



Pós-Graduação em Ciência da Computação

MILTON DEIVSON ALBUQUERQUE CAVALCANTE

**UMA ANÁLISE SOBRE A ENGENHARIA DE REQUISITOS EM SISTEMAS
EMBARCADOS**

Dissertação de Mestrado



Universidade Federal de Pernambuco
posgraduacao@cin.ufpe.br
www.cin.ufpe.br/~posgraduacao

RECIFE
2017

MILTON DEIVSON ALBUQUERQUE CAVALCANTE

**UMA ANÁLISE SOBRE A ENGENHARIA DE REQUISITOS EM SISTEMAS
EMBARCADOS**

*Trabalho apresentado ao Programa de Pós-graduação em
Ciência da Computação do Centro de Informática da Univer-
sidade Federal de Pernambuco como requisito parcial para
obtenção do grau de Mestre em Ciência da Computação.*

Orientador: *Jaelson Freire Brelaz de Castro*

Co-Orientador: *Fernanda Maria Ribeiro de Alencar*

RECIFE

2017

Catálogo na fonte
Bibliotecária Elaine Cristina de Freitas CRB 1790

C376a Cavalcante, Milton Deivson Albuquerque
Uma análise sobre a engenharia de requisitos em sistemas embarcados
/ Milton Deivson Albuquerque Cavalcante – 2017.
108 f.: fig., tab.

Orientador: Jaelson Freire Brelaz de Castro
Dissertação (Mestrado) – Universidade Federal de Pernambuco. Cln.
Ciência da Computação, Recife, 2017.
Inclui referências e apêndices.

1. Engenharia de Software 2. Engenharia de Requisitos 3. Sistemas
Embarcados . I. Castro, Jaelson Freire Brelaz de (Orientador). II. Título.

005.1 CDD (22. ed.) UFPE-MEI 2017-277

MILTON DEIVSON ALBUQUERQUE CAVALCANTE

**UMA ANÁLISE SOBRE A ENGENHARIA DE REQUISITOS EM SISTEMAS
EMBARCADOS**

Dissertação apresentada ao Programa de Pós-Graduação
em Ciência da Computação da Universidade Federal de
Pernambuco, como requisito parcial para a obtenção do
título de Mestre Profissional em 02 de março de 2017.

Aprovado em: ____/____/____

Prof. Carla Taciana Lima Lourenço Silva Schuenemann
Centro de Informática/UFPE

Prof. João Henrique Correia Pimentel
Universidade Federal Rural de Pernambuco

Prof. Jaelson Freire Brelaz de Castro
Centro de Informática/UFPE
(Orientador)

RECIFE
2017

*A todos aqueles que de alguma forma estiveram estão
próximos de mim, principalmente minha família e minha
noiva Juliana Alencar que trouxeram paz nos momentos de
angústia e me encorajaram diante dos inúmeros
obstáculos.*

Agradecimentos

É difícil gratular a todos que de alguma forma, nos momentos alegres e ou de aflições fizeram ou fazem parte de minha vida, por essa razão agradeço a todos com carinho.

Agradeço de maneira especial a minha família, que carrego como suporte para todos os momentos de minha vida, a meu pai, Milton Araújo e minha mãe, Maria Soelene que sempre me proporcionaram acesso ao conhecimento e me incentivaram desde criança a importância da educação. Ainda de forma especial, quero agradecer a minha noiva, Juliana Alencar, pelos incentivos e compreensão pelas ausências decorrentes desta jornada.

Imensa gratidão a minha tia Maria José por todo o suporte oferecido na cidade de Recife-PE, pelos momentos de conversas, lembranças familiares, por todos os alertas e cuidados diários para o percurso até a universidade.

Sou grato a comunidade do IFSertão-PE Campus Petrolina Zona Rural, pelo aporte e confiança para conclusão desta etapa da minha vida.

Agradeço aos amigos pelo companheirismo e bons momentos de distrações. Manifesto uma gratidão especial ao meu amigo, Tarcísio Couto, que me ajudou significativamente na construção deste trabalho, foi meu coorientador de maneira informal, mas compartilhou seu conhecimento e muitas horas de trabalhos juntos.

Agradeço a todos os amigos da Turma MPROF[GTI] pelo companheirismo, momentos de alegrias, sabedorias compartilhadas ao longo destes dois anos e pelo empenho de todos nós profissionais que tivemos jornada dupla (trabalho e estudo), mas conseguimos vencer essa batalha.

Agradeço aos meus Orientadores professor Jaelson Castro e professora Fernanda Alencar por toda ajuda, direcionamento, conhecimento compartilhado durante as aulas e pela missão da orientação à distância, todas de grande importância para o desenvolvimento deste trabalho.

O Último e mais importante agradecimento. Muito obrigado meu Deus, criador de todas as coisas, detentor de todo o conhecimento, mantenedor da vida, conhecedor de todo o esforço e caminhada que trilhei até esse momento e que a mim atribuiu alma e missões pelas quais já sabia que eu iria batalhar e vencer. A vida é o meu maior agradecimento!

*“Que os vossos esforços desafiem as impossibilidades, lembrai-vos de que
as grandes coisas do homem foram conquistadas do que parecia
impossível. ”*
(Charles Chaplin)

Resumo

Os Sistemas Embarcados (SE) são amplamente difundidos no nosso cotidiano e estão cada vez mais presentes nas vidas das pessoas. Eles vêm sendo usados em vários domínios, incluindo indústria de telecomunicações, tecnologia de automação, tecnologia médica, automotiva e aeroespacial. A maioria desses sistemas opera em ambientes críticos, que em caso de falhas pode causar graves danos materiais e acidentes envolvendo pessoas. Portanto, para investigar e levantar evidências na Engenharia de Requisitos (ER) no domínio de SE, foi realizada uma Revisão Sistemática da Literatura (RSL), que abrange o período de 1970 a junho de 2016. Deste modo, podemos sintetizar e analisar esses dados disponíveis na literatura formando uma base de conhecimento e fornecendo um caminho para pesquisas futuras voltadas à comunidade de ER. O nosso objetivo foi compreender: (1) como as abordagens contribuem para a solução de problemas de ER no domínio de SE; (2) quais as fases do processo de ER foram apoiadas pelas abordagens; (3) que tipo de requisito (funcional e/ou não-funcional) foram apoiados pelas abordagens; (4) quais as questões em aberto das abordagens encontradas; (5) quais são os domínios que as abordagens suportam; e (6) quais requisitos devem ser considerados durante o desenvolvimento de SE. Observamos que no desenvolvimento de SE, o processo de ER é geralmente inadequado. Entre as dificuldades, isto é, o problema em aberto mais citado, está o tratamento dos Requisitos Não-Funcionais (RNFs), já que não há consenso na comunidade de ER de como descrever e especificar estes requisitos. Em vista disso, foi realizada uma análise complementar dos dados extraídos da RSL para obter informações e responder como a comunidade de desenvolvimento de SE está lidando com os RNFs na fase de especificação do processo de ER. Portanto, foram feitas análises gerais tais como Tipo de contribuição, Tipo da pesquisa, Método de Pesquisa, Contexto da aplicação, bem como perguntas específicas referentes aos RNFs em SE.

Palavras-chave: Engenharia de Requisitos. Sistemas Embarcados. Especificação de requisitos. Requisitos Não-Funcionais.

Abstract

Embedded Systems (ES) are widespread in our daily lives and are increasingly present in people's lives. They have been used in several domains, including telecommunications industry, automation technology, medical technology, automotive and aerospace. Most of these systems operate in critical environments, which in case of failures can cause severe material damage and accidents involving people. Therefore, to investigate and raise evidence in Requirements Engineering (RE) for ES, a Systematic Literature Review (RSL) was carried out, covering studies from 1970 to June 2016. We synthesized and analyzed the data available in the literature forming a knowledge base and providing a way for future research. Our goal was to understand: (1) how are the approaches contributing to the solution of RE problems in the ES domain; (2) which phases of the RE process were supported by the approaches; (3) what kind of requirements (functional and / or non-functional) were supported by the approaches; (4) what are the open problems of the approaches; (5) what are the domains that the approaches support; and (6) what requirements should be considered during ES development. We have observed that during the development of ES, the RE process is often inadequate. Among the difficulties, i.e. the most cited open problem, there is the treatment of Non-Functional Requirements (NFRs), given that there is no consensus on RE community on how to describe and specify these requirements. Therefore, a complementary analysis of the data extracted from the SLR was carried out to obtain information and to answer how the ES development community is dealing with the NFRs in the specification phase of the RE process. Thus, an analysis has performed including information such as the type of contribution, kind of research, research method, application context, as well as specific questions regarding NFRs in ES.

Keywords: Requirements Engineering. Embedded Systems. Specification of Requirements. Non-Functional Requirements.

Lista de Figuras

1.1	Organização da pesquisa	19
2.1	Leitores dos diferentes tipos de especificação de requisitos	22
2.2	Classificação do requisitos não-funcionais	25
2.3	Tipos de <i>softgoals</i>	28
2.4	Modelo de atividades do processo de ER	29
2.5	Ciclo da Engenharia de Requisitos	31
3.1	Processo de revisão sistemática da literatura.	44
3.2	Visão temporal dos estudos.	47
3.3	País de autoria dos estudos.	48
3.4	Tipo de publicação dos estudos.	49
3.5	Métodos de pesquisa dos estudos	51
3.6	Dimensão dos métodos de pesquisa aplicada nos estudos	51
3.7	Problemas de ER nos sistemas embarcados	53
3.8	Domínios de sistemas embarcados.	58
4.1	Tipo dos estudos selecionados	68
4.2	Método de pesquisa dos estudos das revisões.	69
4.3	Contexto de aplicação dos estudos.	70
4.4	Domínios de sistemas embarcados.	73
4.5	Tipos de sistemas e RNFs relevantes	75
4.6	Problemas comuns encontrados na especificação de RNFs nos SE.	77
D.1	Análise de Qualidade 1-24	106
D.2	Análise de Qualidade 25-48	107
D.3	Análise de Qualidade 26-71	108
D.4	Análise de Qualidade 72-75	108

Lista de Tabelas

1.1	Procedimentos metodológicos adotados	18
3.1	Questões de pesquisa e motivações	38
3.2	Técnica de busca automática	40
3.3	Base de dados pesquisados	42
3.4	Seleção de estudos (Inclusão e Exclusão)	43
3.5	Questões de avaliação de qualidade	45
3.6	Tipo de contribuição do estudo	50
3.7	Fases da ER suportadas pelos estudos	54
3.8	Abordagem dos requisitos funcionais e não-funcionais	55
3.9	Problemas em aberto da ER nos estudos selecionados	57
3.10	Requisitos em diferentes ponto de vista	59
4.1	Estudos selecionados sobre os RNFs dos SE	63
4.2	Questões de pesquisa	65
4.3	Tipo de contribuição do estudo	67
4.4	Linguagens dos estudos sobre RNFs	71
4.5	Principais RNFs das abordagens	72
4.6	Domínios das abordagens sobre RNFs dos SE	73
4.7	Problemas em aberto dos estudos sobre RNFs	77

Lista de Acrônimos

AQ	Avaliação de Qualidade
ER	Engenharia de Requisitos
ES	Engenharia de Software
IEEE	Instituto de Engenheiros Eletricistas e Eletrônicos
ISO	Organização Internacional de Normalização
PA	Problemas em Aberto
RNFs	Requisitos Não-Funcionais
RSL	Revisão Sistemática da Literatura
SE	Sistemas Embarcados

Sumário

1	INTRODUÇÃO.....	14
1.1	Caracterização e Justificativa do Problema.....	14
1.2	Objetivos da Investigação.....	16
1.3	Procedimentos Metodológicos.....	17
1.4	Etapas da Pesquisa.....	18
1.5	Estrutura da Dissertação.....	19
2	REFERENCIAL TEÓRICO.....	21
2.1	Requisitos e seus tipos.....	21
2.1.1	Requisitos não-funcionais.....	23
2.1.1.1	<i>NFR Framework (conceitos principais).....</i>	<i>27</i>
2.2	Engenharia de Requisitos.....	28
2.3	Sistemas Embarcados.....	31
2.3.1	Requisitos de sistemas embarcados.....	33
2.4	Considerações Finais.....	35
3	UMA REVISÃO SISTEMÁTICA DA ENGENHARIA DE REQUISITOS PARA SISTEMAS EMBARCADOS.....	36
3.1	Introdução.....	36
3.2	Planejamento.....	37
3.2.1	Elaboração do protocolo.....	37
3.2.2	Questões de pesquisa.....	37
3.3	Condução da Revisão.....	39
3.3.1	Técnica de pesquisa.....	39
3.3.2	Origem dos dados.....	41
3.3.3	Seleção de estudos.....	42
3.3.3.1	<i>Crítérios de inclusão e exclusão.....</i>	<i>42</i>
3.3.4	Análise de qualidade.....	44
3.3.5	Síntese e análise dos dados.....	46
3.4	Resultados e Discussão.....	46
3.4.1	Visão geral e características dos estudos.....	46
3.4.1.1	<i>Visão temporal dos estudos.....</i>	<i>46</i>
3.4.1.2	<i>Origem dos estudos.....</i>	<i>47</i>
3.4.1.3	<i>Tipo do estudo.....</i>	<i>48</i>

3.4.1.4	<i>Tipo de contribuição</i>	49
3.4.1.5	<i>Método de pesquisa</i>	50
3.4.2	Q1 - Como as abordagens contribuem para a solução de problemas de engenharia de requisitos no domínio de sistemas embarcados?.....	52
3.4.3	Q2 - Quais as fases do processo de ER apoiadas pelas abordagens?.....	53
3.4.4	Q3 - Que tipo de requisito (funcional e/ou não-funcional) foi apoiado pelas abordagens?.....	54
3.4.5	Q4 - Quais as questões em aberto das abordagens encontradas?.....	55
3.4.6	Q5 - Quais são os domínios que as abordagens suportam?.....	57
3.4.7	Q6 - Quais requisitos devem ser considerados durante o desenvolvimento de sistemas embarcados?.....	58
3.4.8	Ameaças à validade.....	60
3.4.9	Estudos relacionados.....	61
3.5	Considerações Finais	61
4	ANÁLISE DOS RNFS NOS SISTEMAS EMBARCADOS	63
4.1	Introdução	64
4.2	Questões de pesquisa específicas	65
4.3	Visão geral	66
4.3.1	Tipo de contribuição.....	66
4.3.2	Tipo da pesquisa.....	67
4.3.3	Método de pesquisa.....	68
4.3.4	Contexto de aplicação.....	69
4.4	Q1 - Quais as linguagens usadas para a especificação dos RNFs nos sistemas embarcados?	70
4.5	Q2. Quais os RNFs mais frequentes e seus respectivos domínios nas abordagens dos sistemas embarcados?	71
4.6	Q3. Quais os problemas em aberto mais frequentes encontrados na especificação de RNFs dos sistemas embarcados?	75
4.7	Considerações Finais	77
5	CONCLUSÃO	79
5.1	Contribuições	79
5.2	Limitações do Trabalho	80
5.3	Trabalhos Futuros	81
	REFERÊNCIAS	82
	APÊNDICES	86

APÊNDICE A - GLOSSÁRIO DE TERMINOLOGIAS DE SISTEMAS EMBARCADOS.....	86
APÊNDICE B - PROTOCOLO DA REVISÃO SISTEMÁTICA DA LITERATURA DE ENGENHARIA DE REQUISITOS NOS SISTEMAS EMBARCADOS.....	93
APÊNDICE C - IDENTIFICAÇÃO DOS ESTUDOS DA RSL.....	101
APÊNDICE D - RESULTADOS DA AVALIAÇÃO DE QUALIDADE DA RSL.....	105

1

INTRODUÇÃO

Este capítulo descreve a caracterização e justificativa do problema e os objetivos da investigação deste estudo. Apresenta também os procedimentos metodológicos que serão utilizados, as etapas da pesquisa e sua estrutura distribuída em capítulos.

1.1 Caracterização e Justificativa do Problema

O Sistemas Embarcados (SE) na concepção de BRAUN et al. (2014), são sistemas que foram construídos usando recursos e técnicas específicas para operarem em um ambiente físico e tecnológico. No cotidiano, a sociedade é dependente de vários serviços suportados por SE, muitos deles são transparentes para o usuário final e estão cada vez mais presentes nas vidas das pessoas. Podemos destacar os sistemas embarcados de TVs, controle de tráfego aéreo, monitoramento de estradas, controle do sistema de telecomunicações, controle de equipamentos médicos e sistemas de controle automotivo (MARTINS; OLIVEIRA, 2014a). Esses sistemas e serviços são conhecidos por sua alta complexidade, causada pelo crescente número de funções e pelo crescente número de interações entre elas e o meio o qual o sistema está inserido.

De acordo com SIKORA; TENBERGEN; POHL (2012), os requisitos e funcionalidades dos SE tornam grave muitos problemas que são comuns na engenharia de *software*, visto que requerem atenção prioritária no início do processo de Engenharia de Requisitos (ER) devido a sua elevada complexidade e o risco a qual alguns domínios submetem aos usuários. Ainda no entendimento de SIKORA; TENBERGEN; POHL (2012), a diferença no contexto dos SE, é que os problemas podem ser críticos tanto na execução das funcionalidades do sistema, como na captura equivocada dos requisitos.

Ainda nesse contexto, MARTINS; OLIVEIRA (2014b) acrescenta que, o crescente uso dos SE, a grande complexidade e criticidade de suas funções, a rápida evolução de seus componentes eletrônicos, bem como seu alto grau de confiabilidade e segurança, são aspectos que propõem novos desafios para a ER. Dessa forma, profissionais e engenheiros da área de sistemas embarcados têm buscado abordagens, técnicas, métodos, processos e ferramentas

da engenharia de requisitos, na tentativa de adaptar e aproveitar o que já existe para encontrar novas técnicas que levem em consideração as peculiaridades dos SE, tendo em vista que há uma escassez nessa área (OSSADA, 2010).

Portanto, para investigar e levantar evidências na engenharia de requisitos no domínio de sistemas embarcados, foi realizada uma Revisão Sistemática da Literatura (RSL), que abrange o período de 1970 a junho de 2016, ou seja, os resultados levam em consideração aproximadamente 4 décadas de estudos sobre ER nos sistemas embarcados. Esse período foi escolhido para obter dados de toda evolução da ER nos SE, e, assim, poder sintetizar e analisar esses dados disponíveis na literatura para formar uma base de conhecimento e fornecer um caminho para pesquisas futuras voltadas à comunidade de ER.

Nosso objetivo foi compreender: (1) como as abordagens contribuem para a solução de problemas de engenharia de requisitos no domínio de sistemas embarcados; (2) quais as fases do processo de RE foram apoiados pelas abordagens; (3) que tipo de requisitos (funcional e/ou não-funcional) foram apoiados pelas abordagens; (4) quais as questões em aberto das abordagens encontradas; (5) quais são os domínios que as abordagens suportam; e (6) quais requisitos devem ser considerados durante o desenvolvimento de sistemas embarcados.

Segundo FREITAS et al. (2006), hoje em dia, uma importante preocupação que envolve o projeto desses sistemas embarcados é como lidar com os requisitos e em especial os requisitos não-funcionais. Portanto, se os RNFs não forem devidamente especificados e caracterizados, tornam-se responsáveis por códigos emaranhados, ausência de coesão e problemas críticos que podem ocasionar grandes perdas materiais, risco de danos físicos e compromete todo o projeto.

Ainda nesse contexto, as comunidades de desenvolvimento de *software* perceberam que os RNFs devem ser levados em consideração quanto a sua identificação e especificação o mais cedo possível para melhorar o projeto e a segurança do sistema. Esse fato, motiva e força a separação das preocupações para as fases iniciais do projeto (RASHID et al., 2002). Continuando, RASHID et al. (2002), define também que os sistemas embarcados possuem RNFs muito importantes, que são a preocupação com requisitos temporais, precisão na execução da sua funcionalidade, integração entre *hardware* e *software* e principalmente descrição de aspectos de segurança. Desse modo, a complexidade referente a análise não-funcional aumenta quando esses sistemas são de domínio crítico, pois para lidar com esses requisitos não-funcionais, os desenvolvedores se deparam com descrições na maioria das vezes subjetivas e incompletas. Há uma escassez de métodos, processos e padrões formais específicos para descrever e especificar os RNFs dos SE.

Diante desse cenário, nota-se que no desenvolvimento de sistemas embarcados, que o processo de ER encontra-se em um estado inicial devido a falta de processos, técnicas e ferramentas voltadas às especificidades dos SE. É necessário uma boa definição dos requisitos não-funcionais e o desafio é especificá-los sem existir um padrão formal e um consenso da comunidade dos embarcados sobre essa questão. Em vista disso, foi realizada uma análise

complementar dos dados extraídos da RSL para obter informações e responder como a comunidade de desenvolvimento de sistemas embarcados está lidando com os RNFs na fase de especificação do processo de ER. Portanto, foram feitas análises gerais tais como Tipo de contribuição, Tipo da pesquisa, Método de Pesquisa, Contexto da aplicação, bem como perguntas específicas referentes aos RNFs em SE, como, (1) quais as linguagens usadas para a especificação dos RNFs nos sistemas embarcados; (2) Quais os RNFs mais frequentes e seus respectivos domínios nas abordagens dos sistemas embarcados; e (3) quais os problemas em aberto mais frequentes encontrados na especificação de RNFs dos sistemas embarcados.

1.2 Objetivos da Investigação

■ Objetivo Geral

O objetivo central desta pesquisa é investigar, encontrar e analisar evidências na literatura sobre a aplicação da engenharia de requisitos no desenvolvimento de sistemas embarcados, e, assim, fornecer uma base de conhecimento para pesquisas futuras à comunidade de ER.

■ Objetivos Específicos

Para alcançar os objetivo central deste trabalho, foram derivados os seguintes objetivos:

- 1) Identificar e compreender como as abordagens contribuem para a solução de problemas de engenharia de requisitos no domínio de sistemas embarcados;
- 2) Identificar e apresentar quais as fases do processo de ER foram apoiados pelas abordagens;
- 3) Identificar quais os principais tipos de requisitos são tratados no desenvolvimento de sistemas embarcados;
- 4) Identificar quais são os domínios que as abordagens suportam;
- 5) Apresentar quais os principais problemas em aberto, relativos à engenharia de requisitos no domínio de sistemas embarcados;
- 6) Identificar a linguagens usadas para a especificação dos RNFs nos sistemas embarcados;
- 7) Identificar os RNFs mais frequentes e seus respectivos domínios nas abordagens dos sistemas embarcados;
- 8) Identificar os problemas em aberto mais frequentes encontrados na especificação de RNFs dos sistemas embarcados;

1.3 Procedimentos Metodológicos

De acordo com a concepção de que método é um procedimento ou meio de como atingir um determinado objetivo e que o fito da ciência é a busca do conhecimento, é aceitável dizer que o método científico é um conjunto de procedimentos seguidos para alcançar o conhecimento.

Nessa perspectiva, FERRARI (1974) define que o método científico é um meio típico da ciência, ou seja, um instrumento básico que ordena, inicialmente, o pensamento em sistemas e define os procedimentos do cientista ao longo do caminho até alcançar o objetivo científico proposto.

Na visão de LAKATOS E.V. MARCONI (2006), os métodos científicos não são uma exclusividade da ciência, pois também são utilizados para solução de problemas comuns do cotidiano, todavia o autor reforça que sem o emprego dos métodos científicos não há ciência.

Vários pensadores tentaram propor um único método que fosse utilizado por todas as ciências, no entanto, essas tentativas resultaram em pensamentos diferentes e conflitantes entre si. Nos dias de hoje, já é possível a utilização desses diferentes métodos e até mesmo a combinação deles, este fato vai depender do objetivo da investigação e do tipo da pesquisa.

À vista do exposto, com intuito de captar, conhecer e sintetizar dados e evidências sobre o processo de engenharia de requisitos nos sistemas embarcados, foi realizada uma revisão sistemática da literatura e conseguinte uma análise dos dados encontrados. Segundo TORRE-UGARTE et al. (2011), a RSL é uma metodologia rigorosa que propõe identificar os estudos sobre um tema em questão, aplicando métodos explícitos e sistematizados de busca; avaliar a qualidade e validade desses estudos. Cada uma dessas características da RSL são sistematizadas por meio de um protocolo conforme diretrizes propostas por PETERSEN et al. (2008), que levam em consideração critérios que os validam, para minimizar o viés e outorgar qualidade à metodologia.

A Tabela 1.1 apresenta os procedimentos e técnicas utilizados pela pesquisa para alcançar o objetivo geral de acordo com LAKATOS; ANDRADE MARCONI (2010), ou seja, como a pesquisa foi realizada em relação a descrição da natureza, a forma de abordagem, quanto aos objetivos e aos procedimentos técnicos.

Tabela 1.1 Procedimentos metodológicos adotados

Critério	Classificação	Descrição
Natureza da pesquisa	Aplicada	Pesquisa aplicada é feita para resolver questões específicas e práticas.
Objetivo da pesquisa	Exploratória	Pesquisa conduzida para esclarecer e definir a natureza do problema
Procedimento metodológico	Bibliográfica (Revisão Sistemática da Literatura)	Uso de Materiais Publicados; Uso de uma pesquisa para reunir fatos, opiniões e atividades. Investiga um fenômeno contemporâneo dentro de seu contexto real.
Natureza dos dados	Qualitativos e quantitativos	Os dados qualitativos são utilizados para compreender situações, opiniões e motivações, enquanto os quantitativos são utilizados para medições, análises estatísticas, comprovações objetivas e descobrir fatos sobre fenômenos sociais.

A natureza da pesquisa: Este estudo é de pesquisa aplicada, pois objetiva gerar conhecimento e soluções a problemas reais e específicos. Sendo assim, a pesquisa busca por meio da literatura aplicar conhecimentos advindo de diversas fontes a entidades de interesse particular e público no âmbito da engenharia de requisitos nos sistemas embarcados.

Objetivo da pesquisa: Este critério tem característica exploratória, pois está relacionado ao levantamento de informações, classificação, verificação, análise e interpretação do fenômeno observado. É investigado abordagens e conceitos relacionados a engenharia de requisitos nos SE e na especificação de RNFs desses sistemas. Portanto, esses conceitos podem ser usados para melhorar o entendimento da importância da ER nos sistemas embarcados, motivar por meio do levantamento de dados relevantes a comunidade dos embarcados aprofundar estudos e pesquisas sobre esse assunto.

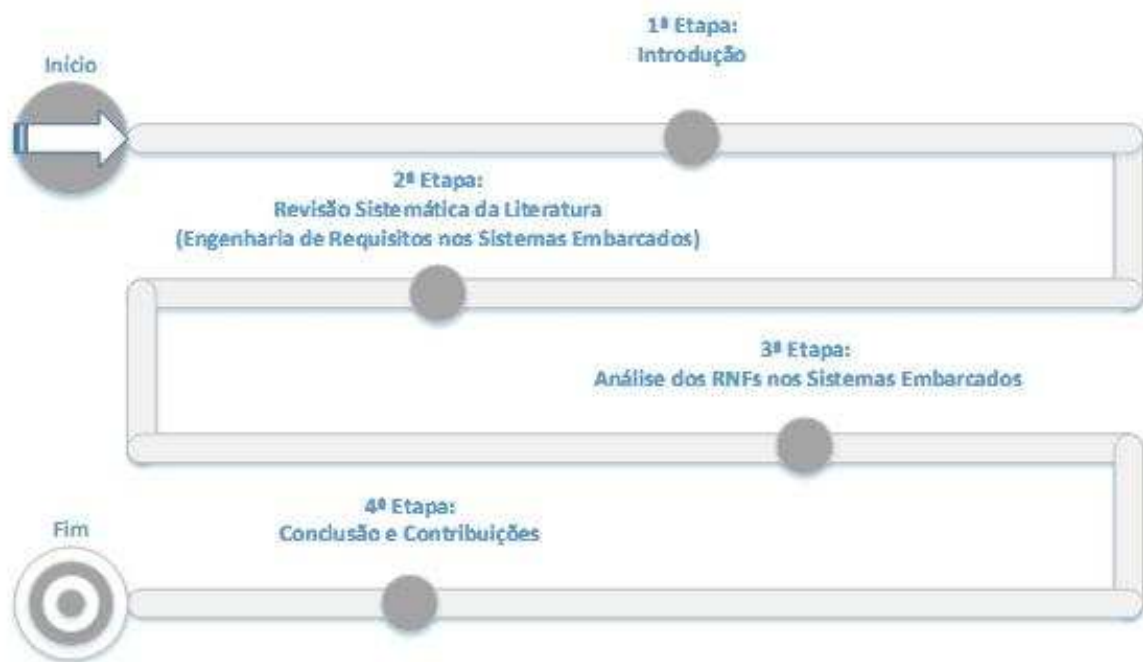
Procedimento metodológico: Os resultados obtidos são associados a materiais publicados, artigos científicos, revistas, e relatórios técnicos. Os dados extraídos por meio de uma RSL serão mapeados, classificados e analisados para cumprir o objetivo deste trabalho.

Natureza dos dados: A pesquisa procura fundamentar verdades de interpretações por meio de dados quantitativos levantados pela revisão (RSL) e avaliações de qualidade dos dados obtidos, ou seja, a forma de abordagem desta pesquisa é quantitativa e qualitativa.

1.4 Etapas da Pesquisa

Para atingir os objetivos levantados, esta pesquisa foi organizada em 4 etapas conforme Figura 1.2:

Figura 1.1 Organização da pesquisa



Fonte: Autoria própria

1ª Etapa: Essa etapa apresenta toda parte introdutória da pesquisa, bem como seus objetivos e procedimentos metodológicos.

2ª Etapa: Desenvolvimento de uma Revisão Sistemática da Literatura que consta o estado da arte referente a engenharia de requisitos nos sistemas embarcados. Essa revisão será alicerce para responder os objetivos da pesquisa e apresentar resultados relevantes na área de engenharia de requisitos nos SE.

3ª Etapa: Desenvolvimento de um levantamento de dados e uma análise sobre os RNFs e sua classificação nos sistemas embarcados. Essa análise dos dados extraídos da revisão da literatura investiga as linguagens/métodos, processos, padrões e questões em aberto sobre os RNFs, bem como sua especificação no processo de ER dos sistemas embarcados. Essa análise foi necessária para poder estender, detalhar e complementar os resultados da RSL.

4ª Etapa: Nessa etapa são apresentadas as conclusões, contribuições e limitações deste trabalho.

1.5 Estrutura da Dissertação

Conforme as etapas supracitadas, este trabalho foi estruturado em capítulos e organizado da seguinte forma:

Capítulo 01 apresenta a introdução deste trabalho, bem como as seções de (caracterização e justificativa do problema, objetivos da investigação e procedimentos metodológicos).

Capítulo 02 apresenta o referencial teórico para suporte deste trabalho.

Capítulo 03 apresenta uma revisão sistemática da literatura da engenharia de requisitos nos sistemas embarcados.

Capítulo 04 apresenta uma análise dos dados extraídos da RSL sobre os RNFs dos sistemas embarcados.

Capítulo 05 apresenta a conclusão deste trabalho, bem como as contribuições da pesquisa, limitações e trabalhos futuros.

2

REFERENCIAL TEÓRICO

Neste capítulo são apresentados os conceitos básicos e é destacada a importância da engenharia de requisitos no processo de desenvolvimento software conforme a visão de diversos autores.

2.1 Requisitos e seus tipos

Os requisitos de um sistema são suas atividades, funções e restrições fornecidas. Esses requisitos auxiliam na identificação das necessidades dos clientes e nas resoluções de problemas, por exemplo, automatizar uma tarefa, controlar um dispositivo, enviar uma solicitação ou explorar informações. Há pelo menos duas classificações desses requisitos. Requisitos funcionais que detalham as funcionalidades e atividades que o sistema disponibiliza, ou seja, são as funções que o usuário espera que o sistema ofereça de acordo com sua proposta de desenvolvimento. Já os requisitos não-funcionais são relacionados por restrições referente ao sistema como segurança, confiabilidade, tempo de resposta, disponibilidade, etc (SOMMERVILLE, 2011).

Ainda em conformidade com SOMMERVILLE (2011), a distinção dos requisitos muitas vezes não são tão claras, de modo que uma exigência do usuário preocupado com a segurança, por exemplo, uma restrição de acesso a usuários autorizados, aparentemente pode ser um requisito não-funcional. Todavia quando desenvolvido e analisado em mais detalhes, este requisito pode gerar outros requisitos que são claramente funcionais, nesse caso, é gerado um requisito funcional de autenticação para atender a exigência do usuário. A figura 2.1 mostra possíveis leitores dos requisitos de usuário e sistema. Percebe-se que os leitores dos requisitos de usuário não há uma preocupação com as instalações dos sistemas e como será sua implementação. Já os leitores dos requisitos do sistema há uma preocupação com o que o sistema fará, como ele irá apoiar os processos de negócios e como irá apoiar na implementação do sistema.

Figura 2.1 Leitores dos diferentes tipos de especificação de requisitos

Fonte: Adaptado de SOMMERVILLE (2011)

Na visão de PRESSMAN (2009), os requisitos são características dos serviços oferecidos pelos sistemas e suas restrições. Esses requisitos representam as necessidades dos clientes de um sistema e soluções para seus problemas, por exemplo, automatizar um serviço, controlar um dispositivo, enviar um pedido ou encontrar informações. O termo requisito, muitas vezes é simplesmente uma declaração abstrata de um serviço ou restrição de sistema, por outro lado, pode ser uma definição formal e detalhada de atividades e funções que o sistema fornece. Isto posto, alguns problemas surgem devido a falta de clareza nesses diferentes tipos de descrições, no entanto o autor faz a distinção usando o termo requisitos de usuários e requisitos de sistemas.

Na concepção de SOMMERVILLE; KOTONYA (1998), requisitos são definidos durante as etapas iniciais de desenvolvimento do sistema, ou seja, como uma especificação do que deve ser implementado. Eles são descrições de como o sistema deve se comportar, informações do domínio de aplicação, restrições sobre os sistemas e operações.

Os autores THAYER; BAILIN; DORFMAN (1997a), destacam que um requisito é uma característica descrita do sistema necessária para o cliente resolver um problema e consequentemente alcançar seu objetivo. Essa característica deve ser implementada para satisfazer um contrato, um padrão, especificação ou uma documentação formalmente imposta.

A maioria dos autores (THAYER; BAILIN; DORFMAN (1997b); PRESSMAN (2009); KRUCHTEN (2000); SOMMERVILLE; KOTONYA (1998); SOMMERVILLE (2011)) consideram a divisão do requisitos em dois tipos: requisitos funcionais (funções do sistema, o que o sistema deve fazer) e requisitos não-funcionais (comportamento do sistema em relação a alguns atributos, por exemplo, segurança, performance, confiabilidade, usabilidade, etc).

Conforme KRUCHTEN (2000), requisitos funcionais são conhecidos pelas funcionalidades de comportamento de um *software* por meio da especificação das condições de entrada e saída que deve possuir. Requisitos não-funcionais são atributos desejados de qualidade que não são descritos pelos requisitos funcionais.

Já na visão de THAYER; BAILIN; DORFMAN (1997b), “Os requisitos funcionais capturam a natureza da interação entre o componente e seu ambiente. Os requisitos não-funcionais restringem os tipos de solução que podem ser consideradas”.

De acordo com SOMMERVILLE (2011), os requisitos de sistema são, constantemente, definidos em requisitos funcionais, requisitos não-funcionais e requisitos de domínio:

- 1) *Requisitos funcionais*. São as definições dos serviços que os sistemas devem fazer, como deve proceder a entradas específicas e como deve se comportar em diversas situações. Em algumas delas, os requisitos funcionais também podem expressar o que o sistema não deverá fazer.
- 2) *Requisitos não-funcionais*. São as restrições desses serviços e funções que são fornecidas pelo sistema. Alguns deles como, restrições de *timing*, restrições no processo de desenvolvimento e nos padrões. Em geral os requisitos não-funcionais são aplicados no sistema como um todo, ou seja, geralmente eles não se aplicam a funções específicas ou características individuais do sistema.
- 3) *Requisitos de domínio*. São requisitos referentes ao domínio da aplicação do sistema, esses expressam as características e restrições no domínio. Os requisitos de domínio podem ser tanto funcionais ou não-funcionais.

2.1.1 Requisitos não-funcionais

Os requisitos não-funcionais, como o próprio nome apresenta, são aqueles que não estão especificamente ligados as funções do sistema. Eles estão ligados às características emergentes do sistema, por exemplo, confiabilidade, tempo de resposta e espaço de armazenamento. Eles também podem estar relacionados a restrições do sistema, como, dispositivos de E/S (entrada e saída) e na representação de informações junto a *interface* do sistema. Dito isto, esses requisitos podem especificar desempenho, disponibilidade, segurança, usabilidade e outras características emergentes. Isto pode significar que os RNFs são de caráter crítico e mais relevantes que os requisitos funcionais individuais, visto que uma falha detectada ou explorada por usuários nos requisitos não-funcionais podem comprometer todo o sistema, torná-lo

indisponível ou até mesmo inútil. Por exemplo, se um sistema de aeronave não atende aos requisitos de confiabilidade, seguramente ele não será homologado para operação; se um sistema de controle em tempo real não atender os requisitos de desempenho, os serviços de controle não funcionarão corretamente (SOMMERVILLE, 2011).

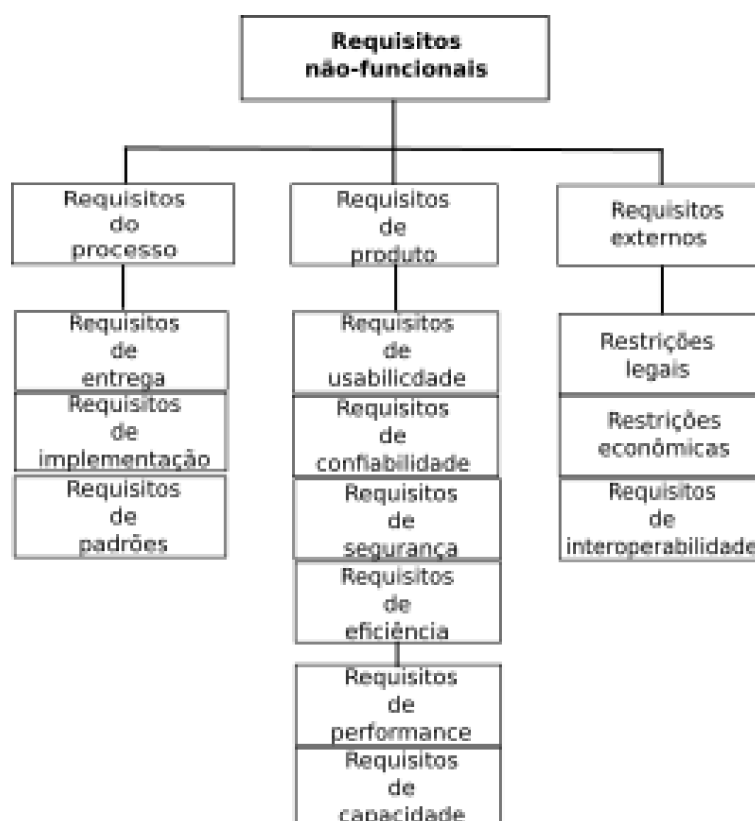
Na mesma perspectiva SOMMERVILLE; KOTONYA (1998), define que RNFs são requisitos que não são especificamente relacionados com a funcionalidade de um sistema. Eles apontam restrições sobre o processo de desenvolvimento de produto que está sendo desenvolvido e especificam as restrições externas que o produto deve atender. Requisitos não-funcionais incluem segurança, usabilidade, confiabilidade e requisitos de desempenho.

Nesse contexto, às necessidades dos usuários, às restrições de orçamento, às políticas organizacionais, à necessidade de interoperabilidade com outros sistemas de *software*, *hardware* ou a fatores externos, como, legislação a respeito de privacidade, normas e diretrizes, estão diretamente ligados aos RNFs que aparecem devido a essas necessidades.

SOMMERVILLE; KOTONYA (1998), classificam os requisitos não-funcionais em três categorias principais:

- 1) *Requisitos do processo*. Esses requisitos, são restrições imposta no processo de desenvolvimento do sistema, pois o cliente muitas vezes influenciar neste processo. Dessa forma os requisitos de processo incluem normas e métodos que devem ser seguidos no processo de desenvolvimento, ferramentas CASE que devem ser utilizadas e relatórios de gestão que dever ser apresentados.
- 2) *Requisitos do produto*. São os requisitos especificados que os sistemas ou subsistemas possuem. Por exemplo, a performance quando a rapidez do sistema, quanto a memória que ele requer, quanto a sua confiabilidade, ou seja, a sua tolerância a falhas, requisitos de portabilidade e usabilidade.
- 3) *Requisitos externos*. Esses requisitos podem surgir de ambos, do produto ou do processo e são derivados conforme o ambiente que o sistemas é desenvolvido. Estes requisitos são baseados nas informações de domínio do sistema, no ambiente organizacional, na interoperabilidade, ou seja, na capacidade de operar junto a outros sistemas, regulamentos de segurança e proteção de dados e até em questões naturais que envolvem a lei da física.

A Figura 2.2 apresenta a classificação dos requisitos não-funcionais. Observa-se no diagrama, que os RNFs podem surgir de características do *software*, ou seja, dos requisitos de produto. Podem surgir das organizações que desenvolvem o *software*, ou seja, dos requisitos de processo e por fim, dos requisitos externos.

Figura 2.2 Classificação do requisitos não-funcionais

Fonte: Adaptado de SOMMERVILLE (2011)

Além das categorias as quais os requisitos não-funcionais pertencem conforme apresentados na Figura 2.2, alguns desses RNFs (performance, confiabilidade, usabilidade, *security* e *safety*)¹ são “pilares” na especificação de requisitos dos sistemas críticos e embarcados, visto que esses sistemas são de computação complexas, submetem os usuários a grandes riscos e até a perdas econômicas. Neste sentido, SOMMERVILLE; KOTONYA (1998), definem que estes sistemas são críticos porque se houver falhas podem causar significativos danos econômicos e físicos as organizações e as pessoas. Os requisitos não-funcionais mais relevantes nesses sistemas são:

- **Confiabilidade.** Requisitos de confiabilidade são restrições ligadas ao tempo de execução dos sistemas e são classificados sob dois títulos separados, a disponibilidade do serviço dos sistemas quando solicitado pelo usuário (disponibilidade) e a quantidade de vezes que o sistema deixa de fornecer o serviço ao usuário final (taxa falhas).
- **Performance.** Requisitos de desempenho atua sobre a velocidade de operação do sistema e seus principais tipos de exigência são tempo de resposta que o sistema

¹Os termos *security* e *safety* serão mantidos no idioma inglês neste trabalho, pois não há uma tradução adequada no idioma português.

retorna a uma solicitação do usuário, a capacidade processamento de dados em um determinado período de tempo e a rapidez com que o sistema coleta dados de entrada por meio de sensores antes que seja sobrescrito pela próxima entrada.

- *Security*. Os requisitos de proteção são incluídos no sistema para garantir o bloqueio do acesso não autorizado a dados e a integridade do sistema contra danos acidentais ou maliciosos.
- *Usabilidade*. Os requisitos de usabilidade estão preocupados com a especificação das interações das interfaces de usuários e interface do usuário final com o sistema. Geralmente é aceito um manual de usuário bem estruturado e detalhado, avisos e mensagens de erros bem informativos, facilidade de ajuda e uma interface de usuário consistente e amigável para melhorar a usabilidade. O autor ainda define, que todo esse processo pode ser limitado por requisitos de usabilidade de nível superior, como, mensurar em termos de anos de experiência a classe de aplicações (requisitos de entrada), o tempo necessário para aprender a utilizar o sistema (requisitos de aprendizagem), a taxa de erros dos usuários finais que manuseiam o sistema em uma velocidade normal (requisitos de manuseio) e o requisito que mensura o grau de satisfação do usuário no uso do sistema (requisito de satisfação).
- *Safety*. São considerados todos os requisitos em sistemas relacionados com a segurança ou até os subconjuntos desse requisitos que estão diretamente relacionados com a garantia do funcionamento seguro do sistema, ou seja, especificados para proteger o sistema contra acidentes ou funcionamento incorreto. Por exemplo, se um sistema não é confiável, ele pode falhar de forma imprevisível, portanto ele não pode estar disponível para funções que exija segurança.

Existem outras descrições para os RNFs como atributos de qualidade e restrições. A denominação requisito não-funcional já é consolidada no meio acadêmico e industrial e por isso será adotada neste trabalho.

Existem também diversas classificações para os tipos de requisitos não-funcionais na literatura, por exemplo COMMITTEE; BOARD (1998a), ISO/IEC 9126 (ISO/IEC, 2001) e CHUNG; CESAR; LEITE (1999).

A norma COMMITTEE; BOARD (1998a) expõe uma lista de exemplos de tipos de RNFs: verificação, aceitação, *interface*, desempenho, recursos, documentação, *security*, *safety*, portabilidade, qualidade, confiabilidade e manutenabilidade.

Já o padrão internacional ISO/IEC 9126 (ISO/IEC, 2001), apresenta 06 características para qualidade de *software*: usabilidade, confiabilidade, funcionalidade, eficiência, manutenabilidade e portabilidade. Esse padrão destaca-se por subdividir essas características para definir melhor os RNFs dos sistemas de *software*.

Já CHUNG; CESAR; LEITE (1999), propõe uma grande lista com mais de uma centena de RNFs, dentre eles, é destacado o detalhamento aos requisitos de acurácia, *security* e desempenho.

Os sistemas de *software* podem ser complexos, de acordo com seus requisitos funcionais e não-funcionais, tais como, confiabilidade, manutenabilidade, desempenho, custos operacionais, entre outros. Os RNFs por exemplo, desempenham uma função crítica no desenvolvimento de *software*, pois se definidos e especificados de forma incorreta, torna-se de difícil correção e podem impactos nos custos (CHUNG et al., 2012).

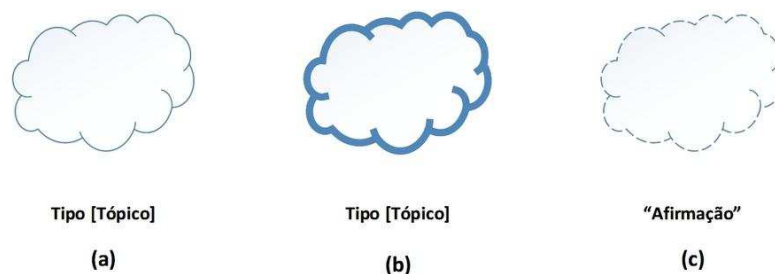
Diante dessas perspectivas, várias abordagens foram descritas na literatura para incorporar os RNFs no desenvolvimento de *software*, no entanto selecionamos o NFR *Framework* proposto por CHUNG; CESAR; LEITE (1999) por ser o mais referenciado na literatura atual.

2.1.1.1 NFR Framework (conceitos principais)

De acordo com CHUNG et al. (2000), o NFR *Framework* é um método sistemático para representar, classificar e analisar os requisitos não-funcionais. Ainda em conformidade com o autor, os RNFs também devem ser atendidos por um sistema de *software* por se tratar de um requisito, no entanto como esses requisitos são subjetivos, o conceito de meta ou *goal* para representá-lo não é indicado. Dessa maneira, o autor definiu o conceito de *softgoal* para representar os RNFs, que nesse caso atende superficialmente, diferente do *goal* que atende absolutamente. O autor relacionou o termo *softgoal* ao termo *satisficing*, que no inglês significa à satisfação de algo relacionado a um determinado nível.

Os *softgoals* são representados por desenhos de nuvens que podem ser conectadas entre si por meio de relações de interdependência. O atendimento de um *softgoal* deve ser feito em conjunto com os demais *softgoals* de um sistema de *software*, pois eles podem atender de forma positiva ou negativa, suficiente ou parcial um outro *softgoal* (CHUNG et al., 2000).

A Figura 2.3, representa os três tipos básicos de *softgoals*: (a) *softgoals* de RNFs, que são os propriamente ditos e representados pela figura de uma nuvem comum; (b) *softgoals* de operacionalização, que atuam como identificadores de estratégias conforme a decomposição de um RNF e são representados pela figura de uma nuvem com bordas escuras; e (c) *softgoals* de afirmação, que servem para registrar decisões e motivos na escolha de uma alternativa de projeto. Esses *softgoals* são representados pela figura de uma nuvem com bordas tracejadas.

Figura 2.3 Tipos de *softgoals*.

Fonte: Adaptado de CHUNG et al. (2000).

O principal meio de representação do método NFR *Framework* é feito por meio de grafos de interdependência de *softgoals*, também conhecido como, SIG (*Softgoal Interdependency Graph*). Esses grafos, descrevem os *softgoals* e seus respectivos relacionamentos, tal como, as observações dos desenvolvedores e *stakeholders*² sobre o nível de satisfação dos RNFs. Esses grafos são elaborados com auxílio de cinco parâmetros: (1) *softgoals*; (2) interdependência; (3) procedimento de avaliação; (4) correlações; e (5) métodos.

Um outro componente importante do NFR *Framework*, são os catálogos, ou seja, um conjunto de conhecimento organizados de requisitos não-funcionais que podem ser aplicados em diferentes domínios de sistemas de *software* e são especificados em três tipos: (1) catálogo de tipos de RNFs (confiabilidade, desempenho, usabilidade, entre outros); (2) catálogo de métodos de refinamento e técnicas de desenvolvimento que auxilia no atendimento dos RNFs; e (3) catálogo de correlações entre *softgoals* (CHUNG; CESAR; LEITE, 1999).

2.2 Engenharia de Requisitos

De acordo com PRESSMAN (2009), a engenharia de requisitos tem como objetivo, fornecer meios no desenvolvimento de *software* para entender o que o cliente necessita, ou seja, analisando suas necessidades, negociando soluções plausíveis para seus problemas, avaliando estas soluções, especificando as ideias, validando a especificação e por fim, o gerenciamento de todo esse processo que está resumido em sete fases distintas: concepção, levantamento, elaboração, negociação, especificação, validação e gestão.

Em consonância com ZAVE (1997), a ER é definida como uma ciência que estuda e se preocupa com a análise e documentação de requisitos, análise das necessidade e especificação de requisitos. Em resumo, a engenharia de requisitos fornece meios apropriados para facilitar as atividades de análise, documentação e verificação.

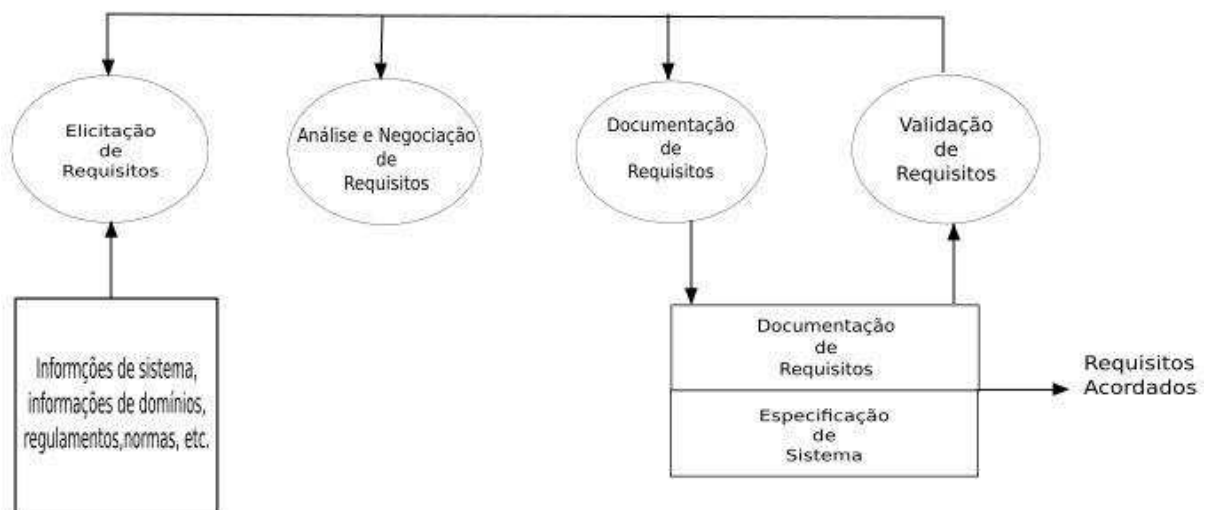
No entendimento de ZAVE (1997), A ER é uma área da engenharia de *software* que se prioriza as metas reais, serviços, funções e restrições do *software*. Além disso, preocupa-se

²O termo *stakeholders* será mantido no idioma inglês, pois não há uma tradução adequada no idioma português.

com o relacionamento entre estes fatores e com sua evoluções e comportamentos através do tempo.

Na perspectiva de SOMMERVILLE; KOTONYA (1998), o processo de ER é constituído por atividades que colaboram para a criação e manutenção do documento de requisitos, que pode ser dividido nas seguintes fases: elicitação, análise e negociação, documentação de requisitos e validação. Inicialmente os requisitos de sistema são descobertos por meio de consulta com os *stakeholders*, a partir de análises de documentos do sistema, conhecimento do domínio e estudos de mercado (elicitação), logo após, os requisitos são analisados em detalhes e suas partes mais interessantes são negociadas para decidir quais requisitos serão aceitos. Esta fase é importante, pois há conflitos de requisitos de diferentes fontes, informações incompletas e requisitos incompatíveis com o desenvolvimento do sistema (análise e negociação), na sequência, os requisitos negociados e acordados são documentados em um nível de detalhe apropriado aos *stakeholders*, por último, deve haver uma verificação cuidadosa dos requisitos em relação a sua consistência e completude, visto que esta fase tem objetivo de detectar problemas nos documentos de requisitos e evitar que sejam usados como base no desenvolvimento do sistema (validação de requisitos). A Figura 2.4 mostra por meio de ícones as atividades de ER supracitadas, que na prática são intercaladas e há uma grande interatividade e *feedback* de uma atividade para a outra.

Figura 2.4 Modelo de atividades do processo de ER

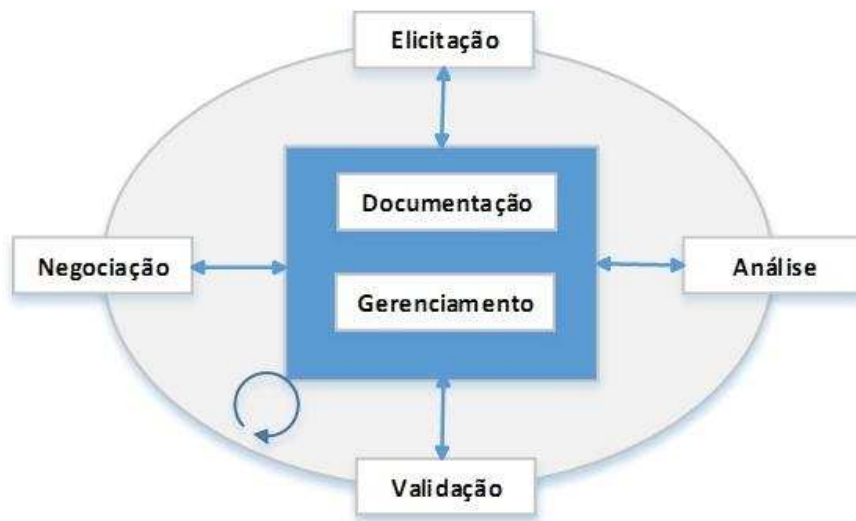


Fonte: Adaptado de SOMMERVILLE; KOTONYA (1998).

Na visão de SOMMERVILLE (2011), o processo para encontrar, analisar e documentar atividades é chamado de engenharia de requisitos. Além disto, este processo é dividido em cinco etapas:

- 1) **Estudo de viabilidade.** É realizado um estudo para avaliar se as necessidades dos clientes e usuários são atendidas com uso das tecnologias de *hardware* e *software*. Faz-se no sistema avaliações técnicas, econômicas e financeiras do ponto de vista mercadológico. Esta análise tem como propósito fornecer um resultado objetivo e rápido sobre a tomada de decisão quanto a prosseguir ou não para uma fase mais detalhada.
- 2) **Elicitação e análise de requisitos.** É a etapa em que os requisitos são observados e identificados. No processo de elicitación, esses requisitos são discutidos com os usuários e clientes, as tarefas e funções dos sistemas são detalhadas e ajudam aos desenvolvedores a produzirem modelos de sistemas e protótipos.
- 3) **Especificação de requisitos.** É o processo de traduzir e especificar informações reunidas em um documento de requisitos. Nesse documento há pelo menos dois tipos de requisitos especificados. Requisitos de usuário que descrevem os requisitos funcionais e não funcionais de modo que usuários que não possuem conhecimento técnico entendam. Requisitos de sistema é uma descrição mais complexa e detalhada das atividades do sistema.
- 4) **Validação de requisitos.** Essa etapa como o próprio nome descreve, serve para validar os requisitos quanto a consistência, realismo e abrangência. O processo de validação permite a identificação de erros e verificar se a especificação atende as perspectivas dos usuários e clientes.
- 5) **Gerenciamento de requisitos.** Essa fase tem o objetivo de compreender e controlar as mudanças que os requisitos podem sofrer. É necessário acompanhar os requisitos individuais e manter as ligações entre os requisitos dependentes, de modo que torne possível avaliar os impactos das mudanças desses requisitos.

Já na Figura 2.5, SOMMERVILLE (2005), descreve as fases da engenharia de requisitos em um ciclo que fornecem o mecanismo apropriado para entender o que o cliente quer, analisando a necessidade, avaliando a viabilidade, negociando uma solução razoável, especificando a solução sem ambiguidade, validando a especificação e gerenciando os requisitos à medida que são transformados em um sistema operacional.

Figura 2.5 Ciclo da Engenharia de Requisitos

Fonte: Adaptado de SOMMERVILLE (2005)

Observa-se que os autores possuem abordagens muito semelhantes no processo de engenharia de requisitos, uma vez que as diferenças apresentadas são apenas na visão das fases deste processo.

2.3 Sistemas Embarcados

Esta seção será abordado conceitos sobre sistemas embarcados, pois este trabalho foi aplicado na área de Engenharia de Requisitos dos Sistemas Embarcados.

Os sistemas embarcados são amplamente difundidos no nosso cotidiano. Eles estão presentes em domínios da indústria aeroespacial, telecomunicações, tecnologia de automação, aviação e principalmente nos domínios de tecnologias médicas e automotivas. Conforme ZAVE (1982), o termo “Embarcado” foi difundido pelo Departamento de Defesa dos EUA em conjunto com seu projeto de desenvolvimento de línguas comuns (Ada). “Embarcados” refere-se ao fato de que esses sistemas estejam embutidos em sistemas maiores cujos propósitos primários não são computação, mas qualquer sistema de computador útil. Um programa de folha de pagamento, por exemplo, é uma parte essencial de uma organização empresarial.

Nesse contexto, LEE; SESHIA (2011) cita que, o uso mais visível de computadores e software é o processamento de informações para consumo humano. A grande maioria dos computadores em uso, no entanto, são muito menos visíveis. Por exemplo, o motor, os freios, os cintos, o *airbag* e o sistema áudio em seu carro. Eles codificam digitalmente a sua voz e convertem em um sinal de rádio para enviá-lo do seu telefone celular para uma estação base.

Eles comandam robôs em uma fábrica, geração de energia em uma usina, processos em uma fábrica de produtos químicos e semáforos em uma cidade. Esses computadores menos visíveis são chamados de sistemas embarcados, e o *software* que eles executam é chamado de *software* embarcado.

Já na visão de FRIEDRICH (2009), os sistemas embarcados podem ser definidos como uma combinação de qualquer dispositivo que inclua um computador programável, peças mecânicas ou eletrônicas, projetadas para desempenhar uma função dedicada, mas que não se destine a ser um computador de uso geral. O termo embarcado implica que esses sistemas são de fato, uma parte fundamental de um sistema maior. Esses sistemas também recebem definições como, sistemas de tempo real e podem ser encontrados em grande escala no uso cotidiano das pessoas, desde os mais simples até os de alta complexidade e criticidade, como exemplos, os sistemas de roteadores de rede e os de aplicação aeroespacial. O autor ainda define que, todos os sistemas embarcados têm uma funcionalidade dedicada, pois também são sistemas dedicados. A funcionalidade dedicada significa que os sistemas foi projetado para uma função específica e que possui tarefas predefinidas no seu *hardware* e *software*.

Referente a complexidade dos SE, ZAVE (1982) descreve que, os sistemas embarcados possuem características que são notáveis em relação as demais aplicações de computador. Na maioria das vezes são grandes (50 000 a 100 000 linhas de códigos), são de longa duração (10 a 15 anos), as alterações são contínuas devido a evolução dos requisitos do sistema e suas revisões podem ser da mesma magnitude do desenvolvimento original. Esses sistemas estão na classe das mais importantes, pois possui uma computação complexa e uma expansão rápida em volume e variedade.

Nesse contexto, FRIEDRICH (2009), relata que a variedade de aplicações e plataformas podem ter propriedades baseadas nos sistemas embarcados muito diferentes. As necessidades de *hardware* podem ser alcançadas com processadores comuns de propósito geral. Como exemplo, os dispositivos *high end*³, incluindo telefones celulares, eletrônicos como, TV, DVD, entre outros. No entanto, outras plataformas e sistemas embarcados podem requerer um *hardware* específico e microprocessadores capazes de lidar com restrições de tempo. Em casos excepcionais que o tempo de resposta são extremamente pequenos, os microprocessadores podem ser substituídos por dispositivos de lógica eletrônica com cabo rígido, assim as funcionalidades podem ser modificadas e atualizadas por meio de programação de *software*. É possível acrescentar funcionalidades novas aos dispositivos por meio de uma atualização de *software*, desde que o *hardware* não precise de uma modificação ou o espaço de memória disponível seja grande o bastante para escrever as alterações. Por exemplo, podemos citar novamente os dispositivos *high end*, pois apresentam milhões de linhas de código com diversos graus de garantia de correção. Dessa forma, eles podem incorporar ou acoplar componentes de terceiros e até mesmo, sistemas operacionais completos (como o Linux) que podem ser

³Termo em inglês para definir o tipo de tecnologia de um dispositivo. Significa a reprodução de áudio feita por um aparelho de som com a maior fidelidade possível ao som real.

instalados pelo fabricante, pelos fornecedores e até mesmo pelo usuário final. Diante do exposto, um problema evidente preocupa os fabricantes desses sistemas embarcados, pois essas alterações que podem ser feitas muitas vezes sem preparação adequada do *hardware* e em tempo de execução, fato que pode ocasionar falhas e comprometimento do dispositivo.

Ainda em conformidade com FRIEDRICH (2009), existe uma grande diversidade de sistemas embarcados, desde pequenos sensores inteligentes a sistemas de controle de aeronaves e simuladores de veículos, no entanto as funções que eles desempenham podem variar muito. Algumas das categorias em que essas funções podem ser colocadas são: Computação (dando a inteligência ao sistema geral); Medição através de sensores; Controle do meio ambiente através de atuadores após decisões tomadas durante a computação; Comunicação (incluindo dados, música, vídeo, etc.); *Interface* humana (por meio de um visor e alguns botões ou um teclado limitado). Portanto, diferentes sistemas embarcados necessitarão de funções diferentes, mas nem todas as funções são necessárias em um sistema embarcado. Por exemplo, um robô nem sempre terá uma *interface* humana.

É evidente e bastante citado pelos autores FRIEDRICH (2009), ZAVE (1982) e LEE; SESHIA (2011) sobre o grande crescimento dos SE e da sua atuação no cotidiano da sociedade, portanto é fundamental para ciência, compreender sobre as especificidades dos diversos serviços prestados pelos sistemas embarcados nos seus respectivos domínios. Desse modo FRIEDRICH (2009), descreve sobre alguns desses domínios:

Automotivo. Inclui desde sistema de controle eletrônico de chassi, sistemas de informação, sistema de computador de bordo e sistemas de segurança.

Aviônica / Aeroespacial. o sistema monitora a decolagem e pouso de aviões, mantém voo, mantém sua rota de voo, e auxilia rota fora do caminho do dano. Ele inclui aeronaves comerciais, aeronaves militares e sistemas de satélites.

Automação Industrial. Inclui controles de fabricação e processo, controladores de movimento, casas inteligentes, interfaces de operação, robótica, distribuição de energia, entre outros.

Telecomunicações. inclui sistemas de informações de tempo real, tais como cotações de ações. Ele inclui infraestrutura e serviços finais.

Equipamentos Médicos e de Saúde. Inclui equipamentos de monitoramento de pacientes, equipamentos de terapia médica, equipamentos de diagnóstico, equipamentos de imagem e sistemas cirúrgicos.

2.3.1 Requisitos de sistemas embarcados

De acordo com ESTRIN et al. (2001), na engenharia de *software*, os requisitos funcionais especificam o comportamento ou funções específicas de um sistema de *software* ou de seu componente. Em geral, os requisitos funcionais definem o que um sistema deve fazer. Eles são suportados por requisitos não-funcionais, que especificam critérios que podem ser usados

para julgar o funcionamento de um sistema, em vez de comportamentos específicos. Em geral, os RNFs definem como um sistema deve ser. Requisitos não-funcionais são muitas vezes chamados qualidades de um sistema. Um sistema de monitoramento televisionado cardíaco pode ser necessário para apresentar o centro médico com uma exibição de dados, como o ritmo cardíaco. Este é um requisito funcional. O quão recente esses dados precisam ser, é um requisito não funcional. Se os dados precisam ser atualizados em tempo real, os arquitetos do sistema devem assegurar que o sistema é capaz de atualizar os dados exibidos dentro de um intervalo aceitavelmente curto da mudança de dados. Tradicional, software embarcado pode ser bastante complexo e tem uma série de requisitos. Estes têm implicações tanto para a aplicação e para a infraestrutura de *software*, como o sistema operacional. Alguns podem ser descritos como:

Computação com recursos limitados. Eles são frequentemente e rigorosamente limitados em relação aos recursos disponíveis. Especialmente por causa das restrições de custo e tamanho que são uma dupla essencial para a produção em massa e forte concorrência industrial, recursos como CPU, memória, dispositivos foram projetados para atender a esses requisitos. Além disso, especialmente para sistemas móveis ou embarcados autônomos de energia é um recurso inestimável. Como resultado dessas restrições, o sistema precisa usar eficientemente seus recursos computacionais. Por exemplo, o sistema operacional deve ser capaz de operar em ambientes restritos.

Requisitos em tempo real. Como muitas aplicações embarcadas interagem profundamente com o mundo real, elas geralmente têm requisitos rigorosos em tempo real. Esses aplicativos requerem funcionalidades como controle de processo, processamento de multimídia, instrumentação, entre outros, ou seja, onde o sistema deve cumprir uma exigência temporal ou prazo. Os prazos são qualificados como suaves, firme ou difícil. Dependendo do tipo de prazo, os métodos para garantir que um determinado prazo seja cumprido são diferentes. Ocasionalmente, os pontos de preempção precisam ser inseridos em caminhos de execução críticos para reduzir a latência de agendamento.

Portabilidade. Muitos tipos diferentes de CPUs, chips periféricos e arquiteturas de memória podem ser usados em sistemas embarcados. Assim, para baixo custo, qualquer sistema operacional embarcado ou outro componente reutilizável que se destina a ser usado em vários aplicativos deve ser comumente portátil para plataformas de *hardware* personalizado.

Alta confiabilidade. Os sistemas embarcados são implantados remotamente, muitas vezes em aplicativos críticos de infraestrutura. As falhas de *software* são, portanto, muito problemáticas e são extremamente caras ou mesmo impossíveis de corrigir.

Ainda em conformidade com ESTRIN et al. (2001), os sistemas embarcados possuem atributos de qualidade que são essenciais não às estruturas de *software* para que suportem esses sistemas, como exemplos, confiabilidade, robustez, estabilidade, manipulação de falhas, segurança, privacidade, escalabilidade e capacidade de atualização.

2.4 Considerações Finais

Neste capítulo foram apresentados breves conceitos e características sobre os requisitos e seus tipos, requisitos não-funcionais, engenharia de requisitos e sistemas embarcados, pois esses conceitos foram utilizados como suporte e para um entendimento em relação às pesquisas desenvolvidas, como exemplo a Revisão Sistemática da Literatura da Engenharia de Requisitos nos Sistemas Embarcados e da Análise dos RNFs nos Sistemas Embarcados. Os conceitos da engenharia de requisitos foram utilizados na elaboração dos protocolos de pesquisa, da busca automática e para a análise dos resultados extraídos das revisões. As cinco fases abordadas neste capítulo foram identificadas na RSL dos embarcados, os tipos de requisitos, as especificidades dos sistemas embarcados e seus domínios de aplicação.

3

UMA REVISÃO SISTEMÁTICA DA ENGENHARIA DE REQUISITOS PARA SISTEMAS EMBARCADOS

Este capítulo reúne e apresenta informações sobre estudos selecionados nas principais bases científicas Internacionais em relação a engenharia de requisitos nos sistemas embarcados, inclui uma avaliação, síntese e apresentação desses estudos de modo que, essa investigação possa responder as questões de pesquisas, bem como tornar possível explorar os problemas da área e as questões em aberto.

3.1 Introdução

Devido à complexidade dos SE, é necessário uma busca de técnicas, métodos e ferramentas, a fim de apoiar a qualidade desses *softwares*, de se adaptar e criarem novas técnicas que leve em consideração as particularidades dos sistemas embarcados. No entanto, existe uma escassez ou uma completa ausência de métodos, técnicas, e ferramentas de engenharia de requisitos, para o desenvolvimento em domínios de SE (OSSADA, 2010).

A fim de verificar isso, propomos uma Revisão Sistemática da Literatura de acordo com KEELE (2007), para avaliar e sintetizar as evidências disponíveis na literatura, para responder às perguntas de investigação sobre a utilização de abordagens, métodos, técnicas e problemas comuns da ER no domínio de sistemas embarcados.

No desenvolvimento da revisão sistemática da literatura em ER nos sistemas embarcados, participaram os estudantes de Pós-Graduação (mestrado e doutorado) Tarcísio Pereira, Deivson Albuquerque e Aêda Monaliza, como orientador e coorientadora respectivamente o professor Jaelson Castro e Fernanda Alencar.

3.2 Planejamento

A RSL foi elaborada de acordo com as etapas sistemáticas propostas por PETERSEN et al. (2008). Dessa forma o processo envolveu atividades que foram agrupadas em três fases principais: planejamento da avaliação, execução da avaliação e elaboração dos relatórios.

Inicialmente, é necessário compreender, como as abordagens de engenharia de requisitos contribuem para a solução de problemas de requisitos no domínio de sistemas embarcados, como as fases do processo de engenharia de requisitos são suportadas pelas abordagens, quais são os domínios que as abordagens apoiam e quais as questões em aberto dessas abordagens. Desse modo, o processo de revisão sistemática da literatura teve início em junho de 2015 e foi concluída em setembro de 2016. Este estudo da literatura de engenharia de requisitos no domínio dos sistemas embarcados teve como abrangência o período entre 1970 a junho de 2016, ou seja, a pesquisa leva em consideração as últimas 4 décadas de estudos nessa área.

3.2.1 Elaboração do protocolo

Um protocolo foi elaborado para sistematizar o processo da revisão a fim de obter o estado da arte em engenharia de requisitos no domínio dos ES, fornecer orientações, dados relevantes da área e identificar as questões em aberto para trabalhos futuros.

O protocolo inicia com um breve *background* sobre tema a ser pesquisado, logo em seguida são apresentadas as questões com suas respectivas motivações, as palavras-chave e *string* de pesquisa, as bases de dados eletrônicas pesquisadas, os critérios de seleção dos estudos, processo de seleção dos estudos primários, avaliação de qualidade, síntese e estratégia de extração de dados e disseminação dos resultados.

A versão final desse protocolo está descrita no Apêndice B.

3.2.2 Questões de pesquisa

As questões de pesquisa têm o propósito de compreender como os processos de ER são utilizados para o desenvolvimento de sistemas embarcados e como desenvolver uma agenda de pesquisa para orientar trabalhos futuros. Dessa forma, pretende-se esclarecer a principal questão de pesquisa: Como a comunidade de desenvolvimento de sistemas embarcados está usando o processo de ER? Para responder essa questão, foram derivadas outras questões relacionadas a ER com aspectos específicos o qual estamos interessados conforme descritos na Tabela 3.1.

Os critérios e a tabela de análise de qualidade podem ser verificadas em: https://drive.google.com/file/d/0BzWTP8Q_rZpBOHhESFRrTUNkMjQ/view

Tabela 3.1 Questões de pesquisa e motivações

Questão de pesquisa	Descrição e Motivação
Q1. Como as abordagens contribuem para a solução de problemas de engenharia de requisitos no domínio de sistemas embarcados?	O objetivo desta questão é identificar e descrever abordagens de engenharia de requisitos para sistemas embarcados. É importante porque fornece um conjunto de contribuições relativas à utilização dessas abordagens para resolver alguns problemas de pesquisa ER bem conhecidas e que podem ser úteis para pesquisas interessadas em usar algumas destas abordagens no ER para o domínio de sistemas embarcados.
Q2. Quais as fases do processo de ER apoiadas pelas abordagens?	Esta questão fornece um ponto de partida para entender o que são as principais fases (elicitação, análise, especificação, validação e gerenciamento) do processo de engenharia de requisitos suportados pelas abordagens e obter um parâmetro sobre quais as fases são mais relevantes no processo de ER nos SE.
Q3. Que tipo de requisito (funcional e/ou não-funcional) foi apoiado pelas abordagens?	A questão é importante para poder identificar a distribuição dos estudos, de modo que possa investigar se as abordagens de ER nos ES estão gerenciando ou levando em consideração os requisitos funcionais ou os não-funcionais.
Q4. Quais as questões em aberto das abordagens encontradas?	A resposta a esta pergunta indica as questões em aberto existentes relacionadas com a engenharia de requisitos no domínio de SE. É importante porque identifica um conjunto de linhas de pesquisa que podem ser levadas em consideração para trabalhos futuros.

Questão de pesquisa	Descrição e Motivação
Q5. Quais são os domínios que as abordagens suportam?	O objetivo desta pergunta é conhecer os diferentes domínios (tecnologia de automação, aviação, tecnologia de energia automável ou tecnologia médica) que as abordagens de ER nos embarcados suportam e identificar o domínio que necessita de uma melhor atenção em relação a ER.
Q6. Quais requisitos devem ser considerados durante o desenvolvimento de sistemas embarcados?	O objetivo desta pergunta é identificar os requisitos que devem ser levados em conta no desenvolvimento de sistemas embarcados, ou seja, os requisitos relevantes que devem ser detalhados e especificados no processo de ER dos SE.

3.3 Condução da Revisão

Na realização da RSL foi feita uma busca automática para identificação dos estudos interessados para este trabalho.

3.3.1 Técnica de pesquisa

Para realização da busca automática foram elaboradas três *strings* em conformidade com as questões de pesquisa, a fim de extrair os resultados esperados das bases eletrônicas que serão expostas adiante. A Tabela 3.2, apresenta os termos de buscas que são formados pelas palavras-chave da pesquisa, entre as palavras-chave percebe-se o conector OR, utilizado para combinar essas palavras, o conector AND é utilizado para conectar um termo ao outro. Dessa forma temos uma *string* formada pelos termos de busca. As três *strings* foram necessárias devido a diferença da sintaxe de busca eletrônica entre as bases digitais de pesquisa.

Tabela 3.2 Técnica de busca automática

#	Termos de busca:
(1)	“requirements engineering” OR “requirements elicitation” OR “requirements specification” OR “requirements management” OR “requirements validation” OR “requirements education”
(2)	“requirements modeling OR requirements modelling”
(3)	“embedded systems” OR “safety critical systems” OR “real time systems” OR “embedded software” OR “embedded product”
(4)	“approach” OR “technique” OR “framework” OR “processes” OR “methods” OR “tool”

1 String:

(“requirements engineering” OR “requirements elicitation” OR “requirements specification” OR “requirements management” OR “requirements validation” OR “requirements education” OR (“requirements modeling” OR “requirements modelling”)) AND (“embedded systems” OR “safety critical systems” OR “real time systems” OR “embedded software” OR “embedded product”) AND (“approach” OR “technique” OR “framework” OR “processes” OR “methods” OR “tool”)

- Engineering Village (377);
 - Science Direct (1114);
 - Scopus (1675);
 - SpringerLink (292);
 - ISI Web of Science (105);
-

2 String:

(“requirements engineering” OR (“requirements modeling” OR “requirements modelling”)) AND (“embedded systems” OR “embedded software” OR “real-time system” OR “real-time”) AND (“approach” OR “framework” OR “processes” OR “tool”)

ACM Digital Library (668) *string* adaptada;

3 String:

(“requirements engineering” OR “requirements modeling”) AND (“embedded systems” OR “embedded software” OR “real-time system” OR “real-time”) AND (“approach” OR “framework” OR “processes” OR “tool”)

IEEEXplore (232) *String* adaptada;

3.3.2 Origem dos dados

A busca da pesquisa foi direcionada às principais bases de dados internacionais conforme descrita na Tabela 3.3. São apresentadas as bases de dados e seu endereço eletrônico correspondente.

Tabela 3.3 Base de dados pesquisados

Base de dados	Endereço Eletrônico
Engineering Village	www.engineeringvillage.com/
Science Direct	www.sciencedirect.com/
Scopus	https://www.scopus.com/
SpringerLink	link.springer.com/?
ISI Web of Science	webofscience.com/
ACM Digital Library	dl.acm.org/
IEEEExplore	ieeexplore.ieee.org/

3.3.3 Seleção de estudos

A RSL seguiu os seguintes passos para responder as questões de pesquisa: (1) identificação da necessidade de uma revisão; (2) desenvolvimento de um protocolo de avaliação; (3) uma busca abrangente de estudos primários; (4) avaliação de qualidade dos estudos incluídos; (5) extração de dados e monitoramento; (6) síntese e análise de dados; e (7) elaboração de relatórios. A ferramenta de software chamada StArt (*State of the Art through Systematic Review*) LAPES (2015) foi utilizada, dando suporte no desenvolvimento da revisão.

A identificação dos estudos selecionados, bem como seus respectivos títulos e autores estão descritos no Apêndice C.

3.3.3.1 Critérios de inclusão e exclusão

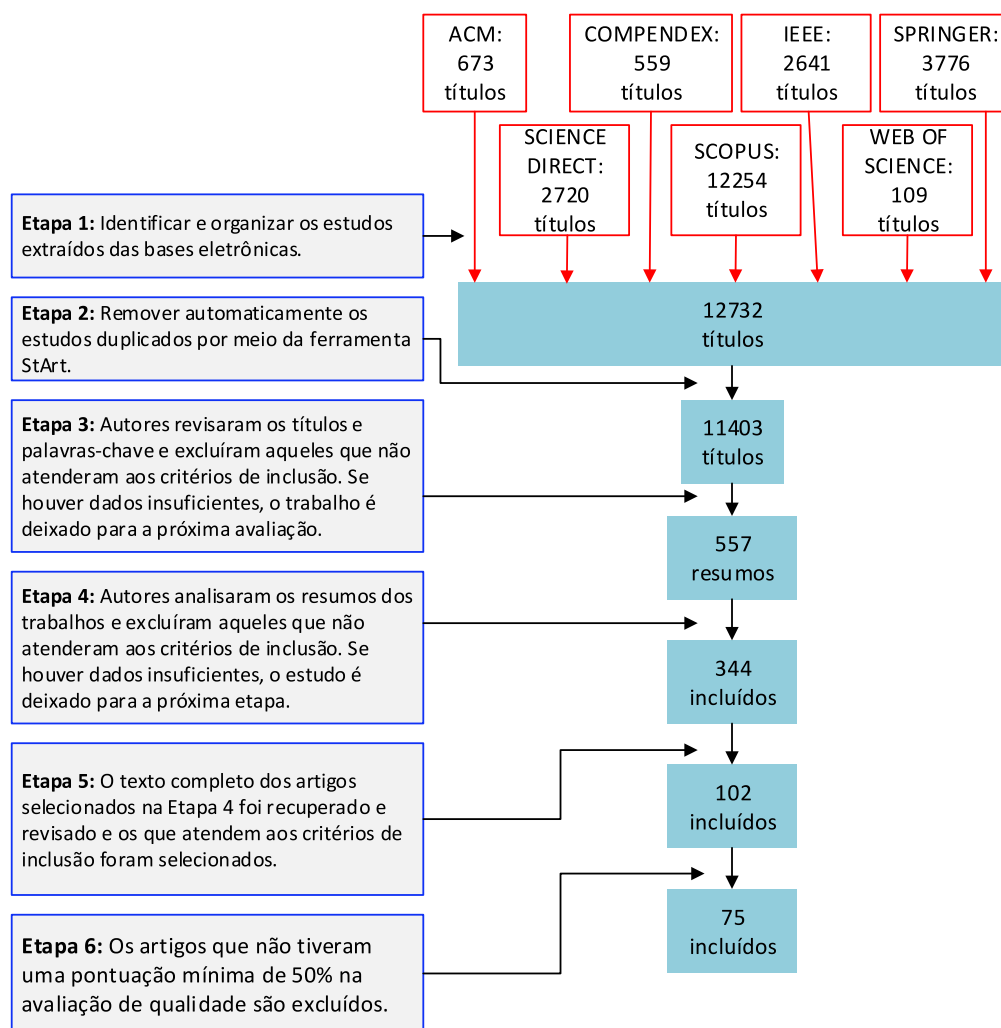
Segundo KEELE (2007), os critérios de inclusão e exclusão dos estudos são essenciais para identificação dos estudos primários, pois esses, fornecem evidências e informações diretas sobre o tema da pesquisa e diminuem a probabilidade de viés.

A tabela 3.4 apresenta os critérios de inclusão e exclusão de estudos utilizados para seleção dos estudos da RSL. Foram incluídos estudos que atenderam aos critérios: estudos primários, estudo revisados por pares, estudos que discutem requisitos engenharia no desenvolvimento de sistemas embarcados e estudos publicados entre 1970 até junho de 2016. Os resultados obtidos são de grande importância, uma vez que leva em conta mais de 4 décadas de pesquisa sobre a ER nos sistemas embarcados. Trabalhos foram excluídos da revisão, se seus estudos eram secundários, estudos curtos, se eles não estão relacionadas com as perguntas de pesquisa, se estudos não são revisados por pares, estudos duplicados, trabalhos que não estão no idioma inglês, estudos que não discutem ER no desenvolvimento SE, literatura cinzenta, artigos redundantes de alguma autoria, estudos que discutem abordagem genérica de engenharia de requisitos para sistemas embarcados e estudos em andamento.

Tabela 3.4 Seleção de estudos (Inclusão e Exclusão)

Critério de Inclusão	Critério de Exclusão
Estudos primários	Estudos eram secundários
Estudo revisados por pares	Estudos curtos (≤ 5 páginas)
Estudos que discutem requisitos engenharia no desenvolvimento de sistemas embarcados	Não relacionados com as perguntas de pesquisa
Estudos publicados entre 1970 até junho de 2016	Estudos que não são revisados por pares
-	Estudos duplicados
-	Estudos que não estão em idioma inglês
-	Estudos que não discutem RE no desenvolvimento ES
-	Literatura cinzenta
-	Estudos redundantes de alguma autoria
-	Abordagens genéricas em RE no ES
-	Estudos em andamento

Conforme ilustração da Figura 3.1 foram selecionados das bases eletrônicas um total de 12.732 estudos, depois da remoção automática de estudos duplicados restaram 11.403 estudos para análise dos títulos e palavras chaves obedecendo os critérios de inclusão e exclusão. Na etapa seguinte restaram 557 estudos para análise do *abstract*. Ao final das análises restaram 75 estudos que passaram por uma análise de qualidade.

Figura 3.1 Processo de revisão sistemática da literatura.

Fonte: Autoria própria

3.3.4 Análise de qualidade

A avaliação da qualidade (AQ) dos estudos selecionados foi realizada por meio de uma técnica de credibilidade, integridade e relevância dos estudos selecionados conforme proposto por KEELE (2007); PETERSEN et al. (2008). Na avaliação dos estudos selecionados foram utilizados os instrumentos descritos na Tabela 3.5. Os estudos selecionados foram avaliados em conformidade com 12 critérios de qualidade, AQ1, AQ2, AQ3, AQ4, AQ5, AQ6, AQ7, AQ8, AQ9, AQ10, AQ11, AQ12 e foram escolhidos em concordância com a literatura, com exceção de AQ7 e AQ11 que foram propostos nessa revisão. Todos os resultados da avaliação de qualidade estão descritos no Apêndice D deste trabalho.

As avaliações de cunho objetivo (AQ2, AQ7 e AQ8), têm com possíveis respostas, Sim ou Não. Para resposta Sim, é atribuído ao estudo 1,0 ponto, se a resposta for Não, é atribuído 0,0 ponto. As avaliações de sentido subjetivo (AQ1, AQ3, AQ4, AQ5, AQ9, AQ10

e AQ11), as respostas possíveis serão Sim, Não ou Parcial, em relação a resposta Parcial, sua pontuação é atribuída o valor de 0,5 ponto, pois essa resposta não atende de forma direta a questão levantada nessa avaliação. A questão AQ6 foi adotada com propósito de identificar o número de fases da ER que o estudo aborda. Dessa forma a resposta para essa questão foi determinada considerando as 05 fases da ER (elicitação, análise e negociação, especificação, validação e gerenciamento) e segue a seguinte fórmula: **(Número de Fases)/(Total de Fases)**, o numerador da equação é caracterizado pelo número de fases que o estudo aborda e o denominador caracteriza-se pelo número total de fases da ER definido anteriormente. A questão de avaliação AQ12, foi definido o valor de 1,0 ponto para estudos aplicados na indústria e 0,5 pontos se não.

Tabela 3.5 Questões de avaliação de qualidade

Questões	Pontuação
AQ1 - Há uma justificativa para o motivo pelo qual o estudo foi realizado?(GALSTER et al., 2014)	S=1, N=0, P=0.5
AQ2 - O estudo é baseado em pesquisa ou é apenas uma lição aprendida com base na opinião de especialistas?(DYBAA; DING-SOYR, 2008)	S=1, N=0
AQ3 - Existe uma exposição clara dos objetivos da pesquisa?(DYBAA; DINGSOYR, 2008)	S=1, N=0, P=0.5
AQ4 - A abordagem proposta é descrita de forma clara? (ACHI-MUGU et al., 2014)	S=1, N=0, P=0.5
AQ5 - O contexto da pesquisa foi descrito em um nível adequado (indústria, laboratório, produtos utilizados e assim por diante)? (ACHI-MUGU et al., 2014)	S=1, N=0, P=0.5
AQ6 - Quantas fases da engenharia de requisitos apoiam o estudo?(DERMEVAL et al., 2015)	$X/5 = \frac{\text{(Número de Fases)}}{\text{(Total de Fases)}}$
AQ7 - As abordagens suportam mais de um domínio?	S=1, N=0
AQ8 - O estudo é apoiado por alguma ferramenta?(DERMEVAL et al., 2015)	S=1, N=0
AQ9 - Os resultados do estudo são discutidos? (DERMEVAL et al., 2015)	S=1, N=0, P=0.5
AQ10 - Há discussão de alguma limitação do estudo? (DING et al., 2014)	S=1, N=0, P=0.5

Questões	Pontuação
AQ11 - Há uma descrição das questões em aberto do estudo? (DING et al., 2014)	S=1, N=0, P=0.5
AQ12 - O estudo também agrega valor ao setor da indústria? ((ACHI-MUGU et al., 2014); (DYBAA; DINGSOYR, 2008))	S=1, P=0.5

No final da avaliação de qualidade do estudo selecionado é calculado seu escore de qualidade por meio da seguinte equação: $[(\text{soma das respostas})/12] * 100$, o numerador é caracterizado pela soma da pontuação das respostas de (AQ1, AQ2, AQ3, AQ4, AQ5, AQ6, AQ7, AQ8, AQ9, AQ10, AQ11 e AQ12), o denominador é o número de questões de avaliação, após a divisão, multiplicamos por 100 para obter o escore final que será um valor de 0 a 100.

3.3.5 Síntese e análise dos dados

A análise de qualidade e o processo de extração dos dados dos estudos selecionados foram realizados conforme leitura da introdução, conclusão e o texto completo do estudo. Dessa forma o conteúdo foi armazenado e posteriormente utilizado de forma adaptada do proposto por KEELE (2007), para que fosse avaliado em uma nova etapa de modo que alcance os objetivos desse trabalho. A organização desses dados mostram as seguintes informações dos estudos: identificação do estudo, autores, ano, título, citações do Google Acadêmico, país, fonte do artigo, tipo de estudo, contexto de aplicação, método de pesquisa, índice de qualidade e a maturidade da pesquisa. Todas essas informações foram organizadas e importadas para uma planilha eletrônica por meio da ferramenta StArt.

3.4 Resultados e Discussão

Nesta Seção, são apresentados e discutidos os resultados das informações extraídas dos 75 estudos selecionados na revisão, bem como todas as suas características que foram utilizadas para responderem as questões de pesquisa.

3.4.1 Visão geral e características dos estudos

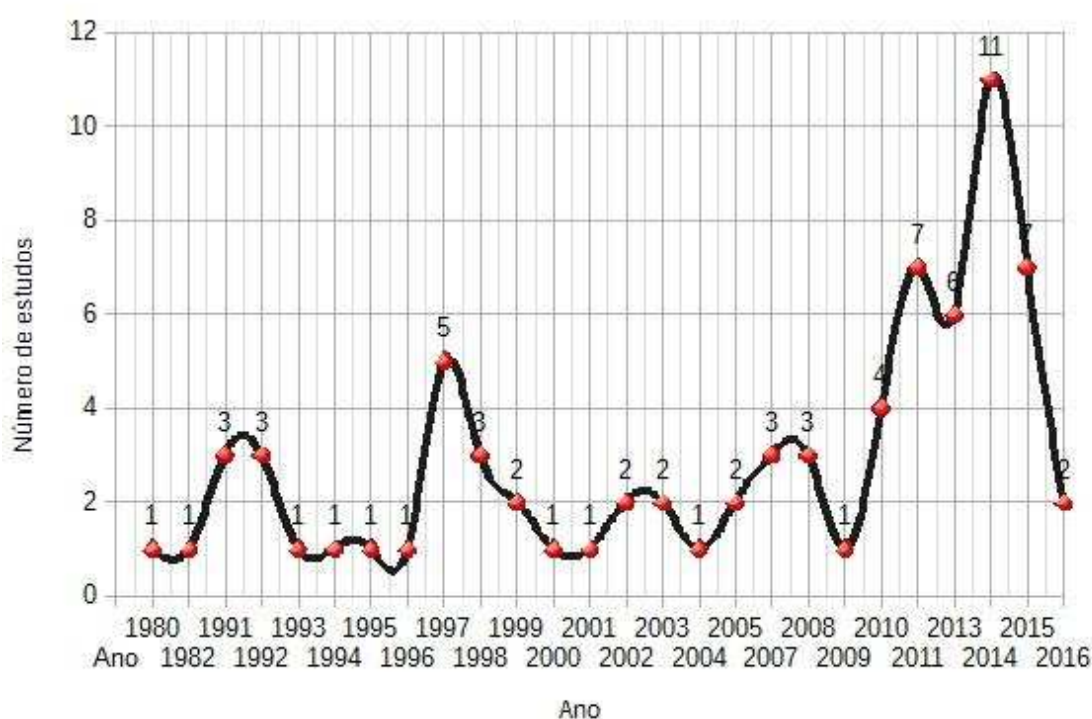
De acordo com os dados extraídos nessa revisão, nas próximas linhas é apresentado uma visão geral e características dos estudos selecionados.

3.4.1.1 Visão temporal dos estudos

Os estudos analisados nesta revisão foram publicados entre 1980 e 2016, pois na década de 70 nenhum trabalho foi identificado. De acordo com a Figura 3.2, entre o período de 1980 a 1996, a quantidade de estudos publicados permanece uniforme com quantitativos de

01 a 03 estudos por ano. No ano de 1997, nota-se uma crescente no número de publicações, um total de 05, já a partir de 1998 até 2007, apresenta novamente uma uniformidade no número de estudos publicados com variação entre 01 a 03. No intervalo entre 2008 a 2015, observa-se um aumento constante de publicações, com destaque para o ano de 2014, que teve um total de 11 estudos publicados (14,66%), esse foi o ano com maior número de trabalhos publicados, acompanhado de 2011 e 2015 com 07 trabalhos cada (9,33%), seguido de 2013 com 6 estudos (8,00%), 2008 e 2010 com 4 estudos (5,33%). O ano de 2016 foram 02 publicações (2,66%), no entanto a contabilização foi feita até o mês de junho, ficando de fora os meses que apresentam um número maior de conferências na área dessa revisão, pois a revisão foi encerrada nesse mês.

Figura 3.2 Visão temporal dos estudos.



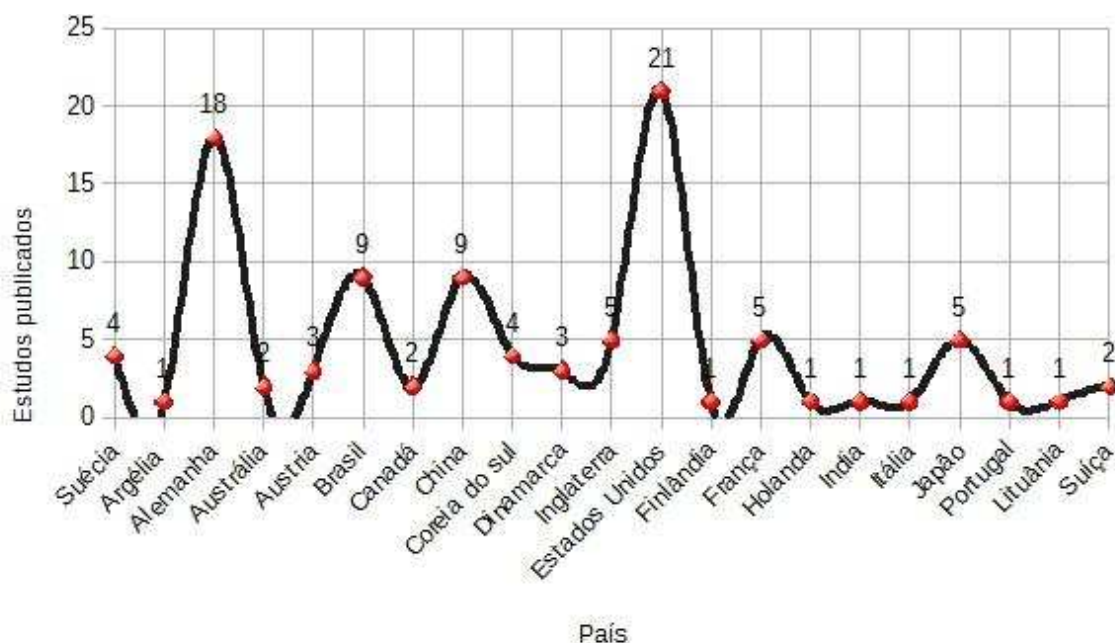
Fonte: Autoria própria

3.4.1.2 Origem dos estudos

Ainda em conformidade com os dados extraídos dos 75 estudos selecionados, a Figura 3.3, apresenta os países que mais contribuíram para a área dos embarcados. Verifica-se que os Estados Unidos e Alemanha possui um número de estudos publicados, 21 e 18 (28% e 24%) respectivamente, bem superior aos demais países, em seguida o Brasil e a China possuem 09 estudos cada (12%), a França, Inglaterra e Japão com 05 trabalhos cada (6,66%), seguido de Suécia e Coreia do Sul com 04 trabalhos (5,33%) e o restante dos países com número de

publicações entre 01 e 03.

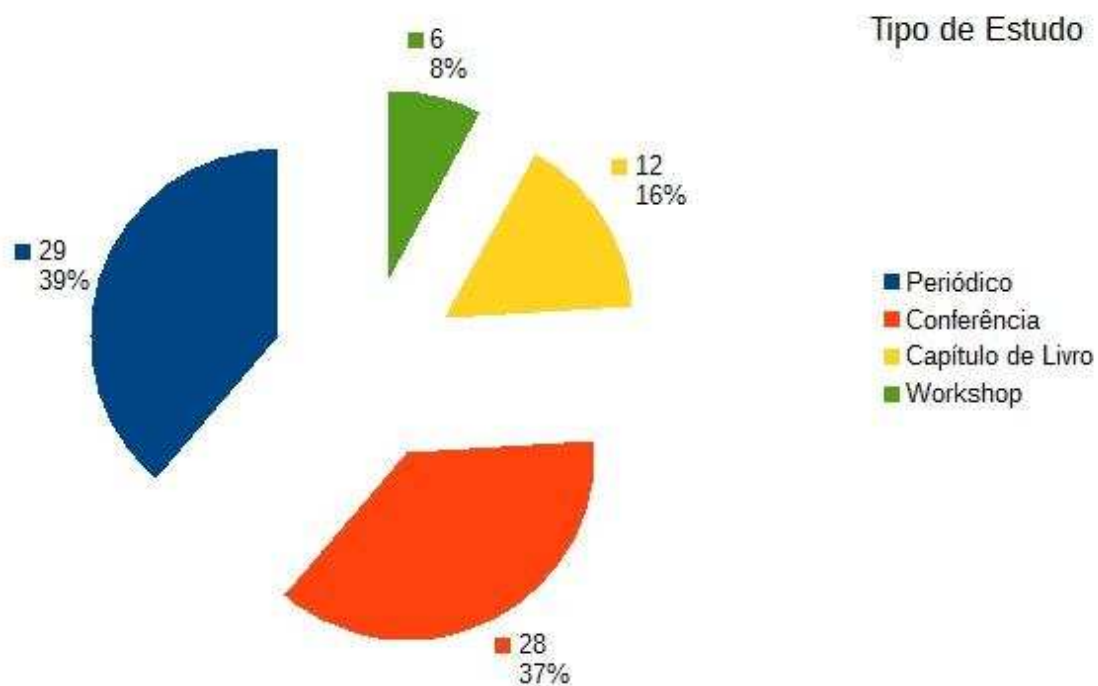
Figura 3.3 País de autoria dos estudos.



Fonte: Autoria própria

3.4.1.3 Tipo do estudo

Na Figura 3.4 é apresentado uma visão do tipo dos estudos que foram selecionados. observa-se que 29 dos estudos (39%) são do tipo periódico, 28 (37%) foram publicados em conferências, 12 trabalhos (16%) são capítulos de livros e 06 trabalhos (8%) foram publicados em *workshops*.

Figura 3.4 Tipo de publicação dos estudos.

Fonte: Autoria própria

3.4.1.4 Tipo de contribuição

Foram descritos também os tipos de contribuição de cada estudo selecionado. Conforme PETERSEN et al. (2008), a análise dos resultados concentra-se na apresentação das frequências das publicações para cada categoria. Isso possibilita ver quais categorias foram enfatizadas em pesquisas anteriores e, assim, identificar lacunas e possibilidades de pesquisa futura. A Tabela 3.6 mostra qual o tipo de contribuição/categoria que o estudo ofereceu, destacamos o tipo abordagem, que foi presente em 44 estudos (58,66%) dos 75 selecionados, seguido pelo tipo método, que teve 16 trabalhos (21,33%). Estudos que tiveram contribuição com uma ferramenta foram 11 trabalhos (14,66%), estudos que contribuíram com um *framework* foram 09 trabalhos cada (12%), estudos que contribuíram com um processo foram 08 trabalhos (10,66%) e os estudos que contribuíram com uma técnica foram 04 trabalhos (5,33%).

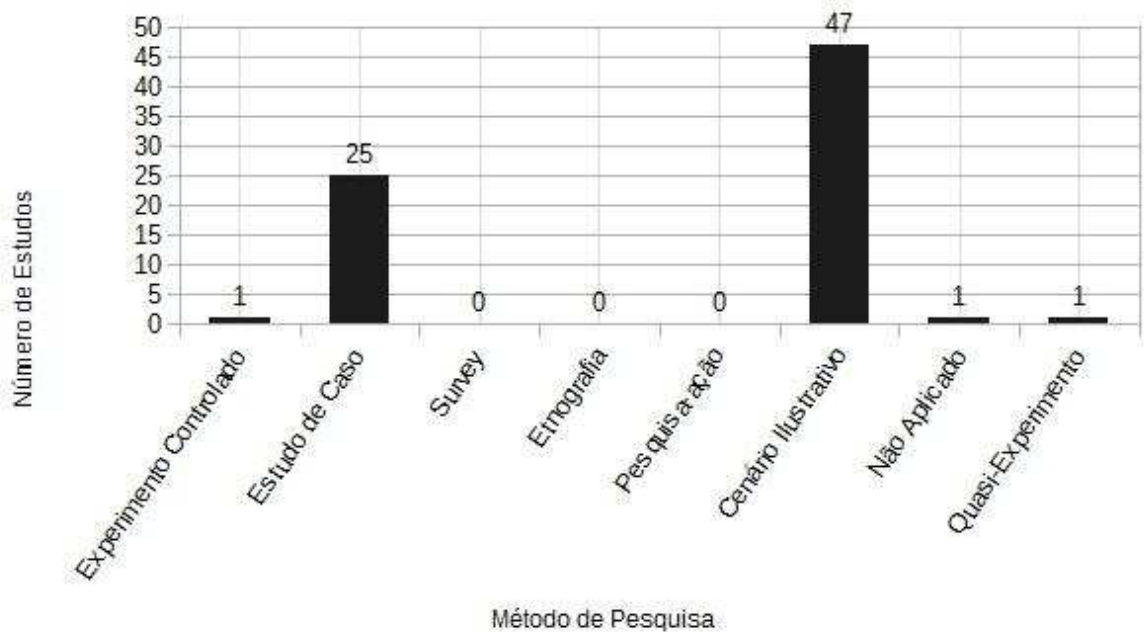
Tabela 3.6 Tipo de contribuição do estudo

Contribuição	Identificação dos estudos	Quantidade	%
Abordagem	S01, S09, S14, S15, S19, S21, S22, S24, S25, S28, S30, S33, S34, S37, S38, S39, S40, S42, S43, S44, S45, S46, S47, S51, S52, S53, S54, S55, S57, S58, S60, S62, S64, S65, S66, S67, S68, S69, S70, S71, S72, S73, S74, S75	44	58,66
Método	S03, S10, S12, S13, S16, S18, S20, S27, S35, S36, S56, S59, S68, S69, S72, S74	16	21,33
Ferramenta	S05, S06, S07, S20, S22, S28, S47, S56, S59, S66, S75	11	14,66
Modelo	S04, S07, S17, S18, S36, S66, S70, S71, S75	09	12
<i>Framework</i>	S02, S11, S14, S23, S40, S48, S49, S50, S61	09	12
Processo	S06, S08, S09, S32, S35, S36, S41, S63	08	10,66
Técnica	S05, S26, S29, S31	04	5,33

3.4.1.5 Método de pesquisa

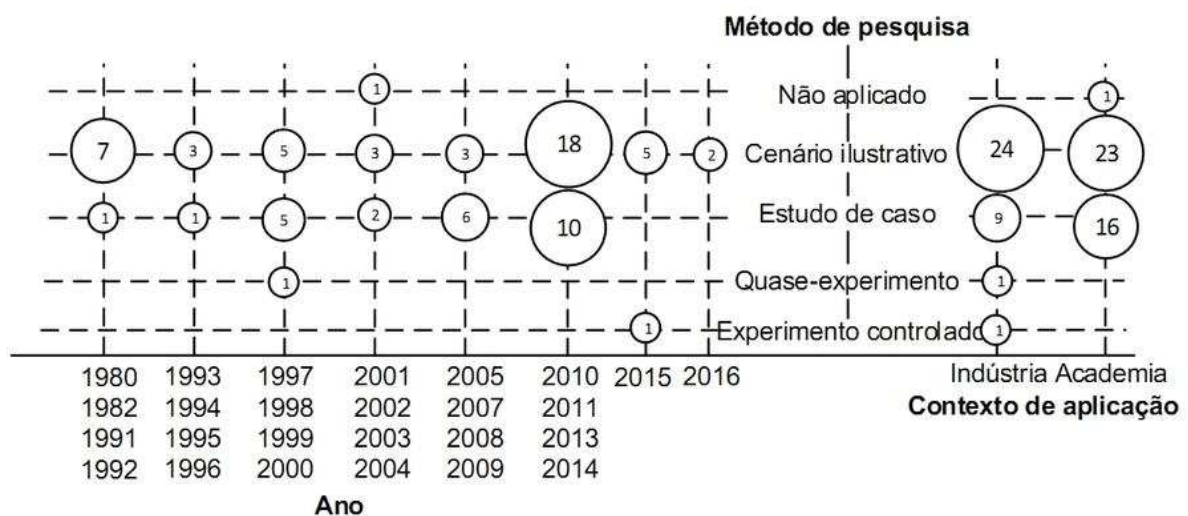
Refrente ao método de pesquisa dos estudos selecionados, sua classificação foi feita de acordo com DERMEVAL et al. (2015), que se basearam nas propostas por EASTERBROOK et al. (2008) (experimento controlado, quasi-experimento, estudo de caso, *survey*, etnografia e pesquisa-ação). Foram adicionadas mais duas categorias definidas conforme DERMEVAL et al. (2015), cenário ilustrativo e não aplicado. O cenário ilustrativo refere-se aos estudos que apresentam apenas um cenário ou um exemplo de uso para avaliar suas contribuições e a categoria não aplicado, refere-se a estudos que não apresentam nenhum método de pesquisa.

De acordo com a Figura 3.5, a maioria dos métodos de pesquisa dos estudos selecionados foram classificados como cenário ilustrativo 47 trabalhos (62,66%), logo em seguida o método estudo de caso com 25 artigos (33,33%), os métodos experimento controlado, quasi-experimento e não aplicado, tiveram 01 trabalho cada (1,33%), *survey*, etnografia e pesquisa-ação não foram encontradas classificações de métodos nos estudos. De acordo com a Figura 3.5 e com esses dados extraídos, nota-se que a pesquisa na área de engenharia de requisitos nos sistemas embarcados são focadas em pequenos exemplos e estudos de caso como avaliação, dessa forma os outros métodos não são utilizados.

Figura 3.5 Métodos de pesquisa dos estudos

Fonte: Autoria própria

A Figura 3.6 apresenta um gráfico representando três características dos estudos citadas anteriormente: ano de publicação, contexto de aplicação e método de pesquisa. Analisando o lado esquerdo da Figura 3.6, observa-se o número de estudos sobre seu método de pesquisa e seu ano de publicação. No lado direito identifica-se o quantitativo de estudos e a proporção sobre o contexto de aplicação (acadêmico ou industrial). Nota-se que a maioria dos estudos concentram-se a partir do ano de 2010 a 2016, totalizando 36 trabalhos (48%) e que o contexto de aplicação mais utilizado foi o acadêmico com 40 estudos (53,33%).

Figura 3.6 Dimensão dos métodos de pesquisa aplicada nos estudos

Fonte: Autoria própria

3.4.2 Q1 - Como as abordagens contribuem para a solução de problemas de engenharia de requisitos no domínio de sistemas embarcados?

Como resposta a primeira questão de pesquisa, a Figura 3.7 mostra que a maioria dos estudos contribuíram para resolver problema de Ambiguidade, inconsistência e / ou incompletude, totalizando 18 trabalhos (24%). As soluções encontradas foram desde a criação de uma abordagem de validação, uma especificação de requisitos bem definida, até uma definição de uma nova linguagem específica de engenharia de requisitos (DSL) e de padrões de linguagem. O segundo problema mais frequente foi Gerenciamento de Requisitos / evolução, com 15 estudos (20%). As soluções encontradas nos estudos abordaram a criação de metamodelos de rastreabilidade de requisitos, técnicas para documentação dos requisitos de software e processo de reutilização de requisitos.

Na sequência vem problemas de especificação de requisitos e problemas de requisitos não-funcionais com 13 (17,33%) e 12 (16%) estudos respectivamente. Foram abordadas como soluções para esses problemas, o desenvolvimento de linguagens formais, o uso de linguagens naturais para especificação, abordagens com verificação de requisitos e processos para desenvolver uma especificação de requisitos com soluções para resolver problemas. Referente aos requisitos não-funcionais, foram encontradas definições para modelar os RNFs, bem como métodos para especificar e verificar os RNFs, também foram proposta soluções por meio de linguagens naturais para resolverem os problemas dos RNFs nos sistemas embarcados. Esses dois problemas (especificação e requisitos não-funcionais) foram objetos de estudo principais deste trabalho e serão abordados no Capítulo 4.

Seguindo a sequência, o problema de validação de requisitos teve um total de 12 estudos (16%). Soluções são a extensão de uma linguagem formal, uma metodologia para a definir a verificação dos requisitos, e uma abordagem orientada por modelo para ER.

Problemas que não foram claramente detalhados totalizaram 06 estudos (8%) e trabalhos que não especificaram o problema foram 05 (6,66%). Cinco estudos (6,66%) relataram problemas de elicitação de requisitos. As soluções definem o desenvolvimento de técnicas de apoio à engenharia de requisitos e ao tratamento de casos específicos de elicitação e especificação de requisitos. A engenharia orientada por modelos e o uso de *frameworks* para transformações de um modelo para o outro foram as formas abordadas para solucionar os 04 estudos (5,33%) com problemas de integração entre requisitos e arquitetura. Em seguida, a interoperabilidade dos modelos de requisitos são abordados por 02 trabalhos (2,66%) e soluções como, a definição de um conceito genérico de gerenciamento de domínios e um processo de desenvolvimento baseado em modelos.

Uma abordagem de engenharia de requisitos para sistemas de controle em tempo real é proposta como solução para o único (1,33%) problema encontrado de representação do conhecimento de domínio. O último problema relacionado na Figura 3.7 é o de comunicação

de requisitos, com apenas 01 abordagem (1,33%). A solução sugerida é uma metodologia que descreve a fases de elicitação, modelagem, rastreabilidade, verificação e validação dos requisitos.

Figura 3.7 Problemas de ER nos sistemas embarcados



Fonte: Autoria própria

3.4.3 Q2 - Quais as fases do processo de ER apoiadas pelas abordagens?

Em conformidade com a Tabela 3.7, nota-se que todas as fases da engenharia de requisitos foram contempladas nos estudos selecionados. A fase de especificação foi a que obteve o maior número de abordagens nos estudos, 56 trabalhos (74,66%), esse resultado foi esperado, uma vez que existem diversas maneiras de especificar requisitos, linguagem natural, linguagem formal, linguagem semiformal, linguagem de restrição, entre outras.

A validação foi a segunda fase mais presente nos trabalhos, 33 no total (44%). Esse número expressivo na validação também foi esperado, pois muitos dos sistemas embarcados são de natureza crítica e uma falha qualquer pode ocasionar grandes perdas materiais e danos físicos dependendo do domínio que o sistema esteja inserido.

Na sequência, a fase de análise e negociação foi representada com 30 estudos (40%). Percebe-se que 26 trabalhos (34,66%) estão presentes na fase de especificação, ou seja, para engenharia de requisitos em sistemas embarcados a fase de especificação (hardware, ambiente ou software) a análise é muito importante na condução em conjunto com esse

processo. A análise da engenharia de requisitos torna-se interessante e complexa, uma vez que o desenvolvimento de sistemas embarcados envolve diferentes perfis de *stakeholders*.

O quantitativo de estudos diminuiu na fase de elicitação, 19 trabalhos (25,33%), no entanto nota-se um interesse da comunidade de sistemas embarcados em fornecer alguma orientação para a elicitação de requisitos. Verifica-se também que 10 (13,33%) desses 19 trabalhos foram selecionados dos mesmos estudos que abordaram especificação, ou seja, a maioria dos estudos dessa fase partilharam do mesmo estudo de especificação, com exceção de S08, S18, S22, S40, S48, S59 e S61.

Em comparação com outras fases, o gerenciamento obteve um número baixo de estudos, 14 trabalhos (18,66%), portanto fica evidente que pouca atenção é dada a gestão pela comunidade dos embarcados, apesar dessa área conter muitos problemas em aberto.

Entre os 75 estudos selecionados, apenas um (S07) reuniu todas as fases do processo de ER e um outro (S33), abordou quatro fases. Dito isto, podemos concluir que a grande maioria dos estudos contempla apenas parte do processo de ER nos sistemas embarcados.

Tabela 3.7 Fases da ER suportadas pelos estudos

Fase	Estudos	Quantidade	%
Especificação	S02, S03, S05, S06, S07, S09, S10, S11, S12, S13, S14, S15, S16, S17, S19, S20, S21, S23, S24, S26, S27, S28, S29, S32, S33, S34, S35, S37, S38, S42, S43, S44, S45, S46, S47, S49, S50, S51, S52, S54, S55, S56, S57, S58, S60, S63, S64, S65, S66, S68, S69, S70, S71, S73, S74, S75	56	74,66
Validação	S01, S05, S07, S08, S09, S10, S11, S12, S17, S18, S19, S21, S22, S28, S31, S34, S39, S41, S44, S45, S49, S51, S53, S54, S55, S57, S58, S61, S62, S64, S66, S71, S72	33	44
Análise e Negociação	S02, S03, S05, S06, S07, S11, S14, S16, S17, S19, S24, S25, S33, S34, S36, S37, S38, S40, S49, S50, S51, S55, S56, S64, S67, S68, S69, S70, S73, S75	30	40
Elicitação	S07, S08, S13, S16, S18, S22, S26, S32, S33, S40, S42, S48, S56, S58, S59, S61, S63, S70, S73	19	25,33
Gerenciamento	S04, S07, S08, S09, S18, S30, S33, S36, S46, S47, S68, S69, S74, S75	14	18,66

3.4.4 Q3 - Que tipo de requisito (funcional e/ou não-funcional) foi apoiado pelas abordagens?

Ao verificar a Tabela 3.8, identifica-se nos estudos a presença dos requisitos funcionais e não-funcionais. Como esperado, a grande maioria, 63 trabalhos (84%) abordaram os

requisitos funcionais, no entanto, trinta e um (31) estudos (41,33%) que abordaram requisitos funcionais isoladamente. Já os requisitos não-funcionais, foram abordados por 44 estudos (58,66%), no entanto observa-se que 12 estudos (16%) abordaram requisitos não-funcionais isoladamente. Já os estudos que abordaram ambos os requisitos foram 32 trabalhos (42,66%). Esse resultado indica que há uma preocupação dos autores com uma especificação mais abrangente, isto é que consideram ambos os tipos de requisitos. Como citado anteriormente, os requisitos não-funcionais foram abordados por 44 (58,66%) dos estudos, porém as questões de pesquisa Q1, Q2 e mais especificamente a próxima questão (Q4) apontam que a especificação de requisitos não-funcionais nos sistemas embarcados é um desafio e que merece uma atenção especial.

Tabela 3.8 Abordagem dos requisitos funcionais e não-funcionais

Requisitos	Estudos	Quantidade	%
Funcionais	S01, S09, S10, S11, S14, S15, S16, S20, S21, S23, S26, S27, S28, S29, S30, S32, S36, S37, S41, S44, S45, S53, S54, S55, S56, S58, S59, S60, S67, S71, S72	31	41,33
Não-funcionais	S03, S13, S24, S25, S34, S35, S40, S43, S49, S57, S63, S64	12	16
Ambos	S02, S04, S05, S06, S07, S08, S12, S17, S18, S19, S22, S31, S33, S38, S39, S42, S46, S47, S48, S50, S51, S52, S61, S62, S65, S66, S68, S69, S70, S73, S74, S75	32	42,66

3.4.5 Q4 - Quais as questões em aberto das abordagens encontradas?

Por meio dos dados extraídos dos 75 estudos selecionados, foi possível também identificar os problemas em aberto da engenharia de requisitos nos sistemas embarcados. Conforme a Tabela 3.9, é perceptível que 26,66% dos estudos não apresentaram questões em aberto. Esse resultado foi inesperado, pois pode levar a uma conclusão equivocada que esses estudos estão concluídos e não há mais nada a contribuir. Na sequência, a especificação de requisitos não-funcionais foi a questão em aberto e trabalhos futuro mais abordados nos estudos (12 trabalhos, 16%). Os principais pontos sobre a especificação de requisitos nesses trabalhos eram relativos a verificabilidade, usabilidade, performance, confiabilidade e segurança, tendo em vista o fator crítico da necessidade desses requisitos nos sistemas embarcados. Em seguida, a modelagem de requisitos apareceu em 11 trabalhos (14,66%) e foi o segundo problema em aberto mais frequente, pois é relacionado a modelagem de sistemas (ambiente, *hardware*, requisitos de *software*) e requisitos funcionais e não-funcionais. O terceiro problema em aberto mais citado foi o desenvolvimento de ferramentas automatizadas com

10 estudos (13,33%), essa questão visa o desenvolvimento de ferramentas para apoiar as abordagens dos estudos relacionados. Os trabalhos que não foram possíveis identificar as questões em aberto de forma clara, contabilizaram um total de 07 estudos (9,33%). Questões em aberto que ajudam a reduzir a ambiguidade, inconsistência e/ou incompletude tiveram um total de 06 trabalhos (8%). Também com 06 trabalhos, a validação de requisitos foi abordada como problemas em aberto. Como exemplos dessa questão o estudo S48 visa a investigação de validação de requisitos como trabalho futuro, o estudo S57 pretende realizar a validação por verificação formal, o S62 quer saber como validar uma simulação de execução e os estudos S71 e S72, pretendem utilizar verificação formal para auxiliar os sistemas de desenvolvimento e simplificar a verificação de requisitos. Cinco estudos (6,66%) destacaram como questões em aberto a validação do estudo proposto. O objetivo era validar o estudo em questão por meio da aplicação na indústria ou em um estudo de caso. Em seguida 04 (5,33%) estudos indicam a interoperabilidade dos modelos de requisitos como questão em aberto. Já a migração entre ferramentas de gerenciamentos de requisitos é discutida pelo estudo S07, esse trabalho é um dos 04 estudos (5,33%) que abordam nas questões em aberto a interoperabilidade dos modelos de requisitos. em seguida a gestão / evolução de requisitos, totalizaram também 04 estudos. A evolução entre os requisitos comportamentais e o design funcional são exemplos das questões em aberto propostas por S44, bem como o conceito de variabilidade por S47, a escalabilidade de domínios problemáticos por S63 e o uso de links de rastreabilidade por S69. Já a especificação de requisitos foram questões em aberto de 04 trabalhos. Esses estudos têm como questões a especificação funcional (S21), protótipo (S28), padrão (S67) e modelo (S72). Apenas 02 estudos (S09 e S15, 2,7%) estão interessados em aplicar suas abordagens para outros domínios e não apenas aquele discutido no estudo. Dois trabalhos (2,7%) discutem questões relacionadas com a elicitación de requisitos, S61 incluirá uma melhoria da notação e S70, um guia sistemático para a elicitación. Ao final da Tabela 3.9, observa-se um estudo (S04) que está interessado em evoluir o trabalho para representar o conhecimento do domínio e um estudo (S38) que visa a integração entre requisitos e arquitetura. Como a especificação de RNFs foi problema em aberto mais citado (12 estudos, 16%), no próximo capítulo será realizada uma análise mais detalhada deste desafio.

Tabela 3.9 Problemas em aberto da ER nos estudos selecionados

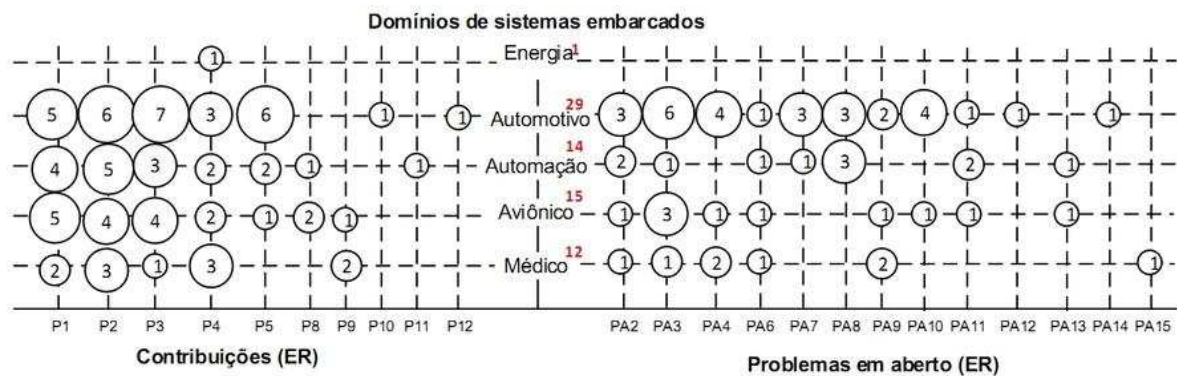
ID	Problemas em aberto	Estudos	Quantidade	%
PA1	Não informado	S02, S05, S08, S16, S19, S24, S29, S33, S34, S35, S40, S42, S43, S46, S49, S50, S51, S53, S55, S56	20	26,66
PA2	Especificação de RNF	S01, S13, S20, S27, S37, S41, S45, S48, S60, S61, S63, S64 S01, S03, S13, S22,	12	16,00
PA3	Modelagem de requisitos	S26, S39, S52, S63, S68, S74, S75 S03, S07, S10, S11,	11	14,66
PA4	Ferramentas de suporte auto- matizada	S15, S17, S20, S37, S66, S71	10	13,33
PA5	Não claro	S06, S12, S14, S23, S25, S32, S54	07	9,33
PA6	Ambiguidade, inconsistência e/ou incompletude	S03, S10, S11, S36, S65, S70	06	8,00
PA7	Validação de requisitos	S48, S57, S62, S63, S71, S73	06	8,00
PA8	Validação de abordagem pro- posta	S30, S31, S59, S61, S66	05	6,66
PA9	Interoperabilidade dos modelos de requisitos	S07, S17, S18, S66	04	5,33
PA10	Gerenciamento/evolução de re- quisitos	S44, S47, S63, S69	04	5,33
PA11	Especificação de requisitos	S21, S28, S67, S72	04	5,33
PA12	Suporte a outros domínios	S09, S15	02	2,66
PA13	Elicitação de requisitos	S61, S70	02	2,66
PA14	Representação do conheci- mento do domínio	S04	01	1,33
PA15	Integração entre requisitos e ar- quitetura	S38	01	1,33

3.4.6 Q5 - Quais são os domínios que as abordagens suportam?

Ao analisar a Figura 3.8, percebe-se que o domínio mais frequentes nos embarcados é o automotivo, com total de 29 trabalhos (38,66%), seguido pelo domínio de tecnologia aviônica, com 15 estudos (20%). Na sequência o domínio automação tecnológica, com 14 estudos (18,66%) e tecnologia médica com 12 trabalhos (16%). Concluindo a relação de domínios abordados, o energético possui 01 estudo (1,33%).

Analisando novamente a Figura 3.8, nota-se que o domínio automotivo, a contribuição P3 (especificação de requisitos) está relacionada com 07 estudos, enquanto o problema em aberto OP11(especificação de requisitos) tem apenas 01 estudo. Dessa forma, podemos concluir que a especificação de requisitos no domínio automotivo é bem explorada. A contribuição P4 (requisitos não-funcionais) está associada a três estudos, e o problema aberto OP2 (requisitos não-funcionais) também tem três estudos. Por meio deste resultado, podemos dizer que embora os requisitos não-funcionais sejam parcialmente explorados, ainda possuem questões em aberto a serem respondidas (vide Capítulo 4). Seguindo, essa forma de análise é possível identificar se os demais domínios estão sendo bem explorados, não explorados ou parcialmente explorados.

Figura 3.8 Domínios de sistemas embarcados.



Fonte: Autoria própria

3.4.7 Q6 - Quais requisitos devem ser considerados durante o desenvolvimento de sistemas embarcados?

A investigação da RSL da engenharia de requisitos dos SE resultou na identificação de 56 requisitos relevantes no desenvolvimento dos SE. Desses 56 requisitos, 22 (39,29%) são requisitos não-funcionais. Geralmente, esses RNFs consistem em atributos de qualidade (por exemplo, desempenho, confiabilidade, capacidade de sobrevivência, robustez, safety, safety critical, security, tempo real); Energia, economia de combustível, baixa emissão de gases, consumo de tempo, consumo de recursos) e requisitos de interfaces externas (por exemplo, interface do usuário, usabilidade, interação homem-máquina). Conforme a Tabela 3.10, os requisitos identificados foram separados em 4 níveis (empresa, sistema, contexto, requisitos) para dividir as tarefas da engenharia de requisitos, facilitar a identificação e o entendimento desses requisitos (PEREIRA et al., 2016). A maioria dos requisitos, com um número de 39 (70%) foram do nível de requisitos, seguido pelo nível da empresa (9 requisitos, 16%), nível do sistema (5 requisitos, 9%) e nível de contexto (3 requisitos, 5%).

Tabela 3.10 Requisitos em diferentes ponto de vista

Nível	Requisitos identificados	Quantidade	%
Nível de requisitos	Espaço de exploração; Requisitos Operacionais; Requisitos de detecção de falhas; Requisitos de monitoramento; Particionamento de Requisitos; Requisitos funcionais de segurança; Requisitos de fabricação; Requisitos de reutilização de componentes; Programas de restrições; Requisitos relacionados ao <i>hardware</i> ; Restrições de <i>hardware</i> ; Requisitos e normas de infraestrutura de produtos; Integração entre <i>hardware</i> e <i>software</i> ; Recursos compartilhados; Detecção de falhas; tolerância a risco; Urgência de transições; requisitos de desempenho; Requisitos de <i>safety</i> ; Requisitos de interação homem-máquina; Cronometragem de requisitos; Requisitos de confiabilidade; Requisitos de <i>security</i> ; <i>interface</i> de usuário; Área; Peso; Consumo de energia; Total de energia; Usabilidade; Capacidade de sobrevivência; Robustez; Requisitos em tempo real; Custo de desenvolvimento; Economia de combustível; Baixa emissão de gases; Tempo de consumo; Consumo de recursos; Requisitos temporais; Requisitos críticos de <i>safety</i> ;	39	70
Nível de empresa	Lidar com múltiplos stakeholders; Pressão do mercado; Migração de normas; Leis de controle de voo; Requisitos do processo de certificação; Exigências e padrões de infraestrutura de processos; Ambiente de execução; Fornecedores;	9	16
Nível de sistema	Comportamento desejado; Comportamento indesejado; Pressupostos ambientais; Requisitos comportamentais; Comportamento do sistema;	5	9
Nível de contexto	Pressupostos ambientais; Requisitos comportamentais; Comportamento do sistema;	3	5

Nível da empresa. As exigências classificadas neste nível estão relacionadas aos stakeholders envolvidos no desenvolvimento de sistemas embarcados, padrões, políticas e leis que devem orientar o processo de desenvolvimento, o ambiente de execução dos SE e as relações com fornecedores e fabricantes. O mercado é compartilhado por fornecedores, fabricantes e fornecedores de ferramentas que precisam encurtar a troca de processos entre eles e a reutilização de software em diferentes linhas de produtos. Esta questão levanta o desafio da redução do tempo de desenvolvimento. Já o tratamento de múltiplos stakeholders nos SE é uma exigência problemática, pois requer especialistas em domínios de várias áreas, como engenheiros mecânicos para contexto físico, engenheiros elétrico para o contexto de

hardware, bem como especialistas em Interface Homem-Máquina (IHM) para os aspectos de usabilidade, além de engenheiros de requisitos e desenvolvedores de software (KRUGER et al., 2010).

Nível de sistema. Os requisitos classificados nesse nível estão relacionados ao comportamento do sistema, ao ambiente exigido pelo sistema e às funcionalidades que o sistema deve executar. O Comportamento desejado e comportamento do sistema depende dos desejos do cliente para o produto embarcado de acordo com GOMES (2009). Dessa forma, as expectativas do cliente deve ser gerenciadas, bem como o comportamento indesejado e as reações do sistema que devem ser evitadas. Assim, é possível encontrar maneiras diferentes de tratá-los. As premissas ambientais (expectativas) são condições que o ambiente do sistema deve realizar para o funcionamento correto do sistema (BLOUIN; SENN; TURKI, 2011).

Nível de requisitos. Os requisitos classificados neste nível estão relacionados a requisitos funcionais e não-funcionais, requisitos de *hardware*, restrições de *software* e *hardware*, integração entre *hardware* e *software*, requisitos do fabricante e padrões de infraestrutura de produtos.

Em sistemas embarcados, falha ou problemas de experiência ocorrem devido a várias razões, incluindo obstrução ambiental, manipulação incorreta do sistema, atraso de reposta, etc (FRIEDRICH, 2009).

Nível de contexto. Os requisitos nesse nível estão relacionados a pressupostos ambientais, requisitos comportamentais e comportamento do sistema. De acordo com ALI; DALPIAZ; GIORGINI (2010), um contexto é um estado parcial do mundo que é relevante para o objetivo de um ator. Portanto, devemos analisar os requisitos citados anteriormente para identificar e especificar contextos que poderiam afetar a execução do sistema. Requisitos neste nível devem descrever a execução do sistema, o local em que o contexto está localizado e os recursos disponíveis para usar (PEREIRA, 2014). A descrição desses itens pode ser usada para modelar e especificar contextos.

3.4.8 Ameaças à validade

Conforme o pensamento de BITTENCOURT et al. (2015), que cita o viés na seleção dos estudos e a imprecisão na extração dos dados como as principais ameaças à validade das revisões sistemáticas. No entanto, com o propósito de evitar essas ameaças, diretrizes propostas por KEELE (2007) foram seguidas e aplicadas no desenvolvimento de um protocolo de pesquisa confiável e auditável, pois essas diretrizes sistematizam a pesquisa em etapas, que facilita a identificação de erros e eliminação das ameaças intrínsecas à RSL. Esse protocolo foi validado por meio de comparação com outros protocolos de RSL já publicados.

A *string* de pesquisa utilizada em uma RSL tem que ser avaliada diversas vezes para evitar o risco de omissão de estudos relevantes. Esse risco é dependente de como a lista de palavras-chave foi definida, ou seja, se uma palavra-chave de um assunto relevante para

pesquisa não for inserida na *string*, muitos estudos relevantes podem ficar de fora da seleção. Para diminuir essa ameaça, foram levados em consideração as questões de pesquisas e um conjunto de artigos para construir uma taxonomia inicial. Enquanto o protocolo era elaborado, executamos a *string* de busca diversas vezes nas bases científicas digitais. A medida que esse processo se repetia, eram identificadas as falhas e melhorias eram aplicadas na *string* para obter resultados mais satisfatórios.

Durante a extração dos dados, questões e decisões subjetivas podem ter ocorrido, pois em alguns casos, os estudos não forneciam dados claros e objetivos. Portanto, é possível existir classificações erradas nos estudos em que foram aplicados os critérios de inclusão e exclusão.

3.4.9 Estudos relacionados

Como trabalho relacionado, há uma revisão sistemática da literatura sobre elicitación e especificação de requisitos nos sistemas embarcados proposta por SOUSA et al. (2015). Esse estudo difere do nosso no intervalo de tempo, nas atividades de engenharia de requisitos aplicadas, no número de bases digitais pesquisadas, nas perguntas de pesquisa e na abrangência do tema pesquisado. A RSL considerou aproximadamente 4 décadas de estudos publicados sobre ER dos SE, levou em consideração todas as atividades do processo de ER e foram pesquisadas sete bases digitais, enquanto SOUSA et al. (2015), levou em consideração o intervalo de 2000 a 2014, o autor relacionou atividades apenas para duas fases da ER (elicitación e especificação) e sua busca foi feita em seis bases. De acordo com a abrangência da RSL deste trabalho, é importante ressaltar que as questões de pesquisas relacionadas são diferentes, de forma que a RSL deste trabalho tem um propósito mais amplo, ou seja, um alcance maior e contemplar todo o processo de engenharia de requisitos dos sistemas embarcados.

3.5 Considerações Finais

Inicialmente levantamos dados que incluíram técnicas, linguagem/métodos e processos que auxiliaram para analisar os estudos selecionados. Em seguida esta revisão auxiliou nas respostas das questões de pesquisa e atenderam ao objetivo principal que era, entender e obter informações de como a comunidade de desenvolvimento de sistemas embarcados utiliza o processo de engenharia de requisitos, e, assim, fornecer um caminho de pesquisas para trabalhos futuros.

Foram selecionados e analisados setenta e cinco (75) estudos. Os dados extraídos dessa revisão, permitiu identificar informações relevantes para área de ER nos sistemas embarcados como, doze (12) contribuições, cinco (05) domínios relacionados, quinze (15) tipos de problemas em aberto, entre outros dados relevantes apresentados anteriormente. Conforme

resultados e respostas das questões de pesquisa, é necessário uma atenção as contribuições, ambiguidade, inconsistência e/ou incompletude, gerenciamento/evolução de requisitos e a especificação de requisitos, uma vez que elas correspondem por 44% dos estudos.

Por meio dos resultados, podemos concluir que nenhum dos estudos possui um processo de engenharia de requisitos para o domínio dos SE que contemple todas as fases, ou seja, não há um processo de engenharia de requisitos completo para os SE. As abordagens encontradas procuravam adaptar processos de ER usados comumente na engenharia de software, contemplando uma ou duas fases do processo para solução de um problema específico da ER.

Destacamos também a necessidade de automatização de ferramentas de suporte, dado que estudos propõem soluções para problemas que pode ser resolvidos de forma automatizada. A modelagem de requisitos também foi destacada, pois necessita de soluções para se modelar o ambiente, hardware e software.

Ainda em conformidade com os resultados, há uma grande diversidade de problemas em aberto. No entanto destacamos a especificação de requisitos não-funcionais como o problema em aberto mais mencionado nos estudos (12 trabalhos, 16%), tendo em vista que a maioria dos estudos tem suas preocupações voltadas para o requisitos funcionais. Assim, os RNFs não são explorados de forma sistemática, ou seja, por meio de um processo ou método padronizado pela ER nos SE.

Os resultados também evidenciaram e reforçam que uma atenção especial deve ser dada aos RNFs. Apesar de 12 estudos (16%) focarem exclusivamente em RNFs, a RSL indica que a especificação dos requisitos não-funcionais é o problema aberto mais citado (12 trabalhos, 16%). Portanto, se faz necessário uma nova investigação para melhor analisar como os RNFs estão sendo tratados nos SE.

4

ANÁLISE DOS RNFS NOS SISTEMAS EM- BARCADOS

Em virtude do tratamento de requisitos não-funcionais em SE ter sido o problema em aberto mais citado, neste capítulo apresentamos uma análise mais específica deste aspecto. Como fonte de dados usaremos 21 estudos (vide Tabela 4.1 abaixo). Eles incluem os artigos que identificaram esta dificuldade de tratar dos RNFs (S01, S13, S20, S27, S37, S41, S45, S48, S60, S61, S63, S64) como sendo um problema em aberto, bem como aqueles trabalhos que tratam exclusivamente deste tipo de requisito (S03, S13, S24, S25, S34, S35, S40, S43, S49, S57, S63, S64). Portanto, esta análise foi realizada nestes 21 estudos para estender, complementar os resultados da RSL e detalhar informações sobre os RNFs dos SE.

Tabela 4.1 Estudos selecionados sobre os RNFs dos SE

ID	Autores	Títulos
S01	Aceituna, et al.	Interactive requirements validation for reactive systems through virtual requirements prototype
S03	Aoyama, et al.	AORE (aspect-oriented requirements engineering) methodology for automotive software product lines
S13	Carvalho, et al.	Using Aspect-Oriented Concepts in the Requirements Analysis of Distributed Real-Time Embedded Systems
S20	Faulk, et al.	The Core method for real-time requirements
S24	Galle, et al.	Capturing and verifying performance requirements for hard real time systems
S25	Galvao et al.	A case study using a protocol to derive safety functional requirements from Fault Tree Analysis

ID	Autores	Títulos
S27	Hansen, et al.	From safety analysis to software requirements
S34	Kwon, et al.	Specification and analysis of timing requirements for real-time systems in the CBD approach
S35	Lattermann, et al.	Methodological approach to the requirement specification of embedded systems
S37	Liu, et al.	Extending EAST-ADL2 to support aspectual requirement specification and analysis for automotive software
S40	Markose, et al.	A Systematic framework for structured object-oriented security requirements analysis in embedded systems
S41	Mutz, et al.	Seamless model based development process of automotive systems
S43	Petrov, et al.	An Aspect-Oriented Approach for Designing Safety-Critical Systems
S45	Ravn, et al.	Specifying and verifying requirements of real-time systems
S48	Roudier, et al.	Towards the Model-Driven Engineering of Security Requirements for Embedded Systems
S49	Saeed, et al.	An approach for the risk analysis of safety specifications
S57	Smarandache, et al.	Applicability of SIGNAL in safety critical system development
S60	Vogelsang, et al.	Supporting concurrent development of requirements and architecture: A model-based approach
S61	Wikan et al.	Enforcing safety requirements for industrial automation systems at runtime position paper
S63	Xiaohong et al.	Modeling Timing Requirements in Problem Frames Using CCSL
S64	Zafar, et al.	Integrating safety and security requirements into design of an embedded system

1

4.1 Introdução

No processo de desenvolvimento de sistemas, muitas vezes, os requisitos não-funcionais são tratados de forma inadequada. De acordo com RSL apresentada no Capítulo 3, a especificação de RNFs é abordada como o maior problema em aberto na ER dos sistemas embarcados. Porém, não foram abordados detalhes sobre esses problemas e suas consequências nos diversos domínios dos sistemas embarcados, bem como as contribuições existentes, as linguagens mais utilizadas, os requisitos mais frequentes e os domínios em que esses requisitos são abordados.

¹Os estudos referente aos RNFs selecionados na RSL da engenharia de requisitos dos SE também estão disponíveis em: <https://drive.google.com/file/d/0B4fzzuUJsa71OUhaUVpJYnQ4c1E/view>

À vista disso, por meio de uma análise dos dados extraídos da RSL é possível responder como a comunidade de desenvolvimento de sistemas embarcados está lidando com os RNFs na fase de especificação do processo de ER. Para responder essa pergunta, foram derivadas questões específicas de acordo com os aspectos de ER nos quais estamos interessados. Portanto, foram feitas análises gerais tais como Tipo de contribuição, Tipo da pesquisa, Método de Pesquisa, Contexto da aplicação, bem como perguntas específicas referentes aos RNFs em SE. As respostas a estas perguntas ajudarão comunidade de desenvolvimento de sistemas embarcados na compreensão da identificação e classificação dos requisitos não-funcionais nos SE. Além disso, a análise permitiu relacionar os métodos, ferramentas, linguagens, domínios e detalhar problemas já identificados na revisão sistemática da literatura.

4.2 Questões de pesquisa específicas

Para complementar a RSL referente aos resultados obtidos sobre RNFs, bem como sua especificação, foram aplicadas questões de pesquisa para entender como a comunidade dos embarcados estão lidando com esses RNFs. Dessa forma, as questões servem como um guia para investigar a especificação de requisitos não-funcionais e identificar, métodos, técnicas, modelos, domínios, problemas em aberto e contribuições presentes nesses resultados, assim, fornecer dados mais detalhados, facilitando uma agenda de pesquisa para trabalhos futuros na engenharia de requisitos dos SE. Isto posto, foram propostas as questões conforme Tabela 4.2.

Tabela 4.2 Questões de pesquisa

Questão de pesquisa	Descrição e motivação
Q1. Quais as linguagens usadas para especificação dos RNFs nos sistemas embarcados?	O propósito desta questão é identificar linguagens usadas para especificação de RNFs dos sistemas embarcados. É importante porque fornece um conjunto de contribuições relativas à utilização dessas linguagens para resolver alguns problemas de pesquisa na especificação dos requisitos não-funcionais

Questão de pesquisa	Descrição e motivação
Q2. Quais os RNFs mais frequentes e seus respectivos domínios nas abordagens dos sistemas embarcados?	Esta questão fornece uma identificação e entendimento de quais RNFs são mais utilizados nos SE, bem como os seus respectivos domínios. Por exemplo, é possível identificar que um determinado domínio possui um RNF mais frequente.
Q3. Quais os problemas em aberto mais frequentes na especificação de RNFs dos sistemas embarcados?	O objetivo desta pergunta é identificar diferentes tipos de problemas em aberto referente a especificação de RNFs nos SE, bem como a identificação desses problemas relacionados seus domínios.

4.3 Visão geral

De acordo com a revisão sistemática da literatura da ER dos sistemas embarcados, são apresentadas nas próximas linhas uma visão geral e análise dos resultados referente aos 21 estudos identificados sobre os RNFs dos sistemas embarcados.

4.3.1 Tipo de contribuição

Com a seleção e classificação dos estudos da RSL, foi possível filtrar o tipo de contribuição dos 21 estudos referente aos RNFs. De acordo com PETERSEN et al. (2008), ao analisar os resultados com foco na apresentação das frequências das publicações para cada categoria. Isso possibilita entender quais categorias foram enfatizadas em pesquisas anteriores e, assim, identificar lacunas e possibilidades de pesquisa futura. Portanto, verificando a Tabela 4.3, nota-se que a maioria dos estudos o tipo de contribuição foi por meio de uma abordagem (11 trabalhos, 52,38%). Por exemplo o estudo S43, que cita o desenvolvimento de sistemas de aviação como um processo pesado e tedioso. Além das funções necessárias, os desenvolvedores devem considerar diversos e muitas vezes requisitos não-funcionais conflitantes, como *safety*, performance e eficiência energética. Esse estudo apresenta uma abordagem orientada a aspectos suportada por uma cadeia de ferramentas que lida com requisitos não-funcionais de forma integrada. Também discute como a abordagem pode ser aplicada ao desenvolvimento de sistemas críticos de segurança e fornece resultados experimentais.

Na sequência dos resultados da Tabela 4.3, verifica-se o tipo de contribuição linguagem/método presente em 23,81% dos estudos (05 trabalhos). Como exemplo, o estudo S03 apresenta uma metodologia AORE (*Aspect-Oriented Requirements Engineering*) que permite modelar as colaborações entre os sistemas distribuídos de software embarcados automotivos em termos de aspectos e gerar várias linhas de produtos, assegurando um conjunto de requisitos não-funcionais, incluindo *safety*, performance e custo. Já os tipo de contribuição *framework* esteve presente em 03 estudos (14,28%), como exemplo, o estudo S49 cita que experiências em sistemas críticos de segurança, que mostraram que as falhas introduzidas

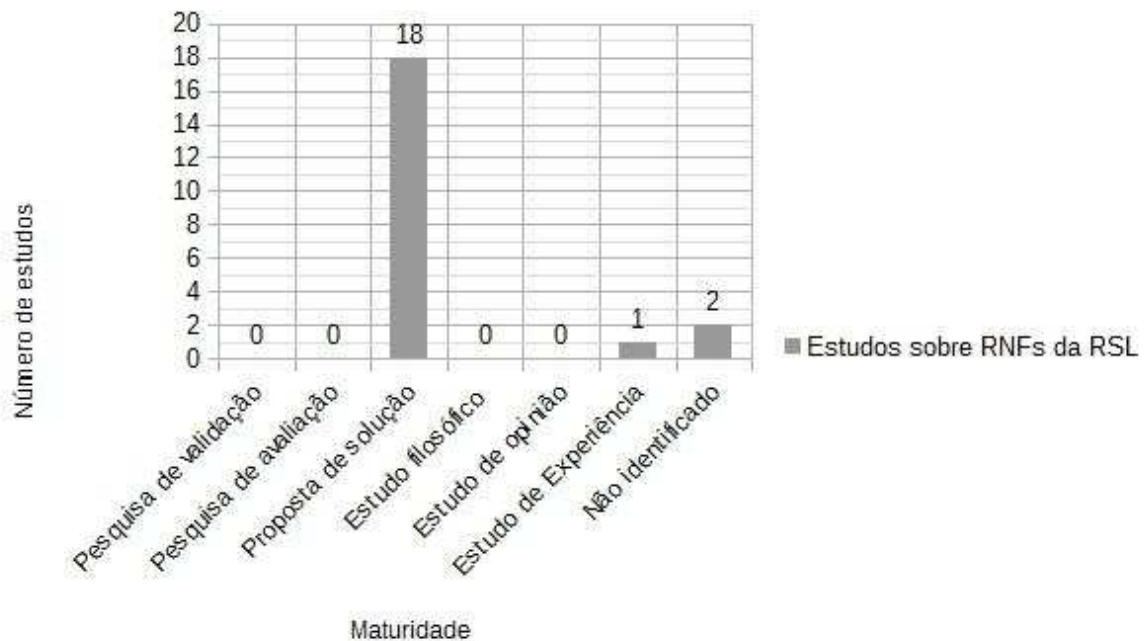
durante a análise de requisitos podem e causam acidentes. Portanto, uma metodologia é proposta para um desenvolvimento sistemático de especificações de requisitos de segurança com base em um *framework* para estruturar a análise e a aplicação de técnicas formais, que tem como objetivo de localizar e eliminar falhas durante a fase de especificação dos requisitos, evitando que seja feito durante o desenvolvimento ou na fase operacional do sistema. Por último, aparece o tipo de contribuição por meio de um processo, que esteve presente em 02 trabalhos (9,52%).

Tabela 4.3 Tipo de contribuição dos estudos sobre RNFs dos SE

Contribuição	Identificação dos estudos	Quantidade	%
Abordagem	S01, S24, S25, S34, S37, S40, S43, S45, S57, S60, S64	11	52,38
Linguagem/Método	S03, S13, S20, S27, S35	05	23,81
<i>Framework</i>	S48, S49, S61	03	14,28
Processo	S41, S65	02	9,52

4.3.2 Tipo da pesquisa

Conforme a Figura 4.1, o levantamento dos resultados referente aos RNFs da revisão, mostra a proposta de solução como destaque (86% dos estudos). Esse resultado mostra que as abordagens direcionam seus estudos com propostas de solução a problemas identificação e classificação dos RNFs no processo de ER. Segundo WIERINGA et al. (2006), essas propostas podem ser uma nova solução para o problema ou uma extensão significativa de uma técnica existente. Na sequência, 02 estudos não deixaram claros qual a maturidade da pesquisa e apenas 01 abordagem foi classificada como um estudo de experiência.

Figura 4.1 Tipo dos estudos selecionados

Fonte: Autoria própria

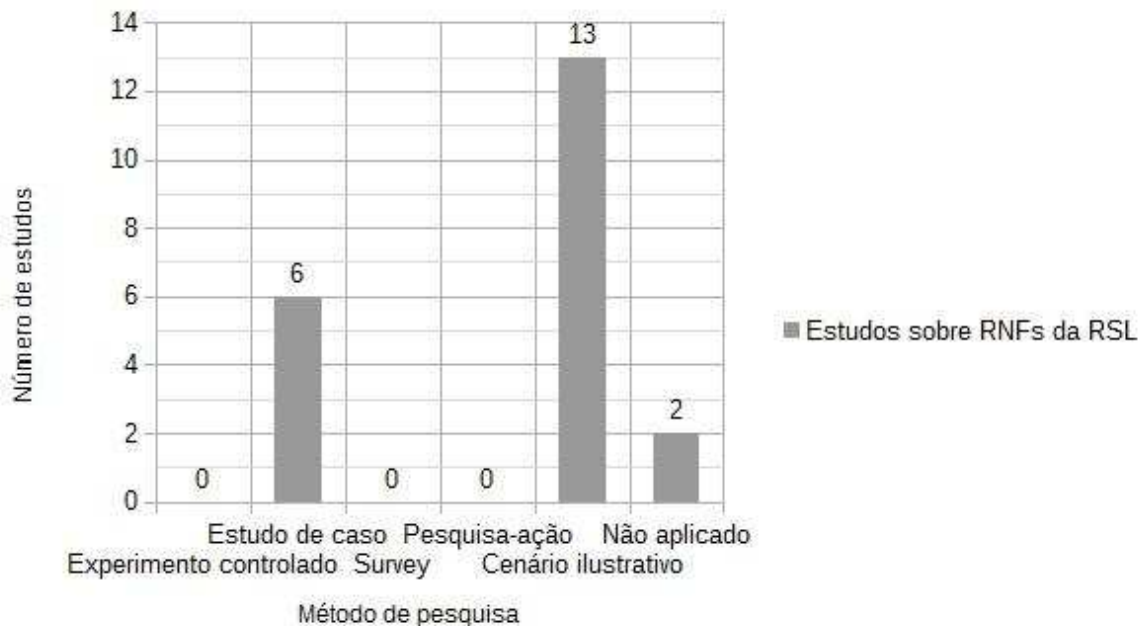
4.3.3 Método de pesquisa

Em conformidade com a Figura 4.2, os métodos aplicados nas abordagens selecionadas sobre os RNFs, destaca-se o cenário ilustrativo, que possui a maioria dos estudos (13 trabalhos, 61,90%). Esse resultado era esperado, tendo em vista que, os trabalhos utilizam pequenos cenários para exemplificar suas abordagens e a maioria dos 21 estudos identificados são desenvolvidos na academia, como exemplo, o trabalho S13, que descreve um pequeno exemplo de um sistema de porta de trem para ilustrar os conceitos sobre verificação das propriedades de segurança de um sistema de tempo real. Desse modo, foi apresentado uma adaptação do uso do Statechart baseado em uma nova semântica operacional. A abordagem é motivada pela problemática dos RNFs segurança e custo, pois geralmente são subjetivos e conflitantes entre si, assim, lidar com esses problemas requer um grande conjunto de conhecimentos sobre esses requisitos, porém existem poucos padrões para adquirir esse conhecimento. A solução proposta, foi criar quatro tipos de padrões de requisitos não-funcionais para capturar e reutilizar os conhecimentos desses requisitos.

Já o estudo de caso, foi o segundo método mais adotado pelas abordagens (06 trabalhos, 28%). Segundo YIN (2015), esse método é utilizado quando o fenômeno a ser estudado é amplo, complexo e não pode ser estudado fora do contexto onde ocorre naturalmente. Esse resultado evidencia a complexidade do requisitos não-funcionais dos SE, como exemplo o estudo S24, reporta uma experiência prática com um estudo de caso da vida real da indústria aeronáutica, são discutidos os problemas práticos encontrados, como a identificação de definições

na técnica de caso de uso que não são adequadas para especificar os RNFs dos SE.

Figura 4.2 Método de pesquisa dos estudos sobre RNFs da RSL.

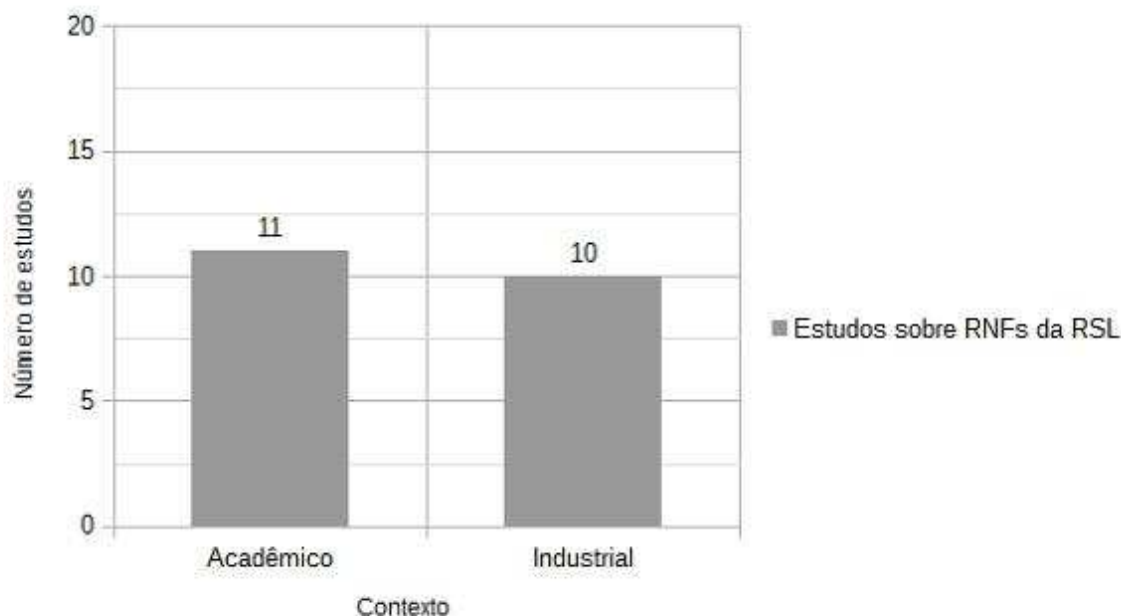


Fonte: Autoria própria

4.3.4 Contexto de aplicação

Também identificamos o contexto de aplicação dos 21 estudos. A Figura 4.3, apresenta a classificação e aponta um resultado interessante. Os estudos estão em categorias distintas de forma uniforme, ou seja, o contexto acadêmico teve 11 estudos (52,38%) e o industrial 10 (47,62%). Portanto, podemos concluir que a comunidade industrial também busca esforços para investigação na área de especificação de requisitos não-funcionais dos SE. Como exemplo, o estudo S24 descreve os resultados de um projeto para a Agência Espacial Européia com objetivo de produzir diretrizes para o desenvolvimento de software de tempo real para aplicações de naves espaciais. Por exemplo, o projeto desse estudo produziu orientações para requisitos por meio de um sistema de anotações concebido para capturar o desempenho e outros requisitos não-funcionais, de modo que as ferramentas de suporte ao método existente não são afetadas pela presença das anotações e que podem ser processadas por ferramentas adicionais.

Figura 4.3 Contexto de aplicação dos estudos.



RSL	Acadêmico	Industrial
NFRs SE	S01, S13, S20, S25, S27, S34, S35, S37, S40, S43, S45	S03, S24, S41, S48, S49, S57, S60, S61, S63, S64

Fonte: Autoria própria

4.4 Q1 - Quais as linguagens usadas para a especificação dos RNFs nos sistemas embarcados?

De acordo com a Tabela 4.4, as principais linguagens adotadas nos estudos referentes aos RNFs foram UML e suas adaptações com participação em 03 estudos (14,28%) dos 21 trabalhos selecionados e as linguagens naturais com mesmo percentual, em seguida, a linguagem SysML e StateChart apareceram em 02 estudos (9,52%). Diversas outras linguagens foram também citadas em um único estudo, a linguagem PAISLey em S25, SPEcTRM-RL (*Stands for Requirements Engineering Language*) em S34, RDAL (*Requirements Definition and Analysis Language*) em S43, TUM (*Timed Usage Models*) em S48, CFRL (*Causal Function Representative Language*) em S1, THL (*Time History Logic*) em S20, ActivityCharts em S23 e linguagem Ada 95. Nota-se um grande uso da linguagem UML e suas adaptações e das linguagens naturais na especificação de RNFs dos sistemas embarcados. Como exemplo de uma adaptação da UML, o estudo S63 utiliza uma linguagem de modelagem para aplicações de tempo real e embarcadas, OMTE (*Modeling and Analysis of Real Time and Embedded systems*), ela define um modelo de tempo amplamente expressivo que incorpora uma inter-

pretação cronometrada de modelos UML. Essa linguagem inclui a noção de tempos físicos e lógicos, tendo em vista os requisitos de tempo, por exemplo, sistemas aplicados ao setor de transporte, trens, carros e aeronaves controlados por computadores.

Referente as linguagens naturais que estão entre as mais abordadas, o estudo S27 utiliza uma linguagem natural para investigar o requisito não-funcional *safety* por meio de uma árvore de falhas, nos sistemas embarcados críticos de segurança, ou seja, como esses sistemas lidam com os riscos identificados em uma análise de segurança.

Um resultado não esperado foi a ausência do *NFR-Framework*, tendo em vista que ela é uma linguagem específica para representação e análise de requisitos não-funcionais sendo bastante utilizada no desenvolvimento de sistemas de propósito geral.

A maioria das abordagens contribuem com propostas de soluções por meio de linguagens que foram propostas para outros usos e que foram adaptadas para o caso de RNFs de SE. Portanto, fica evidente a falta de uma linguagem padrão para a especificação dos requisitos não-funcionais dos SE.

Tabela 4.4 Linguagens dos estudos sobre RNFs

Linguagem	Estudos	Quantidade	%
UML	S24; S27; S63	03	14,28
Linguagem Natural	S27; S35; S37	03	14,28
SysML	S27; S63	02	9,52
StateChart	S03; S13	02	9,52
PAISLey	S25	01	4,76
SPEcTRM-RL	S34	01	4,76
RDAL	S43	01	4,76
TUM	S48	01	4,76
CFRL	S01	01	4,76
THL	S20	01	4,76
ActivityChartsem	S23	01	4,76
Ada 95	S24	01	4,76
Não Especificado	S60; S61; S62; S45	04	19,05

4.5 Q2. Quais os RNFs mais frequentes e seus respectivos domínios nas abordagens dos sistemas embarcados?

Após extração dos dados selecionados da revisão sistemática da literatura também foi possível identificar os requisitos não-funcionais e seus respectivos domínios mais abordados dentre os 21 estudos selecionados. De acordo com a Tabela 4.5, os requisitos performance e verificabilidade foram abordados em 05 estudos cada (23,81%), na sequência, o requisito

4.5. Q2. QUAIS OS RNFS MAIS FREQUENTES E SEUS RESPECTIVOS DOMÍNIOS NAS ABORDAGENS DOS SISTEMAS EMBARCADOS? 72

não-funcional usabilidade está presente em 03 estudos (14,28%). Já os requisitos *safety*, confiabilidade e interoperabilidade, estão presentes em 02 estudos (9,52%) cada e outros 02 estudos abordavam sobre RNFs, mas não especificaram um requisito.

Tabela 4.5 Principais RNFs das abordagens

Requisitos não-funcionais	Quantidade	%
Performance	05	23,81
Verificabilidade	05	23,81
Usabilidade	03	14,28
<i>Safety</i>	02	9,52
Confiabilidade	02	9,52
Interoperabilidade	02	9,52
Não Especificado	02	9,52

De acordo com os resultados já expostos na Tabela 4.5, que identifica os RNFs (performance, verificabilidade, usabilidade, *safety*, confiabilidade e interoperabilidade) abordados nos estudos, foi possível destrinchar de forma mais detalhada esses RNFs na forma de baixa emissão de gás, requisitos de tempo, custo de desenvolvimento, economia de combustível, tempo real, robustez, energia total, consumo de tempo, peso, sobrevivência, consumo de energia e interface de usuário.

Já a Tabela 4.6, mostra os domínios abordados nos 21 estudos. Nota-se que os domínios automação tecnológica e automotivo estão presentes em 04 estudos cada (19,05%), seguidos pelo domínio de tecnologia médica com 03 estudos (14,28%). Na sequência aparece o domínio de tecnologia aviônica com 02 estudos (9,52%) e o domínio menos encontrado nas abordagens foi o de tecnologia energética com 01 (4,76%) estudo. Os demais estudos (07, 33,33%) não deixaram claros quais domínios eram referenciados na pesquisa e outros abordaram mais de um domínio.

4.5. Q2. QUAIS OS RNFS MAIS FREQUENTES E SEUS RESPECTIVOS DOMÍNIOS NAS ABORDAGENS DOS SISTEMAS EMBARCADOS? 73

Tabela 4.6 Domínios das abordagens sobre RNFs dos SE

Domínios	Quantidade	%
Automação Tecnológica	04	19,05
Automotivo	04	19,05
Tecnologia Médica	03	14,28
Tecnologia Aviônica	02	9,52
Tecnologia Energética	01	4,76
Não Especificado	07	33,33

A Figura 4.4, apresenta um dimensionamento entre os requisitos não-funcionais encontrados nas fases de especificação e os domínios em que esses requisitos foram aplicados.

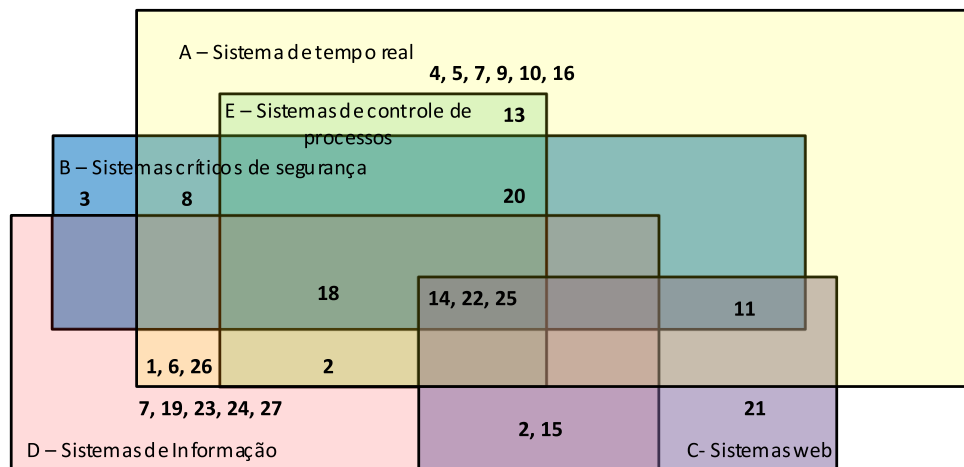
Percebe-se que os domínios foram abordados de maneira uniforme, com destaque ao domínio automotivo e automação tecnológica com 04 abordagens cada (21,01%) e seus estudos relacionaram os RNFs uniformemente com 01 estudo para performance, *safety*, confiabilidade, interoperabilidade e usabilidade. O resultado evidencia que esses domínios abordam a maioria dos RNFs encontrados.

Figura 4.4 RNFs e domínios dos SE.



De acordo com a Figura 4.4, é possível notar por meio do preenchimento das bolhas referente ao requisito não-funcional performance, que esse requisito contempla todos os domínios levantados, com exceção da tecnologia de energia. Portanto, a performance está sendo explorada nos diversos domínios pela comunidade dos embarcados e esses estudos apontam o tempo de resposta e desempenho, como fundamentais para o bom funcionamento desses sistemas. O trabalho S4 apresenta os resultados de uma extensa análise sistemática da literatura que classifica cinco principais RNFs frequentemente considerados. Os autores resumem os resultados na Figura 4.5, por meio de um dimensionamento entre os tipos de sistemas e os requisitos não-funcionais mais relevantes os quais foram encontrados na pesquisa. Ao verificar os resultados da Figura 4.4 e os da Figura 4.5, percebe-se que os RNFs performance, *safety* e usabilidade estão presentes na maioria dos domínios ou tipos de sistemas. Ainda na análise, podemos concluir que, os RNFs verificabilidade e performance referentes (nº 14 e 26) aos sistemas críticos de segurança e sistemas de tempo real da Figura 4.5, também estão presentes no domínio de tecnologia médica. Esse resultado era esperado, pois muitos sistemas da tecnologia médica possuem como características de desempenho de tempo real e podem também ser classificados como um sistema crítico de segurança. O estudo S25 da revisão de ER dos sistemas embarcados e classificado como um estudo especificação de RNFs no domínio de tecnologia médica, apresenta uma linguagem denominada PAISLey. A pesquisa também define que os SE são caracterizados por exigências de desempenho urgentes (performance), desse modo a linguagem propõe um tratamento formal, mas com características intuitivas desse requisito não-funcional. Os autores também definem que a melhor maneira de especificar e analisar os requisitos nos embarcados é iniciando com um modelo de ambiente e trabalhar para que o modelo suporte uma operação desejada pelo usuário. Portanto, fica claro o esforço da comunidade dos embarcados da tentativa de achar um padrão formal para resolver os problemas da subjetividade, dos conflitos de requisitos e da descrição trabalhosa para especificação dos requisitos não-funcionais nos diversos domínios, ou seja, esses problemas são comuns independente do domínio aplicado.

Figura 4.5 Tipos de sistemas e RNFs relevantes



Legenda:

1 - Precisão	10 - Instabilidade	19 - Reusabilidade
2 - Disponibilidade	11 - Integridade	20 - <i>Safety</i>
3 - Comunicação	12 - Interoperabilidade	21 - Escalabilidade
4 - Compatibilidade	13 - Manutenibilidade	22 - <i>Security</i>
5 - Integridade	14 - Performance	23 - Padronização
6 - Confidencialidade	15 - Privacidade	24 - Rastreabilidade
7 - Conformidade	16 - Portabilidade	25 - Usabilidade
8 - Dependabilidade	17 - Provabilidade	26 - Verificabilidade
9 - Extensibilidade	18 - Confiabilidade	27 - Viabilidade

Fonte: Traduzido de MAIRIZA; ZOWGHI; NURMULIANI (2010)

4.6 Q3. Quais os problemas em aberto mais frequentes encontrados na especificação de RNFs dos sistemas embarcados?

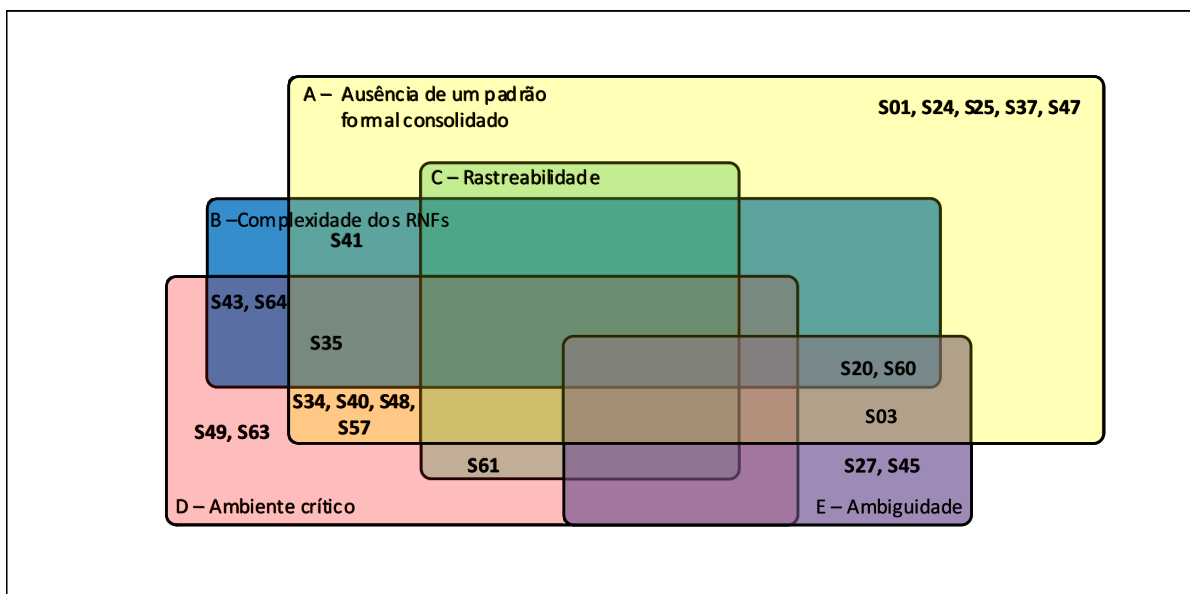
Conforme visto no Capítulo 3, o problema em aberto mais citado foi a especificação dos RNFs. Nesta nova análise identificamos quais os problemas que tornam a especificação tão difícil. A Figura 4.6 dimensiona e relaciona os estudos aos problemas encontrados. Com destaque para o problema A (Ausência de um padrão formal consolidado), que foi citado em 14 (66,67%) estudos. Como exemplo, o estudo S03, em que os autores TANURHAN; SCHMERLER; MULLER-GLASER (1996) definem que é fundamental para o desenvolvimento

de software embarcado automotivo a classificação de múltiplos requisitos não-funcionais de *safety*, custo e baixas possibilidades de conflitos. Várias abordagens têm sido sugeridas para essa classificação, todavia as abordagens convencionais são limitadas no âmbito de certos aspectos de um sistema, ou seja, precisa-se de um modelo abrangente para lidar com vários requisitos não-funcionais em todo sistema embarcado.

Continuando com a análise da Figura 4.6, nota-se que os autores SAEED; LEMOS; ANDERSON (1991) do estudo S20, citam como problema na especificação de RNFs nos sistemas embarcados a complexidade dos RNFs (Problema B), pois esses requisitos muitas vezes são incompletos, mal definidos e mudam constantemente causando conflitos com outros requisitos. Por exemplo, entre os requisitos não-funcionais mais abordados na RSL estão performance, usabilidade e confiabilidade, mas esses requisitos possuem interdependências e conflitos entre si. A usabilidade por exemplo, propõe que os usuários realizem suas tarefas de forma prática e rápida, ou seja, sem dúvidas e sem erros causados pela dificuldade de usar. No entanto ao inserir medidas de segurança no sistema para prover autenticação, autorização e confidencialidade, a usabilidade será modificada negativamente, pois mais passos serão adicionados para fazer as mesmas ações.

Em relação a ambiguidade (problema E), o trabalho S20 de SAEED; LEMOS; ANDERSON (1991) cita os problemas dos RNFs serem mal entendidos e classificados de forma equivocada. Portanto, as situações em que os requisitos não-funcionais serão abordados no sistema, as características comportamentais desses requisitos e as ações do sistema para atender ao requisito, todos devem estar claros e bem descritos, assim torna mais fácil o gerenciamento de falhas e compreensão correta dos atributos a serem atendidos.

O problema D (Ambiente crítico), também se destacou com um número representativo de 10 (47,62%) estudos. A maioria desses estudos definem como problema, o ambiente crítico que a maioria dos sistemas embarcados são aplicados. Um exemplo, o estudo S63 cita que é fundamental que os requisitos de tempo e precisão sejam especificados e considerados nos sistemas embarcados. Por exemplo, os sistemas de sinalização para trens de alta velocidade, os requisitos de tempo e precisão são especificados para evitar colisão de trens, ou seja, esse ambiente e esses requisitos são críticos, pois as falhas podem causarem grandes perdas materiais e acidentes graves envolvendo pessoas. Tendo em vista os ambientes críticos dos sistemas embarcados e a importância dos requisitos não-funcionais de desempenho e segurança, este estudo propõe uma linguagem chamada PAISLey que oferece uma especificação formal e mais intuitiva desses requisitos, pois muitas vezes eles possuem interdependências que podem favorecer a atuação do outro requisito ou podem causar problemas. Por exemplo, a segurança possui interdependências potencializadas nos SE devido ao seu ambiente crítico, ou seja, uma vez que aumenta-se a segurança de acesso ao sistema, poderá influenciar negativamente o seu desempenho, tornando a execução de serviços mais lenta, por exemplo, o caso do sistemas de sinalização de trens, essa medida poderia causar um grave acidente.

Figura 4.6 Problemas comuns encontrados na especificação de RNFs nos SE.

Fonte: Autoria própria

De acordo com a Tabela 4.7, o problema mais frequente encontrado nos estudos foi a ausência de um padrão formal consolidado (A) com 14 estudos (66,67%). O problema D (Ambiente crítico) vem logo em seguida com presença de 10 estudos (47,62%). Já o problema B (Complexidade dos RNFs) aparece em 06 estudos (28,57%). Na sequência dos problemas mais abordados, o problema E (Ambiguidade) esteve presente em 05 estudos (23,81%) e por último, o problema C (Rastreabilidade) foi abordado em apenas 01 estudo (4,76%).

Tabela 4.7 Problemas em aberto dos estudos sobre RNFs

Problemas	Estudos	Quantidade	%
A	S01, S03, S20, S24, S25, S34, S35, S37, S40, S41, S47, S48, S57, S60	14	66,70
B	S20, S35, S41, S43, S60, S64	06	28,57
C	S61	01	4,76
D	S34, S35, S40, S43, S48, S49, S57, S61, S63, S64	10	47,62
E	S03, S20, S27, S45, S60	05	23,81

4.7 Considerações Finais

Este capítulo apresentou uma análise mais detalhada de como os RNFs tem sido tratado na Engenharia de Requisitos dos Sistemas Embarcados. Como fonte de informação consideramos os trabalhos que identificaram este problema como um desafio em aberto, bem

como os artigos que trataram da especificação de requisitos não-funcionais no contexto de SE. No total foram considerados 21 estudos.

Os dados extraídos desta análise contribuem fornecendo informações como: (1) visão geral no âmbito dos RNFs dos sistemas embarcados por meio das informações como, tipo de contribuição (Abordagem, Linguagem/método, *Framework* e Processo), maturidade da pesquisa (Pesquisa de validação, Pesquisa de avaliação, Proposta de Solução, Estudo filosófico, Estudo de opinião e Estudo de experiência), método de pesquisa (Experimento controlado, Estudo de caso, *Survey*, Pesquisa-ação e Cenário ilustrativo), contexto de aplicação (Acadêmico e Industrial); (2) com a identificação das linguagens/métodos mais utilizados para soluções de problemas referente aos RNFs dos sistemas embarcados; (3) identificação dos RNFs mais frequentes e seus respectivos domínios nos sistemas embarcados; (4) e a identificação dos problemas em aberto mais frequentes relacionados a especificação dos RNFs dos sistemas embarcados.

5

CONCLUSÃO

A engenharia de requisitos dos sistemas embarcados apresenta constantes estudos a fim de buscar soluções para desafios crescentes devido ao grande uso e dependência desses sistemas pela sociedade atual. O Capítulo 3 apresentou por meio de uma RSL o estado da arte relacionado a ER nos sistemas embarcados. Foram apresentados uma visão geral dos estudos, as contribuições, fases envolvidas no processo de ER, os domínios de aplicação, os requisitos envolvidos dos SE, os problemas em aberto e os trabalhos futuros. A revisão também auxiliou nas respostas das questões de pesquisa que ajudarão a comunidade dos embarcados a direcionar seus esforços nos problemas em aberto específicos da engenharia de requisitos.

Nesse panorama, foi relacionada uma grande diversidade de problemas em aberto, no entanto a especificação dos requisitos não-funcionais foi considerado o problema a ser investigado, dessa forma, no capítulo 4 foi realizada uma análise desse problema e dos RNFs abordados como forma de estender e complementar os resultados da revisão sistemática da literatura. Portanto, com a análise foi possível obter resultados complementares como, visão geral, levantamento de informações das linguagens/métodos mais utilizadas, os requisitos com seus respectivos domínios e identificação/classificação dos problemas em aberto referente a especificação dos requisitos não-funcionais dos sistemas embarcados.

O restante deste trabalho apresenta as contribuições alcançadas, limitações do trabalho e trabalhos futuros.

5.1 Contribuições

Após a conclusão desse trabalho, algumas contribuições foram descritas a seguir:

- **Revisão Sistemática da Literatura.** A RSL foi realizada para identificar o estado da arte na engenharia de requisitos nos sistemas embarcados. Foram levantados dados de aproximadamente 4 décadas de pesquisas na literatura, os quais foram analisados, sintetizados e extraídos como suporte para responder a perguntas de

pesquisa sobre as abordagens, métodos, técnicas, linguagens, processos, problemas em aberto e trabalhos futuros para apoiar a ER no domínio dos sistemas embarcados. No final da análise foram incluídos 75 estudos relevantes para ER dos sistemas embarcados, 12 contribuições de ER, 05 domínios de sistemas embarcados, 15 problemas em aberto, os métodos de pesquisas empregados e o contexto de aplicação de cada estudo.

- **Análise.** Uma análise mais detalhada de como os RNFs tem sido tratados nos Sistemas Embarcados. Os resultados apresentados podem contribuir para a comunidade do SE da seguinte maneira: (1) melhorar o entendimento da importância de especificar os RNFs nos sistemas embarcados, tendo em vista o ambiente e a própria criticidade dos seus requisitos; (2) motivar a comunidade de engenharia de requisitos nos SE a alcançar um consenso no desenvolvimento de um padrão para especificação dos RNFs e que esse padrão possa estar inserido em um outro processo específico de ER nos sistemas embarcados; (3) os RNFs apresentados nessa análise (performance, usabilidade, *safety*, confiabilidade, verificabilidade, interoperabilidade, confidencialidade, manutenibilidade, eficiência) devem informar e motivar a comunidade de pesquisa a realizar estudos aprofundados sobre esses requisitos, tendo em vista que eles também foram analisados e relacionados a seus respectivos domínios, bem como os problemas em aberto mais comuns; (4) identificação de quais os problemas na especificação dos RNFs nos embarcados são mais frequentes; (5) Os desenvolvedores serão capazes de ter uma noção prévia durante a especificação de requisitos sobre quais os RNFs devem obter atenção no projeto em que estão a trabalhar conforme o tipo de sistema ou domínio; (6) Dados e informações como linguagens/métodos utilizados, bem como seus domínios e os problemas em aberto mais comuns relacionados aos RNFs dos SE foram sintetizados e podem ser utilizadas como uma agenda de pesquisa.

5.2 Limitações do Trabalho

Relacionado a revisão sistemática da literatura da ER dos Sistemas Embarcados é possível destacar ameaças à validade como, o método de busca (*string*) que muitas vezes não contempla todos os estudos da área, outra limitação desse trabalho foi não buscar pesquisas escritas em português, devido a maioria dos estudos da área serem escritos em inglês e o grande quantitativo de estudos já encontrados (12.732), o que tornaria inviável a seleção e análise de mais estudos em curto período de tempo. Destaca-se também a limitação no levantamento dos requisitos dos SE, tendo em vista que foram somente identificados e classificados, faltando um detalhamento geral no contexto desses requisitos.

Em relação à análise apresentada no Capítulo 4 foi limitada, já que foram considerados

apenas os estudos que haviam identificado RNFs como um problema em aberto e os trabalhos que tratam da especificação dos RNFs. Portanto, a análise não contemplou outras fases do processo de ER.

5.3 Trabalhos Futuros

A revisão sistemática da literatura deve ser continuada, algumas questões de pesquisa podem contribuir para a ER dos embarcados, como exemplo, a questão futura Q7 - Qual o tipo (baseada em cenários, orientado a objeto, etc) de modelagem de requisitos de *software* foi apoiada pelas abordagens? Esse questionamento poderá fornecer a identificação dos principais tipos de modelagem de requisitos de software (com base em cenários, requisitos textuais orientado a objeto e assim por diante) que foram suportados pelas abordagens, poderá identificar quais os tipos de requisitos estão atraindo mais atenção da comunidade de sistemas embarcados, os pontos fortes e fracos de cada tipo. Outra questão futura como, Q8 - As pesquisas de ER dos sistemas embarcados seguem uma linha de proposta e resultados conduzidos pela indústria ou pela academia? Por meio desse questionamento podemos investigar se os resultados das pesquisas ou contribuições estão sendo transferidos da academia para indústria ou vice-versa. A RSL dispõe de dados para uma investigação detalhada para que possamos analisar o processo de engenharia de requisitos dos sistemas embarcados e dos sistemas de propósito geral, dessa forma obter dados e resultados para a construção de um processo padrão para os SE.

Ainda como trabalho futuro, pretende-se investigar de forma detalhada os problemas, as definições e atributos dos RNFs de acordo com seus respectivos domínios, pois estes foram identificados apenas no âmbito geral. Por exemplo, o requisito não-funcional performance foi um dos mais encontrados nos estudos selecionados e que possui atributos como, tempo de resposta, latência, espaço, velocidade de execução, uso de memória, entre outros. No entanto, esse trabalho de identificação e relação ao tipo de domínio aplicado nos SE, bem como os problemas e especificidades desses atributos não foram encontrados na revisão da literatura realizada neste trabalho. Ainda como trabalho futuro, pretendemos investigar os principais requisitos não-funcionais que causam conflitos nos diversos domínios dos SE, elaborar um catálogo com RNFs específicos para sistemas embarcados, criar um mecanismo de críticas que explore os aspectos dos SE e verifique se são de natureza crítica e conflitante com os requisitos não-funcionais, como exemplo, o aumento da segurança na execução de uma atividade poderá impactar no requisito não-funcional usabilidade, consequentemente causar algum dano crítico dependendo da natureza ou domínio do sistema embarcado.

Uma outra linha de pesquisa deverá abordar especificamente a influência que os requisitos não-funcionais exercem sobre o desenvolvimento e arquitetura dos sistemas embarcados.

Por fim, uma linha de pesquisa que investigue as diferenças na perspectiva de como a comunidade de software considera a noção do RNFs nos domínios dos sistemas embarcados e de sistemas de propósito geral.

REFERÊNCIAS

- ACHIMUGU, P.; SELAMAT, A.; IBRAHIM, R.; MAHRIN, M. N. A systematic literature review of software requirements prioritization research. **Information and Software Technology**, [S.l.], v.56, n.6, p.568 – 585, 2014.
- ALI, R.; DALPIAZ, F.; GIORGINI, P. A goal-based framework for contextual requirements modeling and analysis. **Requir. Eng.**, [S.l.], v.15, n.4, p.439 – 458, 2010.
- BITTENCOURT, I. I.; BARANAUSKAS, M. C.; PEREIRA, R.; DERMEVAL, D.; ISOTANI, S.; JAQUES, P. A systematic review on multi-device inclusive environments. **Universal Access in the Information Society**, [S.l.], p.1 – 36, 2015.
- BLOUIN, D.; SENN, E.; TURKI, S. Defining an annex language to the architecture analysis and design language for requirements engineering activities support. In: MODEL-DRIVEN REQUIREMENTS ENGINEERING WORKSHOP (MODRE), 2011. **Anais...** [S.l.: s.n.], 2011. p.11 – 20.
- BOWEN, T. P.; WIGLE, G. B.; TSAI, J. T. **Specification of software quality attributes**. [S.l.]: Rome Air Development Center, Air Force Systems Command, 1985.
- BRANSTAD, M.; POWELL, P. B. Software engineering project standards. **IEEE transactions on software engineering**, [S.l.], n.1, p.73–78, 1984.
- BRAUN, P.; BROY, M.; HOUDEK, F.; KIRCHMAYR, M.; MÜLLER, M.; PENZENSTADLER, B.; POHL, K.; WEYER, T. Guiding requirements engineering for software-intensive embedded systems in the automotive industry. **Computer Science-Research and Development**, [S.l.], v.29, n.1, p.21 – 43, 2014.
- CHUNG, L.; CESAR, J.; LEITE, S. P. Non-functional requirements in software engineering. , [S.l.], 1999.
- CHUNG, L.; NIXON, B. A.; YU, E.; MYLOPOULOS, J. The NFR framework in action. In: **Non-Functional Requirements in software engineering**. [S.l.]: Springer, 2000. p.15 – 45.
- CHUNG, L.; NIXON, B. A.; YU, E.; MYLOPOULOS, J. **Non-functional requirements in software engineering**. [S.l.]: Springer Science & Business Media, 2012. v.5.
- COMMITTEE, I. C. S. S. E. S.; BOARD, I. S. IEEE Recommended Practice for Software Requirements Specifications. In: **Anais...** [S.l.: s.n.], 1998.
- COMMITTEE, I. C. S. S. E. S.; BOARD, I.-S. S. IEEE recommended practice for software requirements specifications. In: **Anais...** [S.l.: s.n.], 1998.
- DAVIS, A. M. A comparison of techniques for the specification of external system behavior. **Communications of the ACM**, [S.l.], v.31, n.9, p.1098–1115, 1988.
- DERMEVAL, D.; VILELA, J.; BITTENCOURT, I. I.; CASTRO, J.; ISOTANI, S.; BRITO, P.; SILVA, A. Applications of ontologies in requirements engineering: a systematic review of the literature. **Requirements Engineering**, [S.l.], p.1 – 33, 2015.

- DING, W.; LIANG, P.; TANG, A.; VLIET, H. V. Knowledge-based approaches in software documentation: a systematic literature review. **Information and Software Technology**, [S.l.], v.56, n.6, p.545 – 567, 2014.
- DYBAA, T.; DINGSOYR, T. Empirical studies of agile software development: a systematic review. **Information and software technology**, [S.l.], v.50, n.9, p.833 – 859, 2008.
- EASTERBROOK, S.; SINGER, J.; STOREY, M.; DAMIAN, D. Selecting empirical methods for software engineering research. In: **Guide to advanced empirical software engineering**. [S.l.]: Springer, 2008. p.285 – 311.
- ESTRIN, D.; BORRIELLO, G.; COLWELL, R.; FIDDLER, J.; HOROWITZ, M.; KAISER, W.; LEVESON, N.; LISKOV, B.; LUCAS, P.; MAHER, D. et al. **A Research Agenda for Networked Systems of Embedded Computers, NAP Washinton, DC 2001**. [S.l.]: ISBN 0-309-07568-8, 2001.
- FERRARI, A. T. **Metodologia da ciência**. [S.l.]: Kennedy Editora, 1974.
- FREITAS, E. P.; WEHRMEISTER, M. A.; PEREIRA, C.; WAGNER, F.; JR, E. S.; CARVALHO, F. Using Aspects to Model Distributed Real-Time Embedded Systems. In: WORKSHOP ON ASPECT-ORIENTED SOFTWARE DEVELOPMENT, FLORIANOPOLIS, BRAZIL. **Proceedings...** [S.l.: s.n.], 2006.
- FRIEDRICH, L. A Survey of Operating Systems Infrastructure for Embedded Systems. , [S.l.], 2009.
- GALSTER, M.; WEYNS, D.; TOFAN, D.; MICHALIK, B.; AVGERIOU, P. Variability in software systems—a systematic literature review. **Software Engineering, IEEE Transactions on**, [S.l.], v.40, n.3, p.282 – 306, 2014.
- GOMES, L. **Behavioral Modeling for Embedded Systems and Technologies: applications for design and implementation: applications for design and implementation**. [S.l.]: IGI Global, 2009.
- KEELE, S. Guidelines for performing systematic literature reviews in software engineering. In: **Technical report, Ver. 2.3 EBSE Technical Report. EBSE**. [S.l.: s.n.], 2007.
- KITCHENHAM, B.; BRERETON, P.; BUDGEN, D.; TURNER, M.; BAILEY, J.; LINKMAN, S. G. Systematic literature reviews in software engineering - A systematic literature review. **Software Technology**, [S.l.], v.51, n.1, p.7 – 15, 2009.
- KRUCHTEN, P. The Rational Unified Process An Introduction Second Edition. Adison-Wesley. **Reading, Massachusetts**, [S.l.], 2000.
- KRUGER, I.; FARCAS, C.; FARCAS, E.; MENARINI, M. 7 Requirements Modeling for Embedded Realtime Systems. In: **Model-Based Engineering of Embedded Real-Time Systems**. [S.l.]: Springer, 2010. p.155 – 199.
- LAKATOS, E. M.; ANDRADE MARCONI, M. de. Fundamentos da metodologia científica. In: **Fundamentos da metodologia científica**. [S.l.]: Atlas, 2010.
- LAKATOS E.V. MARCONI, M. A. **Fundamentos de metodologia científica**. 6.ed. [S.l.]: Atlas, 2006.

LAPES. **Start of the art thought systematic review tool**. 2015.

LEE, E. A.; SESHIA, S. A. **Introduction to embedded systems: a cyber-physical systems approach**. [S.l.]: Lee & Seshia, 2011.

LIGGESMEYER, P.; TRAPP, M. Trends in embedded software engineering. **IEEE software**, [S.l.], v.26, n.3, 2009.

MARTINS, L. E. G.; OLIVEIRA, T. de. A case study using a protocol to derive safety functional requirements from fault tree analysis. In: REQUIREMENTS ENGINEERING CONFERENCE (RE), 2014 IEEE 22ND INTERNATIONAL. **Anais...** [S.l.: s.n.], 2014. p.412–419.

MARTINS, L. G.; OLIVEIRA, T. D. A case study using a protocol to derive safety functional requirements from Fault Tree Analysis. In: REQUIREMENTS ENGINEERING CONFERENCE (RE), 2014 IEEE 22ND INTERNATIONAL. **Anais...** [S.l.: s.n.], 2014. p.412 – 419.

NQA, A. Quality Assurance Requirements for Nuclear Facility Applications. **American Society of Mechanical Engineers**, [S.l.], 1.

OSSADA, J. **GERSE: guia de elicitação de requisitos para sistemas embarcados de pequeno e médio porte**. 2010. Tese (Doutorado em Ciência da Computação) — Dissertação de Mestrado do Programa de Pós-Graduação em Ciência da Computação—Universidade Metodista de Piracicaba-Piracicaba.

PEREIRA, T.; ALBUQUERQUE, D.; SOUSA, A.; ALENCAR, F.; CASTRO, J. Towards a Metamodel for a Requirements Engineering Process of Embedded Systems. In: COMPUTING SYSTEMS ENGINEERING (SBESC), 2016 VI BRAZILIAN SYMPOSIUM ON. **Anais...** [S.l.: s.n.], 2016. p.93–100.

PEREIRA, T. C. BVCCoN-Tool: uma ferramenta para apoiar uma abordagem de configuração de processos de negócio dinâmicos. , [S.l.], 2014.

PETERSEN, K.; FELDT, R.; MUJTABA, S.; MATTSSON, M. Systematic mapping studies in software engineering. In: INTERNATIONAL CONFERENCE ON EVALUATION AND ASSESSMENT IN SOFTWARE ENGINEERING, 12. **Anais...** [S.l.: s.n.], 2008. v.17, n.1.

PRESSMAN, R. S. **Engenharia de software**. [S.l.]: AMGH Editora, 2009.

RASHID, A.; SAWYER, P.; MOREIRA, A.; ARAÚJO, J. Early aspects: a model for aspect-oriented requirements engineering. In: REQUIREMENTS ENGINEERING, 2002. PROCEEDINGS. IEEE JOINT INTERNATIONAL CONFERENCE ON. **Anais...** [S.l.: s.n.], 2002. p.199 – 202.

RUMBAUGH, J.; BLAHA, M.; PREMERLANI, W.; EDDY, F.; LORENSEN, W. **Modelagem e projetos baseados em objetos**. [S.l.]: Campus, 1994. v.8.

RZEPKA, W. Guest Editors' Introduction Requirements Engineering Environments: software tools for modeling user needs. **Computer**, [S.l.], v.18, n.4, p.9–12, 1985.

SAEED, A.; LEMOS, R. de; ANDERSON, T. The role of formal methods in the requirements analysis of safety-critical systems: a train set example. In: FAULT-TOLERANT COMPUTING, 1991. FTCS-21. DIGEST OF PAPERS., TWENTY-FIRST INTERNATIONAL SYMPOSIUM. **Anais...** [S.l.: s.n.], 1991. p.478 – 485.

- SIKORA, E.; TENBERGEN, B.; POHL, K. Industry needs and research directions in requirements engineering for embedded systems. **Requirements Engineering**, [S.l.], v.17, n.1, p.57 – 78, 2012.
- SOMMERVILLE, I. Integrated requirements engineering: a tutorial. **IEEE software**, [S.l.], v.22, n.1, p.16 – 23, 2005.
- SOMMERVILLE, I. **Software Engineering**. 9th Edition.ed. [S.l.]: Addison-Wesley, 2011.
- SOMMERVILLE, I.; KOTONYA, G. Requirements engineering: processes and techniques. , [S.l.], 1998.
- SOUSA, A.; AGRA, C.; MELO, J.; ALENCAR, F. Elicitação e Especificação de Requisitos em Sistemas Embarcados: uma revisão sistemática. **Workshop em Engenharia de Requisitos: Lima, Peru**, [S.l.], 2015.
- TANURHAN, Y.; SCHMERLER, S.; MULLER-GLASER, K. Integrated design process with MESA/MERLAN. In: COMPUTER-AIDED CONTROL SYSTEM DESIGN, 1996., PROCEEDINGS OF THE 1996 IEEE INTERNATIONAL SYMPOSIUM ON. **Anais...** [S.l.: s.n.], 1996. p.333 – 338.
- THAYER, R. H.; BAILIN, S. C.; DORFMAN, M. **Software requirements engineerings**. [S.l.]: IEEE Computer Society Press, 1997.
- THAYER, R. H.; BAILIN, S. C.; DORFMAN, M. **Software requirements engineerings**. [S.l.]: IEEE Computer Society Press, 1997.
- THAYER, R. H.; DORFMAN, M. Software requirements engineering glossary. , [S.l.], 1997.
- TORRE-UGARTE, M. C. De-la; TAKAHASHI, R. F.; BERTOLOZZI, M. R. et al. Revisão Sistemática: noções gerais. **Revista da Escola de Enfermagem da USP**, [S.l.], v.45, n.5, p.1260–1266, 2011.
- WIERINGA, R.; MAIDEN, N.; MEAD, N.; ROLLAND, C. Requirements engineering paper classification and evaluation criteria: a proposal and a discussion. **Requirements Engineering**, [S.l.], v.11, n.1, p.102 – 107, 2006.
- YIN, R. K. **Estudo de Caso-**: planejamento e métodos. [S.l.]: Bookman editora, 2015.
- ZAVE, P. An operational approach to requirements specification for embedded systems. **Software Engineering, IEEE Transactions on**, [S.l.], n.3, p.250 – 269, 1982.
- ZAVE, P. Classification of research efforts in requirements engineering. **ACM Computing Surveys (CSUR)**, [S.l.], v.29, n.4, p.315 – 321, 1997.



GLOSSÁRIO DE TERMINOLOGIAS DE SISTEMAS EMBARCADOS

A.1 Escopo

Este glossário define os termos utilizados no domínio de engenharia de requisitos de sistemas e software. O conteúdo abordado neste trabalho possui termos específicos da engenharia de software, engenharia de requisitos e sistemas embarcados. A maioria dos conceitos foram retirados do glossário de engenharia de requisitos proposto por THAYER; DORFMAN (1997), o restante de artigos e livros.

A.2 Estrutura do Glossário

As terminologias deste glossário são organizadas em ordem alfabética. Um termo pode consistir de uma única palavra, como “Embarcado”, ou uma frase, como “Sistema Embarcado”, ou um acrônimo, como “SE”.

A.3 Terminologia da Engenharia de Requisitos dos Sistemas Embarcados

Adaptabilidade. Na análise de requisitos de software, uma taxa na qual a solução de software pode Adaptar a uma nova exigência.

Ambiente de engenharia de requisitos. Em engenharia de sistemas / software, fornece ao engenheiro de requisitos os mecanismos apropriados para facilitar a análise, documentação e atividades de verificação (RZEPKA, 1985).

Ambiguidade. Um problema da especificação de requisitos no desenvolvimento de software em que as declarações de requisitos podem ser interpretadas de diversas maneiras.

Análise. O processo de estudar um sistema dividindo em partes (funções ou objetos) e determinando como essas partes se relacionam entre si para entender o todo.

Caso de uso. Na engenharia de software (requisitos), um método de descrever as possíveis sequências de interações entre o sistema e um ou mais atores em resposta a algum estímulo inicial por um dos atores. É um conjunto de cenários possíveis, cada um começando com algum evento inicial de um ator para o sistema e seguindo a transação resultante para sua conclusão lógica (RUMBAUGH et al., 1994).

Classe de domínio. Classes de um domínio de problema que são significativas fora de qualquer aplicativo.

Cliente. (1) A engenharia de sistemas / sistemas de software, um indivíduo ou organização que especifica os requisitos, aceita formalmente a entrega de um novo produto ou hardware / software modificado e sua documentação. O cliente pode ser interno ou externo à organização-mãe do projeto e não implica necessariamente uma transação financeira entre cliente e desenvolvedor. (2) A pessoa ou pessoas que pagam pelo projeto e geralmente (mas não necessariamente) decidem os requisitos; O cliente pode não ser o usuário.

Compatibilidade. Na engenharia de software, uma métrica de qualidade que pode ser usada para medir a capacidade de dois ou mais sistemas trocar informações.

Complexidade. Na engenharia de software, uma métrica de qualidade que pode ser usada para medir o grau de complicação de um sistema ou componente do sistema.

Confiabilidade. Na engenharia de software, uma métrica de qualidade que pode ser usada para medir até que ponto o software executará sem qualquer falha dentro de um período de tempo especificado (BOWEN; WIGLE; TSAI, 1985).

Consistência. (1) Na engenharia de software, uma métrica de qualidade que pode ser usada para medir as características de software que fornecem design uniforme, técnicas de implementação e notação. (2) Na engenharia de software (especificações de requisitos), uma especificação é consistente na medida em que as suas disposições não conflitam entre si.

Dados. (1) Uma representação de fatos, conceitos ou instruções de forma formalizada adequada para comunicação, interpretação ou processamento por meios humanos ou automáticos. (2) Os meios de comunicação de conceitos, planos, descrições, requisitos e instruções relativos a projetos técnicos, materiais, sistemas e serviços. Estes podem incluir especificações, normas, desenhos de engenharia, listas associadas, manuais e relatórios, incluindo relatórios científicos e técnicos; Eles podem ser na forma de documentos, displays, registros de som, cartões perfurados e dados digitais ou analógicos. (3) Na análise de requisitos de software, os dados podem ser categorizados como entradas e saídas, dados armazenados ou dados transitórios (resultados temporários).

Desenvolvedor. Na indústria, a pessoa ou pessoas que produzem um produto para um cliente. O cliente e o desenvolvedor podem ser membros da mesma organização. Um desenvolvedor às vezes é chamado de fornecedor (COMMITTEE; BOARD, 1998b).

Disponibilidade. (1) Na engenharia de software, uma métrica de qualidade de soft-

ware que pode ser usada para medir a probabilidade de que o software será capaz de executar sua função de sistema quando necessário. (2) A razão entre o tempo de funcionamento do sistema de computador e o tempo total de operação. (3) A capacidade de um sistema de software executar sua função designada quando necessário.

Documentação. (1) Uma coleção de documentos sobre um determinado assunto. (2) O processo de geração de um documento. (3) Qualquer informação escrita ou pictórica descrevendo, definindo, especificando, reportando ou certificando atividades, requisitos, procedimentos ou resultados (NQA, 1). (4) Na engenharia de software, comumente usado para significar documento do usuário, manual de manutenção de software, manual do operador. *Consulte também documento de entrega, documento, documentação de software, documentação de manutenção de software, documentação do usuário.*

Documento. Um meio de dados e os dados gravados nele. Os dados geralmente têm permanência e podem ser lidos por humanos ou máquinas. O termo documento é freqüentemente usado para descrever apenas itens legíveis por humanos, por exemplo, documentos técnicos, documentos de design e documentos de gerenciamento.

Eficiência. Na engenharia de software, uma métrica que pode ser usada para medir: (1) a extensão relativa à qual um recurso é usado (isto é, armazenamento, espaço, tempo de processamento, tempo de comunicação) (RZEPKA, 1985); (2) a medida em que o software executa suas funções pretendidas com um consumo mínimo de recursos de computação (BRANSTAD; POWELL, 1984).

Engenharia. A ciência interessada em colocar o conhecimento científico para usos práticos. *Consulte também engenharia de requisitos de software, engenharia de software, engenharia de sistemas, engenharia de requisitos de sistema.*

Engenharia de requisitos. Na engenharia de sistemas, a ciência e a disciplina relacionadas com a análise e documentação de requisitos. Ele compreende análise de necessidades, análise de requisitos e especificações de requisitos. *Ver também requisito.*

Engenharia de requisitos do sistema. A ciência e a disciplina relacionadas com a análise e documentação dos requisitos do sistema. Trata-se de transformar uma necessidade operacional em uma descrição de sistema, com parâmetros de desempenho do sistema e uma configuração do sistema, através do uso de um processo iterativo de definição, análise, tradeoff estudos e prototipagem. *Ver também requisito.*

Engenharia de software. (1) A aplicação prática da ciência da computação, gestão e outras ciências para a análise, concepção, construção e manutenção de software e sua documentação associada. (2). Uma ciência de engenharia que aplica o conceito de análise, design, codificação, testes, documentação e gestão para a conclusão bem-sucedida de grandes programas de computador custom-built. (3). A aplicação sistemática de métodos, ferramentas e técnicas para alcançar um requisito ou objetivo estabelecido para um sistema de software eficaz e eficiente. (4). A aplicação de princípios científicos a: (1) a transformação precoce de um problema em uma solução de software de trabalho e (2) a manutenção subsequente desse

software até o final de sua vida útil (DAVIS, 1988).

Especificação de requisitos. Na engenharia de sistemas / software, um documento que estabelece as funções que o software deve executar, o nível de desempenho necessário (velocidade, precisão, etc.), a natureza das interfaces necessárias entre o produto de software e seu ambiente, o tipo e severidade de restrições no projeto, e a qualidade do produto final. Sinônimo de especificação externa. *Consulte também especificação de requisitos de software, especificação de requisitos do sistema.*

Especificação de requisitos de performance. Especificação que estabelece os requisitos de desempenho para um sistema / software ou componente de sistema / software (BRANSTAD; POWELL, 1984). *Consulte também especificação de requisitos funcionais, especificação de requisitos de software (SRS).*

Especificação de requisitos funcionais. Na engenharia de sistemas / sistemas de software, uma especificação que estabelece os requisitos funcionais para um sistema / sistema de software ou componente de sistema / software. Às vezes sinônimo de exigência funcional. *Ver também requisito funcional.*

Especificação externa. Sinônimo de especificação de requisitos. Contraste com especificação interna.

Especificação formal. Na engenharia de software: (1) uma especificação escrita e aprovada em conformidade com as normas estabelecidas; e (2) um documento que utiliza principalmente a matemática para especificar os requisitos do sistema.

Estado da arte. Na engenharia, a tecnologia atual (mais recente), mas não necessariamente em uso. Contraste com o estado-da-prática.

Fase. (1) O estágio de desenvolvimento em um produto ou atividade; e (2) na engenharia de sistemas de software, uma das etapas do modelo de ciclo de vida.

Fase de ciclo de vida. Uma fase de um modelo de ciclo de vida.

Hardware. O design mecânico, magnético e eletrônico, estrutura e dispositivos de um computador. *Consulte também hardware do computador.*

Incompletude. Na engenharia de software (requisitos), especificações em que muitos dos aspectos dos requisitos são deixados de fora. O leitor deve preencher essas lacunas usando seu próprio conhecimento ou fazendo perguntas.

Interoperabilidade. Na engenharia de software, uma métrica de qualidade que pode ser usada para medir: (1) o esforço relativo para acoplar o software de um sistema ao software de outro sistema (RZEPKA, 1985); e (2) a capacidade de dois ou mais sistemas para trocar informações e usar-se mutuamente as informações que foram trocadas (BRANSTAD; POWELL, 1984). *Veja também compatibilidade.*

Iteração. Na engenharia de software, a repetição de uma etapa anterior ou processo como mais informações tornam-se disponíveis. Por outras palavras, o processo “n” pode ser iterado com o processo “n - 1” em que cada processo “n - 1” é lido parcialmente ou na totalidade com base na informação que foi determinada durante o processo n. (2) Na programação, o

processo de execução repetida de uma determinada sequência de instruções de linguagem de programação até que uma determinada condição seja atendida ou enquanto uma dada condição for verdadeira (BRANSTAD; POWELL, 1984).

Linguagem formal. (1) Uma língua cujas regras são explicitamente estabelecidas antes da sua utilização. Sinônimo de linguagem artificial. (2) No desenvolvimento de software, uma linguagem de computador que é usada para representar um requisito de software que pode ser provado “correto” por meio de métodos de prova de correção ou pode ser traduzido diretamente em um programa de aplicativo operacional que implementa os requisitos. *Veja também o linguagem.*

Linguagem natural. Uma língua falada por pessoas, ao contrário de uma linguagem formal ou uma linguagem usada por computadores. Contraste com a linguagem formal.

Manutenabilidade. Na engenharia de software, uma métrica de qualidade que pode ser usada para medir a facilidade de esforço para localizar e corrigir uma falha de software dentro de um período de tempo especificado (RZEPKA, 1985).

Modelo de aplicação. Um modelo de uma aplicação particular.

Portabilidade. Na engenharia de sistemas de software, uma métrica de qualidade que pode ser usada para medir o esforço relativo para transportar o software para uso em outro ambiente ou para converter software para uso em outro ambiente operacional, configuração de hardware ou ambiente de sistema de software (RZEPKA, 1985).

Qualidade do software. Na engenharia de software: (1) a totalidade das características e características de um produto de software que tem a sua capacidade de satisfazer determinadas necessidades, por exemplo, de acordo com as especificações; (2) o grau em que o software possui uma combinação desejada de atributos; (3) o grau em que um cliente ou usuário percebe que o software atende às suas expectativas compostas; (4) as características compostas de software que determinam o grau em que o software em uso atenderá às expectativas do cliente; (BRANSTAD; POWELL, 1984) e (5) atributos de software que afetam seu valor percebido, por exemplo, correção, confiabilidade, manutenção e portabilidade.

Requisito. Na engenharia de sistema / software: (1) uma capacidade necessária por um usuário para resolver um problema ou alcançar um objetivo; (2) uma capacidade que deve ser atendida ou possuída por um sistema ou componente do sistema para satisfazer um contrato, norma, especificação ou outro documento imposto formalmente; (3) o conjunto de todos os requisitos que constituem a base para o subsequente desenvolvimento do software ou componente de software; e (4) breve descrição às vezes usado em vez do termo especificação de requisitos de software. Consulte também requisito de software, requisito do sistema. *Consulte também especificação de requisitos, especificação de requisitos de software, especificação de requisitos do sistema.*

Requisito de performance. Um requisito do sistema / sistema de software especifi-

cando uma característica de desempenho que um sistema / sistema de software ou componente de sistema / software deve possuir; Por exemplo, velocidade, precisão e frequência.

requisitos do sistema. Na engenharia de sistemas: (1) uma capacidade do sistema necessária para resolver um problema ou alcançar um objetivo; (2) uma capacidade do sistema que deve ser satisfeita (ou possuída) por um sistema ou componente para satisfazer um contrato, padrão, especificação ou outro documento formal; e (3) o conjunto de todos os requisitos do sistema que constitui a base para o subsequente desenvolvimento do sistema ou componente do sistema (BRANSTAD; POWELL, 1984). *Consulte também a especificação de requisitos de sistema (SRS), requisito de software.*

Requisito essencial. Na engenharia de software (especificações de requisitos), um grau de necessidade que implica que estes são requisitos obrigatórios. O produto de software não será aceito sem eles. Contraste com requisito desejável, requisito opcional. *Consulte também requisito, requisito de software.*

Requisito funcional. Um requisito de sistema / software que especifica uma função de um sistema / sistema de software ou componente de sistema / software deve ser capaz de executar. São requisitos de software que definem o comportamento do sistema, ou seja, o processo ou transformação fundamental que os componentes de software e hardware do sistema realizam em entradas para produzir saídas. *Consulte também especificação de requisitos funcionais.*

Security. Na engenharia de software (métricas de qualidade), trata-se de tolerância a falhas contra falhas intrusivas deliberadas de fontes internas ou externas. 2. Na engenharia de software, o estabelecimento e aplicação de salvaguardas para proteger dados, software e hardware de computador de modificação, destruição ou divulgação acidental ou mal-intencionada.

Sistema de computador. (1) Um sistema composto por um computador, um equipamento periférico, como discos, impressoras, terminais e o software necessário para fazê-los operar em conjunto. (2) Uma unidade funcional, constituída por um ou mais computadores e software associado, que utiliza armazenamento comum para todo ou parte de um programa e também para todo ou parte dos dados necessários para a execução do programa; Executa programas escritos pelo usuário ou designados pelo usuário; Executa manipulação de dados designada pelo usuário, incluindo operações aritméticas e operações lógicas; E pode executar programas que se modificam durante sua execução. Um sistema de computador pode ser uma unidade autônoma ou pode consistir em várias unidades interligadas, ou pode ser incorporado num sistema de processamento de dados.

Sistemas de computador embarcado. Sistema de computador integrado a um sistema maior cujo principal objetivo não é computacional, por exemplo, um sistema de computador em uma arma, aeronave, automóvel, rede de comunicação, sistema inteligente de ponto de venda, e caixa eletrônico. *Veja também sistema de computador.*

Software - Em informática, processamento de dados e engenharia de software, uma seqüência de instruções adequadas para processamento por computador. O processamento pode incluir o uso de um assembler, um compilador, um intérprete ou um tradutor para preparar o programa para execução, bem como para executá-lo (BRANSTAD; POWELL, 1984). *Sinônimo de código, código de computador, programa de computador. Contraste com hardware.*

Software embarcado. Software para um sistema de computador embarcado. *Veja também sistema de computador embarcado.*

Tempo real. Em engenharia de sistemas / software: 1. Pertencente ao processamento de dados fornecidos por um processo externo de acordo com os requisitos de tempo impostos pelo processo externo. Este termo também é usado para descrever sistemas que operam em modo conversacional ou processos que podem ser influenciados pela intervenção humana enquanto em andamento. 2. Pertencente ao tempo durante o qual um processo físico transpira; Por exemplo, o desempenho de uma computação durante o tempo real em que o processo físico relacionado ocorre, de modo que os resultados da computação podem ser usados na orientação do processo físico (BRANSTAD; POWELL, 1984).

Verificabilidade. Na engenharia de software (métrica), uma métrica usada para medir o esforço relativo para verificar a operação e o desempenho especificados do software (RZEPKA, 1985). 2. Na análise de requisitos de software, uma especificação de requisitos de software só é verificável se todos os requisitos nela declarados forem verificáveis. Um requisito só é verificável se existir algum processo de custo-benefício finito pelo qual uma pessoa ou máquina pode verificar se o produto de software real, tal como construído, satisfaz o requisito.

Verificação de requisitos. *Ver verificação*

B

PROTOCOLO DA REVISÃO SISTEMÁTICA DA LITERATURA DE ENGENHARIA DE RE- QUISITOS NOS SISTEMAS EMBARCADOS

**Protocolo de Revisão Sistemática Sobre
Abordagens de Engenharia de Requisitos para Sistemas Embarcados**
Deivson Albuquerque

1. Introdução

Diferentes domínios de sistemas embarcados como o automotivo, o aviônico e a tecnologia médica têm de lidar com a complexidade do sistema causada pelo número crescente de funções e pelo crescente número de interações entre essas funções (SIKORA; TENBERGEN; POHL, 2012). À medida que a complexidade aumenta, o risco de erros aumenta significativamente e pode resultar em atraso e encerramento do projeto. De acordo com LIGGESMEYER; TRAPP (2009), mais de 50% dos problemas que ocorrem em sistemas embarcados são causados após a entrega do sistema. Assim, um bom processo de engenharia de requisitos tem que ser seguido para minimizar os erros e fornecer um projeto bem sucedido. O objetivo do estudo descrito neste protocolo é revisar o estado atual das abordagens de engenharia de requisitos para sistemas embarcados e fornecer diretrizes para realização da RSL.

2. Questões de Pesquisa e Motivações

Questões de pesquisa	Descrição e Motivação
RQ1. Como as abordagens contribuem para a solução de problemas de engenharia de requisitos no domínio de sistemas embarcados?	O objetivo desta questão é identificar e descrever abordagens de engenharia de requisitos para sistemas embarcados. É importante porque fornece um conjunto de contribuições relativas à utilização dessas abordagens para resolver alguns problemas de pesquisa ER bem conhecidas e que podem ser úteis para pesquisas interessadas em usar algumas destas abordagens no ER para o domínio de sistemas embarcados.
RQ2. Quais as fases do processo de ER apoiadas pelas abordagens?	Esta questão fornece um ponto de partida para entender o que são as principais fases (elicitação, análise, especificação, validação e gerenciamento) do processo de engenharia de requisitos suportados pelas abordagens e obter um parâmetro sobre quais as fases são mais relevantes no processo de ER nos SE.
RQ3. Que tipo de requisito (funcional e/ou não-funcional) foi apoiado pelas abordagens?	A questão é importante para poder identificar a distribuição dos estudos, de modo que possa investigar se as abordagens de ER nos ES estão gerenciando ou levando em consideração os requisitos funcionais ou os não-funcionais.
RQ4. Quais as questões em aberto das abordagens encontradas?	A resposta a esta pergunta indica as questões em aberto existentes relacionadas com a engenharia de requisitos no domínio de SE. É importante porque identifica um conjunto de linhas de pesquisa que podem ser levadas em consideração para trabalhos futuros.
RQ5. Quais são os domínios que as abordagens suportam?	O objetivo desta pergunta é conhecer os diferentes domínios (tecnologia de automação, aviónica, tecnologia de energia automóvel ou tecnologia médica) que as abordagens de ER nos embarcados suportam e identificar o domínio que necessita de uma melhor atenção em relação a ER.
RQ6. Quais requisitos devem ser considerados durante o desenvolvimento de sistemas embarcados?	O objetivo desta pergunta é identificar os requisitos que devem ser levados em conta no desenvolvimento de sistemas embarcados, ou seja, os requisitos relevantes que devem ser detalhados e especificados no processo de ER dos SE.

3. Palavras-chave e *Strings* de Pesquisa

#	Termos de busca:
(1)	"requirements engineering" OR "requirements elicitation" OR "requirements specification" OR "requirements management" OR "requirements validation" OR "requirements education"
(2)	"requirements modeling OR requirements modelling"
(3)	"embedded systems" OR "safety critical systems" OR "real time systems" OR "embedded software" OR "embedded product"
(4)	"approach" OR "technique" OR "framework" OR "processes" OR "methods" OR "tool"

1 String:

("requirements engineering" OR "requirements elicitation" OR "requirements specification" OR "requirements management" OR "requirements validation" OR "requirements education" OR ("requirements modeling" OR "requirements modelling")) AND ("embedded systems" OR "safety critical systems" OR "real time systems" OR "embedded software" OR "embedded product") AND ("approach" OR "technique" OR "framework" OR "processes" OR "methods" OR "tool")

Engineering Village (377);

Science Direct (1114);

Scopus (1675);

SpringerLink (292);

ISI Web of Science (105);

2 *String*:

("requirements engineering" OR ("requirements modeling" OR "requirements modelling")) AND ("embedded systems" OR "embedded software" OR "real-time system" OR "real-time") AND ("approach" OR "framework" OR "processes" OR "tool")

ACM Digital Library (668) (*string* adaptada);

3 String:

("requirements engineering" OR "requirements modeling") AND ("embedded systems" OR "embedded software" OR "real-time system" OR "real-time") AND ("approach" OR "framework" OR "processes" OR "tool")

IEEEXplore (232)(*String* adaptada);

4. Base de dados eletrônicas

O processo de busca é de forma eletrônica, realizada em sites específicos. Os sites indicados são mostrados na tabela a seguir.

Fontes a serem pesquisadas

Fonte
IEEEXplore
ACM Digital Library
Engineering Village
Science Direct
Scopus
SpringerLink
ISI Web of Science

Pesquisas específicas também serão contatadas diretamente:

Fernanda Maria Ribeiro Alencar

Jaelson Freire Brelaz de Castro

5. Critérios de Seleção dos Estudos

The study selection criteria are governed both by the inclusion and exclusion criteria and are provided below.

#	Critérios de Inclusão
1	Primary studies
2	Estudos revisados por pares

#	Critérios de Inclusão
3	Estudos que usam alguma etapa da engenharia de requisitos para suportar o desenvolvimento de sistemas embarcados
4	Estudos publicados entre 1070 a junho de 2016
5	Satisfaz o limiar mínimo de qualidade

#	Critérios de Exclusão
1	Estudos Secundários
2	Estudos curtos ($< = 5$ páginas)
3	Estudos não revisados por pares
4	Estudos duplicados (apenas um dos estudos é incluído)
5	Estudos não escritos em inglês
6	Estudos que não trata do tema proposto
7	Estudos que não utilizam qualquer fase da engenharia de requisitos no desenvolvimento de sistemas embarcados
8	Literatura cinzenta
9	Estudo redundante de alguma autoria*

* Quando um estudo for publicado em mais de uma revista / conferência, a versão mais completa será considerada.

6. Processo de seleção de estudos primários

O candidato relevante e os estudos selecionados serão selecionados por uma única pesquisa. Os estudos rejeitados serão verificados por outra pesquisa. Manteremos uma lista de candidatos que foram rejeitados com as razões para a rejeição (critérios de exclusão).

7. Avaliação de Qualidade

A avaliação da qualidade (QA) dos estudos selecionados será realizada por meio de uma técnica de pontuação para avaliar a credibilidade, integridade e relevância dos estudos selecionados. O instrumento de avaliação utilizado é apresentado a seguir. Q1, Q2, Q3, Q4, Q5, Q6, Q8, Q9, Q10, Q12, Q13 e Q14 foram adotados da literatura, enquanto Q7 e Q11 foram propostos.

#	Questões	Resposta Possível
1	Há uma justificativa plausível para realização do estudo? (GALSTER et al., 2014)	$Y = 1, N = 0, P = 0.5$
2	O artigo é baseado em pesquisa (ou é apenas um relatório de lições aprendidas com base na opinião de especialistas)? (DYBAA; DINGSOYR, 2008)	$Y = 1, N = 0$
3	Existe uma declaração clara dos objetivos da pesquisa? (DYBAA; DINGSOYR, 2008)	$Y = 1, N = 0, P = 0.5$
4	A abordagem proposta está claramente descrita? (ACHIMUGU et al., 2014)	$Y = 1, N = 0, P = 0.5$
5	O contexto da pesquisa foi descrito em um nível adequado (indústria, laboratório, produtos utilizados, etc.)?(ACHIMUGU et al., 2014)	$Y = 1, N = 0, P = 0.5$
6	Quantas fases do processo de engenharia de requisitos o estudo suporta? (DERMEVAL et al., 2015)	Nº de fases / Total de fases =
7	As abordagens suportam mais de um domínio (automação tecnológica, automotivo, aviônico, tecnologia de energia ou tecnologia médica)?	$Y = 1, N = 0$
8	O estudo é apoiado por uma ferramenta? (DERMEVAL et al., 2015)	$Y = 1, N = 0$
9	Existe uma discussão sobre os resultados do estudo? (DERMEVAL et al., 2015)	$Y = 1, N = 0, P = 0.5$
10	A limitação deste estudo é explicitamente discutida? (MARTINS; OLIVEIRA, 2014a)	$Y = 1, N = 0, P = 0.5$
11	Existe uma descrição clara das questões em aberto relacionadas com os estudos que foram realizados?	$Y = 1, N = 0, P = 0.5$
12	A pesquisa também agrega valor à comunidade industrial? (ACHIMUGU et al., 2014), (DYBAA; DINGSOYR, 2008)	$Y = 1, P = 0.5$
13	Existe informação suficiente sobre o estudo para que possa ser replicado? (ACHIMUGU et al., 2014)	$Y = 1, N = 0$
14	Ameaças validade dos estudos são discutidas e medidas são aplicadas? (ACHIMUGU et al., 2014)	$Y = 1, N = 0, P = 0.5$

8. Estratégia de Extração de Dados e Síntese

A fim de orientar a extração de dados, a coleta de dados de KITCHENHAM et al. (2009) foi adotada.

#	Dados do Estudo	Descrição	RQ Relevante
1	Identificação dos estudos	ID exclusivo para o estudo	Visão geral do estudo
2	Data da extração dos dados		Visão geral do estudo
3	Autores, ano, título, país		Visão geral do estudo
4	Fonte do artigo		Visão geral do estudo
5	Tipo de artigo	Periódico, conferência, <i>workshop</i> , capítulo de livro	Visão geral do estudo
6	Contexto de aplicação	Industrial, acadêmico	Visão geral do estudo
7	Método de pesquisa (baseado em EASTERBROOK et al. (2008))	experimento controlado, estudo de caso, <i>survey</i> , etnografia, pesquisa-ação, cenário ilustrativo, estudo não aplicado	Visão geral do estudo
8	Pontuação de qualidade dos estudos	Pontuação baseada na avaliação de qualidade	Visão geral do estudo
9	Objetivos	Quais os objetivos do estudo? Quais os problemas de ER abordados nos estudos?	RQ1
10	Tipo de contribuição (baseado em PETERSEN et al. (2008))	Quais são os tipos de contribuições dos estudos? (Método, Ferramenta, processo, modelo) Qual é o nível de suporte da ferramenta? (O nível é baseado na quantidade de fases de ER)	RQ6
11	Fases dos requisitos (baseado em PETERSEN et al. (2008))	Quais as fases de ER abordadas pelos estudos? (elicitação, análise e negociação, especificação, validação e gerenciamento)	RQ2
12	Estilo de requisitos de <i>software</i>	Quais foram os estilos de modelagem de requisitos de software abordados nos estudos?	RQ3

#	Dados do Estudo	Descrição	RQ Relevante
13	Tipos de requisitos	Quais foram os tipos de requisitos (requisitos funcionais ou não-funcionais) abordados nos estudos?	RQ4
14	Domínios	Quais os domínios suportados nos estudos (automação tecnológica, automotivo, aviônico, tecnologia energética ou tecnologia médica)?	RQ5
15	Questões em aberto	Quais são os problemas em aberto das abordagens?	RQ7

9. Disseminação

Os resultados do estudo devem ser de interesse para a comunidade de engenharia de software, bem como pesquisas interessadas em ER para sistemas embarcados. Por esse motivo, planejamos relatar os resultados nos seguintes formatos:

- Em periódicos ou conferência de artigos;
- Em um capítulo da dissertação de Mestrado;
- Página da *WEB*;



IDENTIFICAÇÃO DOS ESTUDOS DA RSL

ID	Autores	Títulos
S01	Aceituna, et al.	Interactive requirements validation for reactive systems through virtual requirements prototype
S02	Anderson, et al.	Analysis of timeliness requirements in safety-critical systems
S03	Aoyama, et al.	AORE (aspect-oriented requirements engineering) methodology for automotive software product lines
S04	Arpinen, et al.	Meta-model and UML profile for requirements management of software and embedded systems
S05	Ben-Abdallah, et al.	The integrated specification and analysis of functional, temporal, and resource requirements
S06	Ben-Abdallah, et al.	Specification and analysis of real-time systems with PARAGON
S07	Blouin, et al.	Defining an annex language to the architecture analysis and design language for requirements engineering activities support
S08	Boulanger, et al.	Requirements engineering in a model-based methodology for embedded automotive software
S09	Braun, et al.	Guiding requirements engineering for software-intensive embedded systems in the automotive industry: The REMsES approach
S10	Brink et al.	Automatic analysis of embedded systems specified in Astral
S11	Brink, et al.	Experiences with analysis of formal specifications in Astral
S12	Busse, et al.	Formal Specification and Automated Verification of Safety-Critical Requirements of a Railway Vehicle with Frama-C/Jessie
S13	Carvalho, et al.	Using Aspect-Oriented Concepts in the Requirements Analysis of Distributed Real-Time Embedded Systems
S14	Delemos, et al.	A TRAIN SET AS A CASE-STUDY FOR THE REQUIREMENTS ANALYSIS OF SAFETY-CRITICAL SYSTEMS

ID	Autores	Títulos
S15	Denger, et al.	Higher quality requirements specifications through natural language patterns
S16	Djouab, et al.	An ASPIRE-based method for quality requirements identification from business goals
S17	Doering, et al.	A model driven engineering approach based on aspects for high speed scientific X-rays cameras
S18	Dubois et al.	A Model for Requirements Traceability in a Heterogeneous Model-Based Design Process: Application to Automotive Embedded Systems
S19	Dutertre, et al.	Formal requirements analysis of an avionics control system
S20	Faulk, et al.	The Core method for real-time requirements
S21	Fernandes, et al.	Expressing Environment Assumptions and Real-time Requirements for a Distributed Embedded System with Shared Variables
S22	Ferrante, et al.	BCL: A compositional contract language for embedded systems
S23	Fidge, et al.	DISCIPLINED APPROACH TO REAL-TIME SYSTEMS-DESIGN
S24	Galle, et al.	Capturing and verifying performance requirements for hard real time systems
S25	Galvao et al.	A case study using a protocol to derive safety functional requirements from Fault Tree Analysis
S26	Goldsack, et al.	REQUIREMENTS ENGINEERING FOR REAL-TIME SYSTEMS
S27	Hansen, et al.	From safety analysis to software requirements
S28	Heimdahl et al.	Specification based prototyping of control systems
S29	Heninger, et al.	Specifying Software Requirements for Complex Systems: New Techniques and Their Application
S30	Jiale et al.	A Context-based Information Retrieval Technique for Recovering Use-Case-to-Source-Code Trace Links in Embedded Software Systems
S31	Jiale et al.	The observer-based technique for requirements validation in embedded real-time systems
S32	Koerner, et al.	Transferring Research Into the Real World: How to Improve RE with AI in the Automotive Industry
S33	Krueger, et al.	Requirements modeling for embedded realtime systems
S34	Kwon, et al.	Specification and analysis of timing requirements for real-time systems in the CBD approach
S35	Lattemann, et al.	Methodological approach to the requirement specification of embedded systems

ID	Autores	Títulos
S36	Lee, et al.	Reusable SW requirements development process: Embedded SW industry experiences
S37	Liu, et al.	Extending EAST-ADL2 to support aspectual requirement specification and analysis for automotive software
S38	Loniewski, et al.	Model-driven requirements engineering for embedded systems development
S39	Lundqvist, et al.	A TASM-Based Requirements Validation Approach for Safety-Critical Embedded Systems
S40	Markose, et al.	A Systematic framework for structured object-oriented security requirements analysis in embedded systems
S41	Mutz, et al.	Seamless model based development process of automotive systems
S42	Nasr, et al.	Eliciting and specifying requirements with use cases for embedded systems
S43	Petrov, et al.	An Aspect-Oriented Approach for Designing Safety-Critical Systems
S44	Pohl, et al.	Detecting and Correcting Outdated Requirements in Function-Centered Engineering of Embedded Systems
S45	Ravn, et al.	Specifying and verifying requirements of real-time systems
S46	Ribeiro, et al.	Application of an Extended SysML Requirements Diagram to Model Real-Time Control Systems
S47	Rota et al.	Integrating UML, MARTE and sysml to improve requirements specification and traceability in the embedded domain
S48	Roudier, et al.	Towards the Model-Driven Engineering of Security Requirements for Embedded Systems
S49	Saeed, et al.	An approach for the risk analysis of safety specifications
S50	Saeed, et al.	ON THE SAFETY ANALYSIS OF REQUIREMENTS SPECIFICATIONS FOR SAFETY-CRITICAL SOFTWARE
S51	Saeed, et al.	The role of formal methods in the requirements analysis of safety-critical systems: a train set example
S52	Saiedian, et al.	Scenario-based requirements analysis techniques for real-time software systems: a comparative evaluation
S53	Schneider, et al.	Validating requirements for fault tolerant systems using model checking
S54	Seridi-Bouchelaghem, et al.	A Novel Formal Specification Approach for Real Time Multi-Agent System Functional Requirements
S55	Siegl, et al.	Model based requirements analysis and testing of automotive systems with Timed Usage Models
S56	Slotosch, et al.	From Requirements to Validated Embedded Systems
S57	Smarandache, et al.	Applicability of SIGNAL in safety critical system development

ID	Autores	Títulos
S58	Tanurhan et al.	Integrated design process with MESA/MERLAN
S59	Tuunanen, et al.	A Method and Tool for Wide Audience Requirements Elicitation and Rapid Prototyping for Mobile Systems
S60	Vogelsang, et al.	Supporting concurrent development of requirements and architecture: A model-based approach
S61	Wikan et al.	Enforcing safety requirements for industrial automation systems at runtime position paper
S62	Wu, et al.	Requirements specifications checking of embedded real-time software
S63	Xiaohong et al.	Modeling Timing Requirements in Problem Frames Using CCSL
S64	Zafar, et al.	Integrating safety and security requirements into design of an embedded system
S65	Zave, et al.	An Operational Approach to Requirements Specification for Embedded Systems
S66	Zhou, et al.	Towards feature-oriented requirements validation for automotive systems
S67	Zhu, et al.	A Formal Model for Service-Based Behavior Specification Using Stream-Based I/O Tables
S68	Mole, et al.	Model-based systems engineering with requirements variability for embedded real-time systems
S69	Roopak, et al.	Requirements engineering of industrial automation systems: Adapting the CESAR requirements meta model for safety-critical smart grid software
S70	Rui, et al.	Ontology-based requirements generation for credibility validation of safetycritical system
S71	Nesredin, et al.	ReSA: An ontology-based requirement specification language tailored to automotive systems
S72	Takoshima, et al.	Assessing the quality of software requirements specifications for automotive software systems
S73	Musat, et al.	Semi-formal representation of requirements for automotive solutions using sysML
S74	SporerGeorg, et al.	A Lean Automotive E/E-System Design Approach with Integrated Requirements Management Capability
S75	Fengjie, et al.	Model-based requirements specification of real-time systems with UML

D

RESULTADOS DA AVALIAÇÃO DE QUALIDADE DA RSL

As Figuras B.1, B.2, B.3 e B.4 apresentam os resultados da avaliação de qualidade que foi realizada conforme as questões descritas na Tabela 3.5. Os resultados finais são mostrados no lado direito das tabelas. De acordo com DERMEVAL et al. (2015), um mínimo de 50% de qualidade é uma boa forma de estabelecer um limiar de qualidade aceitável para os estudos. Neste estudo, utilizamos também o mínimo de 50% de limiar de qualidade para incluir os estudos nos resultados finais. Os critérios e a tabela de análise de qualidade podem ser verificadas em: https://drive.google.com/file/d/0BzWtp8Q_rZpBOHhESFRrTUNkMjQ/view

Figura D.1 Análise de Qualidade 1-24

ID	Author	Year	Context	Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8	Q9	Q10	Q11	Q12	Total Score	Qual. (%)
S01	Accituna, et al.	2011	Academy	1.0	1.0	1.0	1.0	1.0	0.2	0.0	1.0	0.5	0.0	1.0	0.5	8.2	68.3
S02	Anderson, et al.	1991	Academy	1.0	1.0	1.0	1.0	0.5	0.4	0.0	0.0	0.0	1.0	0.0	0.5	6.4	53.3
S03	Aoyama, et al.	2008	Academy	1.0	1.0	1.0	1.0	1.0	0.4	0.0	0.0	1.0	0.0	1.0	0.5	7.9	65.8
S04	Arpinen, et al.	2011	Industry	1.0	1.0	0.5	0.5	0.5	0.2	1.0	1.0	0.5	0.0	0.5	1.0	7.7	64.2
S05	Ben-Abdallah, et al.	1997	Academy	0.5	1.0	0.0	1.0	1.0	0.6	0.0	1.0	1.0	1.0	0.0	0.5	7.6	63.3
S06	Ben-Abdallah, et al.	1999	Academy	1.0	1.0	1.0	1.0	1.0	0.4	1.0	1.0	1.0	1.0	0.5	0.5	10.4	86.7
S07	Blouin, et al.	2011	Academy	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	0.5	11.5	95.8
S08	Boulanger, et al.	2009	Academy	1.0	1.0	1.0	1.0	1.0	0.6	0.0	1.0	0.0	0.0	0.0	0.5	7.1	59.2
S09	Braun, et al.	2014	Industry	1.0	1.0	1.0	0.5	0.5	0.6	1.0	1.0	0.5	0.5	0.5	1.0	9.1	75.8
S10	Brink et al.	1998	Academy	1.0	0.0	1.0	1.0	1.0	0.4	0.0	1.0	0.0	0.5	1.0	0.5	7.4	61.7
S11	Brink, et al.	1997	Academy	0.5	1.0	1.0	1.0	0.0	0.6	0.0	1.0	1.0	1.0	1.0	0.5	8.6	71.7
S12	Busse, et al.	2011	Academy	0.5	1.0	1.0	0.5	1.0	0.4	0.0	1.0	0.5	1.0	0.5	0.5	7.9	65.8
S13	Carvalho, et al.	2007	Academy	1.0	1.0	1.0	1.0	1.0	0.4	0.0	0.0	1.0	0.0	1.0	0.5	7.9	65.8
S14	Delemos, et al.	1992	Industry	1.0	1.0	0.5	0.5	1.0	0.4	1.0	0.0	0.5	0.0	0.5	1.0	7.4	61.7
S15	Denger, et al.	2003	Academy	1.0	1.0	1.0	1.0	1.0	0.2	0.0	0.0	1.0	0.5	1.0	0.5	8.2	68.3
S16	Djonah, et al.	2014	Industry	1.0	1.0	1.0	1.0	1.0	0.6	0.0	0.0	0.5	0.0	0.0	1.0	7.1	59.1
S17	Doering, et al.	2013	Industry	1.0	1.0	1.0	1.0	1.0	0.6	0.0	0.0	1.0	0.5	1.0	1.0	9.1	75.8
S18	Dubois et al.	2010	Academy	1.0	1.0	1.0	1.0	1.0	0.6	1.0	1.0	0.0	0.0	1.0	0.5	9.1	75.8
S19	Dukertre, et al.	1997	Industry	1.0	1.0	1.0	1.0	1.0	0.6	0.0	1.0	1.0	1.0	0.0	1.0	9.6	80.0
S20	Faulk, et al.	1992	Academy	1.0	1.0	1.0	1.0	1.0	0.2	0.0	1.0	0.0	0.0	1.0	0.5	7.7	64.2
S21	Fernandes, et al.	2008	Industry	0.0	1.0	1.0	1.0	1.0	0.4	0.0	1.0	0.0	0.0	1.0	1.0	7.4	61.7
S22	Ferrante, et al.	2014	Academy	1.0	1.0	0.5	1.0	1.0	0.4	0.0	1.0	0.0	0.0	1.0	0.5	7.4	61.7
S23	Fidge, et al.	1992	Industry	0.5	1.0	1.0	1.0	0.5	0.2	1.0	0.0	0.5	0.5	0.5	1.0	7.7	64.2
S24	Galle, et al.	1997	Industry	0.0	1.0	1.0	1.0	1.0	0.4	0.0	1.0	0.0	0.0	0.0	1.0	6.4	53.3

Figura D.2 Análise de Qualidade 25-48

[illegible]

Figura D.3 Análise de Qualidade 26-71

ID	Author	Year	Context	Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8	Q9	Q10	Q11	Q12	Total Score	Qual. (%)
S49	Saeed, et al.	1994	Academy	0.5	1.0	1.0	1.0	1.0	0.6	0.0	0.0	0.0	0.5	0.0	0.5	6.1	50.8
S50	Saeed, et al.	1995	Industry	0.5	1.0	0.5	1.0	0.5	0.2	0.0	0.0	1.0	0.5	0.0	1.0	6.2	51.7
S51	Saeed, et al.	1991	Academy	1.0	1.0	0.5	1.0	1.0	0.6	0.0	0.0	0.5	0.0	0.0	0.5	6.1	50.8
S52	Saiedian, et al.	2005	Industry	1.0	1.0	1.0	1.0	1.0	0.2	1.0	0.0	1.0	0.5	1.0	1.0	9.7	80.8
S53	Schneider, et al.	1998	Industry	1.0	1.0	1.0	0.5	1.0	0.2	0.0	0.0	1.0	0.0	0.0	1.0	6.7	55.8
S54	Seridi et al.	2010	Industry	1.0	1.0	0.5	0.5	1.0	0.4	1.0	0.0	0.5	0.0	0.0	1.0	6.9	57.5
S55	Siegl, et al.	2010	Industry	1.0	1.0	0.5	0.5	1.0	0.6	0.0	0.0	0.5	0.5	0.0	1.0	6.6	55.0
S56	Slotosch, et al.	2001	Industry	1.0	1.0	1.0	0.5	1.0	0.6	0.0	1.0	0.0	0.0	0.0	1.0	7.1	59.2
S57	Smarandache, et al.	1999	Academy	1.0	1.0	1.0	1.0	1.0	0.4	0.0	0.0	0.5	0.0	1.0	0.5	7.4	61.7
S58	Tanurhan et al.	1996	Industry	1.0	1.0	0.5	0.5	1.0	0.6	0.0	1.0	0.0	0.0	0.0	1.0	6.6	55.0
S59	Tuunanen, et al.	2004	Industry	1.0	1.0	1.0	1.0	1.0	0.2	0.0	1.0	0.5	0.0	1.0	1.0	8.7	72.5
S60	Vogelvang, et al.	2014	Industry	1.0	1.0	1.0	1.0	1.0	0.2	0.0	1.0	1.0	0.5	0.5	1.0	9.2	76.7
S61	Wikan et al.	2011	Academy	1.0	1.0	1.0	1.0	1.0	0.4	0.0	0.0	1.0	1.0	1.0	0.5	8.9	74.2
S62	Wu, et al.	2002	Academy	1.0	1.0	1.0	1.0	0.0	0.2	0.0	1.0	0.0	0.0	0.5	0.5	6.2	51.7
S63	Xiaohong et al.	2011	Academy	1.0	1.0	1.0	1.0	1.0	0.4	0.0	0.0	0.0	0.0	1.0	0.5	6.9	57.5
S64	Zafar, et al.	2005	Academy	1.0	1.0	1.0	0.5	0.0	0.6	1.0	1.0	0.5	0.0	0.5	0.5	7.6	63.3
S65	Zave, et al.	1982	Academy	1.0	1.0	1.0	1.0	1.0	0.2	0.0	0.0	0.0	1.0	1.0	0.5	7.7	64.2
S66	Zhou, et al.	2014	Industry	0.5	1.0	0.5	0.5	1.0	0.4	0.0	1.0	0.5	0.5	0.5	1.0	7.4	61.7
S67	Zhu, et al.	2014	Academy	1.0	1.0	1.0	1.0	0.5	0.2	1.0	0.0	0.5	0.0	1.0	0.5	7.7	64.2
S68	Mole, et al.	2015	Industry	0.5	1.0	1.0	0.5	0.5	0.6	0.0	1.0	1.0	0.5	0.5	1.0	8.1	67.5
S69	Roopak, et al.	2015	Industry	1.0	1.0	1.0	0.5	0.5	0.2	1.0	0.0	1.0	1.0	0.5	1.0	8.7	72.5
S70	Rui, et al.	2015	Academy	1.0	1.0	1.0	0.5	0.5	0.6	0.0	0.0	1.0	1.0	1.0	0.5	8.1	67.5
S71	Nesredin, et al.	2015	Industry	1.0	1.0	1.0	0.5	1.0	0.4	0.0	0.0	0.5	0.5	1.0	1.0	7.9	65.8

Figura D.4 Análise de Qualidade 72-75

ID	Author	Year	Context	Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8	Q9	Q10	Q11	Q12	Total Score	Qual. (%)
S72	Takoshima, et al.	2016	Academy	1.0	1.0	1.0	1.0	0.5	0.2	0.0	0.0	0.0	0.0	1.0	0.5	6.2	51.6
S73	Musat, et al.	2015	Industry	0.5	1.0	0.5	0.5	1.0	0.6	0.0	1.0	0.0	0.5	0.5	1.0	7.1	59.1
S74	SporelGeorg, et al.	2015	Academy	1.0	1.0	1.0	0.5	0.0	0.4	0.0	1.0	0.0	0.0	1.0	0.5	6.4	53.3
S75	Goncalves, et al.	2015	Academy	1.0	1.0	1.0	1.0	1.0	0.6	1.0	1.0	1.0	0.5	0.0	0.5	9.6	80.0