

# Melhoria de Documentos de Requisitos: um mapeamento sistemático da literatura.

Roberto Azevedo<sup>1</sup>, Aêda Sousa<sup>1</sup>, Ricardo Ramos<sup>2</sup>, e Fernanda Alencar<sup>1</sup>

<sup>1</sup>Universidade de Pernambuco, UPE, Recife, Brasil  
{rcostape,fernandaalenc, aedasousa}@gmail.com

<sup>2</sup>Universidade Federal do Vale do São Francisco, UNIVASF, Juazeiro, Brasil  
ricargentonramos@gmail.com

**Resumo.** Diversas técnicas são utilizadas, tanto pela indústria quanto pela academia, para a especificação de requisitos de um sistema de software. Dentre essas, destacam-se casos de uso e descrição textual que, como outras, apresentam problemas com relação a requisitos que foram abandonados; estruturas demasiadamente longas ou muito pequenas, pouco significativas e até desnecessárias; requisitos duplicados, fragmentados e espalhados. Tudo isso diminui a facilidade de entendimento, a modularidade e a reusabilidade do documento de requisitos, acarretando a inserção de erros nas fases posteriores do processo de desenvolvimento. Para esses casos, o quanto antes os problemas forem encontrados, menor será o custo para removê-los. Nesta pesquisa, utilizou-se o mapeamento sistemático da literatura como procedimento metodológico, com o objetivo de verificar, com bases em evidências, o estado atual de estudos sobre a melhoria de documentos de requisitos. A ideia foi verificar como essas melhorias foram validadas e se utilizam alguma ferramenta de apoio. Diversos estudos propõem diferentes técnicas para avaliação e melhoria de documentos de requisitos por meio de refatorações e padrões, a fim de minimizar ou remover deficiências. Após análise dos estudos identificados, melhorias serão propostas à abordagem AIRDoc (Approach to Improve Requirements Documents), bem como o desenvolvimento de uma ferramenta para apoiar a execução do processo de forma efetiva.

**Palavras-Chave.** mapeamento sistemático, documentos de requisitos, melhoria, qualidade.

## 1 Introdução

A maioria dos problemas de software surge de deficiências na maneira em que os requisitos de software são elicitados, gerenciados e expressos [33]. Elaborar um documento de requisitos de software com qualidade é essencial para o sucesso de qualquer projeto de desenvolvimento de software, tendo em vista a importância da sua utilização em diversas etapas do projeto [37]. No entanto, a avaliação da qualidade de um documento de requisitos não é um processo simples, principalmente devido aos diversos tipos de propostas, abordagens, técnicas e ferramentas, muitas vezes contraditórias quando se trata dos atributos de qualidade a serem avaliados e as metodologias utilizadas [33].

A descrição textual, por meio da linguagem natural, é a principal representação utilizada nos documentos de requisitos elaborados na indústria, o que implica que os documentos de requisitos estejam suscetíveis à ambiguidade, ou seja, a mesma frase pode ser interpretada de várias formas diferentes [6]. Outra técnica bastante utilizada é o modelo de Caso de Uso, o qual permite modelar as necessidades dos usuários em um documento de requisitos, por meio da Descrição do Caso de Uso e de uma possível representação em um Diagrama de Casos de Uso. A modelagem de Casos de Uso é uma ferramenta poderosa para a captura de requisitos funcionais de sistemas de software, pois possibilitam especificar a interação entre o sistema de software e seus usuários [34]. Entretanto, por serem geralmente descritos por textos, significa que eles também são afetados por questões como a ambiguidade, redundância, inconsistência e incompletude.

Diversas pesquisas foram realizadas, a fim de lidar com esses problemas de qualidade do documento de requisitos. A literatura fornece modelos [37][38], diretrizes para a criação de requisitos de software [18], e listas de verificação para inspecionar Casos de Uso [36]. Outros buscam alguma forma de automatizar, com apoio de ferramentas, a detecção desses erros em documentos de requisitos descritos em Linguagem Natural [39].

A AIRDoc [13] propõe a utilização de métricas para a detecção de problemas recorrentes em documentos de requisitos especificados com Casos de Uso. Com a aplicação das métricas, um engenheiro (ou equipe de qualidade) identificará trechos no documento de requisitos que poderão ser melhorados com a utilização de um catálogo de refatorações de requisitos. Atualmente, a AIRDoc não é apoiada por ferramenta de automatização, o que dificulta sua utilização devido à quantidade de atividades necessárias para a sua correta aplicação.

Baseado nesse contexto, este trabalho reúne a síntese das informações de um mapeamento sistemático da literatura (MSL) cujo objetivo é verificar, por evidências, estudos focados na melhoria de documentos de requisitos; como as propostas foram validadas; e, se utilizam alguma ferramenta para automatização. O mapeamento inclui estudos a partir de 1986, ano em que os primeiros Casos de Uso foram divulgados à comunidade acadêmica por Ivar Jacobson [32], e segue um protocolo de pesquisa adaptado de Kitchenham [28]. A pesquisa iniciou com 3616 estudos, sendo 3297 resultantes de buscas automáticas a bases de dados acadêmicos e 319 do repositório do WER. Após a etapa de seleção, 27 trabalhos relevantes à pesquisa foram analisados, a fim de obter informações necessárias para responder as questões de pesquisa definidas.

O artigo está estruturado nas seguintes seções: a seção 2 apresenta o processo de mapeamento sistemático da literatura. Na seção 3 são descritos e analisados os resultados alcançados com a pesquisa. Na seção 4 são descritas possíveis ameaças à validade dos resultados. A seção 5 apresenta alguns trabalhos relacionados. E por fim, a seção 6 traz as conclusões dos estudos realizados e trabalhos futuros.

## 2 Método de Pesquisa

O Mapeamento Sistemático apresentado foi conduzido baseado na metodologia utilizada para a realização de uma revisão sistemática da literatura recomendada pela Kitchenham [28]. Algumas partes da metodologia utilizada no estudo foram apresentadas por outras revisões sistemáticas [29][30][35]. A Figura 1 apresenta o processo utilizado para a realização do MSL.

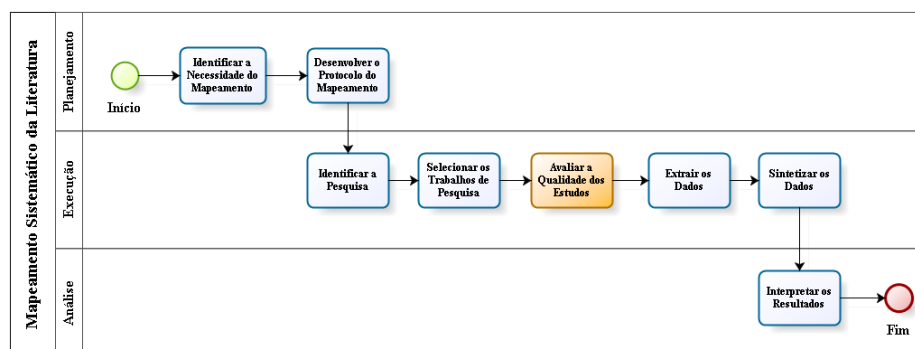


Fig. 1. Processo de Condução do Mapeamento Sistemático. Adaptado de [28].

### 2.1 Questões de Pesquisa e Contexto

Para verificar o estado atual de estudos com foco na melhoria de documentos de requisitos, três Questões de Pesquisa (QPs) foram formuladas com a sua respectiva descrição, conforme Tabela 1.

Tabela 1 - Questões de Pesquisa e Descrição.

Questão de Pesquisa	Descrição
QP1 – Como os requisitos são especificados nas abordagens?	Pretende-se identificar a forma como os requisitos são documentados (e.g., Modelos de Casos de Uso, Ponto de Vista, Descrição Textual) nas abordagens (técnicas, processos, frameworks, métodos) focadas na melhoria de documentos de requisitos. É importante, pois provê um conjunto de contribuições voltadas para problemas de pesquisa já conhecidos, os quais podem ser úteis para pesquisadores que estão interessados no uso de alguma dessas abordagens para melhoria de Documentos de Requisitos.
QP2 – Quais abordagens são apoiadas por ferramentas de automatização?	A resposta para esta questão permite identificar quais das abordagens têm alguma ferramenta que as apoiam. Importante investigar como essas ferramentas são utilizadas e o grau de automatização das mesmas.
QP3 – Existem transformações	Objetiva-se identificar as diferentes formas de melhoria

aplicadas a Documentos de Requisitos visando à melhoria destes?	nos documentos de requisitos, e.g., <i>Patterns</i> , Diretrizes ( <i>Guidelines</i> ), Refatorações, <i>Checklists</i> . Ajudar a identificar qual transformação de melhoria necessita de mais esforço e experiência do responsável pela melhoria de documentos de requisitos.
---	---

## 2.2 Strings de Busca

Baseado nas questões de pesquisas, termos e palavras-chaves são identificados, a fim de definir a *String* de busca. Algumas modificações e ajustes foram necessários para permitir a execução da *String* nos diversos mecanismos de busca utilizados. A Tabela 2 apresenta a *String* de busca utilizada nesse mapeamento.

**Tabela 2 - String de Busca.**

<b><i>String</i></b>
(quality AND (improvement OR evolution OR refactoring) AND ("software requirement" OR "functional requirement") AND ("requirement specification" OR "requirement document"))

## 2.3 Critério de Exclusão e Inclusão

O MSL conduzido nesse artigo tem como objetivo verificar estudos primários focados na melhoria de documentos de requisitos publicados entre os anos de 1985 e 2015. Estudos que não foram escritos em Inglês, bem como Revisões Sistemáticas, *Surveys* dentre outros estudos secundários foram descartados.

É importante fazer a distinção entre os conceitos de melhoria de documentos de requisitos, utilizado neste artigo, e validação. O objetivo da fase de validação é detectar, junto aos clientes, se os requisitos estão bem escritos, livres de ambiguidade e completos. Já a melhoria de documentos de requisitos realiza a verificação destes e outros problemas, porém ela é realizada pela equipe de desenvolvimento, sem a participação do cliente.

A Tabela 3 apresenta todos os Critérios de Inclusão e Exclusão utilizados no mapeamento.

**Tabela 3 - Critérios de Inclusão e Exclusão.**

# Critérios de Inclusão
1 Estudos Primários
2 Estudos publicados entre Janeiro de 1985 até Novembro de 2015
3 Estudos que tenham como objetivo principal propor, explicitamente, alguma abordagem para melhoria de documentos de requisitos
# Critérios de Exclusão
6 Estudos Secundários

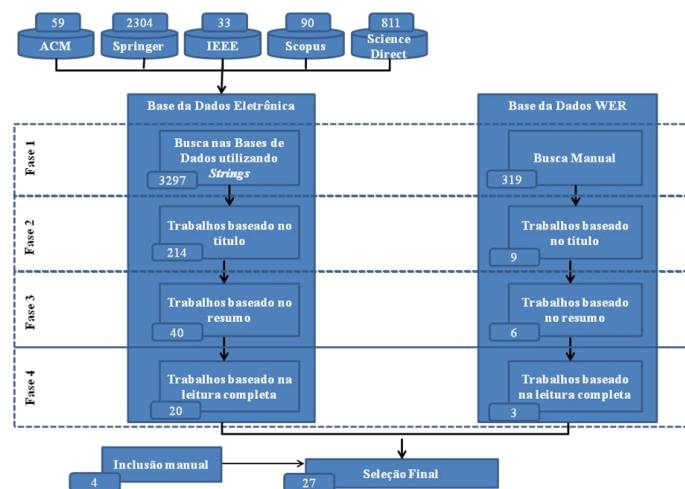
- 7 Artigos curtos ( $\leq 5$ )
- 8 Estudos Duplicados (apenas um será considerado)
- 9 Estudos que não tratam de abordagens de melhoria em documentos de requisitos
- 10 Estudo redundante de algum autor (a versão mais completa será considerada)
- 11 Artigos não inscritos em Inglês
- 12 Estudos não relacionados às questões de pesquisa
- 13 Estudo Inacessível
- 14 Estudos que não tratam de requisitos funcionais
- 15 Estudos que propõem melhorias em documentos de requisitos apenas na validação

## 2.4 Bases de Pesquisa

Foram utilizadas cinco bases de dados neste trabalho, sendo elas: IEEE Xplore, ScienceDirect, Scopus, SpringerLink e ACM Digital Library. Também foi realizada uma busca manual no Repositório do WER dos trabalhos publicados entre os anos de 1998 e 2015, os quais foram adicionados aos demais artigos.

## 2.5 Seleção dos Estudos

As buscas foram realizadas e os resultados passaram por 4 fases de seleção, conforme estabelecido no protocolo da pesquisa e apresentado na Figura 2.



**Fig 2.** Fases para Seleção dos Trabalhos, adaptado de [30].

A partir dos resultados das buscas eletrônicas nas bases de dados, referente à Fase 1 do protocolo, foi utilizada a ferramenta StArt Tool1, a qual facilitou a execução das demais fases do protocolo. Posteriormente, a Fase 2 do protocolo foi executada, na

<sup>1</sup> <http://lapes.dc.ufscar.br/software/start-tool>

<sup>2</sup> <http://wer.inf.puc-rio.br/WERpapers/>

qual os trabalhos tiveram seus títulos lidos, tendo como resultado 214 artigos dos 3297 iniciais. Em seguida, a Fase 3 consistiu em avaliar os resultados por meio da leitura dos resumos e disponibilidade por completo do artigo, sendo selecionados 40 dos 214. Na Fase 4, todos os trabalhos selecionados foram lidos por completo, restando para a seleção final 20 artigos. Para a busca manual no Repositório do WER<sup>2</sup>, na Fase 2, foram lidos todos os 319 títulos dos artigos. A partir da leitura do título, apenas 9 artigos foram selecionados. Na Fase 3, foram selecionados 5 artigos, sendo um dos motivos de exclusão um estudo redundante. Por fim, na Fase 4, um artigo foi rejeitado por estar duplicado com um artigo selecionado na busca eletrônica e outro por não tratar do objetivo da pesquisa, restando apenas 4 artigos. Em complemento, 4 trabalhos foram incluídos manualmente. A Tabela 4 apresenta a lista dos estudos selecionados (busca eletrônica e busca manual). O identificador foi definido como Trabalho mais Sequência Numérica no formato (T#) sem levar em consideração ordem alfabética ou cronológica.

**Tabela 4** - Estudos Selecionados.

<b>ID</b>	<b>Ano</b>	<b>Título</b>	<b>Referência</b>
<b>T1</b>	2015	A benchmarking process to assess software requirements documentation for space applications	[18]
<b>T2</b>	2005	Achieving high quality of use-case-based requirements	[3]
<b>T3</b>	2010	Ambiguity detection: Towards a tool explaining ambiguity sources	[6]
<b>T4</b>	2015	A methodology for the classification of quality of requirements using machine learning techniques	[11]
<b>T5</b>	2010	A method to evaluate the suitability of requirements specifications for offshore projects	[10]
<b>T6</b>	2008	Automated formal specification generation and refinement from requirement documents	[15]
<b>T7</b>	1989	Evaluation method for user requirements documents	[2]
<b>T8</b>	2010	Functional requirement improvements through size measurement: A case study with inexperienced measurers	[17]
<b>T9</b>	2014	Identifying duplicate functionality in textual use cases by aligning semantic actions	[4]
<b>T10</b>	2012	Improving the quality of software requirements specifications with Semantic Web technologies	[1]
<b>T11</b>	2010	Improving the quality of use case models using antipatterns	[9]
<b>T12</b>	2012	NLARE, a Natural Language Processing Tool for Automatic Requirements Evaluation	[7]
<b>T13</b>	2003	On the interplay between consistency, completeness, and correctness in requirements evolution	[20]
<b>T14</b>	2008	Practically relevant quality criteria for requirements documents	[16]
<b>T15</b>	2009	Quality improvement for use case model	[13]
<b>T16</b>	2008	Reducing Ambiguities in Requirements Specifications Via	[12]

		Automatically Created Object-Oriented Models	
<b>T17</b>	2006	Requirements quality control: a unifying framework	[14]
<b>T18</b>	2012	Supporting Learning Organisations in Writing Better Requirements Documents Based on Heuristic Critiques	[8]
<b>T19</b>	2014	Using model transformation to refactor use case models based on antipatterns	[5]
<b>T20</b>	2014	Using syntactic and semantic analyses to improve the quality of requirements documentation	[19]
<b>T21</b>	2005	A Content Analysis Technique for Inconsistency Detection in Software Requirements Documents	[22]
<b>T22</b>	2008	Can Rules of Inferences Resolve Coordination Ambiguity in Natural Language Requirements Specification	[23]
<b>T23</b>	2007	Requirements for Tools for Ambiguity Identification and Measurement in Natural Language Requirements Specifications	[21]
<b>T24</b>	2009	Ontologies in checking for inconsistency of requirements specification	[24]
<b>T25</b>	2001	An Automatic Quality Evaluation for Natural Language Requirements	[25]
<b>T26</b>	2001	An XML – based Approach for the Automatic Verification of Software Requirements Specifications	[26]
<b>T27</b>	2014	Rapid Requirements Checks with Requirements Smells: Two Case Studies	[27]

## 2.6 Avaliação da Qualidade

A avaliação da qualidade (QA) dos estudos selecionados foi obtida 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. As questões Q1, Q2, Q3, Q4 e Q5 foram adotadas e adaptadas a partir da literatura [29][30][35]. Este instrumento usa uma escala de pontuação com três graus (Sim = 1 ponto, Não = 0 ponto, e parcialmente = 0.5 pontos).

**Tabela 5 - Critérios da Avaliação da Qualidade**

# Critérios de Inclusão	Possíveis Respostas
1 O estudo descreve claramente o objetivo da abordagem, técnica, método ou processo?	S = 1, N = 0, P = 0.5
2 O estudo apresenta alguma ferramenta de automatização para tratar a melhoria de documentos de requisitos?	S = 1, N = 0, P = 0.5
3 O estudo explicita como a abordagem, técnica, método, processo ou ferramenta foi validado?	S = 1, N = 0, P = 0.5
4 As limitações do estudo são apresentadas?	S = 1, N = 0, P = 0.5
5 O estudo define quais os tipos de problemas que a abordagem trata?	S = 1, N = 0, P = 0.5

## 2.7 Extração dos Dados e Síntese

A fim de responder as Questões de Pesquisa deste trabalho, alguns dados específicos foram extraídos dos artigos selecionados, tais como: Identificador do Estudo, Autores, Título, Ano de Publicação e País, Contexto da Aplicação, Método de Pesquisa, Objetivos, Tipo de Contribuição, Forma de Especificação, Ferramenta Proposta, Validação e Tipos de Melhoria. Esses dados foram utilizados para realizar o fichamento dos estudos e responder as Questões de Pesquisa.

## 3 Análise dos Resultados

Nessa seção, são apresentados os resultados alcançados através da extração dos dados e as respostas das questões de pesquisa definidas. Foram identificados 27 estudos, conforme apresentado na Tabela 3.

### 3.1 Ano de Publicação

Tendo em vista que as primeiras aparições de Casos de Uso são datadas do ano de 1986 [32], foram considerados estudos publicados a partir do ano de 1985. A Figura 3 apresenta a quantidade dos trabalhos selecionados por ano de publicação. Pode-se verificar que a preocupação com a qualidade dos Documentos de Requisitos é constante, sendo realizadas pesquisas em praticamente todos os anos a partir de 2001. Isso se deve ao fato da importância que os Documentos de Requisitos têm em todo o processo de desenvolvimento de software.

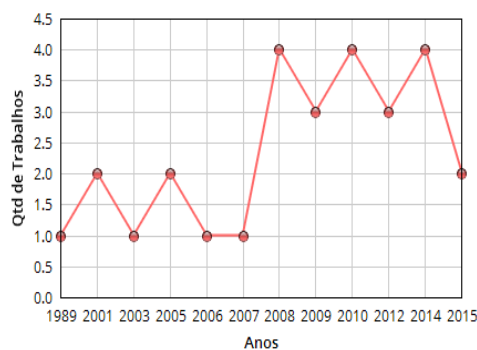


Fig. 3. Quantidade de Trabalhos Selecionados por ano de Publicação.

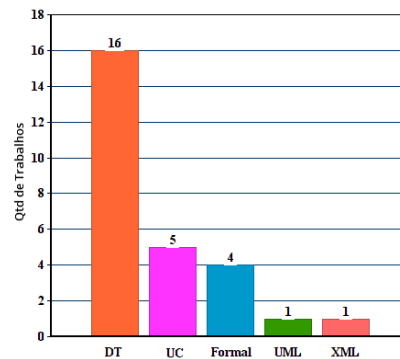


Fig. 4. Formas de Especificação.

### 3.2 QP1 – Como os requisitos são especificados nas abordagens?

O objetivo desta questão é identificar a forma como os requisitos são especificados, tais como Modelos de Casos de Uso, Ponto de Vista, Descrição Textual, nas abordagens focadas na melhoria de documentos de requisitos. A distribuição dos estudos dentro destas categorias é apresentada na Figura 4.



Pode-se verificar que 59,25% dos trabalhos analisados foram especificados utilizando Descrição Textual, tendo em vista esta ser a forma mais utilizada para especificação de requisitos funcionais [31]. Em seguida, tem-se a utilização de Modelos de Casos de Uso com 18,51%, valor relativamente baixo, dado a grande utilização desta técnica nos dias de hoje. Uma possível causa deste valor está relacionada com a *String* de busca utilizada. A inclusão do termo “Caso de Uso” possibilitaria encontrar mais resultados relevantes sobre esta técnica.

Em seguida, as especificações formais totalizaram 14,81%. Este valor, próximo da especificação por Casos de Uso, deve-se ao fato da formalização facilitar a checagem de problemas em Documentos de Requisitos de forma automática, uma vez que utilizam notações matemáticas para especificá-los.

### 3.3 QP2 - Quais abordagens são apoiadas por ferramentas de automatização?

A resposta para esta questão permite identificar quais das abordagens tem alguma ferramenta que as apoiam. É importante investigar como essas ferramentas são utilizadas e o grau de automatização das mesmas. A Tabela 6 apresenta como os trabalhos foram classificados de acordo com o grau de automatização que a ferramenta possibilita.

**Tabela 6** - Ferramentas para automatização.

Tipo	Estudos	Quantidade	%
Automática	T03 T04, T06, T07, T09, T10, T11, T12, T18, T19, T20, T21, T25, T26	14	52%
Semi-automática	T13, T16, T23, T24, T27	5	19%
Manual	T01, T02, T05, T08, T14, T15, T17, T22	8	29%

Por automatização semi-automática, entende-se que algum grau de intervenção humana é utilizado no processo de melhoria, ou seja, alguma avaliação humana deve ser feita para que se obtenha algum resultado. Grande parte da automatização proposta pelos trabalhos é focada na detecção de problemas na especificação de requisitos, principalmente por meio da análise léxica e sintática das descrições textuais. Estes valores podem diferenciar bastante caso a busca fosse realizada em trabalhos de outras línguas diferentes do inglês, tendo em vista as características linguísticas de cada idioma.

É importante frisar que alguns trabalhos sugerem a utilização de mais de uma técnica para a detecção e correção de problemas em Documentos de Requisitos, como é o caso de T2, o qual combina, de forma sistemática, diretrizes para criação de Casos de Uso (prevenção), inspeções de Casos de Uso (detecção) e Simulação (problemas de consistência entre Casos de Uso). O estudo sugere ainda que a abordagem seja aplicada apenas para os Casos de Usos mais críticos, tendo em vista o esforço adicional necessário para aplicar as três técnicas.

### 3.4 QP3 – Existem transformações aplicadas a Documentos de Requisitos visando à melhoria destes?

O objetivo dessa questão é identificar as diferentes formas de melhoria nos documentos de requisitos, tais como: *Patterns*, Diretrizes (*Guidelines*), Refatorações, *Checklists*. Ela ajudará a identificar qual transformação de melhoria necessita de mais esforço e experiência do responsável pela melhoria de documentos de requisitos. A Tabela 7 apresenta essas informações.

**Tabela 7 - Melhorias propostas.**

Melhoria	Estudos	Quantidade	%
<i>Checklist</i>	T01	1	4,34%
Diretrizes	T02, T05, T08, T17	4	14,81%
<i>Patterns / AntiPatterns</i>	T11, T19	2	8,69%
Refatorações	T15	1	4,34%
Regras	T22	1	4,34%
Não Aplicável	T03, T04, T06, T07, T09, T10, T12, T13, T14, T16, T18, T20, T21, T23, T24, T25, T26, T27	14	51,85%

Pode-se verificar baseado no resultado encontrado, que a maioria das propostas, 51,85% dos estudos, não propõe um mecanismo de melhoria. Isso se deve ao fato desses trabalhos terem como proposta uma ferramenta para detecção automática de problemas na especificação dos requisitos.

Apesar de T11 e T19 terem um autor em comum, o que acarretaria na exclusão de algum deles de acordo com o critério de exclusão, eles foram mantidos na condução do mapeamento, tendo em vista que as propostas dos trabalhos são diferentes. Enquanto T11 propõe a definição de *Antipatterns*, a partir de um conjunto de regras pré-definidas que são aplicáveis para qualquer modelo de Caso de Uso, T19 propõe uma abordagem para transformação de modelos a partir da detecção de *Antipattern* e Refatoração. T19 lista um conjunto de 14 *Antipatterns* e 21 Refatorações. Essas Refatorações são as possíveis soluções para os *Antipatterns* detectados.

### 3.5 Resultado da Avaliação da Qualidade

A avaliação da qualidade dos estudos selecionados é útil para aumentar a precisão dos resultados dos dados extraídos. A Tabela 8 apresenta os valores obtidos baseado nas questões da avaliação da qualidade definidas na Tabela 5.

**Tabela 8 - Resultado da avaliação da qualidade dos estudos selecionados, adaptado de [30].**

ID	Q1	Q2	Q3	Q4	Q5	Total	ID	Q1	Q2	Q3	Q4	Q5	Total
<b>T01</b>	1	0	1	1	1	4	<b>T15</b>	1	0	1	1	1	4
<b>T02</b>	1	0	1	1	1	4	<b>T16</b>	1	1	1	1	1	5
<b>T03</b>	1	1	1	1	1	5	<b>T17</b>	1	0	0.5	0.5	1	3

<b>T04</b>	1	1	1	1	1	5	<b>T18</b>	1	1	0	1	1	4
<b>T05</b>	1	0	1	1	1	4	<b>T19</b>	1	1	1	1	1	5
<b>T06</b>	1	1	0.5	1	1	4.5	<b>T20</b>	1	1	1	0.5	0.5	4
<b>T07</b>	1	1	0.5	1	1	4.5	<b>T21</b>	1	1	1	0.5	1	4.5
<b>T08</b>	1	0	1	1	1	4	<b>T22</b>	1	0	1	1	1	4
<b>T09</b>	1	1	1	1	1	5	<b>T23</b>	1	0.5	1	1	1	4.5
<b>T10</b>	1	1	1	1	1	5	<b>T24</b>	1	1	0.5	1	1	4.5
<b>T11</b>	1	1	1	1	1	5	<b>T25</b>	1	1	1	1	1	5
<b>T12</b>	1	1	1	1	1	5	<b>T26</b>	1	1	0	1	1	4
<b>T13</b>	1	1	0.5	0.5	1	4	<b>T27</b>	1	1	1	1	1	5
<b>T14</b>	1	0	1	0	1	3							
			Q1	Q2	Q3	Q4	Q5	Total					
<b>Pontuação</b>			1	0.69	0.82	0.89	0.98	4.38					
<b>Média</b>													

Nenhuma pontuação ficou abaixo de 3, porém os trabalhos T14 e T17 necessitaram de uma nova análise baseado nos critérios de inclusão e exclusão, a fim de se garantir que a seleção dos estudos foi realizada de forma correta. A média geral da pontuação ficou em 4.38, o que é um valor aceitável para este mapeamento sistemático.

#### 4 Validade da Pesquisa

Apesar da grande quantidade de artigos retornados pela pesquisa automática nas bases de dados, diversos trabalhos foram excluídos da fase de Extração de Dados devido à impossibilidade de acessá-los. Esses trabalhos poderiam contribuir para este mapeamento, permitindo uma conclusão mais precisa do mesmo.

Outro fator a ser considerado é a diferenciação entre avaliação e melhoria de documentos de requisitos. Enquanto o primeiro está focado em detectar possíveis problemas na especificação, o segundo se preocupa nas possíveis técnicas para corrigir esses erros. Com isso, alguns trabalhos não faziam muito bem essa diferenciação, o que pode ter ocasionado o descarte de muitos artigos durante a fase de Seleção.

Vários estudos secundários: revisões sistemáticas, *Surveys* dentre outros; já foram conduzidos [29;30;33], os quais podem trazer maiores detalhes para os resultados alcançados nesta pesquisa. Porém, os mesmos foram descartados de acordo com os critérios de exclusão do protocolo utilizado, tendo em vista a necessidade de se obter informações específicas de cada estudo primário.

#### 5 Trabalhos Relacionados

Outros estudos secundários [29;30;33] também foram realizados, a fim de verificar ambiguidade, inconsistência, incompletude, omissões, dentre outros erros em documentos de requisitos em determinadas formas de especificação de requisitos.

Tiwari e Gupta [29] examinam estudos que tratam da evolução dos casos de uso, suas aplicações, avaliações de qualidade, questões abertas, e as direções futuras.

Foram analisados 119 artigos, verificando-se que cerca de vinte modelos de casos de uso foram propostos e aplicados a vários problemas de especificação de software tanto para descrições informais até as notações formais.

Ding et. al [30] buscam entender como as abordagens baseadas no conhecimento podem ser empregadas para melhorar a qualidade do Documento de Software. Foram selecionados e analisados 60 artigos, dos quais doze atributos de qualidade de documentos de software e nove categorias de benefícios do uso de abordagens baseadas no conhecimento em documentação de software foram identificados.

Saavedra et. al [33] apresentam um conjunto de atributos de qualidade normalmente encontrados em documentos de requisitos e quais trabalhos propõem diretrizes e modelos para detectar problemas nesses atributos.

A pesquisa aqui apresentada não focou em um formato específico de documento de requisito, nem em um conjunto limitado de atributos de qualidade. Buscou-se verificar, de forma abrangente, o estado atual de estudos focados na melhoria de documentos de requisitos e na existência de ferramenta de apoio para as abordagens existentes.

## 6 Conclusões e Trabalhos Futuros

O estudo permitiu observar que existem diversas pesquisas focadas na avaliação e melhoria dos Documentos de Requisitos. Ressalta-se que a maioria dos estudos selecionados está focada na detecção dos problemas existentes nos requisitos especificados, sendo essas avaliações realizadas automática, semi-automática e manualmente. Estas últimas por meio de técnicas bastante conhecidas, tais como Inspeções, Revisões e Técnicas de Leituras.

O mapeamento permitiu verificar, também, que todas as ferramentas propostas nos estudos selecionados estão focadas diretamente na detecção dos erros existentes no Documento de Requisitos, ou seja, não foi verificada qualquer ferramenta que apoiasse um processo de forma automatizada.

Como trabalho futuro, pretende-se realizar uma análise mais crítica dos resultados encontrados, buscando fazer comparações entre as abordagens, a fim de identificar aspectos positivos e negativos em cada uma delas. Pretende-se, ainda, desenvolver uma ferramenta para apoiar a execução da AIRDoc [13], possibilitando definir e otimizar o processo de forma efetiva. Esta ferramenta auxiliará engenheiros de software a avaliar a qualidade dos documentos de requisitos e propiciará a construção de softwares com menos erros e conseqüentemente, softwares construídos em menor tempo e mais baratos de serem mantidos. Por fim, será realizado um estudo qualitativo dos resultados encontrados, a fim de adicionar novas refatorações ao catálogo proposto pela AIRDoc.

## Referências

- [1] V. Castañeda, L. Ballejos, and M. L. Caliusco, "Improving the Quality of Software Requirements Specifications with Semantic Web Technologies," 2008.
- [2] D. Cordes and D. Carver, "Evaluation method for user requirements documents," *Inf. Softw. Technol.*, vol. 31, no. 4, pp. 181–188, 1989.

- [3] C. Denger, B. Paech, and B. Freimut, "Achieving high quality of use-case-based requirements," *Inform. - Forsch. und Entwicklung*, vol. 20, no. 1–2, pp. 11–23, 2005.
- [4] A. R. M. A. Diaz-Pace, "Identifying duplicate functionality in textual use cases by aligning semantic actions," *Softw. Syst. Model.*, vol. NA, 2014.
- [5] Y. A. K. El-Attar, "Using model transformation to refactor use case models based on antipatterns," *Inf. Syst. Front.*, vol. NA, 2014.
- [6] B. Gleich, O. Creighton, and L. Kof, "Ambiguity detection: Towards a tool explaining ambiguity sources," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 6182 LNCS, pp. 218–232, 2010.
- [7] C. Huertas and R. Juárez-Ramírez, "NLARE, a natural language processing tool for automatic requirements evaluation," *Proc. CUBE Int. Inf. Technol. Conf. - CUBE '12*, p. 371, 2012.
- [8] E. Knauss, K. Schneider, "Supporting Learning Organisations in Writing Better Requirements Documents Based on Heuristic Critiques," *REFSQ 2012*: 165-171., 2012.
- [9] M. E.-A. Miller, "Improving the quality of use case models using antipatterns," *Softw. Syst. Model.*, vol. 9, no. 2, 2010.
- [10] S. Overhage, O. Skroch, and K. Turowski, "A Method to Evaluate the Suitability of Requirements Specifications for Offshore Projects," *Bus. Inf. Syst. Eng.*, vol. 2, pp. 155–164, 2010.
- [11] E. Parra, C. Dimou, J. Llorens, V. Moreno, and A. Fraga, "A methodology for the classification of quality of requirements using machine learning techniques," *Inf. Softw. Technol.*, vol. 67, pp. 180–195, 2015.
- [12] D. Popescu, S. Rugaber, N. Medvidovic, and D. M. Berry, "Reducing Ambiguities in Requirements Specifications Via Automatically Created Object-Oriented Models," *Innov. Requir. Anal. From Stakeholders' Needs to Form. Des.*, vol. 1, pp. 103–124, 2008.
- [13] R. Ramos, J. Castro, F. Alencar, J. Araújo, A. Moreira, and R. Penteado, "Quality improvement for use case model," *SBES 2009 - 23rd Brazilian Symp. Softw. Eng.*, pp. 187–195, 2009.
- [14] A. K. Sakkinen, "Requirements quality control: a unifying framework," *Requir. Eng.*, vol. 11, no. 1, 2006.
- [15] G. C. Sampaio, "Automated formal specification generation and refinement from requirement documents," *J. Brazilian Comput. Soc.*, vol. 14, no. 1, 2008.
- [16] T. Simon, J. Streit, and M. Pizka, "Practically Relevant Quality Criteria for Requirements Documents," *Itestra.De*, vol. 2, pp. 115–121, 2008.
- [17] S. Trudel and A. Abran, "Functional Requirement Improvements through Size Measurement: A Case Study with Inexperienced Measurers," *2010 Eighth ACIS Int. Conf. Softw. Eng. Res. Manag. Appl.*, pp. 181–189, 2010.
- [18] P. C. Vêras, E. Villani, A. M. Ambrosio, M. Vieira, and H. Madeira, "A benchmarking process to assess software requirements documentation for space applications," *J. Syst. Softw.*, vol. 100, pp. 103–116, 2015.
- [19] K. Verma, A. Kass, and R. Vasquez, "Using syntactic and semantic analyses to improve the quality of requirements documentation," *Semant. Web*, vol. 5, pp. 405–419, 2014.
- [20] D. Zowghi and V. Gervasi, "On the interplay between consistency, completeness, and correctness in requirements evolution," *Inf. Softw. Technol.*, vol. 45, no. 14, pp. 993–1009, 2003.
- [21] N. Kiyavitskaya, N. Zeni, L. Mich, and D. M. Berry, "Requirements for tools for ambiguity identification and measurement in natural language requirements specifications," *Requir. Eng.*, vol. 13, pp. 207–239, 2008.

- [22] A. Fantechi and E. Spinicci, "A content analysis technique for inconsistency detection in software requirements documents," *Proc. VIII Work. Requir. Eng.*, pp. 245–256, 2005.
- [23] S. Tjong and D. Berry, "Can Rules of Inferences Resolve Coordination Ambiguity in Natural Language Requirements Specification?," *Wer*, pp. 205–210, 2008.
- [24] P. Kroha, R. Janetzko, and J. E. Labra, "Ontologies in checking for inconsistency of requirements specification," *3rd Int. Conf. Adv. Semant. Process. - SEMAPRO 2009*, pp. 32–37, 2009.
- [25] F. Fabbrini, M. Fusani, S. Gnesi, and G. Lami, "An Automatic Quality Evaluation for Natural Language Requirements," *REFSQ 2001 Proc. Seventh Int. Work. RE Found. Softw. Qual.*, vol. 1, pp. 150–164, 2001.
- [26] A. Durán, B. Bernárdez, A. Ruiz, and M. Toro, "An XML – based Approach for the Automatic Verification of Software Requirements Specifications," *Wer01*, 2001.
- [27] H. Femmer, D. M. Fernández, E. Juergens, M. Klose, I. Zimmer, and J. Zimmer, "Rapid Requirements Checks with Requirements Smells: Two Case Studies," *Proc. 1st Int. Work. Rapid Contin. Softw. Eng.*, pp. 10–19, 2014.
- [28] B. Kitchenham, *Procedures for Performing Systematic Reviews*, Keele, Reino Unido, Keele University 33, 2004.
- [29] S. Tiwari and A. Gupta, "A systematic literature review of use case specifications research," *Inf. Softw. Technol.*, vol. 67, pp. 128–158, 2015.
- [30] W. Ding, P. Liang, A. Tang, and H. Van Vliet, "Knowledge-based approaches in software documentation: A systematic literature review," *Inf. Softw. Technol.*, vol. 56, no. 6, pp. 545–567, 2014.
- [31] Denger, C., Berry, D. M., Kamsties, E., 2003. Higher quality requirements specifications through natural language patterns. In: *Proceedings of the IEEE International Conference on Software—Science, Technology & Engineering (SwSTE'03)*, pp. 1–11.
- [32] I. Jacobson, *Object-oriented development in an industrial environment*, in: *Proceedings of the Conference on Object-Oriented Programming, Systems, Languages & Applications*, ACM, 1987, pp. 183–191.
- [33] R. Saavedra, L. Ballejos, and M. Ale, "Quality Properties Evaluation for Software Requirements Specifications: An Exploratory Analysis," *Wer. Inf. Puc-Rio. Br.*, pp. 1–14.
- [34] A. Cockburn, *Writing Effective Use Cases*, vol. 1, Addison-Wesley, Boston, 2001.
- [35] T. Dingsøy and T. Dyba, "Empirical studies of agile software development: A systematic review," *Inf. Softw. Technol.*, vol. 50, pp. 833–859, 2008.
- [36] B. Anda, D. Sjøberg, M. Jørgensen, *Quality and understandability of use case models*, in: J. Knudsen (Ed.), *ECOOP 2001 Object-Oriented Programming, Lecture Notes in Computer Science*, vol. 2072, Springer, Berlin Heidelberg, 2001, pp. 402–428.
- [37] Boehm BW (1984) *Verifying and validating software requirements and design specifications*. *IEEE Software* 1(1):75–88.
- [38] WIEGERS, K. E.: "Software Requirements", Microsoft Press, Second Edition. (2003).
- [39] U. S. Shah and D. C. Jinwala, "Resolving Ambiguities in Natural Language Software Requirements," *ACM SIGSOFT Softw. Eng. Notes*, vol. 40, no. 5, pp. 1–7, 2015.