INPE e o IAE e a comparação de abordagens apresentadas.....	72
3.6. Comparação das abordagens apresentadas	74
4 PROCESSO PROPOSTO	97
4.1. Tratamento de mudanças	104
4.1.1. Análise de solicitação	105
4.1.1.1. Análise preliminar	106
4.1.1.2. Análise de impacto	110
4.1.1.3. Classificação	119
4.1.1.4. Verificação.....	122
4.1.1.5. Validação.....	123
4.1.1.6. Aprovação	126
4.1.2. Alteração contratual	128
4.1.3. Implementação das mudanças solicitadas	129
4.1.4. Geração e manutenção da <i>baseline</i>	130
4.2. Monitoramento	132
4.2.1. Rastreabilidade	133
4.2.2. Auditoria	134
4.2.3. Detecção de defeitos.....	135
4.3. Avaliação	142
4.3.1. Volume	143
4.3.2. Custo.....	144
4.3.3. Prazos	145

4.3.4.	Desempenho	145
4.4.	Geração de conhecimento e gerenciamento de base de conhecimento	146
4.5.	Fluxo para aplicação do processo proposto	157
5	DEMONSTRAÇÃO DA APLICAÇÃO DO PROCESSO PROPOSTO ...	169
5.1.	A análise da solicitação de mudança	170
5.2.	Análise de impacto.....	174
5.3.	Análise de verificação e validação da mudança	180
5.4.	Geração e manutenção de <i>baseline</i>	183
5.5.	Monitoramento	184
5.6.	Avaliação	185
5.7.	O aprimoramento do modelo existente: criação e gerenciamento de <i>baseline</i> e de base de conhecimento.	187
5.8.	Considerações gerais sobre a demonstração do modelo	189
6	CONCLUSÕES.....	191
6.1.	Sugestões para trabalhos futuros	194
	REFERÊNCIAS BIBLIOGRÁFICAS	197
	APÊNDICE A – CÓDIGOS PARA OS FORMULÁRIOS DE DADOS DA MUDANÇA PROPOSTA	201
	APÊNDICE B – CÓDIGOS PARA PÁGINA DE BASE DE CONHECIMENTOS	219
	APÊNDICE C - COMANDOS SQL PARA CRIAR A ESTRUTURA DA BASE DE DADOS PARA AS PÁGINAS DE EXEMPLO.	223

1 INTRODUÇÃO

1.1. Contexto

Quando tratamos o processo de gerenciamento de projetos, a literatura registra, em consenso geral, a importância de se estabelecer um processo formal para o correto levantamento de requisitos, ao longo do processo de desenvolvimento de produtos/serviços. O levantamento adequado das necessidades que nucleiam o projeto, e um posterior tratamento destes requisitos, dentro do seu ciclo de vida, são essenciais para o sucesso do empreendimento.

São inúmeras as referências que relatam o custo crescente, ao longo do processo de desenvolvimento de projetos, devido à existência de erros, em face da incorreta elicitação dos requisitos. Desta forma, é comum depararmos com recomendações para o estabelecimento de método com foco na solução de tal problema.

Não bastasse o desafio de obter o levantamento completo e adequado de requisitos, mesmo com a certeza de que esta fase tenha sido bem resolvida, o problema não é completamente resolvido, pois as fases subsequentes carregam um nível de complexidade, tão, ou mais alto do que o inicial.

Convém ressaltar que são inúmeras as tarefas inerentes ao processo de gerenciamento dos requisitos, dentre as quais podem ser citadas: A identificação correta de *stakeholders*; a tradução dos requisitos de linguagem natural para um padrão técnico formal; a seleção do padrão mais adequado ao contexto do projeto; o estabelecimento de procedimentos para manter a ligação entre os requisitos originais e os requisitos derivados; a seleção de métodos adequados para controle das alterações, que por diversos fatores surgem, naturalmente, ao longo do ciclo de vida; o processo de documentar, adequadamente os requisitos e de estabelecer rotinas para verificação e validação ao longo do processo. Todas estas tarefas dão uma ideia da

amplitude do problema, que se apresenta apropriadamente resumido pela seguinte citação:

Cada projeto tem êxito ou falha pela qualidade de seus requisitos. Eles estabelecem o escopo de todo o trabalho subsequente e dizem à equipe de projeto o que os usuários querem. Sem bons requisitos, os projetos falham, atrasam, ultrapassam o orçamento ou produzem sistemas que jamais serão usados (ALEXANDER, 2002).

Embora vários métodos e ferramentas tenham sido desenvolvidos nos últimos anos na tentativa de resolver tais questões, percebe-se, ao vivenciar a prática do dia a dia, que os problemas oriundos da má gestão de requisitos continuam a ser detectados. Cronogramas em desacordo, orçamentos “extrapolados” e produtos que não atendem ao solicitado sugerem que a questão continua em aberto, o que já foi constatado por diversos autores e que pode ser ilustrado com a citação, a seguir:

Apesar de mais de trinta anos de pesquisa no processo pelos quais os requisitos de sistema são identificados, enunciados, validados e gerenciados, profissionais de IS e gerentes organizacionais continuam a lutar com os desafios inerentes às mais básicas questões de projeto: O que é isto que nós queremos criar? Que características e funcionalidades deve possuir nosso artefato projetado? Com quais restrições deve nosso sistema estar em conformidade? No seu âmago estas considerações, fundamentais, são questões sobre requisitos e Engenharia de Requisitos e elas continuam a desafiar a resolução simples (HANSEN, 2011).

1.2. Objetivo do trabalho

O objetivo desta pesquisa é propor e demonstrar a aplicação de um modelo com processos para Gerenciamento de Requisitos de sistemas aeroespaciais, com aplicabilidade geral à Engenharia Aeroespacial, abrangendo o controle

das alterações ocorridas ao longo do desenvolvimento de um sistema/produto. Como objetivos específicos têm-se:

- a) propor um processo de Gerenciamento de Requisitos, tomando por base modelos existentes e aplicáveis ao contexto aeroespacial;
- b) demonstrar a aplicação do modelo proposto no contexto espacial do INPE, utilizando ferramentas disponíveis no ambiente.

Para atingir tais objetivos foram obedecidas as etapas descritas na Metodologia.

1.3. Metodologia

De forma a atingir o objetivo proposto neste trabalho foram seguidos os seguintes passos:

- I- Rever a literatura sobre Gerenciamento de Requisitos e analisar, comparativamente, pelo menos duas, dentre as abordagens existentes, que servirão de base para compor um modelo de Gerenciamento de Requisitos, que seja adaptável e aplicável aos sistemas Aeroespaciais, em particular aos sistemas direcionados ao INPE.
- II- Discutir os principais pontos críticos inerentes ao processo de gerenciamento de requisitos proposto, bem como as principais técnicas utilizadas atualmente para análise e eliminação de conflitos.
- III- Selecionar uma dentre as ferramentas atualmente disponíveis de gestão de requisitos e utilizá-la para a aplicação do modelo proposto.
- IV- Realizar uma aplicação piloto no domínio de Engenharia Aeroespacial do INPE.

1.4. **Motivação e justificativa**

A motivação para elaborar esta proposta de pesquisa surgiu com a experiência de enfrentamento das dificuldades de controle de alterações nos requisitos ao longo do desenvolvimento de ensaios aerodinâmicos e, em outras oportunidades, em desenvolvimento de sistemas intensivos de software usados na Engenharia Aeroespacial, em face da inadequação do modelo de gerenciamento de requisitos então utilizado.

Os requisitos de um sistema exigem controles rigorosos para garantir a segurança no desenvolvimento de produtos no setor aeroespacial e a qualidade das informações resultantes. É preciso documentá-los antes da realização do evento para garantir a presença de todas as condições críticas, ou não, exigidas.

É comum ocorrerem alterações dos requisitos ao longo do processo. Tais alterações precisam ser documentadas e devem refletir as evoluções, ou modificações, no contexto em que se inserem e, como em qualquer solicitação de serviços relacionados a sistemas, é comum que surjam novos requisitos, ou solicitações de exclusão por parte dos usuários. Dentre outras, elencamos, a seguir, algumas razões que podem provocar tais solicitações:

- a) ocorrências de conflitos entre os diversos requisitos. Tais conflitos devem ser resolvidos o mais breve possível. Há que se buscar a consistência e a completude dos requisitos para garantir a qualidade dos resultados;
- b) ocorrências de solicitações de alterações devido à nova legislação, regulamentação, ou alterações tecnológicas, que podem resultar em acréscimos ou exclusões de requisitos.

Outra preocupação observada é que durante o desenvolvimento são geradas versões diferenciadas do documento que norteia o processo, propiciando

revisões sucessivas de novos processos de Engenharia de Requisitos para os requisitos adicionais, modificados, ou excluídos.

Sommerville e Sawyer (1997) citam que poucas organizações têm um processo de Engenharia de Requisitos padronizado e explicitamente bem definido. Em geral, simplesmente definem o resultado do processo, ou seja, o documento de requisitos. Na maioria são os indivíduos envolvidos, responsáveis pela tarefa, que decidem o que, como e quando fazer (SOMERVILLE; SAWYER, 1997).

Isto, em certa medida, ocorre também no INPE, conforme mencionado por Almeida (2011) quando afirma que o uso de ferramentas para automação do gerenciamento de requisitos fica prejudicado pela inexistência de processos adequados de preparação dos requisitos, impossibilitando, desta forma, o gerenciamento dos mesmos. Almeida afirma, ainda, que:

A introdução de processos adequados de preparação de requisitos, do gerenciamento de requisitos e de ferramentas de automatização pode ajudar a reduzir riscos e custos e fazer melhor uso da escassa mão-de-obra disponível (ALMEIDA, 2011, p. 2).

Apesar disso, a questão de gerenciamento de requisitos parece não receber ainda a devida atenção, pois pesquisas recentes de opinião apontam para a escassez de tempo investido em processos de gerenciamento. Em um relatório recente do PMI (2014) é citada a carência em estudos sobre a questão, conforme a seguir:

Apesar da importância óbvia dos requisitos para o valor do projeto e resultados nos negócios, existe uma escassez de pesquisas significativas em como as organizações percebem e abordam este componente crítico de projetos e programas (PMI, 2014).

Este mesmo relatório conclui que as organizações podem mitigar os efeitos negativos de um gerenciamento fraco de requisitos e algumas das medidas referem-se à padronização, formalização e o amadurecimento de processos de

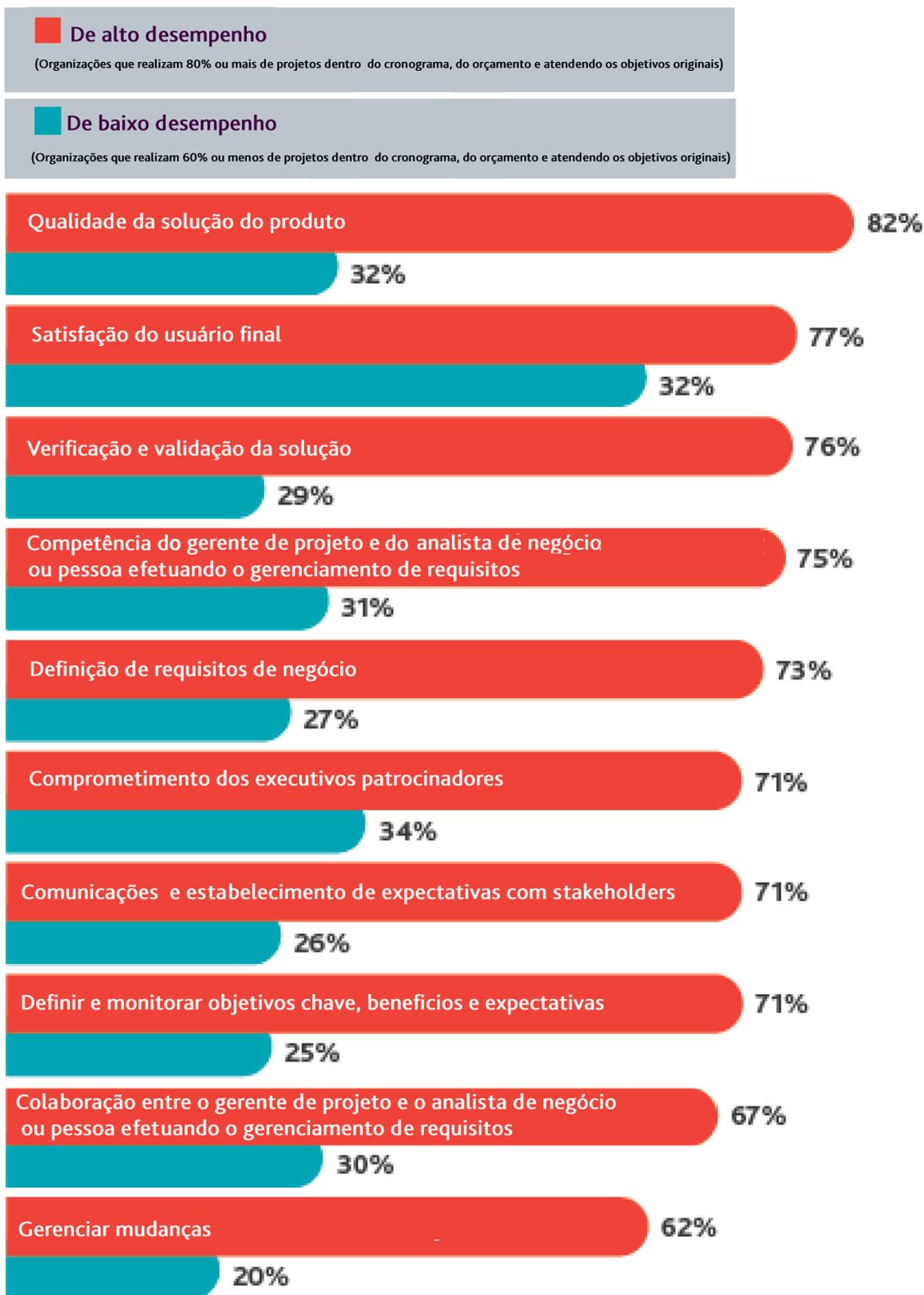
gerenciamento de requisitos, além da aplicação consistente de boas práticas de gerenciamento de requisitos e de garantir que a organização, em toda a sua extensão, valorize o gerenciamento de requisitos como uma competência crítica para projetos e programas, colocando o devido esforço nesta atividade.

Segundo o PMI

As organizações de alto desempenho reconhecem a importância dos processos e práticas de gerenciamento de requisitos para seus projetos, pois como mostra nosso estudo, as de alto desempenho estão significativamente mais propensas a usar um processo formal para validação de requisitos de projetos quando comparadas com suas concorrentes de baixo desempenho (PMI, 2014).

O relatório mostra que quando comparadas com as de baixo desempenho, as organizações de alto desempenho informam sair-se significativamente melhor em cada uma das 10 práticas importantes de gerenciamento de requisitos, o que denota claramente níveis mais altos de proficiência entre as práticas e processos de gerenciamento de requisitos para as organizações de alto desempenho, como ilustra a Figura 1.1.

Figura 1.1 - Resultados das 10 principais práticas de gerenciamento de requisitos em organizações de alto desempenho.



Fonte: Adaptada de PMI (2014).

1.5. Organização da pesquisa

O Capítulo 1 apresenta uma breve introdução. Inclui o contexto, objetivos do trabalho, a metodologia, a motivação, a justificativa e a organização da pesquisa.

O Capítulo 2 descreve, de forma resumida, os conceitos básicos e uma revisão bibliográfica e contém o conhecimento necessário para suporte ao desenvolvimento do trabalho proposto.

O Capítulo 3 apresenta uma comparação de quatro abordagens utilizadas atualmente no campo da Engenharia de Requisitos.

O Capítulo 4 apresenta o modelo proposto para gestão de requisitos, oriundo da comparação dos quatro principais abordagens comparadas e descritas no capítulo 3.

O Capítulo 5 apresenta os resultados da aplicação piloto no domínio da Engenharia Aeroespacial do INPE;

Finalmente, no Capítulo 6 é apresentada a discussão dos resultados obtidos e as conclusões obtidas com a experiência empreendida, incluindo recomendações e sugestões para trabalhos futuros.

2 REVISÃO BIBLIOGRÁFICA - CONCEITOS BÁSICOS

A fundamentação teórica do modelo proposto por esta pesquisa, que é a proposta e aplicação de um método para gerenciamento de requisitos de sistemas aeroespaciais, está sustentada pelos seguintes pilares:

- a) conceitos básicos da Engenharia de Requisitos;
- b) a importância do investimento de tempo e da utilização dos processos da Engenharia de Requisitos para o sucesso do desenvolvimento de sistemas aeroespaciais;
- c) quatro abordagens, e ferramentas, ora utilizadas, para o gerenciamento de requisitos.

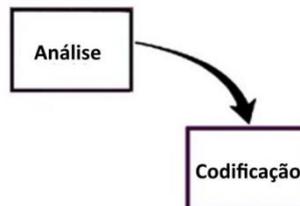
2.1. Engenharia de Requisitos

A Engenharia de Requisitos teve sua origem a partir do esforço de desenvolvimento de software, em resposta à crescente complexidade para a gestão da criação de programas de computador, diante do fenômeno denominado “software *crisis*”, ocorrido ao final dos anos 60. Tal crise resultou da atuação de diversos fatores e evidenciou a necessidade de tratamento mais cuidadoso no início do processo de desenvolvimento de software. É possível inferir que, até então, quase tudo era feito sem, ou com pouco, planejamento e sem metodologia definida. Iniciava-se a codificação de um programa a partir do zero, praticamente; o produto final começava a ser produzido sem que fosse utilizado um processo, ou metodologia formal, que o guiasse. Assim, era possível iniciar a produção sem que se soubesse, ao certo, o que se pretendia, gerando retrabalho e custos altos para a correção de erros sobre o que já estava pronto.

Em 1970, Royce (1970) identificou a necessidade de novas metodologias para auxiliar na tarefa de criação de programas e atribuiu ao termo “requisito”, caráter fundamental para a execução do processo de desenvolvimento, apresentando modelos que podem ser visualizados através das Figuras 2.1 e

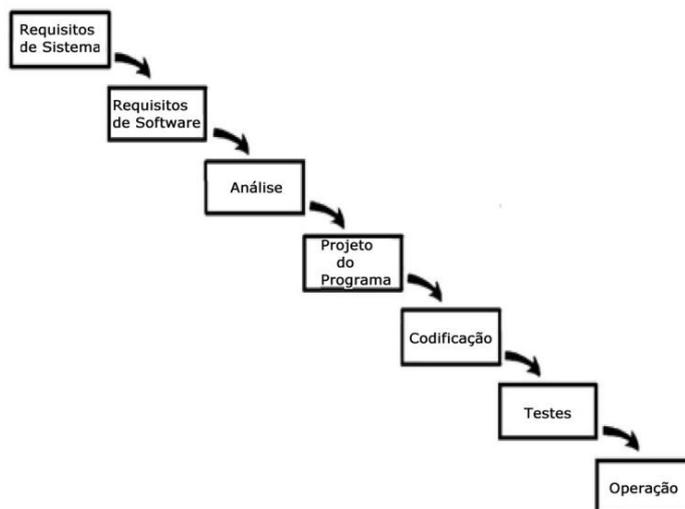
2.2, que mostram a abordagem do desenvolvimento de pequenos programas e de grandes programas, respectivamente. Este último pode ser visto como precursor do que viria a ser conhecido como modelo “*waterfall*” para desenvolvimento (ROYCE, 1970).

Figura 2.1 - Desenvolvimento de pequenos programas.



Fonte: Adaptado de Royce (1970).

Figura 2.2 - Desenvolvimento de grandes programas.



Fonte: Adaptado de Royce (1970).

Diversos trabalhos posteriores abordaram problemas relacionados aos requisitos, dando prosseguimento à evolução da Engenharia de Requisitos.

Somente para citar alguns estudiosos sobre o assunto, Meyer (1985), abordou o uso de linguagem natural para especificação de requisitos de forma crítica e apresentou uma proposta alternativa, baseada em formalismo matemático (MEYER, 1985).

Gotel e Finkelstein (1994), em um trabalho que se tornou referência no assunto, apresentaram baseados em estudos empíricos, os desafios para o efetivo tratamento da rastreabilidade de requisitos, ressaltando o porquê de a rastreabilidade ser fundamental para processos de desenvolvimento de software (GOTEL; FINKELSTEIN, 1994).

Nuseibeh e Easterbrook (2000) apresentaram uma visão geral da Engenharia de Requisitos aplicada a software, discorrendo sobre as atividades que são o núcleo da Engenharia de requisitos: elicitacão, modelagem e análise de requisitos, comunicacão, negociaçao e a evoluçao de requisitos. Ressaltam que sistemas de software bem sucedidos sempre evoluem à medida que o ambiente em que operam e os requisitos dos *stakeholders* mudam e, desta forma, afirmam que o gerenciamento de mudançao é uma atividade fundamental para a RE. Apresentam a Engenharia de requisitos como uma atividade multidisciplinar, empregando uma variedade de técnicas e ferramentas nos diversos estágios de desenvolvimento e em diferentes tipos de domínios de aplicaçoes (NUSEIBEH; EASTERBROOK, 2000).

Kauppinen et al. (2004) trataram dos desafios e fatores de sucesso nos processos de Engenharia de Requisitos aplicados às organizaçoes.

A maioria dos autores mencionados focalizaram seus estudos nos problemas originados no processo de gerenciamento de requisitos. A seguir, serão apresentadas as definiçoes e conceitos relatados pela literatura sobre os problemas relacionados com os requisitos e, em particular, com seu gerenciamento.

2.1.1. Definiçoes básicas

A maioria da literatura consultada mostrou que a conceituacão, embora não possa ser classificada como incipiente, ainda carece de consenso e padronizaçao. Grande parte dos termos encontrados percebe-se, nitidamente, serem originários da área de software, o que provoca viés de interpretaçao, uma vez que nem sempre tais termos são adequados para a aplicaçao geral.

Desta forma, buscou-se a apresentação de um conjunto de definições mais comuns encontradas na literatura e aplicadas em diversos domínios.

2.1.1.1. Requisito

Convém, inicialmente, apresentar algumas definições para o termo requisito, considerando ser este o principal conceito no cerne desta pesquisa.

Uma declaração que identifica um produto, ou processo operacional ou funcional, ou restrições ou características de projeto, não ambígua, possível de ser testada ou medida, e necessária para a aceitabilidade do processo ou produto (pelos clientes ou diretrizes da garantia da qualidade interna) (IEEE, 2005, p. 9).

Um requisito é um atributo necessário em um sistema, uma declaração que identifica uma capacidade, característica ou fator de qualidade de um sistema, de forma que este (o sistema) tenha valor e utilidade para o cliente ou usuário (YOUNG, 2004, p. 222).

Uma necessidade, desejo, capacidade ou demanda, estabelecidos em acordo, para pessoal, equipamento, instalações ou outros recursos ou serviços, em quantidades específicas para períodos de tempo determinados, ou em um momento específico no tempo, expressos como uma declaração mandatória. A Forma aceitável para uma declaração de requisito é que esta seja individualmente clara, correta, possível de se obter, sem ambiguidade, e pode ser validada ao nível da estrutura do sistema em que foi declarada. Tratadas em pares de declarações de requisitos, ou em conjunto, coletivamente, os requisitos não podem ser redundantes, e devem estar adequadamente relacionados com os termos utilizados e não devem estar em conflito entre si. (NASA, 2007, p. 274).

Uma declaração que identifica uma característica ou restrição de um sistema, processo ou produto, que é não ambígua, clara, única, consistente, individual e possível de ser verificada e que é considerada necessária para a aceitabilidade pelo *stakeholder* (INCOSE, 2011, p. 364).

Requisitos são a base para cada projeto, definindo o que os *stakeholders* – usuários, clientes, fornecedores, desenvolvedores, empresa – necessitam de um potencial novo sistema e também o que o sistema deve fazer de maneira a satisfazer essa necessidade (HULL et al., 2011, p.6).

As definições mencionadas acima são similares e convergem para a aceitação, implícita, de que um requisito é uma descrição de uma necessidade de *stakeholder* (interessado) e que, mediante um processo decisório pode, ou não, ser atendida por uma característica de um produto/serviço.

Importante notar a relevância dos estudos sobre requisitos a partir da definição de Engenharia de Sistemas proposta pela ECSS.

Engenharia de Sistemas é definida como uma abordagem interdisciplinar governando o esforço técnico total para transformar requisitos em uma solução de sistema (EUROPEAN SPACE AGENCY (ESA), 2009, p. 12).

Esta definição não deixa dúvidas quanto ao fato de que o resultado final esperado, ou seja, o atendimento a uma necessidade só será atingido pela correta definição e gerenciamento dos requisitos inicialmente obtidos.

A necessidade de apresentar as quatro próximas definições vem do fato delas aparecerem, com frequência, em textos sobre Engenharia de sistemas.

2.1.1.2. Processo

Conjunto de atividades interagindo ou inter-relacionadas, que transformam entradas em saídas (EUROPEAN SPACE AGENCY (ESA), 2012).

Conjunto de atividades utilizado para converter entradas em saídas desejadas, de forma a gerar resultados esperados e satisfazer um propósito (NASA, 2007).

2.1.1.3. Diagrama IDFO

IDEFO é um subconjunto do SADT (Structured Analysis and Design Technique), colocado em domínio público por Softech Inc. por requisição do Departamento de Defesa dos Estados Unidos da América.

É uma ferramenta de modelagem onde as letras de IDEF originam-se de ICOM DEFinitions; onde ICOM são as primeiras letras de Inputs, Controls, Outputs, Mechanisms (Entradas, Controles, Saídas e Mecanismos).

O IDEFO fornece um método formal para representação de processos ou sistemas, onde o diagrama básico é muito simples e é composto de um retângulo representando um processo ou atividade e quatro setas no contorno, uma em cada face, cujas iniciais dão origem ao termo ICOM e significam:

- a) Inputs - são a matéria prima a ser transformada durante a atividade;
- b) controls - São os controles que influenciam ou direcionam os trabalhos do processo, por exemplo: normas de segurança, requisitos de usuário, planos;
- c) mechanisms - São os mecanismos que fazem o processo funcionar, por exemplo: profissionais, ferramentas, máquinas.
- d) outputs - As saídas que são o resultado da atividade e são transmitidos para a outros processos.

A Figura 2.3, a seguir, mostra a representação gráfica do IDEFO.

Figura 2.3 - Diagrama IDEF0



Fonte: adaptada pelo autor de SYQUE (2017).

2.1.1.4. Modelo de referência CMMI

O modelo CMMI (Capability Maturity Model Integration) desenvolvido pelo SEI (Software Engineering Institute) fornece diretrizes para avaliação e aprimoramento de processos de desenvolvimento de software através da definição de 5 níveis de maturidade para o processo de desenvolvimento:

- a) inicial;
- b) repetível;
- c) definido;
- d) gerenciado;
- e) otimizado.

Para cada um destes níveis há um conjunto de áreas-chave de processo (KPA) a ser implantado.

A gerência de requisitos é uma das áreas chave de processos para o atingimento do nível dois de maturidade ou o nível "REPETÍVEL".

Neste nível existem processos básicos de gerenciamento de projetos e os aspectos de custo, de prazo e de funcionalidade são monitorados.

Vale ressaltar que o CMMI estabelece as premissas para o cumprimento das metas sem detalhar os passos para implantação e recomenda que os recursos humanos que tratarão os processos deverão ser detentores dos conhecimentos relativos às diversas áreas técnicas inerentes a processos de engenharia, com foco no gerenciamento e desenvolvimento de requisitos, nas soluções técnicas, na integração de produto, na verificação e na validação, bem como nos conhecimentos específicos sobre métodos, ferramentas e procedimentos.

2.1.1.5. Procedimento

Forma documentada para realizar uma atividade ou um processo, de uma maneira controlada (EUROPEAN SPACE AGENCY (ESA), 2012).

2.1.1.6. Atividades

Conjunto de tarefas que descrevem o esforço técnico para realizar um processo e auxiliar a gerar os resultados esperados (NASA, 2007).

Conjunto de tarefas coesas de um processo (INCOSE, 2011).

2.1.1.7. Sistema

(1) A combinação de elementos que funcionam juntos para produzir uma capacidade visando atender uma necessidade. Os elementos incluem todo *hardware*, software, equipamento, instalações, pessoal, processos e procedimentos necessários para esse propósito. (2) O produto final (que executa funções operacionais) e produtos facilitadores (que fornecem serviços de suporte aos produtos finais operacionais) que constituem um sistema (NASA, 2007).

Conjunto de funções, interagindo ou inter-relacionadas, constituídas para alcançar um objetivo especificado (EUROPEAN SPACE AGENCY (ESA), 2012).

Um conjunto integrado de elementos, subsistemas, ou montagens que realizam um determinado objetivo. Estes elementos incluem produtos (*hardware*, *software*, *firmware*), processos, pessoas, instalações, serviços e outros elementos de suporte (INCOSE, 2011).

2.1.1.8. Stakeholder

O termo *stakeholder* é relativamente novo, o que se nota pelo fato de não estar presente nas normas:

- a) IEEE Std 610.12-1990 - *IEEE Standard Glossary of Software Engineering Terminology*;
- b) IEEE Std 1233, 1998 Edition- *IEEE Guide for Developing System Requirements Specifications*;
- c) IEEE Std 830-1998-IEEE *Recommended Practice for Software Requirements Specifications*;
- d) ECSS-E-10 PART1B (18 NOV 2004) *System Engineering—Part 1: Requirements and Process*.

Constata-se que em 2005 o termo *stakeholder* foi utilizado na edição da Norma IEEE Std 1220-2005 - *Systems Engineering-Application and Management of the Systems Engineering Process* como:

Uma parte que tenha direito, participação ou reivindicação sobre um sistema, ou sobre a posse das características, que satisfaçam as suas necessidades e expectativas. (IEEE, 2005)

A citação, a seguir, propõe uma definição abrangente, e precisa, sobre o termo *stakeholder*.

Qualquer pessoa, grupo de pessoas, entidade ou órgão que é afetado ou tem interesses (positivos ou negativos), influencia ou impõe requisitos a um sistema ou produto. São exemplos de *stakeholders*: quem desenvolve, opera, usa e repara o sistema, quem paga pelo sistema e por suas despesas; agências reguladoras; seus proprietários, vítimas, fornecedores, patrocinadores, defensores, competidores. Alguns autores distinguem *stakeholders* de produto (que afetam somente o produto) e de processo (que afetam os processos de obtenção do produto) (ALMEIDA, 2011, p. 8).

Nota-se que o escopo do termo se estendeu substancialmente, indo além do restrito conceito de usuário, ou cliente, para uma definição mais ampla, atingindo a identificação de qualquer agente impactado, positiva ou negativamente, pelo projeto, podendo ser o governo, os usuários, os agentes atuantes diretamente no projeto, outros sistemas, bem como a sociedade como um todo. Esta visão torna este ponto crucial na Engenharia de Requisitos, visto que a ausência de um *stakeholder*, por falha de identificação, pode impactar, em diversos graus, todos os processos posteriores.

2.1.1.9. Engenharia de requisitos

Este é um conceito em que se observa a existência de discordância entre os estudiosos do assunto, Wiegers e Beaty (2013) comentam que a confusão sobre a terminologia estende-se ao ponto de como denominar toda a disciplina e expressa que a sua preferência é por chamar todo o domínio de Engenharia de Requisitos, enquanto outros autores preferem usar o termo Gerenciamento de Requisitos. Wiegers e Beaty (2013) explicitam, ainda, sua visão de que a Engenharia de Requisitos deve ser dividida em dois tópicos: Desenvolvimento de Requisitos e Gerenciamento de Requisitos (WIEGERS; BEATY, 2013).

Cabe ressaltar que tal visão será adotada ao longo deste trabalho.

Sommerville e Sawyer (1997) definem que um processo de Engenharia de Requisitos é um conjunto estruturado de atividades para desenvolver, validar e

manter um documento de requisitos de um sistema e que o processo completo deve incluir as atividades que serão executadas, a estrutura, o cronograma destas atividades, os respectivos responsáveis, as entradas e saídas para cada atividade, e as ferramentas que serão utilizadas para a Engenharia de Requisitos (SOMMERVILLE; SAWYER, 1997).

Adotando uma definição bem parecida com a de Sommerville e Sawyer, Hull et al. (2011) afirmam que a Engenharia de Requisitos é o conjunto de todas as atividades do ciclo de vida do sistema/produto dedicadas à identificação de requisitos de usuário, análise desses requisitos para a derivação de requisitos adicionais, a respectiva documentação na forma específica e a validação do documento de requisitos comparado às necessidades dos usuários, assim como todos os processos que dão suporte a tais atividades (HULL et al., 2011, p.8)..

O termo aparece, ainda, em normas internacionais conforme apresentado pelas seguintes citações:

Função interdisciplinar que arbitra entre os domínios do fornecedor e do adquirente para estabelecer e manter os requisitos a serem atendidos pelo sistema, software ou serviço de interesse NOTA: Engenharia de Requisitos preocupa-se com a descoberta, elicitacão, desenvolvimento, análise, determinacão de métodos de verificacão, validacão, comunicacão, documentacão e gerenciamento de requisitos (IEEE, 2011, p. 6).

Engenharia de Requisitos é um termo relativamente novo, que foi inventado para cobrir todas as atividades envolvidas com a descoberta, documentacão e manutençao de um conjunto de requisitos para um sistema baseado em computador. O uso do termo “Engenharia” implica que técnicas sistemáticas e repetíveis devem ser usadas para garantir que os requisitos do sistema sejam completos, consistentes, relevantes, etc. O termo “engenharia de Requisitos” originou-se no conhecimento

consolidado da Engenharia de Sistemas; (IEEE COMPUTER SOCIETY - ACM COMMITTEE, 2000, p. 2).

2.1.1.10. Rastreabilidade de requisitos

A rastreabilidade de requisitos é um termo que surge com diversas definições, conforme pode ser visto nas citações, a seguir:

Uma associação discernível entre duas ou mais entidades lógicas como requisitos, elementos de sistema, verificações, ou tarefas (NASA, 2007, p. 132).

Atributo do requisito que conecta cada requisito individual ao seu requisito de nível mais alto, dentro do conjunto de requisitos (EUROPEAN SPACE AGENCY (ESA), 2009, p. 12).

(1) O grau ao qual um relacionamento pode ser estabelecido entre dois ou mais produtos do processo de desenvolvimento, especialmente produtos tendo um relacionamento predecessor-sucessor ou mestre-subordinado para com o outro, isto é, o grau no qual os requisitos e o projeto de um dado elemento do Sistema coincidem (IEEE, 1990, p. 78).

(2) O grau ao qual cada elemento em um produto de desenvolvimento de software estabelece sua razão de existência; por exemplo, o grau ao qual cada elemento em um gráfico de bolhas referencia o requisito que ele satisfaz (IEEE, 1990, p. 78).

A habilidade de determinar a necessidade do cliente a uma conectividade ou relacionamento de requisito, ou o relacionamento de um requisito evidente para um descendente e vice versa. A habilidade de rastrear um requisito durante todo o processo de desenvolvimento do sistema. Absolutamente crítico para todos os sistemas (YOUNG, 2004, p. 223).

Segundo Hull et al. (2011), a rastreabilidade pode contribuir com os seguintes benefícios:

- a) maior confiança no atendimento dos objetivos;
- b) habilidade para avaliar o impacto de uma mudança;
- c) habilidade de rastrear o progresso;
- d) habilidade para balancear custo/benefício.

Trata-se de processo fundamental para a análise de impacto. Se um requisito for alterado, é imprescindível que seja possível a identificação da extensão do impacto da alteração sobre os demais requisitos, de forma que os efeitos, decorrentes, possam ser revisados e mensurados. Além disso, a rastreabilidade possibilita a validação dos requisitos, pois permite o mapeamento do relacionamento existente entre as necessidades descritas e as funcionalidades entregues.

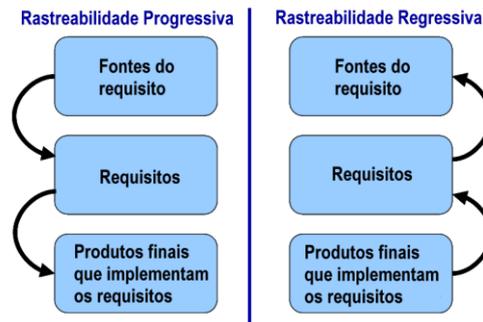
A rastreabilidade dos requisitos apresenta-se como uma das tarefas mais difíceis dentro da Engenharia de Requisitos, tanto sob o ponto de vista de representação dos relacionamentos, pois normalmente é uma relação de vários (requisitos) para vários, tornando virtualmente impossível a visualização global, quanto sob o ponto de vista de manutenção de estruturas de dados relacionados e suas ligações que formam uma malha complexa.

Hull et al. (2011) citam, ainda, que a rastreabilidade permite outros tipos de análise, além da análise de impacto, como: 1) a análise de derivação, que trabalha em sentido inverso à análise de impacto pois, partindo de elementos de mais baixo nível determina que requisito de alto nível lhe deu origem; assim, elementos de baixo nível que não apresentam uma origem, potencialmente, adicionam custo sem benefício; e 2) a análise de cobertura, que pode ser usada para determinar se todos os requisitos rastreiam para níveis mais baixos até os testes; e, na ausência de tal trilha, há uma indicação de que o requisito não será atendido ou testado (HULL et al., 2011).

Conforme Westfall (2007) a rastreabilidade, então, é essencial para o acompanhamento do relacionamento entre um requisito único, no nível de

produto, até a sua fonte e também o relacionamento de cada requisito único, no nível de produto, até o produto final ao qual o requisito foi alocado e as boas práticas de rastreabilidade permitem a rastreabilidade bidirecional, que significa que a cadeia de elos pode ser percorrida de forma progressiva e regressiva, conforme mostrado na figura 2.4, a seguir:

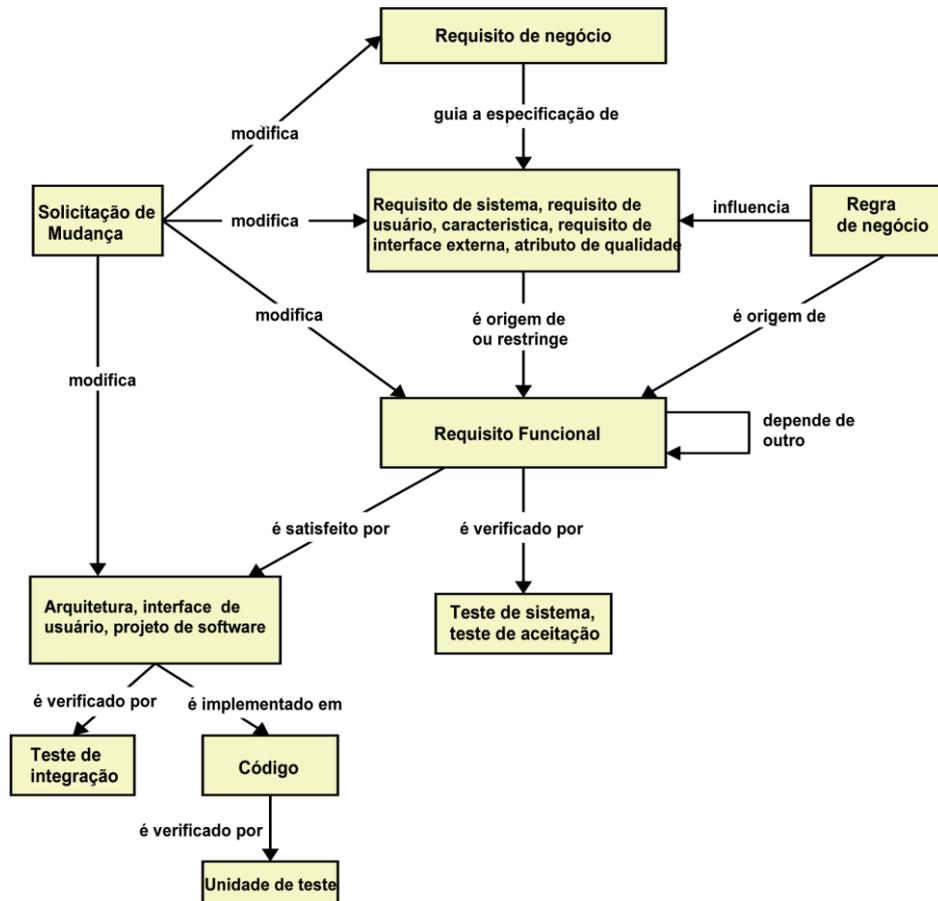
Figura 2.4 - Rastreabilidade bidirecional.



Fonte: Adaptada pelo autor a partir de Westfall (2007).

Diversos tipos de relacionamentos de rastreabilidade são apresentados por Wieggers e Beaty (2013) e mostrados através da Figura 2.5, a seguir:

Figura 2.5 - Alguns possíveis elos de rastreamento de requisitos.



Fonte: Adaptada pelo autor a partir de Wiegers e Beaty (2013).

Davis (1990) considerou que o processo de rastreamento de requisitos pode ser classificado em quatro tipos:

- a) a partir dos requisitos: associa os requisitos ao projeto e componentes de efetivação;
- b) de volta aos requisitos: associa o projeto e os componentes de implementação aos requisitos. A aderência do sistema aos requisitos deve ser verificada e devem-se evitar projetos para os quais não se exijam requisitos;
- c) em direção a novos requisitos: verifica e associa documentação existente que preceda os documentos de requisitos, aos requisitos relevantes. Isto porque qualquer mudança das necessidades dos

stakeholders ou de natureza técnica pode exigir reavaliação dos requisitos relevantes;

- d) de volta a partir dos requisitos: associa os requisitos às suas fontes em outros documentos, bem como às estruturas que contribuíram para originá-los, necessários para sua validação.

Kotonya e Sommerville (1998) acrescentam aos estudos de Davis (1990) que tais tipos podem ser instanciados por *links* entre informações específicas, no documento de requisitos e em outros documentos do sistema, tais como:

- a) rastreamento das fontes de requisitos: *links* que conectam o requisito aos documentos que os descrevem;
- b) rastreamento da razão dos requisitos: *links* que conectam o requisito à sua justificativa de existência;
- c) rastreamento requisitos-requisitos – *links* que conectam o requisito analisado com outros requisitos que dele dependem, ou sofrem seu efeito, estabelecendo um esclarecimento de dependência em duas direções (dependentes e dependentes-de);
- d) rastreamento requisitos-arquitetura: *links* que conectam o requisito com os componentes do ambiente no qual será operacionalizado;
- e) rastreamento requisitos-projeto: *links* que conectam o requisito com todos os elementos necessários para sua implementação;
- f) rastreamento requisitos-interface: *links* que conectam o requisito com o ambiente externo que irá interagir com o processo de sua implementação.

Segundo Genvigir (2009) o processo de rastreamento começa com a elicitaco, permeando as demais fases do projeto e culminando com a fase de manuteno, tornando-se desta forma um dos aspectos mais importantes do processo de Gerenciamento de Requisitos (GENVIGIR, 2009).

A rastreabilidade fornece informações que permitem a verificabilidade, a identificação de atributos necessários ao sistema, bem como de requisitos desnecessários, além de análise de impactos, reusabilidade, definição de responsabilidades e manutenção (POHL; RUPP, 2011).

Um requisito pode ser abordado de forma regressiva, ou seja, partindo de determinado *status* para sua origem, ou progressiva, partindo da origem e prosseguindo ao longo dos refinamentos. O processo de rastreamento deve contemplar pelo menos uma das abordagens mencionadas.

A classificação do processo de rastreabilidade dos requisitos, sob três aspectos, foram apresentadas nos estudos de Pohl e Rupp (2011).

- a) a rastreabilidade pré-especificação de requisitos (Pré-RS): contempla o relacionamento de rastreabilidade entre os requisitos e objetos que formam a base dos requisitos, por exemplo, a fonte (artefatos prévios);
- b) a rastreabilidade pós-especificação de requisitos (Pós-RS): contempla as informações de rastreabilidade entre requisitos e objetos originários de atividades subsequentes de desenvolvimento;
- c) a rastreabilidade entre requisitos: contempla o mapeamento de interdependências entre os requisitos, tal como a informação de que um requisito refina, generaliza ou substitui outro.

Pohl e Rupp (2011) defenderam, ainda, que a representação dos requisitos pode ser realizada de várias formas, tais como:

- a) referências textuais e *hyperlinks*: realizada pela anotação do objeto-alvo como uma referência textual no requisito (objeto inicial), ou por um *hyperlink* entre o objeto inicial e o alvo;
- b) matrizes de rastreabilidade: representam o cruzamento das informações referentes à rastreabilidade dos requisitos de um sistema;

- c) grafos de rastreabilidade: corresponde a um grafo em que os nós representam os objetos, e as linhas representam os relacionamentos entre os objetos.

A representação do rastreamento permite obter e gerir informações de objetos envolvidos no processo de desenvolvimento do sistema que podem contemplar informações dos *stakeholders* e protocolos utilizados para as entrevistas no momento da elicitação de requisitos (POHL; RUPP, 2011).

Kotonya e Sommerville (1998) sugerem o uso de tabelas de rastreabilidade para mapear os relacionamentos entre requisitos, ou entre requisitos e componentes de projetos. As tabelas, em geral, possuem dupla entrada, sendo que os requisitos são listados ao longo dos eixos horizontais e verticais, permitindo que sejam marcados nas células da tabela se existe relacionamento entre eles. Desta forma as tabelas de rastreamento que mostram as dependências devem ser definidas como uma matriz quadrada, rotulando-se linhas e colunas com as denominações dos requisitos sob análise, conforme modelo apresentado na Tabela 2.1, a seguir.

Tabela 2.1 - Tabela de rastreabilidade entre requisitos.

	Requisito 01	Requisito 02	...	Requisito n
Requisito 01		X	X	
Requisito 02	X		X	X
...				
Requisito n	X		X	X

Fonte: Adaptada pelo autor a partir de Kotonya e Sommerville (1998).

2.1.1.11. Baseline

Um ponto fundamental para o desenvolvimento deste trabalho consiste na dificuldade para o gerenciamento da *baseline* de requisitos, devido à sua complexidade, o volume de alterações e os impactos destas ao longo do ciclo

de desenvolvimento. Assim, é importante o entendimento do termo, o qual é apresentado, a seguir, demonstrando tais dificuldades observadas, em geral, pelos estudiosos sobre o assunto:

Estados aprovados de requisitos e do projeto de um produto, em um marco fundamental, que servem como referência para atividades durante todo o ciclo de vida do produto (EUROPEAN SPACE AGENCY (ESA), 2012, p. 18).

Uma *baseline* de requisitos é um conjunto de requisitos que foram revistos e acordados, e serve como base para desenvolvimento futuro (WIEGERS; BEATY, 2013, P.39).

Um conjunto acordado de requisitos, projetos ou documentos que terá as mudanças controladas através de aprovação formal, e de um processo de monitoramento (NASA, 2007, p. 267).

Uma *baseline* é uma cópia congelada de um módulo. São, tipicamente, criadas em estágios significativos do projeto, isto é, um conjunto de requisitos é normalmente transformado em *baseline* imediatamente antes de uma revisão e, então, imediatamente depois que as mudanças resultantes da revisão tenham sido incorporadas (HULL et al., 2011).

2.2. Características de requisitos

Partindo da premissa de que os requisitos são a matéria prima básica, a partir da qual um produto/serviço será derivado/construído, é fundamental, e mandatório, que tais requisitos possuam características que garantam que o resultado, ao serem considerados no desenvolvimento do projeto, seja sempre o mesmo, independente dos métodos, e/ou ferramentas, aplicadas e/ou da atuação do grupo de projeto.

A literatura apresenta inúmeras características de requisitos quando eles são considerados individualmente. Os principais são sumarizados através da

Tabela 2.2, apresentada, a seguir (WIEGERS; BEATY, 2013, p. 203-206). (YOUNG, 2004, p. 8); (HULL et al., 2011,p.89).

Tabela 2.2 - Características individuais de requisitos.

Característica	Descrição
Atômico	Cada declaração de requisito deve ser indivisível.
Claro	Deve ser claramente entendido, isto é, deve ser escrito em linguagem simples.
Completo	Deve carregar todas as informações necessárias para definição do requisito.
Correto	Deve expressar de forma acurada uma necessidade.
Conciso	Deve ter o texto estritamente necessário para o entendimento.
Factível	Deve ser possível a implementação do requisito.
Legal	Deve estar em conformidade com as normas, políticas e regulamentos aplicáveis.
Necessário	Deve estar presente para atender a uma necessidade identificada, isto é, não é opcional.
Priorizável	Deve ser possível identificar a importância para o atendimento de uma necessidade.
Não ambíguo	Deve existir uma única forma de interpretação da sentença.
Verificável	Deve ser possível verificar o requisito por algum método.
Alocável	Deve poder ser atribuído a um componente do objeto final.

(Continua)

Tabela 2.2 - Conclusão

Independente de implementação	Deve, simplesmente, expressar uma necessidade sem sugerir/impôr uma solução.
Escrita padronizada	Os requisitos devem ser descritos através de um padrão.
Único	Cada requisito deve ter um identificador próprio.
Isento de cláusulas de escape	Não devem ser usadas expressões como: exceto, a menos que, no caso de, se necessário.

Fonte: Adaptada de Hull et al. (2011) e Wiegers e Beaty (2013, p.203-206).

Outras características, no entanto, só podem ser analisadas de forma combinada sobre conjuntos de requisitos. A Tabela 2.3, a seguir, sumariza as principais:

Tabela 2.3 - Características do conjunto de requisitos.

Característica	Descrição
Consistente	Os requisitos não devem ser conflitantes entre si.
Não redundante	Cada requisito deve ser expresso uma única vez.
Estruturado	O documento de requisitos deve apresentar uma clara estruturação, agrupando requisitos de forma a facilitar o gerenciamento.
Rastreável	As ligações (relacionamentos) de um requisito, tal como um requisito fonte (regressivo), ou requisito derivado (progressivo) devem poder ser percorridas de forma a permitir o controle de consistência, devido às alterações de um determinado requisito.

Fonte: Adaptada de Hull et al. (2011) e Wiegers e Beaty (2013, p.205).

A importância das características específicas que os requisitos devem possuir pode ser ilustrada, por exemplo, pelo caso de software, em que a norma **IEEE Std 830-1998** - *Recommended Practice for Software Requirements Specifications* apresenta oito critérios que definem um bom SRS, conforme segue:

- a) correto;
- b) não ambíguo;
- c) completo;
- d) consistente;
- e) classificado por importância e/ou estabilidade;
- f) verificável;
- g) modificável;
- h) rastreável.

2.2.1. Classificação dos requisitos

A classificação de requisitos é importante na medida em que além de auxiliar na documentação e priorização, oferece, também, uma forma para identificar as possíveis fontes a serem pesquisadas durante o processo de desenvolvimento. No entanto, neste tópico, também, se nota que existe uma grande diversidade de visões quanto aos modos de classificação dos requisitos.

Encontram-se na literatura diversas classificações aplicadas aos requisitos, tais como a classificação por fontes, por tipos e por categorias.

Em (IEEE COMPUTER SOCIETY, 2004, p. 2-2,2-3) podem ser encontrados a classificação por requisitos funcionais e não funcionais; derivados ou propriedade emergente; e aqueles que são relativos a produto ou processo.

2.3. Atividades de engenharia de requisitos

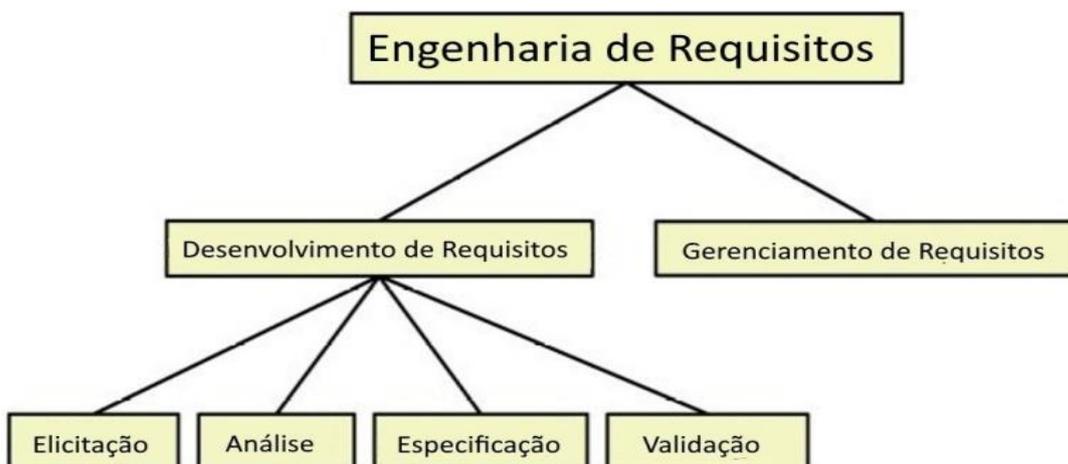
Entre o levantamento de necessidades, a partir da identificação dos *stakeholders*, e de um objeto ou serviço que vá atendê-las, até à entrega de forma completa, ocorre uma série de atividades/processos que não podem, sob qualquer ótica, ser classificados como sendo de simples execução.

Particularmente, cabe ressaltar que processos iniciais, que deverão realizar a transformação de uma necessidade identificada em uma declaração formal que possa ser verificada e validada, exigem um cuidado extremo, visto que sobre esta *baseline* serão desenvolvidos todos os processos posteriores. Uma boa dimensão da execução de tal tarefa pode ser vislumbrada a partir da seguinte citação:

A Engenharia de Requisitos é senso comum, mas é percebida como sendo difícil e não é bem entendida. Por essas razões, geralmente não é muito bem executada. As pressões crescentes em uma organização são, frequentemente, apontadas como as principais razões para não introduzir uma abordagem mais disciplinada para a Engenharia de Requisitos, mas o seu objetivo será fazer o trabalho de forma apropriada; assim, a tarefa do engenheiro de requisitos é exercitar como melhor auxiliar a empresa a atingir seu objetivo (HULL et al., 2011, p.xi).

As atividades de tratamento de requisitos podem ser divididas, segundo Wiegers e Beatty (2013), em dois ramos principais: Desenvolvimento de requisitos e o posterior gerenciamento de requisitos, conforme mostra a Figura 2.6, a seguir (WIEGERS; BEATTY, 2013, P.15).:

Figura 2.6 - Subdisciplinas de Engenharia de Requisitos de software.



Fonte: Adaptado de Wiegers e Beatty (2013, p.15).

2.3.1. Desenvolvimento de requisitos

2.3.1.1. Elicitação de requisitos

Trata-se de um conjunto de técnicas para obtenção de dados junto aos detentores das informações, para a construção de um sistema ou um produto ou, ainda, para melhorar um processo. A elicitação de requisitos ocorre de forma interativa, em que são usadas várias técnicas sendo a mais conhecida, e provavelmente a mais utilizada, a entrevista com os *stakeholders* para obter informações e conhecimentos que delinearão as necessidades que serão transformadas em requisitos, a partir dos quais serão definidas as funcionalidades para um sistema/produto/serviço a ser desenvolvido. Não se trata de uma simples coleta de dados, como pode parecer à primeira vista, porque nem sempre os dados estão disponíveis, e não há como garantir que todos os requisitos tenham sido obtidos logo de início, daí o cuidado que esta fase requer.

2.3.1.2. Análise

Com o conjunto de requisitos obtido na fase de elicitação dá-se início à fase de análise, em que os requisitos serão avaliados, individualmente e em conjunto, sob os diversos aspectos tais como: atributos, ambiguidades, redundâncias,

consistência, viabilidades técnica e econômica, possibilidades de verificação, e conexões com as necessidades identificadas, além de eventuais conflitos, visto que são obtidos de partes com visões diferentes.

Além disto, os requisitos são classificados e separados por tipos, nível de prioridade, etc. Nesta fase busca-se obter, também, um melhor entendimento do problema a ser solucionado através do desenvolvimento do projeto. Tal compreensão pode remeter a uma nova especificação da necessidade, voltando à fase de elicitação, para solucionar questões mal resolvidas e/ou complementar informações.

2.3.1.3. Especificação

O conjunto de requisitos obtidos, e verificados, na fase de análise, em linguagem natural, deverá ser traduzido para um formato organizado e escrito, através de um padrão que possa ser compreendido, de forma única, tanto pela equipe de desenvolvimento quanto pelos *stakeholders* e submetido aos processos subsequentes de revisão e validação.

É importante ressaltar que o conjunto de requisitos será a forma de comunicação entre os diversos “atores” envolvidos no processo sendo, portanto, de fundamental importância para a melhoria da qualidade da documentação de requisitos gerada.

2.3.1.4. Validação

O fechamento da fase de desenvolvimento de requisitos dá-se pela confirmação de um correto mapeamento que demonstra o relacionamento entre os itens do documento de requisitos e as necessidades identificadas através dos *stakeholders*.

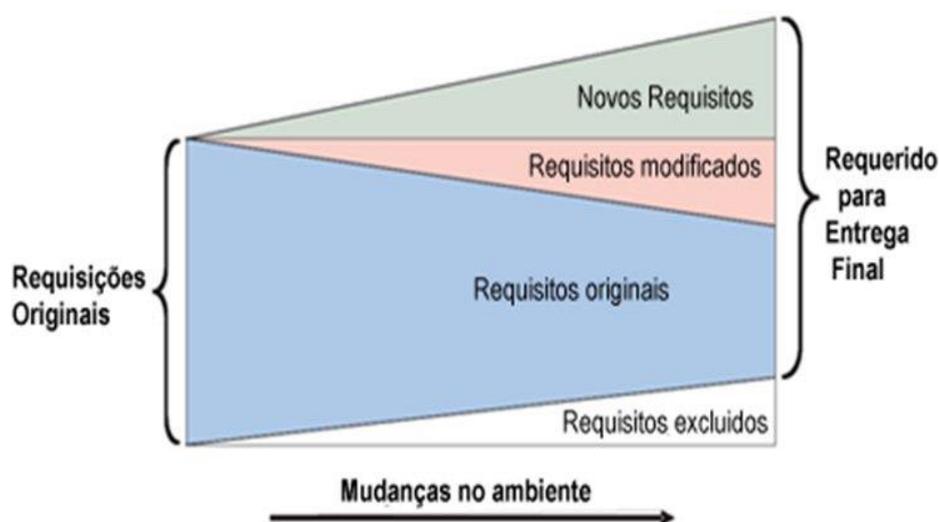
Esta confirmação será obtida através da aplicação de diversos métodos operacionais, que permitirão a certificação de que os requisitos atendem aos objetivos pretendidos.

2.3.2. Gerenciamento de requisitos

A alteração de requisitos durante o ciclo de vida de um produto/serviço é um fato concreto com o qual a Engenharia de Requisitos deve lidar. Esta alteração ocorre devido a diversos fatores, tais como: a evolução de tecnologia, o redirecionamento de objetivos do negócio em função, por exemplo, de concorrência, modificações na legislação e/ou, mesmo, um entendimento mais refinado do problema durante o curso do projeto, o que propicia, também, o aparecimento de novos requisitos.

A Figura 2.7, a seguir, ilustra como as alterações dos requisitos ocorrem ao longo de todo o processo de desenvolvimento, alcançando uma parcela considerável dos requisitos inicialmente estabelecidos.

Figura 2.7 - Mudanças nos requisitos.



Fonte: Adaptado de In cose (2011, p.232).

Este aspecto, dentre outros, demonstra que mesmo o trabalho bem elaborado nas fases anteriores, por si só, não será suficiente para o sucesso do projeto. São necessários processos bem elaborados, controlados e formais, de maneira a manter a documentação atualizada e o controle efetivo sobre o impacto e sobre a abrangência das alterações necessárias.

Albuquerque (2012) reforça a inevitabilidade de mudanças ao longo do ciclo de vida ao fazer a seguinte afirmação:

Mudanças são corriqueiras em projetos de desenvolvimento – à medida que um projeto avança, mudanças no planejamento original ou no produto são naturais e frequentes. Bloquear a sua ocorrência seria como eliminar a oportunidade de incorporação de avanços tecnológicos ou impedir a possibilidade de melhoria em um projeto (ALBUQUERQUE, 2012).

No sentido de tratar mudanças ocorridas em projetos de desenvolvimento, Albuquerque (2012) apresenta um trabalho onde discorre sobre a necessidade da implantação de um sistema de gerenciamento da configuração para o programa espacial brasileiro e focaliza especificamente os processos inerentes ao Gerenciamento de Configuração e Gerenciamento de Documentação.

Hull et al. (2011) afirmam que, se os requisitos apresentam um papel central no desenvolvimento de um sistema, eles precisam ser mantidos; e que mudar o projeto de um produto sem a devida atualização dos requisitos para refletir esta alteração, significa guardar problemas enormes para estágios posteriores do desenvolvimento; e, neste sentido, enfatizam que o gerenciamento de requisitos conecta-se, fortemente, com o gerenciamento de mudanças.

Ainda, tratando das questões do gerenciamento de requisitos, Hull et al. (2011) declaram que existem problemas específicos que tornam este gerenciamento muito mais difícil do que outras atividades de gerenciamento.

Para sustentar tal afirmação listam como problemas a serem enfrentados:

- 1) muito poucas pessoas têm experiência significativa, e isto é devido ao fato de que muito poucas organizações têm um processo de gerenciamento definido que seja aplicado por toda a organização; e, assim sendo, se as pessoas têm pouca experiência com gerenciamento de requisitos, podem, inclusive, não saber que atividades são necessárias para desenvolver requisitos;

- 2) várias pessoas não conseguem distinguir de forma correta a diferença entre requisitos de interessados ou *stakeholders* e requisitos de sistema; assim, costumam partir direto para uma solução proposta quando deveriam estabelecer um conjunto de requisitos independente de solução;
- 3) o método de gerenciamento de requisitos irá depender do tipo de organização onde o trabalho está sendo desenvolvido;
- 4) a dificuldade de monitorar o progresso quando os requisitos estão sendo gerados, o que é traduzido, por exemplo, pela dificuldade de saber se o conjunto de requisitos está completo, de forma a sinalizar que a atividade deve parar;
- 5) a questão permanente das mudanças, foco desta pesquisa, onde é afirmado que o gerenciamento de requisitos deveria focar, primariamente, no gerenciamento das mudanças. Isto porque qualquer mudança proposta envolverá um ou mais requisitos e o impacto ou o efeito de propagação das mudanças propostas são, muito frequentemente, difíceis de avaliar; e que sem esse conhecimento é impossível estimar o impacto nos custos e prazos, oriundos da introdução de uma mudança (HULL et al., 2011).

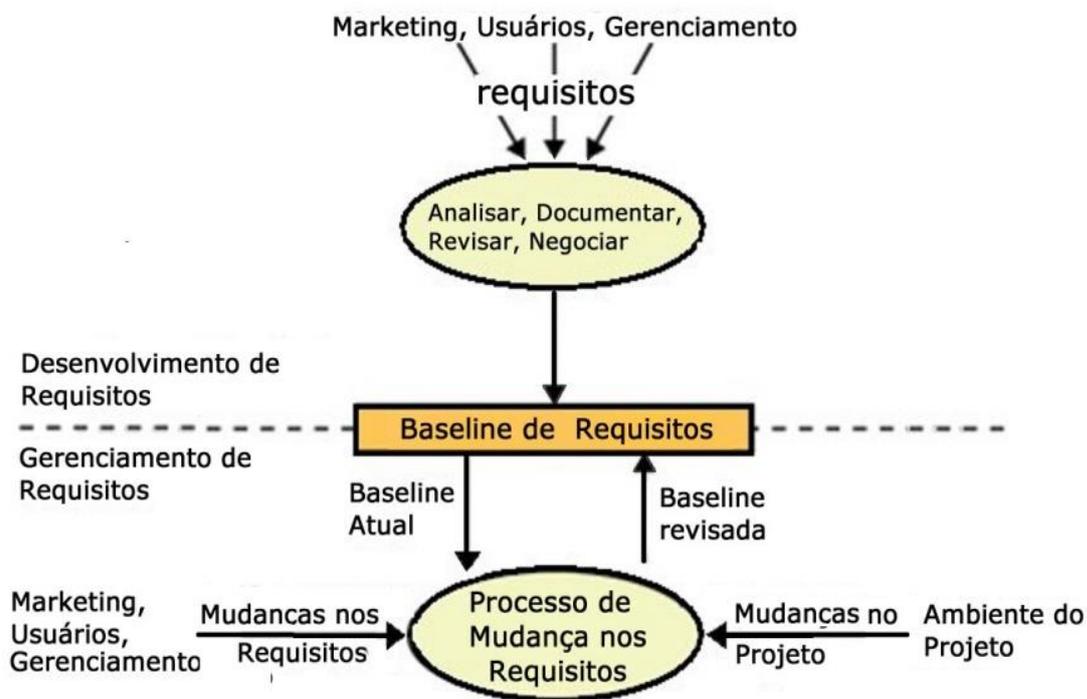
O processo de gerenciamento de requisitos engloba, segundo Wieggers e Beaty (2013), as seguintes atividades:

- a) definir uma *baseline* para os requisitos;
- b) avaliar e incorporar, de forma controlada, as mudanças aprovadas;
- c) manter a documentação de projeto atualizada e sincronizada com os requisitos durante o ciclo de vida;
- d) negociar novos acordos baseados no impacto estimado das alterações;

- e) definir relacionamentos e dependências existentes entre requisitos;
- f) rastrear requisitos individuais em relação aos seus correspondentes projetos, fontes e testes;
- g) manter o controle do *status* dos requisitos e atividades de mudanças, durante o projeto(WIEGERS; BEATY, 2013, P.17).

A Figura 2.8, a seguir, fornece uma visão dos limites entre o desenvolvimento dos requisitos e o gerenciamento de requisitos.

Figura 2.8 - Limites entre o desenvolvimento e o gerenciamento de requisitos.



Fonte: Adaptado de Wieggers e Beaty (2013, P.18).

Para Kotonya e Sommerville (1998), o gerenciamento de mudanças consiste na forma que são utilizados os procedimentos, processos e padrões que serão adotados para gerenciar as mudanças nos requisitos do sistema, e inclui (KOTONYA; SOMMERVILLE, 1998):

- a) as informações exigidas para as mudanças e o conjunto de procedimentos utilizado para realizar as mudanças;
- b) o conjunto de procedimentos usado para avaliar o impacto e o custo da mudança, bem como informações associadas à rastreabilidade;
- c) identificação dos *stakeholders* que formalmente consideram as solicitações de mudanças;
- d) o suporte de software necessário para o controle do processo de mudança.

Para Sommerville (2007) a evolução dos requisitos deve espelhar todas as alterações ocorridas ao longo do desenvolvimento do sistema, tanto no ambiente quanto nos objetivos pretendidos, devendo ser considerados sob duas classes:

- a) estáveis: que correspondem aos requisitos originários do principal objetivo do sistema, que estejam relacionados diretamente com o domínio do problema abordado;
- b) voláteis: que estão propensos a sofrerem diversas alterações durante o ciclo de desenvolvimento do sistema contemplando desde a fase inicial até a fase de operação e manutenção, por exemplo, requisitos ligados á políticas de governo (SOMMERVILLE, 2007).

Ainda de acordo com Sommerville (2007), os requisitos voláteis devem ser classificados conforme os tipos descritos a seguir:

- a) mutáveis: aqueles que são alterados em função das mudanças de ambiente no qual será operado;
- b) emergentes: aqueles que vão surgindo conforme o grau de compreensão do *stakeholder* for aumentando;

- c) consequentes: aqueles que resultam da introdução de sistemas de computação, os quais podem mudar os processos da organização, gerando novos meios de trabalho que por sua vez geram novos requisitos do sistema;
- d) compatíveis: são aqueles que dependem de sistemas, ou processos, específicos dentro da organização que, ao mudarem, podem forçar os requisitos de compatibilidade a evoluírem na mesma proporção destes (SOMMERVILLE, 2007).

Para Kotonya e Sommerville (1998), os fatores que contribuem de forma mais significativa para as alterações de requisitos, dentre outros, podem ser:

- a) erros, conflitos e inconsistências: normalmente tais problemas são detectados ao longo da fase de análise e validação de requisitos, ou durante o processo de desenvolvimento;
- b) evolução do conhecimento dos *stakeholders*: ao longo do ciclo de desenvolvimento do sistema é natural que os *stakeholders* se manifestem solicitando alterações, muitas vezes decorrentes de melhor entendimento do que realmente desejam;
- c) problemas técnicos de prazo e de custos: tais problemas podem ser percebidos no processo de implementação de um requisito. Quanto mais tarde forem percebidos mais oneroso tornar-se-á para o sistema a sua correção;
- d) mudanças nas prioridades dos *stakeholders*: a prioridade dos *stakeholders* pode se alterar ao longo do desenvolvimento do sistema em função de eventos que resultem em mudanças de objetivos, mudanças de equipe e outros;
- e) mudanças ambientais: o ambiente em que o sistema será operado pode ser alterado, o que acarretará mudanças nos requisitos já definidos para garantir operacionabilidade.

Em resumo, pode-se considerar segundo Kotonya e Sommerville (1998) que o processo de gerenciamento de mudanças nos requisitos corresponde ao conjunto de atividades necessárias para documentar, reportar, analisar, definir e avaliar custos e efetivar as mudanças de um conjunto de requisitos (KOTONYA; SOMMERVILLE, 1998).

2.4. Ferramentas

Existem diversos métodos para endereçamento do problema de Engenharia de Requisitos, assim como ferramentas sob a forma de programas de computador. No entanto, provavelmente pela reconhecida complexidade do tema, não se pode afirmar que o assunto esteja esgotado ou que algum destes métodos seja o definitivo, conforme se nota na seguinte citação, que trata de requisitos de software:

A existência de tantas técnicas de representação indica que nenhum método perfeito de representação foi ainda desenvolvido. Se um único destes métodos fosse claramente superior aos outros, então não há dúvida que este seria o padrão de fato usado por todos os projetos de software. Até onde pode ser determinado, nenhum método de representação é usado por mais do que dez por cento das aplicações de software (JONES, 2010, p. 438).

Por outro lado, o volume de informações a serem processadas, requisitos que podem chegar às dezenas de milhares, e a complexidade do tema tornam, virtualmente, impossível o eficiente gerenciamento do processo sem a utilização de uma ferramenta de software específica e adequada ao contexto.

Existe no mercado, atualmente, um grande número de ferramentas para tal fim, ou seja, programas voltados para a tarefa de auxiliar o processo de Engenharia de Requisitos, principalmente no quesito de rastreabilidade, em que as combinações de ligações podem crescer de forma explosiva.

Podem ser encontrados na Internet alguns sites que relacionam tais ferramentas; e, em outros a disponibilização de documentos que auxiliam na avaliação e na escolha da ferramenta com base em suas características e facilidades, sendo estas mostradas em resultados de *benchmarking* de ferramentas a partir de critérios bem estabelecidos. (VOLERE, 2015); (SEILEVEL, 2015).

Vale ressaltar que a escolha de métodos e ferramentas depende, fortemente, do contexto.

2.5. Sistemas aeroespaciais

O desenvolvimento de um processo de gerenciamento de requisitos voltado para sistemas aeroespaciais requer tratamento particular devido à complexidade das características próprias desta área de conhecimento. Torna-se, portanto, oportuno definir o que diferencia este processo neste domínio de outros existentes em outros domínios, como, por exemplo, aqueles que são mais apropriados para gerenciar os requisitos de um produto a ser desenvolvido para a indústria de eletrodomésticos.

Os sistemas aeroespaciais demandam um tratamento mais específico em função de possuírem algumas características bem diferenciadas, tais como:

- a) sistemas, em geral, únicos, ou personalizados para o ambiente de operação;
- b) apresentam, normalmente, ciclos de vida bastante longos;
- c) sistemas de alto custo;
- d) exigem critérios rígidos de qualidade;
- e) alta disponibilidade;
- f) alto desempenho;
- g) tempo de guarda de informações elevado;

- h) critérios de integridade e confidencialidade bem definidos;
- i) critérios para a eliminação de falhas simples ou não;
- j) emprego de redundância.

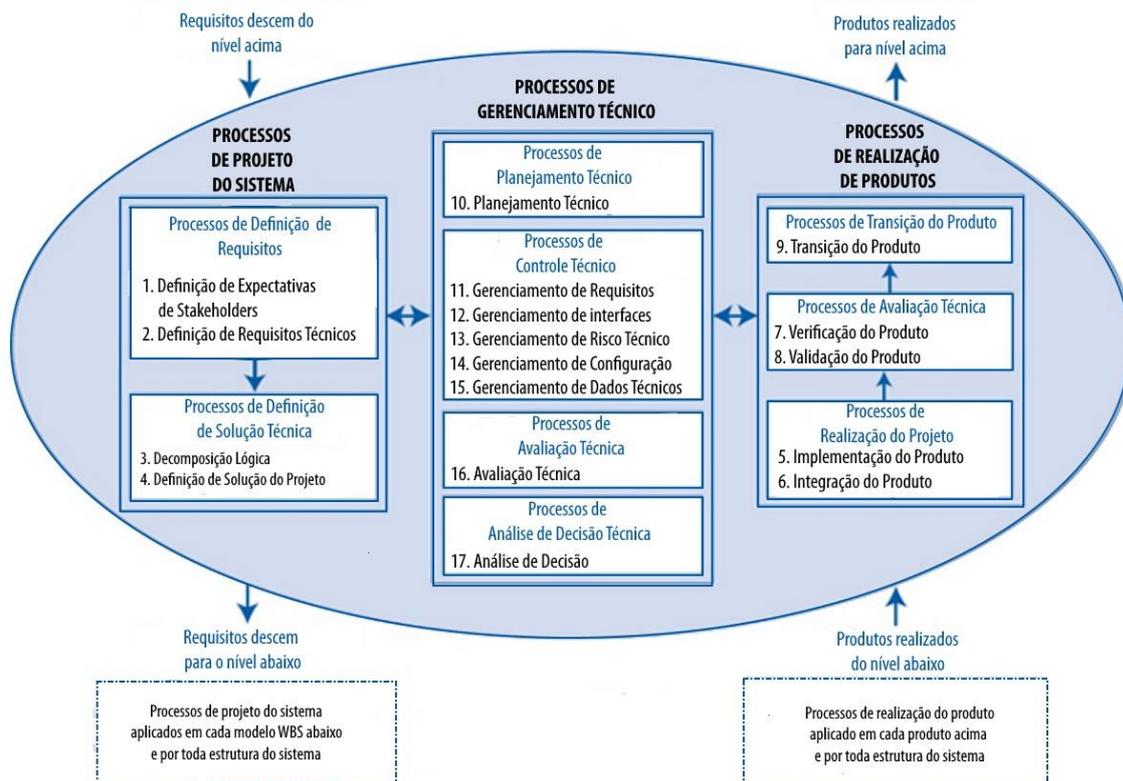
3 ESTUDO DOS MODELOS

A formulação da proposta desta pesquisa partiu dos estudos sobre gerenciamento de requisitos, extraídos de documentos de organizações envolvidas com os problemas de Engenharia de Sistemas, e de Engenharia Aeroespacial. A seguir são apresentadas quatro abordagens extraídas da literatura e que serviram de base para a construção do modelo proposto.

3.1. Abordagem NASA

A NASA apresenta no seu *System Engineering Handbook*, um modelo para desenvolvimento de Engenharia de Sistemas representado pela Figura 3.1, a seguir (NASA, 2007):

Figura 3.1 - Modelo da Engenharia de Sistemas.



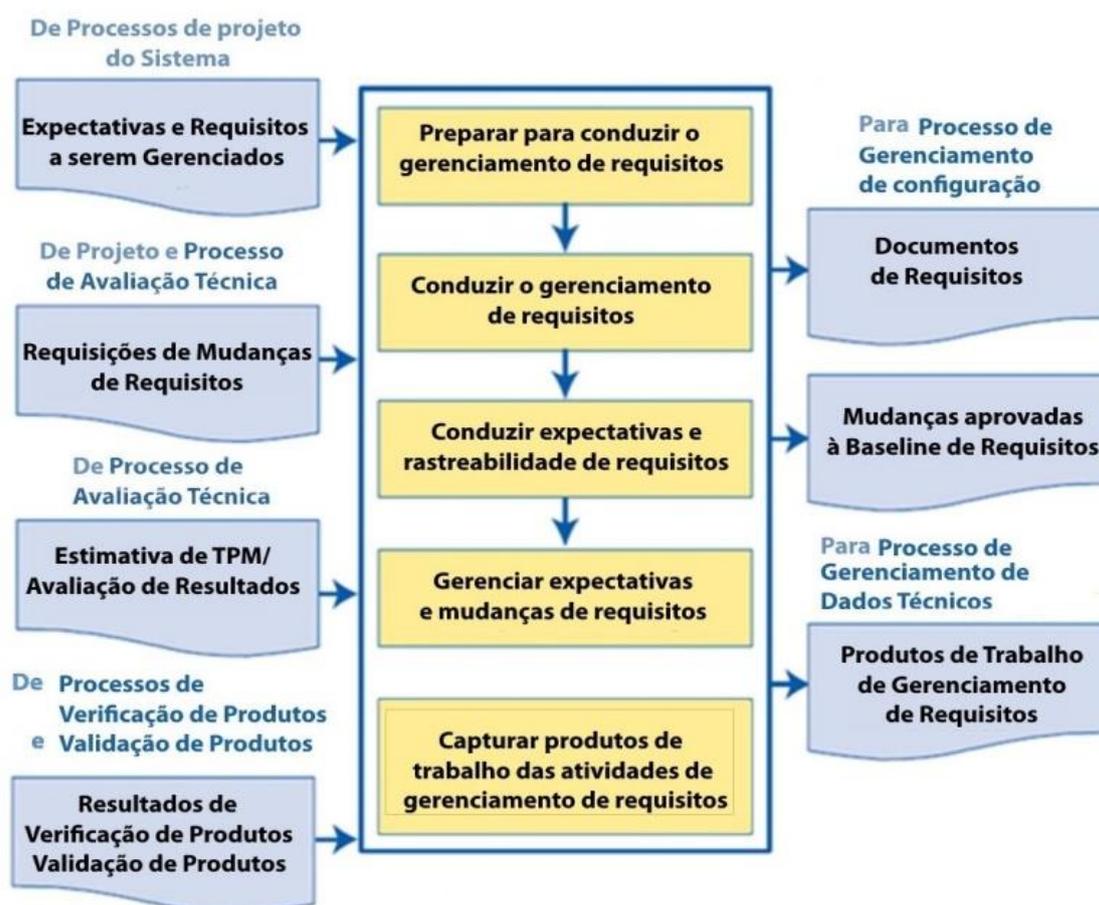
Fonte: Adaptado de NASA (2007, p.5).

Neste modelo, a Engenharia de Sistemas é apresentada de forma gráfica, destacando a execução de três conjuntos de processos principais: Processos

de projeto do sistema, Processos de gerenciamento técnico, e Processos de realização do produto, em que a tarefa de gerenciamento de requisitos é tratada como um dos processos, dentro do gerenciamento técnico.

A NASA aborda a Engenharia de Sistemas apresentando diversos processos descritos, detalhadamente, dentro das várias etapas a serem seguidas pelos métodos. Tais etapas são definidas como: entradas, atividades e saídas. Posteriormente, cada processo é decomposto em subprocessos, mantendo o mesmo nível de detalhamento. Para exemplificar esta abordagem, o item 11 da Figura 3.1 mostra o processo de gerenciamento de requisitos que é apresentado decomposto e descrito detalhadamente através da Figura 3.2, a seguir.

Figura 3.2 - Processo de gerenciamento de requisitos – NASA.



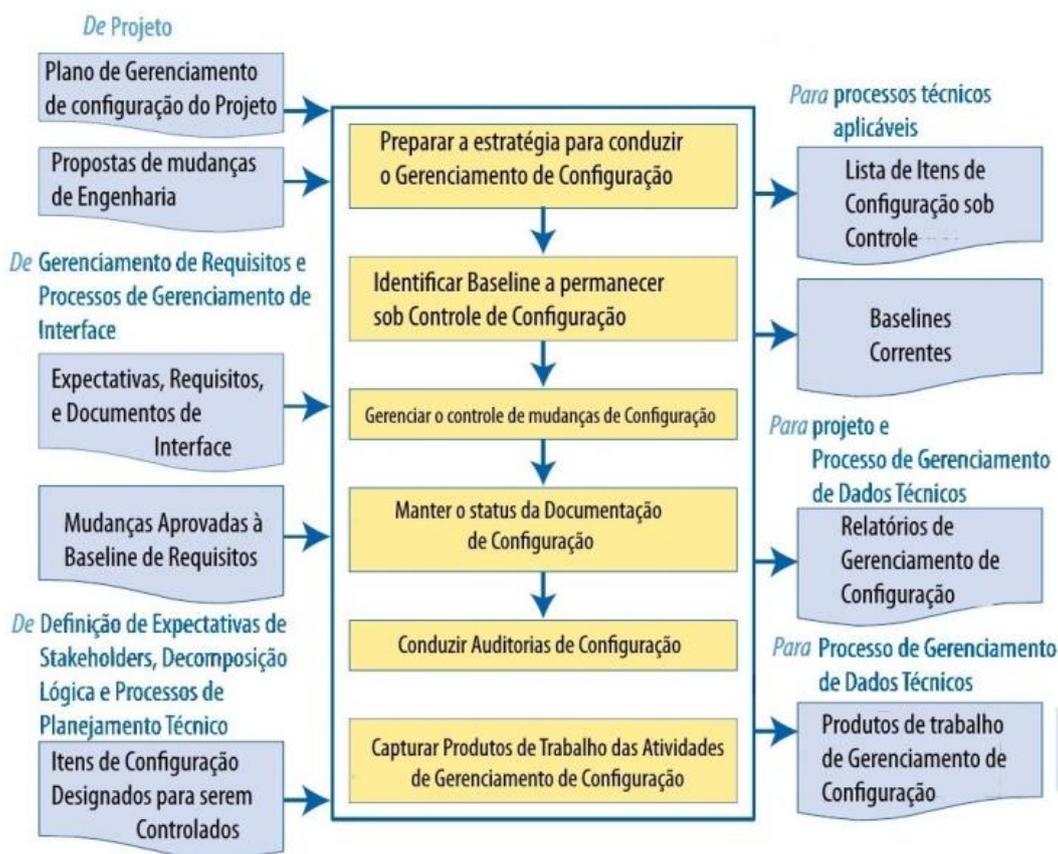
Fonte: Adaptado de NASA (2007, p.131).

Segundo o manual da NASA, o processo de gerenciamento de requisitos é utilizado para:

- a) gerenciar os requisitos do produto identificados, inseridos na *baseline*, e utilizados na definição do modelo WBS durante o desenvolvimento do sistema;
- b) permitir a rastreabilidade bidirecional para os requisitos do modelo WBS;
- c) gerenciar as mudanças realizadas sobre as *baselines* de requisitos estabelecidas ao longo do ciclo de vida dos produtos do sistema. (NASA, 2007).

Vale notar que o gerenciamento de mudanças, apresentado pela Figura 3.3, é explicitamente citado pela NASA como sendo um dos pilares que suporta todo o processo de gerenciamento de requisitos e que é imperativo que todas as mudanças sejam profundamente avaliadas. Tal avaliação permitirá determinar os impactos na arquitetura, projeto, interfaces, CONOPS e em requisitos de baixo e alto nível. Além disso, recomenda que devem ser executados processos rigorosos de verificação e de validação para garantir que os requisitos satisfaçam as necessidades dos *stakeholders* e, conseqüentemente, estejam de acordo com os objetivos definidos para o projeto.

Figura 3.3 - Processo de gerenciamento de mudanças – NASA.



Fonte: Adaptado de NASA (2007, p.151).

As mudanças devem ser submetidas a ciclos de revisão e de aprovação de maneira a permitir a rastreabilidade e garantir que os impactos sejam completamente avaliados considerando-se todas as partes do sistema.

De acordo com o manual, após a certificação e validação, as mudanças solicitadas deverão compor o processo formal de controle de configuração e qualquer alteração, nos requisitos, deve ser aprovada pelo Conselho de Controle de Configuração, ressaltando que o pessoal técnico deve comunicar para todos os interessados, e de forma imediata, os requisitos aprovados.

Em um item denominado “aspectos-chave para o gerenciamento de requisitos”, o manual elenca os seguintes aspectos: a mudança nos requisitos, a retroalimentação da *baseline* de requisitos e o crescimento súbito do volume de requisitos.

Quanto às solicitações de mudanças nos requisitos, afirma que o efetivo gerenciamento de alterações dos requisitos requer um processo que avalie o impacto da alteração proposta antes da sua aprovação e implementação. Isto, normalmente, é efetuado através da execução de um processo de Gerenciamento de Configuração (CM). Para que se possa cumprir esta função, a configuração da *baseline* deve ser documentada e ferramentas devem ser utilizadas para avaliar o impacto na *baseline*, citando como ferramentas para esta avaliação:

- a) margens de desempenho: Devem ser avaliados os impactos da mudança sobre as margens de desempenho do sistema;
- b) lista dos avaliadores de tópicos do CM: para garantir que o pessoal adequado está avaliando as mudanças e fornecendo o real impacto das mudanças;
- c) lista de ameaças e riscos do sistema: os riscos do sistema devem ser utilizados para identificar riscos ao projeto e os custos, atrasos e aspectos técnicos associados, lembrando que alterações na *baseline* podem afetar os efeitos e probabilidades de riscos já identificados, além de introduzir novos riscos no projeto.

É, ainda, apresentado um plano para o gerenciamento de requisitos, recomendando-se abordar o seguinte:

- a) identificação de *stakeholders* relevantes;
- b) definição de agenda para execução das atividades e dos procedimentos do gerenciamento de requisitos;
- c) definição de atribuição de responsabilidades, autoridade e recursos adequados para cumprir as atividades de gerenciamento de requisitos;
- d) definição do nível de controle do gerenciamento de dados/configuração; e

- e) identificação de treinamento necessário para o pessoal que irá executar as atividades de gerenciamento de requisitos.

Para a NASA, há que se considerar, ainda, o uso de ferramentas de software para gerenciamento de requisitos. Ressalta que na seleção da ferramenta é importante definir os procedimentos do projeto, especificar como os requisitos serão organizados no banco de dados, bem como a forma e o contexto em que a ferramenta será utilizada.

3.2. Abordagem ECSS

Como o número de documentos que compõem o conjunto de normas, *handbooks* e memorandos técnicos é elevado, convém, inicialmente, apresentar uma explanação sintética sobre a organização desses documentos.

O sistema de normas da ECSS é resultado de esforço colaborativo entre as agências e indústrias espaciais europeias, que produziu um conjunto consistente de documentos para o tratamento de programas espaciais.

Estas normas estão agrupadas em três ramos principais:

- a) gerenciamento de projetos;
- b) engenharia;
- c) garantia do produto.

As normas apresentam um Sistema de numeração no formato ECSS-<X>-ST-<Número> <Versão>, em que:

<X> representa o ramo, tomando os valores de P ou S para sistema ECSS, E para engenharia, M para gerenciamento e Q para garantia do produto;

<número> é um ou dois grupos de dois dígitos;

<versão> é uma letra de A em diante;

As normas focam, primariamente, no que é necessário para estar em conformidade com cada padrão, ao invés de como os procedimentos deverão ser realizados.

Os *handbooks* apresentam o mesmo sistema de numeração das normas, apenas trocando o -ST- por -HB-.

A ECSS não apresenta um documento específico para o tratamento de gerenciamento de requisitos. As referências ao assunto estão distribuídas em vários documentos. A norma ECSS-M-ST-40C, por exemplo, trata de gerenciamento de informação e de configurações, e nesta podem ser encontradas orientações fundamentais de procedimentos, que são inerentes ao gerenciamento de requisitos, tais como, o estabelecimento de diferentes categorias de *baselines*, códigos para identificação única de componentes, classificação de mudanças e de desvios.

Convém reafirmar que requisitos bem elaborados evitam futuras solicitações de mudanças. Neste sentido a norma ECSS-E-ST-10-06C apresenta uma visão geral do processo para estabelecimento da especificação de requisitos técnicos (TS) na fase 0 e fase A de um projeto, na qual define os diferentes tipos de requisitos técnicos, funcionais, de missão e ambientais, dentre outros e define, também, os requisitos, em geral, que compõe a TS, os requisitos necessários para a atividade de formulação de requisitos técnicos e recomendações sobre a forma de escrever corretamente requisitos, citando as formas verbais a serem utilizadas e restrições na forma de termos a serem evitados,

As análises desta pesquisa foram concentradas em duas normas, que descrevem os padrões relativos ao aspecto de Engenharia de Requisitos, tratados pelo ECSS System.

- a) ECSS-E-ST-10 - *System Engineering General Requirements*;
- b) ECSS-M-ST-40C Rev. 1 - *Configuration and Information Management*.

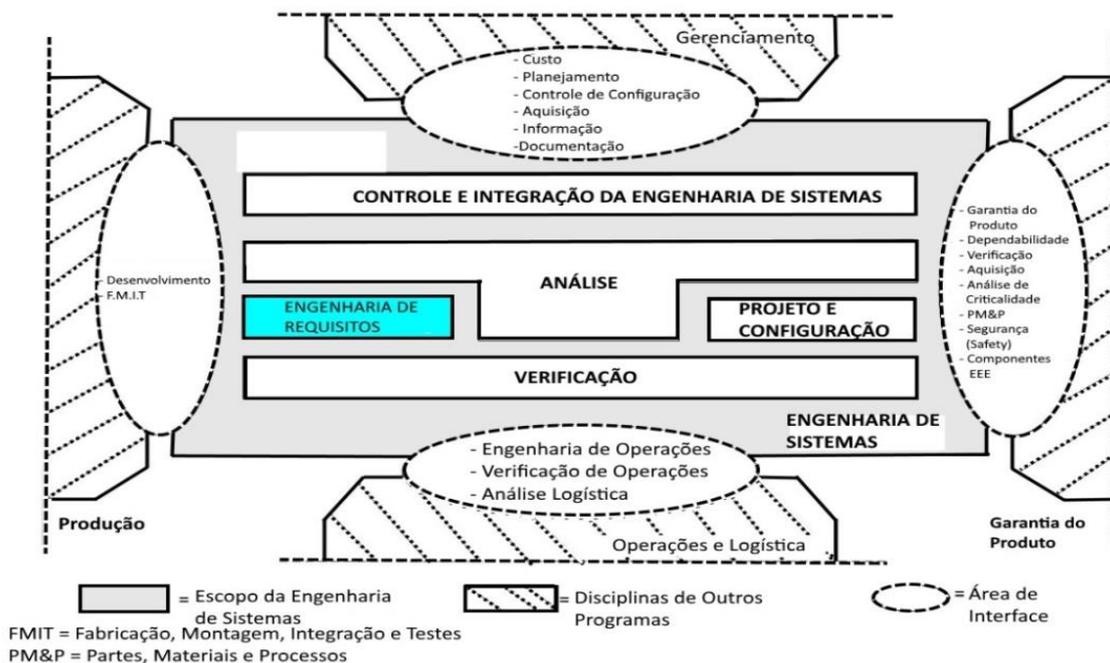
3.2.1. ECSS-E-ST-10C

Esta norma especifica os requisitos necessários para a implantação de Engenharia de Sistemas para o desenvolvimento de produtos e sistemas espaciais.

Apresenta a Figura 3.4 em que são mostradas as fronteiras da Engenharia de Sistemas e seus relacionamentos com outras áreas, tais como produção, gerenciamento e garantia do produto, e sua divisão interna conforme mostrada a seguir:

- a) Engenharia de Requisitos: tópico que consiste na descrição da análise, validação, alocação e manutenção de requisitos e dos processos de análise, configuração e projeto, verificação e controle e integração da Engenharia de Sistemas;

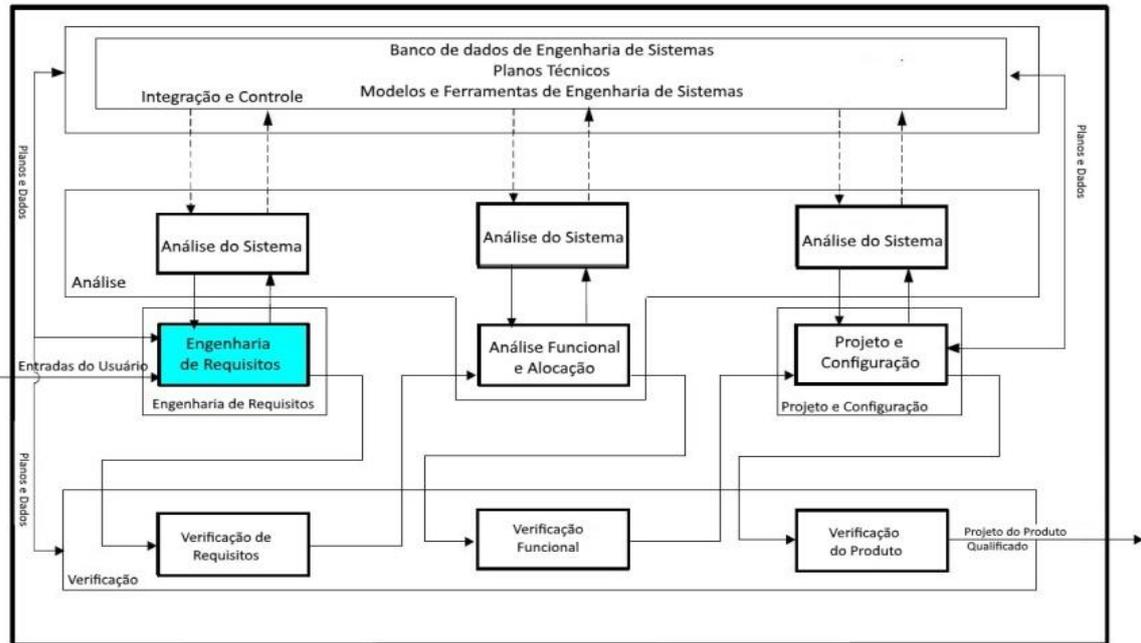
Figura 3.4 - Funções e fronteiras da Engenharia de Sistemas- ECSS.



Fonte: Adaptado de EUROPEAN SPACE AGENCY-ESA (2009).

A Figura 3.5 mostra as funções da Engenharia de Sistemas, seus relacionamentos e suas principais atividades, com destaque para a Engenharia de Requisitos responsável por expressar os requisitos dos *stakeholders* para uma solução a ser entregue.

Figura 3.5 - Funções da Engenharia de Sistemas e relacionamentos-ECSS.



Fonte: Adaptado de EUROPEAN SPACE AGENCY-ESA (2009).

Nesta norma são apresentados os requisitos para implementação da Engenharia de Sistemas, mostrando estes requisitos na forma usual e apresentando como exemplo de requisito geral a seguinte descrição:

A Engenharia de Sistemas da organização deve: derivar, gerar, controlar e manter o conjunto de requisitos para os elementos de baixo nível, definindo suas restrições operacionais e de projeto, e os parâmetros de funcionalidade e desempenho e verificação, aspectos necessários para atender aos requisitos de sistema definidos pelos *stakeholders*.

Afirma a necessidade de estabelecimento de uma matriz de rastreabilidade para todos os requisitos. Tal matriz deverá mostrar, dentre outros aspectos:

- a) os relacionamentos entre os requisitos e suas fontes;
- b) as mudanças no projeto que geram modificações nos requisitos;
- c) os relacionamentos com os requisitos de baixo nível;
- d) os seus fechamentos de verificação.

Além disto, apresenta os requisitos que se referem aos outros processos, necessários, para a Engenharia de Requisitos, tais como: especificação técnica, análise, alocação, métodos de verificação, validação, manutenção e *baseline* de requisitos.

Documentos de Descrição de Requisitos (DRD - *Document Requirements Descriptions*) para vários aspectos da Engenharia de Sistemas são apresentados nos anexos da norma, inclusive o DRD para a Matriz de Rastreabilidade de Requisitos (RTM - *Requirement Traceability Matrix*), que mostra detalhes de interesse, relacionados às comparações que foram realizadas por este trabalho.

Cita que o objetivo da matriz de rastreabilidade é auxiliar na verificação de que todos os requisitos, declarados e derivados, estejam alocados em componentes do sistema e em outros itens de entrega (rastreamento progressivo), assim como determina a fonte de tais requisitos (rastreamento regressivo), afirmando que a rastreabilidade de requisitos inclui informação como capacidades, elementos de projeto e testes, e outras.

Afirma que a RTM garante a identificação de componentes, quando é realizada uma alteração em um requisito, o que facilita a determinação dos desdobramentos da alteração solicitada em termos de custos, benefícios e prazos. Finalmente, fornece mais orientações sob a forma de requisitos e de como devem ser tratadas as interligações entre eles.

3.2.2. ECSS-M-ST-40C Rev. 1

Esta norma está estruturada em duas partes, sendo que em uma das partes são apresentados os processos e na outra parte os requisitos de forma detalhada. Isto permite subsidiar o processo de gerenciamento de documentação e de configuração de produtos para projetos espaciais.

Informa que o Gerenciamento de Configuração e de documentação/informação são processos inter-relacionados para gerenciar projetos e mostra as principais

atividades, as quais foram reproduzidas através da Figura 3.6 e são descritas a seguir:

- a) planejamento e gerenciamento;
- b) implementação das atividades de Gerenciamento de Mudanças (CM – *Change management*), como identificação de configuração, contabilidade de *status*, verificação e auditoria;
- c) implementação das atividades de gerenciamento de documentação e informação (IDM- *information documentation management*), como criação, coleta, revisão, entrega, armazenamento e arquivamento;

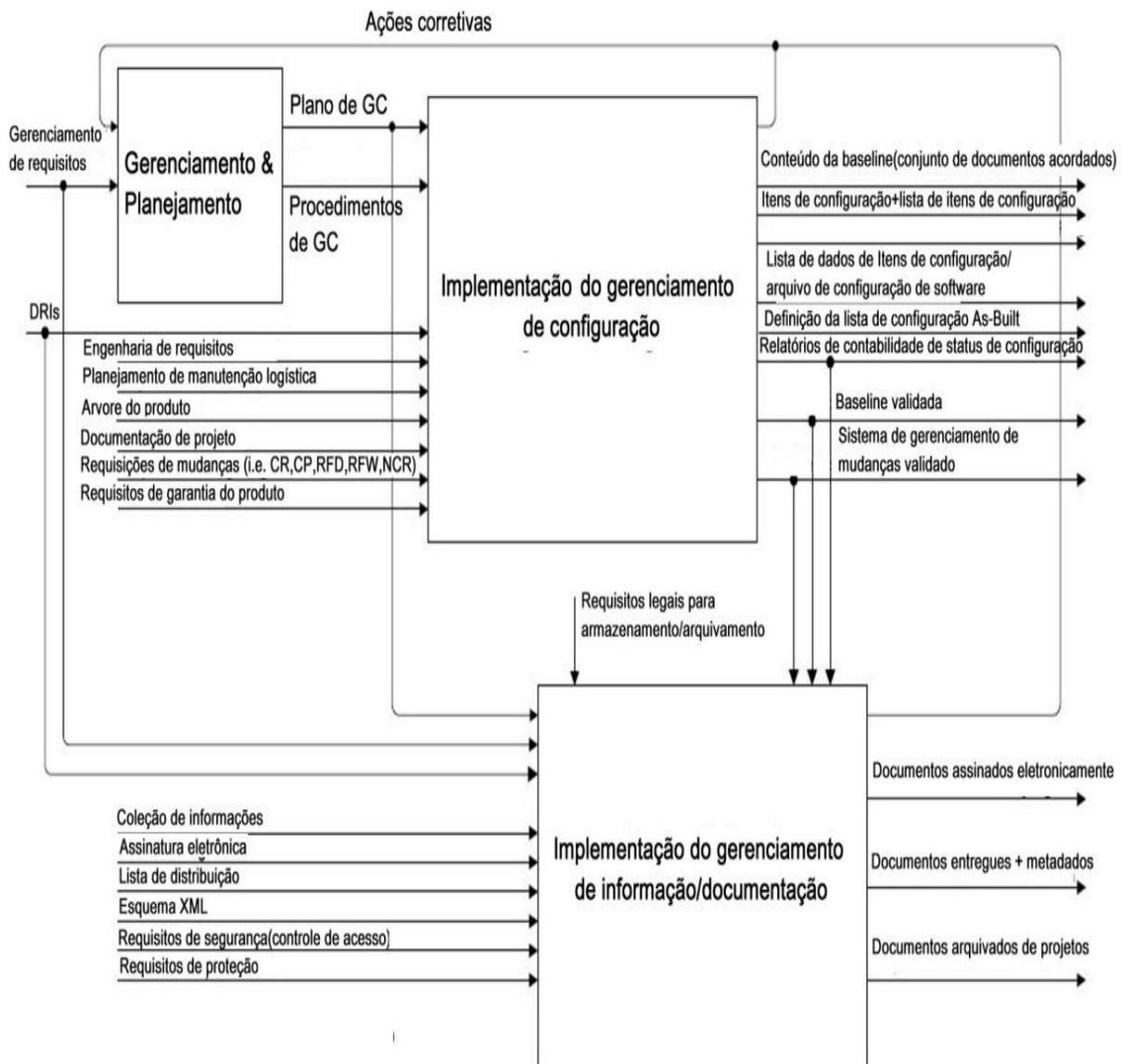
Define Gerenciamento de Configuração como um processo para estabelecer e manter registro, consistente, das características físicas e funcionais de um produto, em comparação ao definido em seu projeto e em seus requisitos operacionais, além de recomendar que tais procedimentos devam ser aplicados durante todo o ciclo de vida do projeto. Este processo permite:

- a) conhecer, em qualquer tempo, a descrição técnica de um produto a partir de documento aprovado;
- b) registrar e controlar a evolução da descrição técnica de produtos do sistema;
- c) fornecer a rastreabilidade da evolução de descrição técnica de um produto;
- d) garantir a consistência das interfaces internas;
- e) verificar e demonstrar, para todos os atores envolvidos, que a documentação representa uma descrição fiel do produto documentado;
- f) identificar a *baseline* corrente de configuração e a configuração “*as-built*” de um produto, para registrar as discrepâncias identificadas

durante a produção, entrega, ou ao longo da operação e da distribuição, para posterior uso;

- g) dar conhecimento a qualquer ator sobre as limitações e as possibilidades operacionais de cada item do produto e da existência de não conformidades e seus efeitos sobre outros itens.

Figura 3.6 - Gerenciamento de Configuração-ECSS.



NOTA: Ações corretivas são aprimoramentos no próprio processo como consequência de lições aprendidas e de qualquer realimentação fornecida no projeto

Gerenciamento de configuração

Fonte: Adaptado de EUROPEAN SPACE AGENCY-ESA (2009).

A norma apresenta a necessidade da definição de um plano de Gerenciamento de Configuração, que tem o propósito de definir os processos e os recursos que serão utilizados para o gerenciamento de forma controlada, e rastreável, durante o ciclo de vida do programa, ou do projeto, e que, além disto, deve ser um modo eficiente para a comparação entre a configuração prevista (*as-designed*) e a atual (*as-built*) do produto entregue.

A implementação do Gerenciamento de Configuração, mostrada na Figura 3.7, a seguir, compreende a definição, organização e a execução e supervisão de várias atividades conforme descritas, a seguir:

a) identificação da configuração:

- para identificar a arquitetura do produto;
- para selecionar os itens de configuração e definir seus documentos;
- para estabelecer a identificação de documentos e produtos;
- para definir requisitos para identificação de mídias de software;
- para estabelecer *baselines* para o gerenciamento do projeto e dos requisitos;

b) controle de configuração:

- para estabelecer e implementar o processo para produtos individuais, sistemas e interfaces;
- para registrar e controlar a configuração do produto, a qualquer tempo de sua evolução;
- para registrar diferentes configurações de um produto;
- para definir e manter repositórios, ou bibliotecas de software, onde todas as *baselines* criadas sejam mantidas em ambiente controlado;
- Para armazenar e manter produtos de software e mídias, incluindo cópias de segurança, em ambiente controlado;

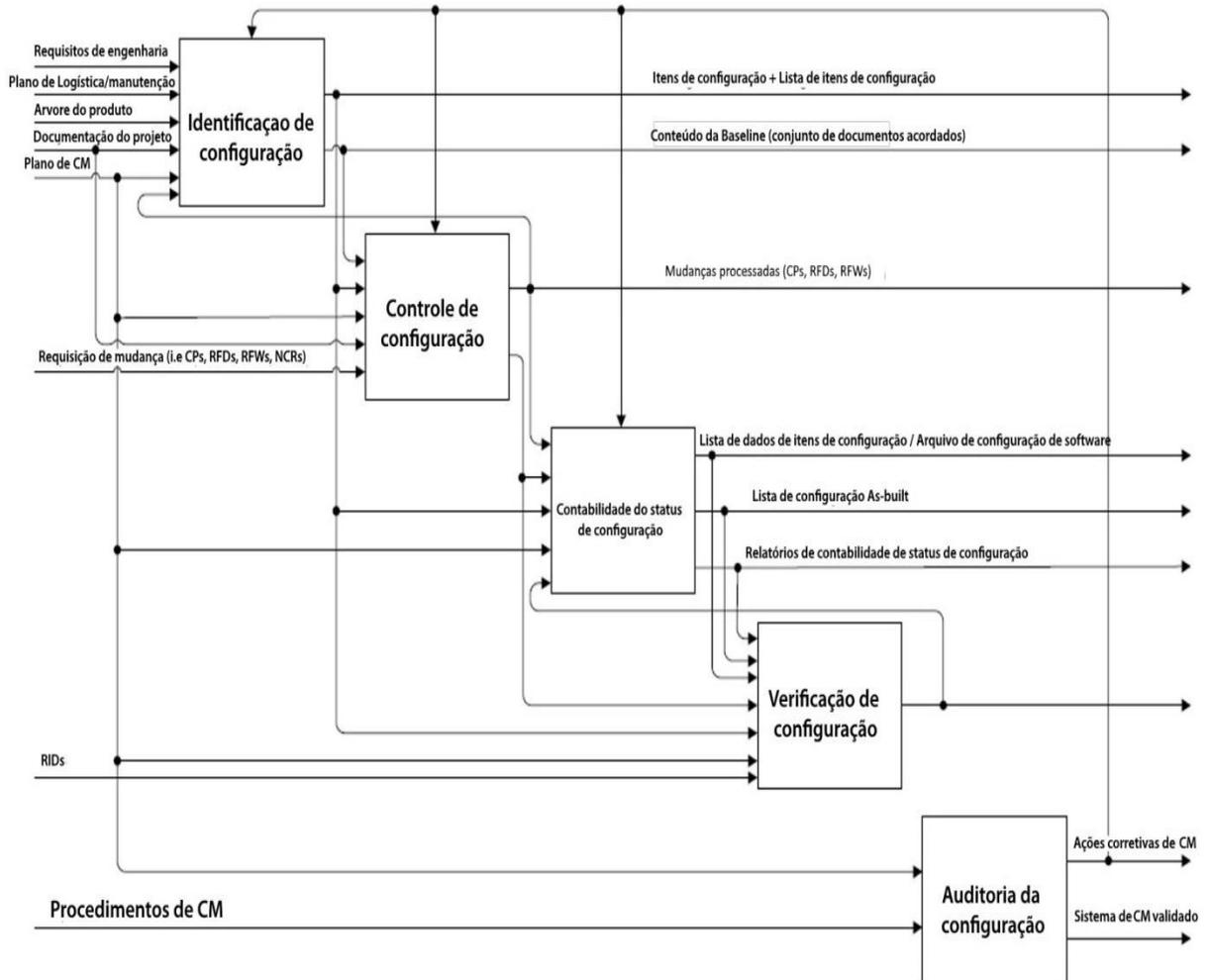
c) contabilidade de *status* de configuração;

- para fornecer a definição do produto, por referência, para *status* de configuração aprovados e registrados;
- para habilitar o acesso às bibliotecas de software, conforme privilégios estabelecidos;

d) auditoria e verificação de configuração:

- para verificar e demonstrar que o produto está em conformidade com as suas características físicas, funcionais e de desempenho, documentadas;
- Para verificar que o Sistema de Gerenciamento de Configuração é efetivo e atende aos requisitos de gerenciamento do programa, ou projeto.

Figura 3.7 - Implementação do Gerenciamento de Configuração-ECSS.



Fonte: Adaptado de EUROPEAN SPACE AGENCY-ESA (2009).

Adicionalmente, a norma descreve em maiores detalhes os requisitos para os processos de:

- a) identificação de configuração: em que são detalhados os esquemas para identificação de itens de configuração, a seleção dos itens de configuração; a *baseline* de configuração, identificação única, identificação de mídias e arquivos digitais;
- b) controle de configuração: apresentando o detalhamento dos procedimentos de alterações, Conselho de Controle de Configuração (CCB), classificação de mudanças, ou divergências e controle de interfaces;
- c) contabilidade de *status* de configuração;
- d) verificação de configuração;
- e) auditoria do processo de Gerenciamento de Configuração;
- f) implementação do gerenciamento de informação/documentação.

Apresenta, ainda, um capítulo com os requisitos para o Gerenciamento de Configuração.

3.3. Abordagem INCOSE

O *Handbook* de Engenharia de Sistemas do INCOSE, versão 3.2.2 de 2011, declara-se consistente com a norma ISO/IEC 15288: 2008– *Systems and software Engineering – System Life Cycle Processes*, com o intuito de garantir a sua aplicabilidade em vários domínios de aplicação. Afirma ainda que a norma internacional apresenta uma descrição genérica dos processos, enquanto o manual vai além, apresentando maior detalhamento dos processos e das atividades, para a execução dos mesmos.

Não há um capítulo exclusivo para os processos de gerenciamento de requisitos, mas tal assunto pode ser encontrado distribuído em dois pontos principais do documento:

- a) capítulo 4 – Processos técnicos;
- b) capítulo 5 – Processo de projetos.

A norma ISO/IEC 15288: 2008 identifica quatro grupos de processos para dar suporte à Engenharia de Sistemas: processos facilitadores organizacionais e de projeto, processos de projeto, processos técnicos e processos de contratos. O *handbook* seguindo esta orientação apresenta os processos de requisito, basicamente dentro dos processos técnicos, nos quais são descritos os procedimentos para definição de requisitos de *stakeholders* e de procedimentos para a análise de requisitos e de processos de projeto, onde é feito o detalhamento do Gerenciamento de Configuração.

O capítulo de processos técnicos, referenciando a ISO/IEC 15288: 2008, afirma que estes processos são utilizados durante todo o ciclo de vida de um sistema, sendo adotados para estabelecer os requisitos para o sistema como base para o esforço de criação de uma solução e permitir a reprodução consistente do produto, quando necessário, além de fornecer suporte ao longo da vida útil do sistema, inclusive no seu período de descarte.

Dentro deste capítulo são detalhados em maior profundidade, através de diagramas de contexto, as entradas, saídas e as respectivas atividades, caracterizando os seguintes processos:

- a) definição de requisitos de *stakeholders*;
- b) análise de requisitos;
- c) projeto da arquitetura;
- d) implementação;

- e) integração;
- f) verificação;
- g) transição;
- h) validação;
- i) operação;
- j) manutenção;
- k) descarte.

Ao longo da descrição do processo de definição de requisitos de *stakeholders* são descritas as atividades relativas ao desenvolvimento de requisitos, tais como:

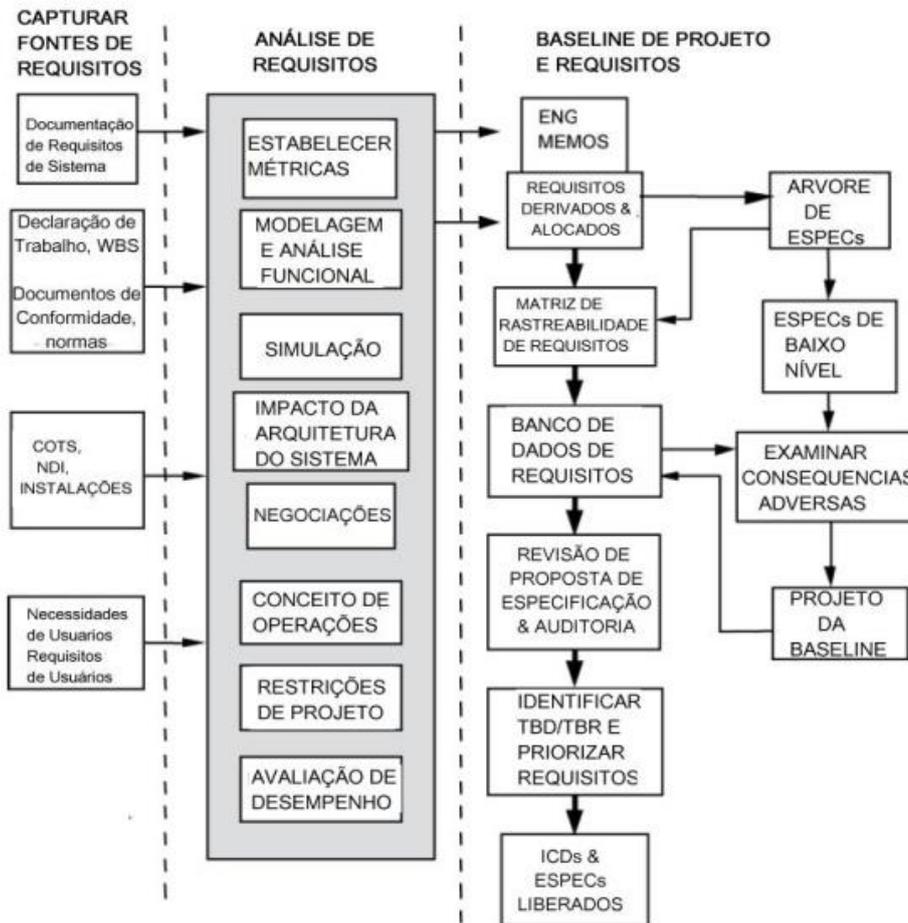
- a) elicitación dos requisitos de *stakeholders*;
 - identificar *stakeholders* que terão interesse no sistema ao longo do ciclo de vida,
 - elicitar os requisitos.
- b) definição de requisitos de *stakeholders*;
 - definição de restrições impostas por contratos legados,
 - construção de cenários para definição de documentos conceituais,
 - estabelecimento do desempenho desejado e crítico do sistema,
 - estabelecimento de medidas de efetividade (MoE) e adequação.
- c) análise e manutenção dos requisitos de *stakeholders*;
 - analisar requisitos quanto à clareza, completude e consistência,
 - negociar modificações para os requisitos impraticáveis,
 - validar, registrar e manter os requisitos de *stakeholders* por todo o ciclo de vida para efeitos de histórico e arquivamento,

- estabelecer e manter a matriz de rastreabilidade de requisitos, para documentar como os requisitos formais irão atender os objetivos pretendidos e obter a concordância formal.

Apresenta como boa prática que uma vez que os requisitos tenham sido estabelecidos sejam formalmente colocados sob controle de configuração; escrever de forma clara e criar declarações com valores quantificáveis e obter as fontes e premissas para cada requisito.

O processo de análise de requisitos é apresentado com o propósito de transformar os requisitos, capturados dos *stakeholders*, em uma visão técnica do produto almejado. Em tal processo são detalhadas as atividades inerentes ao desenvolvimento de requisitos, tais como: a avaliação de características de bons requisitos, a técnica de escrita desses requisitos e, também, é citado que deve ser mantida a continuidade de controle de configuração e de rastreabilidade. Detalham-se, ainda, o processo para definir, derivar e refinar requisitos funcionais e de desempenho, cujo fluxo é apresentado pela Figura 3.8, a seguir:

Figura 3.8 - Fluxo, alocação e derivação de requisitos - INCOSE.

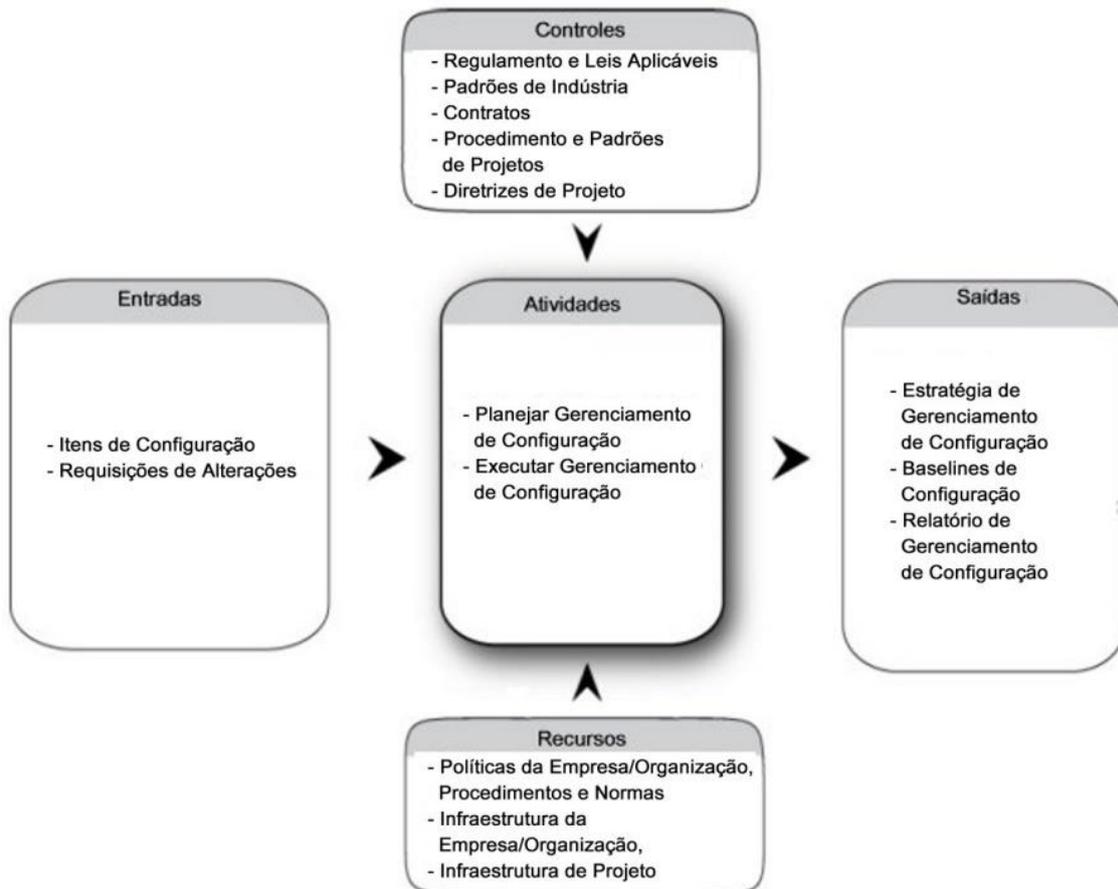


Fonte: Adaptado de INCOSE (2011, p.84).

No capítulo de processos de projetos, o INCOSE trata o Gerenciamento de Configuração definindo o propósito do processo como sendo o estabelecimento e a manutenção do controle dos requisitos, da documentação e dos artefatos produzidos durante o ciclo de vida do sistema, tornando-os disponíveis para as partes envolvidas. Recomenda, ainda, que diante da inevitabilidade das mudanças o gerenciamento do seu impacto no projeto faça parte das atividades relativas ao Gerenciamento de Configuração. Deve ser garantido que cada alteração proposta realmente seja necessária e que a solução recomendada tenha sido a que apresentou melhor custo-benefício.

A presente pesquisa considera este como tópico de maior interesse, pois, aqui, são tratadas as requisições de mudanças nos requisitos. O processo de configuração de mudanças é ilustrado através da Figura 3.9, a seguir.

Figura 3.9 - Diagrama de contexto para o processo de Gerenciamento de Configuração-INCOSE.



Fonte: Adaptado de INCOSE (2011, p.229).

Segundo o *handbook* do INCOSE, o propósito do Gerenciamento de Configuração é definir e manter a integridade de todas as saídas identificadas de um projeto, ou processo, mantendo-as disponíveis para as partes interessadas. Isto é realizado através do controle efetivo da evolução da configuração do sistema durante seu ciclo de vida, sendo fundamental para atingir este objetivo, o estabelecimento, o controle e a manutenção das *baselines de hardware e software*.

O diagrama de contexto apresentado define, dentre outras as atividades, o desenvolvimento de um Plano de Gerenciamento da Configuração (*CMP - Configuration Management Plan*), onde deverá ser estabelecido um controle de configuração que inclua procedimentos para avaliação, aprovação, validação e verificação das requisições de mudanças de engenharia (ECRs) e a execução do gerenciamento. Tal plano deverá contemplar a identificação dos elementos a serem controlados, o estabelecimento e controle das *baselines*, o desenvolvimento e a manutenção de documentação de controle, com a devida comunicação dos estados dos itens à equipe de projeto, e a previsão de execução de auditorias programadas para validação das *baselines*.

A entrada para o processo é composta pelos itens sob controle de configuração e pelas requisições de alterações. A saída constitui-se na estratégia de gerenciamento, nas *baselines* de configuração com os itens colocados sob controle formal de mudanças e no relatório de alterações, onde são documentados os impactos para qualquer processo, organização, decisão (inclusive qualquer notificação da mudança requerida), produto, ou serviço afetado pela alteração requerida.

O Manual do INCOSE possui, ainda, uma seção que apresenta sugestões e abordagens, tais como:

- a) estabelecer um Conselho de Controle de Configuração (CCB) representado por *stakeholders* e pelos participantes das áreas de engenharia do projeto da organização;
- b) iniciar o processo de Gerenciamento de Configuração no estágio inicial do sistema e continuar executando o procedimento até o descarte do produto/serviço.

Em relação à elaboração do plano, o manual INCOSE menciona que os esforços para o Gerenciamento de Configuração deverão ser executados desde o início do projeto e definidos no plano de gerenciamento, o qual estabelece os itens que são controlados, os recursos e o nível de

especialização do pessoal envolvido, as tarefas a serem executadas, as funções e as responsabilidades de cada um, os processos e as ferramentas para Gerenciamento de Configuração que serão utilizadas, assim como metodologias, padrões e procedimentos a serem adotados. Esclarece, ainda, que o controle de configuração mantém a integridade pela facilitação de mudanças aprovadas, prevenindo a incorporação de alterações não aprovadas aos itens que estão sob o controle. As auditorias independentes da configuração deverão avaliar a evolução do produto para garantir a conformidade com as especificações, as políticas e os acordos contratuais, e deverão ser realizadas como suportes formais nas revisões consideradas marcos de decisões.

Explica que uma alteração sobre a configuração corrente deve ser feita através de uma ECP e que esta pode ocorrer sob diversas maneiras, tais como:

- a) uma mudança em requisito, ou escopo, solicitada por um usuário;
- b) de um fornecedor, por um avanço inesperado em alguma tecnologia;
- c) um fornecedor que identifica uma necessidade de mudança no sistema em desenvolvimento.

Como justificativa esclarece que circunstâncias como estas, com potencial considerável para alterar o escopo do projeto, são motivos razoáveis para propor uma ECP e para conduzir uma análise para o entendimento dos efeitos desta nos planos, nos custos e nos cronogramas.

Afirma que uma ECP deve ser aprovada antes de posta em prática e que deve ser analisada, também, sob a perspectiva da não implantação, principalmente em sistemas imaturos, visto que alterações mais tardias podem produzir impactos maiores.

Como resultados mais desejáveis de uma ECP foram elencados:

- a) uma funcionalidade do sistema alterada para atender uma requisição de mudança;

- b) nova tecnologia ou novo produto que expande as capacidades do sistema além do inicialmente desejado pelo cliente;
- c) possibilidade de redução de custos de desenvolvimento, de utilização, ou de suporte;
- d) aprimoramento da disponibilidade, ou confiabilidade, do sistema.

Destaca ainda que é possível observar que os resultados listados nos itens c e d podem reduzir o custo do ciclo de vida e, potencialmente, trazer maior retorno.

As ECPs e ENs ajudam a garantir que o sistema evolua de modo a continuar satisfazendo os requisitos operacionais e os objetivos definidos e que todas as alterações sejam de conhecimento de todas as partes interessadas.

O tópico sobre a abordagem do Gerenciamento de Configuração afirma que este processo mantém controle sobre:

- a) requisitos;
- b) especificações;
- c) documentos de definição de configuração;
- d) controle de configuração;
- e) contabilidade de *status* de configuração.
- f) auditoria de configuração física e funcional.

Na inevitabilidade de alterações estas devem ter suas necessidades confirmadas e a solução apresentada deve ser justificada como sendo a de melhor custo-benefício, dentre as possíveis.

Como abordagem para o Gerenciamento de Configuração sugere executar instruções técnicas e administrativas, vigilância e fornecer serviços para:

- a) identificar e documentar as características físicas e funcionais de cada item de configuração (IC), de forma que ele seja único e acessível;
- b) atribuir um identificador único para cada versão de cada IC;
- c) estabelecer controles que permitam alterações nestas características;
- d) estar em concordância com a versão do produto e garantir produtos consistentes, através de *baselines* destes produtos;
- e) registrar, rastrear e reportar o processamento da alteração e o *status* da implementação e coletar medidas pertinentes às requisições de mudanças e problemas com as *baselines* de produtos;
- f) manter uma rastreabilidade abrangente para todas as transações.

Ressalta que a profundidade da análise a ser efetuada sobre a mudança proposta esteja de acordo com a natureza das mudanças, pois a classificação da mudança é base para o controle de configuração. Assim, cita que algumas vezes as mudanças são classificadas em classe 1, que é uma mudança significativa, envolvendo custos, prazos e desempenho técnico e, portanto, requer a aprovação de *stakeholders* antes da implementação. Já a mudança de classe 2 seria uma mudança menor, afetando erros de documentação ou detalhes internos do projeto e não demandando a aprovação do cliente.

Declara que a implementação de um CCB deve ser implementada no início do projeto e é apontada como ponto central para coordenação, revisão e aprovação de todas as mudanças nas *baselines*, sendo composta por membros de várias áreas, incluindo a Engenharia de Sistemas, engenharia de *hardware* e software, gerenciamento do projeto, qualidade do produto e Gerenciamento de Configuração.

Outro ponto interessante abordado é a contabilidade de *status* de configuração que é um subprocesso para registro e emissão de relatório de informações importantes para o gerenciamento, assim como a situação de documentação

aprovada de características físicas e funcionais e o estado de mudanças propostas e aprovadas. Sugere uma série de possíveis medidas a serem consideradas como:

- a) número de propostas processadas, adotadas, rejeitadas e abertas;
- b) o estado das mudanças propostas abertas;
- c) sumário da classificação das mudanças solicitadas;
- d) número de desvios e *waivers* por item de configuração;
- e) número de relatórios de problemas abertos, concluídos e em andamento;
- f) relatórios de complexidade de problemas e a causa raiz;
- g) o esforço associado com a resolução de problemas e estágio de verificação, quando o problema é identificado;
- h) tempo de processamento e esforço para desvios e *waivers*, ECPs, relatórios de problemas, dentre outros;
- i) atividades causando um número significativo de solicitações de mudanças e a taxa de mudanças nas *baselines*.

Por fim, cita as auditorias de configuração, que são executadas de forma independente pelo Gerenciamento de Configuração e pela Garantia do Produto para avaliar a evolução do produto e garantir a conformidade com as especificações, políticas e acordos contratuais,

3.4. Abordagem IEEE

A abordagem da IEEE analisada refere-se à área de software, proposta pelo seu “*Guide to the software Engineering Body of Knowledge*” (2014), que utiliza como base a norma ISO/IEC 12207:2008 (IEEE 12207-2008). Divide os

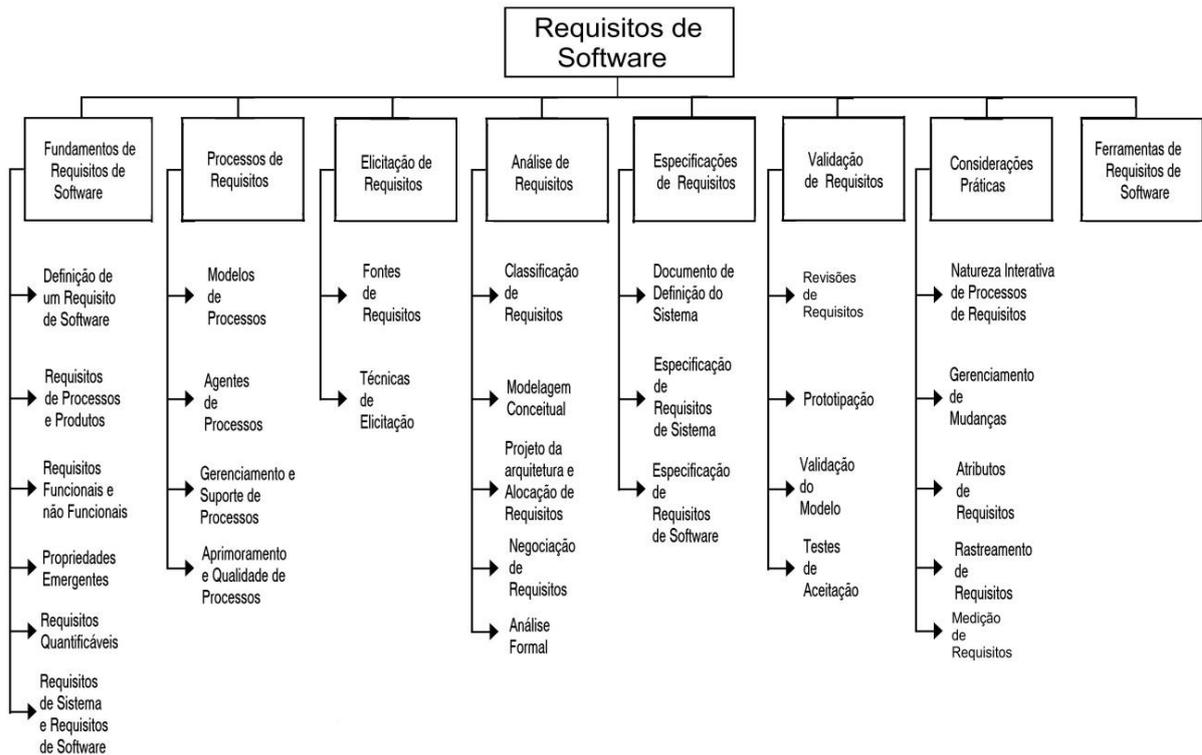
tópicos em quinze áreas de conhecimento (KA), em que cada área é um capítulo do guia.

As duas áreas do guia analisadas para a composição desta pesquisa são: área de conhecimento de requisitos de software, presente no capítulo 1 e área de conhecimento de Gerenciamento de Configuração de Software, presente no capítulo 6.

O capítulo 1, relativo à área de conhecimento de requisitos de software, trata a elicitação, a análise, a especificação e a validação de requisitos de software. Este capítulo possui, também, um tópico que aborda considerações práticas inerentes ao gerenciamento de mudanças, tratado como ponto central para o gerenciamento de requisitos devendo ser executado durante todo o ciclo de vida do produto de software.

Propõe a decomposição de uma estrutura, apresentada pela Figura 3.10, baseada em processos que refletem o fato de que o gerenciamento de requisitos, para ser bem sucedido, deverá ser considerado como um processo que envolve atividades complexas, fortemente acopladas, tanto sequenciais, quanto concorrentes e não apenas como atividades únicas, executadas no início do projeto de desenvolvimento de software. Assim, é um processo iniciado no ponto de partida do projeto e que continua em constante refinamento através do ciclo de vida. Além disso, destaca que o processo deve ser adaptado para cada organização, e contexto, em que o projeto se insere (IEEE COMPUTER SOCIETY, 2014).

Figura 3.10 - Decomposição de tópicos para Requisitos de Software – IEEE.



Fonte: Adaptado de IEEE Computer Society (2004, p.1-2).

De um modo geral, o enfoque maior do capítulo 1 é o desenvolvimento de requisitos. Portanto, existem muitas informações relativas aos processos das fases de elicitação, classificação, análise e validação de requisitos.

O capítulo 6, área de conhecimento de Gerenciamento de Configuração de software, cita a importância do gerenciamento de mudanças, orientando quanto à garantia de que mudanças propostas devem passar por um processo rigoroso de revisão e de aprovação. Ressalta, ainda, que deve ser aplicado um criterioso controle da rastreabilidade de requisitos, de análise de impacto destas mudanças e do Gerenciamento de Configuração, colocando o gerenciamento de mudanças (CM) como ponto central para o Gerenciamento de Requisitos.

No tópico de rastreabilidade, destaca a importância da rastreabilidade para a realização da análise de impacto, ressaltando que a falha na atualização da

rastreabilidade, durante a ocorrência de mudanças, torna as informações não confiáveis para serem utilizadas na análise de impactos destas mudanças.

Quanto ao aspecto do uso de ferramentas de software para gerenciamento de requisitos, classifica-as, basicamente, em duas categorias: ferramentas para modelagem e ferramentas para gerenciamento de requisitos. Quanto ao gerenciamento de requisitos lembra que atende a uma gama de atividades, dentre elas, documentação, rastreabilidade, Gerenciamento de Configuração e que estas duas últimas somente são praticáveis se suportadas pelo uso de uma ferramenta de software.

Como consideração prática, defende a utilidade da realização de medição de requisitos, exemplificando com um número relativo ao “volume” de requisitos, que pode ser útil para avaliação do “tamanho” da mudança nos requisitos e na estimativa de custo das tarefas de desenvolvimento e manutenção.

Ainda considerando a área de conhecimento de Gerenciamento de Configuração de software, são tratados os aspectos de Gerenciamento de Configuração através do detalhamento dos tópicos: processo de gerenciamento de mudanças, identificação de configuração de software, controle de configuração de software, contabilidade de *status* de configuração, auditoria de configuração de software, entrega e gerenciamento de versões de software e ferramentas para Gerenciamento de Configuração de Software.

Define o Gerenciamento de Configuração como a execução do processo de identificação da configuração do sistema, em pontos distintos no tempo, com o objetivo de, sistematicamente, controlar mudanças sobre a configuração e manter a integridade e rastreabilidade ao longo do ciclo de vida do sistema.

Discorre sobre o processo de requisição, avaliação e aprovação de mudanças no software e apresenta um fluxo para o controle de mudanças, conforme mostra a Figura 3.11, a seguir:

Figura 3.11 - Fluxo do Processo de Controle de Mudanças - IEEE.



Fonte: Adaptado de IEEE Computer Society (2004, p.6-9)..

Finalmente este capítulo discute, brevemente, aspectos de ferramentas de Gerenciamento de Configuração, fornecendo uma classificação para elas, conforme segue:

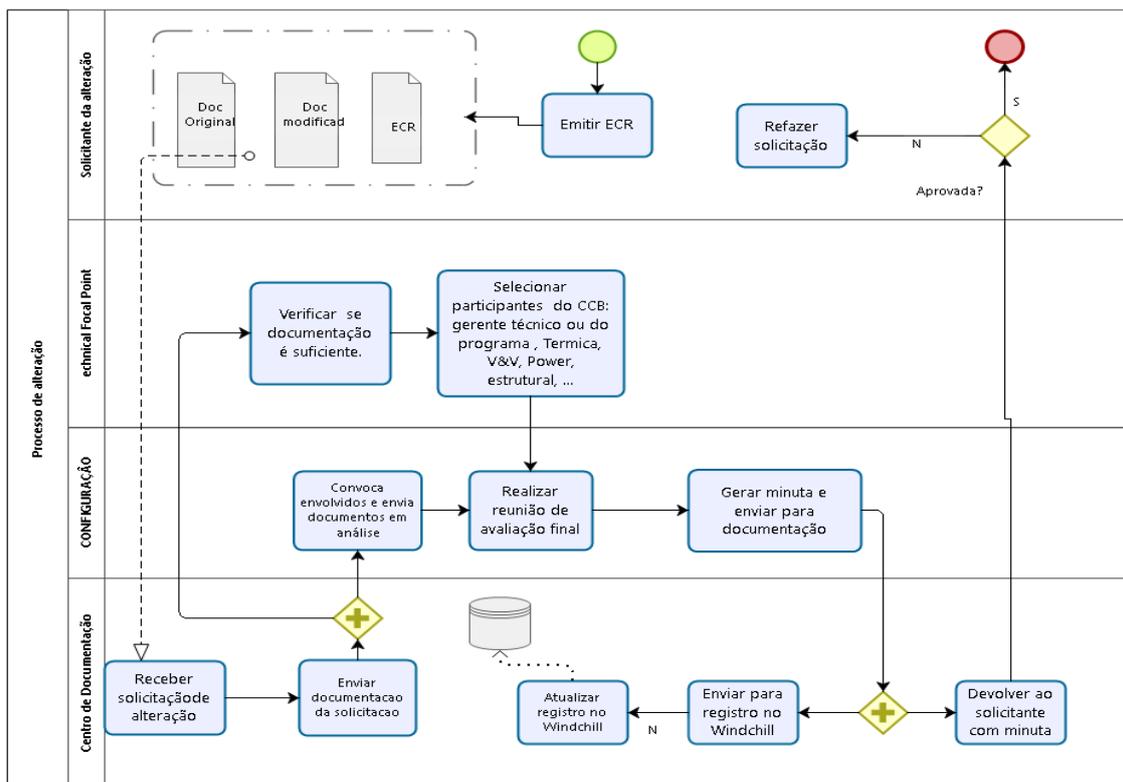
- a) ferramentas de suporte individual: adequadas para pequenas empresas ou grupos de desenvolvimento que não possuam muitas variantes de produtos e que oferecem: ferramentas de controle de versão, ferramentas para a gestão de software executável e ferramentas de controle de mudança.
- b) ferramentas para suporte aos projetos: tipicamente voltadas para ambientes de desenvolvimento distribuído.

c) ferramentas para suporte aos processos corporativos: com capacidade para automatizar parte dos processos empresariais e fornecer suporte para gerenciamento de fluxos de trabalho, funções e responsabilidades. Abrangem processos mais formais incluindo certificação de requisitos (IEEE COMPUTER SOCIETY, 2014).

3.5. Considerações sobre o INPE e o IAE e a comparação de abordagens apresentadas

Considerando que a demonstração do modelo será realizada no âmbito do INPE, é importante mencionar que as atividades de gerenciamento de mudanças de engenharia adotadas pelo INPE obedecem em grande parte o modelo da ECSS. A Figura 3.12, apresentada a seguir, mostra o workflow das atividades praticadas pelo INPE, identificado a partir de informações fornecidas por profissionais que atuam no processo local.

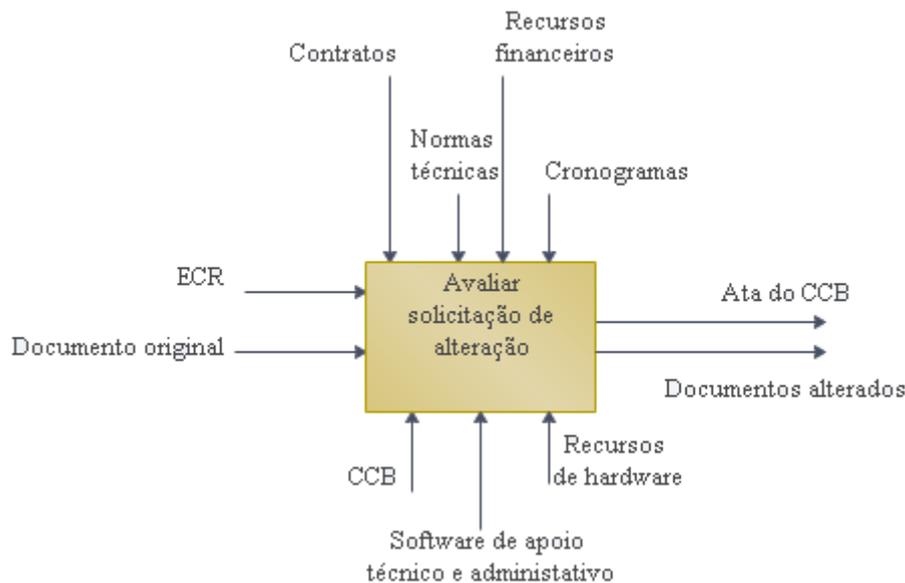
Figura 3.12 – Workflow do processo de mudanças de engenharia no INPE.



Fonte: Elaborada pelo autor

O processo de mudanças de engenharia consiste de entradas na forma de documentos relativos à mudança e que são submetidos a um processo de análise, culminando em saídas na forma de uma ata, com a aprovação ou rejeição, o novo documento da parte modificada e notificação às partes, conforme o diagrama IDEF0 representado na Figura 3.13, a seguir.

Figura 3.13 - Processo para mudanças de engenharia - INPE.



Fonte: Elaborada pelo autor.

No INPE, o controle de mudanças baseia-se na versão de documentos, usando o software PTC *Windchill* como ferramenta de suporte.

No processo atual, as mudanças solicitadas partem da emissão de uma ECR que, juntamente com o documento original que contém a última versão do item a ser alterado, e o documento modificado já com a proposta de alteração incorporada servem como base para a avaliação por parte dos setores envolvidos e para a decisão do CCB. Finalmente, a minuta do documento de análise juntamente com os anexos são enviados para o setor de Controle de Documentação, que providencia a atualização da versão dos documentos no PTC *Windchill* e emite a notificação para as partes interessadas.

É importante esclarecer que o CCB é composto por membros representantes das diversas áreas técnicas nos moldes adotados pelo CMMI para os

processos de engenharia. Tais especializações envolvem conhecimento em gerenciamento e desenvolvimento de requisitos, soluções técnicas, integração de produto, verificação e validação, além de conhecimentos específicos sobre métodos, ferramentas e procedimentos.

É importante salientar que a abordagem adotada pelo INPE baseia-se, em grande parte, nas normas ECSS. Isto garante que todos os seus processos estão incluídos na abordagem proposta por esta pesquisa. Notadamente, a avaliação do detalhamento de proposta de mudanças e documentos afetados, controle de documentação, baseline de documentos, análise de impacto pelo CCB, processo de aprovação e notificação às partes.

Finalizando, cabe ressaltar que, com relação às práticas no IAE, não foi possível identificar um padrão seguido. Cada divisão segue o seu próprio modelo conforme suas necessidades. Desta forma há prejuízo para a sua identificação e representação gráfica.

3.6. Comparação das abordagens apresentadas

A Tabela 3.1, a seguir, apresenta uma comparação entre os modelos NASA, INCOSE, IEEE e ECSS, permitindo a visualização das atividades equivalentes, ou ausentes, em níveis de detalhamentos diferenciados nos quatro documentos.

Na abordagem da NASA, as atividades de gerenciamento de requisitos são tratadas como um processo específico, considerando a *baseline* como entrada para o gerenciamento, em abordagem semelhante à visão de Wiegers e Beaty (2013).

O guia do INCOSE disponibiliza um material detalhado sobre a Engenharia de Requisitos, apresentando capítulos exclusivos para tratar os processos de gerenciamento de requisitos. Tais processos são descritos em capítulos separados, abordando o desenvolvimento de requisitos no capítulo de

processos técnicos e o gerenciamento destes no capítulo de processos de projeto.

Da mesma forma, a documentação do IEEE que é voltada, especificamente, para software, apresenta um capítulo para tratamento geral de requisitos, onde descreve tarefas de desenvolvimento de requisitos e, também, de gerenciamento de mudanças, embora o foco principal sejam os processos iniciais para o desenvolvimento dos requisitos.

Em um capítulo posterior detalha os aspectos do Gerenciamento de Configuração.

Da análise da documentação da ECSS, foram consideradas de maior relevância para esta pesquisa as seguintes normas:

- a) ECSS-S-ST-00-01C - *Glossary of terms* (2012);
- b) ECSS-E-ST-10C - *System engineering general requirements* (2009);
- c) ECSS-E-ST-10-06C - *Technical requirements specification* (2009);
- d) ECSS-E-ST-10-02C - *Verification* (2009);
- e) ECSS-E-ST-40C - *Software* (2009);
- f) (ECSS-M-ST-40C-4.2.1) - *Configuration and information management* (2009).

Estes documentos apresentam os processos de desenvolvimento e de gerenciamento em normas separadas, com ênfase maior sobre o Gerenciamento de Configuração e informação tratado pela norma ECSS-M-ST-40C-4.2.1, com a verificação de requisitos abordada pela norma específica no tópico ECSS-E-ST-10-02C.

Assim, conclui-se que as três abordagens são parecidas, mas observa-se que a documentação da NASA apresenta um guia com maior facilidade de uso.

Cabe ressaltar que todos os documentos analisados consistem em guias que recomendam o que se deve executar para o efetivo gerenciamento de requisitos em todos os seus aspectos, não explicitando exatamente as instruções de como fazer.

A Tabela 3.1, a seguir, mostra um comparativo com a compilação de informações extraídas dos documentos analisados.

Tabela 1.1 - Comparação de pontos-chaves nas abordagens de: NASA, INCOSE, IEEE, ECSS.

NASA Handbook	INCOSE Handbook v3.2.2	IEEE (Swebok v3.0)	Sistema ECSS
Conceito de Gerenciamento de requisitos.			
<p>Refere-se ao gerenciamento das expectativas do <i>stakeholder</i>, das necessidades dos clientes e dos requisitos técnicos do produto com o objetivo de gerenciar os requisitos identificados de produtos, prover rastreabilidade bidirecional e gerenciar as alterações à <i>baseline</i> de requisitos estabelecida, durante o ciclo de vida do produto.</p>	<p>Gerenciamento de Requisitos trata da coleta, análise e validação de requisitos, com todas as comunicações e negociações envolvidas em trabalhar com pessoas.</p> <p>Elicitar e capturar requisitos, gerar um conceito de operações, definir as capacidades do sistema e objetivos de desempenho e definir requisitos não funcionais.</p>	<p>O gerenciamento de requisitos é o processo de priorização, análise, aprovação e elaboração de <i>baseline</i> de requisitos.</p>	<p>A Engenharia de Requisitos consiste na análise e validação de requisitos, alocação de requisitos e manutenção dos requisitos (ECSS-E-ST-10C)</p> <p>A Engenharia de Requisitos deve assegurar:</p> <ul style="list-style-type: none"> a) correta interpretação das necessidades dos clientes e restrições relativas a requisitos técnicos para um produto que satisfaça as necessidades dos clientes, produzido, consolidado e acordado com o cliente; b) geração, controle e manutenção de um conjunto apropriado e coerente de especificações de sistema e de

(Continua)

Tabela 3.1 - Continuação

			<p>mais baixo nível;</p> <p>c) rastreabilidade completa dos requisitos dentro do conjunto estabelecido no item b, regressiva e progressiva, até a verificação final de fechamento.</p>
NASA Handbook	INCOSE Handbook v3.2.2	IEEE (Swebok v3.0)	Sistema ECSS
Plano de gerenciamento de requisitos.			
<p>Apresenta como uma das atividades chave para o sucesso do Gerenciamento de Requisitos o estabelecimento de um plano de execução para este gerenciamento.</p> <p>O plano de Gerenciamento de Requisitos deve:</p> <ul style="list-style-type: none"> a) identificar os <i>stakeholders</i>; b) estabelecer um cronograma; c) atribuir, responsabilidades, autoridades e recursos adequados; d) definir o nível de controle dos dados e Gerenciamento da Configuração; e) identificar treinamento necessário. 	<p>Nos processos de planejamento de projeto afirma que um CMP (<i>Configuration Management Plan</i>) deve ser adaptado para atender os procedimentos individuais do projeto para Gerenciamento de Configuração. A saída primária deste processo é a manutenção da <i>baseline</i> de configuração para o sistema e itens do sistema, onde os itens são colocados sob controle formal, como parte do processo de tomada de decisão.</p>	<p>Declara que os processos do Gerenciamento de Configuração de software, para um dado projeto, devem ser consistentes com o contexto organizacional, restrições aplicáveis, diretrizes comuns aceitas e a natureza do projeto, tendo como principais atividades: identificação de configuração do software, controle de configuração do software, Contabilidade de <i>Status</i> de Configuração do software, auditoria da configuração do software e entrega de versões do software. O resultado destas atividades é gravado como SCMP (<i>Software Configuration Management Plan</i>).</p>	<p>Apresenta plano de Gerenciamento de Configuração cujo objetivo é definir o processo e os recursos para gerenciar a configuração do produto de um modo controlado e rastreável, durante ciclo de vida do produto.</p> <p>(ECSS-M-ST-40C)</p>

(Continua)

Tabela 3.1 - Continuação

NASA Handbook	INCOSE Handbook v3.2.2	IEEE (Swebok v3.0)	Sistema ECSS
Elicitação			
<p>A tarefa primordial em definir as expectativas dos <i>stakeholders</i> é entender os objetivos da missão. Estes objetivos formam a base para desenvolvimento da missão e, assim, devem ser claramente definidos e articulados.</p> <p>A definição dos objetivos é efetuada através da elicitación das necessidades, desejos, capacidades, interfaces externas, premissas e restrições dos <i>stakeholders</i>.</p> <p>O <i>ConOps</i> é um componente importante na captura de expectativas dos <i>stakeholders</i> e requisitos, e no projeto da arquitetura.</p>	<p>Atividades incluídas no processo de definição de requisitos de <i>stakeholders</i>, dentre outras:</p> <p>a) elicitar requisitos de <i>stakeholders</i>: identificar os <i>stakeholders</i> que terão interesse no sistema, através de todo o seu ciclo de vida; elicitar os requisitos- O que o sistema deverá executar, e com que qualidade.</p> <p>b) definir requisitos de <i>stakeholders</i>.</p> <p>c) analisar e manter os requisitos de <i>stakeholders</i>.</p>	<p>Apresenta tópico específico sobre elicitación de requisitos com foco em:</p> <p>a) identificação de fontes.</p> <p>b) aplicação de técnicas de elicitación.</p>	<p>A Engenharia de Sistemas da organização deve analisar os requisitos para o sistema emitidos pelo cliente.</p> <p>A Engenharia de Sistemas da organização deve: gerar, derivar, controlar e manter um conjunto de requisitos para elementos de baixo nível, definindo seu conceito e restrições operacionais e parâmetros de funcionalidade e a verificação necessária para atender os requisitos de sistema emitidos pelo cliente. (ECSS-E-ST-10C)</p>
Análise de requisitos			
<p>Análise funcional é utilizada para delinear tanto os requisitos funcionais como os de desempenho. Os requisitos são divididos em grupos baseados em critérios estabelecidos (i.e. funcionalidade similar, desempenho, acoplamento) para facilitar e focar a análise.</p> <p>Requisitos funcionais e de desempenho são alocados para partições funcionais e subfunções, objetos, pessoas ou processos.</p> <p>Devem ser organizados</p>	<p>O objetivo do processo de análise de requisitos é transformar a visão direcionada a requisito do <i>stakeholder</i> de um serviço desejado, em uma visão técnica do requerido produto, que seja capaz de entregar estes serviços.</p>	<p>A análise de requisitos é efetuada para:</p> <p>a) detectar e resolver conflitos entre requisitos;</p> <p>b) descobrir os limites do software e como ele deve interagir com o seu ambiente organizacional e operacional;</p> <p>c) elaborar os requisitos do sistema para derivar os requisitos de</p>	<p>A Engenharia de Sistemas da organização deve analisar os requisitos formulados pelo cliente. Esta análise propicia transformar estes requisitos em uma solução do sistema. (ECSS-E-ST-10C)</p> <p>A Engenharia de Sistemas da organização deve identificar requisitos que impactam fortemente nos riscos do sistema. (ECSS-E-ST-10C)</p> <p>A Engenharia de Sistemas da organização deve executar a análise dos requisitos, no nível</p>

(Continua)

Tabela 3.1 - Continuação

em uma estrutura hierárquica de árvore.		software. Compreende as fases de classificação, modelagem conceitual, projeto da arquitetura e alocação de requisitos, negociação de requisitos, análise formal.	de profundidade necessário, para identificar elementos impactando nos riscos do sistema (ECSS-E-ST-10C)
NASA Handbook	INCOSE Handbook v3.2.2	IEEE (Swebok v3.0)	Sistema ECSS
Validação			
<p>Validar os requisitos com base em:</p> <ul style="list-style-type: none"> a) expectativas dos <i>stakeholders</i>; b) objetivos da missão e as restrições; c) restrições operacionais; d) critérios de sucesso da missão. <p>Validação de requisitos pode ser dividida em 3 partes:</p> <ul style="list-style-type: none"> a) correção da escrita dos requisitos; b) correção técnica dos requisitos; c) aceitação por parte dos <i>stakeholders</i>. <p>A validação de um produto mostra que o produto cumpre o propósito intencionado, no ambiente intencionado.</p> <p>A validação reflete o documento de <i>ConOps</i>. Testes de validação são conduzidos sob condições realísticas (ou simuladas) no produto final, com o objetivo de determinar a efetividade</p>	<p>A Validação de requisitos é conduzida como parte da elicitação de requisitos, para fornecer garantia antecipada de que os requisitos são os requisitos “certos”, para guiar o processo de desenvolvimento para uma conclusão que satisfaça os <i>stakeholders</i>.</p> <p>É frequentemente baseada na análise de requisitos, exploração da completude e adequação dos requisitos, avaliação de protótipos, simulações, modelos, cenários e maquetes; e pela obtenção de realimentação de clientes, usuários e outros <i>stakeholders</i>.</p> <p>Os objetos da validação são os projetos, protótipos e elementos finais do sistema, assim como a documentação e material de treinamento que descreve o sistema e como utilizá-lo.</p> <p>O processo de validação inclui as seguintes atividades:</p> <ul style="list-style-type: none"> a) planejar a validação; 	<p>Validação de requisitos refere-se ao processo de examinar os requisitos para garantir que eles definem o software certo (i.e, o software que os usuários esperam).</p> <p>Fases da validação:</p> <ul style="list-style-type: none"> a) revisão dos requisitos; b) prototipação; c) validação do modelo; d) testes de aceitação. 	<p>Processo que demonstra que um produto está habilitado a cumprir seu uso pretendido no ambiente operacional pretendido. A verificação é um pré-requisito para a validação. (ECSS-S-ST-00-01C)</p> <p>A intenção deste processo é confirmar que os desempenhos e as funções da <i>baseline</i> de requisitos estão corretamente e completamente presentes no produto final. (ECSS-E-ST-40C)</p> <p>Na fase 0, a Engenharia de Sistemas da organização deve validar, junto com o cliente, os requisitos contra as necessidades expressadas. (ECSS-E-ST-10C)</p>

(Continua)

Tabela 3.1 - Continuação

e adequação do produto para o uso em operações da missão.	<ul style="list-style-type: none"> b) desenvolver uma estratégia de validação; c) executar a validação. 		
NASA Handbook	INCOSE Handbook v3.2.2	IEEE (Swebok v3.0)	Sistema ECSS
Verificação			
<p>Deve ser definido um método de verificação para cada requisito.</p> <p>A verificação de um produto mostra a prova de conformidade com os requisitos, que o produto pode atender cada declaração do tipo “<i>shall</i>”, e isto provado com testes, análises, inspeção ou demonstração.</p> <p>Testes de verificação refletem o conjunto de requisitos aprovado e podem ser efetuados em diferentes estágios do ciclo de vida do produto.</p> <p>Apresenta uma matriz de verificação de requisitos como exemplo para sugerir as informações mínimas que devem ser incluídas na matriz de verificação.</p>	<p>O processo de verificação confirma que o sistema de interesse, e todos os seus componentes, executam as suas funções pretendidas e atendem os requisitos de desempenho alocados a eles (i.e, que o sistema foi construído de forma certa). Métodos de verificação incluem inspeção, análise, demonstração e testes.</p> <p>O processo de verificação inclui as seguintes atividades:</p> <ul style="list-style-type: none"> a) planejamento da verificação; <ul style="list-style-type: none"> - programar atividades, confirmar e instalar sistemas facilitadores de verificação. b) executar a verificação; <ul style="list-style-type: none"> -desenvolver procedimentos de verificação, -conduzir atividades de verificação através de procedimentos estabelecidos, para demonstrar a conformidade com os requisitos. -documentar resultados da verificação e inserir dados na RVTM. 	<p>Verificação é um esforço para garantir que o produto é construído da forma correta, no sentido que o produto final de uma atividade atende as especificações impostas sobre ele na atividade prévia.</p>	<p>Processo que demonstra através do provimento de evidência objetiva, que o produto foi projetado e produzido de acordo com as respectivas especificações e desvios e <i>waivers</i> acordados e está livre de defeitos.</p> <p>NOTA: um <i>waiver</i> pode surgir como uma saída do processo de verificação.</p> <p>Um requisito técnico deve ser verificável usando um ou mais métodos de verificação aprovados.</p> <p>NOTA: Um requisito técnico é verificável, quando os meios, para avaliar se a solução proposta cumpre o requisito são conhecidos.</p> <p>(ECSS-E-ST-10-06C)</p> <p>A Engenharia de Sistemas da organização deve assegurar que para cada requisito contido na especificação técnica de requisitos, um ou uma combinação de métodos de verificação seja identificado. (ECSS-E-ST-10C)</p> <p>A Engenharia de Sistemas da organização deve garantir que para cada requisito contido na especificação de requisitos técnicos, os métodos de verificação estejam refletidos na</p>

(Continua)

Tabela 3.1 - Continuação

			<p>matriz de verificação. (ECSS-E-ST-10C)</p> <p>A norma ECSS-E-ST-10-02C estabelece os requisitos para a verificação de produtos de sistemas espaciais.</p> <p>Ela define os conceitos fundamentais para o processo de verificação, os critérios para definir a estratégia de verificação, e especifica os requisitos para a implementação do programa de verificação. Ela inclui também a lista de documentos esperados (i.e, Documento de Definições de Requisitos – DRDs).</p> <p>(ECSS-E-ST-10-02C)</p>
NASA Handbook	INCOSE Handbook v3.2.2	IEEE (Swebok v3.0)	Sistema ECSS
Rastreabilidade			
<p>Apresenta como uma das atividades chave para o sucesso do gerenciamento de requisitos, estabelecer a rastreabilidade bidirecional entre os requisitos.</p> <p>Classifica a falta de rastreabilidade bidirecional como fonte de riscos para o projeto.</p>	<p>Estabelecer e manter uma matriz de rastreabilidade para documentar como os requisitos formais propõem atender e obter a aceitação dos <i>stakeholders</i>.</p> <p>Todos os requisitos do sistema devem ter rastreabilidade bidirecional, incluindo a sua origem, tal como os requisitos originais dos <i>stakeholder</i>.</p> <p>Manter a rastreabilidade dos requisitos desde o começo da atividade de análise de requisitos.</p> <p>É essencial que a</p>	<p>A rastreabilidade é fundamental para a execução de análise de impacto quando ocorrem mudanças nos requisitos. Se as informações de rastreabilidade não são atualizadas quando mudanças ocorrem, tornam-se não confiáveis para a análise de impacto das mudanças.</p> <p>O uso da rastreabilidade tipicamente aprimora o gerenciamento de produtos de software em desenvolvimento, e a qualidade do processo de software; também fornece garantia aos <i>stakeholders</i> que todos os requisitos foram satisfeitos. A rastreabilidade permite a análise de mudanças uma vez que o software é desenvolvido e</p>	<p>Todos os requisitos técnicos devem possuir rastreabilidade regressiva.</p> <p>Todos os requisitos devem possuir rastreabilidade progressiva. (ECSS-E-ST-10-06C)</p> <p>A Engenharia de Sistemas da organização deve garantir a rastreabilidade, progressiva e regressiva, de todos os requisitos:</p> <ol style="list-style-type: none"> para suas fontes; para os requisitos de mais baixo nível; para alterações no projeto que induzam modificações nos requisitos;

(Continua)

Tabela 3.1 - Continuação

	<p>rastreabilidade seja mantida para garantir que todos os requisitos do nível de sistema sejam satisfeitos no projeto resultante.</p> <p>A rastreabilidade não é um objetivo final em si mesma, mas uma ferramenta que pode ser usada para:</p> <ul style="list-style-type: none"> a) aprimorar a integridade e precisão de todos os requisitos; b) permitir rastreamento do desenvolvimento e alocação de requisitos e gerar medições em geral; c) dar suporte à manutenção mais fácil e implementação da mudança do sistema no futuro. 	<p>lançado, visto que os relacionamentos, para os produtos de desenvolvimento do software, podem ser facilmente percorridos para avaliar o impacto da mudança.</p>	<p>d) para o fechamento de verificação.</p> <p>A Engenharia de Sistemas da organização deve estabelecer a matriz de rastreabilidade de requisitos em conformidade com o anexo N da (ECSS-E-ST-10C).</p> <p>O propósito da RTM é auxiliar a verificar que todos os requisitos declarados e derivados estão alocados nos componentes do sistema e outros entregáveis (rastreamento progressivo).</p> <p>A matriz também é usada para determinar a fonte dos requisitos (rastreamento regressivo).</p> <p>Rastreamento de requisitos inclui rastrear qualquer informação que satisfaça os requisitos como capacidades, elementos de projeto e testes.</p> <p>A RTM é usada, também, para garantir que todos os requisitos são encontrados e para localizar componentes do sistema afetados quando há uma mudança nos requisitos, A habilidade para localizar componentes afetados permite que o impacto de mudanças, em requisitos do sistema, seja determinado, facilitando a determinação de custos, benefícios e prazos. (ECSS-E-ST-10C (Annex N))</p>
--	--	--	--

(Continua)

Tabela 3.1 - Continuação

NASA Handbook	INCOSE Handbook v3.2.2	IEEE (Swebok v3.0)	Sistema ECSS
CCB – Configuration Control Board			
<p>Uma vez que os requisitos tenham sido validados e revistos na revisão de requisitos do sistema, passam a estar sob controle formal de configuração, portanto, qualquer mudança nos requisitos deve ser aprovada pelo CCB.</p> <p>Os engenheiros de sistema, gerente do projeto e outros engenheiros importantes participam no processo de aprovação do CCB para avaliar o impacto da mudança, incluindo custos, desempenho, programação e segurança (<i>safety</i>).</p>	<p>Um CCB completo deve ser implementado no início do projeto para fornecer um ponto central para coordenar, revisar, avaliar e aprovar todas as propostas de mudanças da documentação, em <i>baseline</i>, e as mudanças propostas à configuração, em <i>baseline</i>, incluindo software, <i>hardware</i> e <i>firmware</i>.</p> <p>O conselho é composto por membros de várias disciplinas incluindo Engenharia de Sistemas, Engenharia de <i>Hardware</i> e Software, Gerenciamento do Projeto, Qualidade do Produto, e Gerenciamento da Configuração.</p> <p>As mudanças submetidas à jurisdição do conselho de revisão devem ser avaliadas quanto à necessidade técnica, conformidade com requisitos do projeto, compatibilidade com documentos associados e impactos no projeto.</p>	<p>A autoridade para aceitação, ou rejeição, de uma mudança apoia-se em uma entidade tipicamente conhecida como CCB.</p> <p>Pode haver múltiplos níveis de autoridades para mudanças que dependem de uma variedade de critérios, como a criticidade do item envolvido, da natureza da mudança (por exemplo, impacto em custo e prazos) ou do estágio atual do projeto no ciclo de vida. A composição do CCB, para um dado sistema, varia dependendo destes critérios. Todos os <i>stakeholders</i> apropriados para o nível do CCB devem estar representados.</p>	<p>O CCB é estabelecido a cada nível do projeto como a autoridade relevante para todas as mudanças.</p> <p>O CCB é constituído por representantes permanentes de todos os programas ou disciplinas do projeto, necessárias para a revisão e avaliação das mudanças.</p> <p>Os membros do CCB têm a autoridade para tomada de decisões.</p> <p>Uma mudança a partir de um cliente, pode somente ser implementada após a avaliação e aprovação da solicitação.</p>
NASA Handbook	INCOSE Handbook v3.2.2	IEEE (Swebok v3.0)	Sistema ECSS
Baseline			
<p>As especificações aprovadas, os desenhos, listas de partes, e outros documentos de configuração, estabelecem a <i>baseline</i> daquele produto, que pode ser modificada mais</p>	<p>O resultado do processo de análise de requisitos deve ser uma <i>baseline</i> de requisitos de sistema, completos, precisos, não ambíguos, armazenados no banco de dados de requisitos, acessível a</p>	<p>Uma <i>baseline</i> de software é uma versão formalmente aprovada de um item de configuração (não importando a mídia), que é formalmente atribuída, e fixada, em um</p>	<p>A Engenharia de Sistemas da organização deve estabelecer a lista de documentos que constituem a <i>baseline</i> de requisitos do sistema em contribuição às <i>baselines</i> de configuração.(ECSS-</p>

(Continua)

Tabela 3.1 - Continuação

<p>tarde.</p> <p>Sem uma <i>baseline</i> verificada e os controles de configuração apropriados, modificações posteriores poderão ser custosas ou causar problemas grandes problemas de desempenho.</p>	<p>todas as partes e documentados em uma especificação do sistema aprovada e publicada. As seguintes medidas são frequentemente utilizadas para medir o progresso e a completude da atividade de análise de requisitos.</p> <ul style="list-style-type: none"> a) número ou percentual de requisitos definidos, alocados e rastreados; b) tempo para lançamento do <i>draft</i>; c) número de reuniões realizadas; d) número e tendências de requisitos TBD, TBR, TBS; e) número de problemas de requisitos identificados (i.e, requisitos não declarados em um modo verificável); f) número e frequência de mudanças (inclusões, modificações e exclusões). 	<p>determinado momento do ciclo de vida do Item de Configuração.</p> <p>O termo, também, é usado para referenciar uma versão particular de um Item de Configuração de software que recebeu aceitação. Em qualquer um dos casos, a <i>baseline</i> pode somente ser alterada através de um procedimento formal para controle de mudanças. Uma <i>baseline</i> em conjunto com todas as mudanças aprovadas à <i>baseline</i> representa a configuração corrente.</p>	<p>E-ST-10C)</p> <p><i>Baselines</i> de configuração representam os estados aprovados de requisitos nos principais (Fig. 11 na norma) marcos do programa ou projeto e fornece um ponto de partida para evoluções posteriores. São aplicáveis a <i>hardware</i> e software.</p> <p>Uma <i>baseline</i> de configuração comporta a documentação que descreve as características de um produto.</p> <p>A documentação é formalmente designada como a referência da configuração em um ponto chave no ciclo de vida do produto, constituindo um evento principal na definição do produto. Qualquer alteração subsequente às características do produto, proposta para a sua documentação, está sujeita a um procedimento formal envolvendo todos os atores e disciplinas inerentes, antes de ser incorporada. (ECSS-M-ST-40C)</p>
<p>NASA Handbook</p>	<p>INCOSE Handbook v3.2.2</p>	<p>IEEE (Swebok v3.0)</p>	<p>Sistema ECSS</p>
<p>Gerenciamento de mudanças nos requisitos (Change Management - CM)</p>			
<p>O CM é uma disciplina de gerenciamento aplicada sobre todo o ciclo de vida do produto, para prover visibilidade e controlar mudanças nas características físicas e</p>	<p>Implementar um ciclo de controle de configuração que incorpore avaliação, aprovação, validação e verificação de ECRs (<i>Engineering Change Requests</i>).</p>	<p>O processo de requisição de mudança em software fornece procedimentos formais para a submissão e registro de requisições de mudanças, avaliação e</p>	<p>Define Gerenciamento de Configuração como o processo para estabelecer e manter registro, consistente, das características físicas e funcionais de um produto</p>

(Continua)

Tabela 3.1 - Continuação

<p>funcionais. O CM garante que:</p> <ul style="list-style-type: none"> a) a configuração do produto é conhecida e refletida na documentação do produto; b) qualquer mudança no produto é benéfica e é executada sem consequências adversas; c) as mudanças são gerenciadas. <p>O processo de Gerenciamento de Requisitos deve estar preparado para lidar com as requisições de mudanças nos requisitos, que poderão ser geradas a qualquer tempo durante o ciclo de vida do projeto.</p> <p>Apresenta como uma das atividades chave para o sucesso do gerenciamento de requisitos:</p> <ul style="list-style-type: none"> a) avaliar todas as solicitações de mudanças na <i>baseline</i> de requisitos durante o ciclo de vida do projeto e fazer as alterações, em caso de aprovação pelo conselho de mudança. <p>Apresenta, ainda, como questão chave para o sucesso do gerenciamento de requisitos:</p> <ul style="list-style-type: none"> a) o gerenciamento efetivo de mudanças requer um processo que 	<p>Identificar elementos do sistema a serem mantidos sob controle de configuração.</p> <p>Estabelecer e controlar <i>baselines</i> de configurações durante todo o ciclo de vida do sistema.</p> <p>Desenvolver e manter documentação de controle de configuração e comunicar o <i>status</i> dos itens controlados para a equipe de projeto.</p> <p>Executar auditorias associadas com pontos de controle para validar as <i>baselines</i>.</p> <p>Mudanças podem ser categorizadas em classes em função do seu impacto.</p> <p>Implementação de um CCB no início do projeto para fornecer um ponto central de coordenação.</p> <p>Requisições de desvios/<i>waivers</i> para requerer e documentar desvios temporários de requisitos de identificação de configuração, quando mudanças permanentes, para efeito de conformidade com <i>baselines</i> estabelecidas, não são aceitáveis.</p>	<p>potencial custo e impacto da mudança proposta e a aceitação, modificação adiamento ou rejeição da mudança proposta.</p> <p>A autoridade para aceitar ou rejeitar propostas de mudanças é do CCB (<i>Configuration Control Board</i>).</p> <p>Um processo efetivo para requisições de mudanças em software requer o uso de procedimentos e ferramentas de suporte para originar estas requisições, forçar a aplicação do fluxo definido para o processo, capturar as decisões do CCB e reportar informações do processo de mudanças. Um elo entre esta capacidade da ferramenta e o sistema de relato de problemas pode facilitar o rastreamento de soluções para problemas reportados.</p> <p>O uso de Contabilidade de <i>Status</i> de Configuração pode ser usado como base para diversas medidas.</p> <p>Verificar e identificar a Volatilidade / estabilidade. É útil poder estimar a probabilidade de que um requisito irá mudar durante o ciclo de vida do software.</p>	<p>em comparação ao definido em seu projeto, e os seus requisitos operacionais e que devem ser aplicados durante todo o ciclo de vida. Este processo permite:</p> <ul style="list-style-type: none"> a) conhecer, a qualquer tempo, descrição técnica de um produto a partir de documento aprovado; b) registrar e controlar a evolução da descrição técnica de produtos do sistema; c) fornecer a rastreabilidade da evolução de descrição técnica de um produto; d) garantir a consistência das interfaces internas; e) verificar e demonstrar para todos os atores envolvidos, que a documentação representa uma descrição fiel do produto documentado; f) identificar a <i>baseline</i> corrente de configuração, e a configuração <i>as-built</i> de um produto, para registrar as discrepâncias identificadas durante a produção, entrega ou a operação e distribuição, para posterior uso;
--	---	--	---

(Continua)

Tabela 3.1 - Continuação

<p>avaliar o impacto das mudanças propostas, antes da aprovação e implementação da mudança. Para isto uma <i>baseline</i> de configuração deve ser documentada e ferramentas utilizadas para avaliar os impactos na <i>baseline</i>.</p> <p>O processo de gerenciamento de mudanças deve comunicar a decisão de mudança nos requisitos a todas as organizações afetadas.</p> <p>Medir a funcionalidade de cada mudança em requisito e avaliar o seu impacto sobre o resto do sistema; comparar com as consequências de não aprovação da mudança proposta.</p>			<p>g) dar conhecimento a qualquer ator sobre as limitações e possibilidades operacionais de cada item do produto e da existência de não conformidades e seus efeitos sobre outros itens. (ECSS-M-ST-40C)</p> <p>Cada requisito técnico deve estar sob controle de configuração. (ECSS-E-ST-10-06C).</p>
NASA Handbook	INCOSE Handbook v3.2.2	IEEE (Swebok v3.0)	Sistema ECSS
Avaliação de desempenho			
<p>Dados de medições de desempenho são utilizados para avaliar custos do projeto, cronogramas e desempenho técnico e seus impactos no custo total e cronograma do projeto.</p> <p>MOEs são a medida “operacional” do sucesso, que é fortemente relacionado com o cumprimento da missão ou objetivos operacionais, no ambiente operacional pretendido. São focadas em quão bem a missão</p>	<p>Como declarado na ISO/IEC 15288:2008:</p> <p>O propósito do processo de medições é coletar, analisar e reportar dados, relacionados aos produtos desenvolvidos e processos implementados dentro da organização, para dar suporte ao efetivo gerenciamento dos processos e para, objetivamente, demonstrar a qualidade dos produtos.</p> <p>Medidas fornecem informação objetiva para</p>	<p>Medições no processo de SCM (<i>Software Configuration Management</i>) podem ser projetadas para fornecer informações específicas da evolução do produto, ou um maior conhecimento do funcionamento do processo, para descoberta de oportunidades de melhoria neste. São úteis também para prover uma caracterização do estado atual do processo, assim como uma base para comparação futura.</p>	<p>Não encontrada referencia a métricas ou medições nos documentos quanto a gerenciamento de mudanças ou requisitos.</p>

(Continua)

Tabela 3.1 - Continuação

<p>ou objetivos operacionais foram alcançados e não como foram alcançados, isto é, devem ser independentes de qualquer solução particular.</p>	<p>auxiliar os gerentes de projeto a:</p> <ul style="list-style-type: none"> a) Comunicar-se durante a organização do projeto; b) identificar e resolver problemas de forma precoce; c) efetuar <i>tradeoffs</i> cruciais; d) rastrear objetivos específicos do projeto; e) Defender e justificar decisões. <p>MOEs são a medida “operacional” do sucesso, que é fortemente relacionado com o cumprimento da missão ou objetivos operacionais sendo avaliados, no ambiente operacional pretendido e sob um específico conjunto de condições (i.e, quão bem a solução cumpre o propósito almejado).</p> <p>Medidas de desempenho caracterizam atributos físicos, ou funcionais, relacionados à operação do sistema, medidos ou estimados sob condições de ambiente operacional e/ou teste específicos.</p> <p>As seguintes medidas são frequentemente utilizadas para medir o progresso e a completude da atividade de análise de requisitos.</p> <ul style="list-style-type: none"> a) número ou percentual de requisitos definidos, alocados e rastreados; 	<p>Medição de requisitos é útil para informar o “volume” de requisitos para um produto em particular. Este número é usado para avaliar o tamanho das mudanças nos requisitos e estimar o custo de tarefas de desenvolvimento ou manutenção, ou ser usado simplesmente como denominador em outras medições.</p>	
--	---	--	--

(Continua)

Tabela 3.1 - Continuação

	<p>b) tempo para lançamento do rascunho;</p> <p>c) número reuniões realizadas;</p> <p>d) número e tendências de requisitos TBD, TBR, TBS;</p> <p>e) número de problemas de requisitos identificados (i.e, requisitos não declarados em um modo verificável);</p> <p>f) número e frequência de mudanças (inclusões, modificações, e exclusões).</p> <p>As seguintes medidas são frequentemente usadas avaliar o progresso e conclusão das atividades de alocação e rastreabilidade:</p> <p>a) número e tendência de requisitos no banco de dados;</p> <p>b) número de requisitos TBD, TBR, e TBS.</p> <p>c) número (ou percentual) de requisitos do sistema rastreáveis para cada nível abaixo e o número (ou percentual) de requisitos de baixo nível rastreáveis de volta para os requisitos do sistema.</p> <p>d)</p>		
--	---	--	--

(Continua)

Tabela 3.1 - Continuação

NASA Handbook	INCOSE Handbook v3.2.2	IEEE (Swebok v3.0)	Sistema ECSS
Base de conhecimento (lições aprendidas)			
<p>Lições aprendidas são importantes para processos, projetos e programas futuros, porque eles mostram hipóteses e percepções conclusivas de processos e projetos anteriores. Os profissionais determinam como as lições prévias de processos e atividades impactam os riscos para o projeto atual e implementam aquelas lições aprendidas que aprimoram o projeto e/ou desempenho.</p> <p>Captura de lições aprendidas é função do bom gerenciamento. Muito frequentemente as lições são perdidas porque elas deveriam ser desenvolvidas e gerenciadas durante, através e entre as fases do ciclo de vida. Existe uma tendência a aguardar até a resolução da situação para documentar a lição aprendida, mas resolução do problema no começo é informação valiosa e difícil de ser recriada mais tarde.</p> <p>Gerentes de projeto e a equipe técnica de projeto devem assegurar que as lições aprendidas são registradas no banco de dados da instituição, no final de todas as fases do ciclo de vida, de processos principais de Engenharia de Sistemas, de pontos chave de decisão e de revisões técnicas.</p>	<p>A Engenharia de Sistemas da organização deve planejar e manter a coleta de lições aprendidas em marcos pré-definidos do ciclo de vida do sistema.</p> <p>Para obter os benefícios da captura de lições aprendidas é importante que a organização crie um repositório para elas, na qual várias lições aprendidas, de vários projetos, sejam agrupadas e possam ser disseminadas para projetos novos, ou em andamento.</p>	<p>Lições aprendidas devem ser armazenadas em um banco de dados apropriado.</p>	<p>O centro de testes espacial deve estabelecer e manter um procedimento documentado para o gerenciamento de lições aprendidas, definindo:</p> <ul style="list-style-type: none"> a) o uso de lições aprendidas como entrada para suas atividades futuras; b) uso de resultados de revisões de lições aprendidas para determinar ações apropriadas. (ECSS-Q-ST-20-07C)

(Continua)

Tabela 3.1 - Continuação

NASA Handbook	INCOSE Handbook v3.2.2	IEEE (Swebok v3.0)	Sistema ECSS
Controle de Configuração			
<p>Como no (NASA, 2007) não é dada uma definição formal, apesar de diversas citações ao longo documento, a definição que foi buscada em (NASA, 2008).</p> <p>Controle de Configuração é o gerenciamento de mudanças, ou o controle de mudanças, feitas em um produto, usando um processo de mudança sistemático. É a proposição, justificativa, avaliação, coordenação, disposição e implementação da mudança proposta. Este processo abrange a implementação de todas (e somente) as mudanças aprovadas na configuração de um IC após o estabelecimento da <i>baseline</i> de configuração do IC.</p>	<p>Controle de Configuração estabelece as <i>baselines</i> de configuração e controla as mudanças nessas <i>baselines</i>, durante o ciclo de vida do sistema.</p> <p>O Controle de Configuração gerencia a coleção de itens a serem mantidos na <i>baseline</i> e mantém a integridade de cada IC identificado, facilitando mudanças aprovadas (por ECRs) e prevenindo a incorporação de mudanças não aprovadas na <i>baseline</i>.</p> <p>Deve ser implementado desde a iniciação do projeto.</p>	<p>Controle de Configuração de software está relacionado com o gerenciamento de mudanças durante o ciclo de vida do software. Consiste do processo para determinar que mudanças devem ser implementadas, a autoridade para aprovar certas mudanças, o suporte para a implementação destas mudanças, e o conceito para o desvio formal dos requisitos do projeto assim como os seus <i>waivers</i>.</p> <p>Informações derivadas destas atividades são úteis para medir o tráfego de mudanças e rupturas, assim como aspectos de retrabalho.</p>	<p>Cada requisito técnico deve estar sob Controle de Configuração.</p> <p>Controle de Configuração é uma das atividades do Gerenciamento de Configuração e envolve:</p> <ul style="list-style-type: none"> a) estabelecimento e a manutenção de um processo de controle de mudanças para produtos individuais e sistemas, e suas interfaces internas e externas; b) registro e o Controle da Configuração de um produto, a qualquer tempo durante sua evolução; c) registro das diferentes configurações de um produto; d) definir e manter repositórios, ou bibliotecas de software, onde a versão corrente e o histórico de <i>baselines</i> de software são armazenadas em um ambiente controlado; e) armazenar e manter produtos de software e mídias relevantes, incluindo cópias de segurança em um ambiente controlado

(Continua)

Tabela 3.1 - Continuação

NASA Handbook	INCOSE Handbook v3.2.2	IEEE (Swebok v3.0)	Sistema ECSS
Item de Configuração			
<p>Item de Configuração (IC) é qualquer <i>hardware</i> ou software, ou combinação dos dois, que satisfaz uma função para o usuário final e é projetado para Gerenciamento de Configuração em separado.</p> <p>ICs são, tipicamente, referenciados por um identificador alfanumérico que também serve como base, imutável, para a atribuição de números seriais para identificar, unicamente, unidades individuais de um IC.</p>	<p>Um item de <i>hardware</i>, software, ou composto, em qualquer nível do sistema, hierarquicamente projetado para o Gerenciamento de Configuração. O sistema ou cada um de seus elementos são ICs individuais. Os ICs têm 4 características comuns:</p> <ul style="list-style-type: none"> a) funcionalidade definida; b) substituível como entidade; c) especificação única; d) controle formal da forma, adequação, e função. 	<p>Item de configuração (IC) é um item, ou agregação de <i>hardware</i> e software, ou ambos, que é projetado para ser gerenciado como uma única entidade.</p>	<p>Qualquer nível de produto cujas características físicas e funcionais são registradas de uma forma consistente e recuperável.</p>
NASA Handbook	INCOSE Handbook v3.2.2	IEEE (Swebok v3.0)	Sistema ECSS
Contabilidade de <i>Status</i> de Configuração (CSC)			
<p>A Contabilidade do <i>Status</i> de Configuração (CSC) é o registro e a geração de relatórios dos dados de configuração necessários para gerenciar, de forma efetiva, os Itens de Configuração. Um sistema CSC efetivo fornece de forma precisa, e tempestiva, informações de configuração como:</p> <ul style="list-style-type: none"> a) a documentação de configuração completa, tanto histórica como a corrente, e identificadores 	<p>A Contabilidade do <i>Status</i> de Configuração tem por objetivo desenvolver e manter a documentação de controle de configuração e comunicar os <i>status</i> dos itens controlados para a equipe de projeto.</p> <p>CSC mantém o <i>status</i> da documentação aprovada, que identifica e define as características físicas e funcionais, <i>status</i> das mudanças propostas, e <i>status</i> das mudanças aprovadas.</p> <p>Todas as mudanças autorizadas pelo</p>	<p>A Contabilidade do <i>Status</i> de Configuração de software (CSCS) é um elemento do Gerenciamento de Configuração, que consiste do registro e geração de relatórios de informações, necessárias para gerenciamento efetivo de uma configuração.</p> <p>Várias informações e medidas são necessárias para suportar o processo de CSCS e atender as necessidades de gerenciamento, da geração de relatórios de <i>status</i> de configuração,</p>	<p>A Contabilidade do <i>Status</i> de Configuração compreende a criação e organização da base de dados necessária para realizar o Gerenciamento de Configuração.</p> <p>Fornecer a fonte de informações para dar suporte a todas as atividades e disciplinas do projeto ou programa, através do estabelecimento e manutenção de:</p> <ul style="list-style-type: none"> a) um registro da documentação de configuração aprovada (i.e. conjunto de
			(Continua)

Tabela 3.1 - Continuação

<p>únicos;</p> <p>b) <i>Status</i> de desvios, concessões (<i>waivers</i>) e mudanças propostos, desde o início até a implementação;</p> <p>c) <i>Status</i> e disposição final de ações e discrepâncias, identificadas durante cada auditoria de configuração.</p>	<p>conselho de revisão de configuração (tanto as gerais como as subordinadas) culminam na rastreabilidade abrangente de todas as transações. Pela utilização de <i>status</i> e rastreamento das mudanças no projeto uma mudança gradual do “<i>build to</i>” para o “<i>as-built</i>” é capturada.</p> <p>Medidas sugeridas na Contabilidade de <i>status</i> de Configuração:</p> <p>a) número de mudanças processadas, adotadas, rejeitadas e abertas;</p> <p>b) <i>status</i> de requisições de mudanças abertas;</p> <p>c) Sumário de classificação de requisições de mudanças;</p> <p>d) número de desvios ou <i>waivers</i> por Item de Configuração;</p> <p>e) número de relatórios de problemas abertos, fechados e em progresso;</p> <p>f) complexidade de problemas relatados e a causa raiz;</p> <p>g) esforço associado ao estágio de identificação e à resolução de problemas quando são identificados;</p> <p>h) tempo de</p>	<p>engenharia de software e outras atividades relacionadas. Os tipos de informação disponível incluem a identificação de configurações aprovadas, assim como a identificação e o <i>status</i> corrente de implementação, de mudanças, desvios e <i>waivers</i>.</p>	<p>dados) e os números de identificação correspondentes;</p> <p>b) <i>status</i> de todas as propostas de mudança e de desvios requisitados sobre a configuração estabelecida;</p> <p>c) o <i>status</i> de implementação de desvios e mudanças aprovados;</p> <p>d) a configuração atual de todas as unidades de itens configurados, presentes no inventário operacional.</p> <p>Estes itens em conjunto representam o relatório de <i>status</i> de configuração.</p>
---	--	--	---

(Continua)

Tabela 3.1 - Continuação

	<p>processamento e esforço para desvios, <i>waivers</i>, ECPs, SCNs, ECRs e problemas relatados;</p> <p>i) atividades causando um significativo número de requisições de mudanças e a taxa de mudanças na <i>baseline</i>.</p>		
NASA Handbook	INCOSE Handbook v3.2.2	IEEE (Swebok v3.0)	Sistema ECSS
Classificação de mudanças			
<p>Uma mudança de engenharia é uma iteração na <i>baseline</i> (em rascunho ou estabelecida). As alterações podem ser maiores ou menores.</p> <p>Uma alteração "maior" é uma alteração na documentação de configuração da <i>baseline</i> que tenha um impacto significativo (i.e., requer adaptação de produtos já fornecidos, ou afeta custo, segurança, a especificação de <i>baseline</i>, compatibilidade com produtos de interface, ou operador, ou treinamentos de manutenção).</p> <p>Uma alteração "menor" corrige ou modifica a documentação de configuração ou processos sem impactos na intercambialidade de produtos ou elementos de sistemas, na estrutura do sistema.</p> <p><i>Waiver</i> é um acordo, documentado, intencionalmente</p>	<p>O controle de configuração efetivo requer que a extensão da análise, e a ação de aprovação, para uma mudança de engenharia proposta, estejam de acordo com a natureza da mudança. Assim, a classificação da mudança é uma base primária para o controle de configuração. As mudanças são por vezes categorizadas em duas classes principais:</p> <p>Classe I que são mudanças significativas que afetam custo, cronograma ou desempenho técnico e, normalmente, exigem a aprovação do cliente antes de serem implementadas.</p> <p>Classe II é uma mudança mínima que é, frequentemente, devida a erros em documentação ou detalhes internos de projeto, assim, geralmente, não requer a aprovação do cliente.</p>	<p>A classificação de mudanças não é tratada no documento.</p>	<p>A classificação de mudanças ou desvios define o tipo de aprovação e ciclo de versionamento de acordo com o critério relativo aos impactos no custo, atualização de cronograma, especificação técnica ou outras características técnicas ou contratuais.</p> <p>Cada mudança é classificada como sendo Classe 1 ou Classe 2 e os desvios como sendo significativos ou menores.</p> <p>A mudança ou desvio pode ser reclassificado pelo próximo CCB de nível mais alto.</p> <p>De acordo com seus efeitos, a proposta de mudança, ou requisição de desvio, é processada através de diferentes níveis da organização. O nível apropriado para decidir sua disposição é o nível para o qual os efeitos da mudança, ou desvio, não têm</p>

(Continua)

Tabela 3.1 - Conclusão

<p>liberando um programa, ou projeto, de uma exigência. (Alguns Centros usam desvios, quando anteriores à Implementação, e <i>waivers</i> durante a implementação. <i>Waivers</i> autorizados não constituem uma mudança na <i>baseline</i></p>			<p>repercussão nos compromissos feitos com o cliente. A disposição é, entretanto, transmitida para o cliente para efeito de informação.</p>
NASA Handbook	INCOSE Handbook v3.2.2	IEEE (Swebok v3.0)	Sistema ECSS
Auditoria de configuração			
<p>A condução de Auditorias de Configuração é apresentada, em diagrama de fluxo, como uma das atividades do Gerenciamento de Configuração.</p>	<p>Auditorias de Configuração são realizadas, independentemente, pelo Gerenciamento de Configuração, e pela Garantia do Produto, para avaliar a evolução do produto e garantir conformidade com as especificações, políticas e acordos contratuais. auditorias formais, ou auditorias de configuração física e funcional, são realizadas no encerramento do ciclo de desenvolvimento do produto.</p> <p>O Gerenciamento de Configuração realiza auditorias periódicas no processo, para garantir que o processo de Gerenciamento de Configuração está sendo seguido.</p>	<p>Auditoria de software é um exame, independente, de um produto de trabalho, ou um conjunto de produtos de trabalho, para garantir conformidade com as especificações, padrões, acordos contratuais ou outros critérios. Auditoria de configuração de software determina a extensão para qual um item satisfaz as características físicas e funcionais requeridas.</p>	<p>A Efetividade do sistema de Gerenciamento de Configuração é medida por auditorias, para verificar a correta aplicação dos requisitos de Gerenciamento de Configuração, durante o ciclo de vida do produto, como especificado pelo cliente. As auditorias são conduzidas conforme definido na ECSS-M-ST-10.</p>

Fonte: Elaborada pelo autor a partir de NASA (2007), INCOSE (2011), IEEE Computer Society (2014), European Space Agency- ESA (2008), European Space Agency- ESA (2009), European Space Agency- ESA (2012).

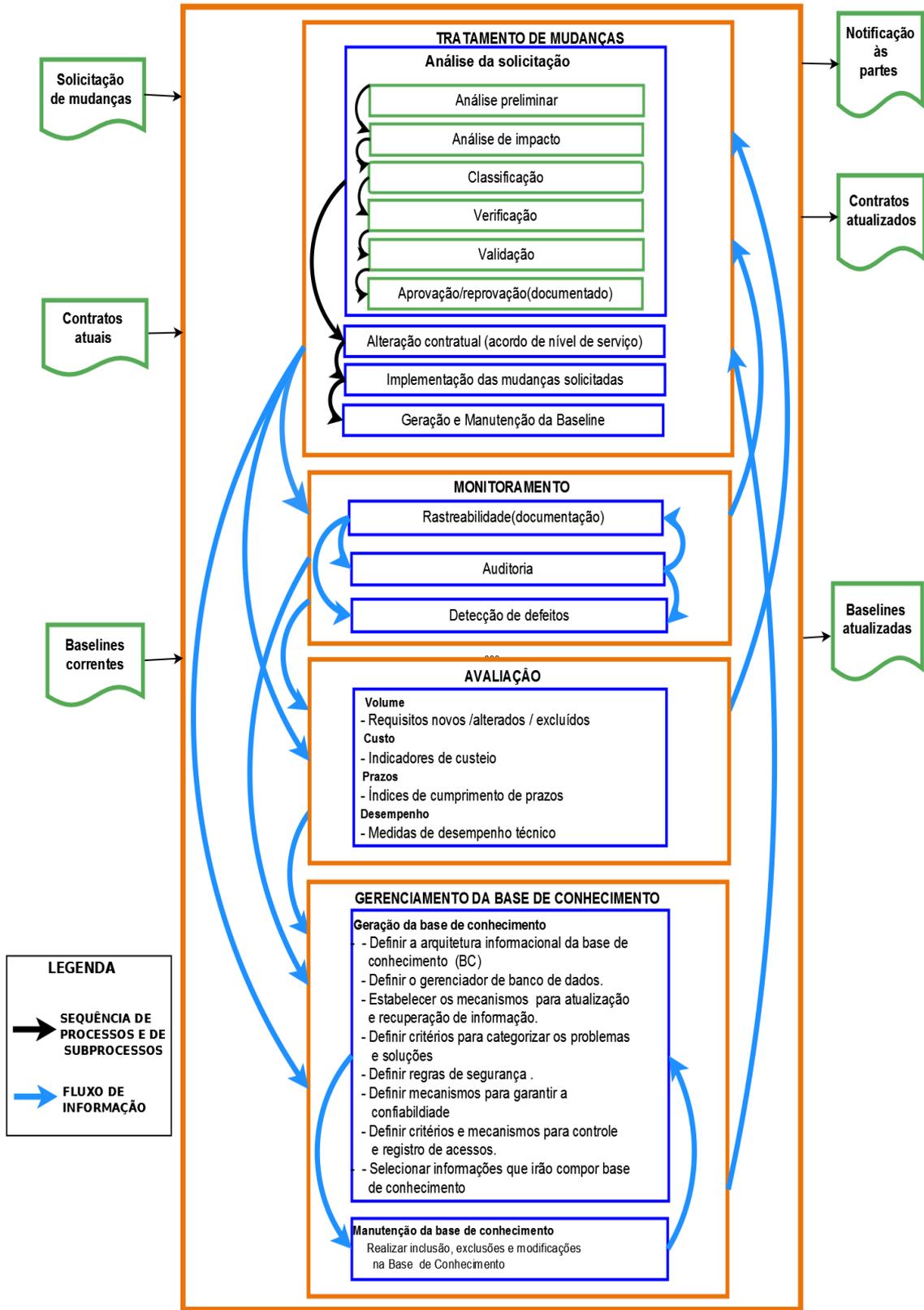
O resultado da reflexão sobre os trabalhos disponibilizados na literatura sobre Gerenciamento de Requisitos, aliado à análise da comparação das abordagens adotadas pela NASA, INCOSE, IEEE e ECSS, serviu de base para a criação do processo proposto para gerenciamento de mudanças de requisitos, descrito no Capítulo 4 desta pesquisa, apresentado a seguir.

4 PROCESSO PROPOSTO

Comparando os quatro processos publicados, buscou-se desenvolver um processo específico que contribuísse para redução das lacunas observadas nas abordagens estudadas, unificando as atividades descritas nos quatro processos no que diz respeito ao gerenciamento de requisitos.

Adotou-se, portanto, para a execução desta pesquisa, o entendimento de que o Gerenciamento dos Requisitos tem seu início após a elaboração de uma *baseline* de missão, não interferindo nos processos de desenvolvimento de requisitos. Assim, o processo de gerenciamento de requisitos concentra-se no conjunto de requisitos já estabelecido em determinado marco do processo, ou seja, na *baseline* de requisitos, que a partir daí deverá ser mantida em face das solicitações de alterações. Sob esta visão é importante ressaltar que o gerenciamento de requisitos deve priorizar e aprovar as mudanças de requisitos, gerenciar tais mudanças e incorporá-las, quando verificadas, validadas e realizadas, à *baseline* de requisitos, a qual deverá se constituir de requisitos validados e rastreáveis ao longo de todo o ciclo de vida do sistema. Sob tal enfoque, obteve-se um processo sustentado pelas atividades inerentes às mudanças, monitoramento e avaliação, conforme pode ser visualizado, graficamente, na Figura 4.1, cuja modelagem foi elaborada com a utilização da ferramenta *DIA Diagram Editor* versão 0.97.2 (LARSSON, 2017).

Figura 4.1 - Processo proposto.



Fonte: Elaborada pelo autor.

O processo de gerenciamento de mudança dos requisitos proposto considera que a mudança solicitada deve ser avaliada através de um processo formal desde o pedido, passando pela justificativa, verificação, análise de impacto, até o aceite e incorporação à *baseline*.

As premissas adotadas para o desenvolvimento deste trabalho consistem na existência de *baselines* de requisitos, bem como de profissionais especializados para a execução das diversas análises técnicas propostas, além de ferramentas de software para dar suporte ao modelo proposto, conforme a definição de processo do CMMI.

Inicialmente propõe-se o estabelecimento de algumas atividades fundamentais para o processo de gerenciamento de alterações de requisitos, sumarizadas a seguir.

O primeiro ponto para o estabelecimento do processo, conforme encontrado em todas as abordagens, é a necessidade de se estabelecer que o processo de aprovação das mudanças seja realizado pelo Conselho de Controle de Configuração (CCB - *Configuration Control Board*).

Conforme citado nos documentos pesquisados, já no início do projeto, o CCB deve ser formalmente estabelecido com integrantes permanentes de todas as áreas de conhecimento envolvidas no projeto, podendo haver a inclusão de participantes adicionais de áreas específicas, quando a solicitação de mudança em análise demandar esta necessidade. Assim, profissionais de engenharia, de administração, de segurança, de tecnologia da informação, de qualidade, de gerenciamento e *stakeholders* afetados devem ser os membros normalmente presentes. Deve ser levado em consideração que a montagem do CCB é guiada por fatores particulares do setor e, este aspecto, aliado ao grau de criticidade, complexidade, fase do projeto e potencial de impacto, pode orientar a subdivisão em diversos níveis para aprovação.

O início do processo de gerenciamento de mudanças ocorrerá através do recebimento de um formulário (físico ou *online*) contendo itens necessários para a inserção no sistema e que servirão de base para a análise da viabilidade, ou não, da alteração proposta. Wiegers e Beaty (2013) sugerem uma relação de itens que deverão compor a solicitação. Tais itens são apresentados na Tabela 4-2.

A coleta de informações pode ser realizada em um estágio inicial por meio da utilização de formulários físicos; no entanto, deve-se considerar que esta prática tem as desvantagens já conhecidas: dentre outras, problemas de volume, de armazenamento e morosidade de circulação; e, por este motivo, tende a ser cada vez menos utilizada. Uma alternativa mais adequada pode ser o uso de arquivo eletrônico, usando um editor de texto como o Microsoft Word, e usando controles Active-X e macros para preenchimento de campos. Esse arquivo eletrônico pode ser, então, compartilhado na rede local, protegido na sua utilização por controles de acesso na forma de credenciais e sistema de permissões baseado nestas credenciais, funcionalidades que são encontradas nos sistemas de arquivos de sistemas operacionais voltados ao uso corporativo. Este modo facilita o processo por permitir a manutenção do arquivo em um repositório específico, onde pode ser acessado para edição, eliminando a necessidade do uso de papel.

Os formulários e roteiros, apresentados nas Figuras 4.4, 4.5, 4.6 e 4.7, sugerem um conjunto de questões a serem respondidas que poderão auxiliar os processos de análises específicas. As respostas poderão ser binárias, atribuindo-se um (1) para respostas positivas, indicando a presença do critério e zero (0) para as respostas negativas, indicando a ausência do critério avaliado. Para a contabilização das respostas sugere-se que esta seja feita por meio de média aritmética ponderada, observando-se o percentual de respostas positivas em relação ao total de questões utilizadas na avaliação; e, logicamente, deverá ser considerada a importância de cada questão para o comprometimento do projeto. Os pesos deverão representar o nível de importância de cada questão em relação ao projeto e deverão ser definidos

pelo analista responsável, uma vez que cada aspecto avaliado pelas questões terá sua importância em função da natureza do projeto e do seu contexto. Desta forma, recomenda-se o uso das questões mencionadas, podendo ser adicionados novos questionamentos dependendo da sensibilidade e conhecimento do analista que estiver realizando a avaliação.

Como exemplo ilustrativo, a Tabela 4-1, a seguir, mostra uma possível ponderação de questões e a obtenção de um resultado quantitativo para a análise preliminar.

Tabela 4-1 - Modelo para suporte de decisão baseado em respostas.

Questão	Peso	Sim (1) / Não (0)	Total
Origem identificada?	2	1	2
Mudança dentro do escopo?	2	1	2
Detalhamento adequado?	2	0	0
Resolve o problema?	3	1	3
Tecnicamente implementável?	3	1	3
Total	12		10
Nível de atendimento aos critérios			83%

Fonte: Elaborada pelo autor.

É importante ressaltar que as análises deverão ser realizadas considerando a aprovação, ou não, da solicitação da mudança. Os impactos e consequências para o projeto deverão ser observados nos dois sentidos e comparados para municiar o tomador de decisão com informações completas e corretas para uma tomada de decisão segura quanto a aceitar ou não a realização da mudança.

Os formulários eletrônicos propostos estão apresentados no capítulo 5 com a aplicação no exemplo demonstrativo.

A proposta para preenchimento dos formulários através de um navegador *web* apresenta a vantagem adicional, dentre outras, de manutenção dos registros

em campos de um banco de dados, os quais podem ser gerenciados de forma mais efetiva e facilitar a obtenção de métricas.

O padrão de formulários para aplicação do modelo proposto, utilizando esta última abordagem, foi desenvolvido utilizando-se o MySQL, que é um sistema de gerenciamento de banco de dados (SGBD) e que utiliza a linguagem SQL (Linguagem de Consulta Estruturada) como interface de acesso para definição de base de dados com os registros do formulários, e PHP/HTML para criação das páginas. Tudo sendo executado sob um servidor de páginas APACHE em plataforma Linux e distribuição UBUNTU na versão 16.04, Os códigos-fonte das páginas mostradas encontram-se no apêndice A.

Para a opção de preenchimento de formulário via Web, a página inicial mostra os *hyperlinks* para acesso aos formulários para preenchimento com as informações das análises e as buscas, conforme pode ser visto na Figura 4.2, a seguir:

Figura 4.2 - Página inicial para acesso aos formulários.



Fonte: Elaborada pelo autor.

O formulário em que serão inseridos os dados de identificação da proposta de mudança é apresentado na Figura 4.3, a seguir.

Figura 4.3 - Página com dados de identificação e detalhes da mudança proposta.

The screenshot shows a web browser window titled 'Mudança_info - Mozilla Firefox'. The address bar shows 'localhost/inpe/form1.php'. The page header includes the INPE logo and the text 'PÓS-GRADUAÇÃO ENGENHARIA E TECNOLOGIA ESPACIAIS'. The main content is a form titled 'Proposta de mudança de requisito'.

Identificação

Tipo de solicitação: Mudança de requisito Desvio Waiver

Número da solicitação: Data de emissão: Solicitante: Projeto:

Status da solicitação

Recebido Em análise Verificado Validado Aprovado Rejeitado Implementado Cancelado Adiado

Detalhamento da solicitação

Título da mudança:

Descrição:

Anexos: Sim Não

Justificativa:

Anexos: Sim Não

Itens afetados:

Anexos: Sim Não

Documentos afetados:

Anexos: Sim Não

[Retorna à página de escolha de formulários](#)

Fonte: Elaborada pelo autor.

A ordem dos procedimentos para realização do tratamento de mudanças pode ser, de forma resumida, visualizada conforme segue:

Análise preliminar da solicitação de alteração → Análise de impactos → Classificação → Verificação → Validação → Aprovação/Reprovação (decisão)

de prosseguimento) → Alteração contratual → Implementação das mudanças solicitadas → Geração e Manutenção da *Baseline*.

O processo proposto para o gerenciamento de mudanças em requisitos é composto por quatro macroprocessos: Tratamento de Mudanças, Monitoramento, Avaliação e Gerenciamento da Base de Conhecimento.

Cada macroprocesso se decompõe em processos e subprocessos, conforme cada aspecto a ser considerado na análise. A seguir serão apresentados os macroprocessos, processos e subprocessos propostos, com a descrição dos principais procedimentos necessários para a execução de cada um.

4.1. Tratamento de mudanças

O tratamento de mudanças é o primeiro dos quatro macroprocessos do modelo gerenciamento de mudanças. É composto pelos seguintes processos:

Processo Análise da solicitação:

Consiste em um conjunto de subprocessos que são responsáveis pelas análises em que se consideram os aspectos mais relevantes, que poderão viabilizar, ou não, o atendimento do pedido. Esta análise é composta por seis subprocessos que tratam da análise preliminar da solicitação, análise dos impactos, classificação, verificação, validação e aprovação/reprovação da solicitação avaliada. O subprocesso que trata a análise de impactos possui um detalhamento em nível mais baixo, que verifica a volatilidade e a estabilidade do item sobre o qual está sendo solicitada a alteração.

Processo de alteração contratual:

Trata da análise e proposição da alteração contratual à luz da aprovação das mudanças solicitadas. Assim, as novas necessidades e definições agregadas, em função da alteração solicitada, deverão ser avaliadas quanto à possível necessidade de atualização e formalização da alteração dos acordos firmados.

Processo de implementação das mudanças solicitadas.

Quando já em fase de desenvolvimento de produto, a alteração aprovada deverá ser implementada. Este processo é totalmente executado pela equipe de desenvolvimento e receberá as instruções para as mudanças após a formalização das alterações contratuais, decorrentes das mudanças solicitadas.

Processo de Geração e Manutenção da *Baseline*

Este processo trata da atualização das *baselines* do projeto e só deve ser acionado quando qualquer mudança for implementada. Como ocorre com a análise de impacto, este processo possui um detalhamento em nível mais baixo ao tratar do Gerenciamento de Configuração e controle de versões responsáveis pela manutenção da *baseline* do projeto, a qual deverá ser atualizada a cada alteração de mudança aprovada.

A partir de *baselines* estabelecidas, as mudanças efetuadas devem ser realizadas através de controle rigoroso, visto que é fundamental para a segurança do projeto que as configurações e o controle de versão de requisitos sejam mantidos sem erros e pronto para suporte às atividades de auditoria.

A seguir serão descritos todos os macroprocessos, processos e subprocessos, com indicativos de como devem ser realizados.

4.1.1. Análise de solicitação

As solicitações de alterações devem ser submetidas a uma série de análises, quando deverão ser classificadas, priorizadas e negociadas. Este Processo inicia o macroprocesso tratamento de mudanças.

Foram acatadas as recomendações de Kotonya e Sommerville (1998) considerando-se que as solicitações de mudanças apoiam-se na detecção de algum problema nos requisitos, podendo este ser originário de uma simples análise do documento de requisitos, de novas necessidades dos stakeholders, de problemas operacionais, erros, conflitos e inconsistências, evolução do

conhecimento dos stakeholders ao longo do ciclo de desenvolvimento do sistema, problemas técnicos de prazo e de custos, mudança nas prioridades dos stakeholders e mudanças ambientais.

Os requisitos são analisados a partir da proposta de alterações, momento em que se verificam quantos requisitos e componentes do sistema, serão afetados pela alteração solicitada, bem como o cálculo estimado do impacto, em termos de prazo e de recursos orçamentários exigidos para a implementação da alteração solicitada.

Durante o processo de gerenciamento de alterações, o sistema deve manter um código de *status* para acompanhamento da solicitação, informando os estados de progresso que podem ser, dentre outros, os indicados a seguir:

- a) recebido;
- b) em análise;
- c) verificado;
- d) validado;
- e) aprovado;
- f) rejeitado;
- g) implementado;
- h) cancelado;
- i) adiado.

4.1.1.1. Análise preliminar

Trata da atividade inicial do macroprocesso tratamento de mudanças e tem como objetivo verificar se a solicitação pode, ou não, ser atendida. As ações a serem seguidas para a análise preliminar são as seguintes:

- a) verificar a correta identificação da origem da solicitação. Qualquer envolvido no projeto pode solicitar mudanças e a solicitação deve ser registrada no momento em que é recebida ;
- b) verificar se as mudanças solicitadas não comprometem o funcionamento das demais características do projeto e se são válidas; caso sejam válidas:
- preencher o respectivo formulário de registro da entrada da solicitação, Figura 4.4, para que o pedido de mudanças possa ser monitorado ao longo do processo de análise,
 - verificar se os requisitos/mudanças solicitadas contribuem para a solução do problema. Tal análise deve ser inicialmente realizada com a participação de toda a equipe do projeto e, posteriormente, com a participação dos patrocinadores do projeto,
 - verificar se os requisitos/mudanças solicitadas estão dentro do escopo das versões atuais,
 - verificar se os requisitos/alterações solicitadas possuem ambiguidades, duplicidades, se estão claras, devidamente descritas e não promovem conflitos com os demais requisitos,
 - verificar se os requisitos/mudanças solicitadas podem ser considerados sem afetar o orçamento, prazo, prioridades e viabilidade técnica,
 - conferir se as mudanças solicitadas não exigem mais detalhamento para que sejam analisadas e se podem ser testadas (verificadas). Caso exijam maior detalhamento, devem ser devolvidos para a unidade de origem para que a solicitação de mudança seja complementada com as informações exigidas.

As solicitações de alterações de requisitos devem ser realizadas e acompanhadas através de formulários específicos que contenham todas as informações necessárias para seu processamento, tais como: data da solicitação, solicitante, prioridade, justificativas e observações adicionais. O papel desse formulário é principalmente garantir que as solicitações sejam realizadas e que sejam documentadas com o mínimo de informações exigíveis para as análises e processos de julgamento que se seguirão, além de garantir a padronização dos procedimentos para facilitar a execução do processo da

rastreabilidade, que é essencial para o gerenciamento de alterações de requisitos. A Tabela 4.2 apresenta os itens sugeridos para os diversos formulários necessários, dentre outros.

Tabela 4.2 - Atributos da solicitação de mudança.

	Atributo	Descrição
1	Identificação do solicitante (Cliente, fornecedor, membro da equipe interna)/ área funcional da mudança	Identifica o originador da requisição de mudança.
2	Identificação da solicitação	Número de identificação única.
3	Status da solicitação	Estado da solicitação.
4	Data	Data da requisição.
5	Projeto	Identificação do projeto.
6	Título da mudança	Título para a mudança solicitada.
7	Descrição da mudança	Descrição completa da mudança solicitada.
8	Itens afetados pela mudança	Relação de itens afetados pela mudança proposta.
9	Documentos afetados pela mudança	Relação de documentos afetados pela mudança proposta.
10	Justificativa da solicitação de mudança	Descreve a lógica que guia a solicitação de mudança.
11	Análise preliminar (analista/data)	Descrição do resultado da análise preliminar, com datas de entrada e saída do processo, responsável e anexos (se existirem).
12	Análise de impacto (analista/data)	Descrição do resultado da análise de impacto, com datas de entrada e saída do processo, responsável e anexos (se existirem). Informação de impacto em custos, cronograma, efeitos estimados em acordos, etc.
13	Classificação (criticidade/prioridade)	Classificação quanto à criticidade, prioridade.
14	Verificação (verificador/ data)	Descrição do Resultado da verificação, com datas, responsável e anexos (se existirem).
15	Validação (Validador/ data)	Descrição do Resultado da validação, com datas, responsável e anexos (se existirem).

(Continua)

Tabela 4.2 - Conclusão

16	Status da última atualização da solicitação.(Atualizador/data)	Estado da solicitação.
----	--	------------------------

Fonte: Adaptada pelo autor a partir de European Space Agency – ESA (2009).

O formulário em que serão inseridos os dados de análise preliminar da proposta de mudança é apresentado na Figura 4.4, a seguir:

Figura 4.4 - Página com dados da análise preliminar da mudança proposta.

The screenshot shows a web browser window with the following elements:

- Browser Title:** "Análise preliminar - Mozilla Firefox"
- Address Bar:** "localhost/inpe/form2.php" with a search bar and navigation icons.
- Page Header:**
 - INPE logo on the left.
 - Image of a satellite in space.
 - Text: "PÓS-GRADUAÇÃO ENGENHARIA E TECNOLOGIA ESPACIAIS"
- Main Content:**
 - Form Title:** "Formulário de análise preliminar"
 - Status:** "Análise preliminar" with radio buttons for "Aprovado" and "Rejeitado".
 - Fields:**
 - Número da solicitação: [text input]
 - Data de entrada: [dd-mm-aaaa]
 - Data de saída: [dd-mm-aaaa]
 - A origem da da solicitação está corretamente identificada? [Sim] [Não]
 - A mudança solicitada está dentro do escopo das versões atuais? [Sim] [Não]
 - O grau de detalhamento da solicitação é adequado? [Sim] [Não]
 - A proposta de novo requisito está escrita corretamente?(ver ajuda de escrita de requisitos)? [Sim] [Não]
 - A mudança proposta resolve o problema? [Sim] [Não]
 - A mudança é tecnicamente possível de ser implementada? [Sim] [Não]
 - Observações:** [text area]
 - Anexos:** [Sim] [Não]
 - Responsável pela análise preliminar:** [text input]
 - Submit:** [Submit button]
- Footer:** Home icon and link: "Retorna à página de escolha de formulários"

Fonte: Elaborada pelo autor.

4.1.1.2. Análise de impacto

Trata-se de subprocesso que compõe o processo análise de solicitação e é responsável pela análise que identificará e descreverá os impactos que as mudanças solicitadas poderão causar sobre os objetivos do projeto e sobre os recursos exigidos. Este subprocesso recebe as informações oriundas da análise preliminar e aprofunda a investigação que, dentre outros aspectos, identificará a volatilidade e estabilidade do requisito submetido à análise. Os resultados da análise de impacto são direcionados para o subprocesso que tratará da classificação da solicitação das alterações.

Esta análise tem como objetivo: 1) identificar todos os elementos que serão afetados por uma alteração solicitada e 2) apresentar uma estimativa dos recursos necessários para realizar a mudança. Além disso, o risco de não se fazer a mudança deve também ser avaliado. A análise de impacto é um processo que depende fortemente do processo de rastreabilidade.

A análise de impacto aborda os seguintes aspectos:

- a) alcance do impacto na árvore de requisitos (efeito sobre os demais requisitos);
- b) custo estimado da alteração proposta (esforço);
- c) impacto estimado para o cronograma;
- d) impacto sobre a operação e desempenho;
- e) alteração de acordos contratuais;
- f) impacto sobre viabilidade técnica do produto;
- g) inserção de novos riscos no projeto.

A avaliação de impacto de uma alteração proposta é tarefa complexa, visto que uma mudança pode ter um escopo apenas local, sem grandes distúrbios para o projeto como um todo, ou poderá afetar consideravelmente o projeto caso esta

esteja fortemente relacionada a outros itens na árvore de requisitos, visto que os aspectos abordados deverão ser avaliados para cada requisito na trilha da alteração. Este espalhamento dos efeitos de uma mudança por vários requisitos torna o uso de ferramenta de software essencial para, através da funcionalidade de verificação de rastreabilidade, avaliar o alcance do efeito da alteração solicitada, ou seja, identificar quantos requisitos e artefatos serão, potencialmente, afetados pela alteração proposta.

Foram consideradas as recomendações de Wiegers e Beaty (2013) para a construção do roteiro e formulário para a execução das atividades de análise de impacto e seu registro. As Tabelas 4.3, 4.4 e 4.5 mostram um roteiro e os formulários para registros do esforço a ser empreendido, para a realização da mudança solicitada e dos resultados da análise.

Tabela 4.3 - Roteiro para auxiliar nas atividades do processo de análise de impacto das mudanças dos requisitos solicitadas.

QUESTÕES PARA AVALIAÇÃO DE IMPACTO DAS MUDANÇAS
A mudança irá aperfeiçoar ou prejudicar a satisfação de algum requisito de usuário?
Os requisitos existentes na <i>baseline</i> estão em conflito com a alteração proposta?
Existem alterações pendentes que são impactadas pela alteração proposta?
Quais são as conseqüências técnicas e contratuais de não fazer a mudança?
Quais são os possíveis efeitos secundários adversos, ou outros riscos de fazer a alteração proposta?
A alteração proposta afetará negativamente o desempenho ou outros atributos de qualidade?
A mudança proposta é viável considerando as restrições técnicas, normativas e contratuais existentes?
A proposta de alteração colocará exigências inaceitáveis sobre quaisquer recursos necessários para o desenvolvimento, teste ou ambientes operacionais?
Alguma ferramenta deve ser adquirida para implementar e testar a mudança?
Como a mudança proposta afetará a seqüência, dependências, esforço ou duração de qualquer tarefa atualmente no plano do projeto?
Será necessária a criação de protótipos ou outra entrada do usuário para validar a alteração?

(Continua)

Tabela 4.3 - Conclusão

Quanto esforço já investido no projeto será perdido se essa mudança for aceita?
A mudança exigirá treinamento adicional de pessoal envolvido no desenvolvimento do produto?

Fonte: adaptado de Wiegers e Beaty (2013).

Tabela 4.4 - Roteiro para identificar artefatos que serão afetados com a realização da mudança solicitada.

QUESTÕES PARA AVALIAÇÃO DA EXTENSÃO
Identificar quaisquer alterações, adições ou exclusões da interface do usuário necessárias.
Identificar quaisquer alterações, adições ou eliminações necessárias em relatórios, bases de dados ou arquivos.
Identificar os componentes de projeto que devem ser criados, modificados ou excluídos.
Identificar os arquivos de código-fonte que devem ser criados, modificados ou excluídos.
Identificar os testes existentes de unidade, integração e sistema a serem modificados ou excluídos.
Estimar o número de novos testes de unidade, integração e sistema necessários.
Identificar documentação que deve ser criada ou modificada.
Identificar qualquer impacto que a mudança proposta terá sobre o Plano de Gerenciamento do Projeto, Plano de Garantia de Qualidade, Plano de Gerenciamento de Configuração ou outros planos.

Fonte: adaptado de Wiegers e Beaty (2013, p.203-206).

Tabela 4.5 - Formulário para a estimativa do esforço a ser empreendido para uma mudança de requisito.

Tarefa	Horas de esforço
Atualizar o SRS ou repositório de requisitos.	
Desenvolver e avaliar um protótipo.	
Criar novos componentes de projeto.	
Modificar componentes de projeto existentes.	
Desenvolver novos componentes de interface.	
Modificar componentes de interface existentes.	
Desenvolver nova documentação.	

(Continua)

Tabela 4.5 - Conclusão

Modificar a documentação existente.	
Desenvolver novo código-fonte.	
Modificar o código-fonte existente.	
Escrever novos testes de unidade e integração.	
Modificar testes de unidade e integração existentes.	
Executar testes de unidade e de integração após a implementação.	
Escrever novo sistema e testes de aceitação.	
Modificar o sistema existente e os testes de aceitação.	
Modificar conjuntos de testes automatizados.	
Desenvolver novos relatórios.	
Modificar relatórios existentes.	
Desenvolver novos elementos de banco de dados.	
Modificar elementos do banco de dados existente.	
Desenvolver novos arquivos de dados.	
Modificar arquivos de dados existentes.	
Modificar vários planos de projeto.	
Atualizar a matriz de rastreabilidade de requisitos.	
Outras tarefas.	
Esforço total estimado.	

Fonte: adaptado de Wiegers e Beaty (2013).

A quantidade de requisitos que serão afetados pela alteração proposta pode ser usada como métrica, mínima, para avaliação da extensão do impacto da alteração em análise.

Foi elaborada a Tabela 4.6 para o registro de frequência de solicitações de mudanças sobre o requisito, indicando uma escala para classificação da volatilidade dos requisitos analisados e reunindo os indicadores básicos que devem resultar da análise de impacto. Tal tabela acata as recomendações de Wiegers e Beaty (2013), de Buckley (1990) e de Grande (1990).

A Tabela 4.6, a seguir, sugere um conjunto de indicadores.

Tabela 4.6 - Indicadores.

Indicador	Como medir
Quantidade de solicitações de mudança por requisito.	Contabilizar todas as solicitações que estejam associadas a cada requisito, no período considerado para avaliação.
Quantidade de solicitações de mudanças aprovadas.	Contabilizar todas as solicitações que estejam associadas a cada requisito e que foram aprovadas, no período considerado para avaliação.
Quantidade de solicitações de mudanças que estejam pendentes de aprovação.	Contabilizar todas as solicitações que estejam associadas a cada requisito e que estejam pendentes, no período considerado para avaliação.
Quantidade de solicitações de mudanças canceladas.	Contabilizar todas as solicitações que estejam associadas a cada requisito e que foram canceladas, no período considerado para avaliação.
Quantidade média de horas de trabalho utilizadas no processo de avaliação.	Total de horas gasto no processo/quantidade de solicitações de alteração de requisitos avaliadas.
Quantidade de solicitações de alterações realizadas, por requisito.	Elaborar uma tabela com os requisitos e as quantidade de solicitações de alterações para cada um, no período considerado para avaliação.

(Continua)

Tabela 4.6 - conclusão

Quantidade de solicitações de alterações de requisitos por <i>stakeholder</i> .	Elaborar uma tabela com os requisitos e as quantidade de solicitações de alterações por requisito para cada solicitante, no período considerado para avaliação.
Esforço a ser empreendido para realizar as mudanças (horas/homem)	Elaborar uma tabela para registrar a quantidade de horas a serem utilizadas para a realização da mudança solicitada.
Comprometimento do cronograma (dias de atraso)	Elaborar uma tabela para registrar os efeitos estimados sobre o cronograma original do projeto.
Comprometimento do orçamento (unidade monetária/mudança)	Elaborar uma tabela para registro do custo estimado para a realização das mudanças solicitadas.
Volatilidade (baixa, media, alta)	Elaborar uma tabela para registro da frequência de solicitação de mudanças sobre cada requisito, de forma a verificar a sua estabilidade.

Fonte: Adaptada pelo autor a partir de Wieggers e Beaty (2013). Buckley (1990) e Grande (2006).

A avaliação do impacto deverá resultar em um relatório com todas as informações obtidas ao longo do processo de análise, bem como mapeamentos e matriz de correlação e verificação da volatilidade e estabilidade do requisito. É importante notar que a volatilidade dos requisitos deverá ser identificada e servirá como parâmetro para medir o grau de instabilidade de um requisito ao longo do desenvolvimento do produto e contribuirá para a análise de riscos e impactos relacionados com as alterações pretendidas. Deverá ser estabelecida a seguinte estrutura:

- a) descritor: elemento que descreva resumidamente o atributo no projeto;
- b) quantidade: quantidade de novas versões do requisito verificada no período de análise;
- c) período de análise: é a quantidade de meses a ser considerada para a observação;
- d) classificação: definir uma escala para classificação da volatilidade observada no projeto no período considerado para análise.

Sugere-se a escala definida na Tabela 4.7, a seguir.

Tabela 4.7 - Verificação de volatilidade dos requisitos.

volatilidade	Quantidade de versões existentes no período	peso
Baixa	De 1 a 2	1
Média	De 3 a 5	2
Alta	Acima de 5	3

Fonte: Adaptada pelo autor a partir de Grande (2006).

Assim, a identificação de volatilidade de requisitos com o percentual de requisitos voláteis propicia a possibilidade de adequar o processo de gerenciamento, a partir de medidas gerenciais específicas, sobre requisitos com alto potencial para mudança.

O formulário em que serão inseridos os dados de análise de impacto da proposta de mudança é apresentado na Figura 4.5, a seguir, enquanto que as Figuras 4.6 e 4.7 apresentam roteiros auxiliares para subsidiar a realização da análise de impacto.

Figura 4.5 - Página com dados para análise de impacto da mudança proposta.

Análise de impacto - Mozilla Firefox

Análise de impacto x +

localhost/inpe/form3.php 70% Search

INPE **PÓS-GRADUAÇÃO**
Formulário de análise de impacto **ENGENHARIA E TECNOLOGIA ESPACIAIS**

Análise de impacto Aprovado Aprovado com restrição Rejeitado

Restrições:

Número da solicitação:

Data de entrada:

Data de saída:

Custo total de esforço estimado (em horas):

Custo total (em R\$):

Atraso estimado no cronograma (em dias):

- Acordos contratuais serão atingidos pela alteração? Sim Não

- A mudança acrescenta algum novo risco ao projeto? Sim Não

- A mudança ameaça a viabilidade técnica do projeto? Sim Não

- O impacto da não adoção da mudança proposta foi analisado? Sim Não

Observações:

Anexos para análise de impacto: Sim Não

Classificação:

Criticidade: Alta Média Baixa

Prioridade: Alta Média Baixa

Responsável pela análise de impacto:

[Retorna à página de escolha de formulários](#)

Fonte: Elaborada pelo autor.

Figura 4.6 - Roteiro para auxílio à análise de impacto da mudança proposta.

Análise de impacto - Mozilla Firefox

Análise de impacto

localhost/inpe/form3_auxiliar.php 70%

INPE

Roteiro para auxiliar nas processo de análise de impacto

PÓS-GRADUAÇÃO
ENGENHARIA E TECNOLOGIA ESPACIAIS

1- A mudança irá aperfeiçoar ou prejudicar a satisfação de algum requisito de usuário? Sim Não

2- Os requisitos existentes na baseline estão em conflito com a alteração proposta? Sim Não

3- Existem alterações pendentes que são impactadas pela alteração proposta? Sim Não

4- A alteração proposta afetará negativamente o desempenho ou outros atributos de qualidade? Sim Não

5- A mudança proposta é viável considerando as restrições técnicas, normativas e contratuais existentes? Sim Não

6- A proposta de alteração colocará exigências inaceitáveis sobre quaisquer recursos necessários para o desenvolvimento, teste ou ambientes operacionais? Sim Não

7- Alguma ferramenta deve ser adquirida para implementar e testar a mudança? Sim Não

8- Será necessária a criação de protótipos ou outra entrada do usuário para validar a alteração? Sim Não

9- A mudança exigirá treinamento adicional de pessoal envolvido no desenvolvimento do produto? Sim Não

10- Quais são as conseqüências técnicas, contratuais de não fazer a mudança?

11- Quais são os possíveis efeitos secundários adversos ou outros riscos de fazer a alteração proposta?

12- Como a mudança proposta afetará a seqüência, dependências, esforço ou duração de qualquer tarefa atualmente no plano do projeto?

13- Quanto esforço que já foi investido no projeto será perdido se essa mudança for aceita?

Responsável pela análise de impacto:

Submit

[Retorna à página de escolha de formulários](#)

Fonte: Elaborada pelo autor.

Figura 4.7 - Roteiro para identificar artefatos afetados pela mudança proposta.

The screenshot shows a Mozilla Firefox browser window with the address bar displaying 'localhost/inpe/form3_artefatos.php'. The page title is 'Análise de impacto'. The main content area features a header with the INPE logo and the text 'Roteiro para identificar artefatos que serão afetados' and 'PÓS-GRADUAÇÃO ENGENHARIA E TECNOLOGIA ESPACIAIS'. Below the header is a list of 8 questions, each followed by 'Sim' and 'Não' radio button options. The questions are:

- 1- Existem quaisquer alterações, adições, ou exclusões da interface do usuário, que são necessárias? Sim Não
- 2- Existem quaisquer alterações, adições ou eliminações necessárias em relatórios, bases de dados ou arquivos?? Sim Não
- 3- Existem componentes de projeto que devem ser criados, modificados ou excluídos? Sim Não
- 4- Existem arquivos de código-fonte que devem ser criados, modificados ou excluídos? Sim Não
- 5- Existem testes existentes de unidade, integração e sistema a serem modificados ou excluídos? Sim Não
- 6- Existem número de novos testes de unidade, integração e sistema necessários? Sim Não
- 7- Existe documentação que deve ser criada ou modificada? Sim Não
- 8- Existem quaisquer impactos que a mudança proposta terá sobre o plano de gerenciamento do projeto, Plano de garantia de qualidade, plano de gerenciamento de configuração ou outros planos? Sim Não

Below the questions is a text input field labeled 'Responsável pela análise de impacto:' and a 'Submit' button. At the bottom left, there is a home icon and a link: 'Retorna à página de escolha de formulários'.

Fonte: Elaborada pelo autor.

De posse de todos os documentos oriundos da criteriosa avaliação, o próximo passo no processo é a classificação e priorização da mudança solicitada.

4.1.1.3. Classificação

Os modelos analisados, de um modo geral, adotam a classificação das mudanças em requisitos em dois níveis:

Classe 1: mudanças significantes que afetam custo, cronogramas ou desempenho técnico e sugerem a necessidade de aprovação do cliente antes de sua implementação.

Classe 2: são mudanças de menor impacto que as da classe 1, afetam, em geral, documentação e erros internos em detalhes do projeto e não requerem aprovação do cliente.

No modelo proposto por esta pesquisa, a classificação constitui um subprocesso que compõe o processo análise da solicitação. O resultado da análise de impacto servirá de base para a classificação das mudanças, conforme o grau de impacto provocado sobre o projeto e permitirá a realização de uma classificação quanto à criticidade da mudança e, conseqüentemente, a priorização para implementação. A classificação adotada, neste modelo, focaliza a criticidade da solicitação de mudança em face dos efeitos sobre o projeto como um todo. Adotou-se, portanto, uma subdivisão da classe 1 em alta e média prioridades. As mudanças classificadas como alta prioridade apresentam alta criticidade por impactarem o custo e o cronograma do projeto, podendo inviabilizar sua conclusão caso não sejam implementadas.

As mudanças classificadas com média prioridade são aquelas que possuem média criticidade, ou seja, apesar de impactarem os custos e/ou cronograma, não comprometem as funcionalidades principais do projeto como, por exemplo, uma mudança gerada a partir de uma nova funcionalidade acrescentada para atender os desejos de *stakeholders* e/ou fornecedores.

E finalmente as mudanças classificadas como de baixa prioridade possuem baixa criticidade e não exigem urgência de implementação, por não comprometerem o andamento do projeto e apresentarem baixo comprometimento de custos e cronograma. Neste caso são consideradas, por exemplo, mudanças originadas por erros de documentação, ou pequenos acertos, ou acréscimos de funcionalidades, que se não forem implementadas não provocarão mudanças substanciais no projeto. São as comumente denominadas mudanças “cosméticas”, por estarem mais relacionadas com a forma de apresentação e acertos em *designs*.

Desta forma, em qualquer situação de mudança, a negociação de recursos e priorização, além de envolver a solução de inconsistências detectadas, exigirá a execução das seguintes atividades:

- a) definir a priorização conforme o seu grau de importância e prioridade para o projeto e a classificação poderá obedecer a seguinte escala:

quanto ao impacto: a análise de impacto será base para a definição quanto à criticidade da mudança solicitada conduzindo o processo de priorização para implementação conforme segue:

- alta: quando a mudança solicitada atingir o objetivo principal do projeto e suas funcionalidades principais, comprometendo profundamente o orçamento e o cronograma e colocando em risco a finalização do projeto devendo ser, portanto, de prioridade máxima a sua implementação,
- média: quando a mudança solicitada atingir itens e artefatos do sistema, provocando alterações no orçamento e/ou cronograma, mas não colocando em risco a finalização do projeto, devendo ser de prioridade média a sua implementação,
- baixa: quando a mudança solicitada atingir itens e artefatos do sistema sem grandes alterações de custos e cronogramas (mudanças “cosméticas”), devendo ser de prioridade baixa a sua implementação,

b) discutir com os *stakeholders*: prazos, recursos e, em alguns casos, até a alteração de requisitos já levantados, face aos impactos das mudanças solicitadas.

A classificação de requisitos, então, prioriza os requisitos em função, dentre outros aspectos, do grau de importância para o projeto, da criticidade e da complexidade. O gerenciamento de mudanças deve atentar para o fato de que alterações podem estabelecer uma nova ordem de prioridades em relação à classificação, originalmente definida. Além disto, em um ambiente com várias solicitações de mudanças estas devem ser agrupadas e organizadas.

Solicitações de mudanças podem ser conflitantes, exigindo um processo de negociação de forma a resolver estes conflitos.

Para a classificação propõe-se a existência dos seguintes campos:

- a) categoria: categoria da alteração solicitada de forma a facilitar o agrupamento considerando os impactos provocados no sistema;

- b) priorização: prioridade em função do grau de importância e da criticidade da mudança, acompanhada dos devidos esclarecimentos para fornecer suporte à solicitação.

4.1.1.4. Verificação

Trata-se de um dos subprocessos que compõem o processo análise da solicitação responsável pela verificação formal e aprofundada das mudanças solicitadas. Convém mencionar que tal verificação deverá ser executada pela área de desenvolvimento, cabendo ao processo de tratamento de mudanças o acompanhamento do *status*, o registro do resultado da verificação e o envio da documentação de registro de ações.

É importante ressaltar que o subprocesso de verificação executado pela área de desenvolvimento confirmará, ou não, o entendimento obtido na análise preliminar e direcionará a tomada de decisão quanto ao prosseguimento do processo de tratamento ou o retorno ao solicitante para pronúncia. A alteração proposta deve ser avaliada sob a ótica da correção técnica do que se solicita e, também, sobre a clareza e completude de sua redação, atentando para o fato de que todas as características que definem um bom requisito devem ser verificadas como, por exemplo, completude, não ambiguidade e consistência.

Um dos aspectos que define um requisito correto é a sua verificabilidade, isto é, a existência da possibilidade de testar o requisito para se determinar se o requisito, na forma como foi definido, encontra-se presente no sistema entregue. (YOUNG, 2004) Para tanto, o campo de descrição contido no formulário deverá ser preenchido segundo as boas práticas para escrita de requisitos e avaliado sob esta ótica.

O INCOSE (2011) cita que o processo de verificação certifica se o sistema e seus componentes executam as funções pretendidas e atendem os requisitos de desempenho alocados a eles, (certificando que o sistema foi construído de forma correta) e que os métodos de verificação abrangem inspeção, análise, demonstração e testes.

Além disto, segundo a ECSS (2009) deve-se assegurar que para cada requisito contido na especificação técnica de requisitos sejam identificados um ou uma combinação de métodos de verificação.

Desta forma, sugere-se um campo que receberá a descrição dos métodos a serem usados no processo de verificação da alteração (análise, revisão de projeto, testes).

O formulário em que serão inseridos os dados de verificação da proposta de mudança é apresentado na Figura 4.8, a seguir:

Figura 4.8 - Página com dados de verificação da mudança proposta.

The image shows a screenshot of a web browser window. The browser is Mozilla Firefox, and the page is titled "verificação - Mozilla Firefox". The address bar shows "localhost/inpe/form4.php". The page content includes a header with the INPE logo and the text "PÓS-GRADUAÇÃO ENGENHARIA E TECNOLOGIA ESPACIAIS". The main content is a form titled "Formulário de verificação". The form has a "Verificação" section with a green gear icon. It includes radio buttons for "Aprovado" and "Rejeitado". There are input fields for "Número da solicitação", "Data de entrada" (with a date picker), "Data de saída" (with a date picker), "Observações" (with a text area), "Anexos para verificação" (with radio buttons for "Sim" and "Não"), and "Responsável pela verificação" (with an input field). A "Submit" button is at the bottom. There is also a link "Retorna à página de escolha de formulários" with a home icon.

Fonte: Elaborada pelo autor.

4.1.1.5. Validação

Trata-se de um dos subprocessos que compõem o processo de análise da solicitação responsável pela validação dos requisitos. Não é demais lembrar que a validação de requisitos é a atividade que trata da certificação de que os

requisitos estabelecidos representam corretamente as expectativas e necessidades acordadas com os *stakeholders*.

Este subprocesso se estende por todo o ciclo de vida do produto em desenvolvimento pois, como já afirmado, a descoberta tardia de um requisito não validado insere problemas que podem inviabilizar o projeto.

Desta forma, o processo de validação desempenha papel fundamental no gerenciamento de alterações em função da possibilidade destas mudanças invalidarem a verificação e validação já efetuadas, em requisitos na linha de sucessão.

A avaliação da alteração do requisito tem como objetivo confirmar a sua consistência e completude: são identificados os possíveis problemas no requisito, antes de se criar o documento que servirá de base para sua implementação e integração ao sistema.

Para validar os requisitos pode ser utilizada a análise manual sistemática dos requisitos, através de mapeamento de cada requisito em relação aos componentes do sistema e de inspeção.

As atividades de inspeção garantem a qualidade do produto ao longo da execução de cada fase do ciclo de vida, auxiliando a equipe de desenvolvimento na solução das dificuldades devido à falta de clareza ou de informações que descrevem os requisitos. Este processo é importante não só para o projeto em andamento, mas também para que, no próximo projeto, defeitos semelhantes não sejam cometidos, ressaltando que o documento de especificação do requisito e sua alteração é a base fundamental para todas as outras fases de desenvolvimento e para a qualidade do sistema.

Pode ser usada, por exemplo, a técnica de prototipação, com o uso de um pequeno modelo executável do sistema e pode ser utilizada a geração de casos de testes, desenvolvendo-se testes específicos e análise automática de consistências. As recomendações a seguir são importantes para auxiliar neste processo:

- a) verificar se todos os *stakeholders* estão de acordo com as mudanças solicitadas e com as consequências identificadas;
- b) levantar os possíveis testes de validação das solicitações;
- c) verificar se os requisitos/alterações solicitados definem exatamente o que o *stakeholder* quer;
- d) verificar se foram incluídas todas as funções necessárias para atendimento das solicitações;
- e) rastrear os requisitos desde a origem, passando pelas interdependências e seus relacionamentos com os diagramas e componentes do projeto.

No formulário proposto incluiu-se um campo para descrição do método a ser usado para validar a alteração. Tal formulário, que acolherá os dados de validação da proposta de mudança é apresentado na Figura 4.9, a seguir:

Figura 4.9 - Página com dados de validação da mudança proposta.

Validação - Mozilla Firefox

Proposta ... x Mudança_i... x Analise de... x Analise pr... x verificação x Validação x Aprovação... x +

localhost/inpe/form5.php 80% Search

Formulário de validação

PÓS-GRADUAÇÃO
ENGENHARIA E TECNOLOGIA ESPACIAIS

Validação Aprovado Rejeitado

Número da solicitação:

Data de entrada:

Data de saída:

Observações:

Anexos para validacao: Sim Não

Responsável pela validacao:

[Retorna à pagina de escolha de formulários](#)

Fonte: Elaborada pelo autor.

4.1.1.6. Aprovação

Trata-se de um dos subprocessos que compõem o processo análise da solicitação responsável pela aprovação, ou não, de requisições de mudanças. Deve ser documentado e deve descrever a lógica utilizada no processo decisório. As atividades que compõem este subprocesso são executadas e os resultados são formalizados pelo CCB.

Nem sempre as solicitações de alterações são aprovadas. Em algumas situações podem ser rejeitadas, caso sejam enquadradas nas condições apresentadas a seguir:

- a) se a solicitação for considerada inválida: Isto ocorre quando se conclui que a alteração não vai agregar valor ao sistema e, portanto, é desnecessária;
- b) se a solicitação provocar modificações que os *stakeholders* não aceitam;
- c) se a solicitação provocar aumento substancial de custo e/ou nos prazos;
- d) se a mudança produzir riscos inaceitáveis para os *stakeholders*.

Os resultados conclusivos devem ser plenamente justificados, apresentando-se o elenco de razões para a aprovação, ou não, da solicitação. Os resultados devem ser comunicados simultaneamente a todos os envolvidos.

Convém lembrar que segundo a ECSS – European Space Agency – ESA (2012), um *waiver* pode ser uma saída válida para um processo de verificação, por ser uma autorização formal para um produto que em algum ponto afastou-se da sua especificação de requisitos.

Já segundo a NASA, um *waiver* autorizado, não constitui uma mudança à *baseline*, porém o modelo proposto neste trabalho sugere o seu registro, tanto na *baseline* quanto na base de conhecimento.

O formulário em que será registrada a aprovação, ou rejeição, da proposta de mudança é apresentado na Figura 4.10, a seguir:

Figura 4.10 - Página com aprovação/rejeição pelo CCB.

Aprovação CCB

PÓS-GRADUAÇÃO
ENGENHARIA E TECNOLOGIA ESPACIAIS

Aprovação do CCB

Aprovado Rejeitado

Número da solicitação:

Responsável CCB:

Ocorrência de: Waiver Desvio Sem ocorrência

Enviar para alteração contratual:

Enviado Data de envio:

Aguardando informação Data:

Enviar para implementação:

Enviado Data de envio:

Aguardando informação Data:

Enviar para para manutenção de baseline de requisitos para:

Operação a ser executada: Inclusão Exclusão Modificação

Enviado Data de envio:

Aguardando informação Data:

Enviar para manutenção de base de conhecimento::

Enviado Data de envio:

Aguardando informação Data:

Submit

[Retorna à página de escolha de formulários](#)

Fonte: Elaborada pelo autor.

4.1.2. Alteração contratual

Trata-se de um dos subprocessos que compõem o processo análise da solicitação. Este subprocesso é composto por atividades que culminam na formalização da atualização dos contratos envolvidos. No caso em que o produto em desenvolvimento seja software, as alterações impostas ou necessárias deverão obrigatoriamente ser pactuadas com os interessados, de maneira a evitar problemas posteriores, inclusive legais. Neste caso, novos

acordos de níveis de serviços ou ajuste dos existentes deverão ser elaborados. A norma ABNT NBR ISO/IEC 20000-1:2011 (ABNT, 2011) recomenda que este pacto deva ser acordado entre os interessados em um determinado serviço de TI e o responsável pelo desenvolvimento do serviço na organização. Deve ser revisado periodicamente, em face das alterações dos requisitos ao longo do desenvolvimento.

Desta forma, monitorar e verificar se o acordo de nível de serviços existente entre os desenvolvedores e os clientes assegura o esclarecimento sobre como os serviços serão realizados tecnicamente, bem como sobre os processos técnicos necessários para a operacionalização do sistema, constatando se os acordos são eficazes e se contemplam as mudanças em requisitos que tenham sido realizadas.

Deverão ser negociadas com os *stakeholders* as novas condições de trabalho em face das mudanças solicitadas e deverão ser especificados recursos adicionais, se for o caso, e prazos estendidos.

O processo de gerenciamento deve prover a ligação entre o requisito modificado e os acordos, de forma a serem avaliados quanto à necessidade de atualização.

4.1.3. Implementação das mudanças solicitadas

Trata-se de um dos processos que compõem o macroprocesso tratamento de mudanças. Após a aprovação da solicitação de mudanças e da alteração contratual devidamente acordada com os *stakeholders*, a solicitação de mudança, caso o produto já esteja na fase de desenvolvimento, é encaminhada para a equipe de desenvolvimento com instruções para a implementação. As atividades deste processo são inteiramente executadas pela equipe de desenvolvimento do projeto, cabendo ao gerenciamento de mudanças o registro do envio e a continuidade do andamento do controle de mudanças quando recebido do retorno da implementação.

4.1.4. Geração e manutenção da *baseline*

Trata-se de um dos processos que compõem o macroprocesso tratamento de mudanças. Este processo deve possuir subprocessos para o Gerenciamento da Configuração e gerenciamento de versões. O gerenciamento de configuração identifica, controla e verifica as versões dos itens de Configuração existentes, em face das mudanças aprovadas.

O Gerenciamento da configuração tem sob sua responsabilidade todos os ativos e configurações do projeto, a manutenção das informações sobre configurações e documentação correspondente atualizada para apoiar todos os outros processos do projeto, o fornecimento de uma base para Gerenciamento de Mudanças e Gerenciamento de Versões, e a manutenção de mecanismos de comparação de registros de configuração com o ambiente corrente para corrigir quaisquer exceções.

Os itens de configuração devem ser atualizados e identificados de forma única, que permita obter informações, dentre outras, sobre localização, número de versão, origem, histórico de modificações com identificação de quem modificou e as justificativas. A cada alteração efetuada, o histórico do evento deverá ser mantido com, no mínimo, as seguintes informações:

- identificação (incluindo a versão);
- descrição completa da alteração efetuada;
- solicitante;
- justificativa da alteração;
- nova redação do requisito.

O processo de controle de versões tem por objetivos:

- a) identificar a versão do documento de requisitos e de cada requisito através de um esquema pré-definido;
- b) registrar as modificações nos requisitos;

- c) manter todas as mudanças realizadas nos requisitos registradas no histórico do projeto. Cada nova situação do projeto em face de modificações é considerada uma configuração.

Um servidor deverá armazenar os arquivos do projeto e suas versões. Tais documentos devem ser disponibilizados aos desenvolvedores que poderão fazer uma cópia local na qual podem ser realizadas alterações. As alterações poderão ser realizadas sob processo de controle de acesso, através de credencial de segurança com registro da identificação do autor e horário da alteração. Estas informações deverão ser enviadas ao servidor para gerar nova versão incorporando as alterações realizadas.

O mecanismo de controle de versões deve atender às necessidades de informações sobre as alterações solicitadas, canceladas, rejeitadas e realizadas, ao longo do ciclo de desenvolvimento do projeto e poderá ser realizado da seguinte forma:

O registro de mudanças poderá ser centralizado em um único servidor ou distribuído, sendo este último recomendado quando as equipes forem muito grandes. Os principais elementos de um sistema de controle de versões são:

- a) item de configuração: cada elemento de informação que é criado ou que seja necessário durante o desenvolvimento de um produto, deve ser identificado de forma única e sua evolução deve ser registrada para que possam ser rastreadas suas modificações ao longo do projeto;
- b) repositório: que é o local de armazenamento de todas as versões dos arquivos;
- c) versão: que representa o estado de um item de configuração que está sendo modificado. Toda versão deve possuir um identificador único;

- d) ramo: que é uma versão paralela ou alternativa. Os ramos não substituem as versões anteriores e são usados concorrentemente em configurações alternativas.

Um mecanismo de controle de versão deve incorporar funções básicas para a garantia de integridade das informações, quais sejam:

- a) atualizar cópia de trabalho: permite realizar uma cópia da versão mais atual existente no repositório para a área de trabalho do desenvolvedor;
- b) realizar alterações: permite efetivar as alterações pretendidas para a versão, na área de trabalho;
- c) verificar alterações: permite obter as informações sobre o estado de arquivos e diretórios na cópia de trabalho e mostrar as diferenças entre duas revisões;
- d) desfazer alterações: permite voltar à edição original da versão do documento de requisitos;
- e) resolver conflitos: permite remover os conflitos existentes na versão da área de trabalho;
- f) submeter alterações: permite enviar a versão alterada na área de trabalho para o repositório de versões.

4.2. Monitoramento

O monitoramento é um dos quatro macroprocessos do modelo proposto. Consiste no acompanhamento contínuo do que ocorre com os requisitos do sistema. É composto por três processos: Rastreabilidade, Auditoria e Detecção de erros.

Após a conclusão do macroprocesso Tratamento de Mudanças, o seu resultado é encaminhado para o macroprocesso Monitoramento, onde os

requisitos serão rastreados e submetidos aos testes para detecção de erros e/ou defeitos. Os resultados do macroprocesso Monitoramento são encaminhados para o terceiro macroprocesso, que é a Avaliação, e disponibilizados para eventuais auditorias.

4.2.1. Rastreabilidade

O rastreamento dos requisitos permite ao desenvolvedor determinar quais componentes poderão ser afetados em face de mudanças.

O processo Rastreabilidade deve ser suportado pela ferramenta de software utilizada, onde deve ser mantida uma matriz de rastreabilidade e se possível uma forma gráfica de mostrar as ligações.

É necessário que a representação da rastreabilidade dos requisitos seja documentada. Desta forma, o gerenciamento de requisitos tem na rastreabilidade um dos seus aspectos mais importantes, já que garante o uso do requisito ao longo de todo o ciclo de vida do sistema.

A representação do rastreamento permite obter e gerir informações de objetos envolvidos no processo de desenvolvimento do sistema que podem contemplar informações dos *stakeholders*, protocolos utilizados para as entrevistas no momento da elicitação de requisitos.

Dependendo do volume de alterações que são solicitadas, conforme sugerido por Grande (2006), o gerenciamento de alterações de requisitos pode exigir o uso de ferramentas específicas de gerenciamento de requisitos, que podem prover facilidades tais como:

- a) formulários eletrônicos de solicitação de mudanças, e disponibilizados aos diversos participantes;
- b) bancos de dados para armazenar e gerenciar os formulários de mudança;

- c) *templates* de mudança que possa ser instanciado de forma a identificar cada responsável por cada atividade do processo;
- d) formulários colaborativos que possam ser compartilhados com pessoas com diferentes responsabilidades, com a possibilidade de notificar os envolvidos quando cada atividade estiver concluída;
- e) *links* diretos para banco de dados de requisitos.

4.2.2. Auditoria

Trata-se de um processo que compõe o macroprocesso monitoramento. É responsável pela auditoria do que foi implementado em relação ao que foi autorizado, resultando em relatório de ocorrências.

Após a realização das alterações, os registros oriundos dos procedimentos de alteração são direcionados para a equipe de auditoria para verificar se:

- a) As alterações especificadas na Ordem de Alteração foram efetuadas?
- b) Alguma modificação adicional foi incorporada?
- c) Foi efetuada a revisão técnica formal?
- d) A data e o autor da alteração foram especificados?
- e) Os atributos do item da configuração refletem a alteração?
- f) Foram seguidos os procedimentos de Gerenciamento de Configuração?
- g) Todos os Itens de Configuração relacionados foram atualizados apropriadamente?
- h) Foi realizada a verificação técnica formal nos itens?

A execução dos procedimentos de auditoria resulta em relatórios sobre ocorrências encontradas. Tais relatórios são encaminhados à equipe de

monitoramento, à base de conhecimento e à equipe de avaliação para as devidas providências de correção de possíveis defeitos e registros.

4.2.3. Detecção de defeitos

Trata-se de um dos processos que compõem o macroprocesso monitoramento. É responsável pela avaliação acurada em busca de defeitos, de informação e documentação, que possam causar problemas durante o funcionamento no sistema.

No que tange aos requisitos, é considerada como defeito qualquer informação constante do documento de requisitos que possa causar um comportamento diferente do esperado no produto final. Portanto, qualquer aspecto que não atenda os níveis de qualidade pré-acordados com os *stakeholders*, ou que provoque ambiguidade ou inconsistência, é considerado defeito.

Estudos de Fagan (1986) apontam que 56% dos defeitos de um software podem ser identificados durante a etapa de Engenharia de Requisitos. Desta forma, foram considerados, no processo proposto, alguns aspectos sobre os diversos tipos de defeitos:

- a) os defeitos nos requisitos e suas alterações devem ser procurados logo no início do desenvolvimento das especificações, isto porque quanto mais tarde um defeito for detectado, maior será o custo de corrigí-lo;
- b) assim como muitos defeitos ocorrem ao longo do processo de elicitação e definição de requisitos, também ocorrem da mesma forma quando das solicitações de alterações;
- c) defeitos clássicos correspondem a fatos incorretos, omissões, inconsistências e ambiguidades;
- d) em geral defeitos que causam falhas (consequências) na funcionalidade promovem o fracasso no sistema e devem ser procurados de forma criteriosa.

Os defeitos podem ser classificados por tipos que ocorrem em subprodutos ou produtos e as suas classificações podem ser importantes para efeito de comparações entre ambientes.

A classificação de defeitos depende da fase do ciclo de vida na qual a inspeção irá ocorrer. Diversos autores sugeriram esquemas de classificações de defeitos que variam conforme sejam dirigidos às fases específicas do ciclo de vida do produto. Nesta pesquisa, abordou-se o documento de requisitos em sua plenitude, ou seja, com suas solicitações de alterações.

Considerando uma abordagem sobre desenvolvimento de software é comum perceber que os defeitos são classificados em esquemas de: classificação de defeitos para requisitos, classificação de defeitos por projeto, classificação de defeitos por código, acatando sugestões de Carver (2002).

Neste caso, foi assimilado apenas o que se refere aos defeitos de requisitos, já que o objetivo do trabalho focaliza os requisitos e suas alterações, tendo sido considerado, ainda, que a evolução de um documento de especificação de requisitos possa ser observada sob sete categorias conforme descritas em estudos de Basili et al (1981), conforme segue:

- a) de escrita: problemas relativamente simples no documento de requisitos, consistindo em erros ortográficos existentes no documento que possam provocar dificuldades no entendimento de seu conteúdo;
- b) ambiguidade: presença de informações no documento que possuam mais de um significado;
- c) omissão: falta de informações no documento;
- d) inconsistência: existência de incompatibilidades em partes do documento;
- e) fato Incorreto: existência de fatos e informações incorretas com respeito ao domínio;

- f) informações em local errado: informações incluídas no documento em seção errada;
- g) fato de Implementação: informação não fornecida e que seja necessária para a implementação;
- h) defeitos adicionais: defeitos que não estejam classificados nas categorias descritas acima.

Estudos sobre software relataram achados sobre os benefícios que foram obtidos ao se realizar inspeções ao longo do ciclo de desenvolvimento do software, sugerindo esquema de classificação de defeitos encontrados em documentos de especificação de requisitos, de projeto, de código, de planos de teste e especificações de teste, porém concentraram-se, apenas, em documentação de requisitos. O esquema sugerido consiste na classificação por categorias, caracterizadas por questões que auxiliam na procura pelos defeitos, baseados nas recomendações de Ackerman et al. (1989).

A Tabela 4.8 apresentada, a seguir, mostra a classificação de defeitos para a avaliação de requisitos e suas alterações.

Tabela 4.8 - Classificação de defeitos para avaliação de requisitos e suas alterações.

Categoria	Questões investigativas.
Compleitude.	<ul style="list-style-type: none"> a) Todas as fontes de entrada foram especificadas? b) Qual é a entrada principal? c) Há necessidade de maior robustez na entrada de dados? d) Todos os tipos de saídas foram identificados? e) Qual é a saída principal? f) Há conjuntos que estão em ambos, na entrada e na saída?

(Continua)

Tabela 4.8 - conclusão

	<p>g) Quais são os tipos de execução?</p> <p>h) Qual é a entrada e a saída para cada tipo de execução?</p> <p>i) Para cada tipo de execução, um valor de saída é especificado para cada valor de entrada?</p> <p>j) Estão definidos todos os mecanismos dos estados iniciais?</p> <p>k) Todas as restrições do ambiente foram descritas?</p> <p>l) São providos exemplos de amostras de entrada/saída apropriados?</p> <p>m) Todos os requisitos de desempenho necessários estão descritos?</p> <p>n) Todos os requisitos de confiabilidade foram especificados?</p>
Ambiguidade.	<p>a) Todos os termos especiais estão claramente definidos?</p> <p>b) Cada sentença tem uma interpretação única no domínio do problema?</p> <p>c) O delineamento da entrada-para-saída está claramente definido para cada tipo de execução?</p>
Consistência.	<p>a) Qualquer um dos requisitos designados está em conflito com o material analisado?</p> <p>b) Existe alguma entrada que está delineada para mais de uma saída?</p> <p>c) Há um conjunto de unidades quantitativas a ser usada?</p>

Fonte: Adaptada pelo autor a partir de Ackerman et al. (1989).

Seguindo recomendações de Martin et al. (1992), é importante extrapolar o exame de um processo de inspeção tradicional investigando as vantagens obtidas em inspeções múltiplas, independente de ser o mesmo documento de requisitos a ser inspecionado. Aos defeitos presentes nos documento de requisitos, incorridos pelos autores do documento de requisito ou da solicitação de alteração, foram acrescentados outros, conforme a experiência dos investigadores, resultando em duas classes de alto nível com subclasses em cada uma.

Tal classificação é útil para se verificar se um novo processo de inspeção poderia afetar o desempenho dos inspetores em encontrar quaisquer dessas classes de defeitos, conforme a visão de Martin et al. (1992). A Tabela 4.9, a seguir, mostra as classes de defeitos e as respectivas características sob inspeção.

Tabela 4.9 - Classes de defeitos e características sob inspeção.

Classe de defeito	Características sob inspeção
Informações perdidas referentes a: funcionalidades, interface, desempenho, ambiente.	a) a informação omitida sobre o comportamento interno do sistema; b) a informação omitida sobre a comunicação do sistema com o ambiente em que se insere; c) omissão de especificações de desempenho; d) omissão de Informações externas sobre infraestrutura envolvida.
Informações erradas (ambíguas, inconsistentes).	a) termos importantes não foram definidos ou permitem interpretações múltiplas; b) existem partes do documento que se contradizem.

Fonte: Adaptada pelo autor a partir de Martin et al. (1992).

As Tabelas 4.10 e 4.11, a seguir, mostram uma categorização obtida a partir da fusão dos achados dos autores Ackerman et al. (1989); e Martin et al. (1992), descritos anteriormente, que atendem o propósito deste trabalho no que diz respeito aos procedimentos de inspeção das solicitações de alteração de requisitos.

Tabela 4.10 - Categorização dos defeitos resultantes dos trabalhos mencionados.

Categoria	Questões investigativas.
Compleitude.	<ul style="list-style-type: none"> a) As fontes de entrada foram especificadas? b) As saídas foram identificadas? c) Há conjuntos presentes na entrada e na saída? d) Quais são os tipos de execução? e) Qual é a entrada e a saída para cada tipo de execução? f) Para cada tipo de execução há correspondente valor de saída para cada entrada? g) Os estados iniciais são definidos para todos os procedimentos? h) Todas as restrições do ambiente foram descritas? i) São apresentadas amostras de entrada/saída? j) Todos os requisitos de desempenho necessários estão descritos? k) Todos os requisitos de confiabilidade foram especificados? l) São informados dados sobre o desempenho do sistema?

(Continua)

Tabela 4.10 - conclusão

	<p>m) Todas as funcionalidades e o comportamento do sistema relativo à comunicação com o ambiente externo foram previstos?</p> <p>n) O padrão de desempenho esperado foi especificado?</p> <p>o) Há informações de requisitos de infraestrutura envolvida para o adequado funcionamento do sistema?</p>
Ambiguidade.	<p>a) Todos os termos especiais estão claramente definidos?</p> <p>b) Cada sentença tem uma interpretação única no domínio do problema?</p> <p>c) O delineamento da entrada-para-saída está claramente definido para cada tipo de execução?</p> <p>d) Termos importantes não foram definidos, ou permitem interpretações múltiplas?</p>
Consistência.	<p>a) Qualquer um dos requisitos designados está em conflito com o material analisado?</p> <p>b) Existe alguma entrada que está delineada para mais de uma saída?</p> <p>c) Há um conjunto de unidades quantitativas a ser usada?</p> <p>d) Existem partes do documento que se contradizem?</p> <p>e) O requisito foi implementado incorretamente?</p>

Fonte: Adaptada pelo autor a partir de Ackerman et al. (1989) e Martin et al. (1992).

Tabela 4.11. - Classes dos defeitos constantes nos trabalhos mencionados.

Classes dos defeitos encontrados	
Graves:	Precisam ser urgentemente corrigidos.
Críticos:	Precisam ser corrigidos antes da entrega para a produção.
Importantes:	Não comprometem a continuidade do processo mas provocam desvios quanto aos objetivos funcionais do sistema.
Pequenos:	Não comprometem a operacionalização do processo mas a estética de como os resultados são apresentados ao usuário.

Fonte: Adaptada pelo autor a partir de Ackerman et al. (1989) e Martin et al. (1992).

Desta forma, será possível obter um processo capaz de detectar defeitos na *baseline* de requisitos ao fazer uso da ferramenta de rastreabilidade, de forma a identificar requisitos incompletos, desnecessários e em não conformidade com normas aplicáveis.

4.3. Avaliação

Trata-se do terceiro macroprocesso que compõe o processo de gerenciamento de mudanças de requisitos proposto. É responsável pela avaliação da alteração dos resultados obtidos com a implementação das mudanças aprovadas em relação ao volume, custos, prazo e desempenho do sistema em face das alterações implementadas.

O macro processo de avaliação, em geral, é suportado pela interpretação de um conjunto de métricas obtido através do monitoramento das alterações solicitadas, que possua as seguintes características:

- a) as métricas devem ser claras, com facilidade para a obtenção e deverão permitir o acompanhamento pontual das solicitações de alterações, do acordo de nível de serviço pactuado;

- b) a coleta das métricas deve ser sistematizada, com definição de períodos de coleta de forma a não degradar o ambiente de desenvolvimento e operacional;
- c) as métricas devem permitir identificar o grau de qualidade do sistema bem como o nível de satisfação dos *stakeholders*, de maneira a manter ativo um processo de melhoria contínua;
- d) o conjunto de métricas deve permitir a projeção do comportamento do sistema, de forma a possibilitar uma previsão de seu comportamento futuro;

O conjunto de métricas deve permitir avaliar:

- a) a tramitação dos pedidos de alterações e seu *status*;
- b) o nível de volatilidade;
- c) a frequência e complexidade dos defeitos, o ônus produzido pelas alterações aprovadas, bem como pelo uso de recursos na análise das alterações não aprovadas e/ou canceladas;
- d) o comprometimento do cronograma.

4.3.1. Volume

Conforme ilustrado na Figura 2.7, a mudança de requisitos é uma realidade e pode acarretar um grande volume de alterações a serem realizadas.

Estudos disponibilizados pela literatura, como os de Christel e Kang (1992) apontam o problema ao se lidar com a volatilidade de requisitos e que estes problemas podem levar a resultados adversos tais como: altos custos de manutenção e atrasos significativos no cumprimento do cronograma, além de provocar intenso uso de força de trabalho na análise dos efeitos das mudanças solicitadas sobre todo o sistema, apresentando como causas da volatilidade:

- a) a evolução no tempo das necessidades dos *stakeholders*;

- b) a contribuição de diversos atores com visões e necessidades frequentemente conflitantes;
- c) complexidade organizacional.

Podemos atribuir, também, como causa da volatilidade a evolução do entendimento dos desenvolvedores sobre as reais necessidades do *stakeholders*.

Desta forma, a observação da volatilidade de requisitos e avaliação do volume de mudanças durante o projeto pode detectar um volume alto de solicitações de alterações, o que pode indicar que os processos de captação e elicitação de requisitos devam ser revistos.

Os resultados do macroprocesso Avaliação são observados ao longo do tempo, de forma a permitir a comparação dos indicadores para identificar o comportamento evolutivo, ou não, da taxa de crescimento de volumes em períodos, observando-se a existência de tendência crescente ou não. Nesse caso para um sistema que apresenta taxas crescentes de volume de solicitações de mudanças dever-se-ão empreender ações para a identificação das causas, principalmente na fase de elicitação de requisitos. Aqui é importante verificar o volume de solicitações de mudanças por solicitante, para agir pontualmente com o fim de mitigar os problemas de volume crescente.

4.3.2. Custo

A análise de custos deverá ser efetuada de forma cuidadosa para garantir a eficiência na alocação dos recursos e que estes sejam suficientes até a finalização do projeto.

O custo por alteração solicitada deve ser estimado e apresentado ao solicitante antes e depois da realização da alteração solicitada. Esse custo deve ser informado de forma cumulativa, ou seja, além da informação do custo individual, deve ser calculado o custo acumulado referente ao total de

alterações solicitadas naquele período e, ainda, o custo acumulado provocado por solicitações de alteração para cada solicitante.

4.3.3. Prazos

O controle bem ajustado de prazos é fundamental para manter o processo de gerenciamento de requisitos dentro do cronograma estabelecido para o projeto. É ponto pacífico na literatura que problemas nos requisitos são um fator preponderante na falha de cumprimento de prazos. Um produto de qualidade que entre no mercado com atraso pode estar fadado ao fracasso.

Os prazos, em geral, são avaliados com base em estimativas da quantidade adicional de horas de trabalho que serão necessárias para realizar as mudanças solicitadas. Esta métrica é necessária tanto para a avaliação de prazos quanto de custos, conforme comentado no item anterior.

Da mesma forma que ocorre na avaliação de custos, dever-se-á estimar o prazo necessário para a realização da alteração em face dos recursos existentes, realizando a apresentação ao solicitante antes e depois da realização da alteração solicitada. Esse atraso a ser incorporado no cronograma deve ser informado de forma cumulativa, ou seja, além da informação do atraso por solicitação, deve ser calculado o atraso acumulado referente ao total de alterações solicitadas naquele período e, ainda, o atraso acumulado provocado por solicitações de alteração para cada solicitante.

Esses indicadores são importantes para se verificar os possíveis pontos de gargalo ao longo do ciclo de vida do sistema/produto, de forma a permitir ações pontuais para reverter a tendência observada de acúmulo de atrasos.

4.3.4. Desempenho

As medidas de desempenho técnico descrevem a eficiência na tratativa dos requisitos e são importantes para os processos de avaliação de riscos e tomadas de decisões, por permitir indicadores que expressam a eficácia e a probabilidade de atingimento dos objetivos técnicos do projeto.

Sem tais medidas é muito difícil verificar os limites que indicam que um projeto está entrando em área crítica e pode ter sua finalização comprometida. O monitoramento desses indicadores proporciona a verificação contínua da comparação entre o planejado e o realizado ao longo do processo de desenvolvimento.

A produtividade da equipe deve ser avaliada, devendo ser observada continuamente ao longo do desenvolvimento do sistema/produto em períodos mais curtos possíveis, para que os pontos em que tal produtividade apresente declínio sejam identificados o mais rápido possível, impedindo a propagação dos erros e degradação do processo.

Deve ser calculada a produtividade por indivíduo prevista no início do projeto e comparada a cada período com a produtividade corrente, de forma a identificar se as alterações solicitadas provocaram declínio no processo. Assim deve ser definido um período padrão que pode ser semanal, por exemplo, caso a volatilidade dos requisitos apresente altas taxas. Deve ser informado aos solicitantes de alterações o comprometimento do sistema em termos de custos, cronogramas e produtividade, em face das alterações solicitadas.

Dessa forma, todas as estimativas realizadas no início do projeto deverão ser comparadas periodicamente, fornecendo os indicadores necessários para tomada de decisões pontuais, de maneira a reduzir ao máximo a reincidência de problemas.

4.4. Geração de conhecimento e gerenciamento de base de conhecimento

Trata-se do quarto macroprocesso que compõe o processo de gerenciamento de mudanças em requisitos proposto. É responsável pelo registro e manutenção de uma base de conhecimento, que é o repositório dos relatos de problemas ocorridos ao longo do desenvolvimento do sistema com a descrição das respectivas soluções. Aborda o problema, a solução e as críticas sobre melhorias dos processos implementados. Trata-se do registro das lições

aprendidas que servirão de guia para futuros projetos e implementação de melhores práticas.

No tópico de diretrizes para Engenharia de Requisitos em sistemas críticos estudos de Sommerville e Sawyer (1997) apontaram que se deve fazer o cruzamento de requisitos contra as experiências de incidentes e as lições aprendidas destas experiências, de forma a assegurar que o incidente não se repita e que os requisitos relacionados ao incidente não sejam deixados fora da especificação. Afirmando, ainda, que a adoção desta diretriz traz como benefícios.

- a) redução da chance de ocorrência de erros ou omissões prévias em requisitos, trazendo para a equipe atual a experiência passada de sistemas similares;
- b) a demonstração de que foram considerados incidentes anteriores na elaboração e tratamento dos requisitos pode ser importante para a avaliação de segurança (*safety*) de um sistema.

Sommerville e Sawyer (1997) apontam, ainda, a questão do planejamento para a inserção de informações obtidas em uma necessária base dados, recomendando que cada incidente ocorrido seja fonte para a geração de um conjunto de regras e cuidados que devam ser observados.

Para o estabelecimento desse conjunto de regras é importante considerar as seguintes atividades:

- a) análise dos incidentes para selecionar quais aspectos do caso em avaliação são relevantes para a geração de regras e de recomendações;
- b) formulação de regras e recomendações específicas para evitar a recorrência do incidente;

- c) verificação por parte de especialistas do domínio e de segurança (*safety*) para garantir que as regras/recomendações não introduzam problemas;
- d) melhoria contínua das regras/recomendações existentes com base em novas experiências com novos incidentes.

As quatro abordagens que foram objetos de comparação por esta pesquisa citam, pelo menos uma vez cada uma, a validade do uso de lições aprendidas como fator importante para a melhoria dos projetos.

A NASA (2007) e o INCOSE (2011) são mais categóricos neste aspecto pois reforçam o conceito declarando a necessidade de repositórios para a inserção e manutenção das lições aprendidas.

A NASA mantém um sitio público para acesso à base de lições aprendidas dos projetos e programas espaciais, que pode ser consultado em: <https://llis.nasa.gov/>.

O conhecimento gerado pelas soluções dos problemas relacionados com as mudanças de requisitos que ocorreram ao longo do projeto deve ser registrado, de forma a contribuir para o aperfeiçoamento dos processos de gerenciamento de mudanças de requisitos.

É importante que algumas considerações sejam tomadas como base, ao se pretender criar uma base de conhecimento a partir de lições aprendidas. Dentre tais considerações sugerem-se:

- a) definir uma estrutura para a organização do conhecimento com mecanismos para selecionar, atualizar e acessar as informações, mantendo a devida segurança das informações que lá estarão, uma vez que contribuirão para tomadas de decisões tanto no projeto em curso quanto em projetos futuros;
 - o estabelecimento de uma política de segurança para tal sistema é uma questão central. Isto porque as características são

diferenciadas para sistemas aeroespaciais, como já citadas anteriormente. Tal política deverá garantir:

- disponibilidade: a informação deve estar disponível para consulta e de forma abrangente;
 - integridade: o usuário ao acessar a informação tem a garantia de que esta é correta e atualizada e não foi alterada de forma indevida;
 - confidencialidade: a informação, classificada em diversos níveis, estará acessível somente aos usuários corretamente credenciados.
- Deve ser estabelecido um processo para controle e registro de acesso e manutenção de dados com o uso de credenciais de segurança, as quais terão poder para aprovar e liberar acesso para quem estiver autorizado a realizar a atualização da base;
- b) definir um processo para categorizar os problemas e respectivas soluções a serem armazenados por categorias e subcategorias, refletindo as características dos problemas ocorridos, bem como estabelecer modelo para padronização do conteúdo da base;
- c) definir um processo para garantir a confiabilidade das informações armazenadas através do estabelecimento de um procedimento para aprovação do conteúdo a ser agregado à base, envolvendo especialistas da equipe e gestor;
- d) definir mecanismo que, além da pronta recuperação da informação, evidencie a sua criticidade e impacto;
- e) estabelecer processo para possibilitar a recuperação da versão anterior;
- f) equipe de suporte e manutenção;

- Deve ser mantida uma equipe com conhecimentos multidisciplinares, tais como Tecnologia da Informação, gerenciamento de conhecimento e gerenciamento de informação.

g) recuperabilidade;

O sistema utilizado deve ser robusto o suficiente para manter controle de transações em curso, de forma a recuperar-se em caso de eventos de paradas ou quedas.

h) interoperabilidade;

Uma característica importante a ser considerada é a possibilidade de migração do sistema para outras plataformas.

i) escalabilidade;

Da mesma forma, um sistema com estas características apresenta potencial de crescimento elevado, o que deve ser levado em conta para o estabelecimento da implantação;

j) critérios de Classificação, recuperação e descarte de informação;

Estabelecer critérios para identificação e categorização de problemas por área e suprir o sistema com mecanismos de busca e critérios eficientes para recuperação da informação desejada.

k) descarte de informação;

Estabelecer regras para avaliação da utilidade da informação mantida na base, com critérios para o seu descarte, tempo de guarda e modificações tecnológicas, dentre outros.

A gerência do conteúdo da base é de grande relevância, pois, com uma base desatualizada, perder-se-á conhecimento precioso que poderia ser reutilizado em decisões críticas para o sistema; dessa forma, a manutenção e a garantia de sua existência através de uma política bem estabelecida e ajustada de cópias de segurança, serão de suma importância para todo o processo.

Assim, o procedimento de registro de lições aprendidas torna explícito o conhecimento agregado ao longo do ciclo de vida do projeto ao documentar os sucessos, falhas e sugestões para aprimoramento do desempenho do projeto; neste caso, com foco no processo de gerenciamento de mudanças em requisitos. É crucial que o processo de seleção das informações seja preciso para evitar que informações desnecessárias tornem o sistema, que certamente será volumoso, mais custoso ainda. Assim, sugere-se observação dos seguintes aspectos:

- a) verificação do conhecimento útil referente às soluções implementadas;
- b) análise de custo/benefício para manutenção da informação na base;
- c) entendimento das justificativas de grandes mudanças em função dos benefícios;
- d) identificação das capacidades agregadas evidenciadas em determinada solução que poderão ser reutilizadas.

Há que se estabelecer métricas e indicadores que possam ser utilizados para avaliar as soluções, seus componentes e demais aspectos envolvidos que possam influenciar a qualidade e custos do produto final, bem como frequência de acessos a tais informações que justifique a permanência dessas na base. Para isso, é necessária a definição de uma estrutura de coleta de informações a partir da *baseline* existente, de forma a demonstrar o progresso ou retrocesso do processo de desenvolvimento do produto à luz das mudanças ocorridas ao longo do seu ciclo de vida.

A base de conhecimento deve abranger, então, conhecimento explícito e, na medida do possível, o conhecimento tácito utilizado para a resolução dos problemas ocorridos ao longo do desenvolvimento do sistema.

A estrutura de base de conhecimento é fundamental para todo processo de gerenciamento de requisitos e, portanto, deve conter elementos suficientes

para uma correta análise, classificação, reutilização e rastreabilidade dentre outras atividades inerentes a um processo de gestão.

Como resultado final do processo de avaliação obter-se-á um rico acervo de soluções oriundas do enfrentamento dos problemas, que devem ser registrados em uma base denominada Lições Aprendidas. Convém ressaltar a importância do elenco de lições aprendidas uma vez que constituem uma base de conhecimento valiosa que deve ser utilizada para realimentar o processo de melhoria contínua da empresa e evitar retrabalho.

Todas as informações obtidas a partir das avaliações e inspeções deverão compor a base de conhecimento, em que os fatos e observações deverão ser estruturados e organizados conforme sugerido a seguir:

As informações que irão compor a base de conhecimento devem ser selecionadas de forma que possam ser debatidas e coletadas as lições aprendidas identificadas pela equipe, considerando sempre a possibilidade de reutilização da informação e, portanto, é importante que se estabeleçam critérios para a determinação do que deva ser considerado.

Dentre outros, podem-se adotar os critérios sugeridos, por Guzzo e Maccari (2012), apresentados a seguir:

- a) o caso ocorrido contribuiu para evitar o fracasso, ou uma situação crítica?
- b) é passível de medição?
- c) agregou valor ao projeto ou aos clientes?
- d) pode ser utilizada para sanar problemas em outros projetos?
- e) propicia orientações objetivas e sugestões para aprimoramento, ou resolução, do objetivo proposto?
- f) quais problemas foram encontrados e como foram resolvidos?

- g) o que funcionou bem no projeto? Isso poderia ter sido feito ainda melhor? Como?
- h) o cronograma precisou ser ajustado?
- i) o planejamento precisou ser ajustado?
- j) o processo de gerenciamento de mudanças foi efetivo? Como poderia ter funcionado melhor?
- k) o processo de comunicação funcionou bem? Como poderia ser melhor?
- l) os objetivos do projeto foram atingidos? Que mudanças poderiam ser sugeridas para novos projetos?
- m) quais, e como, processos utilizados ao longo do ciclo de vida do sistema poderiam ser aperfeiçoados para melhor atender o cliente/projeto? e de que forma?
- n) como garantir que as falhas ocorridas não se repitam em novos projetos?

Considerando a observação desses critérios, é necessário que os resultados sejam compilados em um documento estruturado, que será utilizado para a manutenção da base de lições aprendidas.

O registro das lições aprendidas no sistema deve ser feito de forma sistemática e de preferência com a classificação de cada caso considerado em função do impacto que causou, se contribuiu positivamente para o desenvolvimento do sistema, ou não, e em que área do projeto causou impacto, seu grau de complexidade e em que domínio de conhecimento se insere.

O processamento dessas informações consiste na classificação das lições de forma a possibilitar a elaboração de um plano de melhoria para o sistema e/ou sistemas futuros. As informações devem ser registradas e armazenadas em

local, com mecanismo de recuperação da informação e segurança, citados anteriormente, mas que permita facilidade de acesso aos participantes do projeto e que a cada nova lição armazenada haja a divulgação para todos os participantes do projeto.

Deverão ser contempladas na base de lições aprendidas, as descrições de problemas e soluções ocorridos relacionados, dentre outros, com os seguintes aspectos:

a) *server / workstation* (Equipamento utilizado)

- Deverão ser registradas todas as informações que caracterizem o ambiente em que o sistema será utilizado, de forma a permitir que o usuário seja capaz de identificar em que ponto está ocorrendo algum problema na execução do processo.

b) sistemas Operacionais

- informações sobre configurações e principais problemas frequentes relacionados com os sistemas operacionais devem ser registrados, para serem consultados no momento em que ocorrerem problemas provenientes de conflitos e/ou corrupção de arquivos dos sistemas operacionais.

c) recursos de *hardware*

- os sistemas gerenciadores de banco de dados, bem como software de gerenciamento de requisitos, devem ser bem explorados, pois contém mecanismos de busca e controle de conteúdos que facilitam a manutenção da integridade, além de permitir acesso às informações e seus controles, mantendo o registro do histórico de alterações. Características de tais recursos, ao serem utilizadas, devem ser registradas de forma clara para que possam ser consultadas pelos usuários em caso de dúvidas.

d) registro dos problemas tratados com os *stakeholders* e a estratégia gerencial adotada para solucioná-los;

e) registro de problemas encontrados relativos a requisitos ao longo do processo de desenvolvimento do produto/sistema com suas respectivas soluções.

- f) registro de problemas encontrados relativos a requisitos com suas respectivas soluções ao longo do processo de montagem e integração de sistemas e subsistemas;
- g) registro de problemas encontrados relativos a requisitos ao longo dos testes de verificação e validação.

A Figura 4.11 mostra a proposta de uma página para a base de conhecimentos, utilizando dados da base de conhecimentos pública da NASA para efeito de demonstração, em que foram usados, para o desenvolvimento, o software de banco de dados MySQL em conjunto com PHP/HTML.

Os códigos da página de demonstração e da estrutura da base de dados são apresentados, respectivamente, no apêndice B e C.

Figura 4.11 - Página da base de conhecimento proposta.

The screenshot shows a Mozilla Firefox browser window with the address bar at localhost/inpe/inpe2.php. The page title is 'Base de conhecimento'. The main content area features a header with the text 'PÓS-GRADUAÇÃO ENGENHARIA E TECNOLOGIA ESPACIAIS' and a background image of a satellite. On the left, there is a search box labeled 'Busca' with the placeholder text 'Entre com os termos da busca' and a 'Buscar' button. The main content is a table-like structure with the following fields:

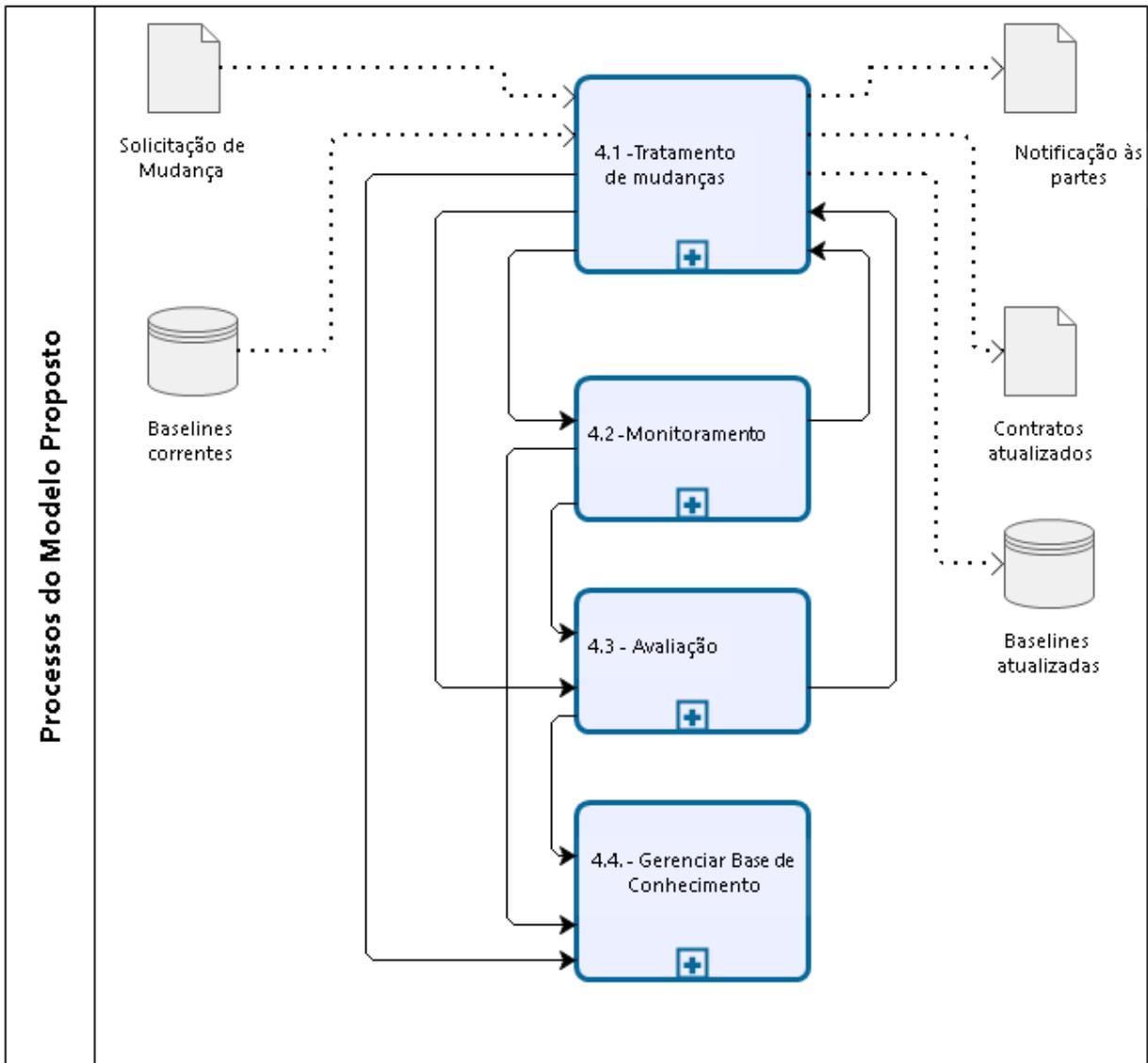
ID:	3377
Data:	2010-07-19
Submetido por:	KSC
Assunto:	Gerenciamento de Requisitos de Software
Problema encontrado:	Os métodos legados (manuais) para gerenciar a implementação de requisitos de software para o Programa de Ônibus Espacial tiveram um grande impacto no custo e no cronograma ao longo de todo o ciclo de desenvolvimento do software.
Solução:	NA
Recorrecia:	Nenhuma
Lições aprendidas:	Métodos manuais para o gerenciamento de requisitos de software são ineficazes e ineficientes, contribuindo para custos excessivos, bem como atrasos no cronograma. Os aspectos da gestão dos requisitos de software incluem a elicitação / especificação, análise, desenvolvimento, rastreamento e alteração dos requisitos de software utilizados durante as fases de implementação e manutenção do ciclo de vida do software. A gestão e a rastreabilidade dos requisitos de software são essenciais para o sucesso da produção de produtos de software confiáveis, de alta qualidade e seguros que atendam aos requisitos e necessidades dos usuários finais de uma forma econômica e oportuna. Os impactos de custo e agendamento resultantes de requisitos de software incompletos, incorretos ou em mudança aumentam mais tarde no ciclo de vida do software. A tecnologia, os processos e as ferramentas atuais do software fornecem métodos automatizados inovadores para facilitar o gerenciamento ótimo dos requisitos de software (por exemplo, IBM Rational DOORS, IBM Rational RequisitePro, software de gerenciamento de requisitos do Cradle). Além disso, uma relação colaborativa entre o cliente que usa o software e o desenvolvedor que fornece o software é primordial para o sucesso do projeto de software. Mais especificamente, os usuários / clientes devem efetivamente definir e comunicar com precisão seus requisitos ao desenvolvedor. Por exemplo, os requisitos definidos pelo usuário devem ser claramente definidos e inequívocos, concisos, completos, autônomos, capazes de serem implementados e testáveis.
Recomendações:	-Adotar e usar processos e ferramentas de software atuais e de última geração para gerenciar requisitos para desenvolvimento de software. -Valorizar e promover a relação de colaboração entre o cliente que usa o software e o desenvolvedor que fornece o software para garantir o sucesso do projeto de software.

Fonte: Elaborada pelo autor, com informações obtidas a partir do sítio público da NASA para demonstração.

4.5. Fluxo para aplicação do processo proposto

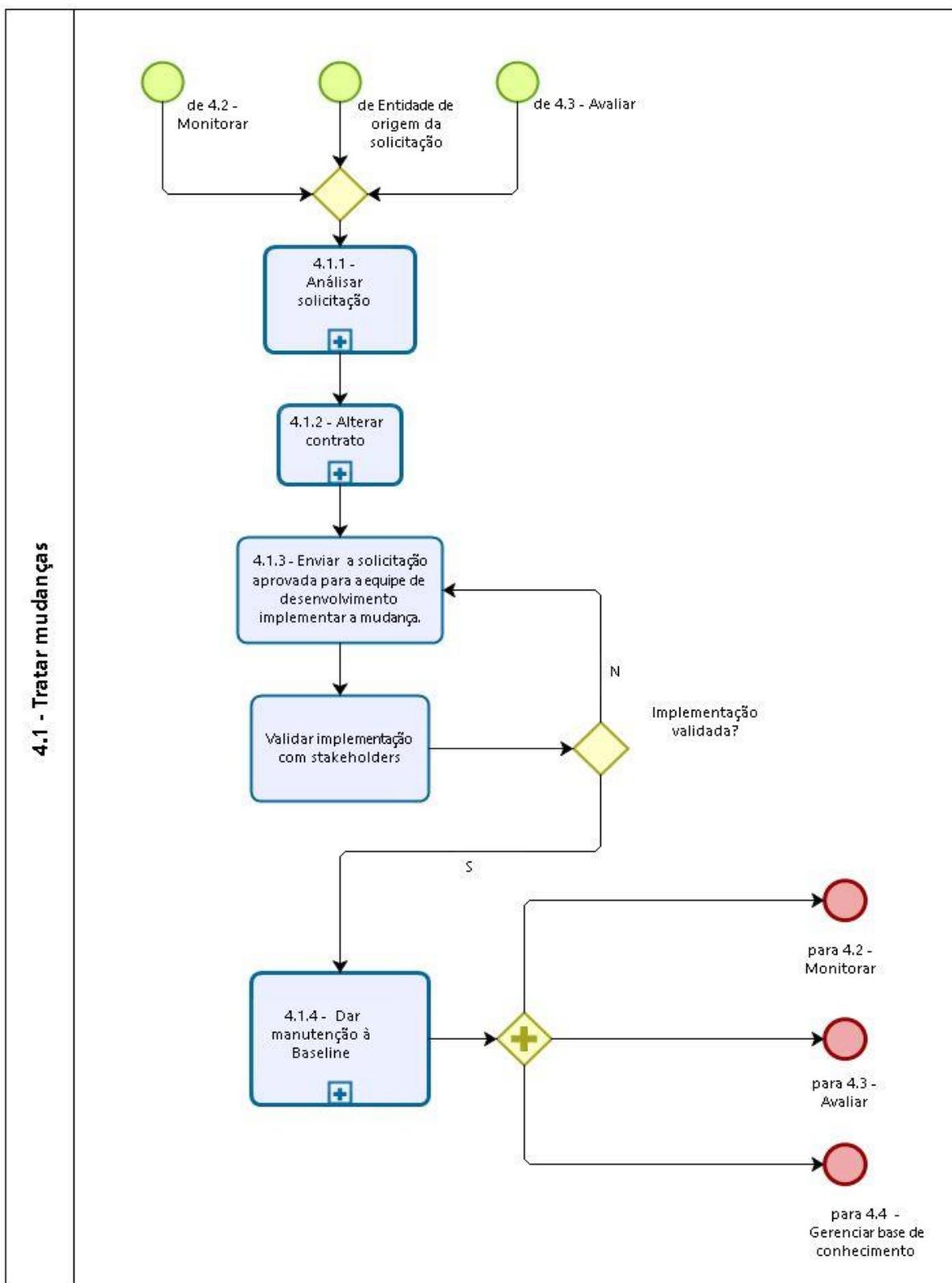
A seguir, são apresentados os fluxos com orientação do como realizar cada processo que compõe o processo proposto, através das Figuras 4.12, 4.13, 4.14, 4.15, 4.16, 4.17, 4.18, 4.19, 4.20, 4.21, 4.22.

Figura 4.12 - Processos do processo proposto.



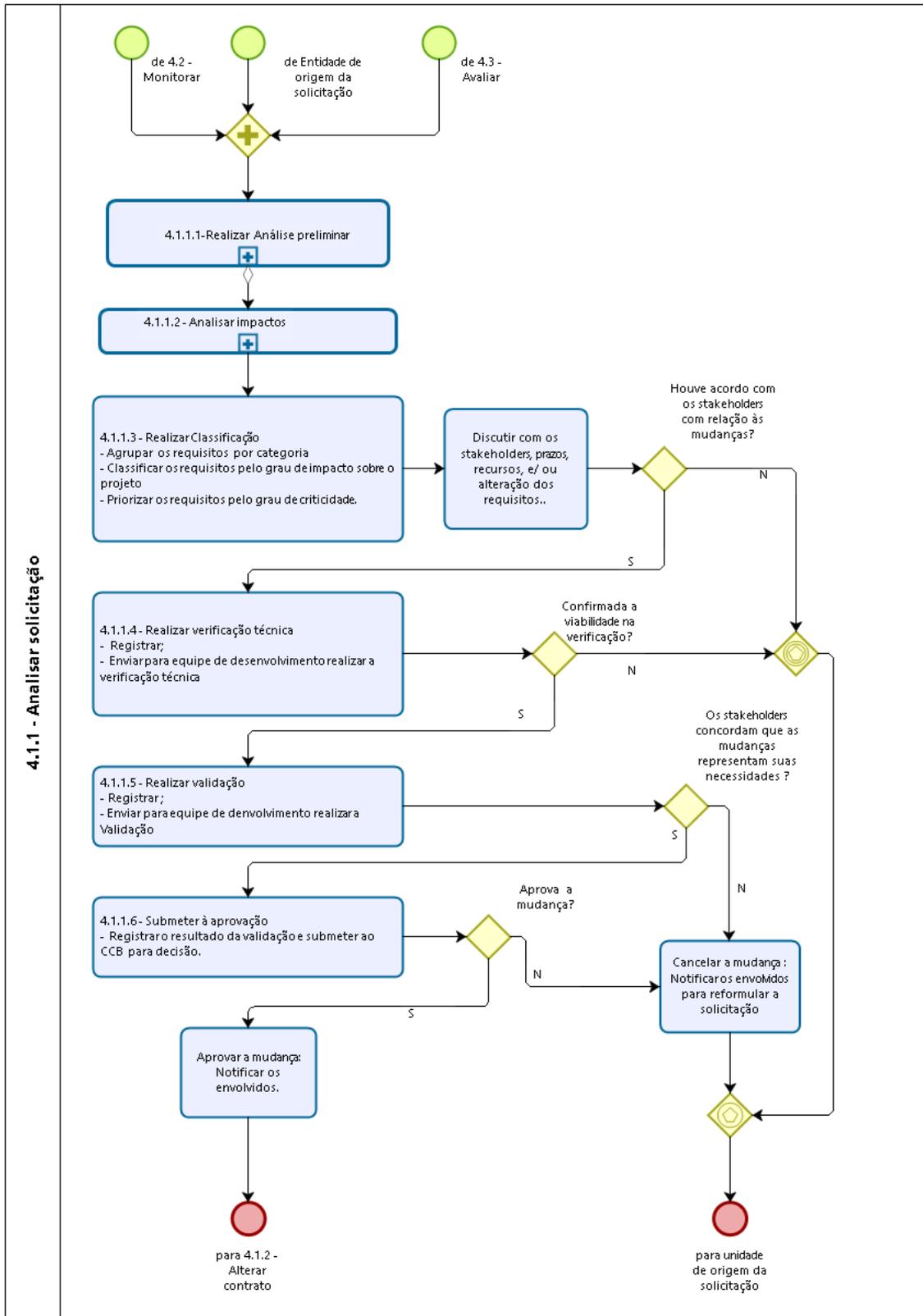
Fonte: Elaborada pelo autor.

Figura 4.13 - Processo 4.1 - Tratar mudanças.



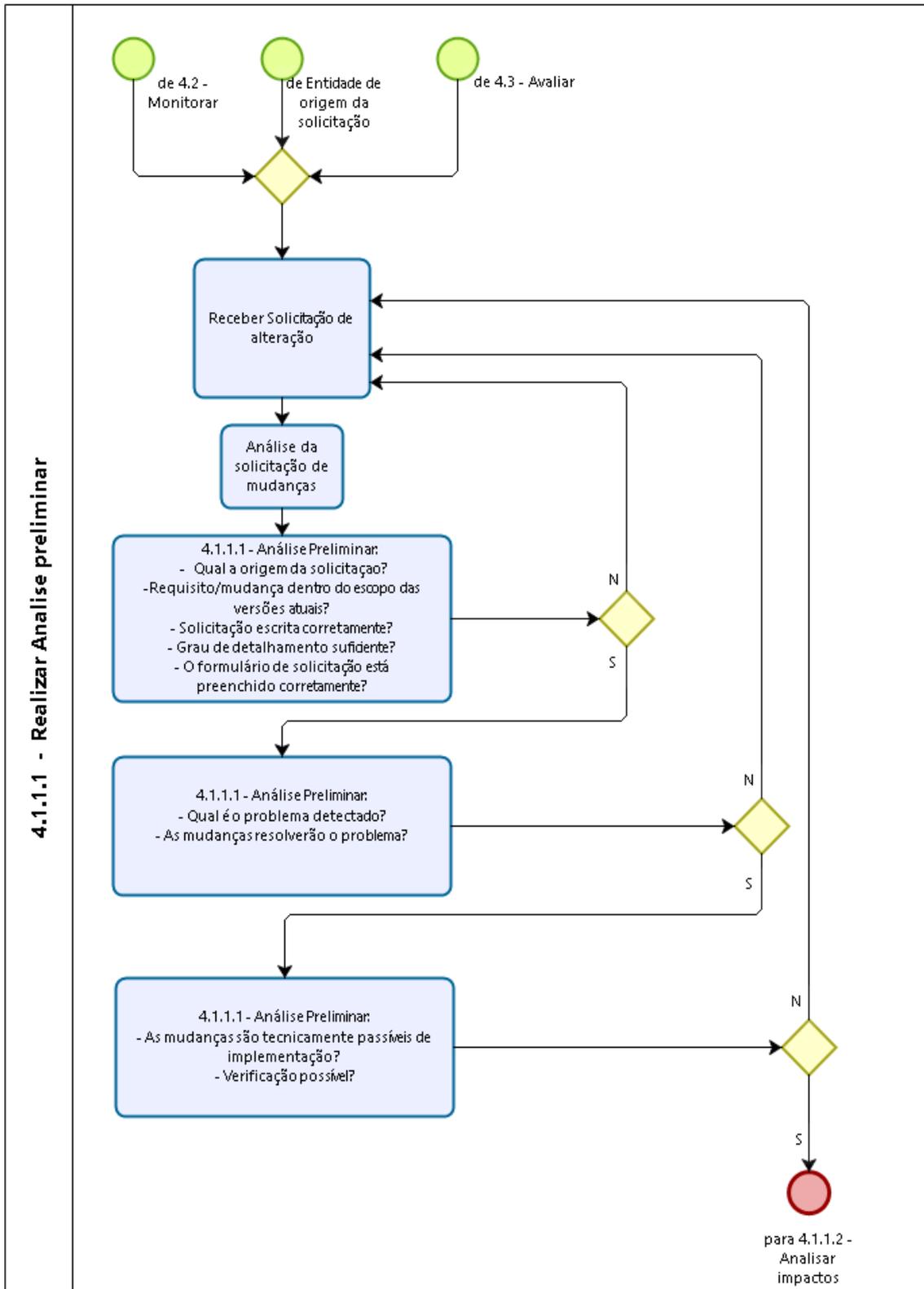
Fonte: Elaborada pelo autor.

Figura 4.14 - Processo 4.1.1 - Analisar solicitação.



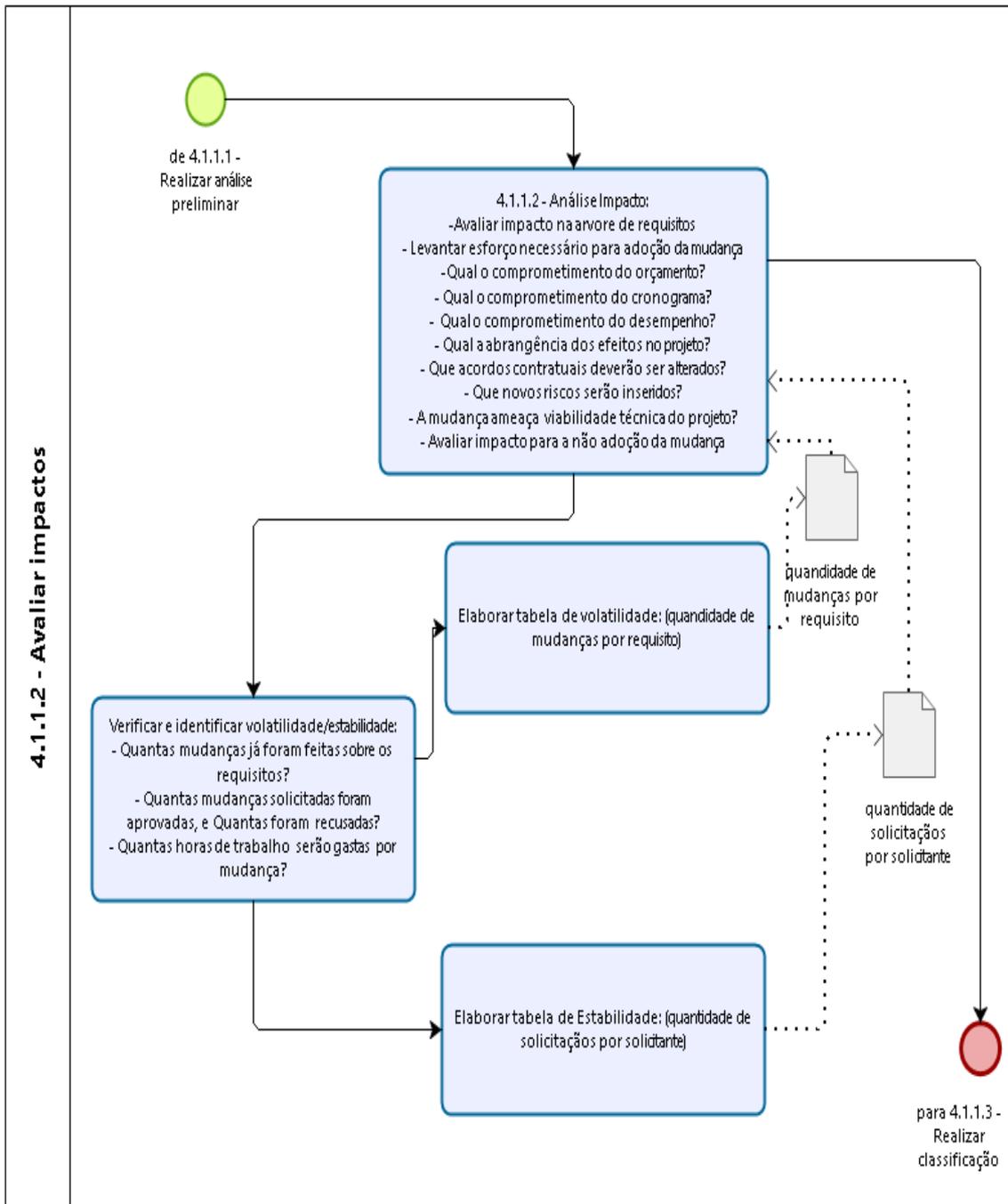
Fonte: Elaborada pelo autor.

Figura 4.15 - Processo 4.1.1.1 - Realizar análise preliminar.



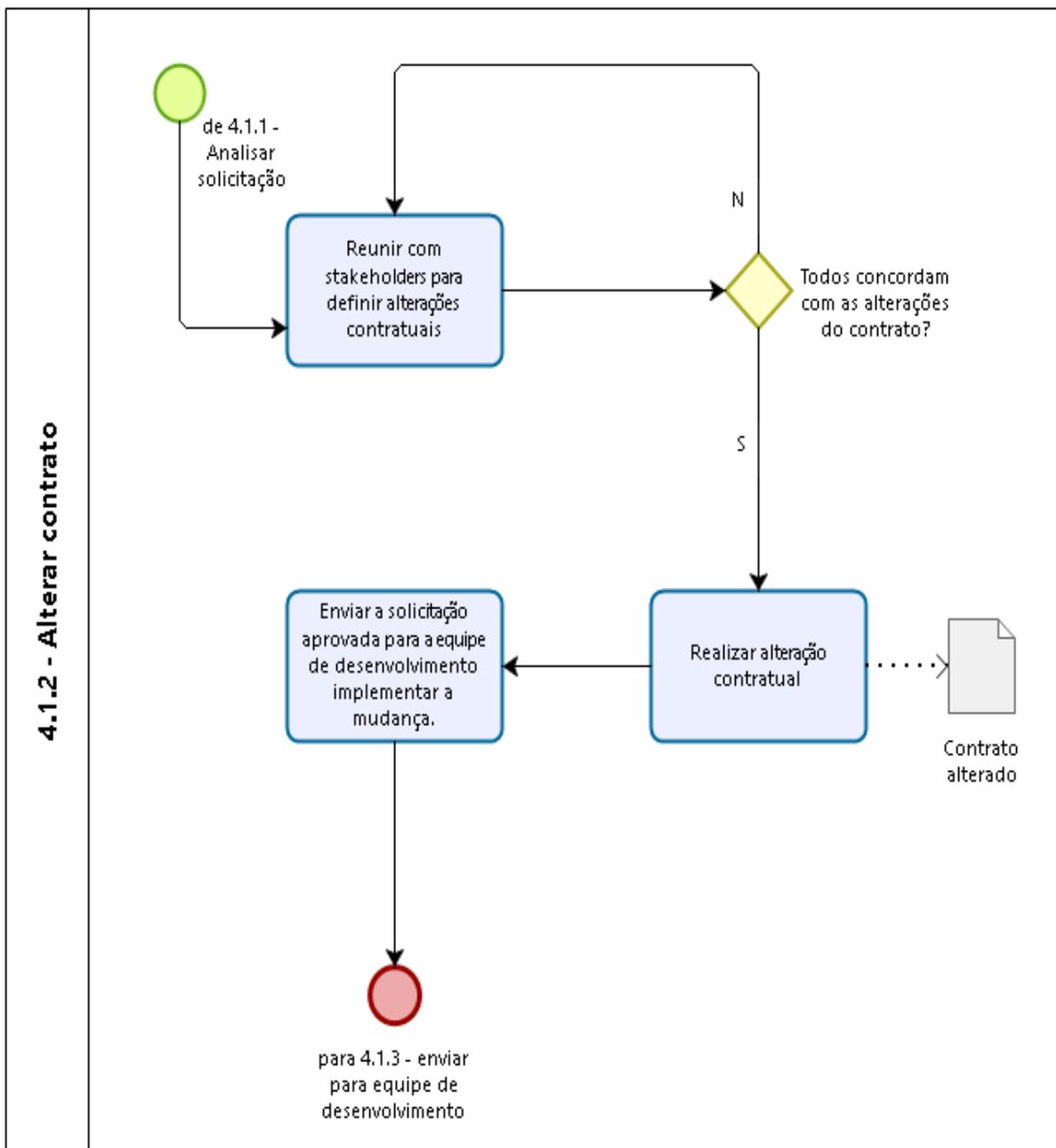
Fonte: Elaborada pelo autor.

Figura 4.16 - Processo 4.1.1.2 - Analisar impactos.



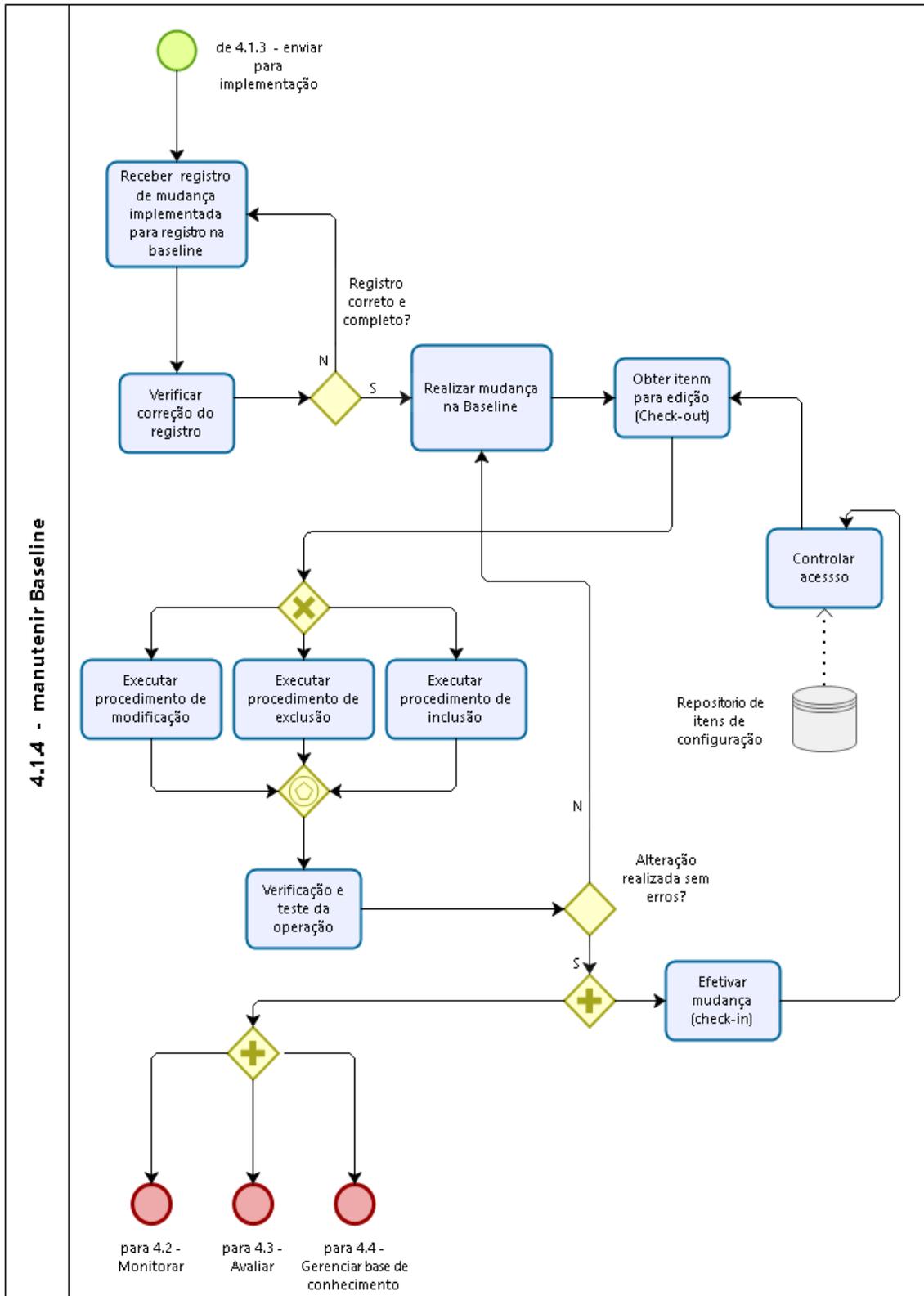
Fonte: Elaborada pelo autor.

Figura 4.17 - Processo 4.1.2 - Alterar contrato.



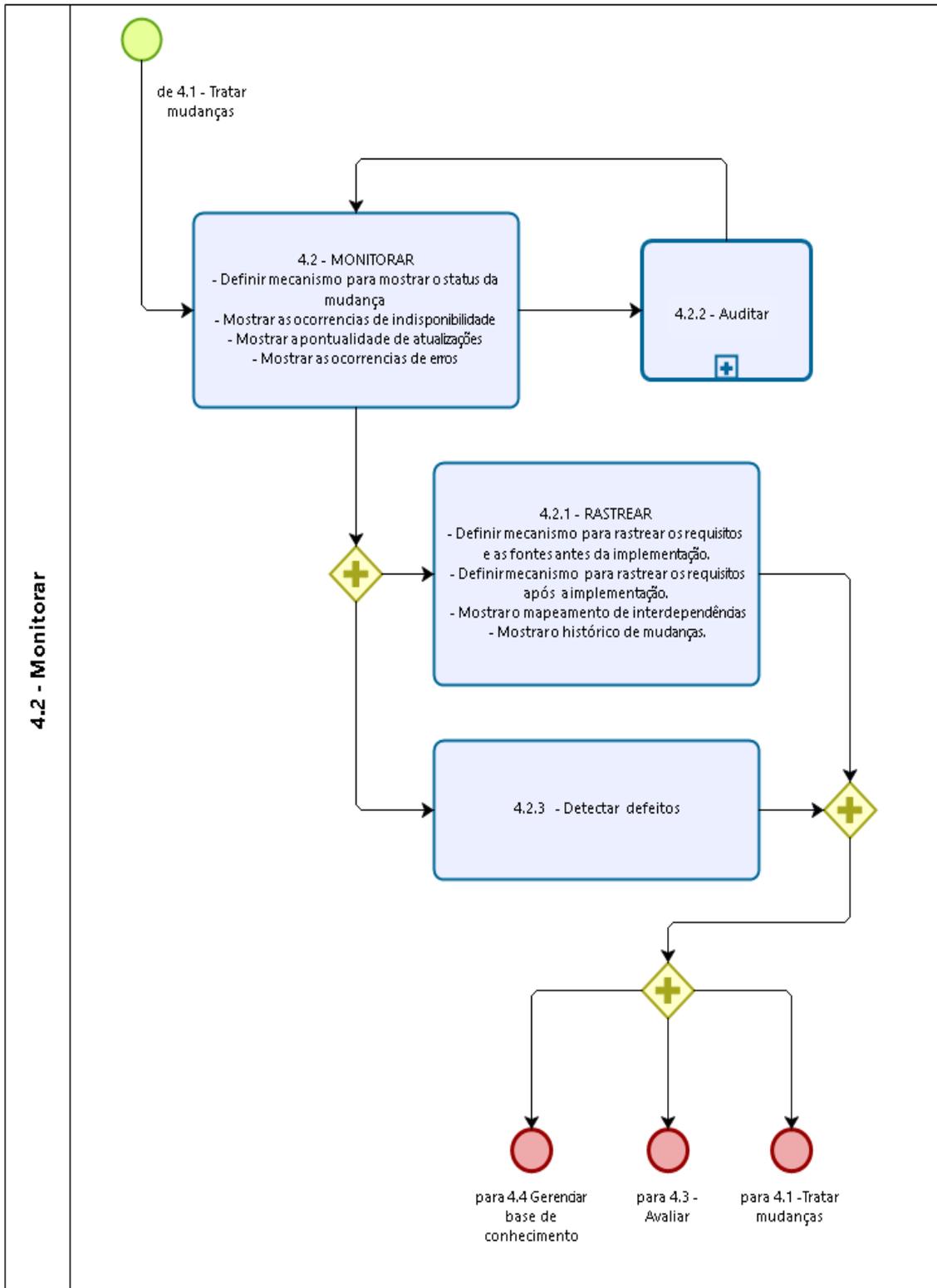
Fonte: Elaborada pelo autor.

Figura 4.18 - Processo 4.1.4 – Dar manutenção à *Baseline*.



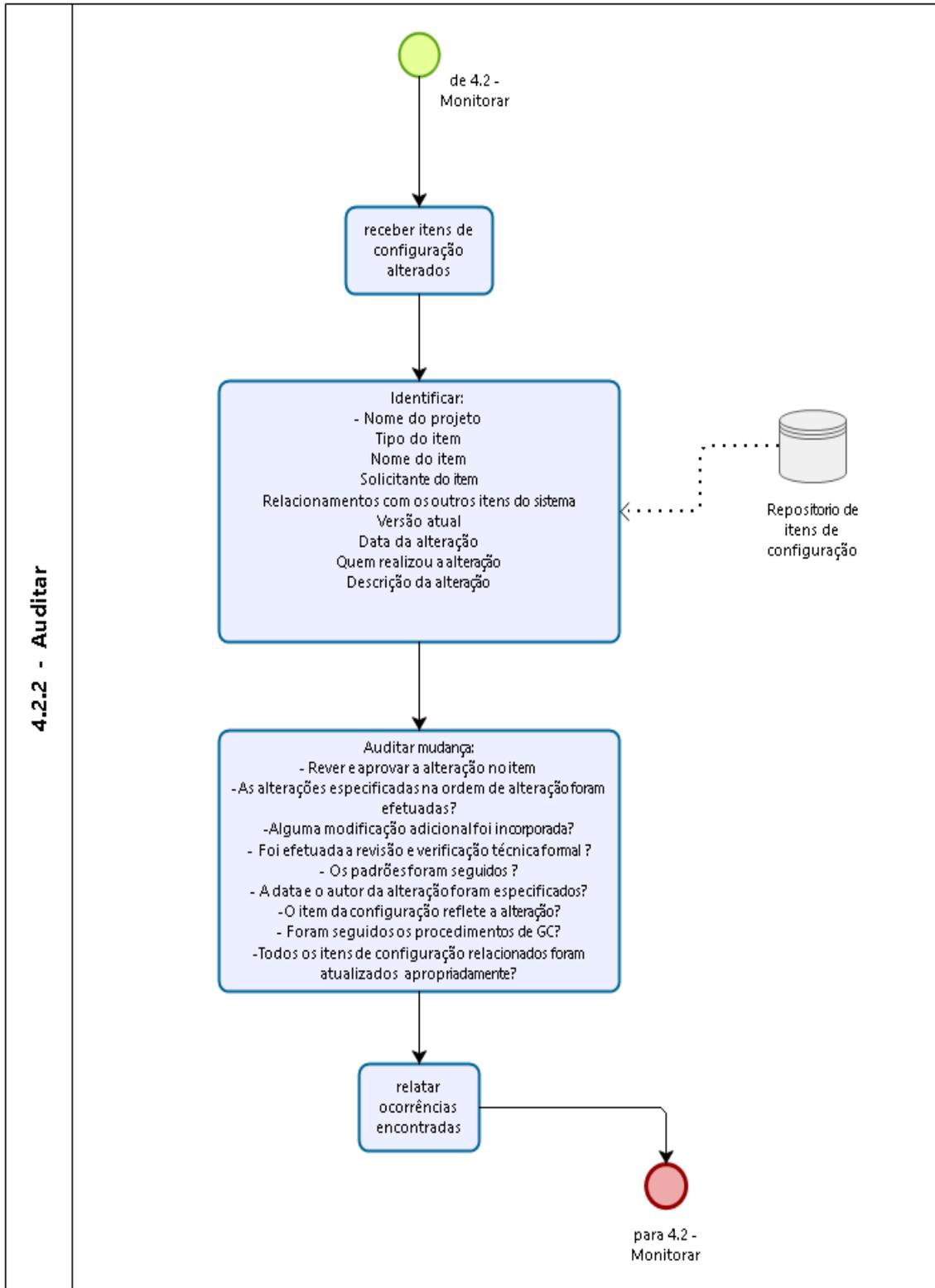
Fonte: Elaborada pelo autor.

Figura 4.19 - Processo 4.2 - Monitorar.



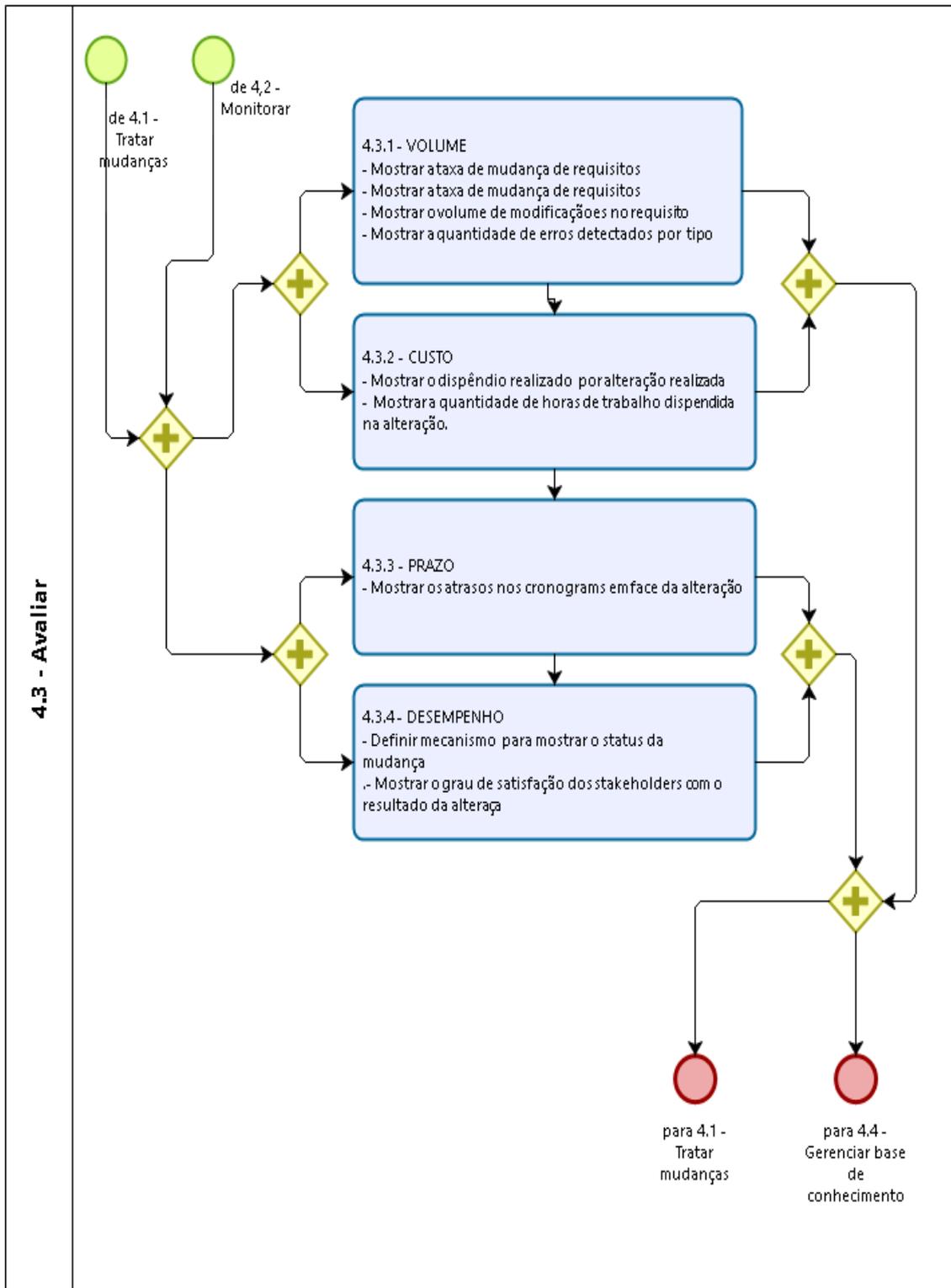
Fonte: Elaborada pelo autor.

Figura 4.20 - Processo 4.2.2 - Auditar.



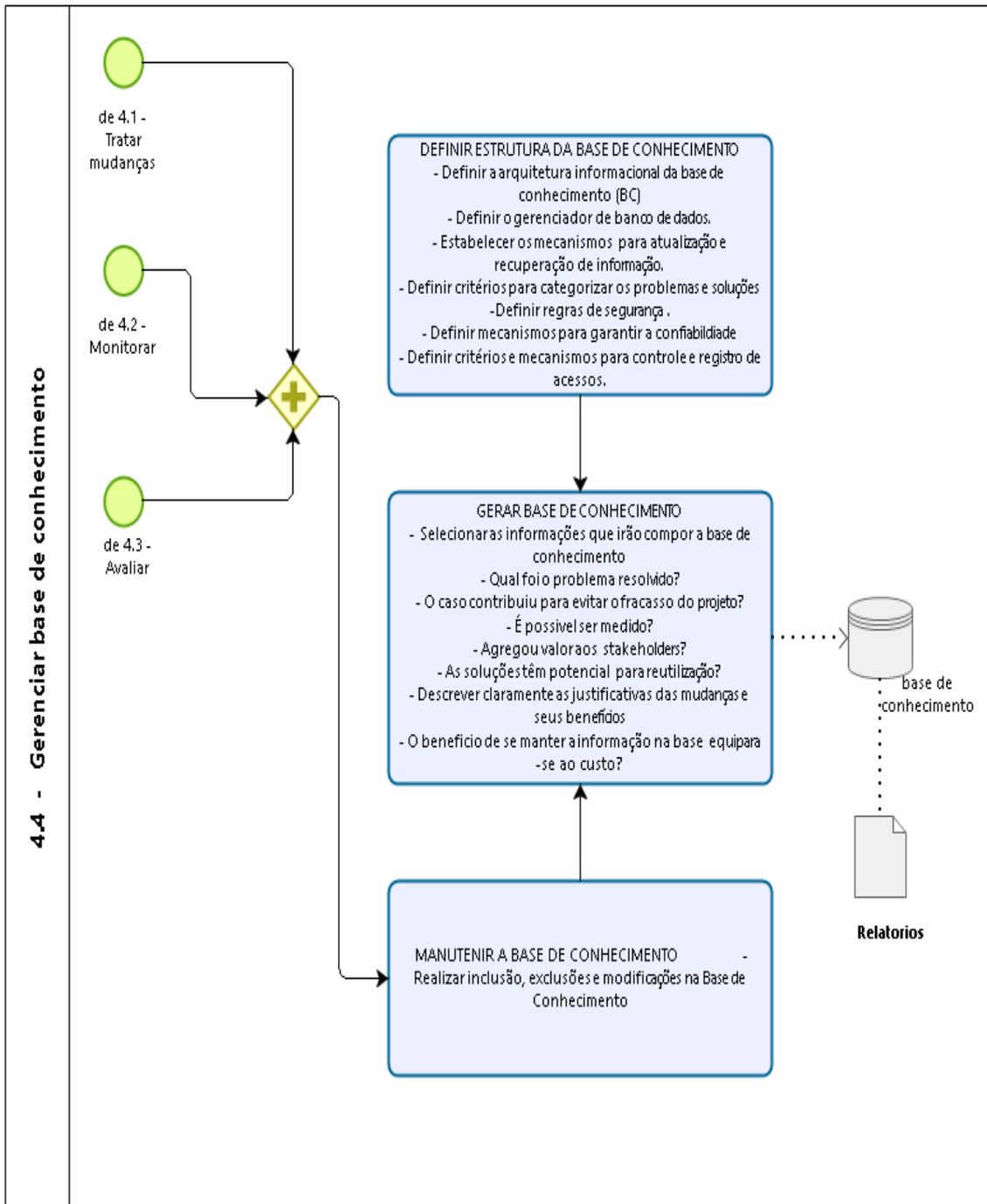
Fonte: Elaborada pelo autor.

Figura 4.21 - Processo 4.3 - Avaliar.



Fonte: Elaborada pelo autor.

Figura 4.22 - Processo 4.4 - Gerenciar base de conhecimento.



Fonte: Elaborada pelo autor.

5 DEMONSTRAÇÃO DA APLICAÇÃO DO PROCESSO PROPOSTO

O objetivo desta aplicação consiste na demonstração da viabilidade de implementação do modelo proposto. Buscou-se verificar a existência de recursos mínimos necessários para a execução das tarefas previstas no modelo.

Desta forma, o modelo proposto foi aplicado sobre um exemplo extraído de uma base de dados de requisitos existente no INPE, utilizada em estudos de simulação, para aprendizagem do uso do software PTC *Integrity*. As informações consideradas na demonstração do modelo proposto referem-se à simulação de uma solicitação de alteração do requisito Req3705m para aumento do tempo de gravação de imagens de 12 para 20 minutos.

O PTC *Integrity* é um software que gerencia processos de desenvolvimento de software e de todo o tipo de sistema conectando artefatos de engenharia incluindo requisitos, modelos, códigos, e planos de teste de forma a prover rastreabilidade durante o ciclo de vida do produto.

Desta forma, em função de sua disponibilidade no ambiente de desenvolvimento do INPE, será utilizado por esta pesquisa como ferramenta de suporte ao método proposto para prover a necessária rastreabilidade, que é fundamental para uma efetiva análise de impacto sobre alterações propostas.

O exemplo de aplicação foi utilizado buscando-se: 1) dados disponíveis em uma base de dados do PTC *Integrity*, que é utilizada para fins de estudo de implantação do software; e 2) informações fictícias adicionais para preenchimento dos formulários do modelo proposto. A base de informações simuladas são exemplos para utilização do suporte do PTC *Integrity* ao longo do processo de avaliação da solicitação, principalmente no que se refere à análise de impacto, que, conforme já citado, depende fortemente da rastreabilidade de requisitos.

É importante ressaltar que os exemplos aqui utilizados, tão somente serviram para identificar os pontos em comum dos processos praticados pelo INPE e o

modelo ora proposto, de forma a assegurar que todas as atividades que vêm sendo praticadas sejam abrangidas pelo modelo proposto e, ainda, revelar as melhorias que poderão ser implementadas no modelo atual praticado pelo INPE, no tocante ao gerenciamento de mudanças de requisitos.

Cabe notar que os requisitos utilizados para demonstração, embora se baseiem em elaboração de produtos, poderão ser utilizados em casos de desenvolvimento de software.

5.1. A análise da solicitação de mudança.

Foi simulada uma solicitação de alteração sobre o requisito original “*The satellite shall have the capability to record data during at least 12 minutes*” para “*The satellite shall have the capability to record data during at least 20 minutes*” O modelo propõe que a solicitação seja submetida a uma análise preliminar, análise de impacto, classificação, verificação, validação e aprovação, antes de ser efetuada qualquer alteração na *baseline*, contratual e/ou atividade de implementação.

Todos esses processos são contemplados pelo modelo adotado atualmente pelo INPE, porém o nível de formalização é diferente do proposto por esta pesquisa.

Para cada processo descrito ao longo da análise de solicitação foi proposto um formulário específico contendo informações importantes, tanto para rastreabilidade quanto para a formalização da documentação do projeto.

O que ocorre na prática são reuniões com diversos profissionais envolvidos nas áreas específicas para debaterem a validade e a aceitação do pedido de solicitação. Essas reuniões ao final emitem uma ata (minuta) com a decisão sobre a aceitação, ou não, do pedido submetido à análise.

Uma vez aceito o pedido de alteração, a decisão e a documentação são enviadas, também, para a verificação e validação, onde é feita a análise do relacionamento do requisito em questão com os demais artefatos do sistema.

O documento proposto para tratamento da solicitação de mudança receberá anexos relativos aos resultados das diversas análises previstas no processo de controle de mudanças.

A seguir, são apresentados os diversos formulários utilizados para registro das informações relativas ao pedido de alteração do requisito, na aplicação do modelo proposto para efeito demonstração.

O formulário inicial é o pedido de Proposta de Mudança de Requisito. Nele são fornecidas informações para identificação do requisito, projeto, descrição da mudança, justificativa da mudança a ser realizada no requisito, bem como a relação de artefatos ou itens do projeto que poderão ser, ou serão, afetados pela mudança solicitada. A Figura 5.1, a seguir, apresenta o formulário inicial do processo de solicitação de mudanças.

Figura 5.1 - Solicitação de mudança em requisito.

 Processo de Mudança de Requisito Página 1 de 8 	
Proposta de mudança de requisito	
Identificação:	
Número da solicitação: <input type="text" value="867_Req3705_01"/>	Solicitante: <input type="text" value="Coordenação Geral de Engenharia e Tecnologia Espacial-ETE"/>
Data de emissão: 10/04/2017	Projeto: <input type="text" value="867"/>
Status da solicitação:	
<input checked="" type="radio"/> Recebido <input type="radio"/> Em análise <input type="radio"/> Verificado <input type="radio"/> Validado <input type="radio"/> Aprovado <input type="radio"/> Rejeitado <input type="radio"/> Implementado <input type="radio"/> Cancelado <input type="radio"/> Adiado	
Detalhamento da solicitação:	
Título da mudança: <input type="text" value="Aumento da capacidade de gravação de dados"/>	
Descrição da mudança: <input "the="" 12="" 20="" at="" capability="" data="" during="" have="" least="" minutes"="" para="" record="" satellite="" shall="" the="" to="" type="text" value="Alteração do requisito de satellite Req3705 com o aumento do tempo de gravação de 12 para 20 minutos. Requisito original: "/>	
Anexos: <input type="radio"/> Sim <input checked="" type="radio"/> Não	
Descrição dos anexos: <input type="text"/>	
Justificativa para a mudança: <input type="text" value="O tempo de gravação de dados para dados obtidos pela camera do satellite deve ser aumentado para 20 minutos em função do aumento da área a ser imageada"/>	
Itens afetados pela mudança: <input type="text" value="1) Verification Method - User acceptance Test 5108; 2) Verification Method - Verification Test 4146; 3) Verification Method - Verification Test 4148; 4) Verification Method - System Test 5106; 5) Requirement - Functional requirement 6424"/>	
Documentos afetados pela mudança: <input type="text" value="Requirement specification document 1339"/>	
Anexos: <input checked="" type="radio"/> Sim <input type="radio"/> Não	
Descrição dos anexos: <input type="text"/>	

Fonte: elaborada pelo autor.

No exemplo demonstrado temos:

Justificativa de mudança: O tempo de gravação de dados para dados obtidos pela câmera do satélite deve ser aumentado para 20 minutos em função do aumento da área para obtenção de imagens.

Os itens afetados pela mudança podem ser fornecidos inicialmente numa visão imediata do proponente; no entanto, pelo grau de complexidade do projeto podem existir relacionamentos não identificados, por isto, a importância de utilização de uma ferramenta de software para suporte se apresenta como indispensável. No exemplo, as informações incluídas foram obtidas a partir da consulta à base de dados do PTC *Integrity*, que fornece a lista de itens afetados. Neste caso, os testes de verificação e requisitos que se apresentam como suspeitos, como segue:

- 1) *user acceptance test 5108;*
- 2) *verification test 4146;*
- 3) *verification test 4148;*
- 4) *system test 5106;*
- 5) *requirement - functional requirement 6424.*

Além disto, a consulta ao PTC *Integrity* forneceu, também, os documentos que podem sofrer alteração devido à mudança solicitada. Neste caso, apenas o documento de especificação dos requisitos (*Requirements Specification Document 1339*) sofreu alteração.

A análise preliminar formalizada pelo formulário na Figura 5.2 certifica que o documento de proposta de mudança de requisito está correto e completo e, desta forma, prossegue-se para a análise de impacto.

Figura 5.2 - Análise preliminar.

ANÁLISE PRELIMINAR:	
Identificação:	
Análise preliminar:	<input checked="" type="radio"/> Aprovado <input type="radio"/> Rejeitado
Numero da solicitação:	<input type="text" value="867 Req3705 01"/>
Data de entrada:	<input type="text" value="10/04/2017"/>
Data de saída:	<input type="text" value="11/04/2017"/>
- A origem da solicitação está corretamente identificada?	<input checked="" type="radio"/> Sim <input type="radio"/> Não
- A mudança solicitada está dentro do escopo das versões atuais?	<input checked="" type="radio"/> Sim <input type="radio"/> Não
- O grau de detalhamento da solicitação é adequado?	<input checked="" type="radio"/> Sim <input type="radio"/> Não
- A proposta de novo requisito está escrita corretamente? (ver ajuda de escrita de requisitos)?	<input checked="" type="radio"/> Sim <input type="radio"/> Não
- A mudança proposta resolve o problema?	<input checked="" type="radio"/> Sim <input type="radio"/> Não
- A mudança é tecnicamente possível de ser implementada?	<input checked="" type="radio"/> Sim <input type="radio"/> Não
Observações: <input type="text"/>	
Anexos:	<input type="radio"/> Sim <input checked="" type="radio"/> Não
Descrição dos anexos:	<input type="text"/>
Responsável pela análise preliminar:	<input type="text" value="Fernando Azevedo"/>

Fonte: elaborada pelo autor.

5.2. Análise de impacto

Na avaliação de impacto os analistas de diversas áreas envolvidas buscam levantar a propagação do efeito que uma mudança de um requisito pode causar sobre o projeto. Diversos aspectos são avaliados para mensurar os possíveis retornos e perdas potenciais, riscos, desafios técnicos, negociações com as partes envolvidas e com interesse no sistema e também, ao contrário,

deve-se avaliar quais são as consequências de não efetuar a alteração proposta.

Neste momento, são coletadas e processadas as informações relevantes para o estudo e, neste caso, apenas para exemplo, foram inseridas estimativas de custos da mudança em R\$ 50.000,00; estimativa de esforço a ser empreendido na implementação da mudança em 120 horas de trabalho e com previsão de atraso no cronograma estimado em 12 dias.

Obtém-se, também, a confirmação da necessidade de alteração dos contratos já pactuados e a certificação de que a mudança solicitada não agrega risco ao projeto nem ameaça a sua viabilidade técnica, e ainda, a confirmação de que a análise de impacto foi considerada, também, sob a hipótese de não se fazer a alteração. Tal mudança foi classificada como de criticidade baixa por não agregar riscos nem comprometer tecnicamente o projeto e prioridade média, considerando que provocará melhoria nos resultados da atuação do satélite, uma vez que haverá maior abrangência da área para obtenção de imagens. A Figura 5.3 apresenta o formulário preenchido com o exemplo para análise de impacto.

Figura 5.3 – Análise de impacto.

Processo de Mudança de Requisito Página 3 de 8

ANÁLISE DE IMPACTO

Análise de impacto: Aprovado Aprovado com restrições Rejeitado

Restrições:

Número da solicitação:

Data de entrada: 12/04/2017
Data de saída: 20/04/2017

Custo total de esforço estimado (em horas):

Custo total (em R\$):

Atraso estimado no cronograma (em dias):

- Acordos contratuais serão atingidos pela alteração? Sim Não

- A mudança acrescenta algum novo risco ao projeto? Sim Não

- A mudança ameaça a viabilidade técnica do projeto? Sim Não

- O impacto da não adoção da mudança proposta foi analisado? Sim Não

Observações

Anexos: Sim Não

Descrição dos anexos:

Classificação:

Críticidade: Alta Média Baixa

Prioridade: Alta Média Baixa

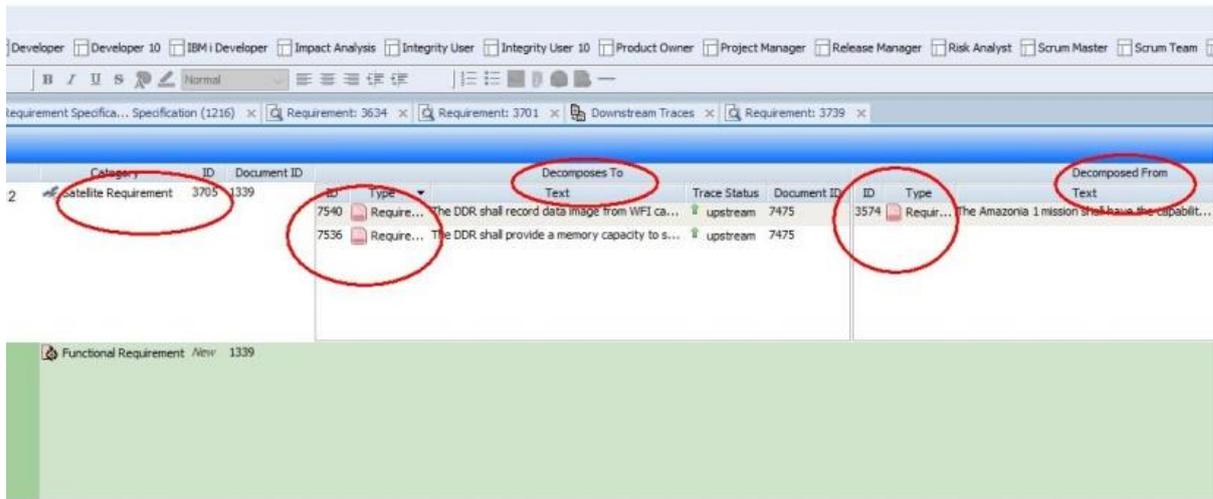
Responsável pela análise de impacto:

Fonte: elaborada pelo autor.

As figuras, a seguir, apresentam algumas telas capturadas do PTC *Integrity*, ilustrando a questão da rastreabilidade do requisito em estudo e a propagação dos efeitos na árvore de requisitos e documentos.

A Figura 5.4 mostra o documento de especificação de requisitos onde se apresenta o Req7502 e os requisitos do qual ele é decomposto e para quais ele se decompõe.

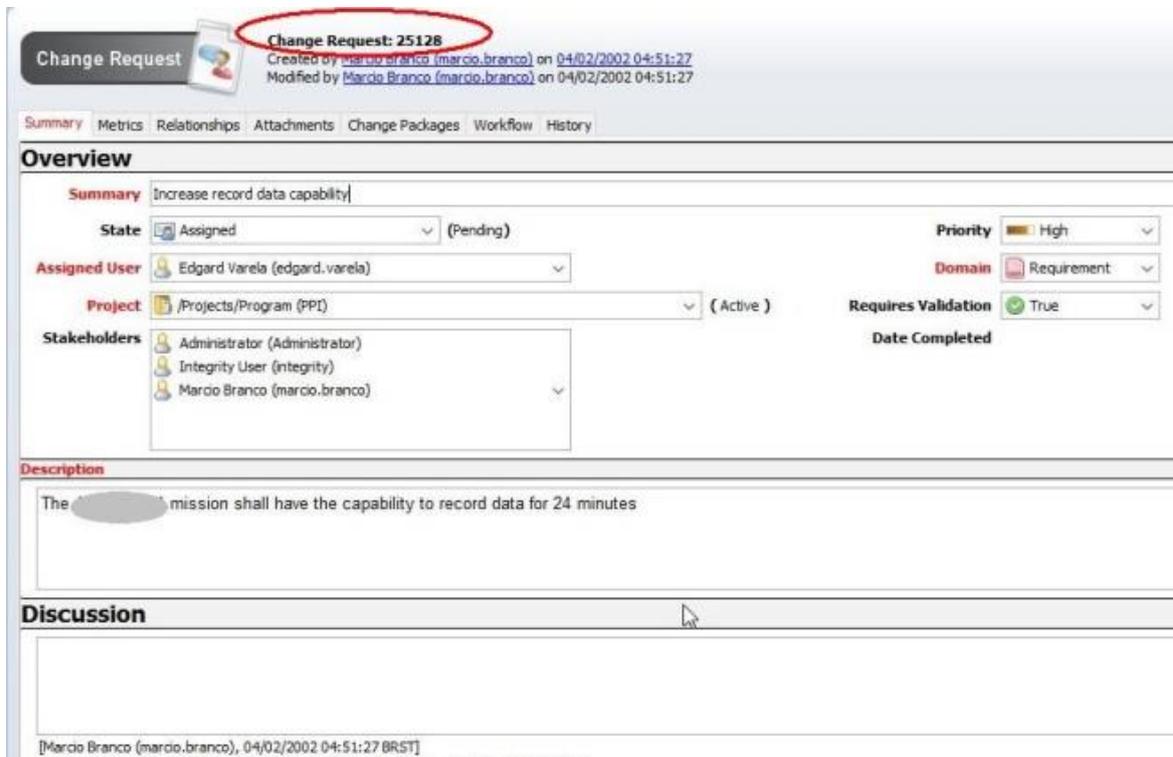
Figura 5.4 - Documento de especificação de requisitos.



Fonte: elaborada pelo autor.

A Figura 5.5 mostra a tela de proposta de mudança de requisito (*Change Request* - CR) do PTC *Integrity*. Este CR deverá, em seguida, ser conectado ao requisito para o qual se propõe a mudança.

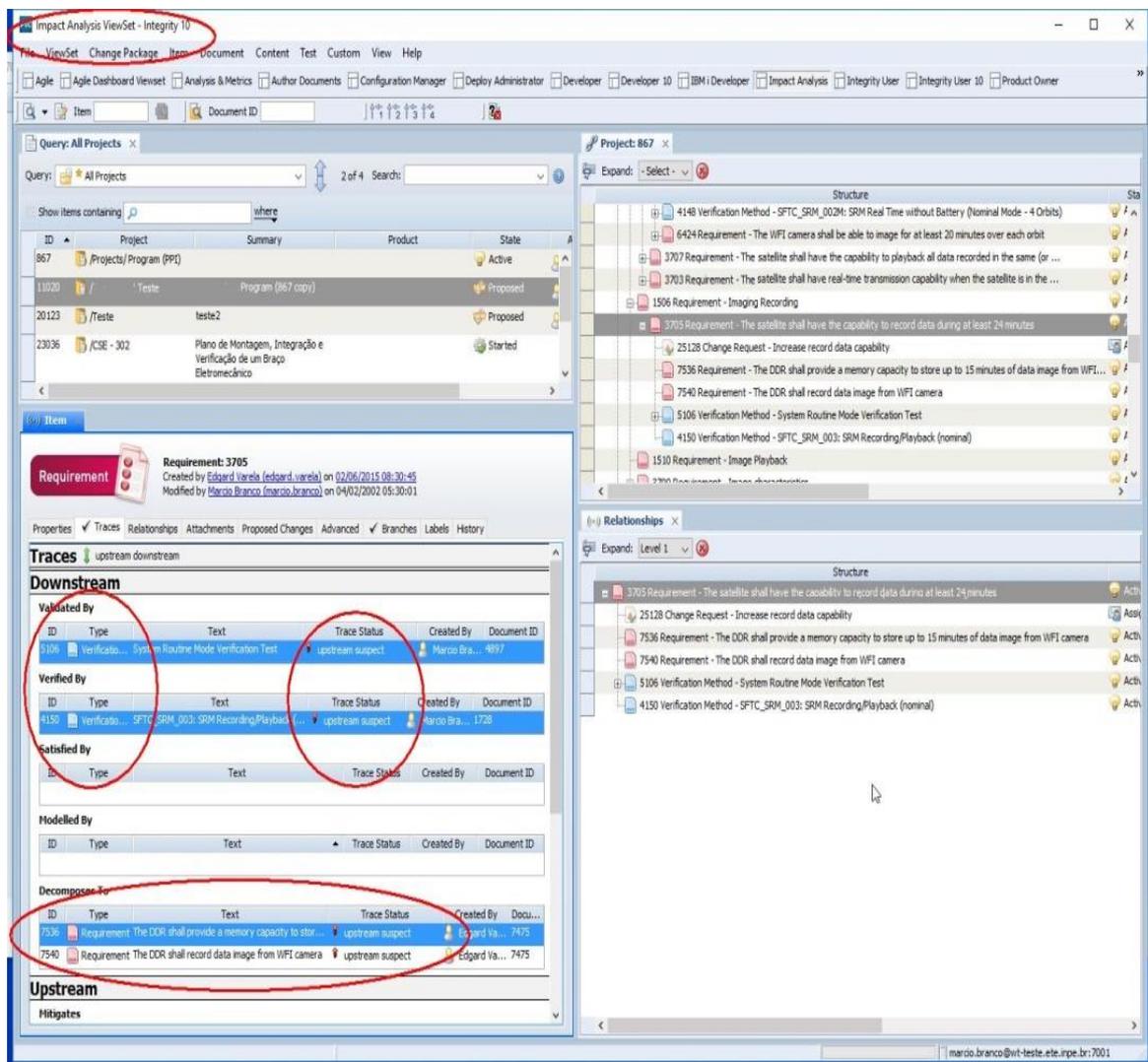
Figura 5.5 – Detalhe de tela de proposta de mudança de requisito.



Fonte: elaborada pelo autor.

A Figura 5.6 mostra a tela do módulo para auxílio à análise de impacto.

Figura 5.6 - Tela de módulo de análise de impacto.



Fonte: elaborada pelo autor.

A Figura 5.7 mostra em detalhe a tela do módulo de análise de impacto onde os requisitos e outros artefatos foram automaticamente marcados como “suspeitos” por estarem conectados ao requisito Req3705, após a solicitação de mudança ter sido conectada ao Req3705.

Figura 5.7 - Detalhe da tela de análise de impacto.

Requirement: 3705
Created by [Edgard Varela \(edgard.varela\)](#) on 02/06/2015 08:30:45
Modified by [Marco Branco \(marco.branco\)](#) on 04/02/2002 05:30:01

Properties Traces Relationships Attachments Proposed Changes Advanced Branches Labels History

Traces upstream downstream

Downstream

Validated By

ID	Type	Text	Trace Status	Created By	Document ID
5106	Verificatio...	System Routine Mode Verification Test	upstream suspect	Marco Bra...	4897

Verified By

ID	Type	Text	Trace Status	Created By	Document ID
4150	Verificatio...	SFTC_SRM_003: SRM Recording/Playback (...)	upstream suspect	Marco Bra...	1728

Satisfied By

ID	Type	Text	Trace Status	Created By	Document ID
----	------	------	--------------	------------	-------------

Modelled By

ID	Type	Text	Trace Status	Created By	Document ID
----	------	------	--------------	------------	-------------

Decomposes To

ID	Type	Text	Trace Status	Created By	Docu...
7536	Requirement	The DDR shall provide a memory capacity to stor...	upstream suspect	Edgard Va...	7475
7540	Requirement	The DDR shall record data image from WFI camer...	upstream suspect	Edgard Va...	7475

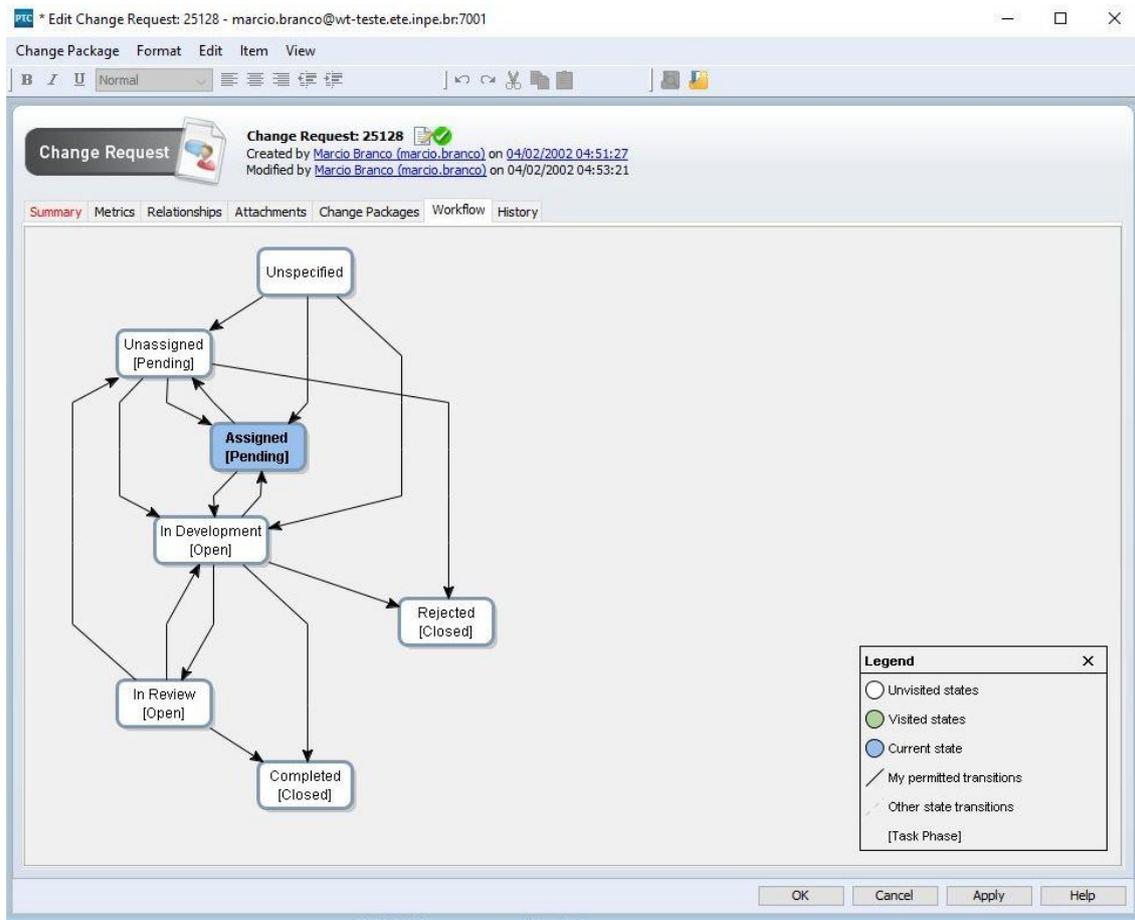
Upstream

Mitigates

Fonte: elaborada pelo autor.

A Figura 5.8 mostra, em detalhe, a tela do acompanhamento do *status* de requisição de mudança Req3705.

Figura 5.8 - Detalhe da tela de análise de impacto.



Fonte: elaborada pelo autor.

A análise de impacto pode levar à conclusão que determinada mudança não é compensatória; e neste, caso podem ocorrer os desvios e/ou *waivers*, que serão documentados no processo.

5.3. Análise de verificação e validação da mudança

Os processos de avaliação de relacionamentos e rastreamento de requisitos e testes de verificação são atualmente realizados com o suporte do software PTC *Integrity*.

Desta forma, em função de sua disponibilidade no ambiente de desenvolvimento do INPE, ele foi utilizado por esta pesquisa como ferramenta

de suporte ao método proposto, para prover a necessária rastreabilidade que é fundamental para uma efetiva análise de impacto sobre alterações propostas.

O formulário de verificação registrou que foram realizados os testes de verificação 4146 e 4148, mencionando que foi observada a rastreabilidade do processo de alteração, agregando ao formulário os resultados dos testes realizados de forma a verificar o novo requisito proposto.

Prosseguindo no formulário de verificação está indicado que foram anexados, também, os resultados dos testes de validação.

Considerando que todos os testes e informações colhidas desde a análise preliminar até os testes de validação, segue-se o registro no formulário de aprovação do CCB, que contém as informações de aprovação e o registro de encaminhamento dos documentos para as áreas competentes para alteração contratual, implementação, e manutenção da *baseline* de requisitos e procedimentos e registros previstos em tais áreas. As Figuras 5.9, 5.10 e 5.11, a seguir, mostram, respectivamente, os formulários de verificação e validação de requisitos e de aprovação pelo CCB preenchidos conforme os dados do exemplo considerado.

Figura 5.9 - Formulário de verificação.

Processo de Mudança de Requisito	
Página 4 de 8	
VERIFICAÇÃO	
Verificação:	<input checked="" type="radio"/> Aprovado <input type="radio"/> Rejeitado
Número da solicitação:	<input type="text" value="867_Req3705_01"/>
Data de entrada:	21/04/2017
Data de saída:	24/04/2017
Observações:	<input type="text" value="Foram realizados os testes de verificação 4146 e 4148 conforme informações de rastreabilidade do requisito em processo de alteração"/>
Anexos da verificação:	<input checked="" type="radio"/> Sim <input type="radio"/> Não
Descrição dos anexos:	<input type="text" value="Resultados do testes realizados"/>
Responsável pela verificação:	<input type="text" value="Fernando Azevedo"/>

Fonte: elaborada pelo autor.

Figura 5.10 - Formulário de validação.

Processo de Mudança de Requisito	
Página 5 de 8	
Validação	
Verificação:	<input checked="" type="radio"/> Aprovado <input type="radio"/> Rejeitado
Número da solicitação:	<input type="text" value="867_Req3705_01"/>
Data de entrada:	20/04/2017
Data de saída:	26/04/2017
Observações:	<input type="text"/>
Anexos da validação:	<input type="radio"/> Sim <input type="radio"/> Não
Descrição dos anexos:	<input type="text"/>
Responsável pela validação:	<input type="text" value="Fernando Azevedo"/>

Fonte: elaborada pelo autor.

Figura 5.11 - Formulário de aprovação do CCB.

Processo de Mudança de Requisito	
Página 6 de 8	
Aprovação do CCB	
Aprovação do CCB:	
<input checked="" type="radio"/> Aprovado <input type="radio"/> Rejeitado	
Projeto	<input type="text" value="867"/>
Número da solicitação:	<input type="text" value="867_Req3705_01"/>
Responsável CCB:	<input type="text" value="Fernando Azevedo"/>
Enviar para alteração contratual:	
<input checked="" type="radio"/> Enviado	
Data de envio: 28/04/2017	
<input type="radio"/> Aguardando informação	
Data: Clique aqui para inserir uma data.	
Enviado para implementação:	
<input checked="" type="radio"/> Enviado	
Data de envio: 28/04/2017	
<input type="radio"/> Aguardando informação	
Data: Clique aqui para inserir uma data.	
Enviado para manutenção de baseline de requisitos para:	
<input type="radio"/> Inclusão <input type="radio"/> Exclusão <input type="radio"/> Modificação	
Data de envio: 28/04/2017	
<input type="radio"/> Aguardando informação	
Data: Clique aqui para inserir uma data.	

Fonte: elaborada pelo autor.

5.4. Geração e manutenção de *baseline*.

No tocante à *baseline*, o que é feito atualmente no INPE é um controle focado na versão de documentos usando o software PTC *Windchill* como ferramenta de suporte. Tal procedimento não se caracteriza como um processo de gerenciamento puro de requisitos, uma vez que o controle não se baseia nos requisitos e nos seus relacionamentos, constantes em tais documentos.

No processo corrente o controle das mudanças solicitadas parte da identificação dos documentos que contém os requisitos a serem alterados. É

emitida uma ECR que, juntamente com o documento original, o documento modificado e a ata com a decisão do CCB são enviados para o setor de controle de documentação que providencia a atualização da versão dos documentos no PTC *Windchill* e é emitida a notificação para as partes interessadas. Cabe neste momento uma breve descrição das funcionalidades da ferramenta PTC *Windchill*.

O PTC *Windchill* é uma solução PLM (*Product Lifecycle Management*) que realiza todo o gerenciamento de produtos e processos de negócios, ao longo de todo o ciclo de vida dos produtos e serviços, baseada em um repositório central de informações de produtos e serviços. O PTC *Windchill* permite uma abordagem sistemática para a criação, configuração, gerenciamento e reutilização de estruturas e informações associadas ao produto, tais como arquivos gerados pelo CAD (*Computer Aided Design*), documentação, requisitos, informações de fabricação, informações de serviços, dados de peças/fornecedor, cálculos e ilustrações.

Finalmente, propõe-se a construção de uma *baseline* de requisitos em que possam ser realizadas consultas *online* e ter-se controle e segurança do que está sendo alterado, com a manutenção de históricos de modificações para uso de todas as partes envolvidas no processo de desenvolvimento. Isto não impede a continuidade da execução das atividades ora praticadas para controle de documentos, já que os documentos lastrearão as informações contidas na *baseline*.

5.5. Monitoramento

O aspecto de monitoramento proposto por este trabalho consiste na utilização de dados obtidos pela rastreabilidade e, eventualmente, verificações sobre o comportamento dos requisitos para atender solicitações da auditoria e do subprocesso de detecção de defeitos, os quais podem ser suportados na prática, como já foi mencionado, pelo software PTC *Integrity*.

As atividades normalmente previstas em processos de monitoramento objetivam identificar estatísticas sobre o comportamento dos requisitos após a implementação. Qualquer problema ou defeito detectado deve ser comunicado imediatamente ao CCB para solução e atualização das informações juntos às diversas áreas.

Neste caso demonstrativo, nenhum erro ou defeito foi mencionado pelo monitoramento após a implementação da mudança.

5.6. Avaliação

O processo de avaliação pode ser suportado, também, pelo PTC *Integrity*, pois este fornece campos onde métricas, úteis para a avaliação de custos e comprometimento de cronograma, podem ser associadas aos requisitos.

É importante ressaltar que o formulário de registro de avaliação, no caso demonstrativo, apresentou informações que indicaram que o atraso previsto de 12 dias no cronograma não comprometeu o cumprimento do cronograma, devido aos ajustes nas tarefas restantes do projeto. Não houve comprometimento técnico, a produtividade da equipe foi mantida estável e, embora tenha havido esforço adicional, em termos relativos, não ocorreu alteração significativa na produtividade.

Nota-se que este formulário foi posicionado após a aprovação da alteração, porque suas informações são repassadas para o processo de monitoramento, que ocorre após a implementação das alterações. Isto porque o processo de avaliação gera informações que são utilizadas em momentos diferentes por diversos processos. Parte das informações geradas é utilizada no processo de análise da proposta, subsidiando o CCB na tomada de decisão. Da mesma forma, parte das informações também é disponibilizada para a auditoria e para a área responsável pela manutenção e gerenciamento da base de conhecimento.

A Figura 5.12, a seguir, mostra o resultado da avaliação do exemplo demonstrativo.

Figura 5.12 - Formulário de resultado de avaliação.

Processo de Mudança de Requisito	
Página 7 de 8	
Resultado da Avaliação	
Número da solicitação: 867_Req3705_01	
Data de entrada: 12/04/2017 Data de saída: 20/04/2017	
Volume:	120
Custo:	R\$ 50.000,00
Prazo:	12
Desempenho:	Não houve comprometimento de desempenho
Incidentes ocorridos:	
Descrição:	Não houve incidentes
Solução:	NA
Ações:	NA
Detalhamento da avaliação:	
Anexos da avaliação: <input checked="" type="radio"/> Sim <input type="radio"/> Não	
<input type="radio"/> Encaminhado para CCB Data: Clique aqui para inserir uma data.	
<input type="radio"/> Encaminhado para gerenciamento de base de conhecimento Data: Clique aqui para inserir uma data.	
<input type="radio"/> Disponível para auditoria Data: Clique aqui para inserir uma data.	

Fonte: elaborada pelo autor.

5.7. O aprimoramento do modelo existente: criação e gerenciamento de *baseline* e de base de conhecimento.

Diante do exposto, observou-se que os subprocessos que compõem os macroprocessos constituintes do modelo ora proposto já são contemplados na prática adotada pelo INPE, embora com nível de formalização e controle diferenciado.

Desta forma, o aprimoramento proposto por esta pesquisa sobre o que ocorre na prática, em geral, concentra-se em dois pontos que foram considerados importantes:

- a) a criação e gerenciamento de uma *baseline* de requisitos implementada na ferramenta (software) de suporte para que funcione de forma *online* e possibilite o gerenciamento direto dos requisitos;
- b) e a criação e gerenciamento de uma base de conhecimentos onde serão registrados todos os problemas vivenciados e as soluções adotadas.

Isto possibilitará, ao longo do tempo, registrar parte do conhecimento tácito utilizado ao longo das reuniões para a análise das solicitações de mudanças.

O que ocorre hoje são análises realizadas em reuniões em que os conhecimentos e experiências pessoais dos integrantes são tomados como base para as decisões. No entanto, embora o resultado seja registrado em atas, o conhecimento e os problemas discutidos não são registrados, o que dificulta a possibilidade de reuso do conhecimento para solução de questões similares em situações futuras, bem como do nivelamento de conhecimento em reuniões futuras com integrantes diferentes.

O formulário da manutenção da base de conhecimentos deve ser mantido na área que administra a base de conhecimentos, para lastrear o histórico das alterações e permitir a identificação de melhores práticas, com os registros das alterações realizadas em função das alterações nos requisitos e os dados dos

requisitos alterados, os requisitos originais, as datas de alteração e os incidentes ocorridos com a descrição das devidas soluções.

Neste caso demonstrativo não houve incidentes e, portanto, não houve registro na manutenção da base de conhecimentos. A Figura 5.13, a seguir, mostra o formulário para manutenção da base de conhecimentos preenchido com os dados do estudo de caso.

Figura 5.13 - Formulário de manutenção da base de conhecimentos.

	Processo de Mudança de Requisito	Página 8 de 8
Manutenção da base de conhecimento		
Número da solicitação: <input type="text" value="867_Req3705_01"/> Data: 12/04/2017		
Enviado para manutenção de base de conhecimento para : <input type="radio"/> Inclusão <input type="radio"/> Exclusão <input type="radio"/> Modificação Data de envio: 28/04/2017		
* Aguardando informação Data: <input type="text"/> Clique aqui para inserir uma data.		
Descrição dos problemas encontrados: <input type="text"/>		
Descrição da solução dos problemas encontrados: <input type="text"/>		
Observações relevantes: <input type="text"/>		

Fonte: elaborada pelo autor.

5.8. Considerações gerais sobre a demonstração do modelo

Observou-se ser possível a aplicação do modelo proposto no ambiente do INPE, com alguns ajustes, considerando existirem ferramentas adequadas de software para dar suporte às atividades mínimas imprescindíveis, tais como: rastreabilidade, análise de impacto e administração do fluxo de documentação, para o estabelecimento do gerenciamento de mudanças de requisitos.

Considerou-se satisfatório o uso do PTC *Integrity* ao longo da execução do processo de rastreabilidade suportando a análise de impacto, pelo fornecimento das informações necessárias para tal tarefa.

Para o processo de geração de conteúdo para a base de conhecimento, as atas do CCB podem ser usadas como uma das fontes de alimentação; no entanto, as informações adicionais, referentes aos processos de análises realizados em cada área envolvida, devem ser coletadas, analisadas, categorizadas e formatadas.

A demonstração de aplicação dos processos do modelo não irá se diferenciar muito caso a alteração a ser feita na *baseline* seja de inclusão, de exclusão ou da alteração de requisitos, que foi a operação demonstrada.

No caso de exclusão, a relação de requisitos, documentos e testes afetados pela exclusão será automaticamente detectada pelo processo de rastreabilidade e apresentada ao analista que irá prosseguir com as análises e seguir o processo exatamente como no caso de uma alteração de requisito.

No caso de uma inclusão de requisito, inicialmente dever-se-á fazer todo o desenvolvimento de requisitos decorrentes, conforme é normalmente feito na fase anterior de preparação de requisitos (fase que não foi objeto deste trabalho) e, posteriormente, inclui-los na *baseline* com suas informações de relacionamentos, usando a ferramenta de software como suporte e prosseguindo para as análises posteriores e processos descritos no modelo proposto.

6 CONCLUSÕES

Os objetivos desta pesquisa foram atingidos conforme descritos a seguir:

- a) Objetivo 1: propor um processo de Gerenciamento de Requisitos, tomando por base modelos existentes e aplicáveis ao contexto aeroespacial: foi definido um processo para gerenciamento de requisitos que tomou por base a análise e reflexão sobre os achados em diversos estudos disponibilizados na literatura sobre Gerenciamento de Requisitos, e a análise da comparação de quatro abordagens adotadas pelas principais instituições que tratam de Engenharia de Sistemas, contexto aeroespacial e software, a saber, a NASA, ECSS, INCOSE e IEEE;
- b) Objetivo 2: demonstrar a aplicação do modelo proposto no contexto espacial do INPE, utilizando ferramentas disponíveis no ambiente: o modelo foi demonstrado através da simulação de solicitação de modificação de um requisito existente em uma base parcial de requisitos utilizada para estudos de aprendizagem do software PTC *Integrity* montada a partir de dados reais pela equipe de verificação e validação do INPE;

os resultados obtidos com a demonstração de aplicação do modelo, demonstraram que o modelo proposto permite a documentação de todo o processo de mudança de requisitos e revelaram que as ferramentas de suporte atualmente em uso no INPE prestam-se perfeitamente para fornecer as principais informações necessárias para o pleno controle do processo de mudanças.

O software PTC *integrity* realiza registro, análise de impacto, rastreamento e testes de verificação e validação de requisitos e, sem dúvidas, garante o preenchimento dos documentos do processo de mudanças.

No contexto do INPE é importante ressaltar o estudo de Albuquerque (2012) que tratou do aprimoramento do processo de gerenciamento de configurações adotado pelo INPE, uma vez que, por sua similaridade com os processos de gerenciamento de mudanças de requisitos, foram considerados nesta pesquisa. É importante salientar, no entanto, que Albuquerque (2012) não tratou dos aspectos relacionados com o uso de ferramentas de software e com a especificação de roteiros para a aplicação do processo.

Nesta pesquisa procurou-se abordar na proposta, os processos que, normalmente, já são praticados no INPE, de forma a estabelecer facilidades e reuso de conhecimento lá existente, caso a Instituição opte por adotar o processo sugerido por esta pesquisa.

Observou-se que a NASA aborda a atividade de gerenciamento de requisitos de forma separada em processos específicos, não agregando atividades, comumente atribuídas ao desenvolvimento de requisitos.

O guia do INCOSE disponibiliza material sobre desenvolvimento de requisitos abordado no capítulo que trata processos técnicos e engloba, no capítulo de que trata de processos de projetos, as atividades de gerenciamento de *baselines*, nas quais inclui tanto os requisitos do sistema quanto os de ambiente.

A ECSS apresenta uma norma que trata, conjuntamente, o Gerenciamento de Configuração e de Informação, uma específica para verificação e uma para definição de requisitos técnicos.

Da mesma forma, a documentação do IEEE, que é voltada especificamente para software, apresenta um capítulo para tratamento geral de requisitos, onde descreve tarefas de desenvolvimento de requisitos e, também, de gerenciamento de mudanças, embora o foco principal sejam os processos iniciais para o desenvolvimento dos requisitos.

Observando as abordagens publicadas buscou-se criar um processo específico que permitisse reduzir as lacunas observadas nas abordagens estudadas,

propondo a unificação das atividades descritas nas quatro abordagens no que diz respeito ao Gerenciamento de Requisitos.

Assim, conclui-se que as quatro abordagens são parecidas, ressaltando que a documentação da NASA apresenta um guia em formato que permite maior usabilidade.

Cabe ressaltar que todos os documentos analisados consistem em guias que recomendam o que se deve executar para o efetivo Gerenciamento de Requisitos em todos os seus aspectos, isto é, são citados os processos recomendados e, em alguns pontos, algumas atividades inerentes a estes processos; no entanto, estes guias não se aprofundam no detalhamento destas atividades ao ponto de fornecer instruções das atividades e a sequência destas, de forma a servir como um guia de implantação para o interessado, caso este seja o seu intuito.

O processo proposto nesta dissertação apresenta uma contribuição para complementar as abordagens estudadas, sugerindo um roteiro com instruções de como implementar as recomendação obtidas a partir dos documentos analisados.

Atualmente o processo usado no INPE, em termos de controle de versões, é baseado no controle das atas que são emitidas nas reuniões do CCB, agregadas aos documentos originais e suas atualizações, através do software PTC *windchill* que funciona como um repositório de documentos, realizando o controle de um modo geral sobre documentos e não especificamente um controle focado em requisitos.

Certamente que a adoção de um método de controle de mudanças permitirá maior segurança da informação, podendo evitar erros e/ou perdas em caso de invasão e roubo de informações, uma vez que as informações registradas não poderão ser mudadas sem a anuência das partes que realizaram as análises exigidas.

Por outro lado, há necessidade de mudanças de rotinas, o que demanda maior esforço e cooperação dos envolvidos, já que tarefas adicionais de registro deverão ser realizadas.

Em termos de conhecimento, foram apresentados roteiros compostos por questões técnicas para auxiliar os processos e análise de impacto, bem como foi apresentado um critério para tomada de decisão baseado em médias ponderadas a partir do atendimento das questões aplicadas.

6.1. Sugestões para trabalhos futuros

O aprimoramento de um processo ocorre quando há implantação deste, momento em que são identificadas as informações faltantes e são providenciadas as fontes adicionais e/ou informações substitutas para suprir as lacunas.

Como trabalhos futuros sugerem-se, dentre outras iniciativas:

- 1) em termos de aplicação do processo:
 - a) aplicação do processo proposto a casos mais realistas.
 - b) automação do processo proposto, devendo ser implementados o *workflow* e o GED (Gerenciamento Eletrônico de Documentos), para rastrear todos os passos do processo e identificar gargalos, de maneira a pontuar ações corretivas e preventivas.
 - c) após a automação, sugere-se a simulação do processo sobre processos de desenvolvimento de software e de produtos, de forma a realizar os ajustes necessários e agregar, ou suprimir, funcionalidades conforme a exigência do caso.
 - d) após as simulações e ajustes sugere-se a implantação do processo aprimorado, considerando um projeto piloto. Para tal, deverão ser planejados treinamentos e definições de atores e seus respectivos papéis.

- e) os aspectos de críticas de preenchimento dos formulários e testes de segurança da informação deverão ser contemplados ao longo da automação e implementação do processo.
 - f) a implementação da base de conhecimentos, repositório de descrições de incidentes e de soluções sugeridas por esta proposta, adotando-se uma ferramenta de gerenciamento de banco de dados como, por exemplo, o MySQL combinado com o PHP e a integração com as bases dos softwares PTC *Integrity* e PTC *Windchill*, além da implementação dos formulários para funcionar em modo *web*.
- 2) em termos de trabalhos científicos, sugerem-se:
- a) levantamento de custos relacionados a cada módulo do processo proposto para subsidiar a tomada de decisão sobre a adoção plena ou parcial do modelo;
 - b) embora tenham sido apresentadas instruções para a execução das análises, por se tratarem de atividades imprescindíveis e de alto grau de complexidade sugere-se o aprimoramento dos conjuntos de questões apresentadas de forma a definir um padrão para as análises direcionadas para cada domínio de conhecimento necessário para a realização das atividades sugeridas;
 - c) estudos para a seleção de ferramentas adequadas para gerenciamento eletrônico de documentos para viabilizar o acompanhamento do fluxo de documentação de forma automática.

REFERÊNCIAS BIBLIOGRÁFICAS

- ASSOCIAÇÃO BRASILEIRA DE NORMAS TÉCNICAS (ABNT). **NBR ISO/IEC 20000-1: Tecnologia da Informação-Gestão de Serviços Parte 1: Requisitos do sistema de gestão de serviços**. Rio de Janeiro, 2011. p. 37.
- ACKERMAN, A. F. et al. Software inspections: an effective verification process. **IEEE Software**, v. 14, n. 1, 1989.
- ALEXANDER, I. F.; STEVENS, R. **Writing better requirements**. Edimburg: Pearson Education, 2002.
- ALMEIDA, M. C. P. **Proposta de adoção de um processo de captura e rastreamento de requisitos baseada num estudo de caso e num histórico das fases da engenharia de sistemas no INPE**. 2011. 222 p. IBI: <8JMKD3MGP7W/3AQPAJ8>. (sid.inpe.br/mtc-m19/2011/11.17.19.31-TDI). Dissertação (Mestrado em Engenharia e Gerenciamento de Sistemas Espaciais) - Instituto Nacional de Pesquisas Espaciais, São José dos Campos, 2011. Disponível em: <<http://urlib.net/8JMKD3MGP7W/3AQPAJ8>>. Acesso em: 28 mar 2015.
- BASIL, V. R. et al. The empirical investigation of perspective-based reading. **Empirical Software Engineering**, v. 1, n. 2, 1996.
- BIZAGI. **Bizagi - time to digital**. 2016. Disponível em: <<http://www.bizagi.com/en>>. Acesso em: 20 jan. 2017
- BUCKLEY, F. J. **Standard, guideline, and examples on system and software requirements engineering**. California: IEEE, 1990. Ieee Computer Society Press Tutorial Series.
- CARVER, J. **Impact of background and experience on software inspections**. 2002. Ph.D. Thesis - University of Maryland, EUA, College Park, 2002.
- CHRISTEL, G.; KANG, K. C. **Issues in requirements elicitation**. Pittsburgh, Pennsylvania: Software Engineering Institute Carnegie Mellon University, 1992. p. 80.
- DAVIS, A. M. The analysis and specification of system and software. In: THEYER, R. H.; DORFMAN, M. (eds.). **Systems and software requirements engineering**. 14. ed. Los Alamitos, California: IEEE Computer Society Press, 1990. v. 46, p. 119-144.
- EUROPEAN SPACE AGENCY (ESA). **Description, implementation and general requirements**. Noordwijk, The Netherlands: ESA Requirements and Standards Division, ESTEC, 2008. p. 34. (ECSS-S-ST-00C).

EUROPEAN SPACE AGENCY (ESA). **Configuration and information management**. Noordwijk, The Netherlands: ESA Requirements and Standards Division, ESTEC, 2009. p. 34. (ECSS-M-ST-40C Rev. 1).

EUROPEAN SPACE AGENCY (ESA). **Space engineering** - system engineering general requirements. Noordwijk, The Netherlands: ESA Requirements and Standards Division, ESTEC, 2009. p. 100. (ECSS-E-ST-10C).

EUROPEAN SPACE AGENCY (ESA). **ECSS system** - glossary of terms. Noordwijk, The Netherlands: ESA Requirements and Standards Division, ESTEC, 2012. p. 63. (ECSS-S-ST-00-01C).

FAGAN, M. E. Advances in software inspections. **IEEE Transactions on Software Engineering**, v. 12, n. 7, 1986.

GENVIGIR, E. C. **Um modelo para rastreabilidade de requisitos de software baseado em generalização de elos e atributos**. 2009. 200 p. IBI: <8JMKD3MGP8W/34RPLDE>. (INPE-15761-TDI/1504). Tese (Doutorado em Computação Aplicada) - Instituto Nacional de Pesquisas Espaciais, São José dos Campos, 2009. Disponível em: <<http://urlib.net/8JMKD3MGP8W/34RPLDE>>.

GOTEL, O.; FINKELSTEIN, C. **An analysis of the requirements traceability problem**. In: INTERNATIONAL CONFERENCE ON REQUIREMENTS ENGINEERING, 1., 1994, Colorado Springs, CO , USA. **Proceedings...** IEEE, 1994. p. 184-193.

GRANDE, J. I. **Especificação e implementação de uma ferramenta para gerenciamento de requisitos**. 2005. Dissertação (Mestrado em Ciência da Computação) - Universidade Metodista de Piracicaba, Piracicaba, SP, 2006.

GUZZO, C.; MACCARI, E. A. Sistematização de um modelo de lições aprendidas em projetos como contribuição à aprendizagem organizacional. In: SIMPÓSIO DE GESTÃO DA INOVAÇÃO TECNOLÓGICA, 27., 2012, Salvador, Ba. **Anais...** Salvador: ANPAD, 2012.

HANSEN, S. **A socio-technical perspective on requirements engineering**. 2011. Disponível em: <https://etd.ohiolink.edu/!etd.send_file?accession=case1291749802&disposition=inline>. Acesso em: 10 abr. 2015.

HULL, E. et al. **Requirements engineering**. London: Springer-Verlag, 2011.

INSTITUTE OF ELECTRICAL AND ELECTRONICS ENGINEERS (IEEE). **IEEE Std 610.12-1990** - standard glossary of software engineering terminology. New York, 1990.

INSTITUTE OF ELECTRICAL AND ELECTRONICS ENGINEERS (IEEE). **IEEE Std 1220-2005** - systems engineering — application and management of the systems engineering process. New York, 2005.

INSTITUTE OF ELECTRICAL AND ELECTRONICS ENGINEERS (IEEE). **Systems and software engineering** — Life cycle processes — Requirements engineering. Switzerland, 2011. (ISO/IEC/IEEE 29148:2011(E)).

IEEE COMPUTER SOCIETY - ACM COMMITTEE. **Guide to the software engineering body of knowledge**. [S.l.], 2000.

IEEE COMPUTER SOCIETY. **Guide to the software engineering body of knowledge**. Los Alamitos: [s.n.], 2004.

IEEE COMPUTER SOCIETY. **SWEBOK 3.0** - guide to the software engineering body of knowledge. [S.l.]: IEEE, 2014.

INCOSE. **INCOSE systems engineering handbook v. 3.2.2**. San Diego, CA: [s.n.], 2011.

JONES, C. **Software engineering best practices**. [S.l.]: McGraw-Hill, 2010.

KAUPPINEN, M. et al. Implementing requirements engineering processes throughout organizations: success factors and challenges. **Information and Software Technology**, 1 nov. 2004. 937-953.

KOTONYA, G.; SOMMERVILLE, I. **Requirements engineering processes and techniques**. Chichester, England: John Wiley, 1998.

LARSSON, A. Dia diagram editor. **DIA**, 2017. Disponível em: <<https://wiki.gnome.org/Apps/Dia>>. Acesso em: 20 jan. 2017.

MARTIN, J. et al. An experimental study of fault detection in user requirements documents. **ACM Transactions on Software Engineering and Methodology**, v. 1, n. 2, 1992.

MEYER, B. On formalism in specifications. **IEEE Software**, p. 6-26, 1985.

NASA. **NASA/SP-2007-6105 rev1**-systems engineering handbook. Washington: National Aeronautics and Space Administration, 2007.

NASA. **NASA Configuration Management (CM) Standard**. Washington, DC, 2008. 143p. (NASA-STD-0005).

NUSEIBEH, B.; EASTERBROOK, S. Requirements engineering: a roadmap. In: CONFERENCE ON THE FUTURE OF SOFTWARE ENGINEERING, 22., 2000, Limerick, Ireland. **Proceedings...** New York: ACM, 2000. p. 35-46.

PROJECT MANAGEMENT INSTITUTE (PMI). **PMI's pulse of the profession: requirements management — a core competency for project and program success.** Newtown Square: Project Management Institute, 2014.

POHL, K.; RUPP, C. **Requirements engineering fundamentals: a study guide for the certified professional for requirements engineering exam: foundation level.** 1. ed. [S.l.]: O'Reilly Media, 2011.

ROYCE, W. W. **Managing the development of large software systems,** 1970. Disponível em: <<http://www.cs.umd.edu/class/spring2003/cmsc838p/Process/waterfall.pdf>>. Acesso em: 02 abr. 2015.

SEILEVEL. **SEILEVEL.** 2015. Disponível em: <<http://www.seilevel.com/business-analyst-resources/whitepapers/>>. Acesso em: 20 abr. 2015.

SOMMERVILLE, I. **Software engineering´.** 8. ed. Boston, MA: Addison Wesley, 2007.

SOMMERVILLE, I.; SAWYER, P. **Requirements engineering: a good practice guide.** 1. ed. Chichester: John Wiley & Sons, 1997.

VOLERE. **VOLERE.** 2015. Disponível em: <<http://volere.co.uk/tools.htm>>. Acesso em: 20 abr. 2015.

WESTFALL, L. **Bidirectional requirements traceability.** 2007. Disponível em: <<http://www.compaid.com/caiinternet/ezone/westfall-bidirectional.pdf>>. Acesso em: 28 jan. 2017.

WIEGERS, K.; BEATTY, J. **Software requirements.** 3. ed. Redmond, WA: Microsoft Press, 2013.

YOUNG, R. R. **The requirements engineering handbook.** Norwood, MA: Artech House, Inc, 2004.

APÊNDICE A – CÓDIGOS PARA OS FORMULÁRIOS DE DADOS DA MUDANÇA PROPOSTA

Página principal com hyperlinks para os formulários:

```
<html>
<head><title>Proposta de Mudança</title>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1">
<link rel="stylesheet" href="w3.css">
<link rel='stylesheet' href='https://fonts.googleapis.com/css?family=Roboto'>
<link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-
awesome/4.7.0/css/font-awesome.min.css">
</head>
<body background="backgrnd.gif" class="w3-light-grey">
<!-- Container da página -->
<div class="w3-content w3-margin-top" style="max-width:1200px;">
  <!-- Grade -->
  <div class="w3-row-padding">
    <div class="w3-display-container">
      
      <div class="w3-display-bottomleft w3-container w3-text-black">
        <h1>Proposta de mudança de requisito </h1>
      </div>
    </div>
  </div>
  <div class="w3-container w3-card-2 w3-white w3-margin-bottom">
    <h4
    class="w3-text-grey w3-padding-16">
    <i class="fa fa-certificate fa-fw w3-margin-right w3-xxlarge w3-text-teal"></i>
    <a href="form1.php">Formulário de proposta Identificação </a>
  </h4>
</div>
  <div class="w3-container w3-card-2 w3-white w3-margin-bottom">
    <h4
    class="w3-text-grey w3-padding-16">
    <i class="fa fa-certificate fa-fw w3-margin-right w3-xxlarge w3-text-teal"></i>
    <a href="form2.php">Formulário de análise preliminar </a>
  </h4>
</div>
  <div class="w3-container w3-card-2 w3-white w3-margin-bottom">
    <h4
    class="w3-text-grey w3-padding-16">
    <i class="fa fa-certificate fa-fw w3-margin-right w3-xxlarge w3-text-teal"></i>
    <a href="form3.php">Formulário de análise de impacto </a>
```

```

</h4>
</div>
<div class="w3-container w3-card-2 w3-white w3-margin-bottom">
  <h4
    class="w3-text-grey w3-padding-16">
    <i class="fa fa-certificate fa-fw w3-margin-right w3-xxlarge w3-text-teal"></i>
    <a href="form4.php"> Formulário de verificação </a>
  </h4>
</div>
<div class="w3-container w3-card-2 w3-white w3-margin-bottom">
  <h4
    class="w3-text-grey w3-padding-16">
    <i class="fa fa-certificate fa-fw w3-margin-right w3-xxlarge w3-text-teal"></i>
    <a href="form5.php"> Formulário de Validação </a>
  </h4>
</div>
</div> <!-- fim da grade -->
</div> <!-- fim do Container da página -->
</body>
</html>

```

Página do formulário de dados de identificação da proposta de mudança:

```

<html>
<head><title>Mudança_info</title>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1">
<link rel="stylesheet" href="w3.css">
<link rel='stylesheet' href='https://fonts.googleapis.com/css?family=Roboto'>
<link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-
awesome/4.7.0/css/font-awesome.min.css">
</head>
<body background="backgrnd.gif" class="w3-light-grey">
<!-- Container da página -->
<div class="w3-content w3-margin-top" style="max-width:1200px;">
  <!-- Grade -->
  <div class="w3-row-padding">
    <div class="w3-container w3-card-2 w3-white w3-margin-bottom">
      <div class="w3-display-container">
        
        <div class="w3-display-bottomleft w3-container w3-text-black">
          <h1>Proposta de mudança de requisito </h1>
        </div>
      </div>
    </div>
  </div>
</div>

```

```

<?php
// definir variaveis e configurar para valor vazio
$id = $data_em = $solicitante = $projeto = $cowebsite = "";
if ($_SERVER["REQUEST_METHOD"] == "POST") {
    $id = test_input($_POST["id"]);
    $data_em = test_input($_POST["data_em"]);
    $solicitante = test_input($_POST["solicitante"]);
    $projeto = test_input($_POST["projeto"]);
    $status = test_input($_POST["status"]);
    $titulo = test_input($_POST["titulo"]);
    $descricao = test_input($_POST["descricao"]);
    $descricao_anexos = test_input($_POST["descricao_anexos"]);
    $justificativa = test_input($_POST["justificativa"]);
    $justificativa_anexos = test_input($_POST["justificativa_anexos"]);
    $itens_afetados = test_input($_POST["itens_afetados"]);
    $itens_afetados_anexos = test_input($_POST["itens_afetados_anexos"]);
    $documentos_afetados = test_input($_POST["documentos_afetados"]);
    $documentos_afetados_anexos =
test_input($_POST["documentos_afetados_anexos"]);
}
function test_input($data) {
    $data = trim($data);
    $data = stripslashes($data);
    $data = htmlspecialchars($data);
    return $data;
}
?>
<form method="post" action="<?php echo
htmlspecialchars($_SERVER["PHP_SELF"]);?>">
<div class="w3-container w3-card-2 w3-white w3-margin-bottom">
    <h4 class="w3-text-grey w3-padding-16"><i class="fa fa-certificate fa-fw w3-margin-
right w3-xxlarge w3-text-teal"></i>Identificação</h4>
    <label>Número da solicitação: </label>
    <input style="height:30px; width:90px" type="text" name="id"/>
    <label>Data de emissão: </label>
    <input style="height:30px; width:115px" type="date" name="data_em"
placeholder="dd-mm-yyyy" pattern="\d{1,2}-\d{1,2}-\d{4}"/>
    <label>Solicitante: </label>
    <input style="height:30px; width:180px" type="text" name="solicitante"/>
    <label>Projeto: </label>
    <input style="height:30px; width:240px" type="text" name="projeto"/>
    <br>
    <h4 class="w3-text-grey w3-padding-16">Status da solicitação</h4>
    <input type="radio" name="status" value="recebido" checked>Recebido
    <input type="radio" name="status" value="verificado">Verificado

```

```



```

Página do formulário de dados de análise preliminar da proposta de mudança:

```
<html>
<head><title>Analise preliminar</title>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1">
<link rel="stylesheet" href="w3.css">
<link rel='stylesheet' href='https://fonts.googleapis.com/css?family=Roboto'>
<link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-
awesome/4.7.0/css/font-awesome.min.css">
</head>
<body background="backgrnd.gif" class="w3-light-grey">
<!-- Container da página -->
<div class="w3-content w3-margin-top" style="max-width:1200px;">
  <!-- Grade -->
  <div class="w3-row-padding">
    <div class="w3-container w3-card-2 w3-white w3-margin-bottom">
      <div class="w3-display-container">
        
        <div class="w3-display-bottomleft w3-container w3-text-black">
          <h1>Formulário de análise preliminar </h1>
        </div>
      </div>
    </div>
  </div>
</div>
<?php
// define variaveis e atribui valor vazio
$Id = $data_em = $solicitante = $projeto = $cowebsite = "";
if ($_SERVER["REQUEST_METHOD"] == "POST") {
  $analise_pre_anexos = test_input($_POST["analise_pre_anexos"]);
  $data_entrada_analise_pre = test_input($_POST["data_entrada_analise_pre"]);
  $data_saida_analise_pre = test_input($_POST["data_saida_analise_pre"]);
  $analise_pre_resolve = test_input($_POST["analise_pre_resolve"]);
  $analise_pre_tec_implement = test_input($_POST["analise_pre_tec_implement"]);
  $analise_pre_obs = test_input($_POST["analise_pre_obs"]);
  $analise_pre_anexos = test_input($_POST["analise_pre_anexos"]);
  $analise_pre_resp = test_input($_POST["analise_pre_resp"]);
}
function test_input($data) {
  $data = trim($data);
  $data = stripslashes($data);
  $data = htmlspecialchars($data);
  return $data;
}
?>
```

```

<form method="post" action="<?php echo
htmlspecialchars($_SERVER["PHP_SELF"]);?>">
  <div class="w3-container w3-card-2 w3-white w3-margin-bottom">
    <h5 class="w3-text-grey w3-padding-16"><i class="fa fa-certificate fa-fw w3-margin-
right w3-xxlarge w3-text-teal"></i>Análise preliminar </h5>
    <input type="radio" name="analise_pre_anexos" value="aprovado">Aprovado
    <input type="radio" name="analise_pre_anexos" value="rejeitado">Rejeitado
    <label style="color:blue;text-align:right"> ..... Data de entrada: </label>
    <input style="height:30px; width:120px" type="date"
name="data_entrada_analise_pre" placeholder="dd-mm-yyyy" pattern="\d{1,2}-
\d{1,2}-\d{4}"/>
    <label style="color:blue;text-align:center">Data de saída: </label>
    <input style="height:30px; width:120px" type="date"
name="data_saida_analise_pre" placeholder="dd-mm-yyyy" pattern="\d{1,2}-\d{1,2}-
\d{4}"/> <br> <br>
    <label> - A mudança proposta resolve o problema? </label>
    <input type="radio" name="analise_pre_resolve" value="sim">Sim
    <input type="radio" name="an_pre_solve" value="nao">Não <br><br>
    <label> - A mudança é tecnicamente passível de implementação? </label>
    <input type="radio" name="analise_pre_tec_implement" value="sim">Sim
    <input type="radio" name="analise_pre_tec_implement" value="nao">Não
<br><br>
    <label> Observações: </label>
    <textarea name="analise_pre_obs" rows="2" cols="120"></textarea> <br>
    <label> Anexos: </label>
    <input type="radio" name="analise_pre_anexos" value="sim">Sim
    <input type="radio" name="analise_pre_anexos" value="nao">Não <br><br>
    <label>Responsável pela análise preliminar: </label>
    <input style="height:30px; width:660px" type="text" name="analise_pre_resp"/>
<br> <br>
  </div>
  <input type="submit" name="submit" value="Submit">
</form>
<a href="inpe_main.php" onclick="w3_close()" class="w3-padding w3-text-teal">
<i class="w3-xxlarge fa fa-home fa-fw w3-margin-right">
</i>Retorna à pagina de escolha de formulários</a>
<hr>
</div> <!-- fim da grade -->
</div> <!-- fim do Container da página -->
</body>
</html>

```

Página do formulário de dados de análise de impacto da proposta de mudança:

<html>

```

<head><title>Análise de impacto</title>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1">
<link rel="stylesheet" href="w3.css">
<link rel='stylesheet' href='https://fonts.googleapis.com/css?family=Roboto'>
<link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-
awesome/4.7.0/css/font-awesome.min.css">
</head>
<body background="backgrnd.gif" class="w3-light-grey">
<!-- Container da página -->
<div class="w3-content w3-margin-top" style="max-width:1200px;">
  <!-- Grade -->
  <div class="w3-row-padding">
    <div class="w3-container w3-card-2 w3-white w3-margin-bottom">
      <div class="w3-display-container">
        
        <div class="w3-display-bottomleft w3-container w3-text-black">
          <h1>Formulário de análise de impacto </h1>
        </div>
      </div>
    </div>
  </div>
  <?php
  // define variables and set to empty values
  $id = $data_em = $solicitante = $projeto = $cowebsite = "";

  if ($_SERVER["REQUEST_METHOD"] == "POST") {
    $analise_impacto_anexos = test_input($_POST["analise_impacto_anexos"]);
    $data_entrada_analise_impacto =
test_input($_POST["data_entrada_analise_impacto"]);
    $data_saida_analise_impacto = test_input($_POST["data_saida_analise_impacto"]);
    $analise_impacto_obs = test_input($_POST["analise_impacto_obs"]);
    $analise_impacto_anexos = test_input($_POST["analise_impacto_anexos"]);
    $analise_impacto_resp = test_input($_POST["analise_impacto_resp"]);
    $criticidade = test_input($_POST["criticidade"]);
    $prioridade = test_input($_POST["prioridade"]);
  }
  function test_input($data) {
    $data = trim($data);
    $data = stripslashes($data);
    $data = htmlspecialchars($data);
    return $data;
  }
  ?>
  <form method="post" action="<?php echo
htmlspecialchars($_SERVER["PHP_SELF"]);?>">

```

```

<div class="w3-container w3-card-2 w3-white w3-margin-bottom">
  <h5 class="w3-text-grey w3-padding-16"><i class="fa fa-certificate fa-fw w3-margin-right w3-xxlarge w3-text-teal"></i>Análise de impacto </h5>
  <label> <b> Análise de impacto </b> </label>
  <input type="radio" name="analise_impacto_anexos" value="aprovado">Aprovado
  <input type="radio" name="analise_impacto_anexos" value="rejeitado">Rejeitado
  <label style="color:blue;text-align:right"> ..... Data de entrada: </label>
  <input style="height:30px; width:120px" type="date"
name="data_entrada_analise_impacto" placeholder="dd-mm-yyyy" pattern="\d{1,2}-
\d{1,2}-\d{4}"/>
  <label style="color:blue;text-align:center">Data de saída: </label>
  <input style="height:30px; width:120px" type="date"
name="data_saida_analise_impacto" placeholder="dd-mm-yyyy" pattern="\d{1,2}-
\d{1,2}-\d{4}"/> <br> <br>
  <label> Observações: </label>
  <textarea name="analise_impacto_obs" rows="2" cols="120"></textarea> <br>
  <label> Anexos para análise de impacto: </label>
  <input type="radio" name="analise_impacto_anexos" value="sim">Sim
  <input type="radio" name="analise_impacto_anexos" value="nao">Não <br><br>
  <label>Responsável pela análise de impacto: </label>
  <input style="height:30px; width:660px" type="text"
name="analise_impacto_resp"/> <br> <br>
  <label> <b> Classificação: </b> </label> <br>
  <label style="color:blue;text-align:right"> Criticidade: </label>
  <input type="radio" name="criticidade" value="alta">Alta
  <input type="radio" name="criticidade" value="media">Média
  <input type="radio" name="criticidade" value="baixa">Baixa
  <label style="color:blue;text-align:right"> Prioridade: </label>
  <input type="radio" name="prioridade" value="alta">Alta
  <input type="radio" name="prioridade" value="media">Média
  <input type="radio" name="prioridade" value="baixa">Baixa <br><br>
</div>
<input type="submit" name="submit" value="Submit">
</form>
<a href="inpe_main.php" onclick="w3_close()" class="w3-padding w3-text-teal">
<i class="w3-xxlarge fa fa-home fa-fw w3-margin-right">
</i>Retorna à pagina de escolha de formulários</a>
<hr>
</div> <!-- fim da grade -->
</div> <!-- fim do Container da página -->
</body>
</html>

```

Página do formulário de dados de verificação da proposta de mudança:

```

<html>
<head><title>verificacao</title>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1">
<link rel="stylesheet" href="w3.css">
<link rel='stylesheet' href='https://fonts.googleapis.com/css?family=Roboto'>
<link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-
awesome/4.7.0/css/font-awesome.min.css">
</head>
<!-- <body background="backgrnd.gif"> -->
<body class="w3-light-grey">
<!-- Page Container -->
<div class="w3-content w3-margin-top" style="max-width:1200px;">
  <!-- The Grid -->
  <div class="w3-row-padding">
    <div class="w3-container w3-card-2 w3-white w3-margin-bottom">
      <div class="w3-display-container">
        
        <div class="w3-display-bottomleft w3-container w3-text-black">
          <h1>Formulário de verificação </h1>
        </div>
      </div>
    </div>
  </div>
  <?php
  // define variables and set to empty values
  $id = $data_em = $solicitante = $projeto = $cowebsite = "";
  if ($_SERVER["REQUEST_METHOD"] == "POST") {
    $name = test_input($_POST["id"]);
    $email = test_input($_POST["data_em"]);
    $website = test_input($_POST["solicitante"]);
    $comment = test_input($_POST["projeto"]);
    $status = test_input($_POST["status"]);
  }
  function test_input($data) {
    $data = trim($data);
    $data = stripslashes($data);
    $data = htmlspecialchars($data);
    return $data;
  }
  ?>
  <form method="post" action="<?php echo
  htmlspecialchars($_SERVER["PHP_SELF"]);?>">
    <div class="w3-container w3-card-2 w3-white w3-margin-bottom">
      <h5 class="w3-text-grey w3-padding-16"><i class="fa fa-certificate fa-fw w3-margin-
      right w3-xxlarge w3-text-teal"></i>Verificação </h5>

```

```



```

Página do formulário de dados de validação da proposta de mudança:

```

<html>
<head><title>Validação</title>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1">
<link rel="stylesheet" href="w3.css">
<link rel='stylesheet' href='https://fonts.googleapis.com/css?family=Roboto'>
<link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-
awesome/4.7.0/css/font-awesome.min.css">
</head>
<body background="backgrnd.gif" class="w3-light-grey">
<!-- Container da página -->
<div class="w3-content w3-margin-top" style="max-width:1200px;">
<!-- Grade -->
<div class="w3-row-padding">

```

```

<div class="w3-container w3-card-2 w3-white w3-margin-bottom">
  <div class="w3-display-container">
    
    <div class="w3-display-bottomleft w3-container w3-text-black">
      <h1>Formulário de validação</h1>
    </div>
  </div>
</div>
</div>
<?php
// define variables and set to empty values
$id = $data_em = $solicitante = $projeto = $cowebsite = "";
if ($_SERVER["REQUEST_METHOD"] == "POST") {
  $validacao_status = test_input($_POST["validacao_status"]);
  $data_entrada_validacao = test_input($_POST["data_entrada_validacao"]);
  $data_saida_validacao = test_input($_POST["data_saida_validacao"]);
  $validacao_obs = test_input($_POST["validacao_obs"]);
  $validacao_anexos = test_input($_POST["validacao_anexos"]);
  $validacao_resp = test_input($_POST["validacao_resp"]);
}
function test_input($data) {
  $data = trim($data);
  $data = stripslashes($data);
  $data = htmlspecialchars($data);
  return $data;
}
?>
<form method="post" action="<?php echo
htmlspecialchars($_SERVER["PHP_SELF"]);?>">
  <div class="w3-container w3-card-2 w3-white w3-margin-bottom">
    <h5 class="w3-text-grey w3-padding-16"><i class="fa fa-certificate fa-fw w3-margin-right w3-xxlarge w3-text-teal"></i>Validação </h5>
    <input type="radio" name="validacao_status" value="aprovado">Aprovado
    <input type="radio" name="validacao_status" value="rejeitado">Rejeitado
    <label style="color:blue;text-align:right"> ..... Data de entrada: </label>
    <input style="height:30px; width:120px" type="date"
name="data_entrada_validacao" placeholder="dd-mm-yyyy" pattern="\d{1,2}-\d{1,2}-\d{4}"/>
    <label style="color:blue;text-align:center">Data de saída: </label>
    <input style="height:30px; width:120px" type="date" name="data_saida_validacao"
placeholder="dd-mm-yyyy" pattern="\d{1,2}-\d{1,2}-\d{4}"/> <br> <br>
    <label> Observações: </label>
    <textarea name="validacao" rows="2" cols="120"></textarea> <br>
    <label> Anexos para validacao: </label>
    <input type="radio" name="validacao_anexos" value="sim">Sim
    <input type="radio" name="validacao_anexos" value="nao">Não <br><br>

```

```

    <label>Responsável pela validacao: </label>
    <input style="height:30px; width:660px" type="text" name="validacao_resp"/> <br>
<br>
</div>
<input type="submit" name="submit" value="Submit">
</form>
<a href="inpe_main.php" onclick="w3_close()" class="w3-padding w3-text-teal">
<i class="w3-xxlarge fa fa-home fa-fw w3-margin-right">
</i>Retorna à pagina de escolha de formulários</a>
<hr>
</div> <!-- fim da grade -->
</div> <!-- fim do Container da página -->
</body>
</html>

```

Página do formulário de dados de aprovação do CCB:

```

<html>
<head><title>Aprovação CCB</title>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1">
<link rel="stylesheet" href="w3.css">
<link rel='stylesheet' href='https://fonts.googleapis.com/css?family=Roboto'>
<link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-
awesome/4.7.0/css/font-awesome.min.css">
<style>
table, th, td {
    padding: 5px;
    border: 15px ;
    text-align: left;
}
</style>
</head>
<body background="backgrnd.gif" class="w3-light-grey">
<!-- Container da página -->
<div class="w3-content w3-margin-top" style="max-width:1200px;">
<!-- Grade -->
<div class="w3-row-padding">
    <div class="w3-display-container">
        
        <div class="w3-display-bottomleft w3-container w3-text-black">
            <h1>Aprovação CCB</h1>
        </div>
    </div>

```

```

        </div>
    </div>
<?php
// define variables and set to empty values
$id = $data_em = $solicitante = $projeto = $cowebsite = "";
if ($_SERVER["REQUEST_METHOD"] == "POST") {
    $aprovacao_CCB = test_input($_POST["aprovacao_CCB"]);
    $id = test_input($_POST["id"]);
    $CCB_resp = test_input($_POST["CCB_resp"]);
    $ocorrencia = test_input($_POST["ocorrencia"]);
    $Alt_contratual = test_input($_POST["Alt_contratual"]);
    $data_ac = test_input($_POST["data_ac"]);
    $Implementacao = test_input($_POST["Implementacao"]);
    $data_imp = test_input($_POST["data_imp"]);
    $op_baseline = test_input($_POST["op_baseline"]);
    $manut_baseline = test_input($_POST["manut_baseline"]);
    $data_baseline = test_input($_POST["data_baseline"]);
    $knowledge_base = test_input($_POST["knowledge_base"]);
    $data_kb = test_input($_POST["data_kb"]);
}
function test_input($data) {
    $data = trim($data);
    $data = stripslashes($data);
    $data = htmlspecialchars($data);
    return $data;
}
?>
<div class="w3-container w3-card-2 w3-white">
    <h4
        class="w3-text-black w3-padding-16">
        <i class="fa fa-certificate fa-fw w3-margin-right w3-xxlarge w3-text-teal"></i>
        <a> Aprovação do CCB </a>
        <form method="post" action="<?php echo
htmlspecialchars($_SERVER["PHP_SELF"]);?>">
        <div class="w3-container w3-card-2 w3-white w3-margin-bottom">
            <table width="1300" border="0">
                <tr> <td> </td> </tr>
                <tr>
                    <td> <input type="radio" name="aprovacao_CCB" value="aprovado">Aprovado
                        <input type="radio" name="aprovacao_CCB" value="rejeitado">Rejeitado </td>
                </tr>
                <tr>
                    <td width="300"> Número da solicitação: </td>
                    <td width="1000"> <input style="height:30px; width:200px" type="text"
name="id"/> </td>

```

```

</tr>
<tr>
  <td> <label>Responsável CCB: </label> </td>
  <td> <input style="height:30px; width:660px" type="text" name="CCB_resp"/>
</td>
</tr>
</table>
</div>
<div class="w3-container w3-card-2 w3-white w3-margin-bottom">
  <table width="1200" border="0">
    <tr> <td> </td> </tr>
    <tr>
      <td width="300"> <label> <b> Ocorrencia de:</b> </label> </td>
      <td width="300"> <input type="radio" name="ocorrencia"
value="Waiver">Waiver </td>
      <td width="300"> <input type="radio" name="ocorrencia" value="Desvio">Desvio
</td>
      <td width="300"> <input type="radio" name="ocorrencia" value="sem
ocorrencia">Sem ocorrencia </td>
    </tr>
  </table>
</div>
<div class="w3-container w3-card-2 w3-white">
  <table width="1200" border="0">
    <tr> <td> </td> </tr>
    <tr>
      <label> <b>Enviar para alteração contratual:</b> </label>
    </tr>
    <tr>
      <td width="300"> <input type="radio" name="Alt_contratual"
value="Enviado">Enviado </td>
      <td width="300"> <label>Data de envio: </label>
      <input style="height:30px; width:115px" type="date" name="data_ac"
placeholder="dd-mm-yyyy" pattern="\d{1,2}-\d{1,2}-\d{4}"/> </td>
      <td width="300"></td>
      <td width="300"></td>
    </tr>
    <tr>
      <td width="300"> <input type="radio" name="Alt_contratual"
value="Aguardando">Aguardando informação </td>
      <td width="300"> <label>Data: </label>
      <input style="height:30px; width:115px" type="date" name="data_ac"
placeholder="dd-mm-yyyy" pattern="\d{1,2}-\d{1,2}-\d{4}"/> </td>
      <td width="300"></td>
      <td width="300"></td>
    </tr>
  </table>
</div>

```

```

    </tr>
  </table>
</div>
<div class="w3-container w3-card-2 w3-white">
  <table width="1200" border="0">
    <tr> <td> </td> </tr>
    <tr>
      <label> <b> Enviar para implementação: </b> </label>
    </tr>
    <tr>
      <td width="300"> <input type="radio" name="Implementacao"
value="Enviado">Enviado </td>
      <td width="300"> <label>Data de envio: </label>
        <input style="height:30px; width:115px" type="date" name="data_imp"
placeholder="dd-mm-yyyy" pattern="\d{1,2}-\d{1,2}-\d{4}"/> </td>
      <td width="300"></td>
      <td width="300"></td>
    </tr>
    <tr>
      <td width="300"> <input type="radio" name="Implementacao"
value="aguardando">Aguardando informação </td>
      <td width="300"> <label>Data: </label>
        <input style="height:30px; width:115px" type="date" name="data_imp"
placeholder="dd-mm-yyyy" pattern="\d{1,2}-\d{1,2}-\d{4}"/> </td>
      <td width="300"></td>
      <td width="300"></td>
    </tr>
  </table>
</div>
<div class="w3-container w3-card-2 w3-white">
  <table width="1200" border="0">
    <tr> <td> </td> </tr>
    <tr>
      <label> <b> Enviar para para manutenção de baseline de requisitos para: </b>
</label>
    </tr>
    <tr>
      <td width="300"> <label>Operação a ser executada: </label> </td>
      <td width="300"> <input type="radio" name="op_baseline"
value="inclusao">Inclusão </td>
      <td width="300"> <input type="radio" name="op_baseline"
value="exclusao">Exclusão </td>
      <td width="300"> <input type="radio" name="op_baseline"
value="modificacao">Modificação </td>
    </tr>
  </table>

```

```

<tr>
  <td width="300"> <input type="radio" name="manut_baseline"
value="Enviado">Enviado </td>
  <td> <label>Data de envio: </label>
    <input style="height:30px; width:115px" type="date" name="data_baseline"
placeholder="dd-mm-yyyy" pattern="\d{1,2}-\d{1,2}-\d{4}"/> </td>
  <td width="300"></td>
  <td width="300"></td>
</tr>
<tr>
  <td width="300"> <input type="radio" name="manut_baseline"
value="Enviado">Aguardando informação </td>
  <td width="300"> <label>Data: </label>
    <input style="height:30px; width:115px" type="date" name="data_baseline"
placeholder="dd-mm-yyyy" pattern="\d{1,2}-\d{1,2}-\d{4}"/> </td>
  <td width="300"></td>
  <td width="300"></td>
</tr>
</table>
</div>
<div class="w3-container w3-card-2 w3-white">
  <table width="1200" border="0">
    <tr> <td> </td> </tr>
    <tr>
      <label> <b> Enviar para manutenção de base de conhecimento:: </b> </label>
    </tr>
    <tr>
      <td width="300"> <input type="radio" name="knowledge_base"
value="Enviado">Enviado </td>
      <td width="300"> <label>Data de envio: </label>
        <input style="height:30px; width:115px" type="date" name="data_kb"
placeholder="dd-mm-yyyy" pattern="\d{1,2}-\d{1,2}-\d{4}"/> </td>
      <td width="300"></td>
      <td width="300"></td>
    </tr>
    <tr>
      <td width="300"> <input type="radio" name="knowledge_base"
value="Enviado">Aguardando informação </td>
      <td width="300"> <label>Data: </label>
        <input style="height:30px; width:115px" type="date" name="data_kb"
placeholder="dd-mm-yyyy" pattern="\d{1,2}-\d{1,2}-\d{4}"/> </td>
      <td width="300"></td>
      <td width="300"></td>
    </tr>
  </table>

```

```
</div>
<input type="submit" name="submit" value="Submit">
</form>
<a href="inpe_main.php" onclick="w3_close()" class="w3-padding w3-text-teal">
  <i class="w3-xxlarge fa fa-home fa-fw w3-margin-right"></i>Retorna à pagina de
  escolha de formulários</a>
</div>
<hr>
</div> <!-- fim da grade -->
</div> <!-- fim do Container da página -->
</body>
</html>
```


APÊNDICE B – CÓDIGOS PARA PÁGINA DE BASE DE CONHECIMENTOS

B1) código das páginas:

```
<html>
<head><title>Base conhecimento</title>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1">
<link rel="stylesheet" href="w3.css">
<link rel='stylesheet' href='https://fonts.googleapis.com/css?family=Roboto'>
<link rel="stylesheet" href="https://cdn.jsdelivr.net/npm/font-awesome/4.7.0/css/font-awesome.min.css">
<style>
  table {
    font-family: arial, sans-serif;
    border-collapse: collapse;
    width: 80%;
  }
  td, th {
    border: 1px solid #dddddd;
    text-align: left;
    padding: 8px;
  }
  tr:nth-child(even) {
    background-color: #dddddd;
  }
</style>
</head>
<body background="backgrnd.gif" class="w3-light-grey">
<!-- Container da página -->
<div class="w3-content w3-margin-top" style="max-width:1400px;">
<!-- Grade -->
<div class="w3-row-padding">
  <div class="w3-display-container">
    
    <div class="w3-display-bottomleft w3-container w3-text-black">
      </div>
    </div>
  <!-- Coluna esquerda -->
  <div class="w3-third">
    <div class="w3-white w3-text-grey w3-card-4">
      <div class="w3-display-container">

        <div class="w3-display-bottomleft w3-container w3-text-black">
          <h2>Base de Conhecimento</h2>
        </div>
      </div>
    </div>
  </div>
</div>
</body>
</html>
```

```

</div>

<div class="w3-panel w3-padding-16 w3-gray w3-opacity w3-card-2 w3-hover-
opacity-off">
  <h2>Busca </h2>
  <p></p>
  <input class="w3-input w3-border" type="text" placeholder="Entre com os
termos da busca" >
  <p></p>
  <button style="font-size:14px">Buscar <i class="fa fa-search"></i></button>
</div>
</div>
</div> <!-- fim da coluna esquerda -->
<!-- Coluna direita -->
<div class="w3-twothird">
  <?php include 'dbinfo.inc.php';?>
  <?php
  // Criando a conexão com o servidor...
  $conn = new mysqli($servername, $username, $password, $database);
  // Verificando conexão...
  if ($conn->connect_error) {
    die("Connection failed: " . $conn->connect_error);
  }
  $sql = "SELECT * FROM lessons";
  $result = $conn->query($sql);
  if ($result->num_rows > 0) {
    // imprimir cada fileira de dados
    echo "<table>" ;
    while($row = $result->fetch_assoc()) {
      echo "<tr>" . "<td>" . "<b> ID: </b>" . "</td>" . "<td>" . $row["ID_licao"]. "</td>"
. "</tr>";
      echo "<tr>" . "<td>" . "<b> Data: </b>" . "</td>" . "<td>" . $row["data"]. "</td>"
. "</tr>";
      echo "<tr>" . "<td>" . "<b> Submetido por: </b>" . "</td>" . "<td>" .
$row["submetido_por"] . "</td>" . "</tr>";
      echo "<tr>" . "<td>" . "<b> Assunto: </b>" . "</td>" . "<td>" . $row["assunto"] .
"</td>" . "</tr>";
      echo "<tr>" . "<td>" . "<b> Problema encontrado: </b>" . "</td>" . "<td>"
.$row["problema"] . "</td>" . "</tr>";
      echo "<tr>" . "<td>" . "<b> Solução: </b>" . "</td>" . "<td>" . $row["Solucao"] .
"</td>" . "</tr>";
      echo "<tr>" . "<td>" . "<b> Recorrencia: </b>" . "</td>" . "<td>"
.$row["recorrencia"] . "</td>" . "</tr>";

```

```

        echo "<tr>" . "<td>" . "<b> Lições aprendidas: </b>" . "</td>" . "<td>"
.$row["licoes"] . "</td>" . "</tr>";
        echo "<tr>" . "<td>" . "<b> Recomendações: </b>" . "</td>" . "<td>"
.$row["recomendacoes"] . "</td>" . "</tr>";
    }
    echo "</td>" . "</tr>";
    echo "</table>";
} else {
    echo "0 results";
}
$conn->close();
?>
</div> <!-- fim da coluna direita -->
</div> <!-- fim da grade -->
<!-- fim do Container da página -->
</body>
</html>

```


APÊNDICE C - COMANDOS SQL PARA CRIAR A ESTRUTURA DA BASE DE DADOS PARA AS PÁGINAS DE EXEMPLO.

```
-- phpMyAdmin SQL Dump
-- version 4.5.4.1deb2ubuntu2
-- http://www.phpmyadmin.net
-- Host: localhost
-- Generation Time: Apr 21, 2017 at 08:50 PM
-- Server version: 5.7.17-0ubuntu0.16.04.2
-- PHP Version: 7.0.15-0ubuntu0.16.04.4

SET SQL_MODE = "NO_AUTO_VALUE_ON_ZERO";
SET time_zone = "+00:00";

/*!40101 SET @OLD_CHARACTER_SET_CLIENT=@@CHARACTER_SET_CLIENT */;
/*!40101 SET @OLD_CHARACTER_SET_RESULTS=@@CHARACTER_SET_RESULTS */;
/*!40101 SET @OLD_COLLATION_CONNECTION=@@COLLATION_CONNECTION */;
/*!40101 SET NAMES utf8mb4 */;

-- Database: `baseconhecimento_inpe`
-----

-- Table structure for table `lessons`

CREATE TABLE `lessons` (
  `ID_licao` varchar(10) NOT NULL,
  `data` date NOT NULL,
  `submetido_por` varchar(256) NOT NULL,
  `assunto` varchar(256) NOT NULL,
  `problema` text CHARACTER SET latin1 NOT NULL,
  `Solucao` text NOT NULL,
  `recorrencia` varchar(256) NOT NULL,
  `licoes` text CHARACTER SET latin1 NOT NULL,
  `recomendacoes` text CHARACTER SET latin1 NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8;

-- Indexes for dumped tables

-- Indexes for table `lessons`
ALTER TABLE `lessons`
  ADD PRIMARY KEY (`ID_licao`);

/*!40101 SET CHARACTER_SET_CLIENT=@OLD_CHARACTER_SET_CLIENT */;
```

```
/*!40101 SET CHARACTER_SET_RESULTS=@OLD_CHARACTER_SET_RESULTS */;  
/*!40101 SET COLLATION_CONNECTION=@OLD_COLLATION_CONNECTION */;
```