

UNIVERSIDADE FEDERAL DO MARANHÃO
CENTRO DE CIÊNCIAS EXATAS E TECNOLOGIA
CURSO DE PÓS-GRADUAÇÃO EM ENGENHARIA DE ELETRICIDADE

FRANCISCO DE ASSIS MENÊZES BASTOS FILHO

**ELICITAÇÃO DE REQUISITOS NÃO FUNCIONAIS EM CONFORMIDADE COM
POLÍTICAS DE QUALIDADE PARA APLICAÇÕES MÉDICAS**

São Luís
2007

Livros Grátis

<http://www.livrosgratis.com.br>

Milhares de livros grátis para download.

FRANCISCO DE ASSIS MENÉZES BASTOS FILHO

**ELICITAÇÃO DE REQUISITOS NÃO FUNCIONAIS EM CONFORMIDADE COM
POLÍTICAS DE QUALIDADE PARA APLICAÇÕES MÉDICAS**

Dissertação submetida ao Mestrado em Engenharia em
Eletricidade da Universidade Federal do Maranhão
como parte dos requisitos para a obtenção do título de
Mestre em Ciência da Computação.

Orientador: Prof. Zair Abdelouahab.

São Luís
2007

FRANCISCO DE ASSIS MENÊZES BASTOS FILHO

**ELICITAÇÃO DE REQUISITOS NÃO FUNCIONAIS EM CONFORMIDADE COM
POLÍTICAS DE QUALIDADE PARA APLICAÇÕES MÉDICAS**

Dissertação submetida ao Mestrado em Engenharia em Eletricidade da Universidade Federal do Maranhão como parte dos requisitos para a obtenção do título de Mestre em Ciência da Computação.

Aprovada em _____/_____/2007

BANCA EXAMINADORA

Prof. PhD. Zair Abdelouahab (Orientador)

(Membro da Banca Examinadora)

(Membro da Banca Examinadora)

AGRADECIMENTOS

A Deus, por ter me dado força e luz, para concluir mais esta jornada.

A minha família, por ser meu apoio diário incentivando-me a buscar novos conhecimentos.

Ao meu orientador, Professor PhD. Zair Abdelouahab, pelo incentivo, compreensão, paciência e dedicação dispensada à realização desse trabalho.

Ao Professor. Dr. Denivaldo Cícero Pavão Lopes, pelas sugestões e orientações prestadas na conclusão deste trabalho.

A todos aqueles que contribuíram de forma direta ou indireta para realização desse trabalho.

RESUMO

Ao longo dos anos, muitas pesquisas vêm sendo desenvolvidas na tentativa de se encontrar uma solução para os problemas relacionados à engenharia de requisitos. Existe uma unanimidade em certas abordagens, em ressaltar que a fase de requisitos se constitui na etapa mais crucial no processo de desenvolvimento de software. Frequentemente, requisitos de software são mal elicitados, analisados e especificados, sendo estes fatores decisivos para o desenvolvimento de software de baixa qualidade. Requisitos não funcionais têm recebido muito pouca atenção na literatura, são pouco compreendidos mas são críticos no processo de desenvolvimento de software. O fato destes requisitos serem mal elicitados ou não elicitados os tornam mais difíceis de serem tratados e validados. Acreditamos que ao abordarmos os requisitos não funcionais desde a fase de elicitação, estamos contribuindo para a qualidade geral do software que será produzido. Este trabalho aborda dois aspectos relacionados aos requisitos não funcionais: como elicitá-los e como garantir a conformidade desses requisitos com uma política de qualidade, mesmo quando novos requisitos ou novas tecnologias forem adotadas. Para atingir este objetivo, elicitamos os requisitos a partir de um conjunto de necessidades de qualidade definidas para aplicações médicas. Definimos uma política de qualidade baseada nestas necessidades, de modo que possamos medir a conformidade das definições dos requisitos não funcionais especificados no Documento de Requisitos (DRs) com a qualidade desejada, especificada na política de qualidade.

Palavras-Chave: Engenharia de Requisitos, Elicitação, Requisitos, Políticas de Qualidade, Aplicações Médicas.

ABSTRACT

For several years, a lot of research is being done trying to find a solution for various problems concerning requirements engineering. Various approaches have pointed out that the requirements phase is the most important stage in the software development process. In general, requirements that are mistakenly elicited, analyzed and specified, gives rise to the development of low quality software. Non functional requirements gained little attention in the literature and are less understood but they are critical factors to the software development. When these requirements are badly elicited or not elicited at all, this gives rise to a difficulty in their treatment and their validation. Thus, we think that treating non functional requirements from the elicitation phase we contribute to a good software quality. This work deals with two aspects related with non functional requirements, namely how to elicit them and how to guarantee those requirements in conformity with a quality policy, even when new requirements or new technologies are adopted. As to attain this aim, we have elicited the requirements starting from a defined quality set necessary for medical applications. We defined a quality policy based upon these needs in such a way we could measure the conformity of non functional requirements definition specified in the requirements Document (RDs) with the desired quality, specified in the quality policy.

Key words: Requirements Engineering, Elicitation, Requirements, Quality Policy, Medical Applications.

LISTA DE TABELAS

Tabela 1	Atribuição de pesos aos elementos da árvore	53
Tabela 2	Conformidade versus nota	54
Tabela 3	Lista de Requisitos não funcionais Elicitados	64
Tabela 4	Associação de Pesos aos RNFs de Confiabilidade	76
Tabela 5	Atribuição de notas aos RNFs Específicos do RNF Confiabilidade	77
Tabela 6	Cálculo da Conformidade da definição do RNF Confiabilidade	77
Tabela 7	Cálculo da Conformidade da definição do RNF Usabilidade	78
Tabela 8	Cálculo da Conformidade da definição do RNF Segurança	79
Tabela 9	Cálculo da Conformidade da definição do RNF Eficiência	79
Tabela 10	Cálculo da Conformidade da definição do RNF Interoperabilidade	80
Tabela 11	Cálculo da Conformidade da definição do RNF Manutenibilidade	80
Tabela 12	Cálculo da Conformidade da definição do RNF Flexibilidade	80
Tabela 13	Cálculo da Conformidade da definição do RNF Portabilidade	80
Tabela 14	Cálculo da Conformidade da definição do RNF Padrões de Intercomunicação	81
Tabela 15	Cálculo da Conformidade da definição do RNF Padrões de Vocabulário .	81
Tabela 16	Cálculo da Conformidade da definição do RNF Padrões de Interfaces	81
Tabela 17	Cálculo da Conformidade da definição do RNF Padrões de Segurança	81
Tabela 18	Cálculo da Conformidade da definição do RNF Padrões de Formatação e Conteúdo	82
Tabela 19	Resultado da conformidade dos RNFs Primários	82

LISTA DE SIGLAS

ACR	<i>American College of Radiology</i>
CID	Classificação Internacional de Doenças
DICOM	<i>Digital Imaging and Communications in Medicine</i>
DoD	<i>Department of Defense</i>
ECG	Eletrocardiograma
EEG	Eletrencefalograma
ER	Engenheiro de Requisitos
GEHR	<i>Good Electronic Health Record</i>
HL7	<i>Health Level Seven</i>
IEEE	Instituto de Engenharia Elétrica e Eletrônica
IMIA	<i>International Medical Informatics Association</i>
LOINC	<i>Logical Observation Identifier Names and Codes</i>
MIB	<i>Medical Information Bus</i>
NDI	<i>Non Developmental Items</i>
NEMA	<i>National Electrical Manufacturers Association</i>
PACS	<i>Picture Archiving and Communication Systems</i>
RF	Requisito Funcional
RNF	Requisito não Funcional
SEI	<i>Software Engineering Institute</i>
SIH	Sistema de Informações Hospitalar
SNOMED	<i>Standar Nomenclature of Medicine</i>
UdI	Universo de Informações
UMLS	<i>Unified Medical Language System</i>

LISTA DE FIGURAS

Figura 1	Classificação de RNFs proposta por Macedo	24
Figura 2	Classificação de RNFs segundo Sommerville	24
Figura 3	Árvore de Características de Qualidade de Software	25
Figura 4	Taxonomia para RNFs proposta por Cysneiros	26
Figura 5	Estrutura do Documento de Requisitos segundo IEEE830	28
Figura 6	Características de qualidade de SIH	33
Figura 7	Modelo de Qualidade do Prontuário Eletrônico do Paciente	34
Figura 8	Modelo Proposto para Qualidade de Software Médico	40
Figura 9	Possíveis Decomposições para os Requisitos Não Funcionais	41
Figura 10	Classificação de RNFs Proposta	42
Figura 11	Modelo de documento de requisitos proposto	43
Figura 12	Método Proposto	46
Figura 13	Árvore do RNF Confiabilidade e suas decomposições	48
Figura 14	Modelo da Planilha de informações dos RNFs Primários	49
Figura 15	Modelo da Planilha de informações dos RNFs Secundários	49
Figura 16	Modelo da Planilha de informações dos RNFs Específicos	50
Figura 17	Modelo da Política de Qualidade	52
Figura 18	Fórmula para o cálculo da nota de nó pai	54
Figura 19	Árvore de Decomposição dos RNFs Primários	70
Figura 20	Política de Qualidade para o Software	75
Figura 21	Diagrama de Entidade Relacionamento do Projeto	124
Figura 22	Principais casos de uso do Projeto	125
Figura 23	Diagrama de classe da ferramenta.....	128
Figura 24	Interface de Criação e abertura de Projetos	129
Figura 25	Menu Processos da Ferramenta	129
Figura 26	Interface de Cadastros dos Requisitos Primários	130
Figura 27	Interface de Decomposições dos Requisitos Não Funcionais	131
Figura 28	Interface para Inclusão de Requisitos Secundários e Específicos.....	131
Figura 29	Interface de Especificação dos Requisitos	133
Figura 30	Interface das Influências entre os Requisitos Primários	135
Figura 31	Interface de Categorização dos Requisitos Não Funcionais	136
Figura 32	Interface de Registro das Conformidades Desejadas	137

Figura 33	Interface de Associação dos Pesos aos Requisitos	138
Figura 34	Interface de Atribuição de Notas aos Requisitos Específicos	139

PLANILHAS DE REQUISITOS

Planilha 1	Planilha do RNF Primário Confiabilidade	83
Planilha 2	Planilha do RNF Primário Usabilidade	83
Planilha 3	Planilha do RNF Primário Segurança	84
Planilha 4	Planilha do RNF Primário Eficiência	84
Planilha 5	Planilha do RNF Primário Interoperabilidade	85
Planilha 6	Planilha do RNF Primário Manutenibilidade.....	85
Planilha 7	Planilha do RNF Primário Flexibilidade.....	86
Planilha 8	Planilha do RNF Primário Portabilidade	86
Planilha 9	Planilha do RNF Primário Padrões de Intercomunicação	87
Planilha 10	Planilha do RNF Primário Padrões de Vocabulário.....	87
Planilha 11	Planilha do RNF Primário Padrões de Interfaces.....	88
Planilha 12	Planilha do RNF Primário Padrões de Segurança.....	88
Planilha 13	Planilha do RNF Primário Padrões de Formatação e Conteúdo.....	89
Planilha 14	Planilha do RNF Secundário Tolerância a falhas	89
Planilha 15	Planilha do RNF Secundário Recuperabilidade em caso de falha..	90
Planilha 16	Planilha do RNF Secundário Maturidade.....	90
Planilha 17	Planilha do RNF Secundário Aspectos de aquisição de dados.....	91
Planilha 18	Planilha do RNF Secundário Uso de processos inteligentes.....	91
Planilha 19	Planilha do RNF Secundário Aspectos de saída	92
Planilha 20	Planilha do RNF Secundário Busca por diversas informações.....	92
Planilha 21	Planilha do RNF Secundário Saída dos resultados em vídeo ou impressora.....	93
Planilha 22	Planilha do RNF Secundário Aspectos de ajuda.....	93
Planilha 23	Planilha do RNF Secundário Manual do usuário atualizado.....	94
Planilha 24	Planilha do RNF Secundário Mensagens auto explicativas.....	94
Planilha 25	Planilha do RNF Secundário diferentes tipos de mensagens.....	95
Planilha 26	Planilha do RNF Secundário Aspectos Gerais de Usabilidade	95
Planilha 27	Planilha do RNF Secundário Mecanismos de administração de usuários.....	96
Planilha 28	Planilha do RNF Secundário Política de controle de acesso.....	964
Planilha 29	Planilha do RNF Secundário Política de Senhas	97
Planilha 30	Planilha do RNF Secundário Disponibilidade.....	97
Planilha 31	Planilha do RNF Secundário Segurança do ambiente.....	98

Planilha 32	Planilha do RNF Secundário Eficiência de execução.....	98
Planilha 33	Planilha do RNF Secundário Busca de imagens não pode ultrapassar 3 segundos.....	99
Planilha 34	Planilha do RNF Secundário Eficiência de Armazenamento	99
Planilha 35	Planilha do RNF Secundário Interoperabilidade Técnica.....	99
Planilha 36	Planilha do RNF Secundário Interoperabilidade Semântica.....	100
Planilha 37	Planilha do RNF Secundário Interface do usuário.....	100
Planilha 38	Planilha do RNF Secundário Usar padrões gráficos de interfaces	101
Planilha 39	Planilha do RNF Secundário Interface de software.....	101
Planilha 40	Planilha do RNF Secundário Padrões de segurança para web.....	102
Planilha 41	Planilha do RNF Específico Persistência dos dados.....	102
Planilha 42	Planilha do RNF Específico Completeza.....	102
Planilha 43	Planilha do RNF Específico Precisão.....	103
Planilha 44	Planilha do RNF Específico Consistência dos dados.....	103
Planilha 45	Planilha do RNF Específico Capacidade de reverter funções de efeito drástico.....	103
Planilha 46	Planilha do RNF Específico Capacidade de interromper um processamento demorado.....	104
Planilha 47	Planilha do RNF Específico Prevenção e contingência.....	104
Planilha 48	Planilha do RNF Específico Identificação única para o paciente....	104
Planilha 49	Planilha do RNF Específico Importação de dados externos.....	105
Planilha 50	Planilha do RNF Específico Reconhecimento de textos, sons e imagens.....	105
Planilha 51	Planilha do RNF Específico Aquisição de textos, sons e imagens..	105
Planilha 52	Planilha do RNF Específico Estrutura do prontuário do paciente...	106
Planilha 53	Planilha do RNF Específico Apresentação uniforme dos dados do paciente.....	106
Planilha 54	Planilha do RNF Específico Exporta dados do paciente para outros sistemas.....	106
Planilha 55	Planilha do RNF Específico Elaboração de estatística com geração automática de gráficos.....	107
Planilha 56	Planilha do RNF Específico Utiliza código de barra.....	107
Planilha 57	Planilha do RNF Específico Assinaturas digitais dos relatórios....	107
Planilha 58	Planilha do RNF Específico Busca de imagens por conteúdo.....	108
Planilha 59	Planilha do RNF Específico Ajuda <i>online</i> sensível ao contexto....	108
Planilha 60	Planilha do RNF Específico Documentação do usuário de fácil compreensão.....	108
Planilha 61	Planilha do RNF Específico Sumário e índice remissivo na documentação do usuário.....	109
Planilha 62	Planilha do RNF Específico Presença de manual de instalação	109

	com instruções detalhadas.....	
Planilha 63	Planilha do RNF Específico Consistência de vocabulário entre as mensagens e a documentação.....	109
Planilha 64	Planilha do RNF Específico Mensagens de erro com informações necessárias para solução da situação de erro.....	110
Planilha 65	Planilha do RNF Específico Alertas claros para as conseqüências de uma determinada confirmação.....	110
Planilha 66	Planilha do RNF Específico Aprendibilidade.....	111
Planilha 67	Planilha do RNF Específico Operacionalidade.....	111
Planilha 68	Planilha do RNF Específico Atratividade.....	111
Planilha 69	Planilha do RNF Específico Registro das autorizações de acesso	112
Planilha 70	Planilha do RNF Específico Autenticação.....	112
Planilha 71	Planilha do RNF Específico Identificação única de usuários.....	112
Planilha 72	Planilha do RNF Específico Limitar a existência de senhas.....	113
Planilha 73	Planilha do RNF Específico Atualização periódica de senhas.....	113
Planilha 74	Planilha do RNF Específico Não reutilização de senhas antigas....	113
Planilha 75	Planilha do RNF Específico Proibir a auditoria de informações relativas a senhas.....	114
Planilha 76	Planilha do RNF Específico Manipular senhas criptografadas.....	114
Planilha 77	Planilha do RNF Específico Redundância de hardware e software	114
Planilha 78	Planilha do RNF Específico Cópias de segurança	115
Planilha 79	Planilha do RNF Específico Auditoria.....	115
Planilha 80	Planilha do RNF Específico Restringir acesso a sala dos servidores.....	115
Planilha 81	Planilha do RNF Específico Proteção contra incêndios e inundações.....	116
Planilha 82	Planilha do RNF Específico Usar indexação de imagens médicas	116
Planilha 83	Planilha do RNF Específico Armazenar imagens compactadas.....	116
Planilha 84	Planilha do RNF Específico Compatibilidade.....	117
Planilha 85	Planilha do RNF Específico Inteligibilidade.....	117
Planilha 86	Planilha do RNF Específico Analisabilidade.....	117
Planilha 87	Planilha do RNF Específico Modificabilidade.....	118
Planilha 88	Planilha do RNF Específico Estabilidade.....	118
Planilha 89	Planilha do RNF Específico Testabilidade.....	118
Planilha 90	Planilha do RNF Específico Independência dos dados do paciente	119
Planilha 91	Planilha do RNF Específico Adaptabilidade.....	119
Planilha 92	Planilha do RNF Específico Coexistência.....	119

Planilha 93	Planilha do RNF Específico Utilizar protocolos padronizados.....	120
Planilha 94	Planilha do RNF Específico uniformidade do formato dos dados transmitidos.....	120
Planilha 95	Planilha do RNF Específico Utilizar padrões para identificar as informações.....	120
Planilha 96	Planilha do RNF Específico Clareza nos formatos das telas e relatórios.....	121
Planilha 97	Planilha do RNF Específico Utilizar terminologia condizente com a prática médica.....	121
Planilha 98	Planilha do RNF Específico Interface de hardware.....	121
Planilha 99	Planilha do RNF Específico Prover padrões de interoperabilidade e portabilidade.....	122
Planilha 100	Planilha do RNF Específico Padrões XML.....	122
Planilha 101	Planilha do RNF Específico Usar padrões de conteúdo.....	122

SUMÁRIO

	LISTA DE SIGLAS	
	LISTA DE FIGURAS	
	PLANILHAS DE REQUISITOS	
1	INTRODUÇÃO	16
1.1	Descrição do problema	16
1.2	Objetivos	17
1.3	Estrutura da Dissertação	17
2	ESTADO DA ARTE	18
2.1	Introdução	19
2.2	Requisitos	19
2.2.1	Conceitos	19
2.2.2	Requisitos Não Funcionais	22
2.2.3	Classificação dos Requisitos Não Funcionais	24
2.2.4	Documento de Requisitos	26
2.3	Qualidade de Software	29
2.4	Qualidade de Software Médico	30
2.4.1	Modelo de Qualidade para Sistemas de Informações Hospitalares	31
2.4.2	Modelo de Qualidade para Prontuário Médico	33
2.5	Padrões em Software Médico	35
2.5.1	Problemas causados pela ausência de padrões	35
2.5.2	Padrões utilizados em aplicações médicas	36
3	SOLUÇÃO PROPOSTA	40
3.1	Modelo de Qualidade proposto para software médico	40
3.2	Classificação dos requisitos não funcionais proposta	41
3.3	Modelo do documento de requisitos proposto	43
3.4	O Método Proposto	45
3.4.1	Fase de Elicitação	46
3.4.2	Fase de Decomposição e Representação	47
3.4.3	Fase de Especificação	48
3.4.4	Fase de Identificação das Influências	50

3.4.5	Fase de Criação da política de qualidade	51
3.4.6	Fase de Avaliação de Conformidades	52
3.5	Estudo Comparativo	55
4	ESTUDO DE CASO	57
5	FERRAMENTA PARA ELICITAÇÃO DOS REQUISITOS	124
5.1	Modelagem	124
5.2	Prototipagem	129
5.2.1	Criar Projetos	129
5.2.2	Cadastrar requisitos primários	130
5.2.3	Cadastrar decomposição	131
5.2.4	Especificar requisitos	133
5.2.5	Cadastrar influências	134
5.2.6	Categorizar requisitos	136
5.2.7	Registrar conformidades desejadas	137
5.2.8	Associar pesos	138
5.2.9	Atribuir notas	139
6	CONCLUSÃO	141
6.1	Contribuições do Trabalho	142
6.2	Trabalhos Futuros	142
	REFERÊNCIAS	144

1 INTRODUÇÃO

1.1 Descrição do Problema

O desenvolvimento de produtos de software cada vez mais complexos, passíveis de avaliação e com menor custo possível, tem-se tornado um desafio constante para a comunidade de engenharia de software. Diversas técnicas, metodologias e ferramentas vêm sendo propostas com o intuito de suportar e auxiliar a produção de software de qualidade (CASTRO 2001). Neste contexto, uma das etapas mais críticas está relacionada à engenharia de requisitos. Solucionar problemas de requisitos incompletos e inconsistentes ainda nas etapas iniciais do processo de desenvolvimento possibilita a obtenção de produtos de software mais confiáveis e diminui os custos de manutenção.

A crescente complexidade dos sistemas de software e o aumento da exigência de qualidade por parte dos clientes vêm impulsionando o mercado a cada dia produzir mais software que atendam, não somente as funcionalidades exigidas, mas também a aspectos não funcionais exigidos pelos clientes. Estes aspectos não funcionais devem ser tratados como RNFs (Requisitos Não Funcionais) do software. Devem ainda ser tratados desde o início do seu desenvolvimento (CHUNG 1999), estendendo este tratamento por todo o ciclo de vida do software.

Os requisitos não funcionais têm recebido muito pouca atenção na literatura e são muito menos compreendidos do que os outros fatores menos críticos ao desenvolvimento de software (CHUNG 1999). Quando citados nos processos de desenvolvimento, estes requisitos são abordados de maneira secundária e altamente informal do ponto de vista da elicitação. Este tipo de abordagem faz com que, quando tratados, os requisitos não funcionais sejam freqüentemente contraditórios, difíceis de serem considerados e validados durante o desenvolvimento de software.

Apresentamos nesta dissertação um processo para lidar com requisitos não funcionais nas as etapas iniciais do processo de desenvolvimento de software. Propomos uma estratégia que se baseia no uso de um conjunto de necessidades de qualidade para definir uma política de qualidade para o software que será desenvolvido. A construção desta política de qualidade, durante a fase de requisitos, é norteada pelo modelo das políticas proposto por (ROCHA, ABDELOUAHAB 2004).

A quantificação do grau de conformidade dos requisitos com a política de qualidade possibilita uma visão antecipada da qualidade final do software, bem como um

menor tempo de entrega e menores custos de manutenção. Contudo, quando novas tecnologias e novas funcionalidades forem adotadas, a política de qualidade deve ser revista e possivelmente alterada para responder aos possíveis conflitos introduzidos.

Neste contexto, pretendemos demonstrar que, uma vez criada a política de qualidade para o software, ela pode continuar consistente com as definições dos RNFs especificadas no documento de requisitos, mesmo quando novas tecnologias e funcionalidades forem adotadas para o software. Portanto, há uma necessidade de evoluir as abordagens para o desenvolvimento de políticas de qualidade de software.

1.2 Objetivos

Para atingir os objetivos descritos na seção 1.1, este trabalho visa:

- √ Investigar abordagens sobre requisitos de software, enfatizando os requisitos não funcionais;
- √ Investigar abordagens de qualidade de software;
- √ Abordar a qualidade de software médicos;
- √ Investigar os principais padrões de qualidade utilizados em software médicos e os problemas causados pelo não uso;
- √ Propor um método para elicitação de requisitos não funcionais em obediência com a política de qualidade;
- √ Mostrar a aplicabilidade do método em um estudo de caso.

1.3 Estrutura da Dissertação

Além deste capítulo introdutório, este trabalho possui outros cinco capítulos e dois apêndices, como veremos a seguir:

- √ *Capítulo 2 – Estado da Arte*

Este capítulo ilustra o estado da arte, considerando a fase da engenharia de requisitos abordando: os principais conceitos relacionados a etapa de requisitos; a qualidade de software; a qualidade de software médicos e os principais padrões utilizados nesses software.

- √ *Capítulo 3 – Solução Proposta*

O terceiro capítulo se destina a solução proposta para elicitação de requisitos não funcionais para aplicações de software médico, em obediência com a política de qualidade e um estudo comparativo da metodologia proposta com outros trabalhos.

√ *Capítulo 4 – Estudo de Caso*

O quarto capítulo foi destinado à aplicação do método em um estudo de caso para elicitar os requisitos não funcionais de um sistema de informações hospitalares.

√ *Capítulo 5 – Ferramenta para elicitação dos requisitos*

Este capítulo se destina a demonstração de uma ferramenta implementada para solução proposta destacando a modelagem e um protótipo.

√ *Capítulo 6 – Conclusão*

Finalmente, no quinto capítulo são feitas as considerações finais, com um relato das perspectivas de trabalhos futuros e das contribuições desse trabalho.

2 ESTADO DA ARTE

2.1 Introdução

O objetivo deste capítulo é abordar sobre requisitos e qualidade de software, destacando os requisitos não funcionais e suas classificações. Também almeja descrever sobre a qualidade de software, qualidade de software médicos e os padrões de qualidade utilizados nestes tipos de aplicações.

Logo, na seção 2.1, abordagens sobre requisitos de software são apresentadas, enfatizando: seus conceitos, os RNFs, suas classificações e o documento de requisito. Na seção 2.2 são descritas abordagens sobre qualidade de software. A seção 2.3 descreve sobre a qualidade de software médicos enfatizando os modelos de qualidade para Sistemas de Informações Hospitalares e para o Prontuário Médico.

Finalmente a seção 2.4 descreve abordagens sobre os padrões em software médico destacando os problemas causados pelo não uso de tais padrões.

2.2 Requisito

2.2.1 Conceitos

A etapa de Engenharia de Requisitos se constitui na fase crítica do ciclo de vida de um sistema, pois uma compreensão completa dos requisitos de software é fundamental para um bem-sucedido desenvolvimento do mesmo (CARVALHO 2001). A base do Software são seus requisitos, por isso é necessário que o produto final satisfaça todas as necessidades dos *stakeholders*, se não toda a estrutura poderá ficar comprometida. Quanto mais cedo se detectam problemas de requisitos no ciclo de desenvolvimento de um software, mais fácil e menos dispendioso é sua correção (NUSEIBEH 2000).

É objetivo da Engenharia de Requisitos, sistematizar o processo de definição dos requisitos de software. Para isso estabelece um processo que deve lidar com diferentes pontos de vista, e usar uma combinação de métodos, ferramentas e pessoal. Deste processo, obtém-se como produto um modelo do qual um documento chamado requisitos é produzido. Esse processo acontece num contexto previamente definido a que chamamos de Udi (Universo de Informações). Universo de Informações é o contexto no qual o software deverá ser desenvolvido e operado, inclui todas as fontes de informação e pessoas relacionadas ao software.

Para melhor entendimento da Engenharia de Requisitos é necessário definir o que significa requisito. Segundo o IEEE (Instituto de Engenharia Elétrica e Eletrônica) e Brown (1998) um requisito é definido como:

- √ Uma condição ou capacidade necessária para um usuário resolver um problema ou atingir um objetivo;
- √ Uma condição ou capacidade que deve ser atingida ou possuída por um sistema ou componente de sistema para satisfazer um contrato, padrão, especificação, ou outro documento de formalidade;
- √ Uma representação documentada de uma condição ou capacidade como nos dois conceitos anteriores.

Definir requisitos, geralmente, é expressar, em forma de documento, quais características uma entidade deve possuir para ser aceita (PAULA F. 2001). Um requisito é:

- √ Explícito, se descrito num documento de especificação de requisitos;
- √ Implícito, quando não é documentado, porém cobrado em decorrência de sua experiência no momento de utilização;
- √ Normativo, quando decorre de lei, regulamentos, padrões e outros tipos de regras que devem ser obedecidas.

Quando se definem requisitos para um ambiente, um sistema de informação ou um produto, as três formas de apresentação de um requisito devem ser levadas em consideração. Em geral, as necessidades explícitas são expressas na definição de requisitos propostos após o levantamento das necessidades. O enfoque da qualidade centrado no atendimento a esses requisitos é denominado "conformidade com os requisitos" (ISO/IEC 9126, 2001).

Tendo-se o entendimento de requisito, pode-se agora definir Engenharia de Requisitos. Na literatura existem diversos conceitos:

- √ “Engenharia de Requisitos é a aplicação disciplinada de demonstrar princípios, métodos, ferramentas e notações para descrever o comportamento pretendido de um sistema e suas restrições associadas” (HSIA 1993);
- √ “Engenharia de Requisitos é o ramo da Engenharia de Software que se interessa por metas do mundo real com respeito a funções e restrições nos sistemas de software. Preocupa-se com a relação desses fatores com especificações precisas do comportamento do software, e a sua evolução com o passar do tempo através de famílias de software” (FINKELSTEIN 1995);

- √ “A Engenharia de Requisitos consiste de todas as atividades envolvidas no descobrimento, documentação e gerenciamento de um conjunto de requisitos para um sistema baseado em computador” (SOMMERVILLE 1998).

Segundo o IEEE, referenciado em Alves (2001), Engenharia de Requisitos corresponde ao processo de aquisição, representação, refinamento e verificação das necessidades do cliente para um sistema de software, objetivando-se ter uma especificação completa e correta dos requisitos de software. Os conceitos refletem o mesmo significado e têm como essência o conhecimento do que é desejado por todos os interessados no produto final. Os requisitos devidamente especificados e modelados servem de base para a análise. A seguir serão abordados mais alguns conceitos relacionados a Engenharia de Requisitos.

A abordagem de Fook (2001) define *stakeholders* como: “Qualquer pessoa que é afetada de alguma forma pelo sistema”; “São pessoas que têm influência direta ou indireta nos requisitos do sistema”; “Qualquer pessoa que tem algum interesse na empresa ou sistema”; “São indivíduos ou organizações que estão prestes a ganhar ou perder com o sucesso ou fracasso de um sistema”. Desta forma, os que pagam, os que projetam e os que utilizam o sistema são considerados *Stakeholders* e a interação entre eles é de fundamental importância para o sucesso da Engenharia de Requisitos.

Entendemos por requisitos de software, sentenças que expressam as necessidades dos usuários e que condicionam a qualidade do software. Em função disso classificamos requisitos como funcionais e não funcionais. O primeiro está relacionado à funcionalidade do software, enquanto o segundo expressa qualidades específicas que o software deve ter ou restrições que o software deve atender. As três sentenças a seguir exemplificam essa distinção:

- √ O sistema deve prover uma interface de entrada para registrar os resultados dos exames clínicos, de laboratórios e radiológicos de um paciente (RF);
- √ Exames de laboratórios fora da faixa de normalidade, somente o bioquímico responsável pode efetuar a entrada do resultado (RNF de confidencialidade);
- √ O sistema deve recuperar e mostrar as 10 (dez) imagens mais semelhantes a uma imagem de consulta, com o tempo máximo de 5 segundos após a requisição da consulta (RF , RNF de desempenho).

De acordo com Davis (1993), a diferença entre requisitos funcionais e não-funcionais está no fato de os primeiros descreverem “o que” o sistema deve fazer, enquanto que os outros fixam restrições sobre “como” os requisitos funcionais serão implementados.

Nesta dissertação, abordaremos em detalhe os requisitos não funcionais, uma vez que estes são os que têm maior relação com o nosso trabalho.

2.2.2 Requisitos Não Funcionais

De maneira geral, a complexidade de um software é determinada por suas funcionalidades e por requisitos gerais que fazem parte do desenvolvimento do software como, desempenho, confiabilidade, manutenibilidade, portabilidade, custos entre outros (CHUNG 1999). Estes requisitos, denominados por Roman (1985) de requisitos não funcionais, são também conhecidos como atributos de qualidade (KELLER 1990), restrições, objetivos entre outros. Adotaremos neste trabalho a nomenclatura requisitos não funcionais, já que este termo vem sendo bastante usado no meio acadêmico.

Durante o desenvolvimento de software, estes requisitos possuem um papel crítico. Alguns aspectos inerentes à sua natureza aumentam a complexidade com que eles devem ser tratados, por exemplo:

- √ Subjetividade: permite que pessoas diferentes tenham visões, interpretações e conceitos diferentes, relacionados ao mesmo requisito;
- √ Relatividade: sua interpretação e importância dependem diretamente de cada sistema em particular, além disso, sua realização é relativa, por exemplo: um RNF pode ser essencial para uma pessoa enquanto para outra não;
- √ Interatividade: eles interagem entre si, assim a realização de um RNF pode influenciar de maneira positiva ou negativa outros requisitos;
- √ Limitações: o tempo de resposta é inerente ao projeto que ainda não é conhecido na fase de requisitos;
- √ Não existem regras bem definidas para determinar quando um RNF está otimizado.

Os RNFs criam conflitos e contradições entre eles próprios, uma vez que a tentativa de satisfazer um pode prejudicar ou ajudar a satisfazer outros. A seguir destacamos alguns exemplos de conflitos entre estes requisitos:

- √ Segurança x Flexibilidade: a segurança precisa utilizar estruturas específicas, que possam garantir certas limitações, enquanto a flexibilidade requer estruturas mais gerais;
- √ Segurança x Desempenho: o processo adicional requerido para controlar o acesso ao sistema faz com que o tempo de resposta seja maior;

- √ Usabilidade x Desempenho: processos adicionais para facilitar as tarefas dos usuários, geralmente aumentam o tempo de execução de determinados eventos.

Apesar do reconhecimento da importância dos RNFs, os métodos para especificar sistemas têm sua abordagem muito mais centrada nos requisitos funcionais do que nos RNFs. No desenvolvimento de aplicações médicas, os RNFs desempenham um papel mais crítico ainda. Problemas na elicitacão deste tipo de requisito podem trazer conseqüências desastrosas, visto que falhas em produtos de software médicos podem resultar em graves danos pessoais ou, até mesmo, a perdas de vidas humanas (ROCHA 2001).

Alguns RNFs são comuns a diversos domínios de aplicações, outros adquirem maior relevância em domínios específicos. Por exemplo, os requisitos a seguir têm grande relevância e visibilidade quando aplicados em software médicos, tais como:

- √ Segurança: a segurança de um sistema médico envolve fatores como a privacidade das informações do paciente e segurança nas transações. Quanto à privacidade das informações, é necessário garantir que as informações fornecidas pelos pacientes não serão utilizadas de forma indevida. Já a segurança das transações é uma questão técnica que deve ser resolvida através do uso de técnicas como criptografia e controle de acesso;
- √ Usabilidade: a usabilidade em aplicações médicas é um fator extremamente relevante, pois um software médico fácil de usar motiva seus usuários e reduz a resistência por parte dos profissionais da área da saúde quanto ao uso;
- √ Desempenho: a velocidade com que determinadas tarefas são executadas pode contribuir para satisfação dos usuários. A questão do desempenho em aplicações médicas está ligada principalmente a fatores técnicos como tamanho dos arquivos e quantidade de informações manipuladas;
- √ Consistência: inconsistência em informações médicas pode colocar o profissional da área da saúde em dúvida, quanto a confiabilidade da informação. Isso pode comprometer a segurança do profissional no ato de tomar uma decisão;
- √ Disponibilidade: a grande maioria das aplicações médicas deve funcionar vinte e quatro horas por dia e sete dias por semana. Mecanismos de segurança e tolerância a falhas contribuem positivamente para garantir a disponibilidade do software;
- √ Manutenibilidade: a questão da manutenibilidade de sistemas médicos está diretamente ligada à disponibilidade, já que manutenção em aplicações médicas deve ser feita sem que o serviço fique indisponível;

- √ Padronização: a padronização em aplicações médicas é um requisito indispensável para possibilitar a troca de informações entre os diversos tipos de aplicações interligadas.

2.2.3 Classificação dos Requisitos Não Funcionais

Segundo Macedo (1999), os requisitos não funcionais podem ser divididos em: requisitos de qualidade geral, de abrangência e de operação. Além disso, podem ser classificados como requisitos não funcionais primários e requisitos não funcionais específicos. Algumas classificações dos RNFs podem ser encontradas na literatura. Macedo (1999) propõe a classificação de RNFs apresentada na Figura 1.

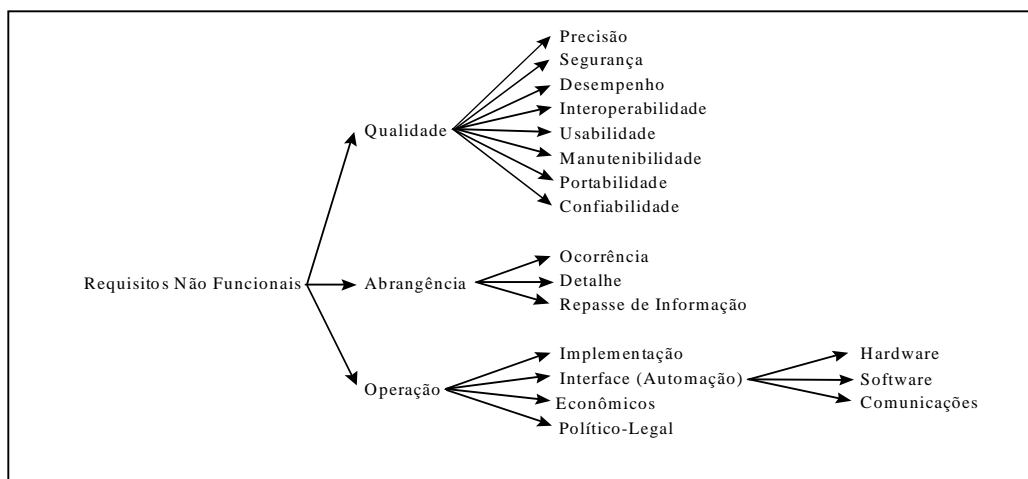


Figura 1 - Classificação de RNFs proposta por (MACEDO 1999)

A Figura 2 apresenta a classificação de RNFs proposta por Sommerville (1992).

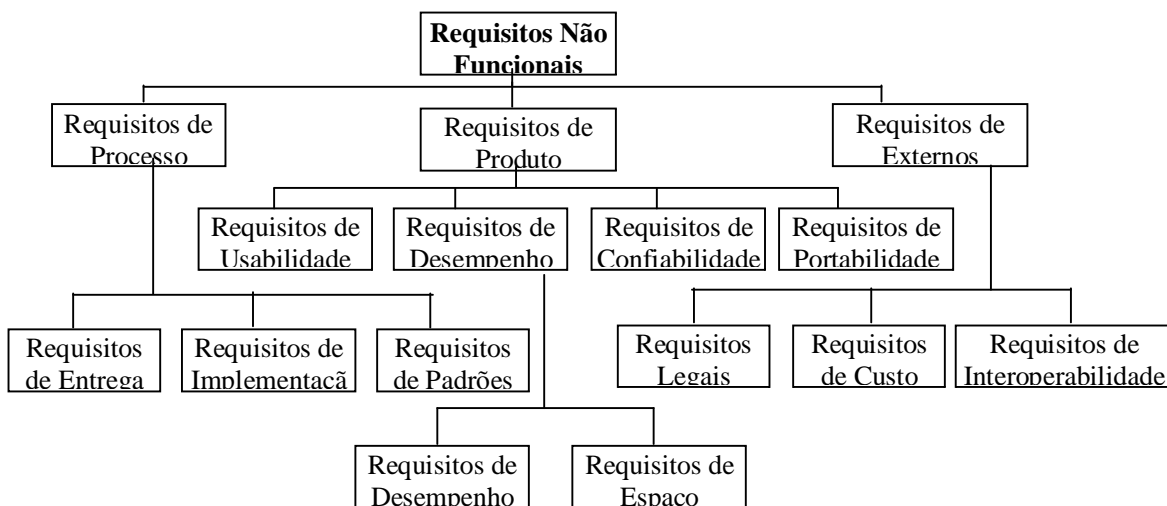


Figura 2 - Classificação de RNFs segundo (SOMMERVILLE 1992)

Outra classificação para RNFs foi proposta por Boehm (1976), Figura 3, na qual ele define uma árvore mínima de padrões de qualidade que um software deveria apresentar.

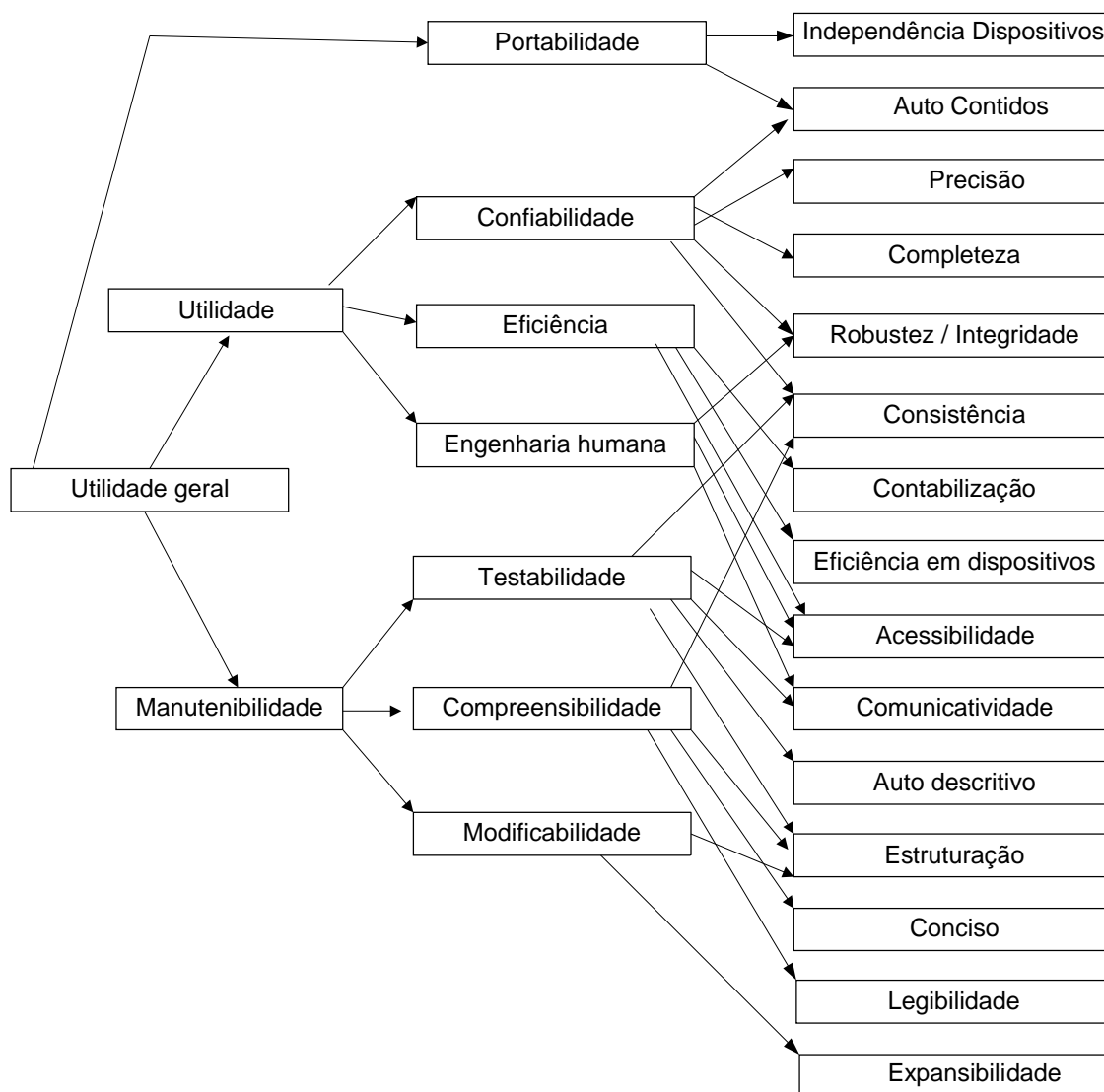


Figura 3 - Árvore de Características de Qualidade de Software (BOEHM 1976)

Uma outra fonte que classifica os RNFs é o padrão internacional ISO 9126 (ISO9126), que descreve seis características para definir a qualidade de um software. Cysneiros (2002) propôs uma taxonomia que classifica os RNFs em primários e específicos. Segundo este autor, RNFs primários são mais abstratos e representam propriedades como: Confiabilidade, Rastreabilidade, Precisão e Segurança. Já os RNFs específicos são oriundos de decomposições dos RNFs primários, tendo portanto um nível de abstração menor.

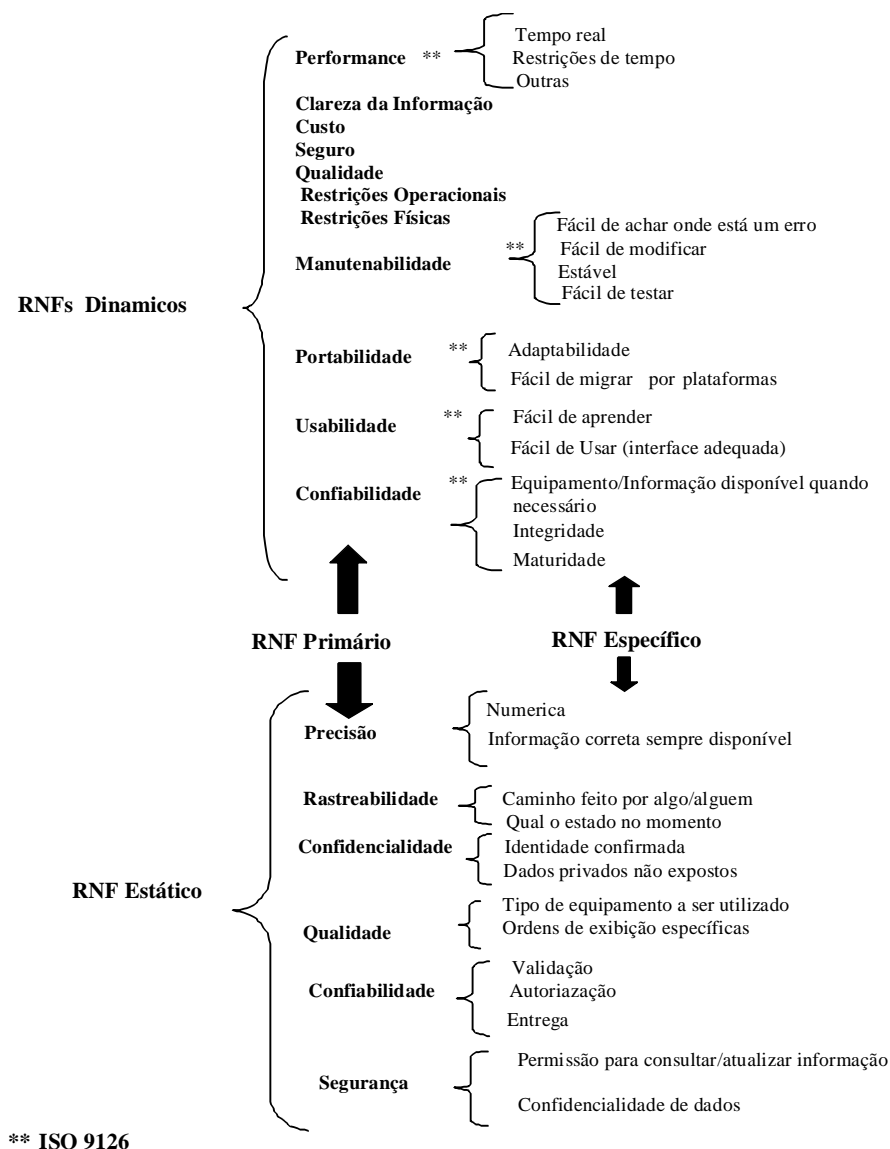


Figura 4 – Taxonomia para RNFs proposta por (CYSNEIROS 2002)

2.2.4 Documento de Requisitos

Uma vez identificados e negociados, os requisitos devem ser documentados para que possam servir de base para o restante do processo de desenvolvimento. A linguagem natural é capaz de ser compreendida por todos os participantes do processo de requisitos e, dessa forma, pode ser utilizada para representá-los (KOTONYA 1996). Sempre que for necessário, pode-se incluir à documentação dos requisitos tais como diagramas, gráficos, tabelas entre outros que facilitem o entendimento e a elucidação das informações registradas. Davis (1993) destaca que, apesar da vantagem da linguagem natural ser entendida por todos

os participantes do processo de requisitos, ela apresenta um alto grau de ambigüidade favorecendo o aparecimento de inconsistências.

O documento de requisitos é a declaração oficial do que é desejado do sistema. Descreve seus requisitos e é utilizado para prover comunicação entre usuários, desenvolvedores e gerentes. Possui um conjunto diversificado de usuários, abrangendo desde a alta gerência até usuários finais e engenheiros de software. Por isso, em muitos casos, existem versões distintas do documento (SOMMERVILLE 2003). Não existe um nome padrão único para representá-lo, em diferentes organizações, diferentes denominações são utilizadas como especificação funcional, especificação dos requisitos do sistema etc. Seguem alguns tópicos que o documento de requisitos poderá abordar:

- √ Serviços e funções que o sistema deve prover;
- √ Restrições sobre as quais o sistema deve operar;
- √ Propriedades gerais do sistema, isto é, comportamento do sistema diante das propriedades emergentes;
- √ Definições de outros sistemas que irão interagir com o sistema em questão;
- √ Informações a respeito do domínio de aplicação do sistema;
- √ Limitações no processo utilizado para desenvolver o sistema;
- √ Especificações de hardware e software no qual o sistema executará.

Adicionalmente, o documento de requisitos deverá conter um capítulo introdutório que provê um resumo do sistema, necessidades de negócio suportadas pelo sistema e um glossário que explica a terminologia utilizada (LEITE 1997). Esse glossário é particularmente importante, pois pessoas de diferentes áreas terão contato com o documento de requisitos, facilitando assim sua compreensão. A seguir exemplificamos alguns usuários do documento de requisitos, com seus respectivos papéis:

- √ Clientes do Sistema: especificam os requisitos e o lêem para avaliar se os mesmos satisfazem suas necessidades;
- √ Gerentes de Projeto: usam os documentos de requisitos para planejarem uma proposta para o sistema e para o processo de desenvolvimento do mesmo;
- √ Engenheiros de Sistema: usam os requisitos para entenderem o sistema em construção e desenvolverem testes de validação do mesmo.

Muitas organizações propuseram uma estrutura para o documento de requisitos. Entre elas o padrão IEEE (IEEE830 1998), sugere uma estrutura para o documento de requisitos, apresentada na Figura 5.

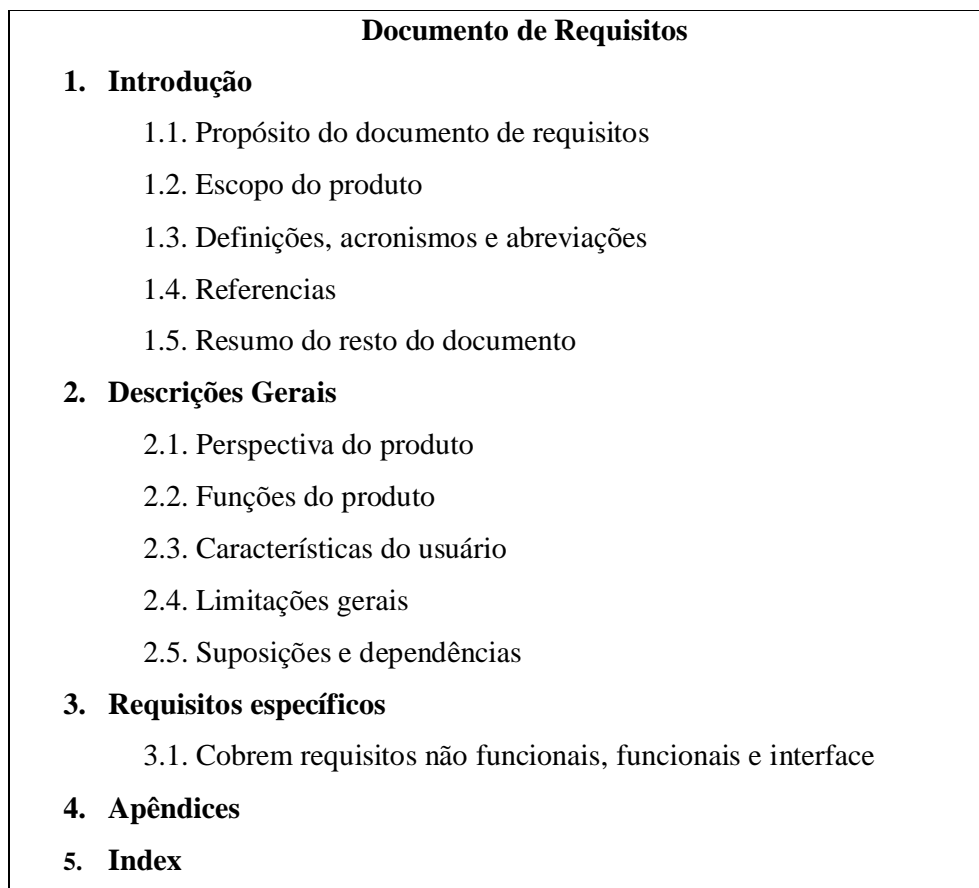


Figura 5 - Estrutura do Documento de Requisitos segundo (IEEE830 1998).

É possível omitir partes e adicionar novas seções, se necessário. Também podemos definir uma lista de partes estáveis e variáveis. Partes estáveis devem aparecer em todos os documentos de requisitos, sejam quais forem os seus leitores. Partes variáveis devem ser incluídas dependendo do grupo alvo. É como se existisse uma versão executiva, mais resumida, e uma versão técnica, contendo todos os detalhes.

Para Sommerville (1998), Documento de Requisitos é a declaração formal dos requisitos de um sistema para clientes, usuários finais e desenvolvedores de software. Também conhecido como 'Documento de Especificação de Software', ele é a síntese da Engenharia de Requisitos, pois contém os requisitos funcionais e não-funcionais do sistema a ser desenvolvido, validado e acordado pelos *stakeholders*. Este documento deve ser consistente, correto e completo com relação a satisfação das necessidades, entendido pelos usuários, clientes, projetistas e provedores, e capaz de servir como base para o projeto (HSIA 1993). Percebe-se então que um bom documento de requisitos deve ser claro, preciso e sem ambigüidades.

Na seção 3.3 deste trabalho, propomos um modelo para o documento de especificação de requisitos, com o objetivo de fornecer um meio padrão para especificar os requisitos elicitados, o qual poderá ser útil tanto para as equipes de desenvolvimento, na tentativa de facilitar a construção de sistemas, quanto para as equipes de análises, nas futuras manutenções corretivas ou evolutivas.

2.3 Qualidade de Software

A cada dia as empresas tornam-se mais dependentes dos seus sistemas de informações. Construir esses sistemas em tempo hábil para serem úteis aos negócios com a qualidade e custos adequados à sua importância para a organização é o desafio que todos os desenvolvedores estão enfrentando. O tema qualidade de software vem sendo aprofundado em suas diferentes visões: qualidade de processo, qualidade de produto e qualidade em uso. Para Rocha (2001), qualidade é um grupo de requisitos que devem ser cumpridos, a fim de se obter a satisfação do usuário final.

Pressman (2001) define qualidade de software como: “conformidade a requisitos funcionais e de desempenho explicitamente declarados. A padrões de desenvolvimento claramente documentados e a características implícitas que são esperadas de todo software profissionalmente desenvolvido”.

Segundo Paula (2001), qualidade do processo é o grau com que ele garante a qualidade dos respectivos produtos de software, enquanto a qualidade do produto é o grau de conformidade com os respectivos requisitos (BOEHM 1978) mostra que a qualidade geral do produto final de um software pode ser melhorada simplesmente tornando os desenvolvedores conscientes de que critérios de qualidade deviam ser cumpridos. Também baseados em uma abordagem quantitativa sob a ótica da qualidade de software, Basili e Musa (1991) propõem modelos e métricas para o processo de engenharia de software de uma perspectiva gerencial.

Abordar qualidade como requisitos não funcionais integra aspectos de qualidade na definição do próprio software. Com isso, o gerenciamento de requisitos passa a lidar com dois aspectos: de funcionalidade e de qualidade. Portanto, gerenciar requisitos torna-se um processo de gerência de qualidade, na medida em que procura tratar da alocação dos requisitos, de maneira a garantir, não só que os requisitos estão sendo seguidos, como também são rastreáveis.

Duas diferentes abordagens são utilizadas em tratamento sistemático de RNFs. Elas podem ser classificadas como orientada a produto e orientada a processo

(CHUNG 1993). A primeira aborda RNFs de forma a classificar o quanto um sistema satisfaz os RNFs que dele são requeridos. Quase todos os trabalhos em RNFs utilizam esta abordagem, orientada a produto, a qual é bem descrita por Keller (1990).

RNFs abordam importantes aspectos relacionados à qualidade de software. Eles são essenciais para que software sejam bem sucedidos. A ausência destes requisitos pode resultar em: software com inconsistência e de baixa qualidade; clientes e desenvolvedores insatisfeitos; tempo e custo de desenvolvimento além dos previstos devido à necessidade de se consertar software que não foram desenvolvidos sob a ótica da utilização de RNFs.

Cysneiros (2002) aborda dois aspectos de como lidar com RNFs: como elicitá-los e como integrá-los aos modelos conceituais. Para isso, este autor propõe uma estratégia que trata da elicitação dos RNFs desde o início do processo de desenvolvimento de software e de como integrar os requisitos elicitados aos modelos conceituais. Segundo este autor, o uso desta estratégia pode levar a ganhos na qualidade do modelo conceitual final, bem como a um processo de desenvolvimento de software mais produtivo.

Contudo, fica evidente que a questão da qualidade é tratada na fase de requisitos através da definição clara dos requisitos não funcionais que o software deverá atender. Um ponto que se observa com certa frequência é que logo após algum tempo em que os requisitos foram elicitados, muitos destes requisitos se tornam obsoletos ou pela evolução do software ou por adoção de novas tecnologias por parte das instituições que adquirem o software. Com o objetivo de garantir que os requisitos não funcionais e a política de qualidade não se tornem obsoletos, propomos uma metodologia para garantir a conformidade dos requisitos elicitados com a política de qualidade, mesmo quando novas tecnologias forem adotadas pela instituição ou novas funcionalidades forem adicionadas ao software em questão.

2.4 Qualidade de Software Médico

Ao se tratar de qualidade, deve-se levar em consideração a atividade final do software e o meio onde este está inserido. Esses são, sem dúvida, fatores que podem alterar o conceito de qualidade, ou seja, o que é válido como característica positiva em um software da área médica, pode não ser para outro domínio.

Segundo Rocha (2001), deve-se atender com alta qualidade os objetivos ao qual o aplicativo se destina e, para tanto, é necessário uma análise a fim de se obter reais objetivos do sistema, bem como quais os níveis de qualidade desejados. Este autor ressalta ainda que diferentes tipos de software médicos possuem características de qualidade específicas que

devem ser consideradas em sua avaliação. Contudo, ele define uma taxonomia de atributos obtidos de forma *top-down* a partir de atributos de mais alto nível até atributos primitivos e independentes, possíveis de serem avaliados. A taxonomia organiza os atributos de qualidade em três objetivos: (i) *utilizabilidade*, que considera as diferentes formas de sua utilização durante o desenvolvimento, manutenção e operação; (ii) *confiabilidade conceitual*, que considera características que garantem que um produto de software atende a seus requisitos, no que se refere ao seu conteúdo; e (iii) *confiabilidade da representação* que considera características relacionadas a necessidade do software estar representado de modo a facilitar o seu entendimento e manipulação pelos diferentes tipos de usuários. Neste contexto, características de qualidade para diferentes tipos de software da área médica foram identificadas. Estas características incorporam as características e sub-características definidas na ISO 9126.

2.4.1 Modelo de Qualidade para Sistemas de Informações Hospitalares

A definição de qualidade em Sistemas de Informações Hospitalares (SIH) deve considerar a facilidade de trabalhos gerenciais e as atividades de rotina. Fusão do software aos objetivos da instituição e automatização parcial ou total dos processos manuais existentes (ROCHA 2001).

Sistemas de Informações Hospitalares são sistemas que tratam de informações clínicas e administrativas referentes ao atendimento do paciente apoiando os profissionais responsáveis por este atendimento (ROCHA 2001). Estes sistemas apóiam a parte administrativa da instituição através de subsistemas de controle de pessoal, controle de material, controle orçamentário e apoiam a equipe médica auxiliando nas atividades de prescrição, história clínica, formulação de diagnóstico, realização de exames e pesquisas. A Figura 6 mostra os fatores e subfatores específicos para avaliação do SIH.

1. UTILIZABILIDADE*1.1. Eficiência*

1.1.1 Comportamento com relação ao tempo, 1.1.2 Comportamento com relação aos recursos, 1.1.3 Conformidade

1.2. Usabilidade

1.2.1 Inteligibilidade, 1.2.2 Apreensibilidade, 1.2.3 Operacionalidade, 1.2.4 Atratividade, 1.2.5 Conformidade, 1.2.6 Disponibilidade da informação, 1.2.7 Concorrência

1.3. Manutenibilidade

1.3.1 Analisabilidade, 1.3.2 Modificabilidade, 1.3.4 Estabilidade, 1.3.5 Testabilidade, 1.3.6 Conformidade

1.4. Portabilidade

1.4.1 Adaptabilidade, 1.4.2 Capacidade de ser instalado, 1.4.3 Coexistência, 1.4.4 Capacidade para substituir, 1.4.5 Conformidade

1.5. Reutilizabilidade

1.5.1 Modularidade, 1.5.2 Adaptabilidade

1.6. Implementabilidade

1.6.1 Viabilidade Econômica, 1.6.2 Viabilidade Financeira, 1.6.3 Viabilidade Tecnológica, 1.6.4 Viabilidade de Cronograma, 1.6.5 Viabilidade Social, 1.6.6 Viabilidade Legal

1.7. Rentabilidade

1.7.1 Lucratividade, 1.7.2 Competitividade, 1.7.3 Valor Comercial, 1.7.4 Valor de Marketing

1.8. Flexibilidade

1.8.1 Extensibilidade, 1.8.2 Adaptabilidade, 1.8.3 Independência de Dados

2. CONFIABILIDADE CONCEITUAL*2.1. Funcionalidade*

2.1.1 Adequação, 2.1.2 Acurácia, 2.1.3 Interoperabilidade, 2.1.4 Segurança, 2.1.5 Conformidade

2.2. Confiabilidade

2.2.1 Maturidade, 2.2.2 Tolerância a falhas, 2.2.3 Recuperabilidade, 2.2.4 Conformidade

2.3. Integridade

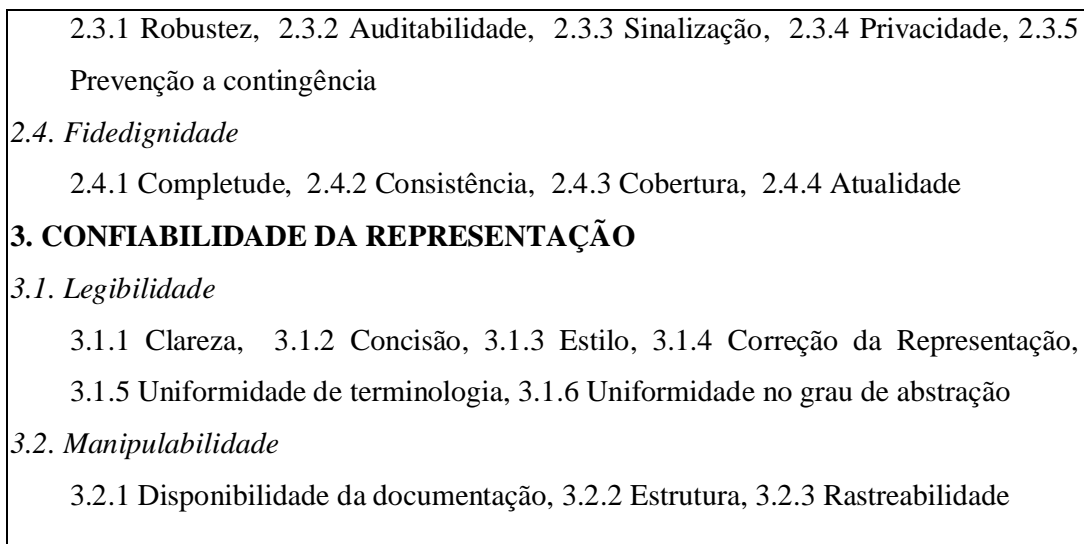
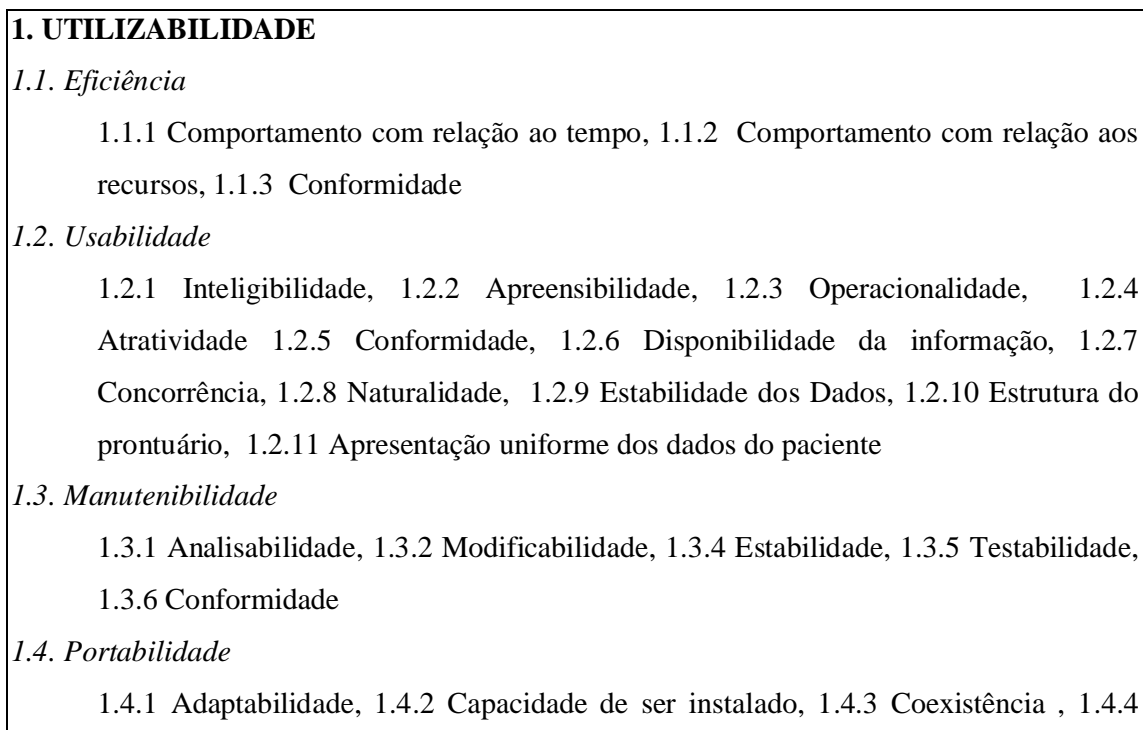


Figura 6 - Características de qualidade de SIH. (ROCHA 2001)

2.4.2 Modelo de Qualidade para Prontuário Médico

Um prontuário médico consiste de um conjunto de documentos pertencentes a um paciente, cuja principal finalidade é facilitar a assistência médica, documentando a história dos exames, diagnósticos e tratamento dados ao paciente. Estas informações são vitais para se garantir a continuidade da assistência médica durante a hospitalização do paciente. A Figura 7 mostra os fatores e subfatores específicos para avaliação destas aplicações (ROCHA 2001).



<p>Capacidade para substituir, 1.4.5 Conformidade</p> <p><i>1.5 Reutilizabilidade</i></p> <p>1.5.1 Modularidade, 1.5.2 Adaptabilidade</p> <p><i>1.6. Implementabilidade</i></p> <p>1.6.1 Viabilidade Econômica, 1.6.2 Viabilidade Financeira, 1.6.3 Viabilidade Tecnológica, 1.6.4 Viabilidade de Cronograma, 1.6.5 Viabilidade Social, 1.6.6 Viabilidade Legal</p> <p><i>1.7. Rentabilidade</i></p> <p>1.7.1 Lucratividade, 1.7.2 Competitividade, 1.7.3 Valor Comercial, 1.7.4 Valor de Marketing</p> <p><i>1.8. Flexibilidade</i></p> <p>1.8.1 Extensibilidade, 1.8.2 Adaptabilidade, 1.8.3 Independência de Dados</p> <p>2. CONFIABILIDADE CONCEITUAL</p> <p><i>2.1. Funcionalidade</i></p> <p>2.1.1 Adequação, 2.1.2 Acurácia, 2.1.3 Interoperabilidade, 2.1.4 Segurança, 2.1.5 Conformidade</p> <p><i>2.2. Confiabilidade</i></p> <p>2.2.1 Maturidade, 2.2.2 Tolerância a falhas, 2.2.3 Recuperabilidade, 2.2.4 Conformidade</p> <p><i>2.3. Integridade</i></p> <p>2.3.1 Robustez, 2.3.2 Auditabilidade, 2.3.3 Sinalização, 2.3.4 Privacidade, 2.3.5 Prevenção a contingência, 2.3.6 Imutabilidade, 2.3.7 Permanência dos dados, 2.3.8 Atributabilidade</p> <p><i>2.4. Fidedignidade</i></p> <p>2.4.1 Completude, 2.4.2 Consistência, 2.4.3 Cobertura, 2.4.4 Atualidade</p> <p>3. CONFIABILIDADE DA REPRESENTAÇÃO</p> <p><i>3.1. Legibilidade</i></p> <p>3.1.1 Clareza, 3.1.2 Concisão, 3.1.3 Estilo, 3.1.4 Correção da Representação, 3.1.5 Uniformidade de terminologia, 3.1.6 Uniformidade no grau de abstração</p> <p><i>3.2. Manipulabilidade</i></p> <p>3.2.1 Disponibilidade da documentação, 3.2.2 Estrutura, 3.2.3 Rastreabilidade</p>

Figura 7 - Modelo de Qualidade do Prontuário Eletrônico do Paciente (ROCHA 2001)

Na seção 3.2, propomos um modelo de qualidade que reúne quatro necessidades de qualidade para *software* médico e um conjunto de requisitos para satisfazer estas necessidades. Este modelo servirá de base para elicitación dos demais requisitos de qualidade para o *software*.

2.5 Padrões em Software Médico

2.5.1 Problemas causados pela ausência de padrões

As instituições de saúde ao desenvolverem um sistema de informação, devem resolver questões como: que protocolo de comunicação utilizar; os bancos de dados serão distribuídos ou centralizados; que padrão utilizar para o armazenamento e transmissão de imagens; a resistência à informatização por partes dos profissionais da área da saúde quanto ao uso de prontuário eletrônico, dentre outros problemas.

Segundo Pisa (2004), o aumento na heterogeneidade dos equipamentos e da infraestrutura de software; a dificuldade de prover acesso unificado aos dados, a custosa dependência de soluções proprietárias para o armazenamento de dados e as rápidas mudanças de demandas clínicas que exigem adaptações nas capacidades dos servidores de dados e soluções para o desenho do sistema. Este autor ressalta ainda que a complexidade dos sistemas de informação em saúde requer forte padronização para seus usuários. Segundo ele, a adoção de padrões de representação, tecnológicos e metodológicos permite que um sistema computacional da área possa se beneficiar de soluções individuais, comerciais e acadêmicas para implementar sua atualização para o gerenciamento do enorme tráfego de dados, da quantidade crescente de pontos de atendimento eletrônicos e da complexidade da informação.

Já Alexandrini (2005) destaca que o uso da padronização no armazenamento, na nomenclatura e no vocabulário médico além de facilitar o uso do sistema, é fundamental para reunir informação clínica no cuidado ao paciente, para recuperar informação no manejo da doença ou para pesquisa, assim como para conduzir a análise de resultados.

Por outro lado, ao longo dos anos muitas instituições brasileiras vêm desenvolvendo sistemas de Registros Eletrônicos em Saúde sem o uso de padrões adequado para este fim. A ausência destes padrões impede o compartilhamento de seus registros com outras instituições e algumas vezes até mesmo internamente (CAMARGO 2002). Para que esse compartilhamento de informações possa acontecer com eficiência, a utilização de tais padrões é imprescindível.

Diante deste cenário, fica evidente que os problemas se agravam a medida que aumenta o número de instituições que implementam sistemas de informações em saúde sem o uso de padrões. Acreditamos que a adoção de padrões por parte destas instituições pode resolver muito dos problemas citados acima. Por este motivo, entendemos que a padronização é um requisito essencial para qualidade de software médico. A seguir abordaremos alguns padrões utilizados em aplicações médicas.

2.5.2 Padrões utilizados em aplicações médicas

De acordo com Leão (2004), os vocabulários na área da saúde são complexos, não estruturados e repletos de diversidades como: variações regionais e especificidades do contexto em saúde. Ainda que os profissionais possam utilizar os vocabulários de sua preferência, é necessário controlar esta utilização para garantir a troca de informações e avaliação em saúde. Segundo este autor, as organizações devem estar preparadas para as mudanças comportamentais e rotineiras que podem surgir com a introdução de um sistema informatizado. Os métodos de pesquisa como o SQL (*Structured Query Language*), que é uma linguagem comum para manipulação de bancos de dados relacionais, não são facilmente empregados para manipular textos livres (CHU 2002).

Segundo O. Jaime (1993), a padronização da comunicação entre os diferentes sistemas computadorizados e da forma de interação com os profissionais de saúde é necessária, mas não é suficiente para assegurar a utilização do registro clínico informatizado na rotina diária. Os padrões de intercâmbio de informações especializadas (por exemplo, entre laboratórios) são de grande ajuda quanto a codificação dos diversos tipos de exames e como seus resultados são comunicados. Entretanto, eles são fundamentais quando se pretende informatizar coisas mais complexas, tais como a história clínica, exame físico, prescrição, evolução, prognóstico, etc.

Para Silva (2003), a padronização do vocabulário médico é fundamental para reunir informação clínica no cuidado ao paciente, para recuperar informação no manejo da doença ou para pesquisa, assim como para conduzir a análise de resultados. As primeiras iniciativas neste sentido são bastante antigas e correspondem aos métodos de classificação e codificação de informações médicas, tais como a CID (Classificação Internacional de Doenças), SNOMED (*Standard Nomenclature of Medicine*), etc... Vários são os trabalhos neste campo, tendo sido inclusive criado um grupo especial na IMIA (*International Medical Informatics Association*) para tratar do assunto.

Uma das grandes iniciativas, em termos de modernização dos métodos de acesso às informações médicas é o projeto gerenciado pela UMLS (*Unified Medical Language System*). Este projeto veio facilitar o acesso ao grande número de informações médicas, inclui informações de literatura, bancos de dados, registros clínicos, sistemas baseados em conhecimento, etc.

Um dos grandes esforços de desenvolvimento de padrões de intercâmbio de informações é o chamado HL7 (*Health Level Seven*), cujo nome se refere ao nível 7 do padrão ISO/OSI para redes de computadores (MASSAD 2003). Tem por objetivo definir padrões para a solicitação de exames, geração dos resultados, censo hospitalar e faturamento. Abrangem padrões para a produção de laudos de diversos procedimentos diagnósticos (ECG, EEG, Ultra-som, etc.).

Outro padrão com objetivos semelhantes ao HL7 é o MEDIX P1157, desenvolvido pelo IEEE (*Institute of Electrical and Electronic Engineers*), o qual pretende ser totalmente compatível com as especificações de redes do padrão ISO. Estes dois padrões prometem convergir em termos da estrutura das mensagens utilizadas.

Com relação ao arquivamento e transmissão de imagens médicas, um padrão já bastante difundido é o ACR-NEMA (MASSAD 2003), desenvolvido pelo ACR (*American College of Radiology*) em colaboração com a NEMA (*National Electrical Manufacturers Association*), e que vem sendo adotado por diversos fabricantes e usuários de sistemas de PACS (*Picture Archiving and Communication Systems*) hospitalares.

A grande aceitação do padrão ACR-NEMA, por parte dos fornecedores e dos usuários, se deve à possibilidade de integração de diferentes equipamentos em uma rede mais ampla da clínica ou do hospital. A partir de 1993 com o lançamento do DICOM3 (*Digital Imaging and Communications in Medicine*) foi possível implementar sistemas PACS e Prontuários Eletrônicos com equipamentos heterogêneos respeitando os sistemas legados das instituições. Esta especificação e a adoção ampla da mesma permitiu que as instituições pudessem ter acesso completo às imagens de seus pacientes, que até então eram armazenadas em formato proprietário. Atualmente, todos os grandes fabricantes de equipamentos médicos o adotam, pois essa característica tornou-se uma exigência básica das instituições (MASSAD 2003).

Além das imagens tradicionais, de duas ou três dimensões, sinais biológicos (unidimensionais) como ECG, EEG, PA (Pressão Arterial), Temperatura e outros, também podem ser tratados de forma análoga para fins de armazenamento e transmissão. O DICOM

estabeleceu o padrão para armazenamento de sinais, abrindo com isto a possibilidade de se incluir estes elementos aos PACS.

Já na área de instrumentação biomédica, um dos principais padrões é o MIB (*Medical Information Bus*), desenvolvido pelo IEEE, com o objetivo de padronizar a comunicação entre os instrumentos utilizados em UTIs (MASSAD 2003). Ele permite a interconexão de equipamentos com o computador central, através de um controlador central de comunicação, que se responsabiliza por solicitar as informações de cada um dos instrumentos.

Atualmente, os laboratórios têm usado o sistema ASTM 1238 ou seu similar HL7, para mandar eletronicamente resultados a outros laboratórios e à hospitais. Porém, como nem todos usam este sistema, há uma incapacidade de comunicação; isto só seria possível se todos os laboratórios e hospitais se adaptassem ao código, muitas vezes próprio, dos laboratórios que enviam estas informações, o que é impossível, uma vez que se recebe dados de vários laboratórios. Devido a isso, o LOINC (*Logical Observation Identifier, Names and Codes*) surge com o objetivo de uniformizar estas informações clínicas e laboratoriais facilitando a troca e análise de resultados (MASSAD 2003). Trata-se de uma base de dados composta por nomes e códigos que identificam observações clínicas e laboratoriais. Se este sistema fosse usado pelos produtores de informação médica como sistema “universal” de comunicação para dados laboratoriais, estes problemas cessariam. Diferentemente dos outros sistemas de código, o LOINC não requer taxa de licença para o seu uso.

Como se vê, ainda há muito o que fazer nesta área. Considerando especificamente a situação brasileira, certamente não seria recomendável desenvolver padrões nacionais, pois a maioria dos instrumentos utilizados em laboratórios, UTIs e serviços radiológicos são importados e, mais cedo ou mais tarde, seguirão estes padrões. O que sugerimos é verificar as peculiaridades nacionais, de modo a propor a complementação necessária, sempre com o devido mapeamento, sobre o padrão genérico. Neste sentido, seria necessário produzir uma camada para efetuar a tradução para o português.

Neste capítulo, abordamos os principais conceitos relacionados a engenharia de requisitos, destacamos algumas abordagens sobre a qualidade de software finalizando com a abordagem sobre a qualidade de software médicos enfatizando o uso de padrões como uma das principais características de qualidade para este tipo de software. Os conceitos abordados servem de base para um melhor entendimento dos trabalhos relacionados a área de engenharia de requisitos. As abordagens sobre qualidade de software serviram de reforço para nossa proposta quando destacaram a necessidade de se tratar os RNF logo nas primeiras fases do

processo de desenvolvimento. Finalmente as abordagens sobre padrões médicos mostraram a necessidade de se adotar padrões quando se desenvolvem software para área da saúde.

3 SOLUÇÃO PROPOSTA

3.1 Modelo de qualidade proposto para Software Médico

Na Figura 8, propomos um modelo genérico para qualidade de software médico a partir do qual os requisitos de qualidade podem ser decompostos. Este modelo foi originado a partir das necessidades de qualidade de software descritas por (LOUCOPOULOS 1995).

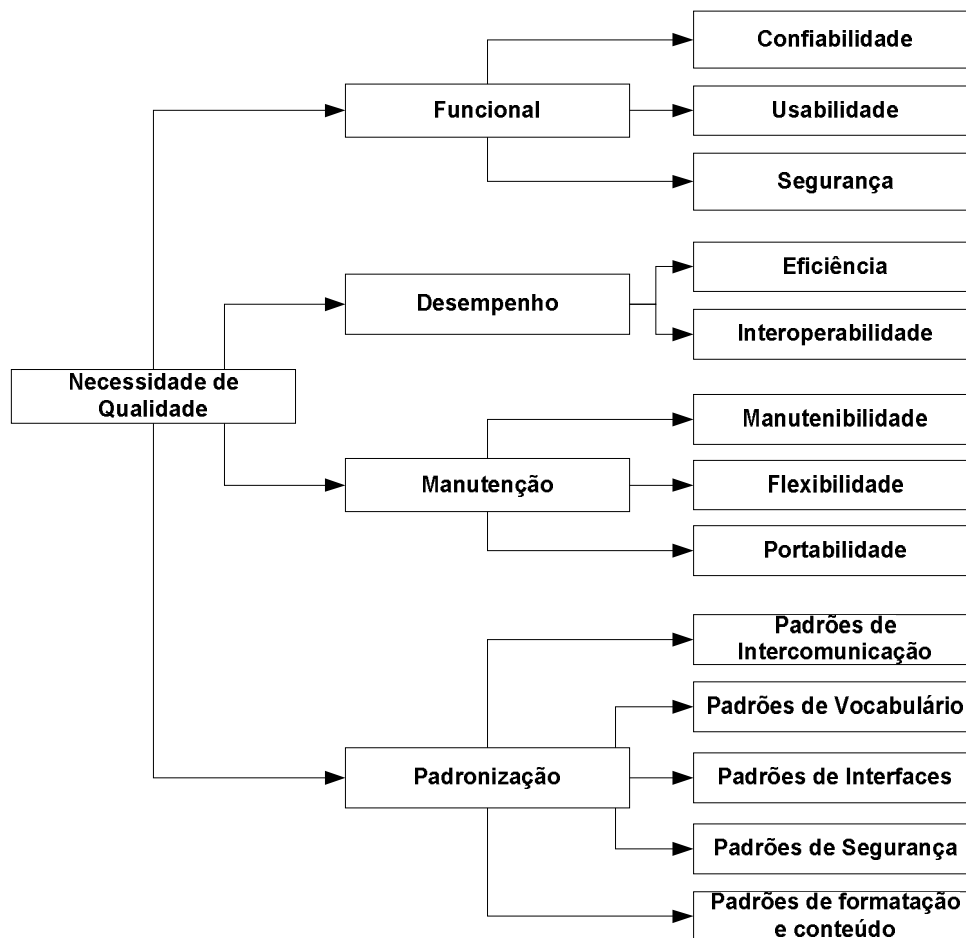


Figura 8 - Modelo Proposto para Qualidade de Software Médico

O modelo proposto na Figura 8, reúne quatro necessidades de qualidade para software médico e um conjunto de requisitos não funcionais para satisfazer a qualidade desejada. A primeira necessidade foi a *Qualidade Funcional* que considera RNFs para garantir que o software atenda a seus requisitos, no que se refere a sua segurança, confiabilidade e usabilidade; em seguida a necessidade da *Qualidade de Desempenho*, considera RNFs para garantir que o software atenda seus requisitos, no que se refere a

quantidade de recursos e código requerido para realizar suas funções; a terceira necessidade foi da *Qualidade Manutenção*, considera RNFs relacionados à necessidade de evolução e correção do software, de modo que facilite a correção de erros e adaptação a novas mudanças; e finalmente a *Qualidade Padronização*, considera RNFs para facilitar a integração do software com os diferentes sistemas de informação já existentes, plataformas computacionais, sistemas operacionais, linguagens de programação e redes de comunicações sem a qual é impossível a sua integração efetiva.

3.2 Classificação dos requisitos não funcionais proposta

Para melhor entendimento da nossa proposta, classificamos os requisitos não funcionais em Primários, Secundários e Específicos. RNFs Primários são metas de alto nível de abstração que devem ser satisfeitas e portanto precisam ser decompostas em submetas denominadas de RNF Secundários até que se encontre as operacionalizações chamadas de RNF Específicos que são as ações necessárias para satisfazer a meta principal. Os RNF Primários representam os requisitos necessários para satisfazer as necessidades de qualidade do software definidas no modelo de qualidade proposto na Figura 8. Os RNF Secundários são decomposições de um RNF Primário ou de outro RNF Secundário, portanto possuem um nível de abstração menor. Os RNFs específicos também são decompostos de um RNF Primário ou Secundário. Diferem dos RNFs Secundários por não sofrerem decomposições e possuem o menor nível de abstração possível. A Figura 9 mostra as possíveis decomposições destes três tipos de requisitos.

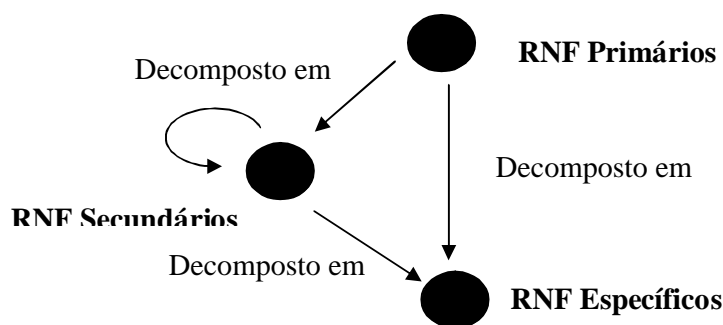


Figura 9 – Possíveis Decomposições para os Requisitos Não Funcionais

A seguir a Figura 10 mostra a classificação proposta pela nossa metodologia para os requisitos não funcionais.

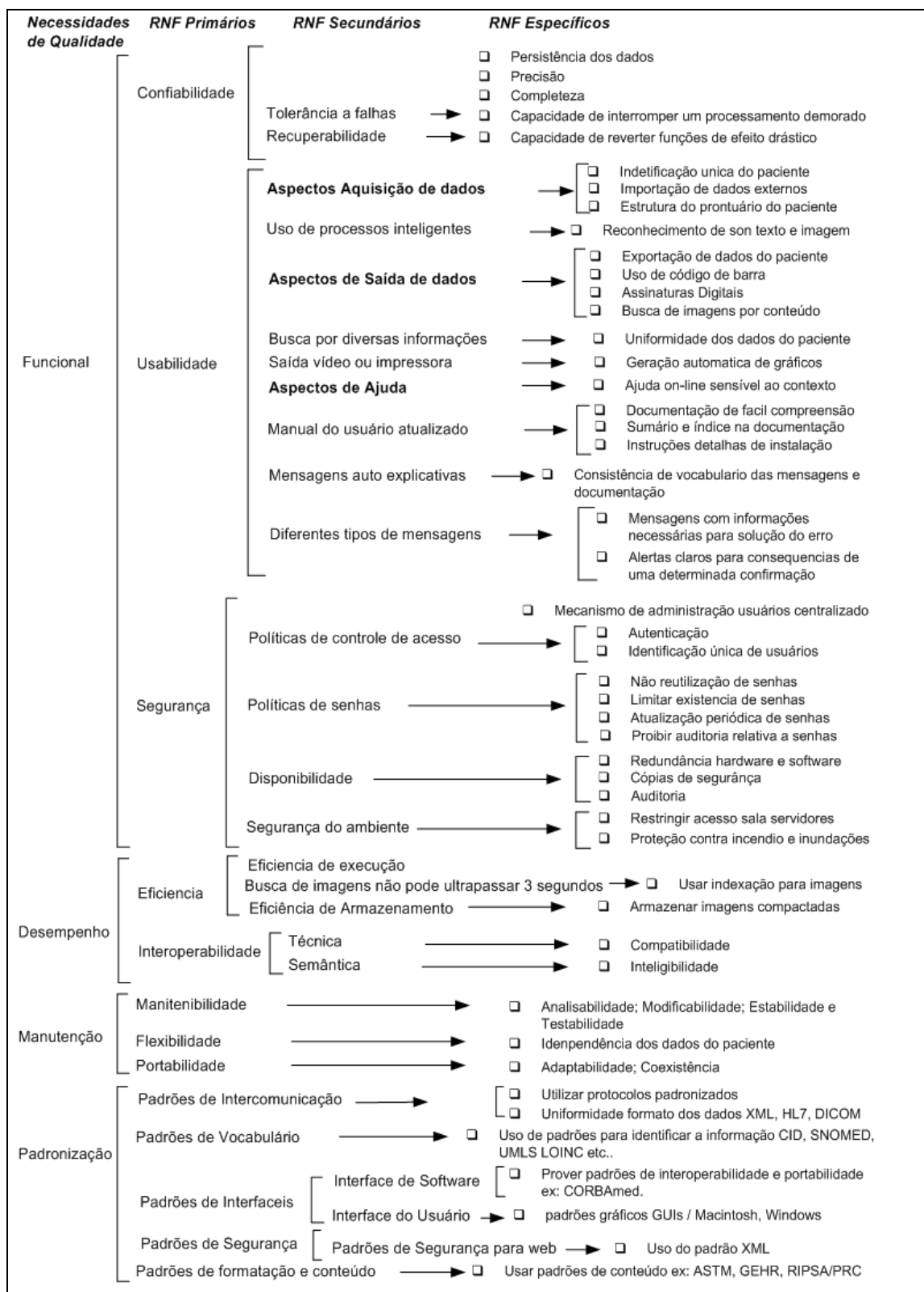


Figura 10 – Classificação dos RNFs Proposta

3.3 Modelo do documento de requisitos proposto

O modelo para o documento de especificação de requisitos proposto na Figura 11, tem o objetivo de fornecer um guia para especificar os requisitos elicitados, o qual poderá ser útil tanto para as equipes de desenvolvimento, na tentativa de facilitar a construção de sistemas, quanto para as equipes de análises, nas futuras manutenções corretivas ou evolutivas.

<p>Cabeçalho</p> <p><Título do documento></p> <p><Autor></p> <p><Endereço></p> <p><Data da versão do documento></p> <p><Número da versão do documento></p> <p>1. Introdução</p> <p>1.1 Objetivo do documento</p> <p>1.2 Escopo do produto</p> <p>1.3 Visão geral do documento</p> <p>2. Descrição Geral do Produto</p> <p>2.1. Funções do produto</p> <p>2.2. Restrições gerais</p> <p>3. Características Gerais</p> <p>3.1. Documentação dos <i>stakeholders</i></p> <p>3.2. Documentação dos objetivos</p> <p>3.3. Documentação das relações de dependências entre os objetivos</p> <p>4. Requisitos</p> <p>4.1. Especificação dos requisitos funcionais</p> <p>4.2. Documentação da Política de Qualidade</p> <p>4.3. Documentação das Planilhas dos Requisitos não funcionais</p> <p>5. Referências</p> <p>6. Apêndice</p> <p>7.1. Glossário</p> <p>7.2. Outras informações</p>
--

Figura 11 - Modelo do documento de requisitos proposto

A parte inicial é composta de um “Cabeçalho”, contendo o Título do Documento, o nome do Autor ou Autores, Endereço, telefone, e-mail, Data de Produção da Versão do documento de especificação, Número da Versão e, se esse documento for revisão de uma variante anteriormente produzida, a Data da Especificação Anterior a esta.

A seção seguinte é a “Introdução”, a qual está dividida em três subseções: Objetivo do Documento, onde será descrito seu propósito; Escopo do Produto, que estabelece uma visão sintética do objetivo do produto que será especificado e também o público a que se destina; e uma Visão Geral do Documento, com a finalidade de prover uma breve explanação do conteúdo do restante do documento, indicando sua estrutura básica. Caso alguma seção ou subseção prevista na proposta seja omitida ou alterada, a omissão ou alteração deverá ser justificada nesse ponto.

Em seguida, a seção “Descrição Geral do Produto” aborda as funções que o sistema visa atender, além de estabelecer os aspectos técnicos que possam limitar o desenvolvimento do produto. Para isso, essa seção está subdividida em dois itens: Funções do Produto e Restrições Gerais.

O item Funções do Produto contém as principais funções que ele desempenhará, assim como uma descrição sintética do objetivo de cada uma. Para efeito de organização, estas funções poderão ser representadas em forma de tabela, cujo conteúdo da tabela será formado por um identificador da função, o nome da função e uma pequena descrição.

Em seguida, o item referente às Restrições Gerais irá expor tanto os aspectos técnicos quanto gerenciais, os quais possam limitar as opções dos desenvolvedores, como por exemplo: restrições legais, de hardware, de linguagens de programação, de desempenho, de confiabilidade, etc.

A seção seguinte trata das “Características Gerais”, que, para efeito de organização, foi dividida em três subseções: Documentação dos *stakeholders*, que conterà a transcrição de todos os *stakeholders* identificados no processo de desenvolvimento; Documentação dos Objetivos, onde serão colocados todos os objetivos elicitados durante a elicitação dos requisitos funcionais e Documentação das Relações de Dependência entre os Objetivos, cuja finalidade será listar todas as relações de dependência dos objetivos identificadas no processo de desenvolvimento.

A seção quatro, “Requisitos”, aborda sobre os requisitos propriamente ditos. Dessa forma, a documentação dessa seção foi dividida em dois itens: especificação dos requisitos funcionais, cuja finalidade é especificar o detalhamento dos cenários e suas ações assim como os seus relacionamentos com os objetivos. O item Documentação da Política de Qualidade

tem a finalidade de especificar a política de qualidade de acordo com o conjunto de necessidade de qualidade definida na primeira fase do método proposto. Já a subseção Documentação das Planilhas dos Requisitos não funcionais tem a finalidade de especificar o detalhamento dos requisitos não funcionais primários, secundários e específicos.

As informações necessárias para que todas as fontes de dados citadas no documento de especificação de requisitos, inclusive aquelas obtidas através de atas de reuniões e memorandos, possam ser recuperadas, se necessário, devem estar descritas na seção “Referências”.

Para finalizar o documento, a seção “Apêndice” apresenta um glossário de termos usado durante a captura de requisitos, para facilitar a comunicação entre analistas e *stakeholders*, pois serve para definições de siglas, palavras e abreviações de termos usados; e outras informações.

Como esse documento é sugerido para especificar requisitos funcionais e não funcionais, seu preenchimento poderá ser completado de forma iterativa e incremental, a partir do início da execução do método, podendo ocorrer paralelamente a elicitação dos requisitos não funcionais.

3.4 O Método Proposto

O método proposto possui seis fases assim denominadas: a *fase de elicitação* que define o conjunto de requisitos não funcionais primários para satisfazer cada necessidade de qualidade do modelo de qualidade proposto; a fase de decomposição e representação responsável pelas ações de decompor e representar os requisitos elicitados; a *fase de especificação* que descreve de forma detalhada os requisitos primários, secundários e específicos para que não fique dúvidas quanto ao seu entendimento; a *fase de identificação das influências* para identificar as influências positivas e negativas que os requisitos não funcionais exercem uns sobre os outros; a *fase de criação da política de qualidade* cujo objetivo é criar a política de qualidade para o software que será produzido e, finalmente, a *fase de avaliação das conformidades* para calcular a conformidade da definição dos requisitos elicitados e medir a obediência com a política existente. A Figura 12 mostra o método proposto com a seqüência de execução de suas fases:

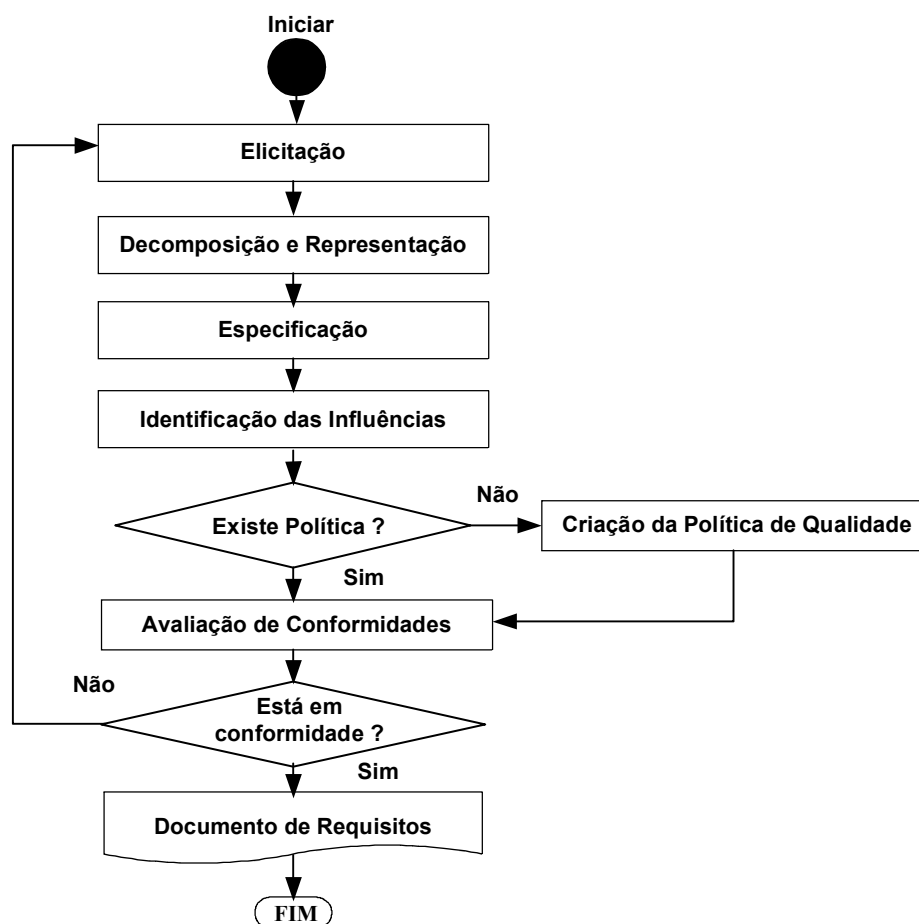


Figura 12 – Método Proposto

3.4.1 Fase de Elicitação

O objetivo desta fase é definir as *Necessidades de Qualidade* para o domínio da aplicação e o conjunto de RNFs que devem atender a cada uma dessas necessidades. Algumas necessidades, por exemplo: funcionalidade, manutenção etc, são comuns a muitos tipos de aplicações, porém existem outras que são mais relevantes a determinadas aplicações e requer um certo conhecimento do engenheiro de requisitos no domínio em questão para identificá-las como necessidade. Contudo, quanto maior este conhecimento, mais facilmente ele irá identificar as necessidades de qualidade e os RNFs Primários para satisfazer estas necessidades. Para que não fiquemos apenas na dependência do conhecimento do engenheiro de requisitos no domínio da aplicação, propomos um modelo para a qualidade de software médico na Figura 8, a partir do qual os requisitos não funcionais podem ser decompostos.

Nesta fase, o Engenheiro de Requisito juntamente com os usuários do sistema, devem analisar se o modelo de qualidade proposto na Figura 8 atende as necessidades de qualidades da aplicação que será construída. Se o modelo de qualidade não atender as necessidades de qualidade para o software, o Engenheiro de Requisitos junto com os usuários podem acrescentar novas necessidades de qualidades, novos requisitos, omitir necessidades ou requisitos contidos no modelo ou até mesmo substituir todas as necessidades e requisitos de modo que o modelo atenda a qualidade necessária para o software.

Estando o modelo de qualidade definido e partindo do princípio de que existe um conjunto de cenários previamente definidos durante a elicitação dos requisitos funcionais, o método propõe que o engenheiro de requisito faça uma análise em cada cenário e descubra quais RNFs podem ser aplicados ao cenário em estudo. Esta análise é importante, para associar os RNFs aos cenários da aplicação. Isso possibilita a descoberta de novos RNFs ou até mesmo novas necessidades de qualidade para o software. A título de documentação, o método sugere que os RNFs elicitados sejam documentados na LRNFE (Lista de Requisitos não Funcionais Elicitados) com a seguinte estrutura (Necessidade de Qualidade, Requisito não funcional, cenários).

3.4.2 Fase de Decomposição e Representação

Os RNFs, inicialmente elicitados, são metas de alto nível de abstração que devem ser satisfeitas, e portanto precisam ser decompostas em submetas chamadas de *RNFs secundários*, até que se encontre as operacionalizações (*RNFs específicos*) que correspondem as ações, necessárias para satisfazer a meta principal. Nesta fase, cada RNF deve ser decomposto até que se possa visualizar o que é necessário para satisfazê-lo, ou seja, suas operacionalizações. A decomposição proposta consiste em um processo manual, onde o engenheiro de requisito deve analisar cada RNF elicitado, verificando se ele pode ser decomposto em outras submetas menos abstratas. Durante esta fase, decomposições sucessivas podem ocorrer e uma regra que determine quando a decomposição deve parar não existe. A identificação de possíveis operacionalizações (*RNFs específicos*) descritas em cenários, pode servir como uma ajuda num processo que alterne abordagens *bottom-up* e *top-down* de decomposição destes Requisitos. Por exemplo, o *RNF Confiabilidade* pode ser decomposto em: *RNF Persistência dos dados*, *RNF Precisão*, *RNF Completas e RNF Tolerância a falhas*. O *RNF Tolerância a falhas* também pode ser decomposto no *RNF Recuperabilidade* em caso de falha.

Durante as decomposições, é importante utilizar o conhecimento próprio do domínio e sempre que possível envolver o cliente na definição destas decomposições, avaliando em conjunto com este se as decomposições estão corretas e são suficientes.

Após a decomposição, os RNFs precisam ser representados de alguma forma. Esta representação deve possibilitar que lidemos com eles de maneira organizada e que facilite sua manipulação e avaliação. Uma forma de representá-los são os grafos propostos por Chung (1999) (1995). Outra estratégia que também pode ser usada para representá-los, é através de uma árvore que lista todos os RNFs Primários, Secundários e Específicos, capazes de modelar a qualidade do software segundo as necessidades do usuário. Este trabalho utiliza a árvore de RNFs para representá-los, pelo fato desta representação facilitar a execução da fase de avaliação empregada pela metodologia. A seguir a Figura 13 mostra um exemplo da representação da árvore de decomposição do RNF Confiabilidade.

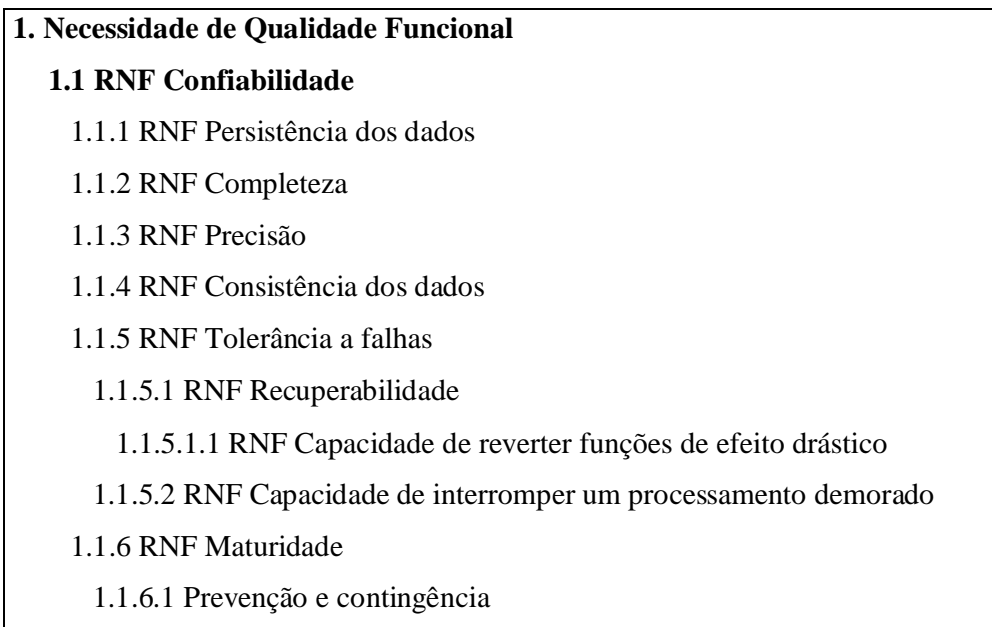


Figura 13 - Árvore do RNF Confiabilidade e suas decomposições

3.4.3 Fase de Especificação

Nessa fase, todos os RNFs elicitados, devem ser especificados de forma descritiva, a fim de não deixar dúvidas quanto aos seus significados. Utilizamos as planilhas de características, subcaracterísticas e atributos propostas por Reis (2004), para especificar respectivamente os RNFs Primário, Secundário e Específico. As planilhas foram adaptadas com algumas informações para atender a metodologia proposta. Por exemplo: a planilha de

característica deu origem a planilha de *RNF Primário* acrescida das informações de *Necessidade, Cenários, Categoria, Influências positivas, Influências negativas, conformidade desejada*. A Figura 14 mostra o modelo da planilha de *RNF Primários* com todas as suas informações:

Tipo: RNF Primário
Necessidade: Título da Necessidade de Qualidade
Código: Código do RNF Título: Título do RNF
Categoria: estabelecida na política de qualidade
Cenários: Código dos cenários que o RNF pode ser aplicado
Quantidade de Filhos:
Definição:
Decomposições: encontrados na fase de decomposição dos RNF
Influências Positivas (descobertas na fase de identificação das Influências)
Influências Negativas (descobertas na fase de identificação das Influências)
Conformidade Desejada: (especificada na política de qualidade)
Conformidade Obtida: (calculada pela conformidade dos RNFs Secundários)

Figura 14 – Modelo da Planilha de informações dos RNFs Primários

A planilha de subcaracterísticas deu origem à planilha de *RNFs Secundários* com adaptações das informações de *Influências positivas e Influências negativas*. A Figura 15 mostra o modelo da planilha de *RNF Secundários* com todas as Informações:

Tipo: RNF Secundário
Código: Código do RNF Título: Título do RNF
Quantidade de Filhos: encontrada na fase de decomposição dos RNFs.
Definição:
Decomposto de: encontrados na fase de decomposição dos RNFs
Decomposições: encontrados na fase de decomposição dos RNFs
Influências Positivas (atribuída na fase de identificação das Influências)
Influências Negativas (atribuída na fase de identificação das Influências)
Peso: atribuído na fase de avaliação de conformidades
Nota: (calculada através das conformidades de suas decomposições)

Figura 15 – Modelo da Planilha de informações dos RNFs Secundários

E finalmente, a planilha de atributos deu origem a planilha de *RNFs Específicos* acrescida das *Influências positivas e Influências negativas*. A Figura 16 mostra o modelo das planilhas de RNF Específicos.

Tipo: RNF Específico	
Código: Código do RNF	Título: Título do RNF
Definição:	
Decomposto de: encontrados na fase de decomposição dos RNFs	
Influências Positivas (atribuída na fase de identificação das Influências)	
Influências Negativas (atribuída na fase de identificação das Influências)	
Peso: atribuído na fase de avaliação de conformidades	
Nota (atribuída pelo engenheiro de requisito através de análise do documento de requisitos)	

Figura 16 – Modelo da Planilha de informações dos RNFs Específicos

3.4.4 Fase de Identificação das Influências

Esta fase consiste em identificar as Influências positivas e negativas entre os requisitos não funcionais (Primários, Secundários e Específicos), de modo que possibilite rastrear os impactos de um no outro. Segundo Cysneiros (2002), uma fonte de identificação de Influências reside no conhecimento do domínio que o engenheiro de software tem. Quanto maior este conhecimento mais facilmente ele irá identificar possíveis Influências com outros RNFs. Nesta atividade, o método utiliza a heurística de comparação por pares proposta por Cysneiros (2002), com a finalidade de garantir que todos os RNFs serão avaliados quanto a influência de um no outro. As influências negativas representam conflitos entre os RNFs. Como estamos tratando da qualidade da especificação dos RNFs ainda na fase de requisitos do processo de desenvolvimento e não do produto já pronto, não iremos tratar a questão dos conflitos desses requisitos já que dois RNFs com Influências negativas podem perfeitamente estar muito bem especificados no documento de requisitos. Estas influências somente serão observadas na fase de teste do processo de desenvolvimento. Neste caso, optamos por não resolver estes conflitos e apenas documentá-los para um tratamento futuro.

Como forma de documentar as influências identificadas nesta fase, inserimos os itens influências negativas e influências positivas nas planilhas dos requisitos não funcionais.

3.4.5 Fase de Criação da política de qualidade

O objetivo desta fase é criar uma política de qualidade para o software que será produzido. A política deve expressar a qualidade e a conformidade mínima pretendida pelos *stakeholders*. Para criar a política propomos os seguintes passos:

- ✓ ***Categorizar os RNFs Primários***: a finalidade desta atividade é categorizar os *RNFs Primários* de acordo com sua relevância para atingir a qualidade desejada e definir a conformidade mínima desejável para categoria de requisitos. O objetivo da categorização é facilitar a atribuição da conformidade para os *RNFs Primários* definidos na política. Propomos neste trabalho as seguintes categorias: *Especificação Forte*, *Especificação Média* e *Especificação Fraca*. Os RNFs devem ser categorizados de maneira que quanto maior for sua relevância para atingir a qualidade desejada, mais forte deve ser sua especificação, ou seja, melhor especificado este requisito deve estar no DRs. A atribuição da conformidade deve ser realizada pelo engenheiro de requisito juntamente com o usuário. Eles devem atribuir valores percentuais de conformidade para cada categoria de requisitos. O percentual de conformidade atribuído a cada categoria representa a conformidade mínima aceitável para os RNFs da categoria. É importante ressaltar que o valor da conformidade atribuída aos *RNFs Primários* da política, pode variar de uma aplicação para outra;
- ✓ ***Representar a política de qualidade***: esta atividade tem o objetivo de representar a política de qualidade. O método utiliza o modelo das políticas proposto por Rocha (2004), introduzindo o item “Descrição das Conformidades”. A primeira parte do modelo é formada por um “Cabeçalho”, o qual é composto pelo título do documento, nome(s) do(s) autor(es), data e número da versão da política de qualidade. A seção seguinte refere-se à “Introdução”, que é constituída pela declaração de propósito, onde será feita uma breve descrição do porquê da política de qualidade ser necessária e importante para a aplicação e uma “declaração das conformidades”, introduzida por esta abordagem, onde serão especificadas as conformidades mínimas aceitáveis para cada categoria de RNF. Em seguida, na seção “Políticas”, as necessidades de qualidade definidas no modelo de qualidade da Figura 8, são mapeadas como políticas de qualidade representadas por seus *RNFs Primários*. A seção final, “Apêndice”, é constituída de duas subseções: glossário, que é destinado à colocação de qualquer definição ou explicação que poderá ajudar na leitura e no completo entendimento da

política; e documentos relacionados, onde deverão ser listadas todas as informações relevantes que poderão ser úteis para complementar a política criada. O modelo da política de qualidade está ilustrado na Figura 17.

<p>Header <Título> <Autor (es)> <Data da Versão > < Numero da Versão ></p>
<p>1. Introdução</p>
<p>1.1. Finalidade e Importância</p>
<p>1.2. Descrição das Conformidades</p>
<p>1.2.1 <i>Especificação Forte</i></p>
<p>1.2.2 <i>Especificação Médica</i></p>
<p>1.2.3 <i>Especificação Fraca</i></p>
<p>2. Políticas</p>
<p>2.1. Política da Qualidade Funcional</p>
<p>2.1.1 <i>RNF Confiabilidade</i></p>
<p>2.1.2 <i>RNF Usabilidade</i></p>
<p>2.1.3 <i>RNF Segurança</i></p>
<p>2.2. Política da Qualidade Desempenho</p>
<p>2.2.1 <i>RNF Eficiência</i></p>
<p>2.2.2 <i>RNF Interoperabilidade</i></p>
<p>2.3 Política da Qualidade Manutenção</p>
<p>2.3.1 <i>RNF Manutenibilidade</i></p>
<p>2.3.2 <i>RNF Flexibilidade</i></p>
<p>2.3.2 <i>RNF Portabilidade</i></p>
<p>2.4 Política da Qualidade Padronização</p>
<p>2.4.1 <i>RNF Padrões de Intercomunicação</i></p>
<p>2.4.2 <i>RNF Padrões de Vocabulário</i></p>
<p>2.4.3 <i>RNF Padrões de Interfaces</i></p>
<p>2.4.4 <i>RNF Padrões de Segurança</i></p>
<p>2.4.5 <i>RNF Padrões de formatação e conteúdo</i></p>
<p>3. Apêndice</p>
<p>3.1. <i>Glossário</i></p>
<p>3.2. <i>Documentos Relacionados.</i></p>

Figura 17 - Modelo da Política de Qualidade

3.4.6 Fase de Avaliação de Conformidades

Esta fase é importante para medir a obediência entre a especificação dos requisitos de qualidade e a política existente. O uso da árvore de RNF definida na *Fase de Decomposição e Representação dos requisitos não funcionais* é indispensável na execução das próximas atividades:

- ✓ **Associar Pesos aos RNFs Secundários e Específicos.** Nesta atividade, os *RNFs Secundários* e *Específicos* são priorizados de acordo com sua importância no contexto do domínio da aplicação. Deve-se associar a cada *RNF Secundário* e *Específico* um peso entre 0 (zero) e 10 (dez). Os pesos são atribuídos primeiramente aos *RNFs Secundários* e em seguida aos *RNFs Específicos*. Quanto mais relevante for o RNF para satisfazer o *RNF Primário*, maior deverá ser seu peso, de forma que quando um *RNF Primário* está decomposto em vários *RNFs Secundários*, a soma dos pesos desses *RNFs Secundário* deve ser 10 (dez). Da mesma forma, quando um *RNF Secundário* está decomposto em *RNFs Específicos*, o somatório dos pesos desses *RNFs Específicos* deve ser igual a 10 (dez). A Tabela 1 ilustra um exemplo da distribuição dos pesos aos *RNFs Secundários e Específicos* originados pela decomposição do *RNF Primário Confiabilidade*.

Política: 1. Qualidade Funcional					
RNF Primário: 1.1 RNF Confiabilidade			Categoria: Especificação Forte		
Conformidade Obtida:					
Decomposições			Pesos	Conformidade	
Código	Denominação			Cálculo	Valor
1.1.1	RNF Persistência dos dados do paciente		1		
1.1.2	RNF Completeza		2		
1.1.3	RNF Precisão		2		
1.1.4	RNF Consistência dos dados		1		
1.1.5	RNF Tolerância a falhas		2		
1.1.5.1	RNF Recuperabilidade		7		
1.1.5.1.1	RNF Capacidade de reverter funções de efeito drástico		10		
1.1.5.2	RNF Capacidade de interromper um processamento demorado		3		
1.1.6	RNF Maturidade		2		
1.1.6.1	RNF Prevenção e contingência		10		

Tabela 1 – Atribuição de pesos aos elementos da árvore

- ✓ **Atribuir Notas aos RNFs Específicos:** Ao se iniciar esta atividade já se obteve a árvore de *RNFs Primários, Secundários e Específicos* estruturada, juntamente com as planilhas de informações preenchidas e os pesos associados a todos os nós da árvore. Nessa atividade os avaliadores devem procurar no documento de requisitos pelos *RNFs Específicos* descritos na árvore, sendo que a cada um desses requisitos da árvore, deve ser associada uma nota, seguindo a escala da Tabela 2. Aos requisitos que não constam no documento, atribui-se a conformidade “não atende” (nota zero), pois

se o documento não prevê, considera-se que não atende ao *RNF Específico* de qualidade, conseqüentemente, o produto final dificilmente atenderá.

Conformidade	Nota
Não Atende o RNF de qualidade	0
Atende com serias restrições	3
Atende Parcialmente	6,5
Atende Completamente	10

Tabela 2 - Conformidade versus nota (REIS 2004)

A conformidade dos RNFs Secundários será calculada através da conformidade dos respectivos RNFs Específicos, associados a seus pesos.

- ✓ **Calcular Notas dos RNF Primários e Secundários** O objetivo desta atividade é efetuar o cálculo que consolida todas as notas de conformidade dos *RNFs Primários e Secundários*, levando em consideração os pesos dos nós. Neste estágio do processo, todos os *RNFs Específicos* estão com suas notas associadas. A nota de cada nó pai é calculada através do somatório das notas vezes os pesos dos nós filhos dividido por 10 (dez). A divisão por 10 (dez) serve para garantir que a nota calculada não ultrapasse o valor 10 (dez). O cálculo é feito utilizando a fórmula da Figura 18.

$$\text{Nota-Nó-Pai} = \left(\sum_{j=1}^n (\text{peso} * \text{nota}) \right) / 10$$

Onde n corresponde ao número de filhos

Figura 18 – Fórmula para o cálculo da nota de nó pai (REIS 2004)

Ao final desta fase já se tem a conformidade de todos os *RNFs Primários* em relação ao que está especificado no documento de requisito, possibilitando uma comparação com a conformidade pretendida especificada na política de qualidade.

- ✓ **Identificar Conformidades e Contradições:** esta atividade consiste em estabelecer os relacionamentos entre a conformidade dos *RNFs Primários* baseada no documento de requisitos e a conformidade estabelecida na política, onde os relacionamentos de cooperação são marcados com o símbolo “✓” quando a conformidade obtida, na

atividade anterior, for maior ou igual a conformidade pretendida, especificada na política de qualidade. Caso contrário, os relacionamentos de conflitos são marcados com um “x” na tabela de resultado das conformidades. A tabela de resultado das conformidades terá a seguinte estrutura (RNF Primário, Conformidade especificada na política de qualidade, conformidade especificada no documento de requisito, resultado do relacionamento);

- ✓ ***Desenvolver uma Estratégia de Ação:*** A execução desta atividade depende do tipo de relacionamento encontrado na tabela de obediências, podendo seguir um dos dois caminhos: retornar ao processo de especificação dos requisitos não funcionais, sempre que um RNF conflite com a política de qualidade; ou atualizar a política, sempre que a mesma não atenda os RNFs especificados.

3.5 Estudo Comparativo

Numa metodologia que se propõe capturar RNFs, medir e avaliar a qualidade da definição destes requisitos, especificados no documento de requisitos, é importante que se defina quais RNFs se pretende capturar e avaliar, assim como a conformidade mínima desejada para cada requisito.

A abordagem proposta se inicia com a elicitação dos RNFs. Nesta fase, a metodologia acrescentou a necessidade da qualidade de padronização às necessidades de qualidade proposta por Loucopoulos (1995), para definir o conjunto de requisitos que se deseja capturar. Esta necessidade foi adicionada por entendermos que a qualidade de padronização é indispensável para aplicações de natureza médica.

Na fase de decomposição e representação dos RNFs, a metodologia proposta utiliza o conhecimento do engenheiro de requisitos e as operacionalizações descritas nos cenários para decompor os RNFs. Chung (1999) divide a decomposição dos RNFs em tipo e tópico. A decomposição por tipo usa os métodos de decomposição existentes num framework, que indicam quais são as decomposições possíveis para um determinado RNF. A decomposição por tópicos trata da decomposição estrutural do problema, ou seja, um RNF aplicado a uma determinada entidade. Quanto a representação dos RNFs, Chung (1999) usa grafos para representá-los, já a nossa abordagem usa uma árvore de RNFs.

A fase de especificação dos RNFs, a metodologia proposta utilizou as planilhas propostas por Reis (2004), com adaptação de algumas informações.

Já a fase de identificação das Influências dos RNFs da abordagem, utiliza a heurística de comparação por pares proposta em Cysneiros (2002), com a finalidade de garantir que todos os RNFs serão avaliados quanto a influência de um no outro.

Um ponto diferencial da metodologia proposta e outras abordagens é a fase de criação da política de qualidade. A metodologia adota um modelo que visa estabelecê-la de maneira mais clara, menos ambígua e contendo, apenas, os aspectos mais relevantes, beneficiando, assim, tanto a organização quanto os usuários.

A fase de avaliação da metodologia proposta foi dividida nas seguintes atividades: Associar pesos aos RNFs Secundários e Específicos(i), Atribuir Notas aos RNFs Específicos(ii), Calcular Notas dos RNFs Primários e Secundários (iii), Identificar Obediências e Contradições (iv) e Desenvolver uma Estratégia de Ação(v). As Atividades (i), (ii) e (iii) são semelhantes respectivamente as fases 3, 4 e 5 da metodologia REQE (REIS 2004), exceto pelo fato de que a metodologia proposta não associa peso aos RNFs Primários (características) porque o objetivo é encontrar a nota de conformidade de cada RNF Primário, e não do documento de requisito como propõe o método REQE. Já as atividades (iv) e (v) são introduzidas pela metodologia proposta para comparar as conformidades obtidas e desejadas.

É importante ressaltar que, embora o método proposto tenha sido criado sob a ótica de capturar RNF e medir a qualidade da definição destes requisitos em aplicações médicas, nada impede que o mesmo possa ser empregado para capturar e medir a qualidade da definição de RNFs em aplicações de outros domínios. Neste caso, será necessário reavaliar a política e o modelo de qualidade proposto de forma que reflita as necessidades de qualidade para o domínio em questão.

4 ESTUDO DE CASO

Considerando os módulos de atendimento, consulta, internação, laboratório e radiologia de um sistema de informações hospitalares trabalhando de forma integrada. O paciente, ao chegar no hospital, deve fazer seu cadastro no atendimento, em seguida ser encaminhado para consulta. O médico deve examiná-lo, fazer o exame clínico e cadastrar o resultado. Após a avaliação, o paciente deve ser liberado ou internado. Vamos considerar a situação de internação. O médico então deve autorizar a internação, solicitar os exames laboratoriais e/ou radiológicos necessários e cadastrar as prescrições para o paciente.

O enfermeiro deve registrar a internação e efetivar as prescrições cadastradas pelo médico. O radiologista e o técnico do laboratório devem receber eletronicamente as solicitações de exames radiológicos e laboratoriais respectivamente, fazer os exames e cadastrar os resultados. O técnico do laboratório deve também emitir etiquetas de identificação das amostras e colher material. Somente o bioquímico responsável pode cadastrar os exames com valores fora da faixa de normalidade, assim como liberar laudos para impressão.

O médico pode consultar os resultados dos exames laboratoriais. Os exames radiológicos podem ser consultados através de busca por conteúdo de imagem, de modo que se possa comparar a imagem radiográfica do paciente com outras imagens de exames já diagnosticados.

O sistema deve disponibilizar um menu de transações para os usuários de acordo com perfil de acesso de cada usuário, de modo que cada usuário só veja as transações de acordo com o seu perfil. O sistema também deve pedir confirmação de usuário e senha antes de efetivar qualquer transação e registrar num arquivo de eventos cada transação efetivada.

Este estudo de caso será utilizado para mostrar a aplicabilidade da metodologia proposta. Consideramos que os requisitos funcionais já estão elicitados e que os seguintes cenários estão definidos.

Cenário: C0 – Autenticação de Usuários **Ator:** Qualquer Usuário do sistema

Pré-Condição: Usuário não Autenticado.

Descrição

1. O sistema disponibiliza interface para autenticação do usuário;
2. O usuário digita sua identificação e senha;
3. O sistema valida a identificação do usuário com a senha;
4. Se a identificação do usuário e a senha não são válidos, o sistema emite mensagem “Acesso Negado”;
5. Caso contrário autentica o usuário;

Pós-Condição: Usuário Autenticado.

Cenário: C1 - Cadastrar Pacientes

Ator: Atendente

Pré-Condição: Paciente não cadastrado no Sistema.

Descrição

6. O atendente seleciona a transação *Cadastrar Pacientes* no menu do sistema.;
7. O sistema disponibiliza a tela de cadastro de pacientes;
8. O atendente digita os dados do paciente;
9. O sistema valida os dados;
10. O sistema pede confirmação de usuário e senha;
11. O sistema verifica permissão para cadastrar paciente;
12. Se usuário não tem permissão sistema emite mensagem “Acesso Negado”;
13. O sistema gera registro com transação, paciente, usuário, data e hora no arquivo de eventos;
14. O sistema armazena as informações do paciente no banco de dados;
15. O sistema emite mensagem “Operação Realizada com Sucesso”;

Pós-Condição: Dados do paciente Cadastrado.

Cenário: C2 - Cadastrar Exames Clínicos do Pacientes

Ator: Médico

Pré-Condição: Paciente Cadastrado

Descrição

1. O médico seleciona a transação *Exames Clínicos* no menu do sistema;
2. O sistema disponibiliza a tela de cadastro dos de exames clínicos;
3. O médico digita os dados do exame clínico do paciente;
4. O sistema valida os dados digitados e pede confirmação de usuário e senha;
5. Se usuário não tem acesso a transação, sistema emite mensagem “Acesso Negado”;
6. O sistema gera registro com transação, paciente, usuário, data e hora no arquivo de eventos;
7. O sistema armazena as informações no banco de dados;
8. O sistema emite mensagem “Operação Realizada com Sucesso”;

Pós-Condição: Exames Clínicos do paciente cadastrados.

Cenário: C3 - Autorizar Internação de Pacientes

Ator: Médico

Pré-Condição: Paciente Cadastrado, situação paciente = “cadastrado”

Descrição

1. O médico seleciona a transação *Autorizar Internação* no menu do sistema.;
2. O sistema disponibiliza a tela para autorização de internação de pacientes;
3. O médico digita os dados para autorizar a internação do paciente;
4. O sistema valida os dados e pede confirmação de usuário e senha;
5. Se usuário não tem acesso a transação o sistema emite mensagem “Acesso Negado”;
6. O sistema gera registro com transação, paciente, usuário, data e hora no arquivo de eventos;
7. O sistema armazena as informações no banco de dados;
8. O sistema emite mensagem “Operação Realizada com Sucesso”;

Pós-Condição: Autorização da Internação do paciente Cadastrada. Situação=”Autorizada”

Cenário: C4 - Autorizar Prescrições do Paciente

Ator: Médico

Pré-Condição: Paciente Cadastrado.

Descrição

1. O médico seleciona a transação *Autorizar Prescrições* no menu do sistema.;
2. O sistema disponibiliza a tela de Autorização de prescrições de pacientes;
3. O médico digita os dados necessários para autorizar a prescrição do paciente;
4. O sistema valida os dados e pede confirmação de usuário e senha;
5. Se usuário não tem acesso a transação o sistema emite mensagem “Acesso Negado”;
6. O sistema gera registro com transação, paciente, usuário, data e hora no arquivo de eventos;
7. O sistema armazena as informações da autorização no banco de dados;
8. O sistema emite mensagem “Operação Realizada com Sucesso”;

Pós-Condição: Autorização de Prescrições do Paciente Cadastrada. Situação=”Autorizada”.

Cenário: C5 - Efetivar Internações.

Ator: Enfermeiro

Pré-Condição: Autorização da Internação Cadastrada. Situação = “Autorizada”

Descrição

1. O enfermeiro seleciona a transação *Cadastrar Internações* no menu do sistema.;
2. O sistema disponibiliza a tela de cadastro de internações de pacientes;
3. O enfermeiro digita os dados do cadastro de internações do paciente;
4. O sistema valida os dados de internação e pede confirmação de usuário e senha;
5. Se usuário não tem acesso a transação o sistema emite mensagem “Acesso Negado”;
6. O sistema gera registro com transação, paciente, usuário, data e hora no arquivo de eventos;
7. O sistema armazena as informações no banco de dados;
8. O sistema emite mensagem “Operação Realizada com Sucesso”;

Pós-Condição: Internação Efetivada. Situação da Internação=”Efetivada”, Situação do paciente=”Internado”

Cenário: C6 - Efetivar Prescrições

Ator: Médico/Enfermeiro

Pré-Condição: Autorização da Prescrições Cadastrada, Situação Prescrição = “Autorizada”

Descrição

1. O médico/enfermeiro seleciona a transação *Efetivar Prescrições* no menu do sistema.;
2. O sistema disponibiliza a tela de efetivação de prescrições e solicita a identificação do paciente;
3. O médico/enfermeiro informa o paciente;
4. O sistema disponibiliza todas as prescrições Autorizadas e não efetivadas para o paciente;
5. O médico/enfermeiro seleciona a prescrição desejada para efetivar;
6. O médico/enfermeiro digita os dados para efetivar a prescrição;
7. O sistema valida os dados e pede confirmação de usuário e senha;
8. Se usuário não tem acesso a transação o sistema emite mensagem “Acesso Negado”;
9. O sistema grava transação, paciente, usuário, data e hora no arquivo de eventos;
10. O sistema armazena as informações no banco de dados;
11. O sistema emite mensagem “Operação Realizada com Sucesso”;

Pós-Condição: Prescrições Efetivadas. Situação da Prescrição = “Efetivada”

Cenário: C7 - Autorizar Alta de Pacientes

Ator: Médico

Pré-Condição: Internação Cadastrada.

Descrição

1. O médico seleciona a transação *Autorizar Alta de Pacientes* no menu do sistema.;
2. O sistema disponibiliza a tela de cadastro dos pacientes;
3. O médico digita os dados para autorizar a alta do paciente;
4. O sistema valida os dados e solicita o usuário e a senha;
5. O sistema armazena as informações no banco de dados;
6. O sistema emite mensagem “Operação Realizada com Sucesso”

Pós-Condição: Alta do paciente autorizada.

Cenário: C8 - Efetivar Alta do Paciente

Ator: Médico/Enfermeiro

Pré-Condição: Alta do Paciente Autorizada, Situação do paciente = “Internado”

Descrição

1. O médico/enfermeiro seleciona a transação *Efetivar Alta de Pacientes* no menu do sistema.;
2. O sistema disponibiliza a tela para efetivação de alta dos pacientes;
3. O médico/enfermeiro digita os dados de alta do paciente;
4. O sistema valida os dados e solicita o usuário e a senha;
5. O sistema armazena as informações no banco de dados;
6. O sistema emite mensagem “Operação Realizada com Sucesso”;

Pós-Condição: Alta do Paciente Efetivada. Situação do paciente=“cadastrado”

Cenário: C9 - Receber Solicitações de Exames

Ator: Técnico laboratório

Pré-Condição: Solicitação de Exames Cadastradas

Descrição

1. O técnico do laboratório seleciona a transação *Receber Solicitações de Exames* no menu do sistema;
2. O sistema disponibiliza a tela de recebimento de solicitações de exames;
3. O técnico do laboratório digita os dados para receber as solicitações de exames;
4. O sistema valida os dados e solicita o usuário e a senha;
5. O sistema armazena as informações no banco de dados;
6. O sistema emite mensagem “Operação Realizada com Sucesso”;

Pós-Condição: Solicitação de Exames Recebida. Situação exame = “Recebido”.

Cenário: C10 - Cadastrar Laudos de Exames

Ator: Técnico do Laboratório

Pré-Condição: Exame solicitado.

Descrição

1. O técnico laboratório seleciona a transação *Cadastrar Laudos* no menu do sistema;
2. O técnico informa o código da solicitação do exame;
3. O sistema disponibiliza os exames solicitados na solicitação;
4. O técnico digita o resultado dos exames;
5. O sistema valida os dados e solicita o usuário e a senha;
6. O sistema armazena as informações no banco de dados;
7. O sistema emite mensagem “Operação Realizada com Sucesso”.

Pós-Condição: Laudo Cadastrado. Situação do Exame = “Concluído”

Cenário: C11 - Consultar Prescrições

Ator: Médico/Enfermeiro

Pré-Condição: Prescrições Cadastradas

Descrição

1. O médico/enfermeiro seleciona a transação *Consultar Prescrições* no menu do sistema;
2. O sistema disponibiliza a tela de consulta prescrições dos pacientes;
3. O médico/enfermeiro digita os dados para consultar as prescrições do paciente;
4. O sistema valida os dados e solicita o usuário e a senha;
5. O sistema disponibiliza as informações da consulta.

Pós-Condição: Prescrições Consultadas.

Cenário: C12 - Solicitar Exames Laboratoriais

Ator: Médico

Pré-Condição: Dados do paciente cadastrados

Descrição

1. O médico seleciona a transação *Solicitar Exames* no menu do sistema;
2. O médico faz pesquisa para buscar o paciente;
3. O médico seleciona os exames laboratoriais que o paciente deve fazer;
4. O sistema solicita o usuário e a senha;
5. O sistema valida o usuário e a senha;
6. O sistema gera automaticamente um código de solicitação;
7. O sistema armazena as informações no banco de dados;
8. O sistema emite mensagem “Operação Realizada com Sucesso”.

Pós-Condição: Solicitação de exame cadastrada, Situação exame = “solicitado”.

Cenário: C13 - Liberar Laudos para Emissão

Ator: Bioquímico Responsável pelo laboratório.

Pré-Condição: Resultado dos Exames Cadastrados

Descrição

1. Bioquímico seleciona a transação *Liberar Laudos* no menu do sistema;
2. O sistema apresenta uma lista de laudos com todos os resultados de exames cadastrados;
3. O bioquímico seleciona os laudos desejados;
4. O sistema solicita o usuário e a senha para confirmar a operação;
5. O sistema valida o usuário e a senha;
6. O sistema armazena as informações no banco de dados;
7. O sistema emite mensagem “Operação Realizada com Sucesso”.

Pós-Condição: Laudos Liberados para Emissão, Situação exame = “liberado”.

Cenário: C14 - Consultar Imagens por Conteúdo

Ator: Médico/Radiologista

Pré-Condição: Imagem Cadastrada

Descrição

1. Médico seleciona a transação *Consultar Imagens* no menu do sistema;
2. O sistema apresenta a tela de consulta;
3. O médico seleciona uma imagem do banco de dados e solicita a consulta;
4. O sistema solicita o usuário e a senha para confirmar a operação;
5. O sistema valida o usuário e a senha;
6. O sistema busca as 10 imagens mais semelhantes a imagem de consulta;
7. O sistema mostra o resultado da consulta.

Pós-Condição: Imagem Consultada

Cenário: C15 - Consultar Solicitações de Exames

Ator: Médico/Enfermeiro

Pré-Condição: Solicitação de Exames Cadastrada.

Descrição

1. O médico seleciona a transação *Consultar Solicitações de Exames* no menu do sistema;
2. O sistema disponibiliza a tela para consultar as solicitações de exames;
3. O médico digita os dados para consultar as solicitações de exames paciente;
4. O sistema valida os dados e solicita o usuário e a senha;
5. O sistema armazena as informações no banco de dados;
6. O sistema emite mensagem “Operação Realizada com Sucesso”.

Pós-Condição: Solicitação de Exames Consultada.

Cenário: C16 - Cadastrar Resultados de Exames

Ator: Técnico laboratório

Pré-Condição: Solicitação de Exames Recebida

Descrição

1. O técnico laboratório seleciona a transação *Cadastrar Resultados de exames* no menu do sistema;
2. O sistema disponibiliza a tela de cadastro de resultado de exames;
3. O técnico do laboratório digita os dados do resultado dos exames do paciente;
4. O sistema valida os dados e solicita o usuário e a senha;
5. O sistema armazena as informações no banco de dados;
6. O sistema emite mensagem “Operação Realizada com Sucesso”.

Pós-Condição: Resultados de Exames Cadastrados.

Cenário: C17 - Emitir Etiquetas com Identificação das Amostras

Ator: Técnico laboratório

Pré-Condição: Solicitação de Exames Recebida

Descrição

1. O técnico do laboratório seleciona a transação *Emitir Etiquetas das Amostras* no menu do sistema;
2. O sistema disponibiliza a tela de emissão de etiquetas das amostras de exames;
3. O técnico do laboratório digita os dados da solicitação do exames;
4. O sistema valida os dados e solicita o usuário e a senha;
5. O sistema armazena as informações no banco de dados;
6. O sistema emite mensagem “Operação Realizada com Sucesso”.

Pós-Condição: Etiquetas com identificação das amostras emitida.

Cenário: C18 – Exportar dados

Ator: Cliente (SUS, ANS) etc...

Pré-Condição: Serviços de exportação disponíveis

Descrição

- 1 O cliente envia requisição ao serviço de exportação disponível pelo sistema;
- 2 O sistema recebe requisição do cliente, processa e envia resultado ao cliente;

Pós-Condição: Requisição Atendida.

<p>Cenário: C19 – Importação de dados</p> <p>Ator: Sistema</p> <p>Pré-Condição: Serviços de Importação disponíveis</p> <p>Descrição</p> <ol style="list-style-type: none"> 1 O Sistema envia requisição a determinada aplicação especificada; 2 A aplicação recebe requisição do sistema, processa e envia resultado ao sistema; 3 O sistema recebe o resultado da requisição, grava no banco de dados ou disponibiliza em forma de consulta ou relatório. <p>Pós-Condição: Importação Efetuada.</p>
--

Partimos do modelo de qualidade proposto Figura 8, para definir os requisitos não funcionais que servirão de base para modelar a qualidade desejada para aplicação.

Analisando os cenários C1 – Cadastrar Pacientes, C10 – Cadastrar Laudos de Exames e C16 – Cadastrar Resultado de Exames, percebemos eles requerem o *RNF Confiabilidade* como forma de garantir a confiabilidade dos dados do paciente, dos laudos e dos resultados dos exames. Já os cenários C13 - Liberar Laudos para Emissão e C12 - Solicitar Exames Laboratoriais requerem uma atenção especial quanto ao *RNF Segurança*, já que somente os médicos podem solicitar exames e somente o bioquímico responsável pode liberar os laudos para emissão.

Continuando com a análise dos cenários, verificamos que o cenário C14 - Consultar Imagens por Conteúdo, requer o requisito não funcional *RNF Eficiência*. Outro exemplo facilmente observado é que o *RNF Usabilidade* se aplicado a todos os cenários pode contribuir para melhorar a usabilidade das operações do sistema.

Necessidades	RNF Primário	Cenários
1. Qualidade Funcional	<i>1.1. RNF Confiabilidade</i>	C1,C10,C16
	<i>1.2. RNF Usabilidade</i>	
	<i>1.3. RNF Segurança</i>	C12, C13
2. Qualidade Desempenho	<i>2.1. RNF Eficiência</i>	C14
	<i>2.2. RNF Interoperabilidade</i>	C18, C19
3. Qualidade Manutenção	<i>3.1. RNF Manutenibilidade</i>	
	<i>3.2. RNF Flexibilidade</i>	
	<i>3.3. RNF Portabilidade</i>	
4. Qualidade Padronização	<i>4.1. RNF Padrões de Intercomunicação</i>	C18, C19
	<i>4.2. RNF Padrões de Vocabulário</i>	C12
	<i>4.3. RNF Padrões de Interfaces</i>	C18, C19
	<i>4.4. RNF Padrões de Segurança</i>	C18, C19
	<i>4.5. RNF Padrões de formatação e conteúdo (para comunicação e troca de informação)</i>	C18, C19

Tabela 3 - Lista de Requisitos não funcionais Elicitados (LRNFE)

A fase seguinte da metodologia proposta será de decomposição dos RNFs Primários de modo a modelar as necessidades de qualidade dos *stakeholders*. Os RNFs foram decompostos seguindo as instruções da fase de decomposição e representação da metodologia proposta descritas na seção 3.4.2 deste trabalho. A seguir será mostrada a decomposição de cada RNF Primário inicialmente elicitado.

- ✓ A *Necessidade de Qualidade Funcional* foi modelada da seguinte maneira:
 - **O RNF Confiabilidade** quando aplicado aos cenários *C1 - Cadastrar paciente e C16 - Cadastrar laudos* requer os requisitos “*Persistências*”, “*Completeza*”, “*Precisão*” e “*Consistência*” dos dados do paciente e dos laudos. O RNF Confiabilidade também requer o requisito “*Tolerância a falhas*” que por sua vez se decompõe em “*Recuperabilidade*” e “*Capacidade de interromper processamento demorado*”. O requisito não funcional “*Recuperabilidade*” foi decomposto no requisito “*Capacidade de reverter funções de defeito drástico*”. Finalmente a confiabilidade do software requer o requisito “*Maturidade*” que pode ser garantido pelo requisito “*Prevenção e Contingência*”;
 - **O RNF Usabilidade** foi decomposto segundo quatro aspectos: “*Aspectos de aquisição de dados*”, “*Aspectos de saída*”, “*Aspectos de Ajuda*” e “*Aspectos Gerais de Usabilidade*”. Decompondo o aspecto de aquisição de dados encontramos os requisitos “*Identificação única para o paciente*”, “*Importação de dados externos*”, “*Aquisição de textos, sinais, imagens, sons e código de barra*”, “*Estrutura do prontuário do paciente*” e “*Uso de processos inteligentes*”. O requisito não funcional “*Uso de processos inteligentes*” foi decomposto em “*Reconhecimento de textos, sons e imagens*”. Quanto aos *Aspectos de saída*, foram decompostos os seguintes requisitos não funcionais “*Busca por diversas informações*”, “*Exporta dados do paciente para outros sistemas*”, “*Saída dos resultados em vídeo ou impressoras*”, “*utiliza código de barra*”, “*assinaturas digitais dos relatórios*” e “*busca de imagens por conteúdo*”. Os requisitos “*Busca por diversas informações*” e “*Saída dos resultados em vídeo ou impressoras*” foram decompostos respectivamente nos requisitos “*Apresentação uniforme dos dados do paciente*” e “*Elaboração de Estatística com geração automática de gráficos*”. Os “*Aspectos de Ajuda*” foram decompostos nos requisitos “*Ajuda on-line sensível ao contexto*”, “*Manual do usuário atualizado*” e “*Mensagens auto explicativas*”. O requisito “*Manual do usuário atualizado*” foi decomposto nos

requisitos “*Documentação do usuário de fácil compreensão*”, “*Sumário e índice remissivo na documentação do usuário*” e “*Presença de manual de instalação com instruções detalhadas*”. O requisito “*Mensagens auto explicativas*” também foi decomposto nos requisitos específicos “*Consistência de vocabulário entre as mensagens e a documentação*”, “*Diferenciação dos tipos de mensagens*”, “*mensagens de erro com informações necessárias para solução do erro*” e “*Alertas claros para as conseqüências de uma confirmação*”. Finalmente o RNF “*Aspectos Gerais de Usabilidade*” foi decomposto nos requisitos “*Aprendibilidade*”, “*Operacionalidade*” e “*Atratividade*”;

- **O RNF Segurança** requer um “*Mecanismos de administração de usuários centralizado*”, “*Política de controle de acesso*”, “*Políticas de Senhas*”, “*Disponibilidade*” e “*Segurança do ambiente*”. O RNF “*Mecanismos de administração de usuários*” requer o requisito “*Registro das autorizações de acesso*”. O RNF Secundário “*Política de controle de acesso*” por sua vez se decompôs nos RNFs Específicos “*Autenticação e Identificação única de usuários*”. Já o RNF Secundário “*Políticas de Senhas*” foi decomposto nos RNFs Específicos “*Limitar a existência de senha*”, “*Atualização periódica de senhas*”, “*Não reutilização de senhas antigas*” e “*Proibir a auditoria de informações relativas a senhas*”. O RNF *Disponibilidade* também foi decomposto nos RNFs Específicos “*Redundância de equipamento*”, “*Copias de Segurança*” e “*Auditoria*”. O RNF “*Segurança do Ambiente*” se decompôs em “*Restringir acesso à sala dos servidores*” e “*Proteção contra incêndios e inundações*”.

✓ A *Necessidade de Qualidade de Desempenho* foi modelada da seguinte maneira:

- **O RNF Eficiência** foi decomposto nos RNFs Secundários “*Eficiência de execução*” e “*Eficiência de armazenamento*”. O RNF “*Eficiência de execução*” quando aplicado ao cenário *C14 – Consultar imagens por conteúdo*, se decompôs em “*Busca de imagens não pode ultrapassar 3 segundos*” que por sua vez se decompôs em “*Usar indexação de imagens médicas*”. Por fim, o RNF “*Eficiência de armazenamento*” se decompôs em “*Armazenar imagens compactadas*”;
- **O RNF Interoperabilidade** se decompôs nos RNFs secundários “*Interoperabilidade técnica*” e “*Interoperabilidade Semântica*”. O RNF “*Interoperabilidade técnica*” se

decompôs em “*Compatibilidade*”. Já o RNF “*Interoperabilidade Semântica*” foi decomposto no RNF específico “*Inteligibilidade*”;

- ✓ A *Necessidade de Qualidade de Manutenção* foi modelada da seguinte maneira:
 - O RNF *Manutenibilidade* foi decomposto nos RNFs Específicos “*Analisabilidade*”, “*Modificabilidade*”, “*Estabilidade*” e “*Testabilidade*”;
 - O RNF *Flexibilidade* foi decomposto no RNF específico “*Independência dos dados do paciente*”;
 - O RNF *Portabilidade* se compôs nos RNFs específicos “*Adaptabilidade*” e “*Coexistência*”;
- ✓ Finalmente o modelo de qualidade para a *Necessidade Padronização* foi assim definido:
 - O RNF *Padrões de Intercomunicação* se compôs nos RNFs específicos “*Utilizar protocolos padronizados*” e “*Uniformidade do formato dos dados transmitidos*”;
 - O RNF *Padrões de Vocabulário* quando aplicado ao cenário *C12 – Solicitar exames laboratoriais*, requer o requisito não funcional “*Utilizar padrões para identificar as informações*”;
 - O RNF *Padrões de Interfaces* foi decomposto em “*Interface do usuário*”, “*Interface de hardware*” e “*Interface de Software*”. O RNF secundário “*Interface do usuário*”, requer os requisitos “*Usar padrões gráficos de interfaces*” e “*Utilizar terminologia condizente com a prática médica*”. O RNF “*Usar padrões gráficos de interfaces*” ainda requer o requisito específico “*Clareza nos formatos das telas de entrada e relatórios*”. Já o RNF “*Interface de Software*” requer o requisito específico “*Prover padrões de interoperabilidade e portabilidade*”;
 - O RNF *Padrões de Segurança* foi decomposto no requisito “*Usar padrões de segurança para web*” e este por sua vez se compôs em “*Padrões XML*”;
 - O RNF *Padrões de Formatação e Conteúdo* foi decomposto no RNF específico “*Usar padrões de conteúdo*”.

Ao final desta fase, cada RNF primário deverá ter sua árvore com suas decomposições. Juntando todas estas árvores de requisitos primários, obtém-se uma árvore geral de decomposição de todos os requisitos não funcionais. Esta árvore é imprescindível

para aplicação da metodologia, pois somente a partir dela as fases seguinte poderão ser executadas. A Figura 19 mostra como ficou a árvore de decomposição dos requisitos não funcionais deste estudo de caso:

1. Qualidade Funcional

1.1 RNF Confiabilidade

- 1.1.1 RNF Persistência dos dados
- 1.1.2 RNF Completeza
- 1.1.3 RNF Precisão
- 1.1.4 RNF Consistência dos dados
- 1.1.5 RNF Tolerância a falhas
 - 1.1.5.1 RNF Recuperabilidade em caso de falha
 - 1.1.5.1.1 RNF Capacidade de reverter funções de efeito drástico
 - 1.1.5.2 RNF Capacidade de interromper um processamento demorado
- 1.1.6 RNF Maturidade
 - 1.1.6.1 RNF Prevenção e contingência

1.2 RNF Usabilidade

- 1.2.1 RNF Aspectos de aquisição de dados
 - 1.2.1.1 RNF Identificação única para o paciente
 - 1.2.1.2 RNF Importação de dados externos
 - 1.2.1.3 RNF Uso de processos inteligentes
 - 1.2.1.3.1 RNF Reconhecimento de textos, sons e imagens
 - 1.2.1.4 RNF Aquisição de textos, sinais, imagens, sons e código de barra
 - 1.2.1.5 RNF Estrutura do prontuário do paciente
- 1.2.2 RNF Aspectos de saída
 - 1.2.2.1 RNF Busca por diversas informações
 - 1.2.2.1.1 RNF Apresentação uniforme dos dados do paciente
 - 1.2.2.2 RNF Exporta dados do paciente para outros sistemas
 - 1.2.2.3 RNF Saída dos resultados em vídeo ou impressora
 - 1.2.2.3.1 RNF Elaboração de estatística com geração automática de gráficos
 - 1.2.2.4 RNF Utiliza código de barra
 - 1.2.2.5 RNF Assinaturas Digitais dos relatórios
 - 1.2.2.6 RNF Busca de imagens por conteúdo.
- 1.2.3 RNF Aspectos de Ajuda
 - 1.2.3.1 RNF Ajuda *on-line* sensível ao contexto
 - 1.2.3.2 RNF Manual do usuário atualizado
 - 1.2.3.2.1 RNF Documentação do usuário de fácil compreensão
 - 1.2.3.2.2 RNF Sumário e índice remissivo na documentação do usuário
 - 1.2.3.2.3 RNF Presença de Manual de instalação com instruções detalhadas
 - 1.2.3.3 RNF Mensagens auto-explicativas
 - 1.2.3.3.1 RNF Consistência de vocabulário entre as mensagens e a documentação
 - 1.2.3.3.2 RNF Diferenciação dos tipos de mensagens
 - 1.2.3.3.2.1 RNF Mensagens de erro com informações necessárias para a solução da situação de erro
 - 1.2.3.3.2.2 RNF Alertas claros para as conseqüências de uma determinada confirmação
- 1.2.4 RNF Aspectos Gerais de Usabilidade
 - 1.2.4.1 RNF Aprendibilidade
 - 1.2.4.2 RNF Operacionalidade

1.2.4.3 RNF Maturidade

1.3 RNF Segurança

1.3.1 RNF Mecanismos de administração de usuários centralizados

1.3.1.1 RNF Registro das autorizações de acesso

1.3.2 RNF Política de controle de acesso

1.3.2.1 RNF Autenticação

1.3.2.2 RNF Identificação única de usuários

1.3.3 RNF Políticas de senhas

1.3.3.1 RNF Limitar a existência de senha.

1.3.3.2 RNF Atualização periódica de senhas

1.3.3.3 RNF Não reutilização de senhas antigas

1.3.3.4 RNF Proibir a auditoria de informação relativa a senhas

1.3.3.5 RNF Manipular senhas criptografadas

1.3.4 RNF Disponibilidade

1.3.4.1 RNF Redundância de hardware e software

1.3.4.2 RNF Cópias de Segurança

1.3.4.3 RNF Auditoria

1.3.5 RNF Segurança do Ambiente

1.3.5.1 RNF Restringir acesso a sala dos servidores

1.3.5.2 RNF Proteção contra incêndios e inundações

2 Qualidade Desempenho

2.1 RNF Eficiência

2.1.1 RNF Eficiência de execução

2.1.1.1 RNF Busca de imagens não pode ultrapassar 3 segundos

2.1.1.1.1 RNF Usar indexação de imagens medicas

2.1.2 RNF Eficiência de Armazenamento

2.1.2.1 RNF Armazenar imagens compactadas

2.2 RNF Interoperabilidade

2.2.1 RNF Interoperabilidade Técnica

2.2.1.1 RNF Compatibilidade

2.2.2 RNF Interoperabilidade Semântica

2.2.2.1 RNF Inteligibilidade

3. Qualidade em Manutenção

3.1 RNF Manutenibilidade

3.1.1 RNF Analisabilidade

3.1.2 RNF Modificabilidade

3.1.3 RNF Estabilidade

3.1.4 RNF Testabilidade

3.2 RNF Flexibilidade

3.2.1 RNF Independência dos dados do paciente

3.3 RNF Portabilidade

3.3.1 RNF Adaptabilidade

3.3.2 RNF Coexistência

4. Qualidade de Padronização

4.1. RNF Padrões de Intercomunicação

4.1.1 RNF Utilizar protocolos padronizados

4.1.2 RNF Uniformidade do formato dos dados transmitidos

4.2. RNF Padrões de Vocabulário

4.2.1 RNF Utilizar padrões para identificar as informações

4.3 RNF Padrões de Interfaces

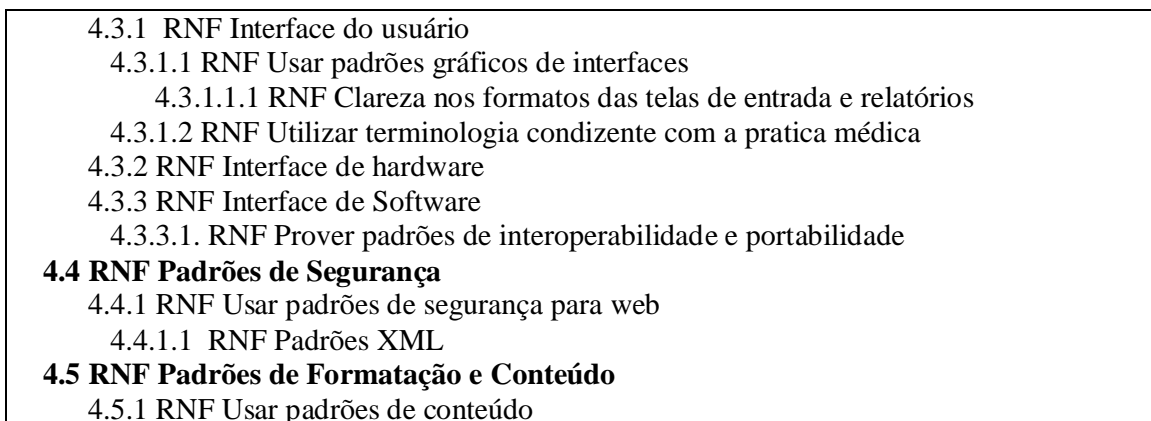


Figura 19 - Árvore de Decomposição dos Requisitos não funcionais

O próximo passo da metodologia é especificar cada RNF de forma descritiva. A metodologia proposta utiliza as planilhas de RNFs Primário, Secundário e Específico para especificar os requisitos não funcionais. Embora as planilhas sejam preenchidas gradativamente em cada fase da metodologia proposta, para efeito de demonstração só mostraremos as planilhas totalmente preenchidas no final do estudo de caso.

De acordo com a metodologia proposta, o próximo passo é a identificação das Influências entre os RNFs. Aplicando as recomendações da fase de identificação das influências descritas na seção 3.4.4 deste trabalho, identificamos as seguintes influências no estudo de caso:

✓ Exemplos de Influências Positivas:

- O “*RNF Manual do usuário atualizado*” contribui positivamente para a realização do “*RNF Manutenibilidade*”;
- O “*RNF Prover padrões de interoperabilidade e portabilidade*” contribui positivamente para realização do requisito “*RNF Interoperabilidade*”;
- Os requisitos “*RNF Redundância de Hardware e Software*” e “*RNF cópias de segurança*” contribuem positivamente para a realização do “*RNF Tolerância a falhas*”;
- Os requisitos “*RNF Uniformidade do formato dos dados transmitidos*” e “*RNF Utilizar padrões para identificar as informações*” contribuem positivamente para a realização do “*RNF Importação de dados externos*”.

✓ Exemplos de Influências Negativas:

- O “*RNF Armazenar Imagens compactadas*” contribui negativamente para a realização do “*RNF busca das imagens não pode ultrapassar 3 segundos*”, já que o processo de descompactar as imagens requer um tempo adicional;
- O “*RNF Segurança*” contribui negativamente para realização do “*RNF Flexibilidade*”, pois enquanto a flexibilidade requer estruturas gerais, a segurança precisa de estruturas mais específicas para garantir certas limitações.
- Outro exemplo de influência negativa foi encontrada com “*RNF Segurança*” que contribui negativamente para a realização do “*RNF Usabilidade*”, pois o processo adicional para controlar o acesso ao sistema aumenta o tempo de execução de certos eventos e dificulta a usabilidade.

Ao final desta fase, as planilhas de RNFs devem ser atualizadas com as Influências positivas e negativas identificadas.

A fase seguinte da metodologia proposta é de criação da política de qualidade que se inicia com a categorização dos RNFs Primários e atribuição das conformidades às categorias. Seguindo as recomendações da atividade de Categorizar os RNFs Primário da fase de criação da política de qualidade descritas na seção 3.4.5 deste trabalho, os RNFs elicitados foram categorizados da seguinte forma:

- ✓ **Especificação Forte:** “*RNF Confiabilidade*”, “*RNF Segurança*”, “*RNF Eficiência*”, “*RNF Manutenibilidade*”, “*RNF Padrões de Intercomunicação*”, “*RNF Padrões de vocabulário*”, “*RNF Padrões de interfaces*”, “*RNF Padrões de segurança*” e “*RNF Padrões de formatação e conteúdo*”;
- ✓ **Especificação Média:** “*RNF Usabilidade*”, “*RNF Interoperabilidade*” e “*RNF Flexibilidade*”;
- ✓ **Especificação Fraca:** “*RNF Portabilidade*”

Para o estudo de caso, de acordo com a seção 3.4.5 deste trabalho foram definidas as conformidades 80%, 65%, 30% para as categorias de especificação Forte, Média e Fraca respectivamente. A Figura 20 mostra como ficou a política de qualidade para o software após a conclusão desta fase:

Política de Qualidade

Francisco Bastos

02/02/2007

Versão: 1

1 INTRODUÇÃO

1.1 DECLARAÇÃO DE PROPÓSITO

Com o propósito de garantir o compromisso e o respeito quanto aos cuidados com as informações dos pacientes durante a realização de exames, consultas, internações etc, essa instituição hospitalar, adota uma política de qualidade para aquisição de software. Esta política tem o objetivo de estabelecer os Requisitos de Qualidade para o software, com a finalidade de garantir que o mesmo tenha a qualidade mínima desejada pela instituição. A aquisição de novas tecnologias, novas funcionalidades para o software poderá levar a mudanças na presente política a qualquer tempo. Contudo, toda alteração deverá ser realizada neste espaço.

1.2 DECLARAÇÃO DE CONFORMIDADES

Esta parte da política de qualidade é reservada para especificar a categoria dos requisitos não funcionais elicitados assim como as conformidades de cada categoria especificada. De acordo com o estudo de caso proposto, os RNFs Primários foram categorizados da seguinte forma:

1.2.1 **Especificação Forte** - “RNF Confiabilidade”, “RNF Segurança”, “RNF Eficiência”, “RNF Manutenibilidade”, “RNF Padrões de Intercomunicação”, “RNF Padrões de vocabulário”, “RNF Padrões de interfaces”, “RNF Padrões de segurança” e “RNF Padrões de formatação e conteúdo”.

1.2.2 **Especificação Média** - “RNF Usabilidade”, “RNF Interoperabilidade” e “RNF Flexibilidade”.

1.2.3 **Especificação Fraca** - “RNF Portabilidade”

Para a aplicação em estudo, a conformidade mínima aceitável de cada categoria será a seguinte:

Categoria	Conformidade Aceitável
Especificação Forte	80%
Especificação Média	65%
Especificação Fraca	30%

É importante ressaltar que a conformidade mínima aceitável de cada categoria definida nesta política, depende da aplicação a que a política está sendo submetida, porém qualquer alteração desta natureza deve ser feita neste local.

2 POLÍTICAS

2.1 POLÍTICA DA QUALIDADE FUNCIONAL

2.1.1 RNF Confiabilidade

Todo sistema é suscetível a falha, a ocorrência de falhas podem ser tolerável desde que o software seja capaz de manter a integridade e a consistência das informações e a frequência com que elas ocorram não comprometam a disponibilidade do software. Por esta razão, esta política é proposta para garantir a capacidade do software em manter um nível de desempenho nos casos de falhas ou de violação nas interfaces especificadas. Para isso, a política sugere que a qualidade e completude das especificações dos requisitos relacionado ao “tratamento de erros”, “prevenção e contingências”, “privacidade”, “tolerância a falhas”, “permanência dos dados”, “completude e precisão dos dados”, “consistência dos dados do paciente” estejam evidentes.

2.1.2 RNF Usabilidade

Muitos usuários criam resistência em usar determinados software pelo fato destes não serem amigáveis e de fácil uso. Por esta razão, estabelece-se esta política com a finalidade de garantir a aquisição de um software de fácil entendimento e aprendizado, que seja legível, fácil de usar e que emita mensagens inteligíveis e informativas para facilitar o uso e o aprendizado. Para garantir estes requisitos, esta política sugere a procura de requisitos relacionados a três aspectos:

- ✓ *Aspectos de aquisição de dados:* neste aspecto sugere-se a evidência de requisitos relacionados a “único identificador para o paciente”, a “existência de importação de dados”, a “manipulação de imagens, sons e textos” e a “estrutura para o prontuário do paciente”;
- ✓ *Aspectos de saída dos dados:* quanto ao aspecto de saída sugere-se que o DRs evidencie requisitos relacionados a “exportação de dados do paciente”, “saída de dados em vídeo ou impressora”, “geração de gráficos”, “assinaturas digitais em relatórios” e “busca de imagens por conteúdo”;
- ✓ *Aspectos de Ajuda:* o aspecto de ajuda busca por requisitos que evidencie “existência de alertas”, “manual do usuário e instalação atualizado fácil de entender”, “ajuda on-line” e “mensagens diferenciadas e auto explicativas”;
- ✓ *Aspectos Gerais de Usabilidade:* busca por requisitos que evidencie “aprendibilidade”, “operacionalidade” e “atratividade” para garantir que o software seja atrativo e facilmente entendido, aprendido e controlado por seus usuários..

2.1.3 RNF Segurança

Os atuais sistemas de segurança das empresas, por mais proteção contra ataques (*firewalls*), encriptações ou sistemas contra vírus que tenham, são facilmente violados pelas pessoas que trabalham com eles. Não há sistema seguro, mas existem medidas que podem ser implementadas para prevenir, detectar ou minimizar problemas desta natureza. Por esta razão é importante pensar na segurança antes e durante o desenho e desenvolvimento do software, envolver os utilizadores na implementação de serviços e mecanismos de segurança e finalmente entender que a segurança deve ser mais uma funcionalidade do sistema, e não algo que impeça sua utilização. Dentro deste propósito esta política busca por evidências de requisitos relacionados a “Mecanismos de administração de usuários centralizado”, “Política de controle de acesso”, “Autenticação”, “Identificação única de usuários”, “Políticas de senhas”, “Cópias de segurança” e

”Proteção contra incêndio e inundação”.

2.2 POLÍTICA DA QUALIDADE DESEMPENHO

2.2.1 RNF Eficiência

A eficiência busca por evidência de requisitos que garantam a quantidade de recursos e código requerido por um programa para realizar suas funções; estes requisitos estão relacionados a “Eficiência de execução e armazenamento”, “Uso de indexação para manipular imagens médicas”, “Tempo estimado para recuperação de imagens” e “Armazenamento de imagens compactadas”.

2.2.2 RNF Interoperabilidade

A interoperabilidade refere-se a possibilidade da interação e troca de dados entre dois ou mais computadores, meios de comunicação, redes, software e outros componentes de tecnologia da informação de forma que se obtenha os resultados esperados. Este requisito contribui positivamente para a política de qualidade desempenho, a medida que evidencia requisitos relacionados a interoperabilidade técnica e semântica como por exemplo “Definição de interfaces abertas”, “Organização e formato de dados e protocolos compatíveis com outras tecnologias” e “Garantia de que a informação trocada é inteligível por qualquer aplicativo”.

2.3 POLÍTICA DA QUALIDADE EM MANUTENÇÃO

Preocupados com a evolução do software e com a correção de possíveis erros durante sua execução, se estabelece esta política com a finalidade de determinar o esforço necessário para fazer modificações especificadas no software, ou seja, diagnosticar deficiências, causas de falhas ou partes a serem modificadas, remover seus defeitos ou adapta-lo a novas mudanças. O software deve apontar as possíveis causas de falhas quando elas ocorrerem, ser fácil de realizar modificações e adaptações sem por em risco sua estabilidade. Contudo a conformidade desta política será avaliada pelos seguintes requisitos:

2.3.1 RNF Manutenibilidade

A manutenibilidade é representada por um conjunto de requisito de qualidade que contribuem positivamente para sua satisfação. Dentre este requisitos devem está evidentes durante a fase de implementação: “Analisabilidade”, “Modificabilidade”, “Estabilidade”, “Testabilidade”, “reutilização e comentário de código” e “Disponibilidade da Documentação”

2.3.2 RNF Flexibilidade

A flexibilidade se refere a capacidade de alterar a funcionalidade presente, sem conseqüências imprevistas sobre o conjunto da estrutura. Um requisito que contribui para sua satisfação quando evidenciado é a “Independência dos dados do paciente”.

2.3.3 RNF Portabilidade

A portabilidade se refere a capacidade do software de ser transferido de um ambiente para outro. Contudo esta política busca a evidência de requisitos que contribuem para “Adaptabilidade e Coexistência” do software.

2.4 POLÍTICA DA QUALIDADE DE PADRONIZAÇÃO

A política de padronização tem a finalidade de facilitar a integração do software com os diferentes sistemas de informação já existentes, plataformas computacionais, sistemas operacionais, linguagens de programação e redes de comunicação sem a qual é impossível a sua integração efetiva. Contudo foram definidos os seguintes RNFs para garantir estes desejos:

2.4.1 RNF Padrões de Intercomunicação

A comunicação do software com outros sistemas de informações pode ser facilitada através do uso de padrões de Intercomunicação. Neste sentido, a política da qualidade padronização busca por evidência do uso de protocolos padronizados que possibilite a uniformização dos dados transmitidos como por exemplo: “XML”, “HL7”, X12, EDIFACT, “DICOM” etc

2.4.2 RNF Padrões de Vocabulário

Como foi dito anteriormente, a padronização do vocabulário médico é fundamental para reunir informação clínica no cuidado ao paciente, para recuperar informação no manejo da doença ou para pesquisa, assim como para conduzir a análise de resultados. Neste sentido, esta política busca por evidência do uso de padrões para identificar as informações médicas como por exemplo: SONMED, CID-10, UMLS, LOINC.

2.4.3 RNF Padrões de Interfaces

Este RNF é usado para definir as interfaces que devem ser suportadas pela aplicação, assim como as soluções para os problemas comuns no projeto da interface do sistemas. Contudo, esta política busca por evidência do uso de padrões para interface do usuário, de hardware e de software”. Como exemplo temos: GUIs/Macintosh, Windows, “Clareza nos formatos das telas de entradas e relatórios”, “Uso de terminologia condizente com a prática médica” e “uso de padrões de interoperabilidade e portabilidade” como CORBamed.

2.4.4 RNF Padrões de Segurança

Este RNF é usado para verificar a existência do uso de padrões de segurança para o software como por exemplo: XML, XKMS, SAML

2.4.5 RNF Padrões de Formatação e Conteúdo

Este RNF é proposto pela política de padronização para avaliar a evidência dos padrões de conteúdo para o software médico. Por exemplo: ASTM, GEHR, RIPS/PRC.

Figura 20 - Política de Qualidade para o Software

Seguindo com o estudo de caso, após a criação da política de qualidade, a última fase da metodologia pode ser aplicada, ou seja, a fase de avaliação de conformidades. Para não estender o estudo de caso, exemplificaremos as três atividades desta fase, usando apenas o RNF Primário *Confiabilidade*, os demais RNFs Primários mostraremos os resultados já calculados. Primeiramente, priorizamos os *RNFs Secundários e Específicos* atribuindo pesos a estes requisitos de acordo com a atividade *Associar Pesos aos RNFs Secundários e Específicos*.

Política: 1. Qualidade Funcional					
RNF Primário: 1.1 RNF Confiabilidade			Categoria: Especificação Forte		
Conformidade Obtida:					
Decomposições			Peso	Conformidade	
Código	Denominação			Cálculo	Nota
1.1.1	RNF Persistência dos dados		1		
1.1.2	RNF Completeza		2		
1.1.3	RNF Precisão		2		
1.1.4	RNF Consistência dos dados		1		
1.1.5	RNF Tolerância a falhas		2		
1.1.5.1	RNF Recuperabilidade em caso de falhas		7		
1.1.5.1.1	RNF Capacidade de reverter funções de efeito drástico		10		
1.1.5.2	RNF Capacidade de interromper um processamento demorado		3		
1.1.6	RNF Maturidade		2		
1.1.6.1	RNF Prevenção e contingência		10		

Tabela 4 - Associação de Pesos aos RNFs de Confiabilidade

Em seguida, obedecendo as recomendações da atividade *Atribuir Notas aos RNFs Específicos*, atribuímos os notas aos *RNFs Específicos*.

Política: 1. Qualidade Funcional					
RNF Primário: 1.1 RNF Confiabilidade			Categoria: Especificação Forte		
Conformidade Obtida:					
Decomposições			Peso	Conformidade	
Código	Denominação			Cálculo	Nota
1.1.1	RNF Persistência dos dados		1	Valor atribuído	3
1.1.2	RNF Completeza		2	Valor atribuído	10
1.1.3	RNF Precisão		2	Valor atribuído	10
1.1.4	RNF Consistência dos dados		1	Valor atribuído	6,5
1.1.5	RNF Tolerância a falhas		2		
1.1.5.1	RNF Recuperabilidade em caso de falhas		7		
1.1.5.1.1	RNF Capacidade de reverter funções de efeito drástico		10	Valor atribuído	3

1.1.5.2	RNF Capacidade de interromper um processamento demorado	3	Valor atribuído	6,5
1.1.6	RNF Maturidade	2		
1.1.6.1	RNF Prevenção e contingência	10	Valor Atribuído	6,5

Tabela 5 - Atribuição de Notas aos RNFs Específicos do RNF Confiabilidade

Finalmente, executamos a atividade *Calcular Notas dos RNFs Primários e Secundários* e aplicamos a fórmula da Figura 17, para calcular os notas dos *RNFs Primários e Secundários*.

Política: 1 Qualidade Funcional				
RNF Primário: 1.1 RNF Confiabilidade		Categoria: Especificação Forte		
Conformidade Obtida: $(1*3+2*10+2*10+1*6,5+2*3,9+2*6,5)/10 = 7,04$ ou 70%				
Decomposições		Pesos	Conformidade	
Código	Denominação		Cálculo	Nota
1.1.1	RNF Persistência dos dados	1	Valor atribuído	3
1.1.2	RNF Completeza	2	Valor atribuído	10
1.1.3	RNF Precisão	2	Valor atribuído	10
1.1.4	RNF Consistência dos dados	1	Valor atribuído	6,5
1.1.5	RNF Tolerância a falhas	2	$(7*3+3*6,5)/10$	3,9
1.1.5.1	RNF Recuperabilidade em caso de falhas	7	$(10*3)/10$	3
1.1.5.1.1	RNF Capacidade de reverter funções de efeito drástico	10	Valor atribuído	3
1.1.5.2	RNF Capacidade de interromper um processamento demorado	3	Valor atribuído	6,5
1.1.6	RNF Maturidade	2	$(6,5*10)/10$	6,5
1.1.6.1	RNF Prevenção e contingência	10	Valor atribuído	6,5

Tabela 6 - Cálculo da Conformidade da definição do RNF Confiabilidade

Após repetir as três atividades para os demais RNFs Primários elicitados, obtivemos os seguintes resultados para cada um desses requisitos:

Política: 1. Qualidade Funcional				
RNF Primário: 1.2 RNF Usabilidade		Categoria: Especificação Média		
Conformidade Obtida: $(3*6,5+3*3,4+2*6,4+2*4)/10 = 50,5$ ou 50%				
Decomposições		Peso	Conformidade	
Código	Denominação		Cálculo	Nota
1.2.1	RNF Aspectos de aquisição de dados	3	$(2*10+2*6,5+2*6,5+2*6,5+2*3)/10$	6,5
1.2.1.1	RNF Identificação única para o paciente	2	Valor atribuído	10
1.2.1.2	RNF Importação de dados externos	2	Valor atribuído	6,5
1.2.1.3	RNF Uso de processos inteligentes	2	$(10*6,5)/10$	6,5
1.2.1.3.1	RNF Reconhecimento de textos, sons e imagens	10	Valor atribuído	6,5
1.2.1.4	RNF Aquisição de textos, sinais,	2	Valor atribuído	6,5

	imagens, sons e código barra			
1.2.1.5	RNF Estrutura do prontuário do paciente	2	Valor atribuído	3
1.2.2	RNF Aspectos de saída	3	$(1*6,5+2*6,5+1*3+2*3+2*0+2*3)/10$	3,4
1.2.2.1	RNF Busca por diversas informações	1	$(10*6,5)/10$	6,5
1.2.2.1.1	RNF Apresentação uniforme dos dados do paciente	10	Valor atribuído	6,5
1.2.2.2	RNF Exporta dados do paciente para outros sistemas	2	Valor atribuído	6,5
1.2.2.3	RNF Saída dos resultados em vídeo ou impressora	1	$(10*3)/10$	3
1.2.2.3.1	RNF Elaboração de estatística com geração automática de gráficos	10	Valor atribuído	3
1.2.2.4	RNF Utiliza código de barra	2	Valor atribuído	3
1.2.2.5	RNF Assinaturas Digitais dos relatórios	2	Valor atribuído	0
1.2.2.6	RNF Busca de imagens por conteúdo.	2	Valor atribuído	3
1.2.3	RNF Aspectos de Ajuda	2	$(4*6,5+3*6,8+3*5,9)/10$	6,4
1.2.3.1	RNF Ajuda on-line sensível ao contexto	4	Valor atribuído	6,5
1.2.3.2	RNF Manual do usuário atualizado	3	$(5*6,5+2*3+3*10)/10$	6,8
1.2.3.2.1	RNF Documentação do usuário de fácil compreensão	5	Valor atribuído	6,5
1.2.3.2.2	RNF Sumário e índice remissivo na documentação do usuário	2	Valor atribuído	3
1.2.3.2.3	RNF Presença de Manual de instalação com instruções detalhadas	3	Valor atribuído	10
1.2.3.3	RNF Mensagens auto explicativas	3	$(7*6,5+3*4,7)/10$	5,9
1.2.3.3.1	RNF Consistência de vocabulário entre as mensagens e a documentação	7	Valor atribuído	6,5
1.2.3.3.2	RNF Diferentes tipos de mensagens	3	$(5*3+5*6,5)/10$	4,7
1.2.3.3.2.1	RNF Mensagens de erro com informações necessárias para a solução da situação de erro	5	Valor atribuído	3
1.2.3.3.2.2	RNF Alertas claros para as conseqüências de uma determinada confirmação.	5	Valor atribuído	6,5
1.2.4	RNF Aspectos Gerais de Usabilidade	2	$(3*6,5+4*3+3*3)/10$	4
1.2.4.1	RNF Aprendibilidade	3	Valor atribuído	6,5
1.2.4.2	RNF Operacionalidade	4	Valor atribuído	3
1.2.4.3	RNF Atratividade	3	Valor atribuído	3

Tabela 7 - Cálculo da Conformidade da definição do RNF Usabilidade

Política: 1. Qualidade Funcional				
RNF Primário: 1.3 RNF Segurança		Categoria: Especificação Forte		
Conformidade Obtida: $(2*10+2*10+1*7,2+3*6,8+2*4,7)/10 = 7,7$ ou 77%				
Decomposições		Peso	Conformidade	
Código	Denominação		Cálculo	Nota
1.3.1	RNF Mecanismos de administração de usuários centralizado	2	$(10*10)/10$	10
1.3.1.1	RNF Registro das autorizações de acesso	10	Valor atribuído	10
1.3.2	RNF Política de controle de acesso	2	$(5*10+5*10)/10$	10
1.3.2.1	RNF Autenticação	5	Valor atribuído	10
1.3.2.2	RNF Identificação única de usuários	5	Valor atribuído	10
1.3.3	RNF Políticas de senhas	1	$(1*6,5+3*6,5+1*3+2*6,5+3*10)/10$	7,2
1.3.3.1	RNF Limitar a existência de senha.	1	Valor atribuído	6,5
1.3.3.2	RNF Atualização periódica de senhas	3	Valor atribuído	6,5
1.3.3.3	RNF Não reutilização de senhas antigas	1	Valor atribuído	3
1.3.3.4	RNF Proibir a auditoria de informação relativa a senhas	2	Valor atribuído	6,5
1.3.3.5	RNF Manipular senhas criptografadas	3	Valor atribuído	10
1.3.4	RNF Disponibilidade	3	$(3*6,5+4*10+3*3)/10$	6,8
1.3.4.1	RNF Redundância de hardware e software	3	Valor atribuído	6,5
1.3.4.2	RNF Cópia de Segurança	4	Valor atribuído	10
1.3.4.3	RNF Auditoria	3	Valor atribuído	3
1.3.5	RNF Segurança do Ambiente	2	$(5*6,5+5*3)/10$	4,7
1.3.5.1	RNF Restringir acesso a sala dos servidores	5	Valor atribuído	6,5
1.3.5.2	RNF Proteção contra incêndios e inundações	5	Valor atribuído	3

Tabela 8 - Cálculo da Conformidade da definição do RNF Segurança

Política: 2. Qualidade Desempenho				
RNF Primário: 2.1 RNF Eficiência		Categoria: Especificação Forte		
Conformidade Obtida: $(5*6,5+5*10)/10 = 8,2$ ou 82%				
Decomposições		Peso	Conformidade	
Código	Denominação		Cálculo	Nota
2.1.1	RNF Eficiência de execução	5	$(10*6,5)/10$	6,5
2.1.1.1	RNF Busca de imagens não pode ultrapassar 3 segundos	10	$(10*6,5)/10$	6,5
2.1.1.1.1	RNF Usar indexação de imagens medicas	10	Valor atribuído	6,5
2.1.2	RNF Eficiência de Armazenamento	5	$(10*10)/10$	10
2.1.2.1	RNF Armazenar imagens compactadas	10	Valor atribuído	10

Tabela 9 - Cálculo da Conformidade da definição do RNF Eficiência

Política: 2. Qualidade Desempenho				
RNF Primário: 2.2 RNF Interoperabilidade		Categoria: Especificação Média		
Conformidade Obtida: $(6*3+4*6,5)/10 = 4,4$ ou 44%				
Decomposições		Peso	Conformidade	
Código	Denominação		Cálculo	Nota
2.2.1	RNF Interoperabilidade Técnica	6	$(10*3)/10$	3
2.2.1.1	RNF Compatibilidade	10	Valor atribuído	3
2.2.2	RNF Interoperabilidade Semântica	4	$(10*6,5)/10$	6,5
2.2.2.1	RNF Inteligibilidade	10	Valor atribuído	6,5

Tabela 10 - Cálculo da Conformidade da definição do RNF Interoperabilidade

Política: 3. Qualidade Manutenção				
RNF Primário: 3.1 RNF Manutenibilidade		Categoria: Especificação Forte		
Conformidade Obtida: $(2*3+2*6,5+4*6,5+2*3)/10 = 5,1$ ou 51%				
Decomposições		Peso	Conformidade	
Código	Denominação		Cálculo	Nota
3.1.1	RNF Analisabilidade	2	Valor atribuído	3
3.1.2	RNF Modificabilidade	2	Valor atribuído	6,5
3.1.3	RNF Estabilidade	4	Valor atribuído	6,5
3.1.4	RNF Testabilidade	2	Valor atribuído	3

Tabela 11 - Cálculo da Conformidade da definição do RNF Manutenibilidade

Política: 3. Qualidade Manutenção				
RNF Primário: 3.2 RNF Flexibilidade		Categoria: Especificação Média		
Conformidade Obtida: $(10*10)/10 = 10$ ou 100%				
Decomposições		Peso	Conformidade	
Código	Denominação		Cálculo	Nota
3.2.1	RNF Independência dos dados do paciente	10	Valor atribuído	10

Tabela 12 - Cálculo da Conformidade da definição do RNF Flexibilidade

Política: 3. Qualidade Manutenção				
RNF Primário: 3.3 RNF Portabilidade		Categoria: Especificação Fraca		
Conformidade Obtida: $(3*6,5+7*6,5)/10 = 6,5$ ou 65%				
Decomposições		Peso	Conformidade	
Código	Denominação		Cálculo	Nota
3.3.1	RNF Adaptabilidade	3	Valor atribuído	6,5
3.3.2	RNF Coexistência	7	Valor atribuído	6,5

Tabela 13 - Cálculo da Conformidade da definição do RNF Portabilidade

Política: 4. Qualidade Padronização				
RNF Primário: 4.1 RNF Padrões de Intercomunicação		Categoria: Especificação Forte		
Conformidade Obtida: $(5*6,5+5*10)/10 = 8,2$ ou 82%				
Decomposições		Peso	Conformidade	
Código	Denominação		Cálculo	Nota
4.1.1	RNF Utilizar protocolos padronizados	5	Valor atribuído	6,5
4.1.2	RNF Uniformidade do formato dos dados transmitidos	5	Valor atribuído	10

Tabela 14 - Cálculo da Conformidade da definição do RNF Padrões de Intercomunicação

Política: 4. Qualidade Padronização				
RNF Primário: 4.2 RNF Padrões de Vocabulário		Categoria: Especificação Forte		
Conformidade Obtida: $(10*10)/10 = 10$ ou 100%				
Decomposições		Peso	Conformidade	
Código	Denominação		Cálculo	Nota
4.2.1	RNF Utilizar padrões para identificar as informações	10	Valor atribuído	10

Tabela 15 - Cálculo da Conformidade da definição do RNF Padrões de Vocabulário

Política: 4. Qualidade Padronização				
RNF Primário: 4.3 RNF Padrões de Interfaces		Categoria: Especificação Forte		
Conformidade Obtida: $(4*6,5+2*10+4*10)/10 = 8,6$ ou 86%				
Decomposições		Peso	Conformidade	
Código	Denominação		Cálculo	Nota
4.3.1	RNF Interface do usuário	4	$(5*6,5+5*6,5)/10$	6,5
4.3.1.1	RNF Usa padrões gráficos de interfaces	5	$(10*6,5)/10$	6,5
4.3.1.1.1	RNF Clareza nos formatos das telas de entrada e relatórios	10	Valor atribuído	6,5
4.3.1.2	RNF Utiliza terminologia condizente com a prática médica	5	Valor atribuído	6,5
4.3.2	RNF Interface de hardware	2	Valor atribuído	10
4.3.3	RNF Interface de Software	4	$(10*10)/10$	10
4.3.3.1	RNF Prover padrões de interoperabilidade e portabilidade	10	Valor atribuído	10

Tabela 16 - Cálculo da Conformidade da definição do RNF Padrões de Interfaces

Política: 4. Qualidade Padronização				
RNF Primário: 4.4 RNF Padrões de Segurança		Categoria: Especificação Forte		
Conformidade Obtida: $(10*6,5)/10 = 6,5$ ou 65%				
Decomposições		Peso	Conformidade	
Código	Denominação		Cálculo	Nota
4.4.1	RNF Padrões de Segurança para web	10	$(10*6,5)/10$	6,5
4.4.1.1	RNF Padrões XML	10	Valor atribuído	6,5

Tabela 17 - Cálculo da Conformidade da definição do RNF Padrões de Segurança

Política: 4. Qualidade Padronização				
RNF Primário: 4.5 RNF Padrões de Formatação e Conteúdo				
Categoria: Especificação Forte				
Conformidade Obtida: $(10 * 6,5) / 10 = 6,5$ ou 65 %				
Decomposições		Peso	Conformidade	
Código	Denominação		Cálculo	Nota
4.5.1	RNF Usar padrões de conteúdo	10	Atribuída	6,5

Tabela 18 - Cálculo da Conformidade da definição do RNF Padrões de Formatação e Conteúdo.

Ao concluirmos o cálculo da conformidade da definição dos *RNFs Primários*, partimos agora para a atividade *Identificar Obediências e Contradições*, a qual consiste em estabelecer os relacionamentos entre a conformidade dos *RNFs Primários* baseada no documento de requisitos e a conformidade estabelecida na política de qualidade.

Necessidades de Qualidade	RNFs Primários	Conformidade		Resultado
		Política	DRs	
Funcional	RNF Confiabilidade	80%	70%	X
	RNF Usabilidade	65%	50%	X
	RNF Segurança	80%	77%	X
Desempenho	RNF Eficiência	80%	82%	✓
	RNF Interoperabilidade	65%	44%	X
Manutenção	RNF Manutenibilidade	80%	51%	X
	RNF Flexibilidade	65%	100%	✓
	RNF Portabilidade	30%	65%	✓
Padronização	RNF Padrões de Intercomunicação	80%	82%	✓
	RNF Padrões de Vocabulário	80%	100%	✓
	RNF Padrões de Interfaces	80%	86%	✓
	RNF Padrões de Segurança	80%	65%	X
	RNF Padrões de Formatação e Conteúdo	80%	65%	X

Tabela 19 – Resultado da conformidade dos RNFs Primários

Como pode se observar na tabela de resultado das conformidades, os resultados relacionados com o símbolo “✓” significa que as especificações dos RNFs definidas no documento de requisitos está em conformidade com a política de qualidade. Por outro lado, as especificações definidas no documento de requisitos para os RNFs que estão com os resultados relacionados com “X” não foram suficientes para atender a política de qualidade. Isso significa que é preciso rever o processo de especificação desses requisitos ou atualizar a política de qualidade. Após a conclusão desta fase, todas as planilhas devem ser atualizadas com as conformidades obtidas.

Ao concluir todas as fases da metodologia já teremos as planilhas com suas informações atualizadas:

Tipo: RNF Primário	
Necessidade: Qualidade Funcional	
Código: 1.1	Título: RNF Confiabilidade
Categoria: Especificação Forte	Cenários: C1,C10 e C16
Quantidade de Filhos: 6	
Definição: é capacidade do software de manter seu nível de desempenho quando utilizado em condições especificadas.	
Decomposições: 1.1.1 RNF Persistência dos dados, 1.1.2 RNF Completeza , 1.1.3 RNF Precisão , 1.1.4 RNF Consistência dos dados, 1.1.5 RNF Tolerância a falhas, 1.1.6 RNF Maturidade.	
Influências Positivas (não identificadas)	
Influências Negativas (não identificadas)	
Conformidade Desejada: 80%	Conformidade Obtida: 70%

Planilha 1 – Planilha do RNF Primário Confiabilidade

Tipo: RNF Primário	
Necessidade: Qualidade Funcional	
Código: 1.2	Título: RNF Usabilidade
Categoria: Especificação Forte	Cenários: C11, C16, C17
Quantidade de Filhos: 4	
Definição: é a capacidade do software de ser entendível, aprendido e usado de forma atrativa pelos usuários, quando utilizado sob condições especificadas.	
Decomposições: 1.2.1 RNF Aspectos de aquisição de dados, 1.2.2 RNF Aspectos de saída, 1.2.3 RNF Aspectos de Ajuda, 1.2.4 RNF Aspectos Gerais de Usabilidade	
Influências Positivas: (não identificadas)	
Influências Negativas: 1.3 RNF Segurança	
Conformidade Desejada: 65%	Conformidade Obtida: 50%

Planilha 2 – Planilha do RNF Primário Usabilidade

Tipo: RNF Primário	
Necessidade: Qualidade Funcional	
Código: 1.3	Título: RNF Segurança
Categoria: Especificação Forte	Cenários: C12, C13
Quantidade de Filhos: 6	
Definição: é a capacidade do software de evitar acesso não autorizado aos dados.	
Decomposições: 1.3.1 RNF Mecanismos de administração de usuários centralizados, 1.3.2 RNF Política de controle de acesso, 1.3.3 RNF Políticas de senhas, 1.3.4 RNF Disponibilidade, 1.3.5 RNF Segurança do ambiente.	
Influências Positivas (não identificadas)	
Influências Negativas: 1.2 RNF Usabilidade, 3.2 RNF Flexibilidade	
Conformidade Desejada: 80%	Conformidade Obtida: 77%

Planilha 3 – Planilha do RNF Primário Segurança

Tipo: RNF Primário	
Necessidade: Qualidade Desempenho	
Código: 2.1	Título: RNF Eficiência
Categoria: Especificação Forte	Cenários: C14
Quantidade de Filhos: 2	
Definição: é o relacionamento entre o nível de desempenho do software e a quantidade de recursos utilizados, sob condições especificadas.	
Decomposições: 2.1.1 RNF Eficiência de execução, 2.1.2 RNF Eficiência de Armazenamento.	
Influências Positivas (não identificadas)	
Influências Negativas (não identificadas)	
Conformidade Desejada: 80%	Conformidade Obtida: 82%

Planilha 4 – Planilha do RNF Primário Eficiência

Tipo: RNF Primário	
Necessidade: Qualidade Funcional	
Código: 2.2	Título: RNF Interoperabilidade
Categoria: Especificação Forte	Cenários: C18, C19
Quantidade de Filhos: 2	
Definição: é a capacidade do software de interagir com um ou mais sistemas.	
Decomposições: 2.2.1 RNF Interoperabilidade Técnica, 2.2.2 RNF Interoperabilidade semântica.	
Influências Positivas 4.3 RNF Padrões de Interfaces	
Influências Negativas (não identificadas)	
Conformidade Desejada: 65%	Conformidade Obtida: 44%

Planilha 5 – Planilha do RNF Primário Interoperabilidade

Tipo: RNF Primário	
Necessidade: Qualidade Manutenção	
Código: 3.1	Título: RNF Manutenibilidade
Categoria: Especificação Forte	Cenários:
Quantidade de Filhos: 4	
Definição: é o esforço necessário para se fazer modificações especificadas no software.	
Decomposições: 3.1.1 RNF Analisabilidade, 3.1.2 RNF Modificabilidade, 3.1.3 RNF Estabilidade, 3.1.4 RNF Testabilidade.	
Influências Positivas 1.2.3.2 RNF Manual do usuário atualizado	
Influências Negativas (não identificadas)	
Conformidade Desejada: 80%	Conformidade Obtida: 51%

Planilha 6 – Planilha do RNF Primário Manutenibilidade

Tipo: RNF Primário	
Necessidade: Qualidade Manutenção	
Código: 3.2	Título: RNF Flexibilidade
Categoria: Especificação Forte	Cenários:
Quantidade de Filhos: 1	
Definição: é a possibilidade de minimizar custos, considerando o ciclo de vida do produto de software, a medida em que permite adequações funcionais de acordo com novas necessidades.	
Decomposições: 3.2.1 RNF Independência dos dados do paciente	
Influências Positivas (não identificadas)	
Influências Negativas: 1.3 RNF Segurança	
Conformidade Desejada: 65%	Conformidade Obtida: 100%

Planilha 7 – Planilha do RNF Primário Flexibilidade

Tipo: RNF Primário	
Necessidade: Qualidade Manutenção	
Código: 3.3	Título: RNF Portabilidade
Categoria: Especificação Forte	Cenários:
Quantidade de Filhos: 2	
Definição: é a capacidade do software de ser instalado em vários ambientes computacionais.	
Decomposições: 3.3.1 RNF Adaptabilidade, 3.3.2 RNF Coexistência	
Influências Positivas (não identificadas)	
Influências Negativas (não identificadas)	
Conformidade Desejada: 30%	Conformidade Obtida: 65%

Planilha 8 – Planilha do RNF Primário Portabilidade

Tipo: RNF Primário	
Necessidade: Qualidade Padronização	
Código: 4.1	Título: RNF Padrões de Intercomunicação
Categoria: Especificação Forte	Cenários: C18, C19
Quantidade de Filhos: 2	
Definição: é a capacidade do software em garantir que o formato dos dados transmitidos e os protocolos utilizados sejam compatíveis com padrões existentes para transmissão de dados relacionados a área da saúde.	
Decomposições: 4.1.1 RNF Utilizar protocolos padronizados, 4.1.2 RNF Uniformidade do formato dos dados transmitidos	
Influências Positivas (não identificadas)	
Influências Negativas (não identificadas)	
Conformidade Desejada: 80%	Conformidade Obtida: 82%

Planilha 9 – Planilha do RNF Primário Padrões de Intercomunicação

Tipo: RNF Primário	
Necessidade: Qualidade Padronização	
Código: 4.2	Título: RNF Padrões de Vocabulário
Categoria: Especificação Forte	Cenários: C12
Quantidade de Filhos: 1	
Definição: é a garantia de que o software utiliza padrões de vocabulário já conhecido pelos profissionais da área da saúde. Isto facilita o uso e o entendimento do software.	
Decomposições: 4.2.1 RNF Utilizar padrões para identificar informações.	
Influências Positivas (não identificadas)	
Influências Negativas (não identificadas)	
Conformidade Desejada: 80%	Conformidade Obtida: 100%

Planilha 10 – Planilha do RNF Primário Padrões de Vocabulário

Tipo: RNF Primário	
Necessidade: Qualidade Padronização	
Código: 4.3	Título: RNF Padrões de Interfaces
Categoria: Especificação Forte	Cenários: C18, C19
Quantidade de Filhos: 4	
Definição: é a capacidade do software de prover interfaces de usuário de hardware e de software compatíveis com os padrões especificados	
Decomposições: 4.3.1 RNF Interface do usuário, 4.3.2 RNF Interface de hardware, 4.3.3 RNF Interface de software	
Influências Positivas (não identificadas)	
Influências Negativas (não identificadas)	
Conformidade Desejada: 80%	Conformidade Obtida: 86%

Planilha 11 – Planilha do RNF Primário Padrões de Interfaces

Tipo: RNF Primário	
Necessidade: Qualidade Padronização	
Código: 4.4	Título: RNF Padrões de Segurança
Categoria: Especificação Forte	Cenários:
Quantidade de Filhos: 1	
Definição: é a capacidade do software de prover a segurança do sistema através da utilização de padrões especificados para este fim.	
Decomposições: 4.4.1 RNF Padrões de segurança para web.	
Influências Positivas (não identificadas)	
Influências Negativas (não identificadas)	
Conformidade Desejada: 80%	Conformidade Obtida: 65%

Planilha 12 – Planilha do RNF Primário Padrões de Segurança

Tipo: RNF Primário	
Necessidade: Qualidade Padronização	
Código: 4.5	Título: RNF Padrões de Formatação e Conteúdo
Categoria: Especificação Forte	Cenários: C18, C19
Quantidade de Filhos: 1	
Definição: é a capacidade do software de manipular dados com o conteúdo e formatação de acordo com os padrões recomendados para o domínio.	
Decomposições: 4.5.1 RNF usar padrões de conteúdo	
Influências Positivas (não identificadas)	
Influências Negativas (não identificadas)	
Conformidade Desejada: 80%	Conformidade Obtida: 65%

Planilha 13 – Planilha do RNF Primário Padrões de Formatação e conteúdo

A seguir as planilhas de Requisitos não Funcionais Secundários serão apresentadas:

Tipo: RNF Secundário	
Código: 1.1.5	Título: RNF Tolerância a falhas
Quantidade de Filhos: 2	Peso: 2 Nota: 3,9
Decomposto de: 1.1 RNF Confiabilidade	
Decomposições: 1.1.5.1 RNF Recuperabilidade em caso de falha, 1.1.5.2 RNF Capacidade de interromper um processamento demorado	
Definição: é a capacidade do software de manter um nível de desempenho estabelecido em caso de defeito no software	
Influências Positivas (não identificadas)	
Influências Negativas (não identificadas)	

Planilha 14 – Planilha do RNF Secundário Tolerância a falhas

Tipo: RNF Secundário		
Código: 1.1.5.1	Título: RNF Recuperabilidade em caso de falha	
Quantidade de Filhos: 1	Peso: 7	Nota: 3
Decomposto de: 1.1.5 RNF Tolerância a falhas		
Decomposições: 1.1.5.1.1 RNF Capacidade de reverter funções de efeito drástico		
Definição: é a capacidade do software recuperar dados diretamente afetados em caso de falhas ou retornar ao seu ultimo estado antes de ocorrer a falha.		
Influências Positivas (não identificadas)		
Influências Negativas (não identificadas)		

Planilha 15 – Planilha do RNF Secundário Recuperabilidade em caso de falha

Tipo: RNF Secundário		
Código: 1.1.6	Título: RNF Maturidade	
Quantidade de Filhos: 1	Peso: 2	Nota: 6,5
Decomposto de: 1.1 RNF Confiabilidade		
Decomposições: 1.1.6.1 RNF Prevenção e contingência		
Definição: é a capacidade do software de evitar defeitos		
Influências Positivas (não identificadas)		
Influências Negativas (não identificadas)		

Planilha 16 – Planilha do RNF Secundário Maturidade

Tipo: RNF Secundário		
Código: 1.2.1	Título: RNF Aspectos de aquisição de dados	
Quantidade de Filhos: 5	Peso: 3	Nota: 6,5
Decomposto de: 1.2 RNF Usabilidade		
Decomposições: 1.2.1.1 RNF Identificação única para o paciente, 1.2.1.2 RNF Importação de dados externos, 1.2.1.3 RNF Uso de processos inteligentes, 1.2.1.4 RNF Aquisição de textos, sinais, imagens, sons e código barra, 1.2.1.5 RNF Estrutura do prontuário do paciente		
Definição: é a capacidade do software de ser entendível, aprendido e usado quanto as funcionalidades de aquisição de dados		
Influências Positivas (não identificadas)		
Influências Negativas (não identificadas)		

Planilha 17 – Planilha do RNF Secundário Aspectos de aquisição de dados

Tipo: RNF Secundário		
Código: 1.2.1.3	Título: RNF Uso de processos inteligentes	
Quantidade de Filhos: 1	Peso: 2	Nota: 6,5
Decomposto de: 1.2.1 RNF Aspectos de aquisição de dados		
Decomposições: 1.2.1.3.1 RNF reconhecimento de textos sons e imagens		
Definição: é a capacidade do software de manipular, dados complexos, como textos, sons e imagens.		
Influências Positivas (não identificadas)		
Influências Negativas (não identificadas)		

Planilha 18 – Planilha do RNF Secundário Uso de processos inteligentes

Tipo: RNF Secundário		
Código: 1.2.2	Título: RNF Aspectos de saída	
Quantidade de Filhos: 6	Peso: 3	Nota: 3,4
Decomposto de: 1.2 RNF Usabilidade		
Decomposições: 1.2.2.1 RNF Busca por diversas informações, 1.2.2.2 RNF Exporta dados do paciente para outros sistemas, 1.2.2.3 RNF Saída dos resultados em vídeo ou impressora, 1.2.2.4 RNF Utiliza código de barra, 1.2.2.5 RNF Assinaturas Digitais dos relatórios, 1.2.2.6 RNF Busca de imagens por conteúdo.		
Definição: é a evidencia de um conjunto de características do software capaz de facilitar seu entendimento, aprendizado e uso quanto as suas funcionalidade de saída tais como: consultas, relatórios e exportação de dados.		
Influências Positivas (não identificadas)		
Influências Negativas (não identificadas)		

Planilha 19 – Planilha do RNF Secundário Aspectos de saída

Tipo: RNF Secundário		
Código: 1.2.2.1	Título: RNF Busca por diversas informações	
Quantidade de Filhos: 1	Peso: 1	Nota: 6,5
Decomposto de: 1.2 2 RNF Aspectos de saída		
Decomposições: 1.2.2.1.1 RNF Apresentação uniforme dos dados do paciente		
Definição: é a capacidade do software de viabilizar diferentes formas de consultar os dados relativos ao prontuário do paciente.		
Influências Positivas (não identificadas)		
Influências Negativas (não identificadas)		

Planilha 20 – Planilha do RNF Secundário Busca por diversas informações

Tipo: RNF Secundário		
Código: 1.2.2.3	Título: RNF Saída dos resultados em vídeo ou impressora	
Quantidade de Filhos: 1	Peso: 1	Nota: 3
Decomposto de: 1.2 2 RNF Aspectos de saída		
Decomposições: 1.2.2.3.1 RNF Elaboração de estatística com geração automática de gráficos;		
Definição: é a capacidade do software de prover funcionalidades que possibilite a saída dos dados em impressoras ou vídeos.		
Influências Positivas (não identificadas)		
Influências Negativas (não identificadas)		

Planilha 21 – Planilha do RNF Secundário Saída dos resultados em vídeo ou impressora

Tipo: RNF Secundário		
Código: 1.2.3	Título: RNF Aspectos de Ajuda	
Quantidade de Filhos: 3	Peso: 2	Nota: 6,4
Decomposto de: 1.2 RNF Usabilidade		
Decomposições: 1.2.3.1 RNF Ajuda on-line sensível ao contexto, 1.2.3.2 RNF Manual do usuário atualizado, 1.2.2.3 RNF Saída dos resultados em vídeo ou impressora		
Definição: é a capacidade do software de ser entendível, aprendido e usado quanto as suas formas de ajuda.		
Influências Positivas (não identificadas)		
Influências Negativas (não identificadas)		

Planilha 22 – Planilha do RNF Secundário Aspectos de ajuda

Tipo: RNF Secundário		
Código: 1.2.3.2	Título: RNF Manual do usuário atualizado	
Quantidade de Filhos: 3	Peso: 3	Nota: 6,8
Decomposto de: 1.2.3 RNF Aspectos de Ajuda		
Decomposições: 1.2.3.2.1 RNF Documentação do usuário de fácil compreensão, 1.2.3.2.2 RNF Sumário e índice remissivo na documentação do usuário, 1.2.3.2.3 RNF Presença de Manual de instalação com instruções detalhadas		
Definição: o objetivo é garantir a existência de manuais e documentação atualizada para auxiliar o usuário a esclarecer dúvidas quanto ao uso do software.		
Influências Positivas (não identificadas)		
Influências Negativas (não identificadas)		

Planilha 23 – Planilha do RNF Secundário Manual do usuário atualizado

Tipo: RNF Secundário		
Código: 1.2.3.3	Título: RNF Mensagens auto explicativas	
Quantidade de Filhos: 2	Peso: 3	Nota: 5,9
Decomposto de: 1.2.3 RNF Aspectos de Ajuda		
Decomposições: 1.2.3.3.1 RNF Consistência de vocabulário entre as mensagens e a documentação, 1.2.3.3.2 RNF diferentes tipos de mensagens		
Definição: é a capacidade do software de emitir mensagens com informações suficientes para ajudar solucionar o problema, vocabulário consistente com a documentação e que ajude o usuário a descobrir o motivo pelo qual a mensagem está sendo exibida.		
Influências Positivas (não identificadas)		
Influências Negativas (não identificadas)		

Planilha 24 – Planilha do RNF Secundário Mensagens auto explicativas

Tipo: RNF Secundário		
Código: 1.2.3.3.2	Título: RNF diferentes tipos de mensagens	
Quantidade de Filhos: 2	Peso: 3	Nota: 4,7
Decomposto de: 1.2.3.3 RNF Mensagens auto explicativas		
Decomposições: 1.2.3.3.2.1 RNF Mensagens de erro com informações necessárias para a solução da situação de erro, 1.2.3.3.2.2 RNF Alertas claros para conseqüências de uma determinada confirmação.		
Definição: é a capacidade do software de emitir diferentes tipos de mensagens, confirmação, consulta, advertência, erro etc... para ajudar o usuário a entender melhor o motivo de cada mensagem		
Influências Positivas (não identificadas)		
Influências Negativas (não identificadas)		

Planilha 25 – Planilha do RNF Secundário Diferentes tipos de mensagens

Tipo: RNF Secundário		
Código: 1.2.4	Título: RNF Aspectos Gerais de Usabilidade	
Quantidade de Filhos: 3	Peso: 2	Nota: 4
Decomposto de: 1.2 RNF Usabilidade		
Decomposições: 1.2.4.1 RNF Aprendibilidade, 1.2.4.2 RNF Operacionalidade, 1.2.4.3 RNF Atratividade		
Definição: é a evidencia de um conjunto de características do software capaz de torná-lo atraente, fácil de ser controlado, aprendido e entendido por seus usuários.		
Influências Positivas (não identificadas)		
Influências Negativas (não identificadas)		

Planilha 26 – Planilha do RNF Secundário Aspectos Gerais de usabilidade

Tipo: RNF Secundário		
Código: 1.3.1 Título: RNF Mecanismos de administração de usuários centralizados.		
Quantidade de Filhos: 1	Peso: 2	Nota: 10
Decomposto de: 1.3 RNF Segurança.		
Decomposições: 1.3.1.1 RNF Registro das autorizações de acesso		
Definição: é a forma como os usuários são controlados e gerenciados.		
Influências Positivas (não identificadas)		
Influências Negativas (não identificadas)		

Planilha 27 – Planilha do RNF Secundário Mecanismos de administração de usuários

Tipo: RNF Secundário		
Código: 1.3.2 Título: RNF Política de controle de acesso		
Quantidade de Filhos: 2	Peso: 2	Nota: 10
Decomposto de: 1.3 RNF Segurança		
Decomposições: 1.3.2.1 RNF Autenticação, 1.3.2.2 RNF Identificação única de usuários.		
Definição: são regras criadas para controlar o acesso de usuários ao software.		
Influências Positivas (não identificadas)		
Influências Negativas (não identificadas)		

Planilha 28 – Planilha do RNF Secundário Política de controle de acesso

Tipo: RNF Secundário		
Código: 1.3.3	Título: RNF Política de senhas	
Quantidade de Filhos: 4	Peso: 1	Nota: 7,2
Decomposto de: 1.3 RNF Segurança		
Decomposições: 1.3.3.1 RNF Limitar a existência de senhas, 1.3.3.2 RNF Atualização periódica de senhas, 1.3.3.3 RNF Não reutilização de senhas antigas, 1.3.3.4 RNF Proibir auditoria de informações relativas a senhas.		
Definição: garantir que existe uma política de senha para conscientizar os usuários da melhor maneira de criar uma senha e a importância de se atualizar esta senha a cada vez que ela expirar.		
Influências Positivas (não identificadas)		
Influências Negativas (não identificadas)		

Planilha 29 – Planilha do RNF Secundário Política de senhas

Tipo: RNF Secundário		
Código: 1.3.4	Título: RNF Disponibilidade	
Quantidade de Filhos: 3	Peso: 3	Nota: 6,8
Decomposto de: 1.3 RNF Segurança		
Decomposições: 1.3.4.1 RNF Redundância de hardware e software, 1.3.4.2 RNF Cópias de segurança, 1.3.4.3 RNF Auditoria.		
Definição: é a capacidade do software de está sempre disponível aos usuários autorizados.		
Influências Positivas (não identificadas)		
Influências Negativas (não identificadas)		

Planilha 30 – Planilha do RNF Secundário Disponibilidade

Tipo: RNF Secundário		
Código: 1.3.5	Título: RNF Segurança do ambiente	
Quantidade de Filhos: 2	Peso: 2	Nota: 4,7
Decomposto de: 1.3 RNF Segurança		
Decomposições: 1.3.5.1 RNF Restringir acesso a sala dos servidores, 1.3.5.2 RNF Proteção contra incêndios e inundações.		
Definição: garantir que o acesso as dependências onde se localizam os servidores é restrito a pessoas autorizadas e que as cópias de segurança sejam protegidas contra incêndios e inundações.		
Influências Positivas (não identificadas)		
Influências Negativas (não identificadas)		

Planilha 31 – Planilha do RNF Secundário Segurança do ambiente

Tipo: RNF Secundário		
Código: 2.1.1	Título: RNF Eficiência de execução	
Quantidade de Filhos: 1	Peso: 5	Nota: 6,5
Decomposto de: 2.1 RNF Eficiência		
Decomposições: 2.1.1.1 RNF Busca de imagens não pode ultrapassar 3 segundos		
Definição: é a capacidade do software de responder aos eventos críticos atendendo aos requisitos de desempenho especificados.		
Influências Positivas (não identificadas)		
Influências Negativas (não identificadas)		

Planilha 32 – Planilha do RNF Secundário Eficiência de execução

Tipo: RNF Secundário		
Código: 2.1.1.1	Título: RNF Busca de imagens não pode ultrapassar 3 segundos	
Quantidade de Filhos: 1	Peso: 10	Nota: 6,5
Decomposto de: 2.1.1 RNF Eficiência de execução		
Decomposições: 2.1.1.1 RNF usar indexação de imagens.		
Definição: é a capacidade do software de consultar o banco de imagens e disponibilizar o resultado para o usuário num tempo não superior a 3 segundos.		
Influências Positivas (não identificadas)		
Influências Negativas (não identificadas)		

Planilha 33 – Planilha do RNF Secundário Busca de imagens não pode ultrapassar 3 segundos

Tipo: RNF Secundário		
Código: 2.1.2	Título: RNF Eficiência de Armazenamento	
Quantidade de Filhos: 1	Peso: 5	Nota: 10
Decomposto de: 2.1.RNF Eficiência		
Decomposições: 2.1.2.1 RNF Armazenar imagens compactadas		
Definição: é a capacidade do software de armazenar dados complexos, como sons, imagens etc.. em banco de dados eficientes que garantam a integridade e que possibilite a recuperação destes dados a qualquer momento.		
Influências Positivas (não identificadas)		
Influências Negativas (não identificadas)		

Planilha 34 – Planilha do RNF Secundário Eficiência de Armazenamento

Tipo: RNF Secundário		
Código: 2.2.1	Título: RNF Interoperabilidade Técnica	
Quantidade de Filhos: 2	Peso: 6	Nota: 3
Decomposto de: 2.2. RNF Interoperabilidade		
Decomposições: 2.2.1.1 RNF Compatibilidade.		
Definição: é a capacidade do software de prover interfaces de comunicação compatíveis com os outros sistemas especificados para troca de informações.		
Influências Positivas (não identificadas)		
Influências Negativas (não identificadas)		

Planilha 35 – Planilha do RNF Secundário Interoperabilidade Técnica

Tipo: RNF Secundário		
Código: 2.2.2	Título: RNF Interoperabilidade Semântica	
Quantidade de Filhos: 1	Peso: 4	Nota: 6,5
Decomposto de: 2.2. RNF Interoperabilidade		
Decomposições: 2.2.2.1 RNF Inteligibilidade.		
Definição: Capacidade do software de garantir que a informação compartilhada com outros sistemas seja reconhecida a partir da definição formal de conceitos do domínio.		
Influências Positivas (não identificadas)		
Influências Negativas (não identificadas)		

Planilha 36 – Planilha do RNF Secundário Interoperabilidade Semântica

Tipo: RNF Secundário		
Código: 4.3.1	Título: RNF Interface do usuário	
Quantidade de Filhos: 2	Peso: 4	Nota: 6,5
Decomposto de: 4.3 RNF Padrões de Interfaces		
Decomposições: 4.3.1.1 RNF Usar padrões gráficos de interfaces, 4.3.1.2 RNF Utilizar terminologia condizente com a prática médica.		
Definição: garantir que a interface do usuário seja construída utilizando padrões gráficos como GUO, Windows, Macintosh e que utilize uma terminologia condizente com a prática médica.		
Influências Positivas (não identificadas)		
Influências Negativas (não identificadas)		

Planilha 37 – Planilha do RNF Secundário Interface do usuário

Tipo: RNF Secundário

Código: 4.3.1.1 **Título:** RNF Usar padrões gráficos de interfaces

Quantidade de Filhos: 1 **Peso:** 5 **Nota:** 6,5

Decomposto de: 4.3.1 RNF Interface do usuário

Decomposições: 4.3.1.1.1 RNF Clareza nos formato das telas de entrada e relatórios.

Definição: é a capacidade do software de prover padrões gráficos de interfaces para facilitar sua usabilidade.

Influências Positivas (não identificadas)

Influências Negativas (não identificadas)

Planilha 38 – Planilha do RNF Secundário Usar padrões gráficos de interfaces

Tipo: RNF Secundário

Código: 4.3.3 **Título:** RNF Interface de Software

Quantidade de Filhos: 1 **Peso:** 4 **Nota:** 10

Decomposto de: 4.3 RNF Padrões de Interfaces

Decomposições: 4.3.3.1 RNF Prover padrões de interoperabilidade e portabilidade

Definição: é a capacidade do software de prover serviços através de interfaces compatíveis com outros sistemas especificados, ou seja, componentes comprados, reusados e/ou componentes desenvolvidos para subsistemas fora do escopo mas que interage com o software em desenvolvimento.

Influências Positivas (não identificadas)

Influências Negativas (não identificadas)

Planilha 39 – Planilha do RNF Secundário Interface de Software

Tipo: RNF Secundário		
Código: 4.4.1	Título: RNF Padrões de Segurança para web	
Quantidade de Filhos: 4	Peso: 10	Nota: 6,5
Decomposto de: 4.4 RNF Padrões de Segurança		
Decomposições: 4.4.1.1 RNF padrões XML		
Definição: é a capacidade do software de prover a segurança dos serviços via web utilizando padrões de segurança especificados para este fim		
Influências Positivas (não identificadas)		
Influências Negativas (não identificadas)		

Planilha 40 – Planilha do RNF Secundário Padrões de Segurança para web

A seguir as planilhas de Requisitos não Funcionais Específicos serão mostradas

Tipo: RNF Específico	
Código: 1.1.1	Título: RNF Persistência dos dados
Definição: é a capacidade do software de gerar histórico das ações executadas e de disponibilizar os registros que são criados	
Decomposto de: 1.1. RNF Confiabilidade	
Influências Positivas (não identificadas)	
Influências Negativas (não identificadas)	
Peso: 1	Nota: 3

Planilha 41 – Planilha do RNF Específico Persistência dos dados

Tipo: RNF Específico	
Código: 1.1.2	Título: RNF Completeza
Definição: Garantir a completeza das informações contidas nos laudos, exames e prontuário do paciente.	
Decomposto de: 1.1 RNF Confiabilidade	
Influências Positivas (não identificadas)	
Influências Negativas (não identificadas)	
Peso: 2	Nota: 10

Planilha 42 – Planilha do RNF Específico Completeza

Tipo: RNF Específico	
Código: 1.1.3	Título: RNF Precisão
Definição: Garantir que as informações contidas nos laudos, exames e no prontuário do paciente sejam precisas.	
Decomposto de: 1.1 RNF Confiabilidade	
Influências Positivas (não identificadas)	
Influências Negativas (não identificadas)	
Peso: 2	Nota: 10

Planilha 43 – Planilha do RNF Específico Precisão

Tipo: RNF Específico	
Código: 1.1.4	Título: RNF Consistência dos dados
Definição: Garantir que as informações contidas nos laudos, exames e no prontuário do paciente sejam consistentes.	
Decomposto de: 1.1 RNF Confiabilidade	
Influências Positivas (não identificadas)	
Influências Negativas (não identificadas)	
Peso: 1	Nota: 6,5

Planilha 44 – Planilha do RNF Específico Consistência dos dados

Tipo: RNF Específico	
Código: 1.1.5.1.1	Título: RNF Capacidade de reverter funções de efeito drástico
Definição: é a capacidade do software de poder desfazer ações executadas sem perder a integridade e a consistência das informações.	
Decomposto de: 1.1.5.1 RNF Recuperabilidade em caso de falha	
Influências Positivas (não identificadas)	
Influências Negativas (não identificadas)	
Peso: 10	Nota: 3

Planilha 45 – Planilha do RNF Específico Capacidade de reverter funções de efeito drástico

Tipo: RNF Específico	
Código: 1.1.5.2	Título: RNF Capacidade de interromper um processamento demorado
Definição: é a capacidade do software de abortar ações de processamento demorado mantendo seu estado antes da ação ser executada.	
Decomposto de: 1.1.5 RNF Tolerância a falha	
Influências Positivas (não identificadas)	
Influências Negativas (não identificadas)	
Peso: 3	Nota: 6,5

Planilha 46 – Planilha do RNF Específico Capacidade de interromper um processamento demorado

Tipo: RNF Específico	
Código: 1.1.6.1	Título: RNF Prevenção e contingência
Definição: é a evidencia de ações preventivas contra falhas no sistema, assim como também de um plano de contingência caso a falha venha acontecer.	
Decomposto de: 1.1.6 RNF Maturidade	
Influências Positivas (não identificadas)	
Influências Negativas (não identificadas)	
Peso: 10	Nota: 6,5

Planilha 47 – Planilha do RNF Específico Prevenção e Contingência

Tipo: RNF Específico	
Código: 1.2.1.1	Título: RNF Identificação única para o paciente
Definição: o objetivo é garantir que o software tenha um único identificador para cada paciente.	
Decomposto de: 1.2.1 RNF Aspectos de aquisição de dados	
Influências Positivas (não identificadas)	
Influências Negativas (não identificadas)	
Peso: 2	Nota: 10

Planilha 48 – Planilha do RNF Específico Identificação única para o paciente

Tipo: RNF Específico	
Código: 1.2.1.2	Título: RNF Importação de dados externos
Definição: é a capacidade do software de prover funcionalidades para obter dados procedentes de outros sistemas.	
Decomposto de: 1.2.1 RNF Aspectos de aquisição de dados	
Influências Positivas (não identificadas)	
Influências Negativas (não identificadas)	
Peso: 2	Nota: 6,5

Planilha 49 – Planilha do RNF Específico Importação de dados externos

Tipo: RNF Específico	
Código: 1.2.1.3.1	Título: RNF Reconhecimento de textos, sons e imagens
Definição: é a capacidade do software de interpretar, reconhecer e manipular dados de som, textos e imagens.	
Decomposto de: 1.2.1.3 RNF Uso de processos inteligentes	
Influências Positivas (não identificadas)	
Influências Negativas (não identificadas)	
Peso: 10	Nota: 6,5

Planilha 50 – Planilha do RNF Específico Reconhecimento de textos, sons e imagens

Tipo: RNF Específico	
Código: 1.2.1.4	Título: RNF Aquisição de textos, sinais, sons e imagens
Definição: tem o objetivo de garantir que o software seja capaz de receber como entrada de dados não somente texto, mas também sons, imagens, sinais etc...	
Decomposto de: 1.2.1 RNF Aspectos de Aquisição de dados	
Influências Positivas (não identificadas)	
Influências Negativas (não identificadas)	
Peso: 2	Nota: 6,5

Planilha 51 – Planilha do RNF Específico Aquisição de textos, sons e imagens

Tipo: RNF Específico	
Código: 1.2.1.5	Título: RNF Estrutura do prontuário do paciente
Definição: visa melhorar a eficiência e organização do armazenamento das informações de saúde, com a promessa de não só substituir o prontuário em papel, mais também elevar a qualidade da assistência à saúde através de novos recursos e aplicações.	
Decomposto de: 1.2.1 RNF Aspectos de Aquisição de dados	
Influências Positivas (não identificadas)	
Influências Negativas (não identificadas)	
Peso: 2	Nota: 3

Planilha 52 – Planilha do RNF Específico Estrutura do prontuário do paciente

Tipo: RNF Específico	
Código: 1.2.2.1.1	Título: RNF Apresentação uniforme dos dados do paciente
Definição: é a capacidade do software de apresentar uniformemente os dados do paciente, facilitando a troca desses dados com os diferentes sistemas especificados sem a necessidade de tradutores.	
Decomposto de: 1.2 2.1 RNF Busca por diversas informações	
Influências Positivas (não identificadas)	
Influências Negativas (não identificadas)	
Peso: 10	Nota: 6,5

Planilha 53 – Planilha do RNF Específico Apresentação uniforme dos dados do paciente

Tipo: RNF Específico	
Código: 1.2.2.2	Título: RNF Exporta dados do paciente para outros sistemas
Definição: é a capacidade do software de prover funções que possibilite enviar dados do paciente para outros sistemas especificados.	
Decomposto de: 1.2 2 RNF Aspectos de saída	
Influências Positivas (não identificadas)	
Influências Negativas (não identificadas)	
Peso: 2	Nota: 6,5

Planilha 54 – Planilha do RNF Específico Exporta dados do paciente para outros sistemas

Tipo: RNF Específico	
Código: 1.2.2.3.1 Título: RNF Elaboração de estatística com geração automática de gráficos;	
Definição: é a capacidade do software de manipular os dados para mostrar como saída resultados estatísticos em forma de gráfico.	
Decomposto de: 1.2 2.3 RNF Saída dos resultados em vídeo ou impressora	
Influências Positivas (não identificadas)	
Influências Negativas (não identificadas)	
Peso: 10	Nota: 3

Planilha 55 – Planilha do RNF Específico Elaboração de estatística com geração automática de gráficos

Tipo: RNF Específico	
Código: 1.2.2.4 Título: RNF Utiliza código de barra	
Definição: é a capacidade do software de utilizar código de barra em suas faturas.	
Decomposto de: 1.2 2 RNF Aspectos de saída	
Influências Positivas (não identificadas)	
Influências Negativas (não identificadas)	
Peso: 2	Nota: 3

Planilha 56 – Planilha do RNF Específico Utiliza código de barra

Tipo: RNF Específico	
Código: 1.2.2.5 Título: RNF Assinaturas Digitais dos relatórios	
Definição: é a capacidade do software de utilizar assinaturas digitalizadas em seus relatórios.	
Decomposto de: 1.2 2 RNF Aspectos de saída	
Influências Positivas (não identificadas)	
Influências Negativas (não identificadas)	
Peso: 2	Nota: 0

Planilha 57 – Planilha do RNF Específico Assinaturas digitais dos relatórios

Tipo: RNF Específico	
Código: 1.2.2.6 Título: RNF Busca de imagens por conteúdo.	
Definição: é uma consulta capaz de recuperar imagens do banco de imagens a partir de uma imagem de consulta.	
Decomposto de: 1.2 2 RNF Aspectos de saída	
Influências Positivas (não identificadas)	
Influências Negativas (não identificadas)	
Peso: 2	Nota: 3

Planilha 58 – Planilha do RNF Específico Busca de imagens por conteúdo

Tipo: RNF Específico	
Código: 1.2.3.1 Título: RNF Ajuda <i>online</i> sensível ao contexto	
Definição: é a capacidade do software em disponibilizar ajuda on-line aos seus usuários.	
Decomposto de: 1.2.3 RNF Aspectos de ajuda	
Influências Positivas (não identificadas)	
Influências Negativas (não identificadas)	
Peso 4	Nota: 6,5

Planilha 59 – Planilha do RNF Específico Ajuda *online* sensível ao contexto

Tipo: RNF Específico	
Código: 1.2.3.2.1 Título: RNF Documentação do usuário de fácil compreensão	
Definição: o objetivo é garantir que a documentação existente seja fácil de entender e que realmente auxilie o usuário na utilização do software.	
Decomposto de: 1.2.3.2 RNF Manual do usuário atualizado	
Influências Positivas (não identificadas)	
Influências Negativas (não identificadas)	
Peso: 5	Nota: 6,5

Planilha 60 – Planilha do RNF Específico Documentação do usuário de fácil compreensão

Tipo: RNF Específico	
Código: 1.2.3.2.2	Título: RNF Sumário e índice remissivo na documentação do usuário
Definição: o objetivo deste requisito é garantir que a documentação do software possibilite a busca rápida de uma determinada informação.	
Decomposto de: 1.2.3.2 RNF Manual do usuário atualizado	
Influências Positivas (não identificadas)	
Influências Negativas (não identificadas)	
Peso: 2	Nota: 3

Planilha 61 – Planilha do RNF Específico Sumário e índice remissivo na documentação do usuário

Tipo: RNF Específico	
Código: 1.2.3.2.3	Título: RNF Presença de Manual de instalação com instruções detalhadas
Definição: o objetivo é garantir que existe uma documentação para auxiliar o usuário na instalação do software.	
Decomposto de: 1.2.3.2 RNF Manual do usuário atualizado	
Influências Positivas (não identificadas)	
Influências Negativas (não identificadas)	
Peso: 3	Nota: 10

Planilha 62 – Planilha do RNF Específico Presença de Manual de instalação com instruções detalhadas

Tipo: RNF Específico	
Código: 1.2.3.3.1	Título: RNF Consistência de vocabulário entre as mensagens e a documentação
Definição: o objetivo é garantir que o vocabulário usado nas mensagens seja o mesmo usado na documentação do software.	
Decomposto de: 1.2.3.3 RNF Mensagens auto explicativas	
Influências Positivas (não identificadas)	
Influências Negativas (não identificadas)	
Peso: 7	Nota: 6,5

Planilha 63 – Planilha do RNF Específico Consistência de vocabulário entre as mensagens e a documentação

Tipo: RNF Específico	
Código: 1.2.3.3.2.1	Título: RNF Mensagens de erro com informações necessárias para solução da situação de erro.
Definição: garantir que a mensagem emitida contenha informações suficientes para orientar o usuário a solucionar a situação que causa o erro.	
Decomposto de: 1.2.3.3.2 RNF Diferenciação dos tipos de mensagens: confirmação, consulta, advertência e erro.	
Influências Positivas (não identificadas)	
Influências Negativas (não identificadas)	
Peso: 5	Nota: 3

Planilha 64 – Planilha do RNF Específico Mensagens de erro com informações necessárias para solução da situação de erro.

Tipo: RNF Específico	
Código: 1.2.3.3.2.2	Título: RNF Alertas claros para as conseqüências de uma determinada confirmação.
Definição: São mensagens de alertas que explicam ao usuário as conseqüências de determinadas ações e que aguardam confirmação para serem executadas ou canceladas.	
Decomposto de: 1.2.3.3.2 RNF Diferenciação dos tipos de mensagens: confirmação, consulta, advertência e erro.	
Influências Positivas (não identificadas)	
Influências Negativas (não identificadas)	
Peso: 5	Nota: 6,5

Planilha 65 – Planilha do RNF Específico Alertas claros para as conseqüências de uma determinada confirmação

Tipo: RNF Específico	
Código: 1.2.4.1	Título: RNF Aprendibilidade
Definição: é a capacidade do software de se fazer entender.	
Decomposto de: 1.2.4 RNF Aspectos Gerais de Usabilidade	
Influências Positivas (não identificadas)	
Influências Negativas (não identificadas)	
Peso: 3	Nota: 6,5

Planilha 66 – Planilha do RNF Específico Aprendibilidade

Tipo: RNF Específico	
Código: 1.2.4.2	Título: RNF Operacionalidade
Definição: é a evidencia de características do software capaz de facilitar o seu controle e aprendizado.	
Decomposto de: 1.2.4 RNF Aspectos Gerais de Usabilidade	
Influências Positivas (não identificadas)	
Influências Negativas (não identificadas)	
Peso: 4	Nota: 3

Planilha 67 – Planilha do RNF Específico Operacionalidade

Tipo: RNF Específico	
Código: 1.2.4.3	Título: RNF Atratividade.
Definição: é a evidencia de características do software capaz de torna-lo atraente aos seus usuários	
Decomposto de: 1.2.4 RNF Aspectos Gerais de Usabilidade	
Influências Positivas (não identificadas)	
Influências Negativas (não identificadas)	
Peso: 3	Nota: 3

Planilha 68 – Planilha do RNF Específico Atratividade

Tipo: RNF Específico**Código:** 1.3.1.1 **Título:** RNF Registro das autorizações de acesso**Definição:** é a evidencia de que todas as autorizações de acesso dadas aos usuários ficarão registradas**Decomposto de:** 1.3.1 RNF Mecanismos de administração de usuários centralizado.**Influências Positivas** (não identificadas)**Influências Negativas** (não identificadas)**Peso:** 10**Nota:** 10

Planilha 69 – Planilha do RNF Específico Registro das autorizações de acesso

Tipo: RNF Específico**Código:** 1.3.2.1 **Título:** RNF Autenticação.**Definição:** é a capacidade do software de garantir que toda entidade que desejar usar o software deve ser autenticada.**Decomposto de:** 1.3.2 RNF Política de controle de acesso.**Influências Positivas** (não identificadas)**Influências Negativas** (não identificadas)**Peso:** 5**Nota:** 10

Planilha 70 – Planilha do RNF Específico Autenticação

Tipo: RNF Específico**Código:** 1.3.2.2 **Título:** RNF Identificação única de usuários.**Definição:** é a capacidade do software de garantir que cada usuário será identificado de maneira única.**Decomposto de:** 1.3.2 RNF Política de controle de acesso.**Influências Positivas** (não identificadas)**Influências Negativas** (não identificadas)**Peso:** 5**Nota:** 10

Planilha 71 – Planilha do RNF Específico Identificação única de usuários

Tipo: RNF Específico	
Código: 1.3.3.1 Título: RNF Limitar a existência de senhas	
Definição: é a capacidade do software de garantir que os usuários que não usarem o sistema por um período de tempo especificado, terão seus acessos suspensos automaticamente.	
Decomposto de: 1.3.3 RNF Política de senhas.	
Influências Positivas (não identificadas)	
Influências Negativas (não identificadas)	
Peso: 1	Nota: 6,5

Planilha 72 – Planilha do RNF Específico Limitar a existência de senhas

Tipo: RNF Específico	
Código: 1.3.3.2 Título: RNF Atualização periódica de senhas	
Definição: é a capacidade do software de garantir que toda senha tenha um prazo de expiração e que o usuário deve atualiza-la.	
Decomposto de: 1.3.3 RNF Política de senhas.	
Influências Positivas (não identificadas)	
Influências Negativas (não identificadas)	
Peso: 3	Nota: 6,5

Planilha 73 – Planilha do RNF Específico Atualização periódica de senhas

Tipo: RNF Específico	
Código: 1.3.3.3 Título: RNF Não reutilização de senhas antigas	
Definição: tem objetivo de garantir que o usuário não possa reutilizar as senhas antigas no momento da atualização das mesmas.	
Decomposto de: 1.3.3 RNF Política de senhas.	
Influências Positivas (não identificadas)	
Influências Negativas (não identificadas)	
Peso: 1	Nota: 3

Planilha 74 – Planilha do RNF Específico Não reutilização de senhas antigas

Tipo: RNF Específico	
Código: 1.3.3.4 Título: RNF Proibir a auditoria de informações relativas a senhas	
Definição: é a capacidade do software em garantir que as senhas de seus usuários não serão conhecidas mesmo quando o sistema estiver sob auditoria.	
Decomposto de: 1.3.3 RNF Política de senhas.	
Influências Positivas (não identificadas)	
Influências Negativas (não identificadas)	
Peso: 2	Nota: 6,5

Planilha 75 – Planilha do RNF Específico Proibir a auditoria de informações relativas a senhas

Tipo: RNF Específico	
Código: 1.3.3.5 Título: RNF Manipular senhas criptografadas.	
Definição: garantir que as informações de senhas para abertura de arquivos ou senhas de usuários se trafegarem na rede sejam enviadas de forma criptografadas.	
Decomposto de: 1.3.3 RNF Política de senhas.	
Influências Positivas (não identificadas)	
Influências Negativas (não identificadas)	
Peso: 3	Nota: 10

Planilha 76 – Planilha do RNF Específico Manipular senhas criptografadas

Tipo: RNF Específico	
Código: 1.3.4.1 Título: RNF Redundância de hardware e software	
Definição: garantir que existe um espelho de servidores para garantir a disponibilidade do sistema em caso de defeito ou falha no servidor que está sendo usado.	
Decomposto de: 1.3.4 RNF Disponibilidade	
Influências Positivas (não identificadas)	
Influências Negativas (não identificadas)	
Peso: 3	Nota: 6,5

Planilha 77 – Planilha do RNF Específico Redundância de hardware e software

Tipo: RNF Específico	
Código: 1.3.4.2 Título: RNF Cópias de segurança	
Definição: garantir que existe espelhos de servidores para assegurar a disponibilidade do sistema.	
Decomposto de: 1.3.4 RNF Disponibilidade	
Influências Positivas (não identificadas)	
Influências Negativas (não identificadas)	
Peso: 4	Nota: 10

Planilha 78 – Planilha do RNF Específico Cópias de segurança

Tipo: RNF Específico	
Código: 1.3.4.3 Título: RNF Auditoria	
Definição: é o processo de assegurar que a atividade de um usuário seja devidamente registrada e revista para detectar eventos suspeitos.	
Decomposto de: 1.3.4 RNF Disponibilidade	
Influências Positivas (não identificadas)	
Influências Negativas (não identificadas)	
Peso: 3	Nota: 3

Planilha 79 – Planilha do RNF Específico Auditoria

Tipo: RNF Específico	
Código: 1.3.5.1 Título: RNF Restringir acesso a sala dos servidores	
Definição: garantir que a política de segurança adotada, restrinja o acesso de pessoas não autorizadas nas dependências onde se localizam os servidores.	
Decomposto de: 1.3.5 RNF Segurança do ambiente	
Influências Positivas (não identificadas)	
Influências Negativas (não identificadas)	
Peso: 5	Nota: 6,5

Planilha 80 – Planilha do RNF Específico Restringir acesso a sala dos servidores

Tipo: RNF Específico	
Código: 1.3.5.2 Título: RNF Proteção contra incêndios e inundações	
Definição: garantir que as informações do sistema, cópias de segurança, servidores estejam situados em locais imunes contra incêndios e inundações.	
Decomposto de: 1.3.5 RNF Segurança do ambiente	
Influências Positivas (não identificadas)	
Influências Negativas (não identificadas)	
Peso: 5	Nota: 3

Planilha 81 – Planilha do RNF Específico Proteção contra incêndios e inundações

Tipo: RNF Específico	
Código: 2.1.1.1.1 Título: RNF Usar indexação de imagens medicas	
Definição: garantir que as imagens médicas armazenadas serão indexadas para agilizar a sua recuperação.	
Decomposto de: 2.1.1.1 RNF Busca de imagens não pode ultrapassar 3 segundos	
Influências Positivas (não identificadas)	
Influências Negativas (não identificadas)	
Peso: 10	Nota: 6,5

Planilha 82 – Planilha do RNF Específico Usar indexação de imagens médicas

Tipo: RNF Específico	
Código: 2.1.2.1 Título: RNF Armazenar imagens compactadas	
Definição: garantir que as imagens sejam armazenadas compactadas para reduzir o espaço de armazenamento no banco de dados.	
Decomposto de: 2.1.2 RNF Eficiência de armazenamento	
Influências Positivas (não identificadas)	
Influências Negativas (não identificadas)	
Peso: 10	Nota: 10

Planilha 83 – Planilha do RNF Específico Armazenar imagens compactadas

Tipo: RNF Específico	
Código: 2.2.1.1 Título: RNF Compatibilidade	
Definição: garantir que o formato e organização dos dados e protocolos utilizados pelo software sejam compatíveis com outras tecnologias.	
Decomposto de: 2.2.1 RNF Interoperabilidade técnica	
Influências Positivas (não identificadas)	
Influências Negativas (não identificadas)	
Peso: 10	Nota: 3

Planilha 84 – Planilha do RNF Específico Compatibilidade

Tipo: RNF Específico	
Código: 2.2.2.1 Título: RNF Inteligibilidade	
Definição: O software deve garantir que o significado da informação trocada seja inteligível por qualquer aplicativo especificado.	
Decomposto de: 2.2.2 RNF Interoperabilidade Semântica	
Influências Positivas (não identificadas)	
Influências Negativas (não identificadas)	
Peso: 10	Nota: 6,5

Planilha 85 – Planilha do RNF Específico Inteligibilidade

Tipo: RNF Específico	
Código: 3.1.1 Título: RNF Analisabilidade	
Definição: Capacidade do software de diagnosticar deficiências e causas de defeitos.	
Decomposto de: 3.1 RNF Manutenibilidade	
Influências Positivas (não identificadas)	
Influências Negativas (não identificadas)	
Peso: 2	Nota: 3

Planilha 86 – Planilha do RNF Específico Analisabilidade

Tipo: RNF Específico	
Código: 3.1.2 Título: RNF Modificabilidade	
Definição: Capacidade do software de receber modificações.	
Decomposto de: 3.1 RNF Manutenibilidade	
Influências Positivas (não identificadas)	
Influências Negativas (não identificadas)	
Peso: 2	Nota: 6,5

Planilha 87 – Planilha do RNF Específico Modificabilidade

Tipo: RNF Específico	
Código: 3.1.3 Título: RNF Estabilidade	
Definição: Capacidade do software de evitar efeitos inesperados a partir de modificações.	
Decomposto de: 3.1 RNF Manutenibilidade	
Influências Positivas (não identificadas)	
Influências Negativas (não identificadas)	
Peso: 4	Nota: 6,5

Planilha 88 – Planilha do RNF Específico Estabilidade

Tipo: RNF Específico	
Código: 3.1.4 Título: RNF Testabilidade	
Definição: Capacidade do software de validar as modificações realizadas.	
Decomposto de: 3.1 RNF Manutenibilidade	
Influências Positivas (não identificadas)	
Influências Negativas (não identificadas)	
Peso: 2	Nota: 3

Planilha 89 – Planilha do RNF Específico Testabilidade

Tipo: RNF Específico	
Código: 3.2.1 Título: RNF Independência dos dados do paciente	
Definição: é a capacidade do software em garantir a independência dos dados do paciente em relação aos programa que os utilizam.	
Decomposto de: 3.2 RNF Flexibilidade	
Influências Positivas (não identificadas)	
Influências Negativas (não identificadas)	
Peso: 10	Nota: 10

Planilha 90 – Planilha do RNF Específico Independência dos dados do paciente

Tipo: RNF Específico	
Código: 3.3.1 Título: RNF Adaptabilidade	
Definição: Capacidade do software de ser adaptado em diferentes ambientes computacionais sem intervenção.	
Decomposto de: 3.3 RNF Portabilidade	
Influências Positivas (não identificadas)	
Influências Negativas (não identificadas)	
Peso: 3	Nota: 6,5

Planilha 91 – Planilha do RNF Específico Adaptabilidade

Tipo: RNF Específico	
Código: 3.3.2 Título: RNF Coexistência	
Definição: Capacidade do software de coexistir com outro sistema independente em um ambiente comum, compartilhando recursos.	
Decomposto de: 3.3 RNF Portabilidade	
Influências Positivas (não identificadas)	
Influências Negativas (não identificadas)	
Peso: 7	Nota: 6,5

Planilha 92 – Planilha do RNF Específico Coexistência

Tipo: RNF Específico	
Código: 4.1.1	Título: RNF Utilizar protocolos padronizados
Definição: o objetivo é garantir que o software utiliza protocolos de comunicação como DICOM, HL7 etc..	
Decomposto de: 4.1 RNF Padrões de Intercomunicação	
Influências Positivas (não identificadas)	
Influências Negativas (não identificadas)	
Peso: 5	Nota: 6,5

Planilha 93 – Planilha do RNF Específico Utilizar protocolos padronizados

Tipo: RNF Específico	
Código: 4.1.2	Título: RNF Uniformidade do formato dos dados transmitidos
Definição: o objetivo é garantir que o software utiliza padrões como XML, HL7, DICOM para formatar os dados transmitidos.	
Decomposto de: 4.1 RNF Padrões de Intercomunicação	
Influências Positivas (não identificadas)	
Influências Negativas (não identificadas)	
Peso: 5	Nota: 10

Planilha 94 – Planilha do RNF Específico Uniformidade do formato dos dados transmitidos

Tipo: RNF Específico	
Código: 4.2.1	Título: RNF Utilizar padrões para identificar as informações
Definição: o objetivo é garantir que o software utiliza padrões como SNOMED, CID-10, UMLS, LOINC para identificar as informações médicas.	
Decomposto de: 4.2 RNF Padrões de Vocabulário	
Influências Positivas (não identificadas)	
Influências Negativas (não identificadas)	
Peso: 10	Nota: 10

Planilha 95 – Planilha do RNF Específico Utilizar padrões para identificar as informações

Tipo: RNF Específico	
Código: 4.3.1.1.1	Título: RNF Clareza nos formatos das telas e relatórios
Definição: é a capacidade do software em garantir que as informações contidas nas telas e nos relatórios sejam de fácil entendimento para o usuário.	
Decomposto de: 4.3.1.1 RNF Usa padrões gráficos de interfaces	
Influências Positivas (não identificadas)	
Influências Negativas (não identificadas)	
Peso: 10	Nota: 6,5

Planilha 96 – Planilha do RNF Específico Clareza nos formatos das telas e relatórios

Tipo: RNF Específico	
Código: 4.3.1.2	Título: RNF Utilizar terminologia condizente com a prática médica
Definição: o objetivo é garantir a terminologia utilizada nos menus e relatórios seja condizente com a prática médica.	
Decomposto de: 4.3.1 RNF Interfaces do usuário	
Influências Positivas (não identificadas)	
Influências Negativas (não identificadas)	
Peso: 5	Nota: 6,5

Planilha 97 – Planilha do RNF Específico Utilizar terminologia condizente com a prática médica

Tipo: RNF Específico	
Código: 4.3.2	Título: RNF Interface de hardware
Definição: garantir que os equipamentos de hardware utilizados, sejam compatíveis com várias tecnologias, incluindo estrutura lógica, endereço físico, comportamento esperado etc...	
Decomposto de: 4.3 RNF Padrões de Interfaces	
Influências Positivas (não identificadas)	
Influências Negativas (não identificadas)	
Peso: 2	Nota: 10

Planilha 98 – Planilha do RNF Específico Interface de hardware

Tipo: RNF Específico	
Código: 4.3.3.1	Título: RNF Prover padrões de interoperabilidade e portabilidade
Definição: o objetivo é garantir que o software utilize padrões como CORBAMED para possibilitar a interoperabilidade e a portabilidade.	
Decomposto de: 4.3.1 RNF Padrões de Interfaces	
Influências Positivas (não identificadas)	
Influências Negativas (não identificadas)	
Peso: 10	Nota: 10

Planilha 99 – Planilha do RNF Específico Prover padrões de interoperabilidade e portabilidade

Tipo: RNF Específico	
Código: 4.4.1.	Título: RNF Padrões XML
Definição: é a capacidade do software de garantir a confidencialidade das informações contidas em documentos XML que estão em trânsito ou armazenadas.	
Decomposto de: 4.4 RNF Padrões de Segurança	
Influências Positivas (não identificadas)	
Influências Negativas (não identificadas)	
Peso: 10	Nota: 6,5

Planilha 100 – Planilha do RNF Específico Padrões XML

Tipo: RNF Específico	
Código: 4.5.1	Título: RNF Usar padrões de Conteúdo
Definição: objetivo é garantir que o software utiliza padrões como ASTM, GEHR, RIPS/PRC para formatar o conteúdo das informações médicas.	
Decomposto de: 4.5. RNF Padrões de Formatação e conteúdo	
Influências Positivas (não identificadas)	
Influências Negativas (não identificadas)	
Peso: 10	Nota: 6,5

Planilha 101 – Planilha do RNF Específico Usar padrões de Conteúdo

Neste capítulo mostramos a metodologia proposta aplicada para um Sistema de Informações Hospitalares. Utilizamos o Modelo de Qualidade Proposto na Figura 8 para elicitar os requisitos de qualidade para o sistema. Mostramos também a decomposição dos requisitos para obter a árvore de decomposição dos requisitos na Figura 18. Identificamos algumas influências entre os requisitos, construímos a política de qualidade ilustrada na Figura 19 e mostramos a fase de avaliação das conformidades detalhando a associação dos pesos aos requisitos secundários e específicos e atribuição das notas aos requisitos específicos. Por último demonstramos como calcular as notas dos requisitos primários e secundários a partir dos pesos e das notas dos requisitos específicos e finalmente identificamos as conformidades e contradições destes requisitos com a política de qualidade.

Por ser um exemplo, cuja finalidade é mostrar a aplicabilidade do método proposto, não foi feito o preenchimento das atualizações relativas as informações dos requisitos não funcionais do software, no documento de especificação de requisitos sugerido nesse trabalho. No próximo capítulo será abordado sobre a ferramenta desenvolvida para automatizar as fases da metodologia proposta. Nesta abordagem destacamos os principais casos de uso da ferramenta, o diagrama de entidade relacionamento com as principais entidades, o diagrama de classe e a prototipagem.

5 FERRAMENTA PARA ELICITAÇÃO DOS REQUISITOS

5.1 Modelagem

A implementação da metodologia proposta tem como requisitos o *cadastro dos projetos* que será responsável pela criação e abertura de projetos, o *cadastro de cenários* responsável pela relação entre os cenários e os requisitos não funcionais, o *cadastro das categorias* para registrar a categoria dos requisitos primário, *cadastro das necessidades de qualidade* responsável pelas informações das necessidades de qualidade definidas para um determinado projeto, *cadastro dos requisitos primários secundários e específicos* responsável pelas informações dos requisitos não funcionais, *cadastro das influências* positivas e negativas existente entre os requisitos, assim como todas as informações relativas aos projetos criados. A Figura 21 mostra o diagrama de entidade relacionamento (DER) das principais entidades utilizado pela ferramenta:

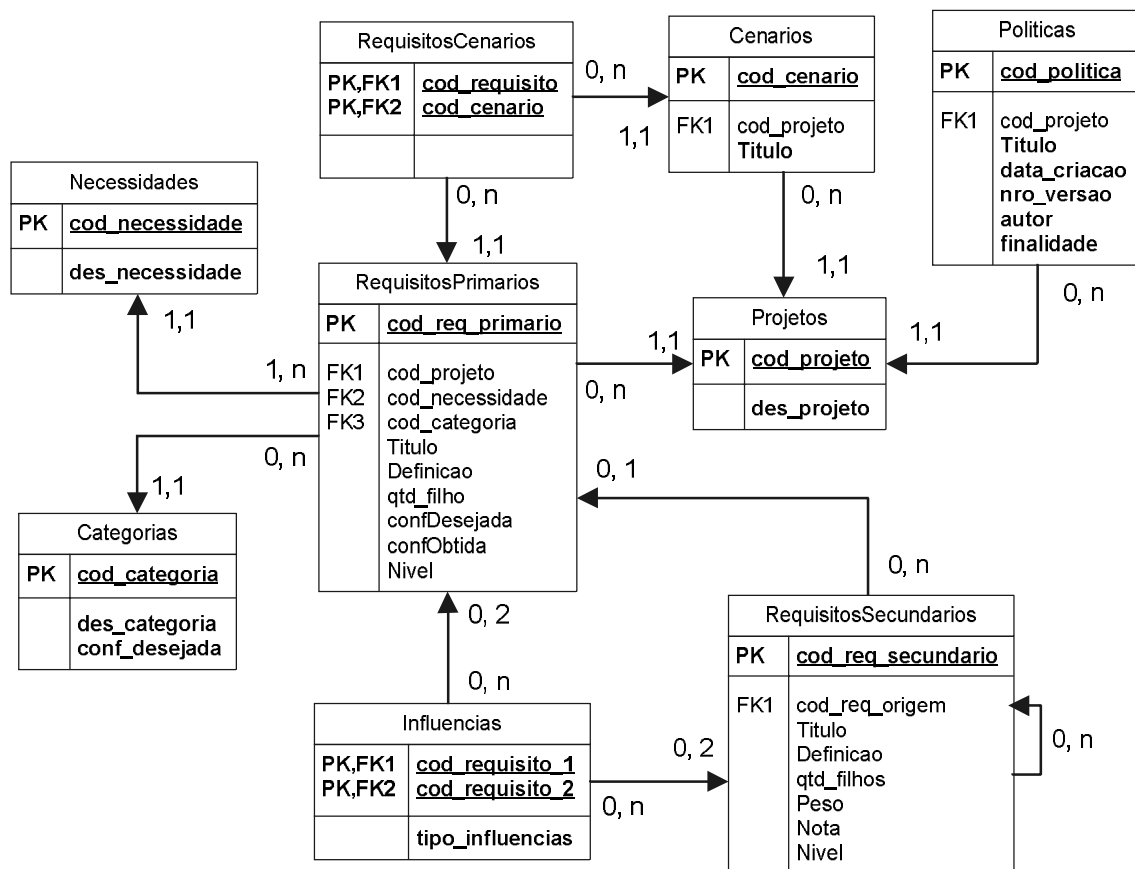


Figura 21 - Diagrama de Entidade Relacionamento do Projeto

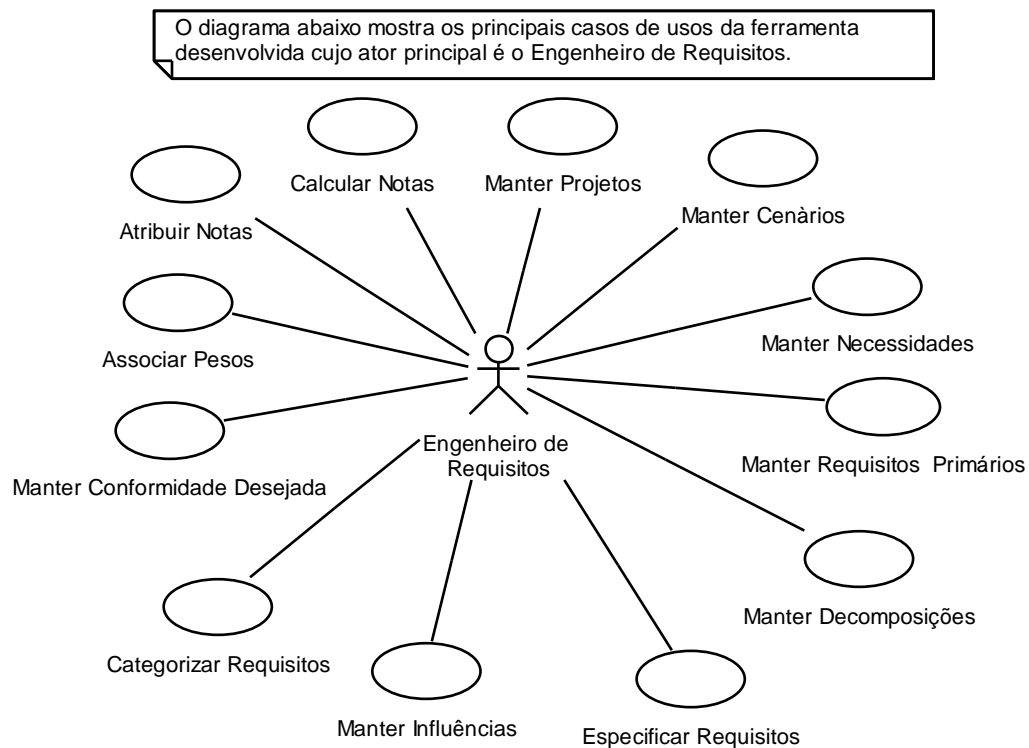


Figura 22 - Principais casos de uso do projeto

A seguir alguns dos casos de usos descritos na Figura 22 são detalhados:

Caso de uso: Manter Projetos.

❖ Cenário principal

- O ER solicita ao sistema, a interface Criar ou Abrir Projeto a partir do menu Projetos, sub-menu Novo ou Abrir;
- O sistema disponibiliza a interface de criação ou abertura de projetos de projetos para o ER;
- Se o ER solicitou novo projeto, ele digita o título do projeto e confirma a inclusão de um novo projeto;
- Sistema gera automaticamente um novo código para o projeto e emite a mensagem de Inclusão realizada com sucesso.
- Caso o ER tenha solicitado abrir projeto, ele seleciona o projeto que deseja abri-lo na lista e em seguida confirma a abertura do mesmo;

- O sistema habilitará todas as funcionalidades disponíveis para trabalhar com o projeto.

Caso de uso: Manter Requisitos Primários.

❖ Cenário principal

- O ER solicita ao sistema a interface Cadastrar Requisitos Primários a partir do menu Processos sub-menu Elicitação;
- O sistema disponibiliza para o ER, a interface de cadastro dos requisitos primários contendo uma lista de requisitos não funcionais já cadastrados;
- O ER deve selecionar um requisito da lista quando desejar fazer uma atualização no mesmo, ou digitar o título e selecionar a necessidade de qualidade que o requisito está vinculado e em seguida confirmar a operação;
- Se a operação for de atualização, o sistema efetiva a alteração realizada no requisito e emite a mensagem Alteração realizada com sucesso;
- Caso a operação seja de inclusão, o sistema gera automaticamente um código para o novo requisito, efetiva a inclusão do mesmo e emite a mensagem Inclusão realizada com sucesso.

Caso de uso: Manter Decomposições.

❖ Cenário principal

- O ER solicita ao sistema a interface Cadastrar Decomposições a partir do menu Processos sub-menu Decomposição e Representação;
- O sistema disponibiliza para o ER, a interface de cadastro das decomposições com uma árvore contendo todos os requisitos primários já cadastrados como nó principal;
- O ER seleciona o requisito que desejar decompor, a partir da árvore;
- O sistema disponibiliza uma interface para o ER digitar o título do novo requisito que será uma decomposição do requisito selecionado na árvore;
- O ER digita o título do requisito e confirma a operação;

- O sistema gera automaticamente um novo código para o requisito, efetua a inclusão do mesmo e emite a mensagem Inclusão realizada com sucesso.

Caso de uso: Manter Influências.

❖ Cenário principal

- O ER solicita ao sistema a interface Cadastrar Influências a partir do menu Processos sub-menu Identificação das Influências;
- O sistema disponibiliza para o ER, a interface de cadastro das influências contendo o tipo de influência (positiva ou negativa) e duas listas de todos os requisitos elicitados e cadastrados;
- O ER seleciona o tipo de influência que irá cadastrar e em seguida confirma a seleção;
- O sistema habilitará as listas de requisitos para o ER;
- O ER seleciona um requisito de cada lista e em seguida confirmar o registro da influência;
- O sistema efetiva a inclusão da influência nos requisitos selecionados e emite mensagem Inclusão realizada com sucesso.

A seguir será mostrado o diagrama de classe da ferramenta. Entre as classes deste diagrama estão: a classe *clsRequisitosPrimarios* que contém todos os requisitos não funcionais primários elicitados para o projeto, a classe *clsRequisitosSecundários* que contém os requisitos não funcionais secundários e específicos decompostos a partir dos requisitos Primários e a classe *clsInfluencias* que contém as informações relativas as influencias positivas e negativas identificadas entre os requisitos não funcionais. A Figura 23 mostra em detalhe o diagrama completo com as classes.

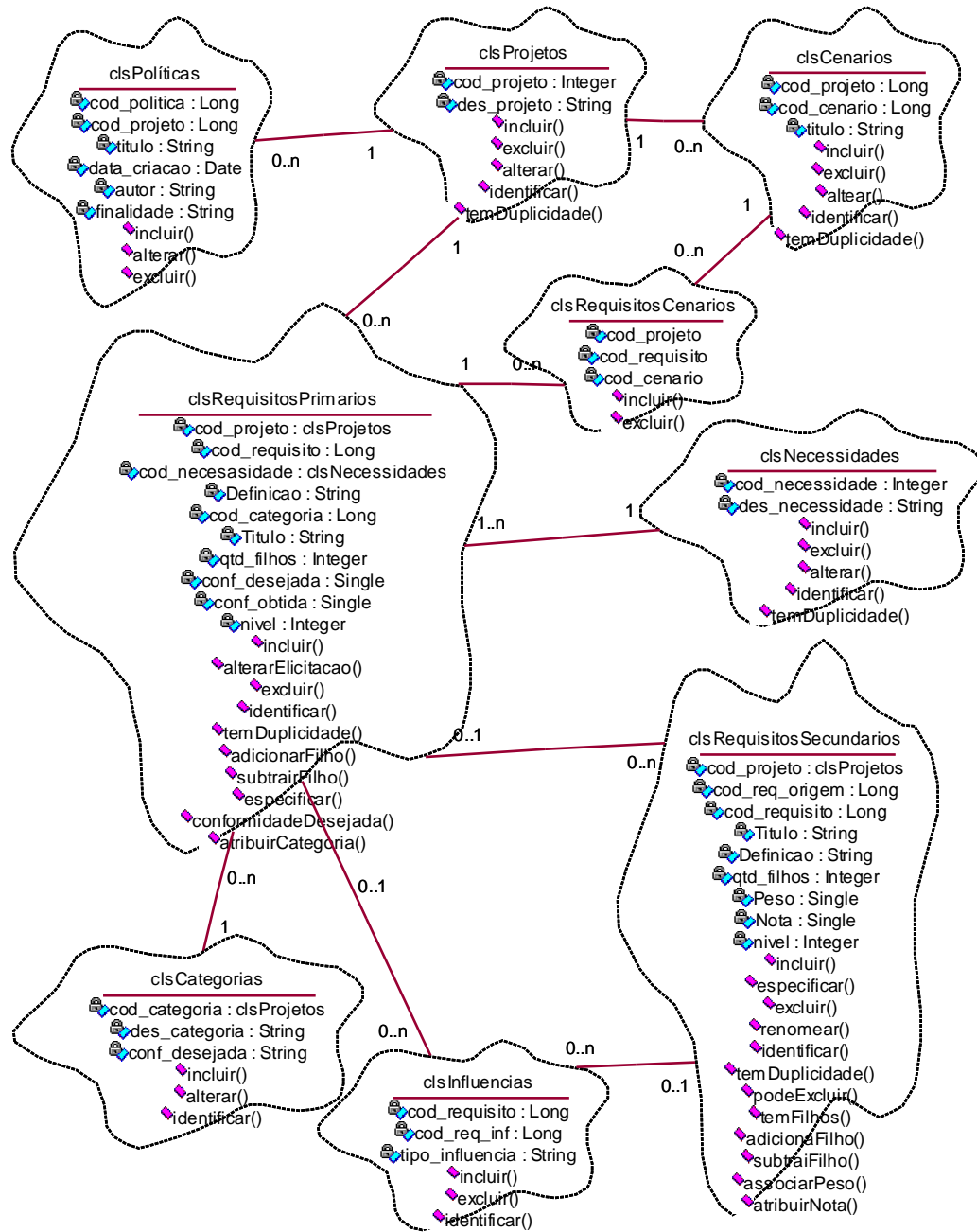


Figura 23 – Diagrama de classe da ferramenta

5.2 Prototipagem

A ferramenta foi desenvolvida utilizando o Visual Basic versão 6.0, como linguagem de programação e o Access 2003 como banco de dados. A seguir as principais funcionalidades da ferramenta serão detalhadas:

5.2.1 Criar Projetos:

O primeiro passo para usar a ferramenta, é criar um projeto e em seguida abri-lo conforme Figura 24.

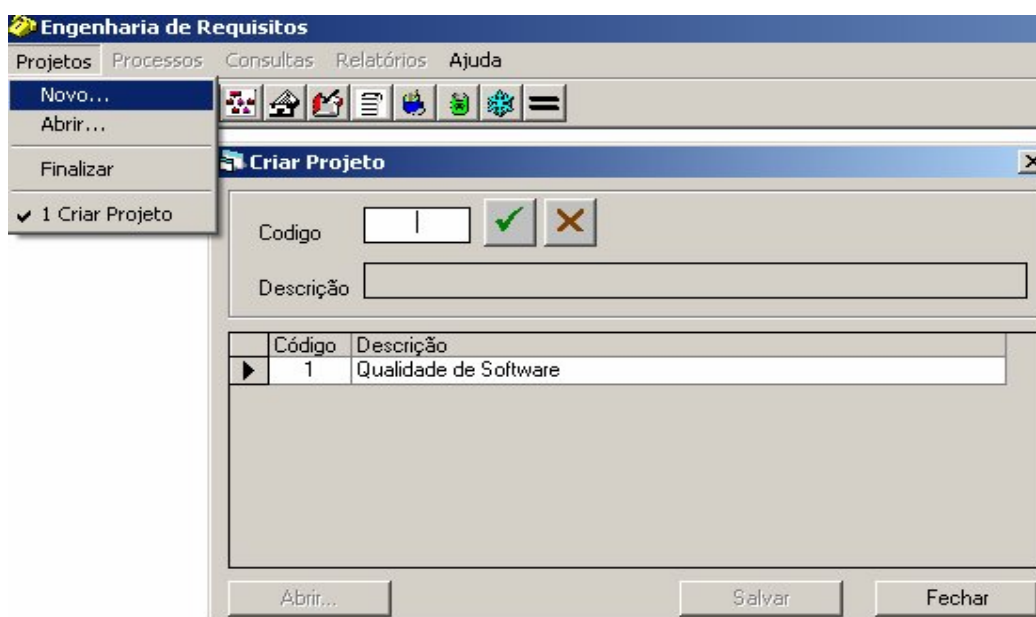


Figura 24 – Interface de Criação e abertura de Projetos

A Figura 25 mostra o menu *processos* que dispõe de todas as funcionalidades necessárias para cadastrar as informações relativas ao projeto aberto, de acordo com as fases da metodologia proposta.

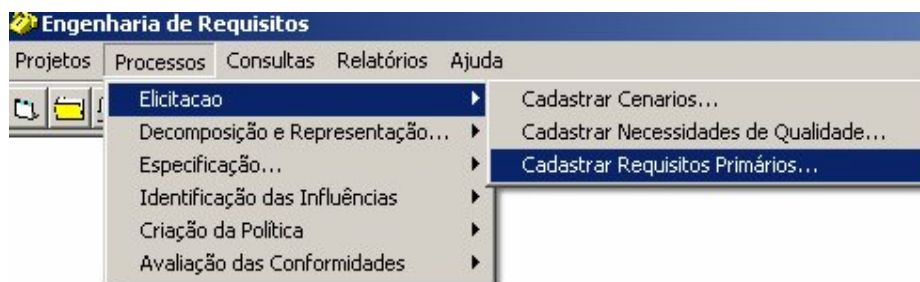


Figura 25 – Menu Processos da Ferramenta

5.2.2 Cadastrar Requisitos Primários

Esta funcionalidade permite incluir, alterar e excluir os requisitos primários elicidados, assim como relacionar estes requisitos aos cenários do projeto. Para executar esta tarefa, as necessidades de qualidade e os cenários especificados durante a elicitação dos requisitos funcionais já devem está cadastrados para o projeto selecionado. A Figura 26 mostra a interface para cadastrar requisitos primários do projeto.

Codigo	Titulo	Qualidade
11	RNF Confiabilidade	Funcional
12	RNF Usabilidade	Funcional
13	RNF Segurança	Funcional
21	RNF Eficiência	Desempenho
22	RNF Interoperabilidade	Desempenho
31	RNF Manutenibilidade	Manutenção
32	RNF Flexibilidade	Manutenção
33	RNF Portabilidade	Manutenção
41	RNF Padrões de Intercomunicação	Padronização
42	RNF Padrões de Vocabulário	Padronização
43	RNF Padrões de Interfaces	Padronização
44	RNF Padrões de Segurança	Padronização

Figura 26 – Interface de Cadastros dos Requisitos Primários

A inclusão dos requisitos primários ocorre com a execução do método *Incluir()* da classe *clsRequisitosPrimários*. Este método é executado quando o usuário aciona o botão Salvar da Figura 26. O trecho de código abaixo ilustra como a inclusão é realizada.

```
Public Function Incluir() As Boolean
    Set RS = New ADODB.Recordset
    sSQL = "INSERT INTO RNFPrimarios (" & _
        " cod_projeto, cod_necessidade" & _
        ",cod_requisito, Titulo" & _
        ",cod_categoria, confDesejada" & _
        ",confObtida, Nivel )" & _
        " VALUES (" & _
        scod_projeto & "," & scod_necessidade & "," & _
        sCod_requisito & "," & sTitulo & "," & _
        scod_categoria & "," & sconfDesejada & "," & _
        sconfObtida & "," & sNivel & ")"
    RS.Open sSQL, oConexao, adOpenDynamic
    Incluir = True
End Function
```

5.2.3 Cadastrar Decomposições

Os requisitos primários são metas de alto nível de abstração que precisam ser decompostas em submetas de mais baixo nível. Esta funcionalidade disponibiliza todos os requisitos não funcionais elicitados e permite que estes requisitos sejam decompostos em outros requisitos formando uma árvore. A interface Figura 27 permite decompor, renomear e excluir os requisitos da árvore. A exclusão só será permitida para os requisitos específicos. Requisitos secundários para serem excluídos somente quando todas as suas decomposições forem excluídas.

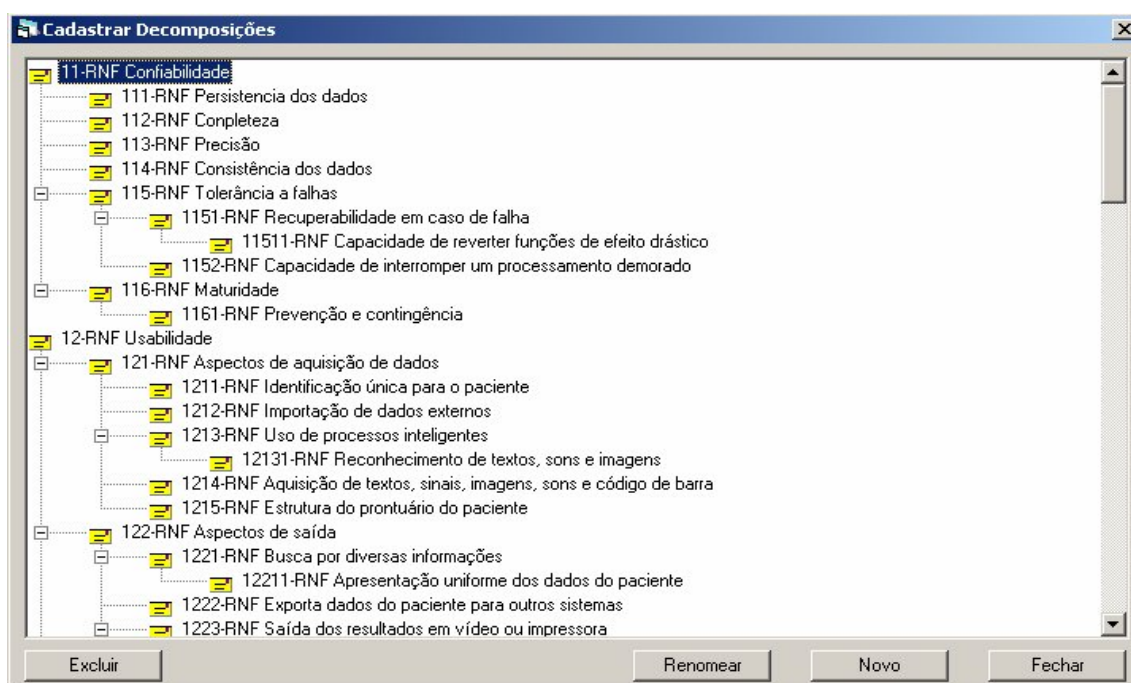


Figura 27 – Interface de Decomposições dos Requisitos Não Funcionais

Ao selecionar um nó da árvore de requisitos ilustrada na Figura 27 e pressionar o botão Novo, a interface da Figura 28 será exibida para que um novo requisito, filho do requisito selecionado seja incluído.

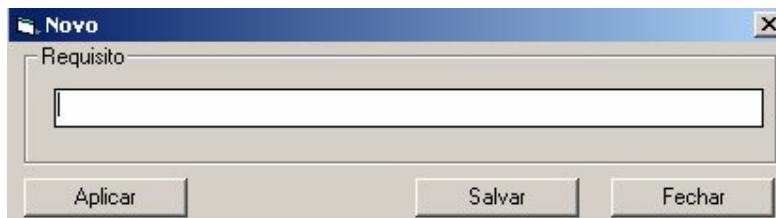


Figura 28 – Interface para Inclusão de Requisitos Secundários e Específicos

Quando o usuário aciona o botão Salvar da Figura 28, dispara o evento deste botão, que corresponde a execução do trecho de código ilustrado abaixo. Logo no início os objetos *oRequisitosPrimarios* e *oRequisitosSecundarios* são instanciados com as classes *clsRequisitosPrimarios* e *clsRequisitosSecundarios* respectivamente. Em seguida executa-se o método *Identificar* da classe *clsRequisitosPrimarios* para saber se o nó pai é um requisito Primário. Se o nó pai é um requisito Primário, então se executa o método *AdicionaFilho* da classe *clsRequisitosPrimarios* para atualizar a quantidade de filhos do nó pai e em seguida chama o método *Incluir* desta classe para adicionar um novo requisito ao nó pai.

```
Private Sub cmdSalvar_Click()
    Set oRequisitosPrimarios = New clsRequisitosPrimarios
    Set oRequisitosPrimarios.Conexao = oConexao
    Set oRequisitosSecundarios = New clsRequisitosSecundarios
    Set oRequisitosSecundarios.Conexao = oConexao

    Select Case G_Operacao
    Case Is = "N" ' NOVO
        With oRequisitosPrimarios
            .cod_requisito = G_codPai
            If .Identificar = True Then ' VERIFICA SE REQUISITO PAI É PRIMÁRIO
                If .AdicionaFilho = True Then ' ADICIONA FILHO AO PAI
                    With oRequisitosSecundarios
                        .Titulo = txtTitulo.Text
                        .qtd_filhos = 0
                        .cod_req_origem = G_codPai
                        .cod_projeto = G_CodigoProjeto
                        .Nivel = 2 ' filho de requisito primário
                    End With
                    If Sequencial(G_codPai) = True Then
                        .cod_requisito = G_codPai & nProximoSeq
                        If .Incluir = True Then ' INCLUI REQUISITO SECUNDÁRIO
                            MsgBox msgINC, vbInformation
                            txtTitulo.Text = ""
                            txtTitulo.SetFocus
                        Else
                            MsgBox msgOPC, vbCritical
                            Exit Sub
                        End If
                    Else
                        MsgBox msgOPC, vbCritical
                        Exit Sub
                    End If
                End If
            End With
        End If
    End Case
End Sub
```

Se o nó pai é um requisito Secundário, então se executa o método *Incluir* da classe *clsRequisitosSecundarios* para incluir um novo requisito ao nó pai e em seguida chama-se método *AdicionaFilhos* desta classe para atualizar a quantidade de filhos do nó pai.

```

Else ' se requisito pai não é primário
  With oRequisitosSecundarios
    .cod_req_origem = G_codPai
    .Titulo = txtTitulo.Text
    .qtd_filhos = 0
    .cod_projeto = G_CodigoProjeto
    .Nivel = G_Nivel + 1
    If Sequencial(G_codPai) = True Then
      .cod_requisito = G_codPai & nProximoSeq
      If .Incluir = True Then ' INCLUI REQUISITO SECUNDÁRIO
        .cod_requisito = G_codPai ' ADICIONA FILHO AO PAI
        If .AdicionaFilhos = True Then
          MsgBox msgINC, vbInformation
          txtTitulo.Text = ""
          txtTitulo.SetFocus
        Else
          MsgBox msgOPC, vbCritical
          Exit Sub
        End If
      Else
        MsgBox msgOPC, vbCritical
        Exit Sub
      End If
    Else
      MsgBox msgOPC, vbCritical
      Exit Sub
    End If
  End With
End If
End With

```

5.2.4 Especificar Requisitos

A especificação dos requisitos deve ser feita após a decomposição de todos os requisitos primários e secundários do projeto. Esta funcionalidade, permite que todos os requisitos elicitados sejam definidos de forma que não fique dúvidas quanto ao seu significado. A Figura 29 mostra a interface que permite especificar todos os requisitos elicitados e decompostos.

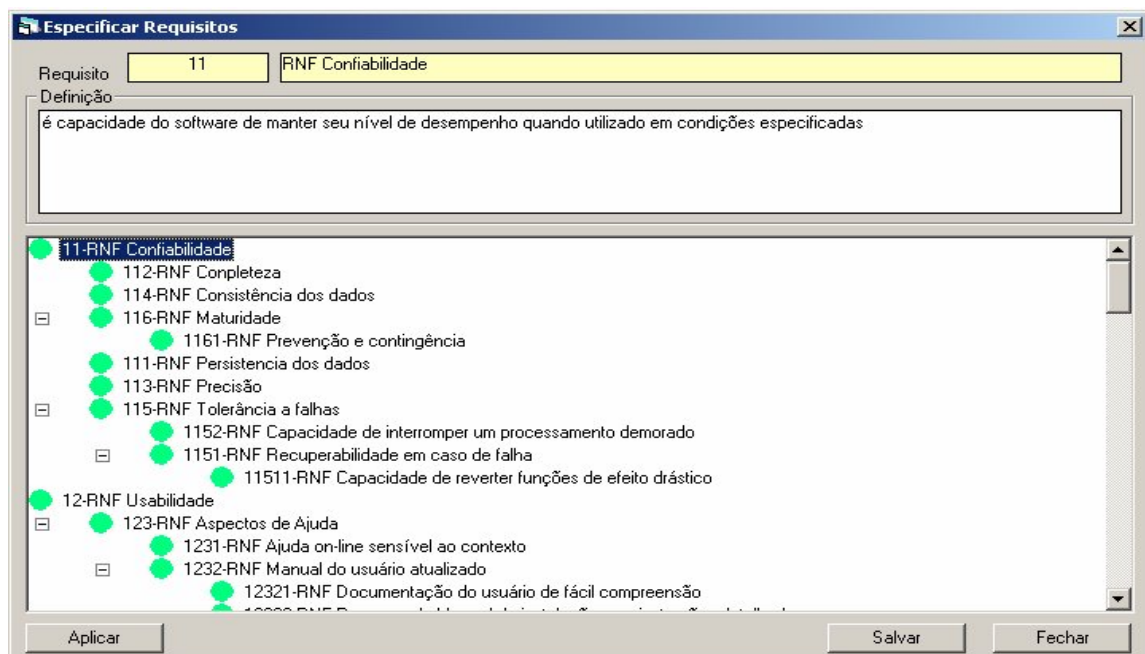


Figura 29 – Interface de Especificação dos Requisitos

Quando o usuário aciona o botão Salvar na Figura 29, dispara o evento deste botão, que corresponde a execução do trecho de código ilustrado na página a seguir. Logo no início é feito um teste na variável *sTipoReq* para saber se o requisitos selecionado é Primário ou Secundário. Se o requisitos for Primário (*sTipoReq = 1*), o objeto *oRequisitosPrimarios* é instanciado a partir da classe *clsRequisitosPrimarios* e em seguida o método *Especificar* desta classe é executado para concretizar a especificação do requisito. Se o requisitos for Secundário ou Especifico (*sTipoReq = 2*), o objeto *oRequisitosSecundários* é instanciado a partir da classe *clsRequisitosSecundários* e em seguida o método *Especificar* desta classe é executado para concretizar a especificação do requisito.

```
Private Sub cmdSalvar_Click()
    If sTipoReq = 1 Then ' requisito primário
        Set oRequisitosPrimarios = New clsRequisitosPrimarios
        Set oRequisitosPrimarios.Conexao = oConexao
        With oRequisitosPrimarios
            .cod_requisito = CLng(txtCodigo.Caption)
            .Definicao = txtDefinicao.Text
            If .Especificar = True Then
                MsgBox msg&LT, vbInformation
            End If
        End With
    ElseIf sTipoReq = 2 Then ' requisito secundario
        Set oRequisitosSecundarios = New clsRequisitosSecundarios
        Set oRequisitosSecundarios.Conexao = oConexao
        With oRequisitosSecundarios
            .cod_requisito = CLng(txtCodigo.Caption)
            .Definicao = txtDefinicao.Text
            If .Especificar = True Then
                MsgBox msg&LT, vbInformation
            End If
        End With
    End If
End Sub
```

5.2.5 Cadastrar Influências

O cadastro das influências consiste em registrar as influências positivas e negativas que determinados requisitos exercem sobre outros requisitos. A Figura 30 mostra a interface que permite disponível para registrar estas influências. O exemplo da Figura 30 mostra as influências positivas identificadas nos projeto.

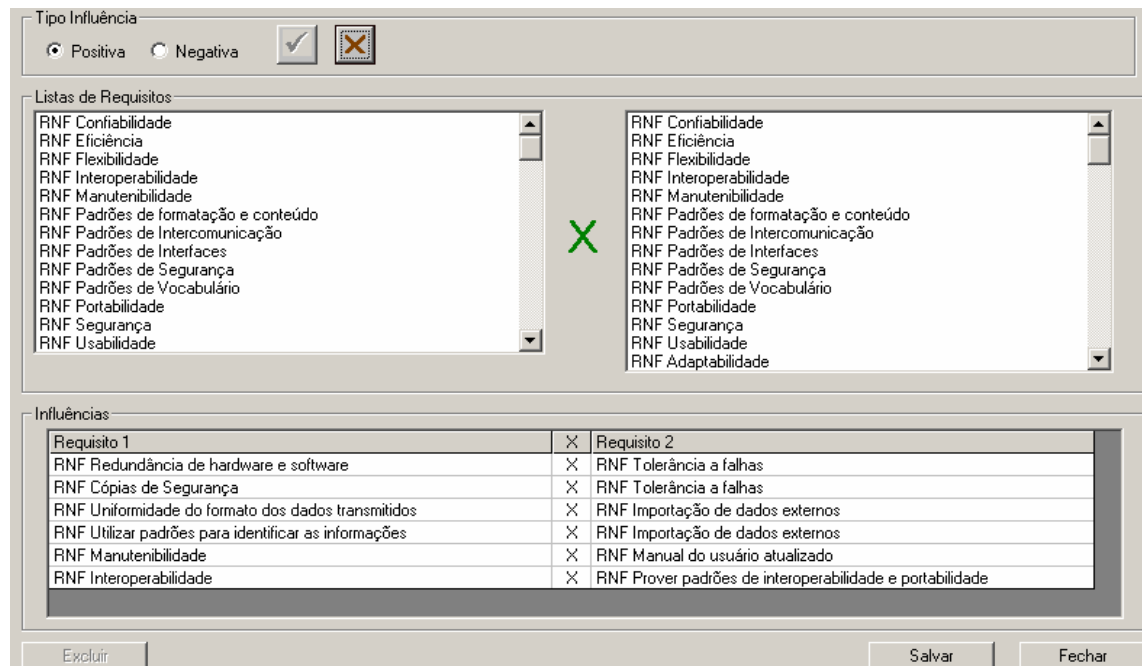


Figura 30 – Interface das Influências entre os Requisitos Primários

Quando o usuário aciona o botão Salvar na Figura 30, dispara o evento deste botão, que corresponde a execução do trecho de código ilustrado a baixo. Logo no início o objeto *oInfluencias* é instanciado a partir da classe *clsInfluencias*, em seguida atribui-se o tipo da influência (Positiva ou Negativa), executa-se o método *Identificar* desta classe para saber se a influência já foi incluída. Se a influência já foi incluída a ferramenta emite mensagem, caso contrário executa o método *Incluir* da classe para registrar a influência.

```

Private Sub cmdSalvar_Click()
    Set oInfluencias = New clsInfluencias
    Set oInfluencias.Conexao = oConexao
    With oInfluencias
        .cod_requisito = lstRequisitos_1.ItemData(lstRequisitos_1.ListIndex)
        .cod_req_inf = lstRequisitos_2.ItemData(lstRequisitos_2.ListIndex)
        If optNegativa.Value = True Then
            .tipo_influencia = "Negativa"
        ElseIf optPositiva.Value = True Then
            .tipo_influencia = "Positiva"
        End If
        If .Identificar = True Then
            MsgBox "Influência já cadastrada", vbInformation
            Exit Sub
        End If
        If .Incluir = True Then
            MsgBox msgINC, vbInformation
        End If
        lstRequisitos_1.Selected(lstRequisitos_1.ListIndex) = False
        lstRequisitos_2.Selected(lstRequisitos_2.ListIndex) = False
    End With
    Call MontaGrid
End Sub

```

5.2.6 Categorizar Requisitos

A categorização dos requisitos consiste em atribuir a cada um dos requisitos primários, uma das categorias definidas pela metodologia. A Figura 31 mostra a interface que permite categorizar os requisitos primários. O exemplo da Figura 31 mostra os requisitos não funcionais *Usabilidade*, *Interoperabilidade* e *Flexibilidade*, categorizados com *Especificação Média*.



Figura 31 – Interface de Categorização dos Requisitos Não Funcionais

Quando o usuário aciona o botão Salvar na Figura 31, dispara o evento deste botão, que corresponde à execução do trecho de código ilustrado na próxima página. Logo no início o objeto *oCategorias* é instanciado a partir da classe *clsCategorias*, em seguida para cada requisito selecionado na lista de requisitos da Figura 31, será executado o método *AtribuirCategoria* desta classe para atualizar a categoria do requisito.

```
Private Sub cmdSalvar_Click()
    Set oReqPri = New clsRequisitosPrimarios
    Set oCategorias = New clsCategorias
    Set oReqPri.Conexao = oConexao
    Set oCategorias.Conexao = oConexao
    With oReqPri
        For I = 0 To lstRequisitos.ListCount - 1
            If lstRequisitos.Selected(I) = True Then
                .cod_requisito = lstRequisitos.ItemData(I)
                .cod_categoria = cboCategorias.BoundText
                oCategorias.cod_categoria = cboCategorias.BoundText
                oCategorias.Identificar
                .confDesejada = oCategorias.conf_Desejada
                .AtribuirCategoria
            End If
        Next I
    End With
End Sub
```

5.2.7 Registrar Conformidades Desejadas

O registro das conformidades desejadas consiste em atribuir uma conformidade em termos percentuais a cada uma das três categorias de requisitos definidas pela metodologia. A Figura 32 mostra a interface que permite atribuir a conformidade às categorias. O exemplo da Figura 32 mostra uma conformidade de 80% para os requisitos categorizados com especificação forte, 65% para os requisitos de categoria especificação média e 30% para os requisitos de especificação fraca.

Codigo	Categoria	Conformidade
4	Especificação Forte	80
2	Especificação Fraca	30
3	Especificação Media	65

Figura 32 – Interface de Registro das Conformidades Desejadas

Quando o usuário aciona o botão Salvar na Figura 32, dispara o evento deste botão. Este evento invoca o método *ConformidadeDesejada()* da classe *clsRequisitosPrimarios*, que corresponde a execução do trecho de código ilustrado a baixo. O código mostra a atualização do requisito com a atribuição da conformidade desejada que corresponde a conformidade da categoria a que ele pertence.

```
Public Function ConformidadeDesejada() As Boolean
    Set RS = New ADODB.Recordset
    sSQL = "UPDATE RNFPrimarios " & _
        "SET confDesejada = '" & sconfDesejada & "'" & _
        " WHERE cod_categoria = " & scod_categoria & _
        " AND cod_projeto = " & scod_projeto
    RS.Open sSQL, oConexao, adOpenDynamic
    ConformidadeDesejada = True
End Function
```

5.2.8 Associar Pesos aos Requisitos

Os pesos devem ser associados aos requisitos secundários e específicos de acordo com sua importância para satisfazer o requisito do qual foi originado. Esta funcionalidade, permite associar os pesos aos requisitos de acordo com a metodologia proposta. A Figura 33 mostra a interface disponível para associar pesos aos requisitos do projeto.

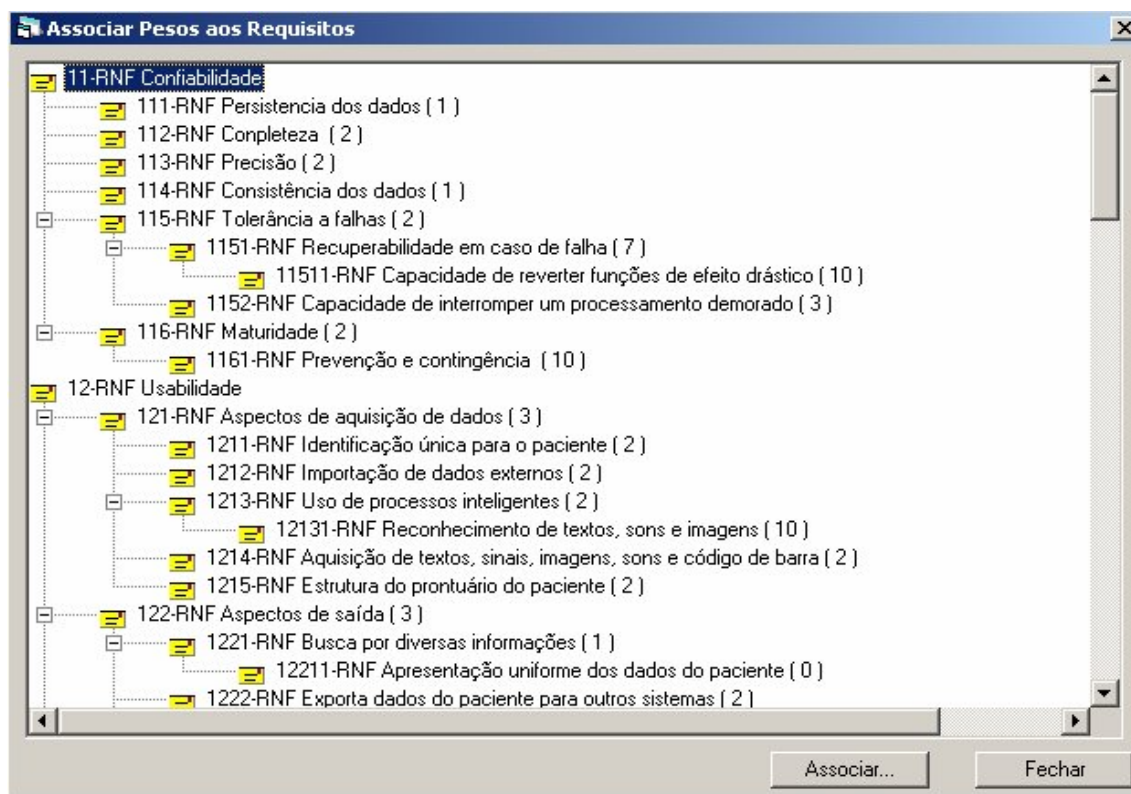


Figura 33 – Interface de Associação dos Pesos aos Requisitos

Quando o usuário aciona o Associar na Figura 33, uma nova interface será exibida pela ferramenta. Nesta nova interface, ao acionar o botão *Salvar*, dispara um evento que evento invoca o método *AssociarPeso()* da classe *clsRequisitosSecundarios*, que corresponde a execução do trecho de código ilustrado a baixo. O código mostra a atualização do requisito com a atribuição do peso atribuído pelo usuário.

```
Public Function AssociarPeso() As Boolean
    Set RS = New ADODB.Recordset
    sSQL = "UPDATE RNFSecundarios " & _
        "SET Peso = '" & sPeso & _
        "' WHERE cod_requisito = " & sCod_requisito
    RS.Open sSQL, oConexao, adOpenDynamic
    AssociarPeso = True
End Function
```

5.2.9 Atribuir Notas

A atribuição das notas aos requisitos específicos, deve ser feita após a associação dos pesos a todos os requisitos secundários e específicos do projeto. Esta funcionalidade, permite atribuir as notas 0 (zero), 3 (três), 6,6 (seis e meio) e 10 (dez) aos requisitos específicos de acordo com a metodologia proposta. A Figura 34 mostra a interface que permite atribuir nota aos requisitos específicos do projeto. Os requisitos primários e secundários terão suas notas calculadas a partir dos requisitos específicos.

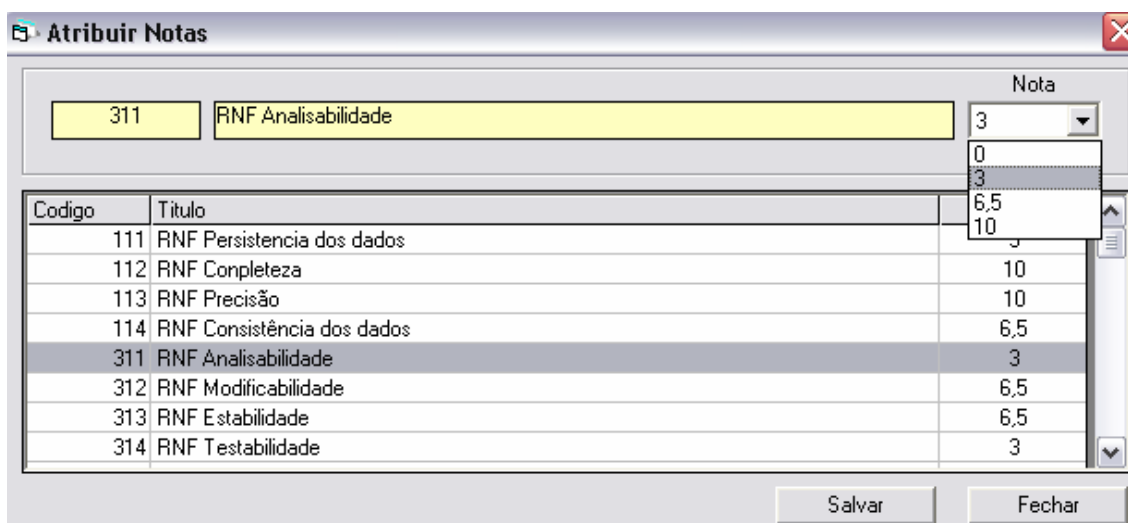


Figura 34 – Interface de Atribuição de Notas aos Requisitos Específicos

Quando o usuário aciona o botão Salvar na Figura 34, dispara o evento deste botão. Este evento invoca o método *AtribuirNota()* da classe *clsRequisitosSecundários*, que corresponde a execução do trecho de código ilustrado a baixo. O código mostra a atualização do requisito com a atribuição da nota..

```
Public Function AtribuirNota() As Boolean
    Set RS = New ADODB.Recordset
    sSQL = "UPDATE RNFSecundarios " & _
        "SET Nota = '" & sNota & "'" & _
        " WHERE cod_requisito = " & sCod_requisito
    RS.Open sSQL, oConexao, adOpenDynamic
    AtribuirNota = True
End Function
```

Neste capítulo abordamos sobre a ferramenta desenvolvida para automatizar o método proposto. Na seção 5.1 os principais casos de usos e o modelo de entidade relacionamentos da ferramenta foram mostrados. Na seção 5.2 as interfaces e fragmentos de código fonte foram ilustrados e comentados com o objetivo de facilitar o entendimento na

utilização da ferramenta. No próximo capítulo faremos uma conclusão do trabalho destacando os trabalhos futuros e nossas contribuições.

6 CONCLUSÃO

Nos últimos anos, temos observado um crescimento substancial do desenvolvimento da engenharia de requisitos. Atualmente várias propostas têm sido desenvolvidas para lidar com os requisitos funcionais de software de forma consistente, enquanto os requisitos não funcionais têm recebido pouca atenção. Entretanto, alguns autores destacam os RNFs como sendo de crucial importância para a obtenção de software que atendam as expectativas dos clientes (CHUNG 1993). De acordo com Boehm (1978) boa parte dos aspectos de qualidade do software é expressa por RNFs que, muitas vezes, são também denominados “atributos de qualidade do software”.

Por outro lado, da mesma forma que os requisitos funcionais, os RNFs estão em constante evolução. Contudo, é importante não apenas sermos capazes de detectar essa evolução, mas também de avaliar suas conseqüências sobre o software em questão. Mostramos neste trabalho uma estratégia que estabelece como elicitar RNFs a partir das necessidades de qualidade dos stakeholders, e avaliar a qualidade do software desde o início do processo de desenvolvimento. Propomos uma política de qualidade na qual se especifica a qualidade mínima desejada para aplicações de software médico. A estratégia proposta, quantifica a definição dos RNFs especificados no documento de requisitos e confronta com a política de qualidade.

No sentido de validar a estratégia, elaboramos um estudo de caso para avaliar se a utilização do método é capaz de garantir a conformidade dos RNFs elicitados para o software com a política de qualidade, mesmo quando novas tecnologias e novas funcionalidades forem adotadas. Simulamos uma situação para um SIH trabalhando de forma integrada com outros sistemas (laboratórios, internação, imagens etc..). Contudo, concluímos que:

- √ Ao quantificar o grau de conformidade das definições dos RNFs especificados no documento de requisitos de software, com a política de qualidade, possibilitamos uma visão antecipada da qualidade final do produto, bem como um menor tempo de entrega e menores custos de manutenção;
- √ Quando novas tecnologias e novas funcionalidades forem adotadas para o software, a política de qualidade deve ser revista e possivelmente alterada para responder aos possíveis conflitos introduzidos ou os RNFs devem ser revistos de modo a atender as necessidades de qualidade especificadas na política;
- √ Uma vez criada a política de qualidade para o software, ela deve continuar consistente com as definições dos RNFs especificadas no documento de requisitos, mesmo com a

adoção de novas tecnologia e funcionalidades ao software. Portanto, há uma necessidade de evoluir as abordagens para o desenvolvimento de políticas de qualidade de software.

6.1 Trabalhos Futuros

Para futuros trabalhos destacamos:

- ✓ Estender a implementação da ferramenta de suporte ao desenvolvimento do método, a qual poderia ser desenvolvida para automatizar todas as fases do método proposto inclusive o preenchimento do modelo da política de qualidade;
- ✓ Estender a abordagem proposta, incluindo a elicitação dos requisitos funcionais;
- ✓ Abordagens sobre a influência que os RNFs exercem na atividade de determinar qual a arquitetura de software que deverá ser utilizada, visto que muitos RNFs afetam diretamente o tipo de arquitetura que deverá ser utilizada caso eles venham a ser satisfeitos;
- ✓ A realização de novos estudos de casos, sem a participação dos autores deste trabalho, para verificar o grau de facilidade de aplicação da estratégia por outras pessoas.

6.2 Contribuições do Trabalho

Este trabalho aborda aspectos pouco tratados na literatura existente e reside numa proposta para lidar com requisitos não funcionais desde as primeiras etapas do processo de desenvolvimento de software. Seu objetivo foi propor um método voltado para sistemas médicos, em obediência com a política de qualidade. O principal foco foi garantir que tal política não se torne obsoleta com a aquisição de novas funcionalidades ao software existente. Deste modo, à medida que novas tecnologias ou funcionalidades são adotadas, os RNFs elicitados deverão permanecer em conformidade com a política de qualidade.

A estratégia classifica os requisitos não funcionais em primários, secundários e específicos, mostra como elicitar RNFs a partir do modelo de qualidade proposto Figura 9 e garante a conformidade deste requisitos com a política de qualidade definida para o software. A metodologia também se apresenta interessante, como um mecanismo de medição da qualidade de software em desenvolvimento ou mesmo consolidado, permitindo assim que a medição de um projeto sob a ótica de RNFs seja realizada mesmo com o projeto em andamento ou em sistemas legados que precisam evoluir.

Destacamos ainda que, embora o método proposto tenha sido criado sob a ótica de RNFs para aplicações médicas, nada impede que o mesmo seja empregado para aplicações de outros domínios. Nesse caso, será necessário reavaliar o modelo de qualidade proposto e política de qualidade para refletir as necessidades de qualidade para o domínio em questão.

Neste trabalho, também propomos um modelo para o documento de especificação de requisitos, com o objetivo de fornecer um meio padrão para especificar os requisitos elicitados, o qual poderá ser útil tanto para as equipes de desenvolvimento, na tentativa de facilitar a construção de sistemas, quanto para as equipes de análises, nas futuras manutenções corretivas ou evolutivas.

Finalmente, desenvolvemos uma ferramenta que implementa as fases do método proposto com o objetivo de ajudar os desenvolvedores a controlar os requisitos não funcionais elicitados de um determinado projeto, gerar a política de qualidade e as planilhas de requisitos não funcionais primários, secundários e específicos para ajudar a compor o documento de requisitos. Esperamos, com o desenvolvimento deste trabalho, ter contribuído para obtermos software que atendam melhor as expectativas de qualidade do cliente e do desenvolvedor.

REFERÊNCIAS

- ALEXANDRINI, Fabio. **Desenvolvimento de uma Metodologia de Interpretação, Recuperação e Codificação Inteligente de Laudos Médicos Independente de Idioma.** (Tese de Doutorado, Universidade Federal de Santa Catarina), Florianópolis, 2005.
- ALVES, C., **Seleção de Produtos de Software** “Utilizando uma Abordagem Baseada em Engenharia de Requisitos”. Dissertação de mestrado, Universidade Federal de Pernambuco, Centro de Informática, 2001.
- BASILI, V.R. and Musa, J. “*The Future Engineering of Software a Management Perspective*”, IEEE Computer 24 1991 pp:90-96
- BASTOS, Francisco e ABDELOUAHAB, Zair. “*Non Functional Requirement Elicitation in Agreement with Standardizing and Quality Policies for Medical Software*”. ICSSEA - 19th International Conference, Paris, France, 2006.
- BOEHM B., **Software engineering.** IEEE Transactions on Computer. 1976.
- BOEHM, B. “*Characteristics of Software Quality*” North Holland Press, 1978.
- BROWN, A. W., Wallnau, K. C., **The current State of CBSE, IEEE Software.** Sep/Oct. 1998.
- CAMARGO, Leoleli. Medicina na Rede; **Jornal Zero Hora**; 15/06/2002.
- CARVALHO, M. C. Abdelouahab, Z., **Um Método para Elicitação e Modelagem de Requisitos Baseado em Objetivos.** Workshop de Engenharia de Requisitos, 4., Buenos Aires, Argentina, 2001. p. 319-337.
- CASTRO, Jaelson; F. B. Victor F.A. SANTANDER.. **Desenvolvendo Use Cases a partir de Modelagem Organizacional.** UFPE, 2001
- CHU, S.S. **Information Retrieval and health / clinical management.** Yearbook of Medical Informatics, 2002.
- CHUNG L., “*Representing and Using Non-Functional Requirements: A Process Oriented Approach*” Ph.D. Thesis, Dept. of Comp. Science. University of Toronto, June 1993. Also tech. Rep. DKBS-TR-91-1.
- CHUNG, L., NIXON, B. “*Dealing with Non-Functional Requirements: Three Experimental Studies of a Process-Oriented Approach*” Proc. 17th Int. Con. on Software Eng. Seattle, Washington, April pp: 24-28, 1995.
- CHUNG, L., NIXON, B., Yu, E. and Mylopoulos, J. “*Non-Functional Requirements in Software Engineering*” Kluwer Academic Publishers 1999.
- CYSNEIROS, L.M. and LEITE, J.C.S.P. “*Integrating Non-Functional Requirements into data model*” 4th International Symposium on Requirements Engineering – Ireland June 1999.

CYSNEIROS, L.M. and Leite, J.C.S.P. **Requisitos não Funcionais: da Elicitação ao Modelo Conceitual** (Tese de Doutorado), PUC/RJ, 2002.

DAVIS, A. *"Software Requirements: Objects Functions and States"* Prentice Hall, 1993.

FINKELSTEIN, Anthony. *Requirements Engineering: a review and research agenda*. City University, Department of computer Science, London EC1V 0HB. 1995.

FOOK, K. DONATO, **Método para Reutilização de Objetivos na Engenharia de Requisitos**. (Dissertação de Mestrado), Universidade Federal do Maranhão, 2001.

HSIA, Pei; DAVIS; Alan and KUNG, David. *Status Report: Requirements Engineering*. IEEE Software. November, 1993.

IEEE Std 830, IEEE. *Recommended Practice for Software Requirements Specifications*. 1998.

ISO/IEC 9126, International Standard Information technology – Software product evaluation – Quality characteristics and guidelines for their use. 1991.

O. JAIME, **O Registro Clínico Computadorizado: Padronização e Codificação**”, Núcleo de Informática Biomédica da Universidade Estadual de Campinas, *Revista Informédica*, 1 (5): 5-8, 1993

KELLER, S.E. et al *"Specifying Software Quality Requirements with Metrics"* in *Tutorial System and Software Requirements Engineering* IEEE Computer Society Press 1990 pp:145-163

KOTONYA, G.; SOMMERVILLE, I. *Requirements Engineering With Viewpoints*. *Software Engineering Journal*, v. 11, n 1, p. 5-11, jan. 1996.

LEÃO, Beatriz et all. **Manual de Requisitos de Segurança, Conteúdo e Funcionalidades para Sistemas de Registro Eletrônico em Saúde**. Versão 2.1, São Paulo, Fevereiro 2004.

LEITE, J, Gustavo Rossi, Vanesa Maiorana, Federico Balaguer, Gladys Kaplan, Graciela Hadad and Alejandro Oliveros, *Enhancing a Requirements Baseline with Scenarios*, *Requirements Engineering Journal*: Vol. 2, N.4, pp. 184 - 198, Springer-Verlag, London. 1997.

LOUCOPOULOS, P. e Karakostas, V. *System Requirements Engineering*. London McGraw-Hill, 1995.

MACEDO, N.M.; LEITE, J.C.S.P *Integrando Requisitos Não Funcionais aos Requisitos Baseados em Ações Concretas*. **Anais** do Workshop Iberoamericano de Engenharia de Requisitos e Ambientes de Software, CYTED & Instituto Tecnológico de Costa Rica (TEC) pp. 1-9,(1999).

MASSAD, E., et al. **O Prontuário Eletrônico do Paciente na Assistência, Informação e Conhecimento Médico**. 2003.

NUSEIBEH, B. Easterbrook, S. *Requirements Engineering: A Roadmap*. Conferencia Internacional de Engenharia de Software, 22, Limerick, Irland, 2000.

PAULA F, PÁDUA de Wilson. **Engenharia de Software** – fundamentos, métodos e padrões. 1ª edição, LTC – livros técnicos e Científicos Editora SA., 2001.

PISA, Ivan et all. MIDster: **Sistema Distribuído de Imagens Médicas Baseado em Modelos Peer-to-Peer e Serviços Web**. UNIFESP 2004.

PRESSMAN, R., *Software Engineering: A Practioner's Approach*. McGraw-Hill, 2000.

REIS, Tiago Pessoa Costa. **REQE – Uma Metodologia para Medição de Qualidade de Aplicações Web na Fase de Requisitos**. (Dissertação de Mestrado) Universidade Federal de Pernambuco, 2004.

ROCHA, Ana et all. **“Qualidade de Software Médico”**, UFRJ. 2001.

ROCHA, Simara e ABDELOUAHAB, Zair. *Requirement Elicitation Based on Goals with Security and Privacy Policies in Electronic Commce*. MA. 2004.

ROMA, G. *“A Taxonomy of Current Issues in Requirements Engineering”*, IEEE Computer Vol. 18, No. 4 Apr. 1985, pp:14-23

SILVA, F.C. et al. **Padronização de Vocabulário Médico**. 2003. Disponível em < ([http://www.virtual.epm.br/material/tis/curr-med/temas/med5/med5t41999/vocabula/ créditos .htm](http://www.virtual.epm.br/material/tis/curr-med/temas/med5/med5t41999/vocabula/creditos.htm))> acesso em: 03 mai/2006.

SOMMERVILLE, I. *Software Engineering* fourth edition, Addison-Wesley, 1992.

SOMMERVILLE, Ian and SAWYER, Pete. *Requirements Enginnering: A good practice guide*. John Wiley & Sons. 1998.

SOMMERVILLE, I; **Engenharia de Software**; tradução André Ribeiro; 6a. Edição, Addison Wesley; São Paulo, ISBN 85-88639-07-6, 2003.

Livros Grátis

(<http://www.livrosgratis.com.br>)

Milhares de Livros para Download:

[Baixar livros de Administração](#)

[Baixar livros de Agronomia](#)

[Baixar livros de Arquitetura](#)

[Baixar livros de Artes](#)

[Baixar livros de Astronomia](#)

[Baixar livros de Biologia Geral](#)

[Baixar livros de Ciência da Computação](#)

[Baixar livros de Ciência da Informação](#)

[Baixar livros de Ciência Política](#)

[Baixar livros de Ciências da Saúde](#)

[Baixar livros de Comunicação](#)

[Baixar livros do Conselho Nacional de Educação - CNE](#)

[Baixar livros de Defesa civil](#)

[Baixar livros de Direito](#)

[Baixar livros de Direitos humanos](#)

[Baixar livros de Economia](#)

[Baixar livros de Economia Doméstica](#)

[Baixar livros de Educação](#)

[Baixar livros de Educação - Trânsito](#)

[Baixar livros de Educação Física](#)

[Baixar livros de Engenharia Aeroespacial](#)

[Baixar livros de Farmácia](#)

[Baixar livros de Filosofia](#)

[Baixar livros de Física](#)

[Baixar livros de Geociências](#)

[Baixar livros de Geografia](#)

[Baixar livros de História](#)

[Baixar livros de Línguas](#)

[Baixar livros de Literatura](#)
[Baixar livros de Literatura de Cordel](#)
[Baixar livros de Literatura Infantil](#)
[Baixar livros de Matemática](#)
[Baixar livros de Medicina](#)
[Baixar livros de Medicina Veterinária](#)
[Baixar livros de Meio Ambiente](#)
[Baixar livros de Meteorologia](#)
[Baixar Monografias e TCC](#)
[Baixar livros Multidisciplinar](#)
[Baixar livros de Música](#)
[Baixar livros de Psicologia](#)
[Baixar livros de Química](#)
[Baixar livros de Saúde Coletiva](#)
[Baixar livros de Serviço Social](#)
[Baixar livros de Sociologia](#)
[Baixar livros de Teologia](#)
[Baixar livros de Trabalho](#)
[Baixar livros de Turismo](#)