



UNIVERSIDADE FEDERAL DE SANTA CATARINA
CENTRO TECNOLÓGICO
PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO

Viviane Duarte Bonfim

PracticesforReD

Abordagem de recomendação de práticas para mitigar dívidas de requisitos em organizações ágeis

FLORIANÓPOLIS - SC

2023

Viviane Duarte Bonfim

PracticesforReD

Abordagem de recomendação de práticas para mitigar dívidas de requisitos em organizações ágeis

Tese submetida ao Programa de Pós-Graduação em Ciência da Computação da Universidade Federal de Santa Catarina como requisito parcial para a obtenção do título de Doutora em Ciência da Computação.

Orientador(a): Prof.(a) Dr.(a) Fabiane Vavassori Benitti.

FLORIANÓPOLIS - SC

2023

RESUMO

[Contexto] As metodologias ágeis surgiram com a proposta de agilizar o desenvolvimento, reorganizando algumas práticas para atender essa necessidade, o que pode comprometer as áreas do ciclo de desenvolvimento, como exemplo, a Engenharia de Requisitos (ER). [Problema] As atividades da ER realizadas inadequadamente podem representar problemas como requisitos mal compreendidos, omitidos, mal especificados, o que denotam ser exemplos de dívidas técnicas e podem afetar outras etapas do desenvolvimento, assim como, ameaçar a qualidade do software. No entanto, as dívidas voltadas à ER, denominadas dívidas de requisitos, são pouco exploradas e, em muitas ocasiões, são desconhecidas, não são monitoradas ou geridas de maneira eficiente, o que provoca um alto custo de manutenção ao longo do tempo. Sobretudo, se as organizações compreendessem melhor seu cenário atual, identificando as suas dívidas de requisitos e as causas dessas dívidas, poderiam adaptar práticas para mitigá-las. [Objetivo] O objetivo do trabalho consiste em propor uma abordagem que permite às organizações ágeis compreenderem o seu cenário atual voltado à existência de dívidas de requisitos, identificando as lacunas que caracterizam essas dívidas, as suas causas e, a partir desse cenário, recomendar um conjunto de práticas e orientações para mitigá-las e preveni-las. [Metodologia] A pesquisa foi conduzida inicialmente por meio de uma investigação do estado da prática apoiada pelo método qualitativo de análise de dados *grounded theory*. O resultado do estado da prática foi unificado com os resultados obtidos em dois estudos sistemáticos da literatura para contemplar o estado da arte: um mapeamento sistemático da literatura que consiste em identificar as causas, consequências e práticas voltadas às dívidas de requisitos e um estudo terciário focado na gestão da dívida. O resultado da unificação do estado da prática e do estado da arte embasou a construção de uma ontologia, denominada “OntoReD”, direcionada às dívidas de requisitos e atua como o “kernel” da abordagem proposta. A abordagem, intitulada “PracticesforReD”, consiste em uma sequência de passos que consome as informações da ontologia, permitindo ao profissional, compreender seu cenário atual sobre as dívidas de requisitos, recomendando práticas para mitigá-las. [Resultado] Como resultado, propõe-se uma abordagem que permitirá aos profissionais das organizações ágeis, identificarem as suas dívidas de requisitos que remetem ao seu cenário atual, visualizarem as causas dessas dívidas e a recomendação e orientação de práticas que poderão mitigá-las e preveni-las. Os profissionais poderão acessar a PracticesforReD por meio de uma solução computacional que será a interface de apoio entre as organizações e a sequência de passos previstos na abordagem.

Palavras-chave: Engenharia de requisitos. Dívida de requisitos. Práticas para mitigar dívidas de requisitos.

ABSTRACT

[Context] Agile methodologies emerged with the proposal to expedite development, reorganizing some practices to meet this need, which can compromise areas of the development cycle, such as Requirements Engineering (RE). [Problem] Inadequately conducted RE activities can pose problems such as misunderstood, omitted, or poorly specified requirements, which are examples of technical debts and can impact other development stages, as well as threaten software quality. However, debts related to RE, called requirements debts, are underexplored and, in many cases, unknown, not monitored, or efficiently managed, leading to high maintenance costs over time. Above all, if organizations better understood their current scenario, identifying their requirements debts and the causes thereof, they could adapt practices to mitigate them. [Objective] The objective of this work is to propose an approach that enables agile organizations to understand their current scenario regarding the existence of requirements debts, identifying the gaps that characterize these debts, their causes, and, based on this scenario, recommend a set of practices and guidelines to mitigate and prevent them. [Methodology] The research was initially conducted through a state-of-the-practice investigation supported by the qualitative data analysis method grounded theory. The result of the state-of-the-practice was unified with the results obtained in two systematic literature reviews to encompass the state-of-the-art: a systematic literature mapping that identifies the causes, consequences, and practices related to requirements debts, and a tertiary study focused on debt management. The result of unifying the state-of-the-practice and the state-of-the-art supported the construction of an ontology, called "OntoReD," focused on requirements debts and acts as the "kernel" of the proposed approach. The approach, titled "PracticesforReD," consists of a sequence of steps that consumes information from the ontology, allowing the professional to understand their current scenario regarding requirements debts, recommending practices to mitigate them. [Result] As a result, an approach is proposed that will allow professionals in agile organizations to identify their requirements debts that refer to their current scenario, visualize the causes of these debts, and the recommendation and guidance of practices that can mitigate and prevent them. Professionals will be able to access PracticesforReD through a computational solution that will be the support interface between organizations and the sequence of steps outlined in the approach.

Keywords: Requirements Engineering; Requirements Debt; Practices to mitigate requirements debts.

LISTA DE FIGURAS

Figura 1 – Procedimentos metodológicos da pesquisa	16
Figura 2 – Estrutura da pesquisa	26
Figura 3 – Modelo teórico	31
Figura 4 – Etapas do Mapeamento Sistemático da Literatura	36
Figura 5 – Seleção dos estudos	38
Figura 6 – Causas que geram dívida de requisitos	42
Figura 7 – Consequências que caracterizam dívida de requisitos	43
Figura 8 – Práticas para minimizar as causas e dívidas de Requisitos	43
Figura 9 – Etapas do estudo terciário	49
Figura 10 – Seleção dos estudos	52
Figura 11 – Estrutura dos conceitos da OntoReD	62
Figura 12 – Hierarquia das classes da OntoReD	63
Figura 13 – Classes implementadas no Protégé e exemplos de propriedades	64
Figura 14 – Data property das instâncias	65
Figura 15 – Exemplos de Instâncias do conceito “CausaReD” da OntoReD	66
Figura 16 – Resultado Consulta 1: Consequências da dívida de requisitos mitigadas pela prática “Treinamento de padrões de elicitação de requisitos”	68
Figura 17 – Proposta da abordagem	70
Figura 18 – Protótipo da solução para o passo 2	74
Figura 19 – Protótipo da solução para o passo 2 - visualizar o cenário atual	75
Figura 20 – Protótipo da solução para o passo 3	75
Figura 21 – Protótipo da solução para o passo 4	76
Figura 22 – Protótipo da solução para o passo 5	77

LISTA DE QUADROS

Quadro 1 – Classificação da DT conforme o Tipo	23
Quadro 2 – Lista de códigos: Codificação inicial e incidência	29
Quadro 3 – Questões de pesquisa do MSL	36
Quadro 4 – Termos de pesquisa	37
Quadro 5 – String genérica	37
Quadro 6 – Critérios de inclusão e exclusão	37
Quadro 7 – Estudos incluídos no MSL	39
Quadro 8 – Questões de pesquisa do Estudo Terciário	50
Quadro 9 – Termos de pesquisa	50
Quadro 10 – String genérica	51
Quadro 11 – Critérios de inclusão e exclusão	51
Quadro 12 – Estudos incluídos no estudo terciário	53
Quadro 13 – Avaliação de qualidade dos estudos incluídos	53
Quadro 14 – Especificação da OntoReD	61
Quadro 15 – Axioma voltado às práticas que podem mitigar as causas das dívidas de requisitos	65
Quadro 16 – Exemplos de Instâncias da OntoReD	66
Quadro 17 – Exemplo de consulta da OntoReD em DL Query	67
Quadro 18 – Cronograma da pesquisa para a tese de Doutorado	80

LISTA DE ABREVIATURAS E SIGLAS

DS	Design Science
DT	Dívida Técnica
ER	Engenharia de Requisitos
GT	<i>Grounded Theory</i>
MSL	Mapeamento Sistemático da Literatura
ReD	Dívida de Requisitos
TDM	Gestão da Dívida Técnica

SUMÁRIO

INTRODUÇÃO	11
1.1 PROBLEMA	12
1.2 DELIMITAÇÃO DO ESCOPO	13
1.3 JUSTIFICATIVA	13
1.4 OBJETIVOS	14
1.4.1 Objetivo geral	14
1.4.2 Objetivos específicos	14
1.5 CONTRIBUIÇÕES DA PESQUISA	15
1.6 METODOLOGIA DA TESE	15
1.6.1 Procedimentos metodológicos	16
1.7 ESTRUTURA DO TRABALHO	19
2 REFERENCIAL TEÓRICO	20
2.1 ENGENHARIA DE REQUISITOS ÁGIL	20
2.1.1 Características da Engenharia de Requisitos Ágil	20
2.2 DÍVIDA TÉCNICA NO DESENVOLVIMENTO DE SOFTWARE	21
2.2.1 Classificação da dívida técnica conforme a dimensão	22
2.2.1.1 <i>Dívida técnica intencional</i>	22
2.2.1.2 <i>Dívida Não intencional</i>	22
2.2.2 <i>Classificação da dívida técnica conforme o tipo</i>	22
2.2.3 Gestão da dívida técnica	23
3 DÍVIDA TÉCNICA E ENGENHARIA DE REQUISITOS: ESTADO DA PRÁTICA	25
3.1 CARACTERIZAÇÃO DA PESQUISA	25
3.2 COLETA DE DADOS	26
3.2.1 Perfil dos participantes: Grupo amostral	27
3.3 ANÁLISE DOS DADOS	28
3.3.1 Codificação inicial	28
3.3.2 Codificação focalizada	29
3.4 CONSTRUÇÃO E APRESENTAÇÃO DO MODELO TEÓRICO PROPOSTO	30
3.5 VALIDAÇÃO DO MODELO TEÓRICO	31
3.5.1 Preparação da validação do Modelo Teórico	32
3.5.2 Realização da validação do Modelo Teórico	32
3.5.3 Resultado da validação do Modelo teórico	32
3.6 CONTRIBUIÇÕES	33
4 ESTADO DA ARTE: ESTUDOS SISTEMÁTICOS	35
4.1 MAPEAMENTO SISTEMÁTICO DA LITERATURA	35
4.1.1 Planejamento	36
4.1.1.1 Objetivo e Questões de Pesquisa	36
4.1.1.2 <i>String de busca e fontes de pesquisa</i>	36
4.1.1.3 <i>Critérios de inclusão e exclusão</i>	37

4.1.2 Execução	38
4.1.2.1 <i>Seleção de dados</i>	38
4.1.2.2 <i>Extração de dados</i>	39
4.1.3 Resultados	39
4.1.3.1 <i>Quais estratégias as organizações ágeis utilizam para identificar dívida de requisitos conforme as etapas da Engenharia de Requisitos em organizações ágeis?</i>	39
4.1.3.1.1 Familiaridade dos profissionais com conceito de Dívida Técnica	39
4.1.3.1.2 Como as organizações ágeis caracterizam Dívida de Requisitos	40
4.1.3.1.3 Estratégias para identificar/evitar dívida de requisitos	41
4.1.3.2 <i>QP 2. Quais as causas na engenharia de requisitos encontrados em organizações ágeis e que geram dívidas de requisitos/dívidas técnicas?</i>	42
4.1.3.3 <i>QP 2.1 Quais as consequências ocasionadas pelas dívidas de requisitos na Engenharia de Requisitos?</i>	42
4.1.3.4 <i>QP 3. Quais metodologias e/ou práticas são empregadas em organizações ágeis que podem minimizar as dívidas de requisitos geradas durante a Engenharia de Requisitos?</i>	43
4.1.3.5 <i>QP 4. Como as organizações ágeis monitoram suas dívidas de requisitos?</i>	44
4.1.4 Discussões	44
4.1.5 Unificação do MSL e Visão Prática - causas das dívidas de requisitos, consequências das dívidas de requisitos e práticas	46
4.1.5.1 <i>Cenários dos estudos</i>	46
4.1.5.2 <i>Visão prática (cenário 1) e MSL (cenário 2): Unificação das causas, consequências e práticas voltadas às dívidas de requisitos</i>	47
4.1.5.3 <i>Apresentação dos resultados: consequências (dívidas de requisitos) versus causas que geram essas dívidas de requisitos - unificação entre MSL e Visão Prática - unificação entre MSL e Visão Prática</i>	47
4.1.5.4 <i>Apresentação dos resultados: causas versus práticas voltadas à dívida de requisitos - unificação entre MSL e Visão Prática</i>	48
4.1.6 Contribuições	48
4.2 GESTÃO DA DÍVIDA TÉCNICA: ESTUDO TERCIÁRIO	49
4.2.1 Planejamento	50
4.2.1.1 <i>Objetivo e questões de pesquisa</i>	50
4.2.1.2 <i>String de busca e fontes de pesquisa</i>	50
4.2.1.3 <i>Mecanismos de busca</i>	51
4.2.1.4 <i>Critérios de inclusão e exclusão</i>	51
4.2.2 Execução	52
4.2.2.1 <i>Seleção dos estudos</i>	52
4.2.3 Avaliação de Qualidade	53
4.2.4 Resultados	53
4.2.5 Análise dos dados e discussões	54
4.3 LIMITAÇÕES E AMEAÇAS À VALIDADE	55
4.4 UNIFICAÇÃO DAS ÁREAS DA GESTÃO DA DÍVIDA (ESTUDO TERCIÁRIO) COM O RESULTADO DA UNIFICAÇÃO DAS CAUSAS, DÍVIDAS DE REQUISITOS E PRÁTICAS DO MSL E VISÃO PRÁTICA	56

4.5 CONTRIBUIÇÕES	57
5 PROPOSTA DE UMA ONTOLOGIA DE DOMÍNIO VOLTADA ÀS DÍVIDAS DE REQUISITOS	59
5.1 APRESENTAÇÃO DA ONTORED	60
5.2 CONSTRUÇÃO DA ONTORED BASEADA NA METHONTOLOGY	60
5.2.1 Especificação	61
5.2.2 Conceitualização	61
5.2.3 Formalização e implementação	63
5.2.3.1 <i>Construção das classes</i>	63
5.2.3.2 <i>Definição dos Relacionamentos da OntoReD – Object Properties</i>	64
5.2.3.3 <i>Definição dos atributos de instância da OntoReD – Data Properties</i>	65
5.2.3.4 <i>Axiomas</i>	65
5.2.3.5 <i>Definição das instâncias dos conceitos da OntoReD</i>	66
5.2.4 Avaliação da OntoReD	67
5.2.5 Documentação da OntoReD	68
5.3 CONTRIBUIÇÃO	68
6 PRACTICESFORRED - PROPOSTA DE UMA ABORDAGEM VOLTADA À RECOMENDAÇÃO DE PRÁTICAS PARA REDUÇÃO DAS DÍVIDAS DE REQUISITOS EM ORGANIZAÇÕES ÁGEIS	70
6.1 ESTRUTURA DA PRACTICESFORRED	70
6.1.1 Passo 1 - Analisar o seu conhecimento e clareza sobre dívida de requisitos do produto	71
6.1.2.1 Passo 1.2 - Compreender e identificar as dívidas de requisitos dos seus produtos	71
6.1.2 Passo 2 - Informar o(s) gaps - problema(s) observado(s) no cenário atual voltado ao processo de requisitos	71
6.1.3 Passo 3 - Sinalizar as causas dos gaps informados anteriormente	72
6.1.4 Passo 4 - Visualizar e compreender as práticas sugeridas pela Ontologia para mitigar as dívidas de requisitos	73
6.1.5 Passo 5 – Elementos que detalham as práticas sugeridas pela Ontologia - OntoReD	73
6.2 SOLUÇÃO COMPUTACIONAL DE APOIO À PRACTICESFORRED	74
7 CONSIDERAÇÕES	78
7.1 ATIVIDADES FUTURAS	79
REFERÊNCIAS	80
APÊNDICES	94

INTRODUÇÃO

A área de desenvolvimento de software passou por algumas transformações devido ao surgimento de novas abordagens de desenvolvimento, como os métodos de desenvolvimento de software ágil durante o manifesto ágil¹, em 2001 (Curcio *et al.*, 2018). Essa abordagem de desenvolvimento foi incentivada pela necessidade de gerenciar melhor os projetos, agilizar o processo de desenvolvimento e acelerar as entregas, proporcionando maior colaboração e comunicação entre os envolvidos, diferenciando-os dos modelos tradicionais de desenvolvimento de software (Elghariani; Kama 2016), o que fortaleceu a sua adoção nas organizações.

Apesar dos benefícios proporcionados pelas abordagens ágeis no desenvolvimento de software, a engenharia de requisitos (ER), no contexto ágil, ainda é considerada um fator preocupante para as organizações pois, embora a ER ágil recomende a documentação e modelagem de requisitos, priorização de requisitos e envolvimento dos stakeholders, essas atividades são informais e ocorrem separadamente, o que dificulta a sua compreensão e aplicação (Saleh; Baharom; Mohamed, 2021).

As atividades da ER quando realizadas de forma inadequada causam falhas no desenvolvimento de software, fortalecendo o surgimento de dívidas técnicas, neste caso, dívidas de requisitos (Melo *et al.*, 2022). No entanto, em algum momento, essas dívidas precisam ser pagas e serão acrescidas de juros no processo de desenvolvimento de software (Freire *et al.*, 2021).

A dívida de requisitos é um tipo de dívida técnica² (DT) que compreende as falhas na especificação de requisitos, caracterizando a distância entre a especificação desejada de requisitos e a implementação real desses requisitos (Li; Avgeriou; Liang, 2015; Guo; Spínola; Seaman, 2016).

Compreender as causas que podem incorrer em dívidas de requisitos e seu impacto no processo de desenvolvimento, poderia auxiliar na mitigação e prevenção de novas dívidas por meio de ações efetivas, porém, essa ainda não é uma prática comum nas organizações, assim como na literatura (Rios *et al.* 2018; Rios *et al.* 2020).

¹ <https://agilemanifesto.org/>

² O termo dívida técnica (DT) foi introduzido na área de desenvolvimento de software por meio de uma metáfora associada à codificação, empregada por Ward Cunningham em 1992, e está relacionada às falhas no software, geralmente ocasionadas por atalhos no desenvolvimento ou aos compromissos assumidos pelos desenvolvedores para atender a uma demanda urgente, convenientes a curto prazo, mas podendo dificultar futuramente a sua manutenção.

Sobretudo, existe uma dificuldade por parte das organizações ágeis de reconhecerem suas dívidas devido ao desconhecimento do conceito de dívida técnica, o que os impede de identificar as causas e os impactos dessas dívidas no seu processo de desenvolvimento de software e, conseqüentemente, definir ações para gerenciá-las, mitigá-las e preveni-las (Behutiye *et al.*, 2017).

Diante desse cenário, observa-se uma carência em pesquisas que auxiliam as organizações ágeis na identificação da existência das dívidas de requisitos, na compreensão dos problemas e das causas dessas dívidas e formas de gerenciá-las e mitigá-las, sendo considerada ainda, uma lacuna a ser preenchida na comunidade científica.

Considerando essa lacuna voltada à dificuldade das organizações ágeis reconhecerem suas dívidas de requisitos e suas causas, esta tese pretende contribuir com uma proposta voltada à mitigação e/ou prevenção da dívida de requisitos, apresentando uma abordagem que recomende um conjunto de práticas e suas orientações para mitigar esse tipo de dívida, a partir do cenário atual da organização.

1.1 PROBLEMA

Desde o manifesto ágil em 2001, a maneira de desenvolver software foi se transformando a partir de princípios e valores que foram conquistando a comunidade de desenvolvimento de software. A partir desse movimento, o cenário de desenvolvimento de software se modificou e atualmente é representado por uma ampla adoção de metodologias ágeis (Rios; Mendonça; Seaman, 2019), conquistando significativamente seu espaço devido sua crescente popularidade e a possibilidade de entregas rápidas e contínuas (Elghariani; Kama, 2016), priorizando a agilidade, flexibilidade (Curcio *et al.*, 2018), e se adaptando às mudanças ao longo do processo de desenvolvimento. Essas mudanças sugerem uma abordagem flexível para o desenvolvimento de software (Schön *et al.*, 2017).

No entanto, mesmo com tantos esforços por parte das organizações e comunidade científica, projetos de software ainda acumulam taxas de fracasso expressivas, incluindo aqueles que utilizam processos ágeis, devido à condução do processo de requisitos (Varalda; Vegas, 2018).

Na ER ágil as atividades previstas na ER tradicional não são sequenciais e não possuem uma clareza sobre elas, pois embora sejam realizadas iterativamente e executadas durante cada um dos ciclos de desenvolvimento, não possuem uma formalidade e a obrigatoriedade das atividades que compõem a ER, sendo apoiadas nas habilidades e conhecimentos dos

colaboradores (Curcio et al. 2018; Dingsøyr *et al.* 2012) e diante do que é estabelecido na na organização. Essa informalidade permite que as atividades da ER sejam realizadas de diferentes maneiras, não realizá-las ou não documentá-las adequadamente (Bjarnason; Wnuk; Regnell, 2011), podendo causar falhas no processo de requisitos ocasionando dívidas de requisitos.

Essas dívidas, dificilmente são reconhecidas ou identificadas. Quando são, as organizações encontram dificuldades em perceber as causas dessas dívidas, o que torna complexo preveni-las e tratá-las, conforme apontam estudos voltados ao entendimento de dívidas técnicas nas organizações (Rios *et al.*, 2020; Mandić; Taušan; Ramač, 2020; Behutiye *et al.*, 2017).

Esses aspectos, reforçam a deficiência das organizações ágeis em compreender as causas e consequências das dívidas de requisitos, para terem condições de estabelecer práticas que permitam gerenciá-las, preveni-las e conseqüentemente, melhorar seu processo de desenvolvimento de software.

Diante do cenário abordado, e considerando o problema relatado, pretende-se, ao final deste estudo, responder às seguintes questões de pesquisa:

QP1. Como auxiliar as organizações ágeis a reconhecerem suas dívidas e as causas que as ocasionam?

QP2. Como recomendar práticas para mitigar as causas e consequências das dívidas de requisitos considerando o cenário atual da organização ágil?

1.2 DELIMITAÇÃO DO ESCOPO

O escopo desta pesquisa limita-se a uma abordagem focada na recomendação de práticas para mitigar dívidas de requisitos, voltadas às atividades da ER, em organizações ágeis.

Não estão previstas neste escopo, recomendação de práticas para outros tipos de dívidas técnicas, assim como as demais fases do processo de desenvolvimento de software e organizações que utilizam metodologias de desenvolvimento tradicionais.

1.3 JUSTIFICATIVA

Estudos demonstram que existe uma carência em trabalhos que exploram a dívida de requisitos (Rios *et al.*, 2020). Essa afirmação foi reforçada pelos estudos presentes no mapeamento sistemático da literatura (capítulo 4), pois demonstram que poucos trabalhos abordam a dívida de requisitos. Aqueles estudos que a mencionam, não tratam esse tipo de

dívida especificamente, apenas dentro do contexto da dívida técnica de modo geral, o que pode contribuir para a falta de compreensão sobre a dívida de requisitos nas organizações ágeis.

No estado na prática (capítulo 3), ao realizar uma pesquisa qualitativa baseada na análise de dados apoiado pela *Grounded Theory* constatou-se, por meio dos relatos dos profissionais das organizações ágeis, que a dívida de requisitos é um termo ainda desconhecido, e que as organizações possuem várias dessas dívidas, mesmo sem percebê-las, informações também confirmadas pela literatura (capítulo 4).

Em ambos os estudos, estado da prática e mapeamento sistemático da literatura, observou-se também, que devido à ausência de compreensão da dívida de requisitos, os profissionais desconhecem a existência dessas dívidas nos seus produtos de software e as suas causas (Rios *et al.*, 2020), o que pode dificultar a mitigação e prevenção dessas dívidas, produzindo efeitos negativos no sucesso dos projetos de software e comprometendo o processo de desenvolvimento e a qualidade do produto de software (Lenarduzzi; Fucci, 2019).

A literatura (capítulo 4), por meio dos estudos que abordam a dívida de requisitos, confirmou que identificar as dívidas de requisitos e as causas que as ocasionam, também é uma lacuna a ser explorada, pois ainda requer maior amadurecimento dos profissionais, para terem condições de reconhecerem suas dívidas e buscarem meios para mitigá-las e evitá-las.

1.4 OBJETIVOS

Os objetivos deste trabalho concentram-se em:

1.4.1 Objetivo geral

O trabalho consiste em propor uma abordagem voltada à recomendação de práticas para mitigar e prevenir as causas e dívidas de requisitos em organizações ágeis.

1.4.2 Objetivos específicos

- a) Investigar o estado da prática e o estado da arte, os elementos que envolvem a dívida de requisitos e a gestão da dívida de requisitos em organizações que adotam desenvolvimento ágil de software e as recomendações para mitigá-las.
- b) Elaborar uma ontologia para centralizar e disponibilizar as informações relacionadas à dívida de requisitos.

- c) Propor uma abordagem que permite às organizações ágeis reconhecerem a existência de dívidas de requisitos, seu cenário atual e recomende práticas para mitigá-las, apoiada pela ontologia.
- d) Desenvolver uma solução computacional que apoie a abordagem proposta.
- e) Avaliar a abordagem voltada à recomendação de práticas para mitigar as dívidas de requisitos e a solução computacional.

1.5 CONTRIBUIÇÕES DA PESQUISA

Destacam-se como as contribuições desta pesquisa:

- I) A construção de uma ontologia de domínio que define uma estrutura, organiza e relaciona as informações, obtidas no estado da prática e no estado da arte. Essas informações abordam a compreensão das causas das dívidas de requisitos, suas consequências e as práticas que poderão mitigá-las considerando a área de gestão da dívida, e consiste, especialmente em apoiar a abordagem proposta (item II). Pretende-se, através da ontologia, disseminar a compreensão sobre dívidas de requisitos, na comunidade acadêmica e nas organizações ágeis para servir de apoio a outros estudos na área, potencializando a pesquisa nesse tipo de dívida, possibilitando a redução das dívidas de requisitos e, conseqüentemente a melhoria do processo de requisitos.
- II) Proposta de uma abordagem que auxilia as organizações ágeis a identificarem a existência da dívida de requisitos no seu produto de software, compreenderem o seu cenário atual refletindo e sinalizando os problemas que caracterizam as dívidas de requisitos, bem como as suas causas e, a partir dessas informações acessar recomendações das práticas que podem reduzi-las,
- III) Recomendação de práticas, seus elementos e orientação para a sua adequação, conforme o seu estado atual.
- IV) Solução computacional que viabiliza a interação entre as organizações ágeis e a abordagem proposta proporcionando uma experiência mais assertiva aos passos previstos na abordagem.

1.6 METODOLOGIA DA TESE

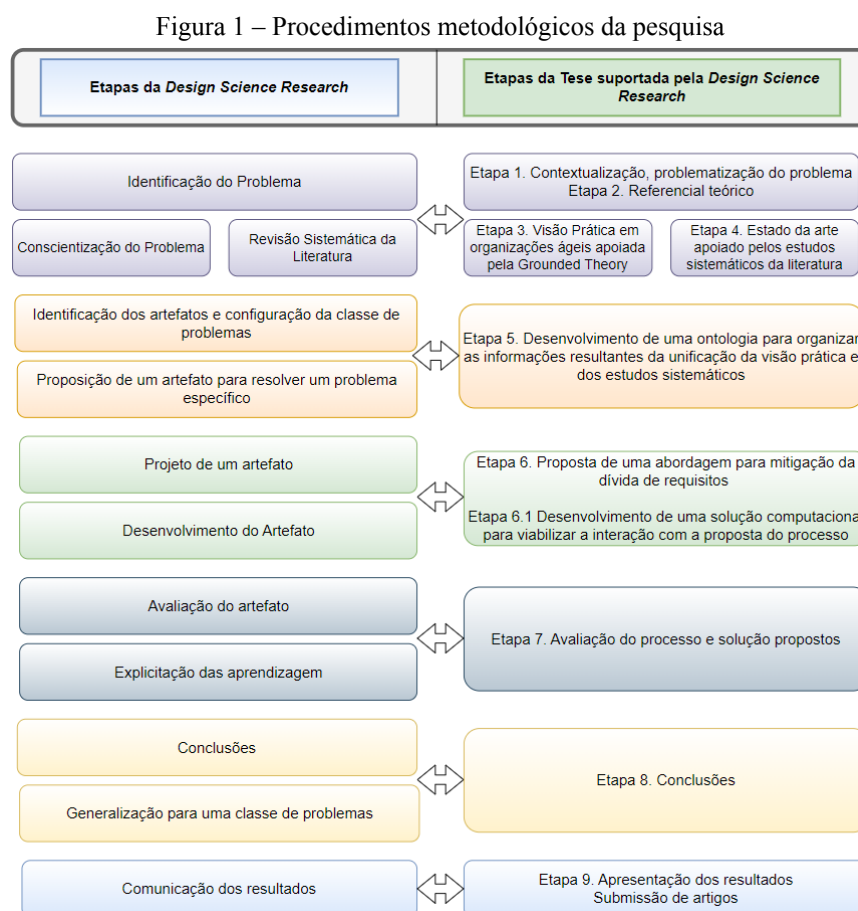
A metodologia do trabalho apresenta os procedimentos metodológicos que sintetizam os passos do desenvolvimento do trabalho, apoiados pela metodologia *design science* (DS), por estar inserido no contexto tecnológico da proposta da tese (Simon, 1996). A *design science research* é o método que atende aos pressupostos da metodologia DS e visa fundamentar e

operacionalizar a condução da pesquisa de um artefato ou uma prescrição para a construção do conhecimento alcançado a partir do entendimento do problema, da construção e avaliação dos artefatos (Dresch; Lacerda; Antunes, 2015).

O processo de *design science research* contempla seis etapas caracterizadas por: identificação e motivação do problema, definição dos objetivos para uma determinada solução, projeto e desenvolvimento, demonstração, avaliação e comunicação (Peffer; Tuunanen; Rothenberger; Chatterjee, 2014), as quais conduzem os procedimentos metodológicos desta tese.

1.6.1 Procedimentos metodológicos

Este estudo é apoiado na proposta sugerida por Dresch; Lacerda; Antunes (2015) para condução de pesquisas que utilizam o método DSR como abordagem metodológica. A Figura 1 demonstra as etapas que conduzem este estudo comparando às etapas da metodologia.



Fonte: Adaptado pelo autor (2021) baseado em Monteiro (2021); Dresch; Lacerda; Antunes (2015)

A Figura 1 evidencia as etapas da pesquisa descritas seguindo as etapas previstas na *Design Science Research* (Dresch; Lacerda; Antunes, 2015).

Etapa 1. Contextualização do problema: Esta etapa evidencia a relevância do estudo destacando o problema, a sua importância para a comunidade científica e organizações e a solução para minimizá-lo por meio de artefatos que contribuirão para alcançar o objetivo.

Etapa 2. Referencial teórico: A etapa 2 é responsável por fundamentar os conceitos envolvidos na pesquisa e descrever como esses conceitos se relacionam e se destacam no estudo, potencializando a delimitação do escopo da pesquisa. Para contemplar esta etapa, apropriou-se de conceitos apoiados pela literatura: Engenharia de requisitos ágil; dívida técnica, especializando-a em dívida de requisitos.

Etapa 3. Visão Prática em organizações ágeis apoiada pela Grounded Theory: A etapa 3 visa observar, por meio de relatos de colaboradores que atuam em organizações ágeis no âmbito nacional e internacional, a relação entre dívida técnica, especificamente a dívida de requisitos, e as atividades da Engenharia de Requisitos conduzidas pelas organizações. Esta etapa foi baseada na metodologia qualitativa denominada *Grounded Theory*, seguindo a proposta de Charmaz (2006) e contempla os seguintes passos:

- Coleta de dados: Para a coleta de dados foram utilizadas duas técnicas: entrevistas e questionários e, após, os dados foram transcritos para auxiliar na próxima etapa.
- Análise dos dados: Nessa etapa, os dados foram codificados utilizando a ferramenta MaxQda, identificando os códigos, e agrupando-os em categorias e subcategorias.
- Construção do modelo teórico: O modelo teórico é construído para facilitar a compreensão da relação entre as categorias resultantes do processo de análise de dados e potencializar o resultado desta fase.
- Validação do modelo teórico: Após a elaboração do modelo teórico foi necessário validá-lo nas organizações.

Etapa 4. Estado da arte apoiado pelos estudos sistemáticos da literatura: Esta etapa corrobora com a etapa 3 abordando o estado da arte, contemplada por meio de dois estudos sistemáticos: mapeamento sistemático da literatura e um estudo terciário. O mapeamento sistemático da literatura visa detectar as causas que incorrem em consequências de dívidas de requisitos e as possíveis práticas para mitigá-las..

O estudo terciário consiste em compreender as atividades da gestão da dívida de requisitos e as práticas sugeridas para sua gestão. Ambos, os estudos seguem a recomendação das etapas conforme Kitchenham e Charters (2007).

Após a realização dos estudos sistemáticos, os resultados do estado da prática, apresentados no capítulo 3, os resultados do MSL e do estudo terciário, foram sintetizados e unificados, apresentando uma base única de informações, fortalecendo a conscientização do problema e evidenciando soluções praticadas nas organizações e apoiadas pela literatura.

A partir dessas evidências, foi possível identificar o artefato como solução para o problema e classes de problemas importantes para dar sequência à pesquisa.

Etapa 5. Desenvolvimento de uma ontologia para organizar as informações resultantes da unificação da visão prática e dos estudos sistemáticos: Visando estruturar, organizar e centralizar todo o conhecimento obtido nos capítulos 3 e 4, foi desenvolvida uma ontologia de domínio que reúne as causas das dívidas de requisitos, suas consequências, as práticas que podem reduzi-las e evitá-las, a gestão da dívida, suas recomendações, e a relação entre esses conceitos envolvidos. Elementos da linguagem Essence (OMG, 2018) e a estrutura da biblioteca de práticas (Jacobson, 2018) apoiam essa lacuna na etapa 6, mas precisam ser previstos na ontologia.

Etapa 6. Proposta de uma abordagem voltada à recomendação de práticas para a mitigação da dívida de requisitos: A etapa 6 propõe um artefato que se caracteriza como uma abordagem que organiza e abriga um conjunto de passos dedicados à mitigação das dívidas de requisitos por meio da recomendação de práticas e orientações que poderão ser adaptadas ao processo de requisitos.

A proposta do processo é apoiada pelo artefato *Product Scoping Canvas* (Pereira-Vale; Ochoa; Perovich 2023), ideia de *gap analysis* e pela Ontologia desenvolvida. A ontologia será o suporte aos problemas, causas e práticas das dívidas de requisitos que serão apresentadas às organizações.

Etapa 6.1 Desenvolvimento de uma solução computacional para viabilizar a interação com a abordagem proposta: A etapa 6.1 formaliza o desenvolvimento de uma solução computacional que servirá de apoio à abordagem proposta.

Etapa 7. Avaliação da abordagem e a solução computacional propostos: Nesta etapa descreve-se o modelo de avaliação da abordagem proposta e a solução computacional, utilizando a abordagem GQM (*goal, questions e metrics*), seguindo etapas de planejamento, execução e análise dos resultados (Basili; Caldiera; Rombach, 1994).

Etapa 8. Conclusões: Esta etapa consiste em descrever as conclusões sobre o trabalho desenvolvido considerando os objetivos definidos inicialmente, os resultados obtidos e as sugestões para os trabalhos futuros.

Etapa 9. Apresentação dos resultados – Submissão de artigos: Os resultados da pesquisa serão submetidos à *journals* e/ou congressos, durante o seu desenvolvimento, com a finalidade de possibilitar publicações e divulgar o estudo realizado, contribuindo, assim, com a comunidade acadêmica e a indústria.

1.7 ESTRUTURA DO TRABALHO

Além do capítulo introdutório responsável por abordar o problema, objetivos e sua relevância, contextualizando-o, este trabalho está organizado nos seguintes capítulos: O capítulo 2 apresenta o referencial teórico dos assuntos que subsidiaram a pesquisa. O capítulo 3 fundamenta um estudo para investigar a relação entre a Engenharia de Requisitos e a Dívida Técnica, potencializando uma visão prática por meio de colaboradores das organizações que adotam desenvolvimento ágil de software. O capítulo 4 apresenta o estado da arte, realizado através de um mapeamento sistemático da literatura, evidenciando os trabalhos desenvolvidos voltados às causas que ocasionam as dívidas de requisitos, as consequências que caracterizam as dívidas e as práticas que podem minimizá-las; um estudo terciário direcionado à gestão da dívida de requisitos, suas atividades ou áreas e a recomendação de práticas para atender cada área da gestão; para finalizar o capítulo 4, apresenta-se também a unificação das informações obtidas no capítulo 3 e no capítulo 4, sintetizando os resultados dos estudos envolvidos. O capítulo 5 descreve e demonstra a construção da ontologia de domínio, intitulada como OntoReD condicionada aos conceitos e os seus relacionamentos, associados à dívida de requisitos, obtidos nos estudos dos capítulos 3 e 4, fundamentais para a ontologia. O capítulo 6 detalha a abordagem proposta, denominada PracticesforReD, que recomenda um conjunto de práticas para mitigar as dívidas de requisitos em ambientes de desenvolvimento ágil de software. No capítulo 7 aborda-se a avaliação da abordagem e a sua solução computacional, apoiados pelo paradigma GQM (objetivos, questões e métricas). Finalizando a pesquisa, o capítulo 8 evidencia as conclusões do estudo e sugestões para a continuidade da pesquisa.

2 REFERENCIAL TEÓRICO

Este capítulo apresenta fundamentos teóricos que apoiam o desenvolvimento da pesquisa para melhor compreendê-la. A seção 2.1 apresenta o contexto da Engenharia de Requisitos Ágil. Na seção 2.2 aborda-se a dívida técnica no contexto do desenvolvimento de software, enfatizando a dívida de requisitos.

2.1 ENGENHARIA DE REQUISITOS ÁGIL

A Engenharia de Requisitos é denominada pelo SWEBOK (2014) como área de conhecimento de Requisitos de Software e consiste em elicitacão, análise, especificacão, validacão, bem como o gerenciamento de requisitos indicado para todo o ciclo de vida do produto de software. Essa área remete a uma grande preocupacão quando as suas atividades são executadas de forma inadequada, pois além de afetar a qualidade do processo de requisitos, prejudica as demais fases do desenvolvimento de software.

Com a popularidade das metodologias ágeis as organizaçoes passaram a adotar princípios voltados ao desenvolvimento de software “mais ágil”, diferente do que acontecia nas abordagens tradicionais, como por exemplo, a comunicacão face-a-face, ciclos curtos de desenvolvimento, entregas contínuas e desenvolvimento iterativo (Rasheed *et al.*, 2021). Porém, a Engenharia de Requisitos também se modificou, dando espaço às atividades que não possuem uma organizaçao e recomendacão tão explícita como na ER tradicional, sendo denominada como Engenharia de Requisitos Ágil.

O termo “engenharia de requisitos ágil” é usado para descrever a “maneira ágil” de planejar, executar e raciocinar sobre as atividades de engenharia de requisitos. No entanto, ainda falta compreensao de como lidar com as atividades de engenharia de requisitos no contexto ágil. Embora seja considerado interessante, a comunidade de desenvolvimento de software ainda não está familiarizada com o papel das práticas de engenharia de requisitos em métodos ágeis (Inayat *et al.*, 2015, apud Jia *et al.*, 2019).

2.1.1 Características da Engenharia de Requisitos Ágil

As abordagens tradicionais e ágeis revelam diferenças expressivas no ciclo de vida do desenvolvimento de software, assim como ocorre na Engenharia de Requisitos, a qual, alguns estudos e pesquisadores denominam Engenharia de Requisitos Ágil.

As atividades da ER ágil adotadas por abordagens ágeis, como Scrum e Extreme Programming (XP), por exemplo, diferem das atividades da ER tradicional (Ochodek; Kopczyńska, 2018). Isso ocorre porque as atividades de engenharia de requisitos ágil não são desenvolvidas sequencialmente e, embora sejam realizadas iterativamente e executadas durante cada um dos ciclos de desenvolvimento, não possui uma formalidade, permitindo que cada profissional, muitas vezes, da mesma organização, as realize sem seguir uma padronização, o que pode comprometer o processo de requisitos (Curcio *et al.*, 2018).

Devido a ER ágil ser considerada informal e baseada nas habilidades e conhecimentos dos profissionais (Dingsøyr *et al.*, 2012), é complicado descrevê-la ou caracterizá-la pois ela ainda é obscura, não apenas para desenvolvedores de software, mas para comunidade científica (Melo *et al.*, 2022).

Complementando, os requisitos na ER ágil são inicialmente definidos com o cliente e listados em formato de lista de necessidades dos stakeholders e, de acordo com Dalpiaz, Brinkkemper (2018), mais de 90% dos praticantes de metodologia ágil usam histórias de usuários para capturar requisitos. A cada duas semanas aproximadamente, eles são discutidos, (por exemplo, no método Scrum), melhor compreendidos e redefinidos como prioridade, para definir o escopo da próxima iteração. O cliente tem participação ativa na colaboração com a equipe de desenvolvimento para facilitar a compreensão e validar constantemente o produto (Inayat *et al.*, 2015).

2.2 DÍVIDA TÉCNICA NO DESENVOLVIMENTO DE SOFTWARE

Dívida Técnica (DT) no desenvolvimento de software foi inicialmente caracterizada como uma metáfora, empregada por Ward Cunningham em 1992, para justificar aos stakeholders, a necessidade de refatoração, vinculando o termo à falta de tratamento de algumas especificações, apenas em nível de código (Kruchten; Nord; Ozkaya, 2012a). Ainda, ressaltou que uma dívida pequena pode acelerar o desenvolvimento de software em curto prazo, porém, pode dedicar tempo extra com código imaturo, e com isso pagar juros sobre essa dívida (Kruchten; Nord; Ozkaya, 2012a; Guo; Spínola; Seaman, 2016).

Mais tarde esse termo evoluiu e ampliou sua prática para outros artefatos do desenvolvimento de software como dívida de: teste, pessoas, arquitetura, requisitos, documentação, ou ainda, de forma geral dívida de software (Kruchten; Nord; Ozkaya, 2012a; Li; Avgeriou; Liang, 2015).

Dessa forma, a dívida técnica tem conquistado espaço no cenário de desenvolvimento de software, desde 2012 aproximadamente, como uma alternativa para entender e apresentar fatores associados à qualidade e custos intrínsecos (Kruchten *et al.*, 2012b).

Sendo assim, a dívida técnica pode estar relacionada aos compromissos aceitos pelos desenvolvedores para atender uma demanda urgente considerando prazos curtos. Essa dívida em algum momento precisa ser paga para manter a integridade do projeto a longo prazo (Kruchten *et al.*, 2012b).

2.2.1 Classificação da dívida técnica conforme a dimensão

Segundo Yli-Huumo, Maglyas, Smolander (2016), Steve Mc Connell classificou a dívida técnica de acordo com a forma que a dívida foi adquirida: dívida técnica intencional e dívida técnica não intencional. Posteriormente, Martin Fowler aperfeiçoou essa classificação e criou os quadrantes da dívida técnica (Guo; Spínola; Seaman, 2016). Acredita-se que esse tipo de categorização de dívida técnica é interessante para identificar as causas da dívida técnica (Fowler, 2009).

2.2.1.1 Dívida técnica intencional

A dívida intencional ocorre quando a equipe, de forma consciente, a adquire por algum motivo, seja para acelerar o desenvolvimento de um projeto e entregá-lo num tempo menor, ou por alguma decisão estratégica (Avgeriou *et al.*, 2016).

2.2.1.2 Dívida Não intencional

A dívida não intencional é caracterizada dessa forma quando a dívida é contraída sem a percepção da equipe, podendo ser pela inexperiência, falta de cuidado ou falta de processo adequado (Guo; Spínola; Seaman, 2016). Diferente da dívida intencional, a não intencional não ocorre devido a uma estratégia da organização de software.

2.2.2 Classificação da dívida técnica conforme o tipo

A dívida técnica também pode estar presente em diferentes atividades ou fases do ciclo de desenvolvimento de software, caracterizando outra forma de classificação,

considerando o seu tipo. O quadro 1 apresenta alguns exemplos dos tipos de dívidas (Melo *et al.*, 2022; Li; Avgeriou; Liang, 2015; Guo; Spínola; Seaman, 2016).

Quadro 1 – Classificação da DT conforme o Tipo

Tipo da Dívida Técnica	Definição
Projeto	Refere-se a dívida técnica identificada por meio da análise do código-fonte e da descoberta de violações dos princípios de um bom design orientado a objetos.
Código	Refere-se aos problemas descobertos no código-fonte, que violam as melhores práticas de codificação e afetam negativamente sua legibilidade e dificultam sua manutenção.
Arquitetura	Refere-se a problemas identificados na arquitetura do produto, que prejudicam os requisitos da arquitetura. Geralmente, essa DT é resultado de soluções iniciais inferior ao ideal, comprometendo aspectos internos de qualidade.
Teste	Refere-se a problemas encontrados nas atividades de teste que afetam sua qualidade.
Documentação	Refere-se aos problemas encontrados na documentação do projeto de software.
Infraestrutura	Refere-se a problemas de infraestrutura que, se presentes na organização do software, atrasam ou dificultam as atividades de desenvolvimento. Esse tipo de DT implica negativamente a capacidade da equipe de produzir um produto de qualidade.
Requisitos³	Refere-se à distância entre a especificação ideal dos requisitos e a implementação real do sistema.
Usabilidade	Refere-se a decisões de usabilidade inadequadas que precisarão ser ajustadas posteriormente.
Serviço	Refere-se a serviços web inadequados que levam à incompatibilidade entre os recursos do serviço e os requisitos do aplicativo.
Versionamento	Refere-se a problemas no controle de versão do código-fonte, como bifurcações de código desnecessárias.

Fonte: (Melo *et al.*, 2022; Li; Avgeriou; Liang, 2015; Guo; Spínola; Seaman, 2016)

2.2.3 Gestão da dívida técnica

A gestão da dívida técnica (TDM) inclui um conjunto de práticas desde o reconhecimento até a sua mensuração. Mesmo sendo considerado um termo utilizado na comunidade de software, atualmente muitas organizações ainda não definiram uma estrutura para gerenciar suas dívidas e os gerentes de projetos buscam métodos e ferramentas para auxiliar nesse processo (Avgeriou *et al.*, 2016). A TDM também compreende as atividades que

³ Dívida de Requisitos: A dívida de requisitos também pode ser chamada de DT de requisitos (LI; AVGERIOU; LIANG, 2015; GUO; SPÍNOLA; SEAMAN, 2016), mas nesta tese será adotado o termo dívida de requisitos, considerando a sigla ReD sugerido por Lenarduzzi e Fucci (2019).

tratam da DT acumulada para torná-la visível e controlável e para manter um equilíbrio entre o custo e o valor do projeto de software (ALBUQUERQUE *et al.*, 2022).

As atividades que compõem a TDM podem ser adotadas para evitar a dívida técnica ou administrar as dívidas técnicas existentes. A seguir são apresentadas essas atividades (Behutiye *et al.*, 2017; Li; Avgeriou; Liang, 2015):

- Identificação ou reconhecimento de DT: Responsável por detectar DT ocasionadas por decisões intencionais ou não intencionais;
- Medição da DT: Quantifica o benefício e o custo da DT através de técnicas;
- Priorização da DT: Responsável por classificar os itens da DT conforme regras estabelecidas para priorizar quais itens devem ser pagos primeiro;
- Prevenção da DT: Responsável por evitar ocorrências de DT;
- Monitoramento da DT: O objetivo é observar as mudanças no custo e benefícios dos itens da DT não corrigidos ao longo do tempo;
- Pagamento da DT: O objetivo é corrigir as DT de um sistema de software através de técnicas como reengenharia e refatoração;
- Documentação da DT: Define uma maneira de apresentar e codificar a DT descrevendo as preocupações de partes interessadas específicas;
- Comunicação em DT: Responsável por deixar visível a DT identificada para as partes envolvidas, para que possa ser discutida, gerenciada e resolvida posteriormente.

3 DÍVIDA TÉCNICA E ENGENHARIA DE REQUISITOS: ESTADO DA PRÁTICA

Este capítulo propõe a investigação da relação entre a Engenharia de Requisitos e a Dívida Técnica em organizações que se apoiam em metodologias ágeis no desenvolvimento de software, com o objetivo de apresentar as evidências observadas a partir da perspectiva da análise qualitativa. Considerando o estudo proposto, pretende-se responder às seguintes questões de pesquisa:

QP 1. Quais as causas na engenharia de requisitos encontradas em organizações ágeis e que geram dívidas técnicas?

QP 1.1 Quais as consequências ocasionadas pelas dívidas técnicas em ER?

QP 2. Quais práticas são empregadas em organizações ágeis que podem minimizar as dívidas técnicas geradas durante a Engenharia de Requisitos?

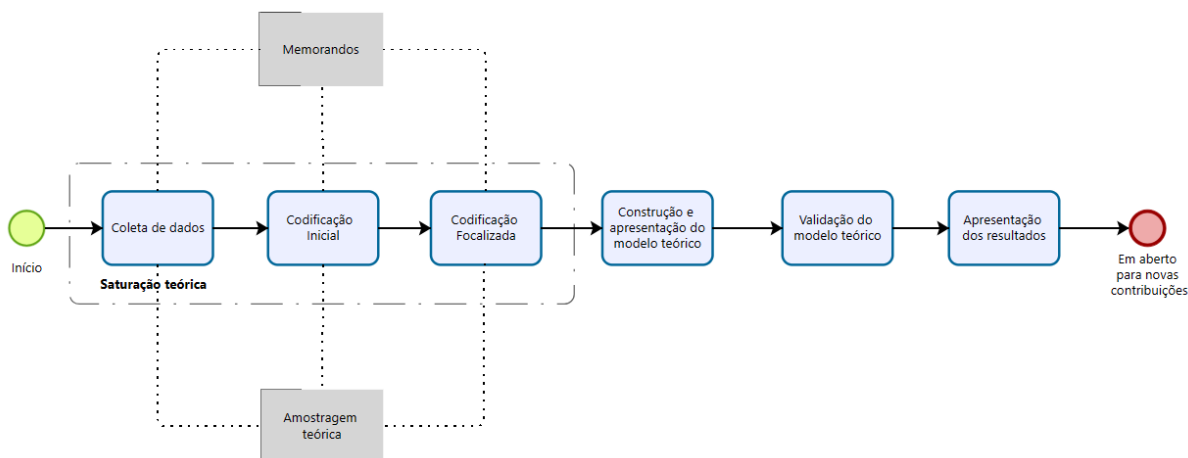
Ao final desta investigação, pretende-se fundamentar essa relação e demonstrar um modelo teórico que represente o resultado do estudo realizado.

3.1 CARACTERIZAÇÃO DA PESQUISA

Para realizar a investigação entre as áreas mencionadas, foram utilizadas as técnicas de análise da *Grounded Theory* (GT), seguindo a abordagem construtivista recomendada por Charmaz (2006) a fim de auxiliar nesse processo. Entende-se que a GT é bem aceita em pesquisas quando não se conhece inicialmente o problema (Melegati; Goldman; Kon; Wang, 2019) e tem se difundindo em vários estudos na área de Engenharia de Software (Hoda; Noble; Marshall, 2012; Stol; Ralph; Fitzgerald, 2016; Barbosa, 2017; Madampe; Hoda; Grundy; Singh, 2020).

Este estudo apresenta o processo utilizado abordando a coleta, a codificação dos dados, a construção e apresentação do modelo teórico como resultado do estudo e a validação do modelo, conforme Figura 2. Destaca-se que se realiza ciclos iterativos entre as etapas da coleta e codificação dos dados através de comparação constante entre elas, e o auxílio de anotações, as quais são chamadas de memorandos, que apoiam a análise interpretativa dos dados. O ciclo encerra quando a saturação teórica é alcançada, ou seja, quando não existe nenhum dado novo para incluir na codificação.

Figura 2 – Estrutura da pesquisa



Fonte: Elaborado pelo autor (2021)

3.2 COLETA DE DADOS

A coleta de dados contemplada nesta pesquisa possui característica exploratória (Easterbrook *et al.*, 2008) e tem como objetivo apoiar o estudo entre engenharia de requisitos e dívidas técnicas em organizações ágeis no âmbito nacional e internacional. As empresas participantes da pesquisa foram selecionadas a partir da rede de contatos dos pesquisadores ou por indicações de colaboradores das empresas.

O número de participantes, bem como, o número de empresas participantes da pesquisa não foi determinado inicialmente, justificando o uso da amostragem teórica e saturação dos dados para finalizar a coleta de dados, pois, quando não existe dado novo significativo para a pesquisa, entende-se que a saturação foi alcançada e a amostragem teórica é suficiente para o estudo (Charmaz, 2014).

Para oficializar a participação na pesquisa, empresa e colaboradores participantes preencheram os seguintes documentos: declaração de ciência e concordância da instituição envolvida⁴ realizada pelo representante legal ou um colaborador autorizado pela empresa; um termo de consentimento livre e esclarecido⁵ preenchido pelo(s) colaborador(es) participante(s) da pesquisa. O preenchimento dos formulários facilitou a aceitação das empresas em participar da pesquisa, evitando, assim, o deslocamento do pesquisador para coletar a assinatura em documentos físicos e permitindo que empresas localizadas em outros estados, e até mesmo fora do Brasil, pudessem participar. Destaca-se ainda, que a coleta de dados ocorreu no período de

⁴ Disponível em: <https://forms.gle/wHeN4xHdjQHxNJgd9>;

⁵ Disponível em: <https://forms.gle/JKUcmyWW9Nsuf8H9>.

pandemia da Covid 19 e, por isso, foi necessário manter o protocolo de cuidados e distanciamento emitido pelo Ministério da Saúde.

Na coleta de dados realizou-se a entrevista intensiva utilizada na GT, orientada por uma conversa mais direcionada, apropriando-se de questões semiestruturadas e proporcionando aprofundamento sobre o tema investigado (Charmaz, 2014). Deste modo, como técnicas de coleta de dados foram utilizadas entrevista e questionário, o que facilitou a obtenção dos dados devido à localização das empresas.

A entrevista intensiva semiestruturada se configurou como a técnica principal da coleta e possui um roteiro previamente definido para auxiliar o entrevistador na sua condução, e está condicionada às duas áreas mencionadas: Engenharia de Requisitos e Dívida Técnica. A preferência pela entrevista semiestruturada justifica-se pela sua flexibilidade durante a condução, pois é possível ajustar as questões conforme a necessidade e sua evolução, proporcionando ao entrevistado o relato de informações não questionadas na entrevista pelo entrevistador (Melegati *et al.*, 2019), sendo que alguns exemplos das questões podem ser visualizados no Apêndice A. O questionário⁶ foi utilizado como segunda opção, sendo preenchido apenas por um participante de uma empresa internacional devido ao fuso horário.

A coleta de dados foi realizada no período de junho de 2020 a janeiro de 2021, com colaboradores das empresas que possuem responsabilidades de gerente de produtos, gerente de projetos ou líder da equipe de desenvolvimento.

As entrevistas aconteceram individualmente para obter visões diferentes sobre o mesmo contexto, totalizando, entre entrevistas e questionário, a participação de 15 colaboradores em 9 empresas brasileiras e 4 colaboradores em 4 empresas internacionais. Todas as entrevistas foram gravadas com autorização dos participantes e, após, realizada a sua transcrição. Cada transcrição das entrevistas contabilizou, aproximadamente, 9 páginas com 5 mil palavras cada uma, caracterizando o total de 171 páginas transcritas e 90 mil palavras.

3.2.1 Perfil dos participantes: Grupo amostral

Os participantes da pesquisa foram escolhidos conforme a área de atuação na empresa, concedendo a preferência para os colaboradores que atuam diretamente com a área de requisitos. O total de empresas nacionais e internacionais, o perfil dos participantes considerando sua função na empresa e sua experiência na função, bem como o segmento das empresas, pode ser visualizado no Apêndice B.

⁶ Disponível em: <https://forms.gle/NqKxG5xBdmDuBZ3H6>.

A diversidade de empresas nacionais e internacionais com atuação em diferentes segmentos, bem como, o perfil dos participantes considerando seu nível de experiência, fundamentado pelo tempo em determinado papel (novato – até 3 anos de experiência; experiente – acima de três anos de experiência), formação e qualificação, proporcionou visões diferentes sobre o objeto de estudo, o que demonstra uma representatividade desses dados e a amostragem teórica necessária durante o processo de coleta e codificação dos dados.

3.3 ANÁLISE DOS DADOS

A análise dos dados foi realizada a posteriori segundo a premissa das técnicas da GT na perspectiva de Charmaz (2014): codificação inicial e focalizada por meio de um processo cíclico, apresentadas nas seções subsequentes.

Durante a análise, foi utilizado o método de comparação no agrupamento dos códigos identificados para produzir maior abstração dos dados coletados na codificação (Hoda; Noble; Marshall, 2010). Para contemplar todo o processo de codificação, foi utilizada a ferramenta MaxQda⁷ para facilitar a organização e manipulação dos dados.

3.3.1 Codificação inicial

A codificação inicial foi a primeira etapa do processo de codificação aplicada, com a análise linha-a-linha de todo o material transcrito, resultado da coleta de dados. Esta é uma etapa de descoberta pois, a partir da leitura das transcrições, a codificação linha-a-linha permitiu a identificação de importantes elementos caracterizando conceitos, os quais foram surgindo e, alinhados ao objeto de estudo acerca das percepções dos participantes, possibilitando uma codificação indutiva.

Os códigos surgiam conforme a observação dos próximos segmentos a serem codificados pois, no momento que determinado segmento era selecionado, verificava-se a existência de algum conceito relacionado ao contexto do segmento ou não. Em caso positivo, associava-se o segmento ao código existente e, em caso negativo, criava-se um código para o segmento analisado. O Quadro 2 apresenta a lista de códigos e a sua incidência (quantidade de segmentos codificados relacionados a determinado código).

Os conceitos identificados totalizaram 2599 segmentos codificados agrupados em 27 códigos. Os exemplos de segmentos codificados de determinado código na codificação inicial,

⁷ MaxQda: <https://www.maxqda.com/>

os quais foram caracterizados conforme sua semelhança, proximidade e/ou relação com o código criado são demonstrados no Apêndice C.

Quadro 2 – Lista de códigos: Codificação inicial e incidência

Lista de Códigos - Codificação Inicial e incidência			
▼ [ícone] Codificação Inicial	0	> [ícone] Estimativas de Tempo	34
> [ícone] Usabilidade	20	> [ícone] Projetos/Produtos	76
> [ícone] Métricas	2	> [ícone] Ferramentas/Tecnologias	87
> [ícone] Melhorias	46	> [ícone] Testes	40
> [ícone] Experiência dos colaboradores	21	> [ícone] Arquitetura	4
> [ícone] Ambientes de Desenvolvimento	6	> [ícone] Bugs	42
> [ícone] Qualidade	3	> [ícone] Metodologia/Processo	32
> [ícone] Manutenção	6	> [ícone] Documentação	105
> [ícone] Inovação	7	> [ícone] Demandas	164
> [ícone] Estimativa de custo	10	> [ícone] Práticas	622
> [ícone] Gestão de risco	1	> [ícone] Requisitos	275
> [ícone] Comunicação	33	> [ícone] Dívida Técnica	260
> [ícone] Stakeholders	11	> [ícone] Papéis/equipe	264
> [ícone] Gerência de Configuração	39	> [ícone] Problemas	351

Fonte: Elaborado pelo autor na ferramenta MaxQda (2021)

3.3.2 Codificação focalizada

Na codificação focalizada, o agrupamento das categorias e subcategorias foi baseado na literatura, considerando o contexto em que os códigos da codificação inicial estavam inseridos e o foco deste estudo: Dívida Técnica, Engenharia de Requisitos, desenvolvimento ágil de software. Deste modo, para fundamentar as subcategorias relacionadas aos requisitos, elas foram apoiadas nas atividades voltadas à Engenharia de Requisitos: elicitação de requisitos, análise de requisitos, especificação de requisitos, validação de requisitos e gerência de requisitos, usando como referência o Swebok (2014) para construção das categorias e subcategorias, e foram agrupadas conforme semelhanças e características.

Para alcançar as subcategorias e categorias finais, foram realizados alguns ciclos de refinamento, identificando a categoria central como “Fatores que impactam na Engenharia de Requisitos”, ou seja, o fenômeno da pesquisa. Esse refinamento resultou em três categorias finais: “causas que geram DT em requisitos”; “consequências que caracterizam DT em requisitos”; e “práticas que podem reduzir e/ou tratar DT em requisitos”.

As categorias finais e as subcategorias que impactam na Engenharia de Requisitos são apresentadas no Apêndice D, e o nível mais profundo (evidências) das subcategorias pode ser consultado no Apêndice E.

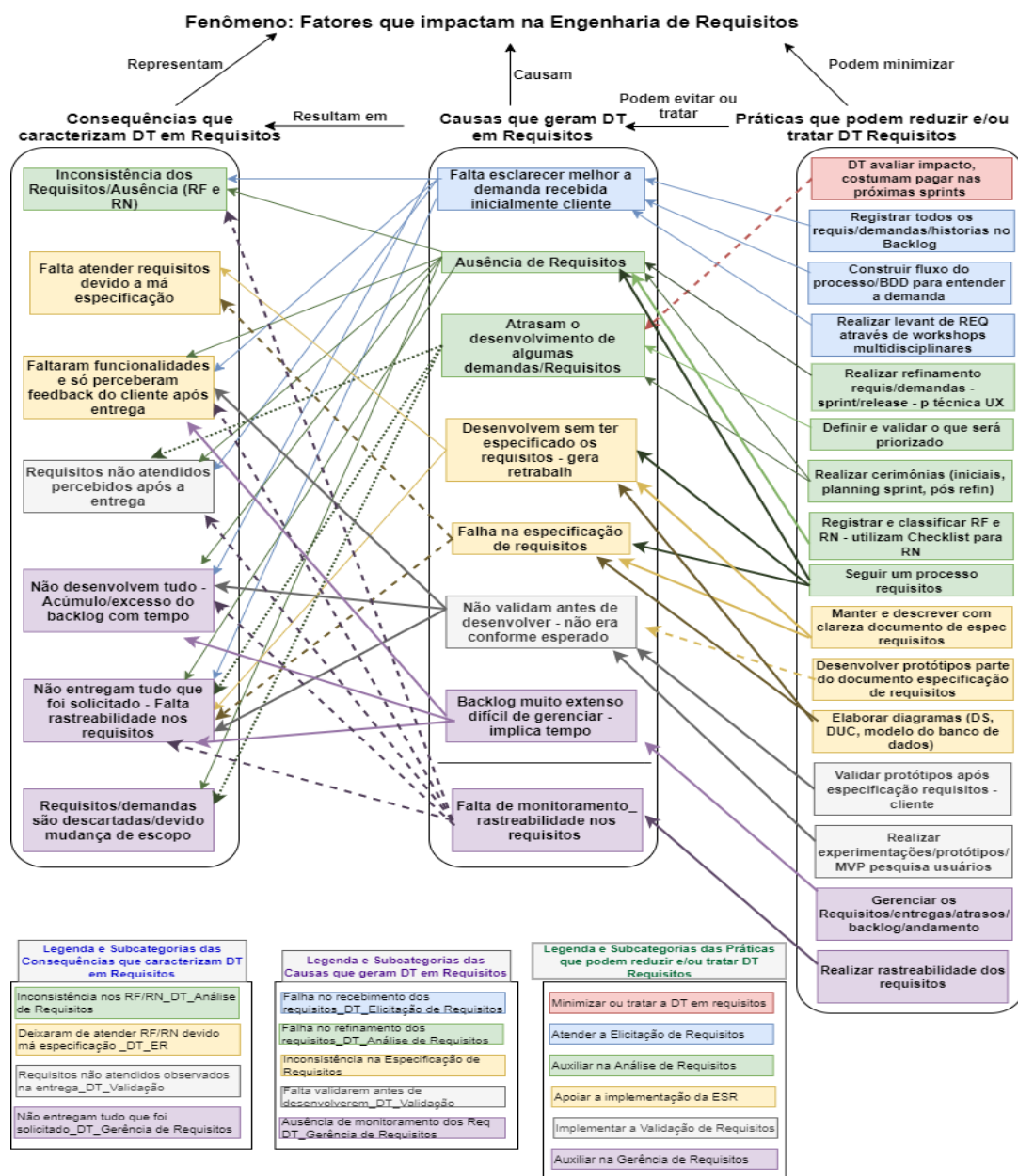
3.4 CONSTRUÇÃO E APRESENTAÇÃO DO MODELO TEÓRICO PROPOSTO

As categorias que envolvem as causas e consequências da ReD são pontos-chave do estudo. Entende-se que as causas podem ser minimizadas com o auxílio de práticas, pois estas têm a capacidade de se adequar ao processo de requisitos, impedindo que dívidas técnicas sejam contraídas durante a etapa inicial do processo de desenvolvimento e assim, reduzir as dívidas decorrentes da Engenharia de Requisitos. As práticas dispõem de atividades relacionadas a artefatos, atividades, cerimônias e processos adotados durante o desenvolvimento de software. Nos Apêndices F, G e H, é possível observar as evidências das subcategorias de cada categoria principal: **“Causas que geram DT em requisitos”**, **“Consequências que caracterizam DT em Requisitos”**, **“Práticas que podem reduzir e/ou tratar DT Requisitos”**, que se manifestaram nos relatos dos participantes da pesquisa e que fundamentaram as evidências na construção do modelo teórico, apresentado na Figura 3.

Após a construção do modelo teórico, fundamentado a partir da codificação focalizada apoiada pelos relatos dos participantes, na qual pode-se observar as causas que geram as ReD conforme as atividades da ER e as práticas que podem evitar os fatores que causam essas dívidas. Por meio do modelo foi possível responder às questões de pesquisa elencadas inicialmente, caracterizando os fatores que impactam na Engenharia de Requisitos.

Destaca-se que é necessário validar o modelo teórico proposto utilizando o auxílio da literatura e da indústria para ter uma maior contribuição na área.

Figura 3 – Modelo teórico



Fonte: Elaborado pelo autor (2021)

3.5 VALIDAÇÃO DO MODELO TEÓRICO

A validação dos dados é uma atividade recomendada em estudos apoiados pela GT sob o olhar da teoria resultante do processo ou sob a percepção do modelo teórico e visa o envolvimento dos participantes da pesquisa avaliando se o resultado da pesquisa sintetizado no modelo teórico corresponde à sua realidade (Strauss; Corbin, 2008). Essa validação também pode se estender a especialistas da área que não participaram da pesquisa, mas que vivenciam o fenômeno do estudo.

3.5.1 Preparação da validação do Modelo Teórico

A validação do modelo teórico configurou-se por meio de um questionário⁸ elaborado na ferramenta survio⁹, contendo questões sobre as três categorias principais que formam o modelo: causas de dívidas de requisitos, as consequências que caracterizam as dívidas de requisitos e as práticas que podem mitigar essas dívidas.

O questionário foi fragmentado em três seções considerando as três categorias principais para a pesquisa, sendo que para cada questão optou-se pelas seguintes possibilidades de resposta: “Acontece frequentemente”; “acontece eventualmente”; “não acontece”; seguidas do campo: “favor justificar sua resposta” para facilitar a compreensão por parte do pesquisador. Ao final de cada seção foi prevista uma questão aberta referente às causas, dívidas e práticas para o colaborador informar situações sobre cada categoria que não foram contemplados nas questões anteriores.

3.5.2 Realização da validação do Modelo Teórico

Após a preparação do questionário de validação do modelo teórico, o formulário foi encaminhado, via e-mail, para os participantes do estudo inicial, o qual deu origem ao modelo teórico, convidando-os a participarem da validação, justificando sua importância para esta pesquisa. Os e-mails foram encaminhados e preenchidos no período de dezembro de 2021 a fevereiro de 2022, destacando que dos 19 participantes da fase inicial da pesquisa, 10 responderam ao questionário da validação. Alguns colaboradores sinalizaram a mudança de empresa e, por isso, não foi possível contar com essas participações na validação.

3.5.3 Resultado da validação do Modelo teórico

Após os colaboradores responderem o questionário de validação do modelo teórico, o resultado¹⁰ foi compilado e sintetizado por meio de gráficos gerados automaticamente pela ferramenta.

Embora alguns participantes apontaram a inexistência de falhas no processo de requisitos que estão diretamente associadas às dívidas de requisitos, outros confirmaram a sua

⁸ Questionário da validação dos dados disponível em: <https://www.surveymonkey.com/survey/d/K2Z6P7X0E1P5Q1L9X>

⁹ <http://survio.com>

¹⁰ https://drive.google.com/file/d/14m_cvUfuto8hrWJ1AD5_o0bFs_Xj7kd8/view?usp=sharing

existência, como a necessidade de esclarecer melhor a demanda; atrasos na entrega; a ausência da especificação de requisitos antes do desenvolvimento; bem como falhas na especificação; times que não costumam validar os requisitos antes de desenvolver e não costumam monitorá-los; ausência de requisitos que são reportados pelos stakeholders após a implantação da versão; requisitos descartados após a mudança de escopo; e falta de rastreabilidade nos requisitos.

Observa-se que mesmo não acontecendo em todas as organizações que participaram da validação, ainda acontecem eventualmente ou frequentemente, o que demonstra a necessidade de meios para mitigá-las e preveni-las.

Questionados sobre as práticas adotadas na organização, da mesma forma que ocorreu na seção que envolve o processo de requisitos, alguns participantes apontaram a existência e outros a ausência de certas práticas, assim como a sua eventualidade. Dentre elas destacam-se: o pagamento das dívidas de requisitos ocorre na próxima sprint, release ou versão; registro de requisitos, demandas ou histórias no backlog; construção do processo de negócio para entender melhor a demanda; refinamento dos requisitos; definem as prioridades; realização de cerimônias iniciais, de planejamento e *review*; possuem um processo de requisitos organizado; especificam claramente os requisitos; elaboram diagramas para especificar os requisitos; validam o protótipo; e monitoram os requisitos, demandas.

Observando esses apontamentos, percebe-se que mesmo alguns dos participantes das organizações adotando práticas no seu processo de requisitos, evidenciam várias falhas que ocasionam as dívidas de requisitos, o que demonstra a necessidade de meios mais efetivos para mitigá-las e preveni-las, a partir de uma maior clareza sobre o que causa as suas dívidas de requisitos e quais as práticas mais adequadas para tratá-las.

3.6 CONTRIBUIÇÕES

O modelo teórico, resultado do estado da prática (visão prática), permitiu obter uma visão aprofundada sobre causas das dívidas de requisitos, suas consequências e práticas que poderão minimizá-las e evitá-las. Essas informações contribuirão para a área envolvida, pois na literatura ainda existem poucos trabalhos desenvolvidos voltados à compreensão das dívidas de requisitos, suas lacunas e mecanismos para tratá-las, por parte das organizações ágeis.

A validação fortaleceu os resultados apresentados no modelo teórico e reforçou a necessidade de buscar instrumentos de apoio que possam auxiliar as organizações a

compreenderem melhor as causas e suas dívidas de requisitos e meios para mitigá-las especificamente.

Destaca-se que o resultado desta pesquisa servirá de apoio para o desenvolvimento das próximas etapas da tese.

4 ESTADO DA ARTE: ESTUDOS SISTEMÁTICOS

Considerando a relevância da literatura em corroborar com o estudo realizado no capítulo 3, este capítulo apresenta um mapeamento sistemático da literatura (MSL) e um estudo terciário. O MSL foi realizado para investigar as causas, dívidas e práticas que envolvem a dívida de requisitos, fortalecendo os resultados obtidos anteriormente no estado da prática, contribuindo com as evidências encontradas e com a possibilidade de novas percepções.

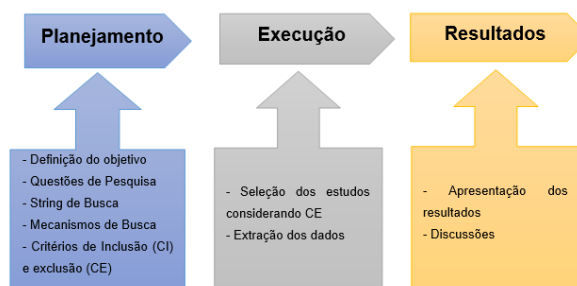
O estudo terciário consiste em facilitar a compreensão das atividades ou áreas que contemplam a gestão da dívida técnica, especialmente a dívida de requisitos, contribuindo com as evidências das práticas identificadas no estado da prática e o MSL.

4.1 MAPEAMENTO SISTEMÁTICO DA LITERATURA

Este MSL visa apoiar o estudo desenvolvido voltado ao estado da prática (capítulo 3), identificando o que está sendo produzido no âmbito da pesquisa. Dessa forma, o mapeamento sistemático apoiará o estado da arte voltado às estratégias que as organizações ágeis utilizam para identificar as dívidas de requisitos, as causas que geram essas dívidas, bem como práticas que podem minimizá-las, colaborando assim, com os resultados e discussões do estudo prático já desenvolvido.

Pesquisando sobre estudos sistemáticos relacionados ao escopo de dívida técnica na Engenharia de Requisitos, foram encontrados alguns trabalhos, como, por exemplo, o MSL de Wang e Huang (2020) que se destaca, especialmente, por abordar as causas que geram as dívidas de requisitos, mas direcionado à medição da dívida, não evidenciando as consequências dessas dívidas e as práticas que poderiam minimizá-las ou tratá-las. Os demais trabalhos encontrados na literatura são direcionados à dívida técnica de modo geral ou trabalhos muito específicos sobre outros temas, como, por exemplo, a priorização da dívida técnica conforme destacam os estudos de Alfayez et. al (2020) e Lenarduzzi et. al (2020). O MSL foi conduzido seguindo a proposta de Kitchenham e Charters (2007) e está organizado da seguinte forma, representada na Figura 4.

Figura 4– Etapas do Mapeamento Sistemático da Literatura



Fonte: Elaborado pelo autor (2023)

4.1.1 Planejamento

No planejamento apresenta-se o protocolo do mapeamento sistemático da literatura.

4.1.1.1 Objetivo e Questões de Pesquisa

O objetivo deste mapeamento sistemático da literatura (MSL) é investigar o estado da arte relacionado à relação entre Engenharia de Requisitos e Dívida Técnica direcionado às causas que geram dívidas de requisitos, consequências que as caracterizam e as práticas que podem minimizar ou tratar essas dívidas em organizações ágeis. O MSL pretende responder às questões de pesquisa, representadas no Quadro 3.

Quadro 3 – Questões de pesquisa do MSL

Questões de pesquisa	
QP 1. Quais estratégias as organizações ágeis utilizam para identificar dívidas de requisitos, conforme as etapas da Engenharia de Requisitos?	
QP 2. Quais as causas na engenharia de requisitos encontrados em organizações ágeis e que geram dívidas de requisitos?	QP 2.1 Quais as consequências ocasionadas pelas dívidas na Engenharia de Requisitos?
QP 3. Quais metodologias e/ou práticas são empregadas em organizações ágeis que podem minimizar as dívidas técnicas geradas durante a Engenharia de Requisitos?	
QP 4. Como as organizações ágeis monitoram suas dívidas de requisitos?	

Fonte: Elaborado pelo autor (2021)

4.1.1.2 String de busca e fontes de pesquisa

A estratégia de busca envolve as fontes e termos de pesquisa mais relevantes conforme o escopo de cada pesquisa. Para este estudo, a string de busca é constituída dos termos de pesquisa, observados no Quadro 4 e da string genérica, apresentada no Quadro 5. Os termos de

pesquisa consideraram termos de estudos sistemáticos similares, conforme (Melo *et al.*, 2021; Wang; Huang, 2020; Lenarduzzi *et al.*, 2020; Heeager; Nielsen, 2018; Behutiye *et al.*, 2017; Schön; Thomaschewski; Escalona, 2017).

Além do termo “dívida de requisitos”, foi incluído também o termo “dívida técnica” que, embora seja um termo genérico da dívida, pode trazer nos seus estudos alguns elementos voltados à dívida de requisitos, justificado pelos poucos trabalhos voltados à dívida de requisitos.

Quadro 4 – Termos de pesquisa

Termos da pesquisa
Dívida de Requisitos: (technical debt OR debt requirements)
Engenharia de Requisitos: (requirements engineering OR software requirements)
Metodologia de desenvolvimento de software ágil: (agile OR scrum OR extreme programming OR kanban OR lean)

Fonte: Elaborado pelo autor (2021)

Quadro 5 – String genérica

String genérica
(“technical debt” OR “debt requirements”) AND (“requirements engineering” OR “software requirements”) AND (“agile” OR “scrum” OR “extreme programming” OR “kanban” OR “lean”)

Fonte: Elaborado pelo autor (2021)

As fontes de busca adotadas no MSL foram definidas considerando as recomendações de Kitchenham e Charters (2007): *ACM Digital Library*, *IEEEExplore*, *Science Direct*, *Springer Link* e *Scopus*. As strings formatadas para cada mecanismo de busca e o número de artigos retornados encontram-se no Apêndice I.

4.1.1.3 Critérios de inclusão e exclusão

Neste MSL foram considerados os estudos que atendem aos seguintes critérios de inclusão (CI) e desconsiderados os que atendem aos critérios de exclusão (CE), ambos apresentados no Quadro 6.

Quadro 6 - Critérios de inclusão e exclusão

Critérios de Inclusão
CI ₁ - Papers publicados entre 2016-2021 (últimos 5 anos)
CI ₂ - Papers escritos em inglês;
CI ₃ - Papers revisados por pares;
CI ₄ - Papers que investiguem as causas de DT em requisitos e suas consequências em organizações ágeis;
CI ₅ - Papers que especificam práticas e/ou artefatos para as atividades da ER que podem minimizar as DT em requisitos, e/ou abordem a gestão de DT, considerando organizações ágeis;

CI ₆ - Publicações que possuem o documento completo.
Critérios de Exclusão
CE ₁ - Papers que não estão em inglês;
CE ₂ - Papers com estudos secundários ou terciários serão excluídos;
CE ₃ - Papers duplicados localizados em mais de um mecanismo de busca ou que tenham o mesmo resultado publicado em mais de um veículo;
CE ₄ - Papers que o documento completo não possa ser localizado ou que contenham apenas resumo ou resumo expandido (com até uma página);
CE ₅ - Papers fora do contexto de Dívida Técnica, Engenharia de Requisitos e organizações ágeis.

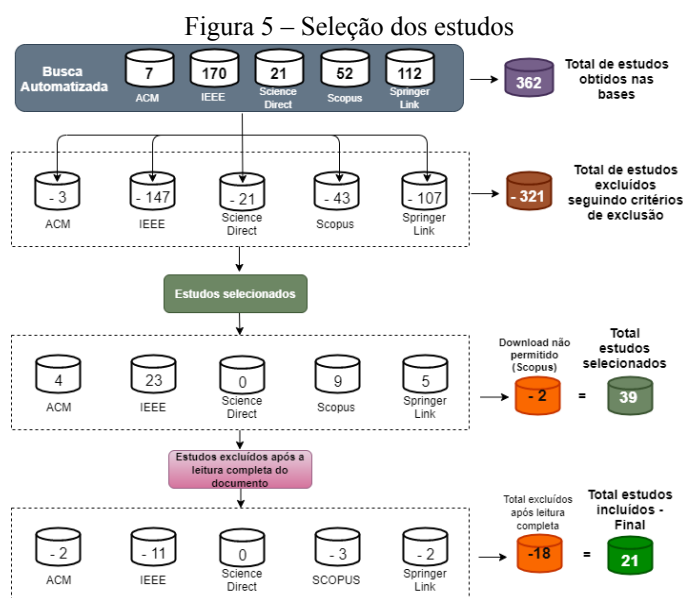
Fonte: Elaborado pelo autor (2021)

4.1.2 Execução

A execução da string nas bases de dados ocorreu em agosto de 2021 resultando em 362 estudos.

4.1.2.1 Seleção de dados

O processo de seleção simplificado dos estudos está retratado na Figura 5, totalizando 21 estudos incluídos para extração de dados, sendo que o processo completo pode ser visualizado no Apêndice J.



Fonte: Elaborado pelo autor (2021)

4.1.2.2 Extração de dados

O quadro 7 apresenta os estudos selecionados para a extração dos dados¹¹.

Quadro 7 – Estudos incluídos no MSL

Identificação	Autor	Título	Mecanismo de busca
A1	FREIRE et al. (2020)	Actions and impediments for technical debt prevention: results from a global family of industrial surveys	ACM -
A2	RINDELL, BERNISMED, JAATUN (2019)	Managing Security in Software: Or: How I Learned to Stop Worrying and Manage the Security Technical Debt	ACM -
I1	MANDIĆ et al. (2021)	Technical and Non-Technical Prioritization Schema for Technical Debt: Voice of TD-Experienced Practitioners	IEEE -
I2	FREIRE et al. (2020a)	Pitfalls and Solutions for Technical Debt Management in Agile Software Projects	IEEE -
I3	FREIRE et al. (2020b)	How Experience Impacts Practitioners' Perception of Causes and Effects of Technical Debt	IEEE -
I4	ROSSER, NORTON (2021)	A Systems Perspective on Technical Debt	IEEE -
I5	SIGG, REYES, ÁVILA (2019)	Towards the Use of a Framework to Make Technical Debt Visible	IEEE -
I6	RAMAC et al. (2020)	Common Causes and Effects of Technical Debt in Serbian IT: IncisTD Survey Replication	IEEE -
I7	BEHUTIVE et al. (2020)	How agile software development practitioners perceive the need for documenting quality requirements: a multiple case study	IEEE -
I8	LENARDUZZI, FUCCI (2019)	Towards a Holistic Definition of Requirements Debt	IEEE -
I9	LENARDUZZI et al. (2019)	An Empirical Study on Technical Debt in a Finnish SME	IEEE -
I10	BESKER, MARTINI, BOSCH (2019)	Technical Debt Triage in Backlog Management	IEEE -
I12	RIOS et al. (2018a)	A Study of Factors that Lead Development Teams to Incur Technical Debt in Software Projects	IEEE -
I13	GHANBARI (2016)	Seeking Technical Debt in Critical Software Development Projects: An Exploratory Field Study (Não informaram se usam metodologias ágeis ou não)	IEEE -
SC1	RIOS et al. (2020)	Hearing the Voice of Software Practitioners on Causes, Effects, and Practices to Deal with Documentation Debt	Scopus -
SC2	ROBIOLO et al. (2019)	Technical Debt and Waste in Non-functional Requirements Documentation: An Exploratory Study	Scopus -
SC3	MENDES et al. (2016)	Impacts of agile requirements documentation debt on software projects: A retrospective study	Scopus -
SC4	KUHLEN, SPECK (2016)	The potentials of a code generator which faces the stress ratio of requirements engineering processes in agile development projects	Scopus -
SL1	KARHAPAA et al. (2021)	Strategies to manage quality requirements in agile software development: a multiple case study	Springer Link -
SL2	RIOS et al. (2020b)	The practitioners' point of view on the concept of technical debt and its causes and consequences: a design for a global family of industrial surveys and its first results from Brazil	Springer Link -
SL3	FERNANDEZ et al. (2017)	Naming the pain in requirements engineering	Springer Link -

Fonte: Elaborado pelo autor (2021)

4.1.3 Resultados

Esta seção descreve os principais resultados dos 21 estudos obtidos por meio da condução do mapeamento sistemático da literatura, os quais estão fragmentados nas seções subsequentes e respondem às questões de pesquisa QP1 a QP4 apresentadas na seção 4.1.1.1.

4.1.3.1 Quais estratégias as organizações ágeis utilizam para identificar dívida de requisitos conforme as etapas da Engenharia de Requisitos em organizações ágeis?

Na QP1 almeja-se entender a familiaridade das organizações com o termo “dívida técnica”, como as dívidas são caracterizadas e se utilizam alguma estratégia para identificá-las.

4.1.3.1.1 Familiaridade dos profissionais com conceito de Dívida Técnica

Considerando a “familiaridade dos profissionais com conceito de Dívida Técnica” pretendia-se compreender o quanto as empresas tinham conhecimento sobre o termo “Dívida

¹¹ A planilha com a extração dos dados pode ser consultada no link: https://docs.google.com/spreadsheets/d/158n_0PcwB10n7mk-yOdvqM4-1K4AI0RY/edit?usp=sharing&ouid=100014371627052372520&rtpof=true&sd=true

Técnica” e, conforme os estudos selecionados, apenas 9,52% dos estudos revelaram situações que destacaram a compreensão com o termo.

O estudo desenvolvido por Ramač et al. (2020) indicou que quando os pesquisadores não contextualizaram o tema, 56% dos participantes relataram que conheciam o conceito de dívida técnica (DT); 21% possuíam apenas compreensão teórica (só leram em livros ou artigos) e 35% tiveram alguma experiência prática em projetos, porém, 44% dos participantes nunca ouviram falar no conceito de DT até então. Após contextualizar sobre DT, 73% dos participantes estavam de acordo com a definição de DT fornecida.

Na pesquisa desenvolvida por Rios *et al.* (2020b), sem abordar o tema para os participantes, 60% revelaram pouca familiaridade com o conceito, 22% indicaram que o termo fazia parte das suas atividades diárias; e, após, a abordagem mais detalhada sobre DT, 80% dos participantes sinalizaram que conheciam o termo.

O relato dos estudos apresentados, revela que ainda não há uma compreensão sobre a DT nas organizações por parte dos profissionais e, conseqüentemente, o que é caracterizado dentro desse contexto. Essa falta de entendimento pode se tornar um obstáculo na identificação das causas de dívidas técnicas e das próprias dívidas, independentemente do tipo da dívida técnica, dificultando implementações de ações práticas que podem minimizar a causa dessas dívidas.

4.1.3.1.2 Como as organizações ágeis caracterizam Dívida de Requisitos

Esta questão tem como finalidade explorar como essas organizações definem as dívidas de requisitos, sendo que apenas dois dos estudos abordaram essa caracterização.

O artigo de Rosser e Norton (2021) visa explorar a metáfora da dívida técnica, que, segundo os pesquisadores, se originou do desenvolvimento ágil, e apresentaram uma compreensão atual do conceito da dívida técnica. Nesse estudo, os autores caracterizaram os diferentes tipos de dívida técnica, dentre eles, a dívida de requisitos, a qual caracteriza como requisitos malformados que não tem a devida clareza, não testáveis ou difíceis de verificar.

Lenarduzzi e Fucci (2019) propuseram uma conceituação para caracterizar a dívida de requisitos, a qual considera as seguintes atividades da engenharia de requisitos: elicitacão de requisitos, especificação de requisitos e gerência de requisitos. Assumindo o conceito proposto por Ernst (2012), a dívida de requisitos “refere-se à distância entre a especificação ótima dos requisitos e a implementação real do sistema, sob suposições e restrições de domínio”, o estudo sinaliza ainda tipos de dívida de requisitos:

A dívida de requisitos tipo 0 expressa as “necessidades incompletas dos usuários” indicando que a dívida contraída foi resultado da negligência, por parte dos profissionais, em absorver as necessidades das partes interessadas. O estudo ainda destaca que esse tipo de dívida de requisitos ocorre quando nem todas as necessidades dos usuários são capturadas após o feedback recebido por meio dos canais de comunicação estabelecidos; ou quando alguns dos canais de comunicação mais relevantes não são considerados.

O tipo 1 da dívida está relacionado à má qualidade dos requisitos e considera a dívida incorrida quando profissionais, como engenheiro de requisitos, por exemplo, documentam os requisitos usando linguagem natural, o que pode violar a norma ISO 29148 voltada à qualidade dos requisitos. Os requisitos formalizados por meio de linguagem natural podem ocasionar ambiguidade por conta das diferentes interpretações que podem acontecer por parte dos desenvolvedores resultando em uma implementação incompatível com as necessidades reais dos stakeholders (tipo 2).

No tipo 2 da dívida de requisitos, Lenarduzzi e Fucci (2019) destacaram a “implementação incompatível” que ocorre quando a implementação realizada pelos desenvolvedores, apoiada pelo documento de especificação de requisitos, não condiz com as necessidades reais dos stakeholders.

4.1.3.1.3 Estratégias para identificar/evitar dívida de requisitos

Esta questão busca identificar as dívidas de requisitos, considerando também estratégias gerais, pois é possível relacioná-las aos requisitos. Ressalta-se que 14,28% dos estudos relataram nas suas pesquisas estratégias que ajudam a identificar a dívida.

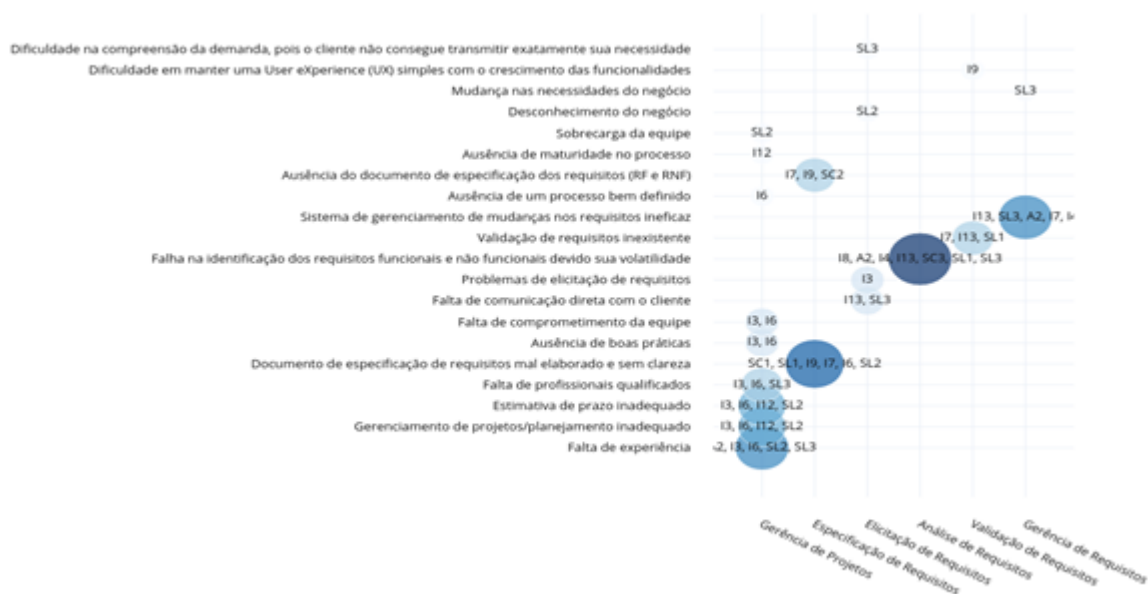
Rosser e Norton (2021) mencionaram no estudo que utilizam ferramentas de análise de requisitos, como analisadores de requisitos, pois podem ajudar a avaliar o nível de dívida em um conjunto de requisitos. Também destacaram que documentar as dívidas pode ser uma alternativa complementar, assim como também pontuado no estudo realizado por Sigg, Reyes e Ávila (2019).

Lenarduzzi e Fucci (2019) destacam que técnicas que classificam automaticamente o feedback dos usuários e das partes interessadas podem ser uma alternativa para identificar dívidas de requisitos. A estratégia para detecção de implementação incompatível consiste em abordagens de rastreabilidade entre a especificação de requisitos e o código-fonte.

4.1.3.2 QP 2. Quais as causas na engenharia de requisitos encontrados em organizações ágeis e que geram dívidas de requisitos/dívidas técnicas?

As causas das dívidas de requisitos e sua classificação estão apresentadas na Figura 6.

Figura 6 – Causas que geram dívida de requisitos

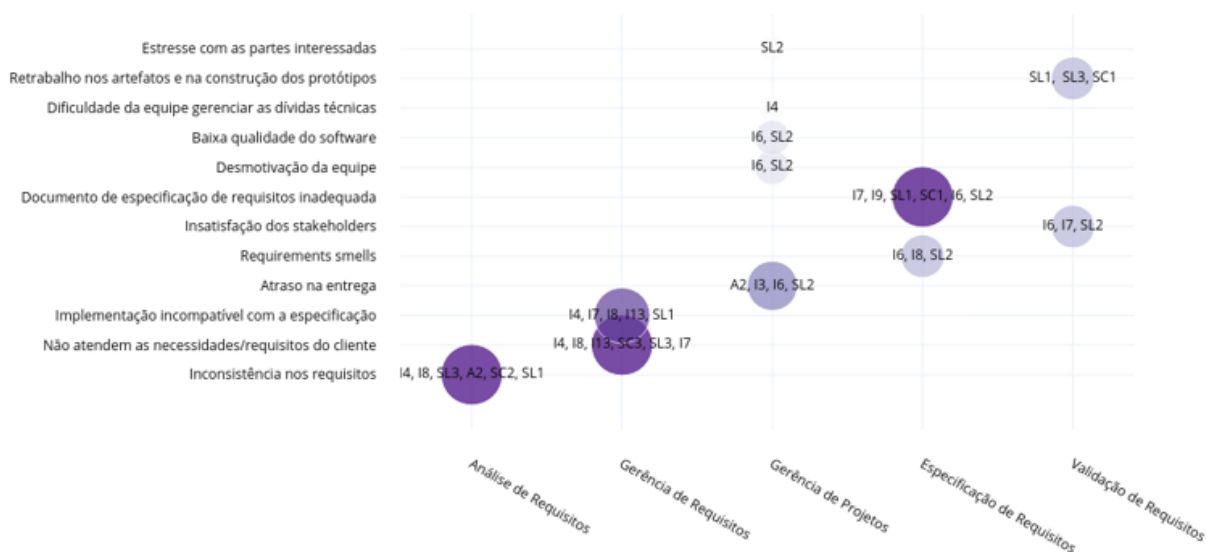


Fonte: Elaborado pelo autor (2021)

4.1.3.3 QP 2.1 Quais as consequências ocasionadas pelas dívidas de requisitos na Engenharia de Requisitos?

A Figura 7 demonstra as 12 consequências reveladas conforme os estudos correspondentes e a sua classificação.

Figura 7 – Consequências que caracterizam dívida de requisitos

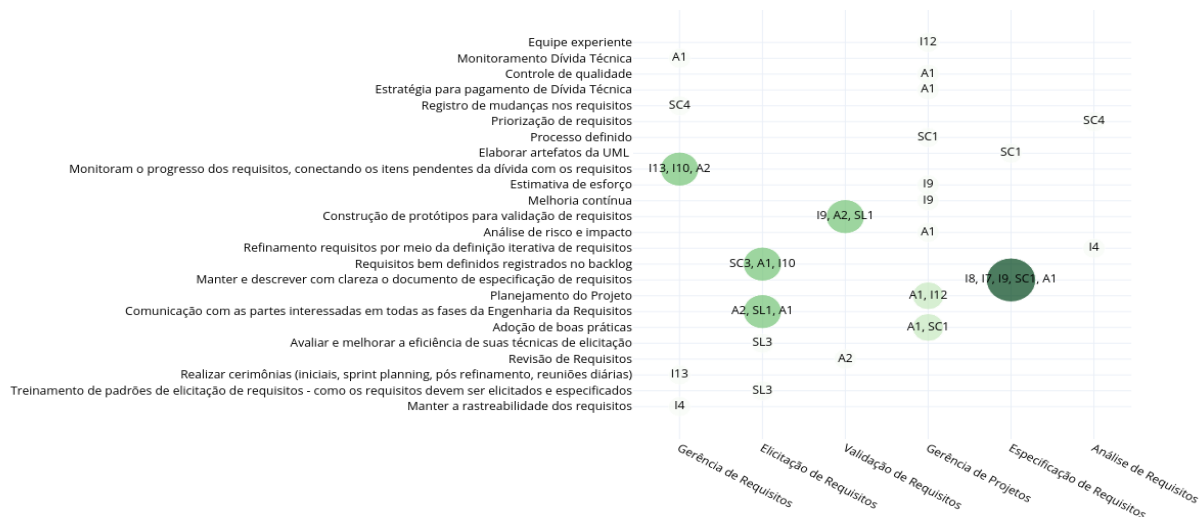


Fonte: Elaborado pelo autor (2021)

4.1.3.4 QP 3. Quais metodologias e/ou práticas são empregadas em organizações ágeis que podem minimizar as dívidas de requisitos geradas durante a Engenharia de Requisitos?

A Figura 8 evidencia as práticas que podem mitigar as causas das dívidas de requisitos.

Figura 8 – Práticas para minimizar as causas e dívidas de Requisitos



Fonte: Elaborado pelo autor (2021)

4.1.3.5 QP 4. Como as organizações ágeis monitoram suas dívidas de requisitos?

Entre os 21 estudos resultantes do MSL, apenas 4 mencionaram aspectos relacionados ao monitoramento das dívidas técnicas nas suas organizações ágeis, conforme apresentado na Figura 9. As etapas para gestão da dívida sinalizadas pelos estudos não estão direcionadas especialmente para dívida de requisitos, apenas de ordem geral, ou seja, para dívida técnica. No entanto, optou-se por mantê-las, pois entende-se que, independentemente do tipo da dívida, as dívidas precisam ser gerenciadas e as suas etapas podem ser adaptadas conforme a necessidade do tipo da dívida.

Figura 9 – Etapas para monitorar a dívida técnica



Fonte: Elaborado pelo autor (2021)

4.1.4 Discussões

Os resultados obtidos no MSL permitiram observar que os estudos analisados apresentam preocupação em entender melhor o cenário que envolve dívida técnica de modo geral, considerando os tipos de dívidas, conforme a etapa do processo de desenvolvimento de software, dentre eles, a dívida de requisitos, pois ainda é considerada uma lacuna a ser explorada.

Diante do exposto, os estudos evidenciaram várias abordagens como, por exemplo: a preocupação em investigar ações preventivas para evitar a ocorrência de dívidas técnicas e os impedimentos de realizar essas ações (Freire *et al.*, 2020); propor um modelo para conceituar as causas, efeitos, práticas e pagamento de dívidas técnicas (Mandić; Taušan; Ramač, 2020); revelar as causas que levam à ocorrência de dívidas técnicas e investigar se essas causas podem ser evitadas (Rios; Spinola; Mendonça; Seaman, 2018a); apresentar a conceituação de dívida de requisitos incluindo a dívida contraída durante a identificação, formalização e implementação de requisitos (Lenarduzzi; Fucci, 2019).

O estudo realizado por Lenarduzzi e Fucci (2019) é o único direcionado especificamente à dívida de requisitos neste MSL, pois os demais abordam nas suas pesquisas a dívida de modo geral, incluindo evidências de diferentes tipos de dívidas, assim como a de requisitos. Essa constatação reforça a necessidade de ter uma compreensão mais aprofundada

sobre o cenário que envolve a dívida de requisitos considerando as evidências relacionadas às causas, consequências e práticas voltadas às dívidas de requisitos reveladas nas atividades que envolvem a Engenharia de Requisitos. A dívida de requisitos “implementação incompatível com a especificação” reforça o conceito de dívida de requisitos proposto por Ernst (2012) e o tipo 2, “implementação incompatível”, referente à dívida de requisitos sugerida por Lenarduzzi e Fucci (2019).

Relacionado à questão de pesquisa QP1 que visa demonstrar as estratégias abordadas para identificar dívida de requisitos, alguns estudos relatam que existe a preocupação de entender a familiaridade com a dívida técnica por parte dos profissionais de organizações ágeis, como as organizações caracterizam as dívidas de requisitos e quais os meios utilizados para identificá-las. Observou-se que vários profissionais ainda desconhecem o termo de dívida técnica. As estratégias relatadas estão condicionadas ao auxílio da documentação, utilização de analisadores de requisitos e ferramentas de análise para avaliar o nível da dívida técnica, entre elas a de requisitos e clareza na definição da dívida e dos seus tipos de dívidas.

Conforme apresentado na QP 2 referente às causas que incorrem em dívida de requisitos e a QP 2.1 associada às consequências, foi possível perceber que os estudos apresentam causas que incorrem em dívidas de requisitos e superficialmente relatam as consequências, mas não relacionam as causas com as consequências apresentadas.

A partir das evidências apresentadas, pertencentes a QP 2 e 2.1 constatou-se que, embora vários dos estudos envolvidos no MSL explorem a dívida técnica para compreender os elementos que causam as dívidas e meios para evitá-las, os relatos indicam a preocupação com esses fatores fragmentados nos diferentes tipos de dívidas, incluindo a dívida de requisitos, porém, não vinculam as causas às consequências e não descrevem como essas causas poderiam ser minimizadas. São informações descentralizadas, reveladas nos diferentes estudos do MSL.

Alguns estudos apresentam as práticas/estratégias (QP 3) utilizadas para minimizar a dívida de requisitos, mas não direcionam quais as causas determinadas práticas podem minimizar. É possível perceber, por meio do registro das práticas, que, embora os estudos não especifiquem quais as causas podem ser minimizadas pelos relatos nos estudos, as práticas fortalecem a sua necessidade para mitigar algumas causas de dívida de requisitos.

Foram revelados também, como as organizações monitoram as suas dívidas (QP 4), destacando as etapas utilizadas ou sugeridas para gerenciá-las. Resumidamente as etapas adotadas ou sugeridas para gerenciar as dívidas concentram-se em: Identificação da dívida técnica, prevenção, monitoramento e pagamento da dívida.

Nota-se que os estudos estão dedicando esforços para entender o contexto de dívida de requisitos demonstrando uma inquietação com essas dívidas nas organizações, e os aspectos relacionados a ela. Contudo, é visível que tais estudos não estabelecem uma conexão entre os elementos desse universo, sejam estratégias para identificá-las, causas, consequências, e quais as práticas poderiam minimizar as dívidas para auxiliar as organizações nesse processo. Esse esforço reforça a importância de oferecer um mecanismo que centralize todo esse conhecimento produzido, no cenário da prática e na literatura, que incentive e auxilie as organizações a identificarem as causas e consequências das dívidas de requisitos e quais as práticas poderão mitigá-las, melhorando assim, o seu processo de Engenharia de Requisitos.

Entende-se que essa disseminação do conhecimento produzido sobre a dívida de requisitos é extremamente relevante para o estado da arte, pois no cenário de desenvolvimento de software evitar que dívidas sejam contraídas no início do desenvolvimento de software, é impedir que custos adicionais se acumulem ao longo do desenvolvimento, gerando juros significativos até o seu pagamento. Em algum momento essas dívidas precisam ser pagas para manter a integridade dos requisitos do projeto em longo prazo, atender os objetivos dos stakeholders e não comprometer a qualidade do software desenvolvido (Kruchten; Nord; Ozkaya, 2012).

4.1.5 Unificação do MSL e Visão Prática - causas das dívidas de requisitos, consequências das dívidas de requisitos e práticas

Esta seção unifica e sintetiza o resultado da pesquisa apresentada no capítulo 3 (visão prática), e do mapeamento sistemático da literatura abordado anteriormente.

Considerando as abordagens envolvidas nas questões de pesquisa de ambos os estudos, visão prática e MSL, pretende-se evidenciar as causas de dívida de requisitos, consequências e práticas em organizações que adotam abordagens ágeis de desenvolvimento de software, desta forma, priorizou-se apenas as evidências que refletem diretamente na dívida de requisitos.

4.1.5.1 Cenários dos estudos

Os resultados apresentados nas próximas seções, respondem às questões de pesquisa envolvidas nos capítulos 3 e 4 da tese, ou seja, em ambos os estudos (visão prática e MSL): “QP 1: Quais as causas que geram dívidas de requisitos, QP 1.1: Quais as consequências que caracterizam as dívidas de requisitos?”; e “QP 2: Quais as práticas podem minimizar as dívidas de requisitos?”

O “cenário 1”, representa o capítulo 3, direcionado a uma visão prática obtida por meio de entrevistas e questionário realizados com colaboradores de organizações que seguem o desenvolvimento ágil de software e o “cenário 2” contemplado a partir do mapeamento sistemático da literatura, apresentado no capítulo 4 da tese.

A identificação dos estudos apresentados nas próximas seções se configura da seguinte forma: Os estudos iniciados pela letra “E” representam os estudos que evidenciam as causas das dívidas, as dívidas e práticas resultantes da visão prática (capítulo 3), ou seja, do cenário 1. Os estudos que iniciam com letras diferentes da letra “E” representam o cenário 2 (MSL).

4.1.5.2 Visão prática (cenário 1) e MSL (cenário 2): Unificação das causas, consequências e práticas voltadas às dívidas de requisitos

Após o levantamento realizado em ambos os cenários, unificou-se as evidências das causas das dívidas, das consequências e práticas voltadas às dívidas de requisitos. Para essa unificação considerou-se a semelhança e duplicidade entre as evidências desses 3 fatores (causas, consequências e práticas) reveladas nos estudos dos dois cenários. O resultado da unificação das causas, consequências e práticas é apresentado no Apêndice K.

Após a unificação, foi elaborado o cruzamento entre as causas que ocasionam as dívidas de requisitos e as consequências, bem como a relação entre as práticas que podem mitigar as dívidas de requisitos e as causas. Entende-se que é preciso evitar as causas das dívidas de requisitos, por meio de práticas que possam tornar isso possível, para que as dívidas de requisitos, ou seja, as consequências das causas, sejam minimizadas.

4.1.5.3 Apresentação dos resultados: consequências (dívidas de requisitos) versus causas que geram essas dívidas de requisitos - unificação entre MSL e Visão Prática - unificação entre MSL e Visão Prática

Para realizar a conexão entre as evidências das consequências e das causas, considerou-se inicialmente as relações reveladas na visão prática (cenário 1) e, posteriormente, observou-se nos estudos do cenário 2 que tais evidências eram reforçadas pela literatura. O resultado dessa relação, considerando os dois cenários, é apresentado no Apêndice L.

4.1.5.4 Apresentação dos resultados: causas versus práticas voltadas à dívida de requisitos - unificação entre MSL e Visão Prática

Após unificar as causas e práticas voltadas à dívida de requisitos dos cenários 1 e 2 foi realizado o cruzamento das causas das dívidas de requisitos e as práticas que poderão mitigá-las, o qual pode ser observado no Apêndice M.

Destaca-se que as práticas apresentadas no cenário 1 que não foram confirmadas pelo cenário 2 (MSL), ou seja, não possui evidência em estudos do MSL, foram reforçadas por outros estudos localizados na literatura e são representados pelo prefixo “L” seguido de um dígito como, por exemplo, “L1”, na última coluna e podem ser visualizados no Apêndice N.

4.1.6 Contribuições

O MSL corroborou com o estudo apresentado no capítulo 3, seguindo o mesmo propósito desenvolvido por pelo pesquisador, porém, obtido através de outras fontes e técnicas.

As evidências, voltadas às dívidas de requisitos, são muito descentralizadas, pois foram encontradas em estudos que abordam a dívida técnica de forma geral, pois ainda existem poucos estudos direcionados a esse tipo específico de dívida.

Sobretudo, uma solução que possibilite às organizações ágeis compreenderem adequadamente a existência das dívidas de requisitos, as causas das dívidas de requisitos, bem como as práticas para minimizá-las, poderia ser uma alternativa relevante para os profissionais dessas organizações melhorarem os seus processos e a qualidade dos seus requisitos, incorporando tais práticas ao seu processo de desenvolvimento de forma efetiva.

Em relação à gestão da dívida de requisitos, observou-se que ainda há poucos estudos voltados à dívida de requisitos que mencionam a gestão desse tipo de dívida. Sobretudo, entender as atividades da gestão da dívida no contexto dos requisitos pode auxiliar na identificação, monitoramento e prevenção de dívidas. A ausência das atividades que constituem a gestão da dívida fortaleceu a necessidade de realizar um estudo sistemático focado especificamente na gestão da dívida de requisitos.

A gestão da dívida de requisitos poderá apoiar a identificação, monitoramento e prevenção das dívidas de requisitos referente aos eixos: causas, dívidas de requisitos e práticas que podem minimizá-las. Optou-se por realizar um estudo terciário sobre a gestão da dívida de requisitos para auxiliar as organizações ágeis a compreenderem as atividades ou áreas envolvidas no processo da gestão da dívida e as práticas relacionadas a cada área. Essas informações poderão auxiliar as organizações a gerir melhor suas dívidas, mitigá-las e preveni-las.

4.2 GESTÃO DA DÍVIDA TÉCNICA: ESTUDO TERCIÁRIO

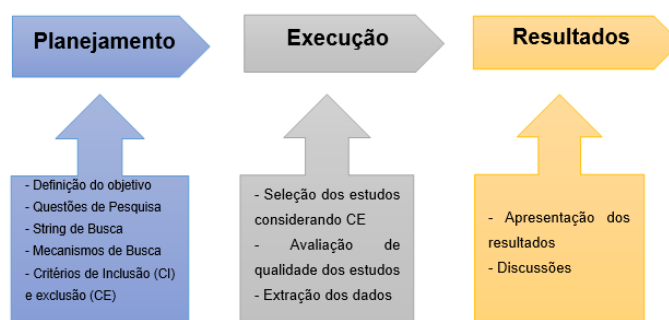
A dívida técnica, contraída ao longo do desenvolvimento de software, independentemente de como ela ocorra, intencional ou não, requer a necessidade de compreendê-la e gerenciá-la para não comprometer a qualidade do software e garantir sua evolução (Melo *et al.*, 2022).

A seção 4.1 desta pesquisa sintetizou as causas, consequências e práticas voltadas à dívida de requisitos, no entanto, além da compreensão dos problemas que levam às dívidas de requisitos e formas de prevenção, é fundamental entender como essas dívidas podem ser gerenciadas, pois a partir da gestão é possível visualizá-las, monitorá-las, pagá-las e documentá-las, dificuldade enfrentada pelos profissionais que atuam no desenvolvimento de software. Realizar um estudo terciário sobre a gestão da dívida de requisitos é interessante e necessário, pois não existe um estudo que envolva um consenso entre as atividades que constituem o processo de gestão de dívida técnica ou dívida de requisitos, apenas abordagens que sugerem certas atividades, mas não seguem uma padronização.

Contudo, os estudos que envolvem a gestão de dívida de requisitos ainda é uma área aberta, pois os pesquisadores têm se dedicado às pesquisas que envolvem a gestão da dívida técnica de maneira geral, como abordado em Jeronimo e Travassos (2022), sem aprofundar atividades que envolvem o tipo de dívida mencionado. Sendo assim, o estudo terciário foi direcionado à gestão da dívida técnica, pois entende-se que é possível adaptar a gestão da dívida técnica para qualquer tipo de dívida. Optou-se pelo estudo terciário voltado à gestão da dívida técnica, pois há diversos estudos secundários já realizados.

O estudo terciário segue as recomendações de Kitchenham e Charters (2007) e está organizado conforme a Figura 9.

Figura 9 – Etapas do Estudo Terciário



Fonte: Elaborado pelo autor (2023)

4.2.1 Planejamento

Nesta seção é apresentado o protocolo para o estudo terciário.

4.2.1.1 Objetivo e questões de pesquisa

O objetivo do estudo terciário é compreender o processo de gestão da dívida técnica, especialmente a gestão da dívida de requisitos, bem como as atividades ou áreas que a compõem. O estudo terciário responderá às questões de pesquisa apresentadas no Quadro 8.

Optou-se por incluir nas questões de pesquisa os termos dívida de requisitos e dívida técnica, pois como a dívida de requisitos ainda é uma área com poucos estudos, é considerada um tipo de dívida técnica e acredita-se que estudos que retratam a gestão da dívida técnica de forma geral, podem fazer referência às atividades ou áreas vinculadas diretamente a algum tipo de dívida, como a de requisitos.

Quadro 8 – Questões de pesquisa do Estudo Terciário

Questões de pesquisa
QP 1. Quais são as atividades ou áreas da gestão da dívida técnica, especialmente a gestão da dívida de requisitos?
QP 2. Quais as práticas sugeridas para atividades ou áreas da gestão da dívida de requisitos e/ou dívida técnica?
QP 3. Quais as limitações enfrentadas na adoção das atividades ou áreas da gestão da dívida de requisitos e/ou dívida técnica?
QP4. Quais ferramentas podem auxiliar na gestão da dívida de requisitos e/ou dívida técnica?

Fonte: Elaborado pelo autor (2023)

4.2.1.2 String de busca e fontes de pesquisa

Para este estudo, a string de busca é constituída dos termos de pesquisa do Quadro 9 compondo a string genérica apresentada no Quadro 10. A definição dos termos de pesquisa foi baseada em algumas revisões sistemáticas (Alves *et. al.* 2016; Li; Avgeriou; Liang 2015) e estudos terciários (Jerônimo; Travassos, 2022; Rios; Mendonça; Spínola, 2018).

Quadro 9 – Termos de pesquisa

Termos da pesquisa
Dívida Técnica: (technical debt OR requirements debt)
Gestão da Dívida Técnica: (manage OR managing OR management OR monitoring OR practices OR techniques)
Estudo secundário: (systematic literature review OR systematic mapping OR systematic study)

Fonte: Elaborado pelo autor (2022)

Quadro 10 – String genérica

String genérica
("technical debt" OR "requirements debt") AND ("manage" OR "managing" OR "management" OR "monitoring" OR "practices" OR "techniques") AND ("systematic literature review" OR "systematic mapping" OR "systematic study")

Fonte: Elaborado pelo autor (2022)

4.2.1.3 Mecanismos de busca

Neste estudo, foram utilizadas busca automatizada e manual. As fontes de busca automatizada foram definidas seguindo recomendações de Kitchenham e Charters (2007), destacando: *ACM Digital Library*; *IEEEExplore*; *Science Direct*; *Springer Link*; e *Scopus*. A string foi ajustada conforme o formato delimitado pelos mecanismos de busca utilizados. A busca manual foi realizada na *International Conference on Technical Debt – Tech Debt*, edições dos anos de 2018, 2019, 2020, 2021 e 2022; e no *International Workshop on Managing Technical Debt* (MTD), edições de 2013, 2014, 2015 e 2016. As escolhas da busca manual justificam-se pela área de abordagem, dívida técnica e gestão da dívida técnica. O número dos artigos retornados conforme o mecanismo de busca, encontra-se no Apêndice O.

4.2.1.4 Critérios de inclusão e exclusão

Neste estudo foram considerados os artigos que atendem aos seguintes critérios de inclusão (CI) e desconsiderados os que atendem aos critérios de exclusão (CE), ambos apresentados no Quadro 11.

Quadro 11 - Critérios de inclusão e exclusão

Critérios de Inclusão
CI ₁ - Papers escritos em inglês;
CI ₂ - Papers revisados por pares;
CI ₃ - Papers que apresentam estudos secundários, como revisão sistemática da literatura e mapeamento sistemático da literatura, que abordam gestão da dívida técnica direcionado às atividades ou fases que a contemplam, especialmente gestão da dívida de requisitos.
CI ₄ - Papers que abordam práticas ou técnicas para as fases da gestão da dívida;
CI ₅ - Publicações que possuem o documento completo.
Critérios de Exclusão
CE ₁ - Papers que não estão em inglês;
CE ₂ - Papers com estudo primário ou terciário serão excluídos;
CE ₃ - Papers duplicados localizados em mais de um mecanismo de busca ou que tenham o mesmo resultado publicado em mais de um veículo;
CE ₄ - Papers que o documento completo não possa ser localizado ou que contenham apenas resumo ou resumo expandido (com até uma página);
CE ₅ - Papers fora do contexto da dívida técnica ou dívida de requisitos, gestão da dívida técnica e estudo secundário.

Fonte: Elaborado pelo autor (2021)

4.2.2 Execução

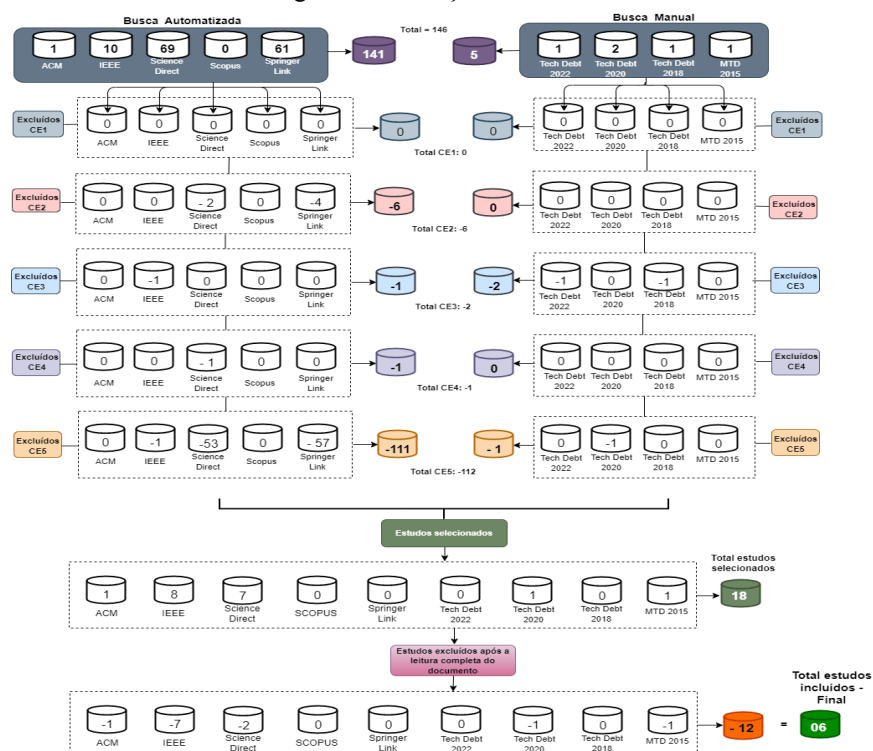
A execução é uma etapa realizada seguindo o protocolo definido no planejamento, seção 4.2.1.

4.2.2.1 Seleção dos estudos

O processo de seleção dos estudos ocorreu em janeiro de 2023 e foi executado conforme descrito no protocolo apresentado na seção 4.2.1 deste estudo terciário. A string de busca foi adaptada e executada nos mecanismos de busca automatizados selecionados e os termos ajustados nas referências da busca manual. Ao final da execução, foram obtidos 146 estudos.

A seleção dos estudos foi apoiada nos critérios de exclusão demonstrados na seção 4.2.1.2, resultando em 18 selecionados (16 de bases automatizadas e 2 da busca manual). Após a leitura do documento completo dos artigos selecionados inicialmente, 12 foram excluídos nesse momento, totalizando assim 06 estudos incluídos na extração de dados, destacados no Quadro 15. A Figura 10 apresenta o resultado do processo de seleção dos estudos. Destaca-se que a busca manual em Tech Debt nas edições de 2021 e 2019 e no MTD, edições 2016, 2014 e 2013 foi realizada, porém não foi identificado nenhum estudo e, por esta razão, não estão demonstrados na Figura 10.

Figura 10 – Seleção dos estudos



Fonte: Elaborado pelo autor (2023)

Quadro 12 – Estudos incluídos no estudo terciário

Identificação do Estudo	Referência	Título	Mecanismo de busca
SD1	MELO et.al (2022)	Identification and measurement of Requirements Technical Debt in software development: A systematic literature review	Science Direct
SD3	AMPATZOGLOU et.al (2015)	The financial aspect of managing technical debt: A systematic literature review	Science Direct
SD5	BEHUTIYE et.al. (2017)	Analyzing the concept of technical debt in the context of agile software development: A systematic literature review	Science Direct
SD6	ALVES et.al (2016)	Identification and management of technical debt: A systematic mapping study	Science Direct
SD7	LI; AVGERIOU; LIANG (2015)	A systematic mapping study on technical debt and its management	Science Direct
II	ALBUQUERQUE et.al (2022)	Comprehending the Use of Intelligent Techniques to Support Technical Debt Management	IEEE

Fonte: Elaborado pelo autor (2023)

4.2.3 Avaliação de Qualidade

A avaliação de qualidade dos estudos selecionados foi apoiada pelo método DARE (Centre for reviews and dissemination, 2014), o qual é recomendado para avaliar a qualidade de estudos sistemáticos da literatura na área de Engenharia de Software (Rios; Mendonça; Spínola, 2018), e propõe 5 questões para determinar os critérios de qualidade da avaliação de qualidade, conforme Apêndice P. Para determinar a pontuação na avaliação da qualidade, foram consideradas recomendações de Kitchenham et al. (2009), a qual deve estar em um intervalo de 0 a 5.

A avaliação da qualidade, apresentada no Quadro 13, permite observar que 66% dos estudos atingiram pontuação máxima de qualidade. Os demais estudos, ficaram acima da média de pontuação. Desta forma, todos os estudos selecionados são considerados nos resultados.

Quadro 13 – Avaliação de qualidade dos estudos incluídos

Identificação do estudo	Referência	Avaliação de qualidade - Critérios de Qualidade					
		CQ1	CQ2	CQ3	CQ4	CQ5	Score
SD1	MELO et.al (2022)	S = 1 pt	S = 1 pt	S = 1 pt	S = 1 pt	S = 1 pt	5 pts
SD3	AMPATZOGLOU et.al (2015)	S = 1 pt	S = 1 pt	S = 1 pt	S = 1 pt	S = 1 pt	5 pts
SD5	BEHUTIYE et.al. (2017)	S = 1 pt	S = 1 pt	S = 1 pt	S = 1 pt	S = 1 pt	5 pts
SD6	ALVES et.al (2016)	S = 1 pt	S = 1 pt	S = 1 pt	N = 0 pt	S = 1 pt	4 pts
SD7	LI; AVGERIOU; LIANG (2015)	S = 1 pt	S = 1 pt	S = 1 pt	S = 1 pt	S = 1 pt	5 pts
II	ALBUQUERQUE et.al (2022)	S = 1 pt	S = 1 pt	P = 0,5 pt	N = 0 pt	P = 0,5 pt	3 pts

Fonte: Elaborado pelo autor (2023)

4.2.4 Resultados

Os resultados obtidos após a extração dos dados¹² e que respondem as questões de pesquisa: “QP 1. Quais são as atividades ou áreas da gestão da dívida técnica/gestão da dívida

¹² A planilha com a extração dos dados pode ser consultada no link: <https://docs.google.com/spreadsheets/d/1QNCSWSILeo-UzaqGgNA2INm8tFEfsyjYDpDV6iJ28L0/edit?usp=sharing>

de requisitos”, “QP 2. Quais as práticas sugeridas para as atividades ou áreas da gestão da dívida de requisitos?”, QP 3. Quais as limitações enfrentadas na implementação das atividades ou áreas da gestão da dívida de requisitos”, QP 4. Quais as ferramentas de apoio para gestão de dívida de requisitos?” podem ser visualizados nos Apêndice Q.

4.2.5 Análise dos dados e discussões

Albuquerque *et al.* (2022) destaca que a gestão da dívida técnica consiste no conjunto de atividades que permitem às organizações evitarem dívidas técnicas no processo de desenvolvimento de software e, no caso da sua existência, torná-las visíveis e controladas. Essa referência às atividades da gestão da dívida técnica não rotula a nenhum tipo específico de dívida técnica, tratando a gestão da dívida de modo geral, sem especializá-la (Jeronimo; Travassos, 2022).

A “QP 1 - Quais são as atividades ou áreas da gestão da dívida técnica, especialmente dívida de requisitos?” revelou, por meio dos estudos secundários SD1, SD3, SD5, SD6, SD7, I1, as atividades que normalmente fazem parte do processo de gestão da dívida técnica. No entanto, observou-se que não existe um consenso sobre as atividades ou áreas que devem compor o processo de gestão da dívida, e nem mesmo uma classificação que as padronize. Embora as atividades: Identificação, Mensuração/Quantificação, Reembolsos/Pagamento e Monitoramento foram apontados pelos seis estudos (SD1, SD3, SD5, SD6, SD7, I1), as atividades Priorização (SD1, SD3, SD6, SD7, I1), Comunicação (SD6, SD7, I1), Representação/Documentação (SD6, SD7, I1) e Prevenção (SD7, I1) não foram mencionados por alguns estudos, justificando tal afirmação. Considerando o resultado obtido na QP 1, acredita-se que as atividades citadas pelos seis estudos são fundamentais para a gestão da DT, e as demais podem ser caracterizadas como atividades ou áreas de apoio para a gestão da DT.

Relacionado à gestão da dívida de requisitos, essa é uma temática que precisa ser melhor investigada, pois apenas o estudo secundário SD1 retrata atividades vinculadas a esse tipo de dívida. Dessa forma, considerou-se nos estudos selecionados, abordagens direcionadas à gestão da dívida técnica e que podem ser adaptadas à gestão da dívida de requisitos.

Considerando a “QP 2. Quais as práticas/técnicas sugeridas para as atividades ou áreas da gestão da dívida técnica?”, apenas dois estudos secundários (SD1 e SD7) destacaram práticas que podem ser implementadas conforme determinada atividade ou área que compõe a gestão da dívida técnica. Foi possível observar que as práticas extraídas dos estudos envolvem unicamente as atividades da gestão da dívida e não estão associadas ao tipo de dívida incorrida,

porém, isso não impede que as práticas sugeridas sejam adaptadas nas organizações. Sobretudo, a empresa deve ter a compreensão do objetivo da atividade da gestão da dívida e da prática para adequá-la conforme sua necessidade e alcançar a efetividade esperada no gerenciamento das suas dívidas técnicas, podendo combinar várias práticas para a mesma área da gestão.

A “QP 3. Quais as limitações enfrentadas na adoção das atividades ou áreas da gestão da dívida técnica?”, evidenciou dificuldades ou limitações associadas à implantação das áreas/fases da gestão da dívida. Neste estudo terciário, apenas dois estudos as destacaram.

Na “QP 4. Quais as ferramentas podem auxiliar na gestão da dívida de requisitos?” almejava-se identificar ferramentas que auxiliassem na gestão da dívida de requisitos, mas o estudo SD1 voltado à gestão da dívida de requisitos ressaltou que não existem ferramentas para a gestão desse tipo de dívida. No entanto, o estudo SD7 identificou uma ferramenta voltada à dívida de requisitos, a ferramenta RE-KOMBINE que visa identificar e medir a dívida de requisitos.

4.3 LIMITAÇÕES E AMEAÇAS À VALIDADE

Para realizar os estudos sistemáticos da literatura (MSL e estudo terciário) observou-se as etapas de planejamento, execução e análise dos resultados. A seleção dos estudos secundários procurou explorar as bases de dados voltadas à busca automatizada conforme recomendação de Kitchenham e Charters (2007). Mesmo com os cuidados necessários para a realização de qualquer estudo, limitações e ameaças à validade podem ocorrer, as quais são abordadas abaixo seguindo as orientações de Ampatzoglou *et al.* (2019):

Viés da inclusão/exclusão: Corresponde a problemas que podem ocorrer durante a seleção do estudo na aplicação dos critérios de inclusão e exclusão, pois podem ocorrer critérios conflitantes ou muito gerais. Para minimizar tal ameaça, procurou-se definir os critérios e segui-los conforme protocolo dos estudos, revisar os critérios de inclusão/exclusão, e apoiar a revisão dos critérios de inclusão/exclusão por um especialista.

Construção da String de Busca: Consiste em problemas que podem acontecer no momento da construção da string de busca, pois os pesquisadores podem utilizar termos muito genéricos e abertos retornando um número significativo de estudos (relevantes e irrelevantes) ou perder o foco da busca, deixando de considerar estudos importantes. Para minimizar esta ameaça, a construção da string foi baseada em strings de estudos existentes conforme pontuadas nos estudos sistemáticos (MSL e estudo terciário) e refinada conforme foi sendo

testada, seguindo termos que representam sinônimos da pesquisa. A string de cada estudo foi compartilhada e revisada por outro pesquisador até a sua definição.

Viés da extração de dados: Está relacionado a problemas que podem surgir durante o processo da extração dos dados, causados pela má elaboração das questões, questões muito abertas, desviando da proposta abordada no protocolo. Também foi utilizado um formulário contendo as questões que apoiaram a extração de cada estudo.

Seleção de bibliotecas digitais: Ameaça referente a problemas voltados à escolha dos mecanismos de busca, os quais podem ser muito amplos ou pouco confiáveis, resultando em estudos irrelevantes. Para mitigar essa ameaça, os mecanismos de busca automatizada adotados seguiram as recomendações de Kitchenham e Charters (2007) em ambos os estudos. Especialmente no estudo terciário foram realizadas também, busca manual, em todas as edições da *International Conference on Technical Debt – TechDebt* e do *International Workshop on Managing Technical Debt* (MTD).

Viés do pesquisador: Compreende a interpretação do pesquisador durante o processo de extração dos dados, pois o pesquisador pode ter uma compreensão equivocada referente a alguma questão envolvida na extração. Para mitigar esta ameaça, procurou-se ser fiel ao protocolo estabelecido. A extração dos dados também foi revisada por outro pesquisador.

Cobertura das questões de pesquisa: Ameaça relacionada ao conjunto de questões de pesquisa que podem desviar o seu foco não atendendo o objetivo dos estudos. Isso pode ocorrer por serem questões muito genéricas ou manter uma decomposição equivocada para as questões. Para minimizar essa ameaça, foram adotadas questões específicas para direcionar ao objetivo de cada estudo, as quais também foram revisadas por outro pesquisador.

4.4 UNIFICAÇÃO DAS ÁREAS DA GESTÃO DA DÍVIDA (ESTUDO TERCIÁRIO) COM O RESULTADO DA UNIFICAÇÃO DAS CAUSAS, DÍVIDAS DE REQUISITOS E PRÁTICAS DO MSL E VISÃO PRÁTICA

Finalizado o estudo terciário, o resultado da junção do MSL e da Visão Prática, seção 4.1.6, foi unificado às práticas deste estudo conforme cada área¹³ e as evidências das práticas. O Apêndice R é representado por oito grandes áreas da gestão da dívida técnica conforme observado na primeira coluna.

¹³ Optou-se por denominar a “área” da gestão da dívida, por considerar que algumas desse conjunto não se caracteriza exatamente como uma fase, mas como área de apoio à gestão. Sendo assim, entende-se que o termo área poderia representar os dois termos sem prejuízo algum para a gestão da dívida técnica.

Cada área é organizada por um conjunto de práticas que definem, de forma geral, o que deve ser atendido. Essas práticas são especializadas em sub práticas genéricas e específicas, conforme observada na terceira coluna, as quais possuem um conjunto de atividades que caracterizam as ações que devem ser desenvolvidas para atender determinada sub prática e, conseqüentemente, a prática relacionada. As sub práticas genéricas são recomendadas para todas as causas das dívidas de requisitos (ausência de destaque em cores).

As sub práticas específicas estão destacadas por cores, sendo que a cor azul representa as práticas específicas resultantes somente do estudo terciário, a cor cinza evidencia as práticas resultantes da unificação das práticas conforme seção 4.1.6.2 e aquelas que não possuem cor são práticas comuns a qualquer tipo de dívida (Apêndice S).

4.5 CONTRIBUIÇÕES

Compreender as áreas da gestão da dívida é fundamental para adequá-las numa possível gestão das dívidas de requisitos incorridas no processo de desenvolvimento de software pois, além de reconhecê-las, é necessário gerenciá-las para ter um controle maior sobre as dívidas incorridas, seu pagamento, monitoramento e prevenção, por exemplo. Concluída a extração e análise dos dados, observou-se que embora os estudos recomendem áreas e fases para a gestão da dívida técnica, pesquisas direcionadas à gestão da dívida de requisitos são escassas, oportunizando investimentos em pesquisas nessa área.

Um ponto relevante é a ausência de uma classificação para orientar a implementação do processo de gestão da dívida técnica, pois as áreas obtidas no resultado da extração revelam que não há um consenso entre os estudos, passo importante para contribuir com esta pesquisa e com as demais pesquisas desenvolvidas por outros autores. A partir deste estudo será possível representar todo o conhecimento produzido sobre as causas, conseqüências e práticas voltadas à dívida de requisitos, bem como as áreas necessárias e recomendadas para a gestão dessas dívidas.

Para organizar e estruturar as informações obtidas nos estudos sistemáticos voltados à dívida de requisitos, será necessário o apoio de um sistema de representação e organização de conhecimento recomendado em pesquisas com um grande volume de informações (FOGL, 1979), que permita reunir o conhecimento abstraído e facilitar o acesso e compartilhamento com a comunidade científica e com profissionais da área de tecnologia, especialmente na área de desenvolvimento de software (Netto, 2017).

Optou-se em representar o volume de informações por meio de uma Ontologia, pois segue uma especificação formal de um conceito e é representado por um vocabulário especializado em um determinado domínio (Gómez-Pérez; Fernández-López; Corcho, 2004). Sobretudo, auxilia a captura, representação, busca, armazenamento e padronização do conhecimento, por meio de um vocabulário consistente e completo (Guarino, 1998, apud Rios *et al.*, 2014).

5 PROPOSTA DE UMA ONTOLOGIA DE DOMÍNIO VOLTADA ÀS DÍVIDAS DE REQUISITOS

A ontologia proposta é o primeiro artefato da contribuição desta tese e visa organizar e estruturar os fundamentos teóricos voltados à dívida de requisitos, considerando conceitos, organização hierárquica entre eles, suas relações e axiomas para formalizar as definições e relações (Hendler; Feigenbaum, 2001), assim como objetos e seus atributos, sintetizando o modelo que representa as informações obtidas, formando a base de conhecimento da área envolvida (Fernández-Pérez; Gómez - López; Corcho, 2004).

A escolha pela representação das informações, por meio de uma ontologia nesta tese, justifica-se pelas seguintes razões: I) determinar um vocabulário comum especializado voltado à dívida de requisitos, caracterizando o domínio desta ontologia; II) sistematizar o conhecimento, pois requer um conjunto de conceitos que permitem descrever elementos fundamentais da área base; III) padronizar, pois promove a relação do vocabulário comum e a sistematização do conhecimento para representar os conceitos que seguem uma padronização facilitando a sua compreensão (Mizoguchi, 2001).

A construção da Ontologia, denominada de OntoReD, reúne um vocabulário comum sobre dívidas de requisitos sintetizando o conhecimento adquirido através da GT descrito no capítulo 3, bem como as evidências obtidas por meio dos estudos secundário e terciário (capítulo 4) a qual, revela um grande volume de dados, que poderá ser compartilhado e complementado por pesquisadores. Dessa forma, novas informações poderão ser agregadas à OntoReD, contribuindo para a mitigação das dívidas de requisitos, a evolução do processo de requisitos e melhoria da qualidade dos requisitos especificados.

A Ontologia proposta também foi motivada para servir de fonte de dados e aproximar o conhecimento, voltado às dívidas de requisitos, das organizações ágeis. A OntoReD permitirá às organizações visualizarem os fatores que ocorrem durante o processo de requisitos e que podem se transformar em dívidas técnicas, neste caso, dívidas de requisitos. Por meio dessa Ontologia as organizações poderão acessar as causas que levam às consequências das dívidas e compreender melhor o seu cenário de requisitos, bem como as práticas que poderão mitigá-las. A recomendação das práticas fornecerá o seu detalhamento e orientações sobre elas.

Espera-se que a Ontologia possa potencializar a qualidade dos requisitos elicitados e especificados em organizações ágeis, evitando o retrabalho, reduzindo custo adicional e, consequentemente, melhorando a qualidade dos produtos de software desenvolvidos.

5.1 APRESENTAÇÃO DA ONTORED

A OntoReD é uma ontologia de domínio que consiste em representar na sua base de informações, as evidências sobre as causas que geram dívidas de requisitos, as consequências ocasionadas pelas causas, práticas que podem mitigá-las, bem como aspectos voltados à gestão da dívida.

A partir das informações organizadas na ontologia é possível definir um vocabulário comum que consiste em um conjunto de elementos designados como classes, relações, instâncias e axiomas (Fernández-López; Gómez-Pérez; Corcho 2004), os quais fazem parte dos elementos que embasam a construção da OntoReD e são conceituados a seguir:

- **Classes:** As classes representam os conceitos de um domínio, podem conter outras classes, seguindo uma taxonomia e possuem um conjunto de instâncias, suas relações, atributos. “CausaReD” e “ConseguenciaReD” são exemplos de classes.
- **Relações:** Determinam as propriedades entre as classes e, consequentemente, entre suas instâncias. As relações representam a semântica de uma ontologia. Como exemplos, apresenta-se “ocasiona” e “mitigadoPor”.
- **Instâncias:** Caracterizam os objetos de uma classe e a base de conhecimento de uma ontologia. “InconsistenciaRequisitos” é um exemplo de uma instância da classe exemplificada, “ConseguenciaReD”.
- **Axiomas:** Representam as regras que permitem associar os relacionamentos entre as instâncias.

O desenvolvimento da OntoReD é amparado pela METHONTOLOGY, considerando os elementos fundamentais descritos anteriormente.

5.2 CONSTRUÇÃO DA ONTORED BASEADA NA METHONTOLOGY

A *Methontology* é uma metodologia para construção de ontologias proposta por Fernández-López, Gómez-Pérez e Juristo (1997) em nível de conhecimento. Essa metodologia é embasada nas principais atividades voltadas ao processo de desenvolvimento de software e em metodologias da engenharia do conhecimento (Fernández-López; Gómez-Pérez; Corcho, 2004).

Optou-se pela *Methontology*, por fornecer orientação sobre o processo de construção da ontologia (Iqbal *et al.*, 2013) e consiste nas seguintes atividades: Especificação, conceitualização, formalização, implementação e manutenção; por ser referência no

desenvolvimento de ontologias voltadas à área da engenharia de software; e, também, porque várias ontologias propostas na área de Computação se apoiam na *Methontology* para sua construção (Vale-Pereira; Fernández 2019; Alsanad; Chikh; Mirza, 2019).

5.2.1 Especificação

Nesta etapa é realizado um estudo sobre o domínio que se deseja representar na ontologia, considerando os seus conceitos fundamentais, o qual já foi realizado e está descrito nos capítulos 3 e 4 desta tese. A especificação apresenta informações gerais da ontologia proposta, considerando o domínio, o tipo de ontologia que está inserida, a proposta da ontologia, bem como o escopo e base de conhecimento que fundamentam a ontologia. O quadro 48 apresenta o cenário geral da Ontologia.

Quadro 14 - Especificação da OntoReD

Especificação da Ontologia
Domínio: Dívida de Requisitos
Conceitualização por: Viviane Duarte Bonfim; Fabiane Barreto Vavassori Benitti
Implementação por: Viviane Duarte Bonfim
Proposta: A ontologia OntoReD é uma ontologia de domínio que consiste em manter uma base de informações sobre causas das dívidas de requisitos, consequências das dívidas de requisitos, práticas que podem mitigar as causas e consequentemente suas consequências, as quais são organizadas por meio de áreas da gestão da dívida. O objetivo da OntoReD é aproximar a base de informações de profissionais de organizações ágeis e permitir uma melhor compreensão das causas das dívidas de requisitos e suas consequências considerando seu cenário atual e, a partir disso, ter acesso às recomendações de práticas que poderão mitigar as causas de suas dívidas de requisitos.
Escopo: Os termos que fundamentam a ontologia são apoiados no resultado dos estudos mencionados nas fontes de conhecimento.
Fonte de Conhecimento: Unificação do conhecimento obtido por meio da visão prática, realizada com apoio de profissionais da área de desenvolvimento de organizações ágeis e da literatura por meio de mapeamento sistemático da literatura e estudo terciário (capítulos 3 e 4 da tese)

Fonte: Adaptado de Fernández-López, Gómez-Pérez, Juristo (1997)

5.2.2 Conceitualização

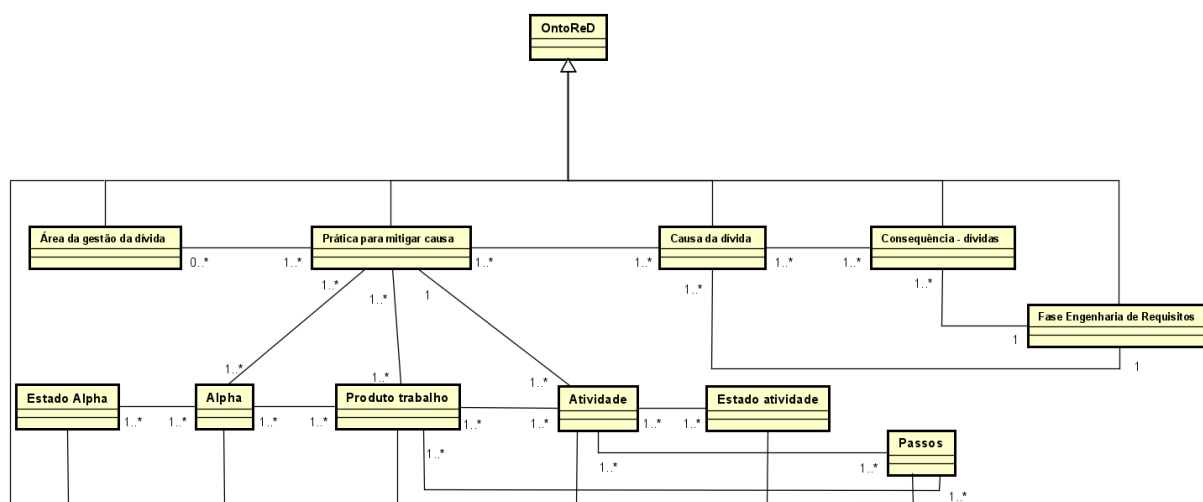
A etapa de conceitualização é destinada à organização e estruturação das informações amparadas pela aquisição do conhecimento obtido por meio das fontes mencionadas na especificação e evidenciado pelo modelo conceitual.

Visando atender o modelo conceitual, Fernández, Gómez-Pérez e Juristo (1997) sugerem elaborar um glossário com os termos relevantes, considerando os conceitos, suas relações, propriedades das instâncias e propriedades envolvendo o domínio da ontologia. Os termos do Glossário são demonstrados no Apêndice S.

A figura 8 retrata a associação entre os conceitos representados pelas classes que estão presentes na OntoReD. Na conceitualização dos conceitos procurou-se preservar os termos obtidos nos estudos realizados que deram origem a eles.

O modelo conceitual, Figura 11, é transformado em um modelo formalizado que será implementado por meio da linguagem da Ontologia (Pérez; López; Corcho, 2004).

Figura 11 – Estrutura dos conceitos da OntoReD



Fonte: Elaborado pelo autor (2023)

O detalhamento da prática segue uma abordagem constituída de alguns elementos inspirados na Biblioteca de Ivar Jacobson (2018) apoiados pela linguagem Essence (OMG, 2018). A estrutura da prática está representada no Apêndice T e são compostos pelos seguintes elementos:

- Ícone que representa o elemento;
- Identificação do elemento, e o que se pretende do elemento. Um elemento pode ser um produto de trabalho, atividade ou alpha;
- Atividade: representa a atividade que pode ser desenvolvida relacionada a determinado elemento.
- Alpha: um alpha representa e mantém o estado de algum elemento, aspecto ou abstração e que possui um estado visando compreender o estado de progresso;
- Prática: Prática a qual o elemento está associado;
- Referências: Recursos externos representam o conteúdo de apoio sobre determinado elemento. Dependendo do elemento, é possível manter outras informações como, por exemplo, referências bibliográficas;
- Espaço da atividade: O espaço da atividade está condicionado ao contexto que a atividade está inserida;

- Estado alpha: o estado alpha representa as etapas necessárias para atender determinado alpha.

5.2.3 Formalização e implementação

A fase de Formalização é responsável pela transformação do modelo conceitual para um modelo formal, que pode ser compreendido por máquinas, e é apoiada por um editor para construção de ontologia, implementando-a (Fernández-López; Gómez-Pérez; Corcho, 2004). Optou-se pelo editor denominado Protégé para implementar a OntoReD.

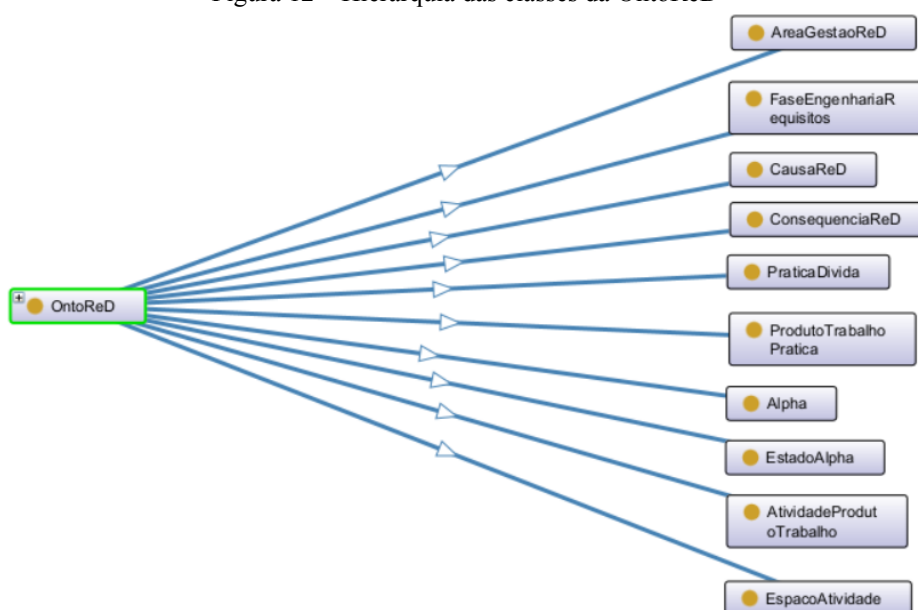
Por convenção, recomenda-se letra maiúscula para denominar o nome das classes e, quando o conceito possui mais de uma palavra para denominar a classe deve-se concatená-la, iniciando cada palavra com a letra maiúscula (HORRIDGE; BRANDT 2011). Por exemplo, o conceito “causa da dívida de requisitos”, está representado na OntoReD como: “CausaReD”.

Para representar as relações associativas da Ontologia, utiliza-se propriedades por meio de dois componentes: *Object Properties* que determina o relacionamento entre dois conceitos e *Data Properties* que caracteriza as propriedades dos dados, ou seja, atributos das instâncias das classes. Ambas as propriedades recomendam-se iniciar sempre por letra minúscula (Mirroir, 2020).

5.2.3.1 Construção das classes

A hierarquia das classes da OntoReD pode ser visualizada na Figura 12.

Figura 12 – Hierarquia das classes da OntoReD

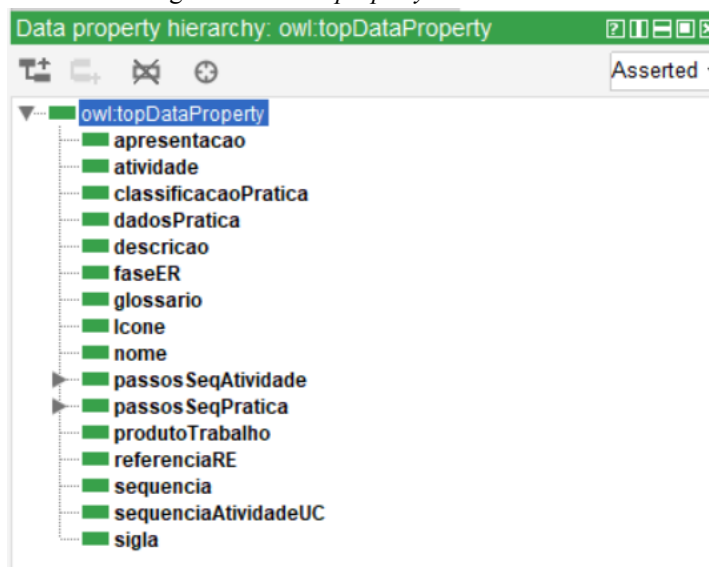


Fonte: Capturada do Protégé – OntoReD (2023)

5.2.3.3 Definição dos atributos de instância da OntoReD – Data Properties

A definição dos atributos das instâncias consiste em representar os dados necessários para cada instância de um conceito (classes), os quais podem ser visualizados na figura 14.

Figura 14 – Data property das instâncias



Fonte: Capturada do Protégé – OntoReD (2023)

5.2.3.4 Axiomas

Os axiomas especificam as regras assumidas como verdadeiras em um determinado domínio, nas quais associa-se os relacionamentos entre os conceitos (classes) e consequentemente, suas instâncias, além de fornecer características descritivas e lógicas para os conceitos.

Para definir os axiomas relevantes na Ontologia, a metodologia prevê especificar as seguintes informações: nome, descrição, expressão lógica que descreve formalmente o axioma, conceitos, atributos e relações voltadas às variáveis presentes no axioma, as quais estão sintetizadas no exemplo apresentado no Quadro 15 (Fernández-Pérez; Gómez- López; Corcho, 2004).

Quadro 15 – Axioma voltado às práticas que podem mitigar as causas das dívidas de requisitos

Nome da regra	Práticas que podem mitigar as causas das dívidas de requisitos considerando a área de gestão da dívida
Descrição	Esta regra é responsável por apresentar as práticas que podem mitigar as causas das dívidas de requisitos, bem como suas consequências, categorizadas por áreas da gestão da dívida

Expressão	if [CausaReD] (?Y) [ocasiona] [ConsequenciaReD] (?X) and [CausaReD] (?Y) [mitigadoPor] [PraticaDivida] (?Z) and [AreaGestaoReD] (?W) [eConstituidaDe] [PraticaDivida] (?Z) then [nome] (?W, “nome da instância”, ?Z, “nome da instância”)
Conceitos	CausaReD, ConsequenciaReD, PraticaDivida, AreaGestaoDividaReD
Atributos	Nome da Área da Gestão da Dívida, Nome da Prática da dívida de requisitos
Relação	ocasiona, mitigadoPor, eConstituidaDe
Variáveis	?W ?Z

Fonte: Elaborado pelo autor (2023)

5.2.3.5 Definição das instâncias dos conceitos da OntoReD

As instâncias dos conceitos da OntoReD formam a sua base de conhecimento e deve manter um nome, o conceito o qual faz parte e os valores de atributos. O Quadro 16 apresenta um exemplo de uma instância e o conceito correspondente.

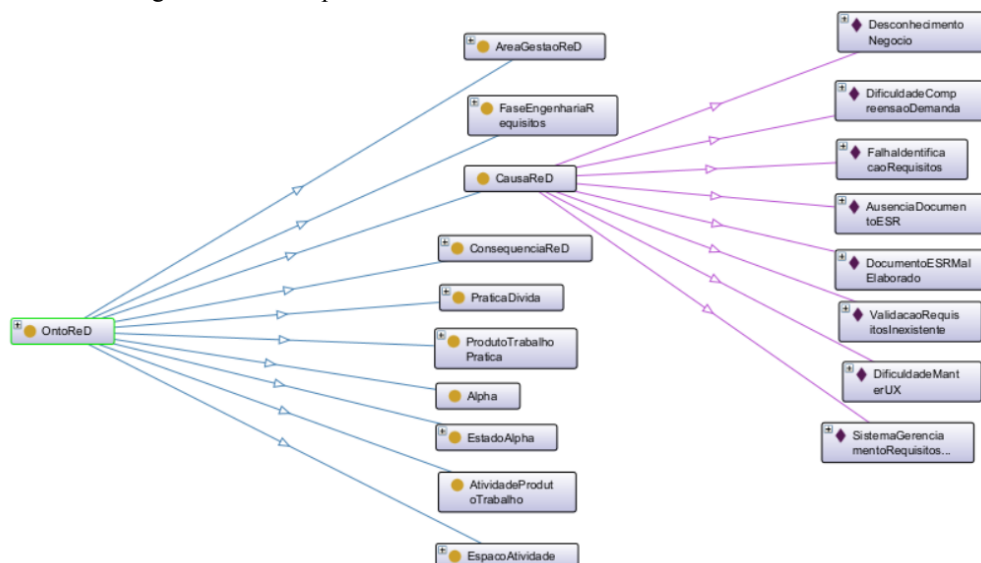
Quadro 16 – Exemplos de Instâncias da OntoReD

Nome da Instância	Nome do Conceito
ValidacaoRequisitosInexistente	CausaReD
ImplementacaoIncompativelComEspecificacao	ConsequenciaReD
RevisaoRequisitos	PraticaDivida
Prevencao	AreaGestãoReD
ValidaçãodeRequisitos	FaseEngenhariaRequisitos

Fonte – Elaborado pelo autor (2023)

Visando demonstrar um exemplo de instâncias de um determinado conceito, apresenta-se na Figura 15 o conceito “CausaReD” e suas instâncias correspondentes.

Figura 15 – Exemplos de Instâncias do conceito “CausaReD” da OntoReD



Fonte: Capturada do Protégé (2023)

Após as atividades orientadas ao desenvolvimento previstas na *METHONTOLOGY*, são propostas atividades de suporte, como a atividade de aquisição de conhecimento (realizado nos capítulos anteriores), integração (não aplicada), avaliação, documentação e gerenciamento de configuração (Iqbal *et al.* 2013; Fernández-Pérez; Gómez- López; Corcho, 2004).

5.2.4 Avaliação da OntoReD

A avaliação proposta na *Methontology*, sinaliza um documento que registre a avaliação, no qual o pesquisador deve descrever como a Ontologia foi avaliada, as técnicas utilizadas, o tipo de erros encontrados e as fontes de conhecimento utilizadas na avaliação e verificar a sua consistência (Fernández-López; Gomés-Pérez; Juristo, 1997).

Na OntoReD foi realizada a sua avaliação técnica pela ferramenta, e seguindo essa característica, a sua consistência foi verificada na opção *Debugger* no Protégé, sendo que todas as inconsistências apontadas foram resolvidas. Para auxiliar na verificação da consistência (*consistency checking*), foi utilizado o mecanismo de inferência (*Reasoner*) Hermit disponível no Protégé para essa finalidade (HORRIDGE, 2008) e que permite a especificação e execução de consultas na ontologia. A avaliação da OntoReD também foi realizada por meio do conhecimento de especialista.

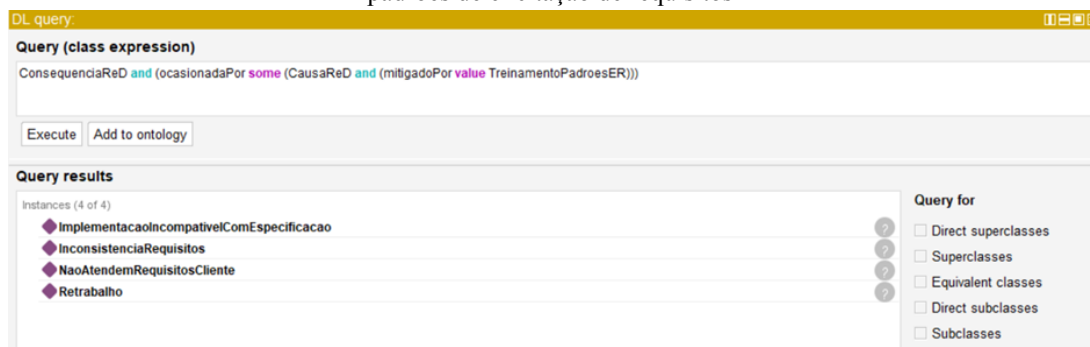
Para demonstrar visualmente a consistência e a coerência da OntoReD, apresenta-se exemplo de consulta como prova de conceito. A consulta, apresentada no Quadro 17, foi definida inicialmente em linguagem natural e, após, convertida em uma DL Query e executada no Protégé por meio do mecanismo de inferência. O resultado da DL Query pode ser visualizado na Figura 16.

Quadro 17 - Exemplo de consulta da OntoReD em DL Query

<i>Consulta 1</i>	<i>Quais são as consequências da dívida de requisitos que podem ser mitigadas pela prática “Treinamento de padrões de elicitação de requisitos”?</i>
<i>DL Query 1</i>	<i>ConsequenciaReD and (ocasionadaPor some (CausaReD and (mitigadoPor value TreinamentoPadroesER)</i>
<i>Resultado 1</i>	<i>Figura 16</i>

Fonte: Elaborado pelo autor (2023)

Figura 16 – Resultado Consulta 1: Consequências da dívida de requisitos mitigadas pela prática “Treinamento de padrões de elicitação de requisitos”



Fonte: Capturada do Protégé (2023)

5.2.5 Documentação da OntoReD

A documentação da OntoReD é pautada, muitas vezes, no código da Ontologia, pois não há diretrizes que formalizam essa documentação, incluindo publicações realizadas em anais e conferências para oficializar essa documentação. No entanto, na *Methontology* propõe-se a atividade de documentação durante todo o processo de desenvolvimento da ontologia (Fernández-López; Gómez-Pérez; Juristo, 1997).

A documentação da OntoReD é pautada no registro de todas as atividades do processo de desenvolvimento da Ontologia, contendo na sua implementação, anotações e um glossário para facilitar a compreensão durante a sua manutenção e estará disponível em <https://github.com/vivi-duarte/OntoReD.git> após a sua finalização. O conhecimento que apoia a OntoReD está fundamentado nas pesquisas realizadas nos capítulos 3 e 4 desta tese.

5.3 CONTRIBUIÇÃO

A OntoReD poderá atuar como um facilitador para organizações ágeis identificarem e reconhecerem seu cenário atual voltado às dívidas de requisitos, compreenderem as causas e acessarem recomendações de práticas que podem mitigar as dívidas de requisitos. Por meio da Ontologia será possível demonstrar as lacunas (gaps) existentes no processo de desenvolvimento de software voltada às atividades associadas aos requisitos e que resultam em dívidas de requisitos e podem comprometer a qualidade dos requisitos.

Para permitir às organizações ágeis a interação com as informações representadas na OntoReD, sem a necessidade de entender o funcionamento da Ontologia e realizar consultas voltadas as lacunas que levam às dívidas, será necessário um mecanismo que facilite essa

interação e proporcione às organizações o acesso de forma simples, ágil e de fácil compreensão.

Para atender a necessidade relatada, propõe-se uma abordagem que visa a recomendação de práticas para mitigar e prevenir as dívidas de requisitos e suas causas em organizações ágeis.

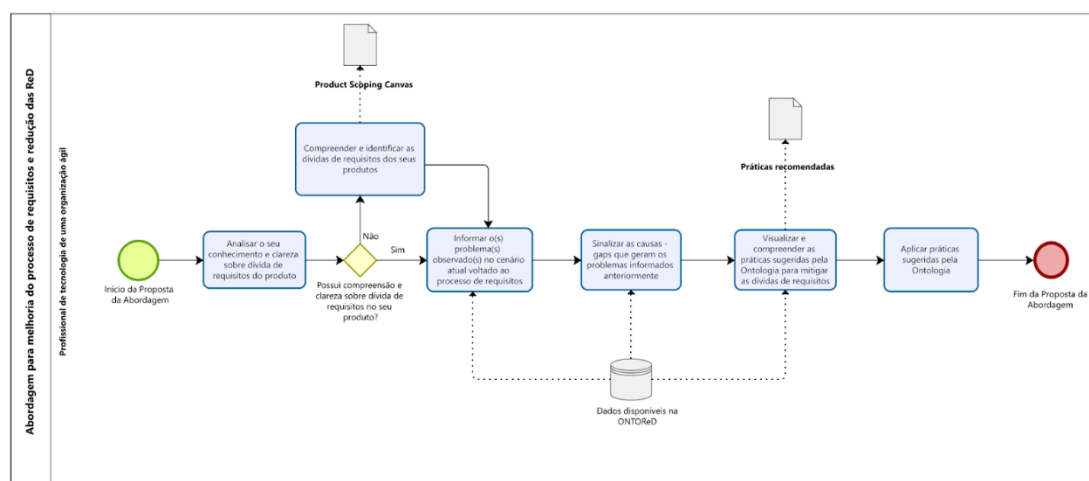
6 PRACTICESFORRED - PROPOSTA DE UMA ABORDAGEM VOLTADA À RECOMENDAÇÃO DE PRÁTICAS PARA REDUÇÃO DAS DÍVIDAS DE REQUISITOS EM ORGANIZAÇÕES ÁGEIS

A PracticesforReD consiste em uma abordagem que permite às organizações ágeis compreenderem seu cenário atual, voltado às causas e dívidas de requisitos e, a partir dessas informações, recomenda-se um conjunto de práticas para mitigá-las.

6.1 ESTRUTURA DA PRACTICESFORRED

A abordagem é organizada por meio de um conjunto de passos apresentados na Figura 17, e podem ser acessados por meio de um profissional da organização. Para interagir com a PracticesforReD, o profissional deve considerar, como referência, um determinado produto de software que será o ponto de apoio da organização para utilizar a abordagem.

Figura 17 - Proposta da abordagem



Fonte: Elaborado pelo autor (2023)

De modo geral, será possível identificar a existência das dívidas de requisitos de um determinado produto de software; informar os gaps - problemas voltados ao processo de requisitos observados pelo profissional (dívidas de requisitos), sinalizar as causas (causas das dívidas de requisitos) dos gaps revelados anteriormente e, a partir dessas informações, o profissional da organização ágil poderá visualizar a recomendação das práticas para os gaps e causas informadas. Para finalizar, prover a compreensão das práticas e seus elementos para ter condições de aplicá-las ao seu processo de requisitos. O processo possui auxílio do *Product*

*Scoping Canvas*¹⁴ (Pereira-Vale; Ochoa; Perovich, 2023), etapas da *gap analysis* (Marra *et al.*, 2018), e dos dados sobre dívida de requisitos presentes na OntoReD.

A *gap analysis* (Marra *et al.*, 2018), permitirá às organizações ágeis, observarem seu cenário atual, sinalizando seus problemas (gaps) no processo de requisitos, caracterizando as dívidas de requisitos e suas causas.

6.1.1 Passo 1 - Analisar o seu conhecimento e clareza sobre dívida de requisitos do produto

O processo inicia com uma análise da percepção do profissional, bem como da organização ágil, sobre o seu conhecimento envolvendo o conceito de dívida de requisitos e a clareza sobre as dívidas de requisitos identificadas no seu produto de software. Caso o profissional não tenha clareza sobre o conceito de dívida de requisitos e a existência de dívidas desse tipo no seu produto, recomenda-se a utilização do *Product Scoping Canvas* (Pereira-Vale; Ochoa; Perovich, 2023) seguindo inicialmente o passo 1.2. Ao contrário, é possível seguir para o passo 2.

6.1.2.1 Passo 1.2 - Compreender e identificar as dívidas de requisitos dos seus produtos

A compreensão e percepção da existência de dívidas de requisitos é amparado pelo *Product Scoping Canvas*, proposto por Pereira-Vale; Ochoa; Perovich (2023) e consiste em uma técnica colaborativa que permite à organização ágil realizar o mapeamento, acompanhamento do seu produto e a sinalização da existência da dívida de requisitos. O *Product Scoping Canvas* contribuirá para a compreensão das dívidas de requisitos, facilitando ao profissional a execução dos passos seguintes do processo proposto.

6.1.2 Passo 2 - Informar o(s) gaps - problema(s) observado(s) no cenário atual voltado ao processo de requisitos

A compreensão na existência de dívida de requisitos em determinado produto, permitirá à organização ágil uma percepção do seu cenário atual, detectando os problemas (gaps) existentes no seu processo de requisitos. Os problemas são pré-estabelecidos, pois são fornecidos pela OntoReD e possuem uma denominação e uma descrição breve para orientar melhor o profissional da organização ágil.

¹⁴ Exemplo do Product Scoping Canvas:
https://miro.com/app/board/uXjVMMYh45Y=?utm_source=notification&utm_medium=email&utm_campaign=daily-updates&utm_content=go-to-board&share_link_id=577208534293

A classificação dos gaps (problemas) é apoiada na adaptação da Matriz de GUT, sendo que para cada gap (problema) reconhecido e informado no seu cenário atual, deve ser sinalizada a tendência (T) do problema e a urgência (U) para solucioná-lo. Após informar a tendência e a urgência, calcula-se o grau de relevância (G) do problema, ou seja, a gravidade, por meio da representação “G x U x T” (Souza, 2022). A matriz adaptada está disponível no Apêndice Z.

A Matriz de GUT, também denominada de matriz de priorização, caracteriza-se como uma ferramenta para a análise dos problemas existentes apoiada por três parâmetros: gravidade, urgência e tendência. A matriz auxiliará na prioridade das ações que devem ser realizadas considerando os parâmetros mencionados (Alves *et al.*, 2016).

- O Grau de relevância (G) demonstra a prioridade para solucionar o problema. O grau de relevância é medido pela UxT e possui os seguintes graus de relevância para solucionar o problema: Baixo, Médio e Alto. Baixo, significa que existe um problema e a organização deverá monitorar e avaliar a sua solução; Médio indica que a organização deve ficar em alerta e estabelecer uma previsão para solucionar o problema; Alto representa a solução imediata devido a sua gravidade.
- Urgência (U): impacto e intensidade que o problema pode gerar caso não seja resolvido, seguindo uma escala de leve, moderado e crítico.
- Tendência ou frequência (T): Representa o potencial de crescimento do problema, seguindo a escala de ocorrência: Eventualmente, Parcialmente e Frequentemente.

Como resultado do passo 2, o profissional poderá visualizar o seu cenário atual com o resumo das informações reveladas anteriormente: Problema identificado, sua tendência, urgência e gravidade.

6.1.3 Passo 3 - Sinalizar as causas dos gaps informados anteriormente

Após os gaps (problemas) serem informados, o profissional poderá especificar, dentre as causas pré-estabelecidas, as que geram cada um dos gaps, considerando as causas que ocorrem no seu cenário atual e que ocasionam os gaps (problemas - consequências das dívidas)).

As causas que geram tais problemas são obtidas por meio dos dados da OntoReD, e caracterizam as causas das dívidas de requisitos. Essas causas apresentam os gaps que precisam ser mitigados para reduzir as dívidas de requisitos.

6.1.4 Passo 4 - Visualizar e compreender as práticas sugeridas pela Ontologia para mitigar as dívidas de requisitos

No passo 4, o profissional poderá visualizar as práticas recomendadas para mitigar as causas das dívidas que evidenciam os gaps no processo de requisitos, as quais também estão disponíveis e formalizadas na OntoReD. A recomendação das práticas é direcionada especificamente para as causas e problemas informados pelo profissional, anteriormente.

Dessa forma, a visualização das práticas recomendadas se configura seguindo a hierarquia: Área da Gestão -> Prática Geral -> Práticas comuns e/ou específicas. O grau de relevância do problema é representado juntamente à prática recomendada para o profissional ter clareza das práticas que devem ser prioridade na sua adequação ao processo de requisitos da organização ágil. O grau de relevância que sinaliza a prioridade da prática, é representado por uma cor, correspondente à escala: baixo, médio e alto, informada no passo 2, sendo: a cor verde representa o grau de relevância baixo; a cor azul representa o grau médio; e a cor vermelha o grau alto.

6.1.5 Passo 5 – Elementos que detalham as práticas sugeridas pela Ontologia - OntoReD

De forma resumida, os elementos que detalham a prática são:

- **Nome:** Identificação da prática;
- **Apresentação:** Elemento que fornece uma descrição geral da prática;
- **Produto de trabalho:** informa o(s) artefato(s) que contempla a prática;
- **Atividade:** descreve a(s) ação (ações) que deve(m) ser realizada(s) no produto de trabalho;
- **Espaço atividade:** Representa o contexto que a atividade está inserida;
- **Alpha:** Define o elemento que contempla o produto de trabalho;
- **Estado Alpha:** Descreve sequencialmente os estados do Alpha;
- **Referências de apoio:** Apresentam informações, como, por exemplo, referências da literatura, que podem complementar as orientações presentes nos elementos da prática;
- **Contexto da ReD:** Sintetiza os problemas e as causas das dívidas de requisitos informados pelo profissional e que estão relacionados à prática recomendada;
- **Glossário:** Apresenta os termos e suas denominações voltados ao contexto da prática.

Cada prática contempla um detalhamento que visa a orientação dos elementos que compõem a prática e o que deve ser desenvolvido para ser atendida. Esse detalhamento segue a notação da Linguagem Essence (OMG, 2018) e as especificidades dos elementos é baseada na Biblioteca de Práticas (Jacobsen, 2018).

6.2 SOLUÇÃO COMPUTACIONAL DE APOIO À PRACTICESFORRED

A solução computacional atuará como um meio de interação entre o profissional e a ontologia, seguindo os passos contemplados na abordagem proposta. Interagindo com essa solução, os profissionais das organizações ágeis terão acesso às recomendações e orientações de práticas que poderão mitigar e evitar as causas e as consequências (gaps) das dívidas de requisitos. Alguns exemplos dessa interação podem ser visualizados nas figuras 18, 19, 20, 21 e 22.

A figura 18 representa o que ocorre no passo 2, descrito na seção 6.1.2 e, permite aos profissionais, compreenderem as suas dívidas de requisitos, selecionando os problemas (gaps) identificados no seu produto de software, a tendência da ocorrência e a urgência para solucionar cada problema.

Figura 18 – Protótipo da solução para o passo 2

← VOLTAR

1 Problemas 2 Cenário Atual 3 Causas 4 Práticas AVANÇAR →

Informe o(s) problema(s) gaps que ocorre(m) no produto de software

Análise de Requisitos ^	
<input checked="" type="checkbox"/> <u>Inconsistência dos requisitos</u>	Frequência: Frequente Impacto: Crítico
Inconsistência nos requisitos é caracterizada quando há falhas na comunicação entre o relato das necessidades reveladas pelos stakeholders e a sua compreensão por parte dos profissionais da organização.	
Especificação de Requisitos v	
Validação de Requisitos v	
Gerência de Requisitos v	

Fonte: Elaborado pelo autor (2023)

A Figura 19 sintetiza o cenário atual da organização, amparado pelas informações evidenciadas na interação com a Figura 18.

Figura 19 – Protótipo da solução para o passo 2 - visualizar o cenário atual

← VOLTAR **✓** Problemas **2** Cenário Atual 3 Causas 4 Práticas AVANÇAR →

Consequência(s) voltadas as dívidas de requisitos do produto

Análise de Requisitos

Inconsistência dos requisitos

Frequência
Frequente

X

Impacto
Crítico

=

Grau de Relevância
ALTO

Inconsistência nos requisitos é caracterizada quando há falhas na comunicação entre o relato das necessidades reveladas pelos stakeholders e a sua compreensão por parte dos profissionais da organização.

Fonte: Elaborado pelo autor (2023)

Na figura 20 são apresentadas as causas que ocasionam os problemas informados anteriormente (Figura 18). Caso o profissional entenda que alguma das causas não interfere no problema informado, é possível desmarcá-la.

Figura 20 – Protótipo da solução para o passo 3

← VOLTAR **✓** Problemas **✓** Cenário Atual **3** Causas 4 Práticas AVANÇAR →

Causas das dívidas de requisitos

Causas que ocasionam - Inconsistência dos requisitos

☒ Dificuldade na Compreensão da Demanda
Dificuldade na compreensão da demanda pode ocorrer porque o cliente não consegue transmitir exatamente sua necessidade.

☒ Falha na identificação dos requisitos
Falha na identificação dos requisitos funcionais e de qualidade pode ocorrer devido sua volatilidade considerando, requisitos mal formulados, incompletos, com pouca clareza ou ambíguo.

☒ Sistema de gerenciamento de requisitos ausente ou ineficaz
Sistema de gerenciamento de requisitos ausente ou ineficaz consiste na falta ou falha de rastreabilidade dos requisitos, prejudicando a gerência dos requisitos.

Fonte: Elaborado pelo autor (2023)

A Figura 21 apresenta a recomendação das práticas para mitigar as causas e as dívidas de requisitos evidenciadas pelo profissional nas Figuras 18 e 20. Ao lado de cada prática é apresentada a prioridade da sua implantação, representada pela cor correspondente ao grau de relevância do problema identificado pelo profissional na Figura 18.

Figura 21 – Protótipo da solução para o passo 4

← VOLTAR **✓** Problemas **✓** Cenário Atual **✓** Causas **4** Práticas AVANÇAR →

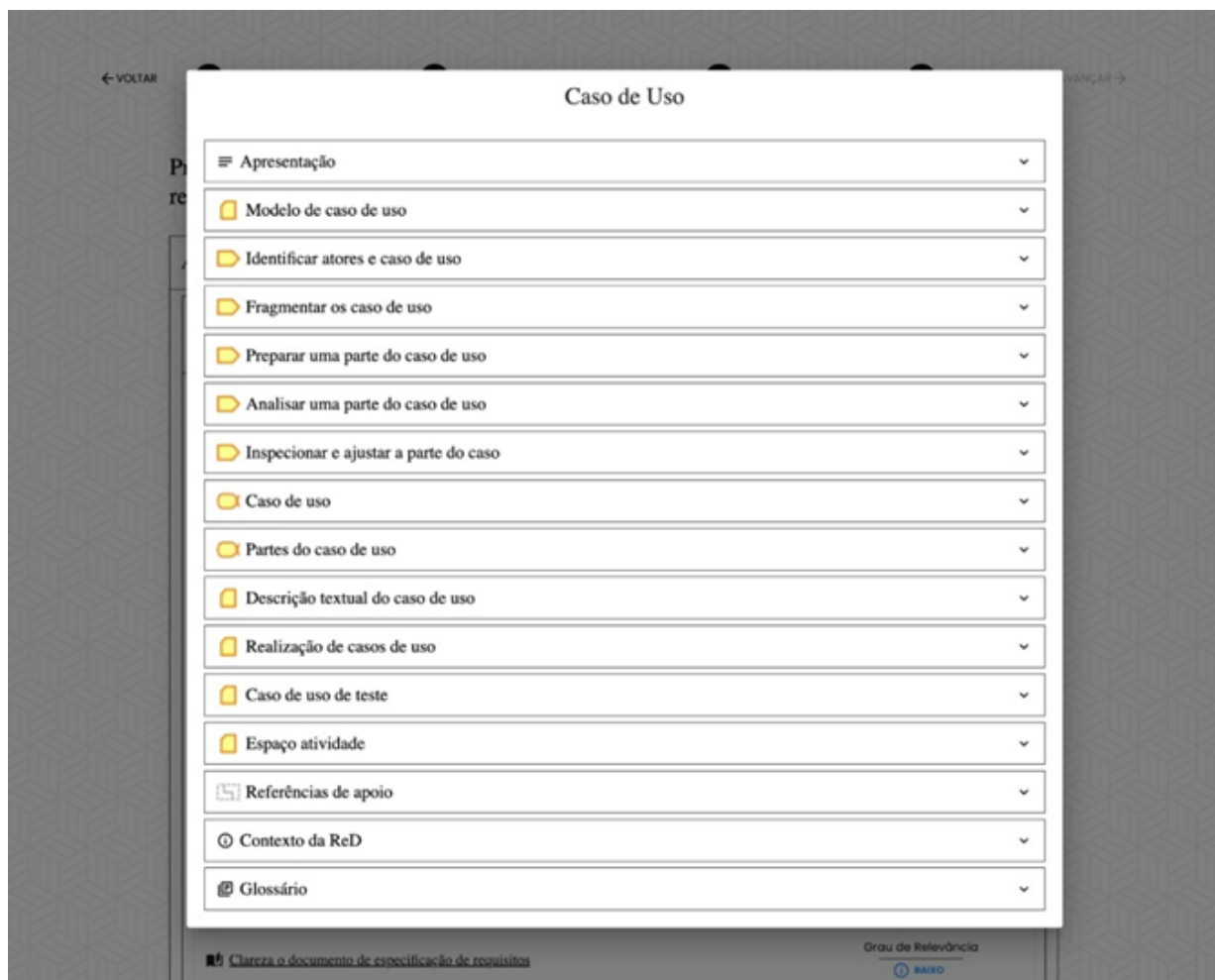
Práticas gerais e específicas para mitigar as causas das dívidas de requisitos e suas consequências

<i>Área da gestão da dívida - Prevenção</i> ^	
<i>Prática - Melhorar os processos de desenvolvimento atuais para prevenir a ocorrência de certos tipos de dívidas técnicas</i>	▼
<i>Área da gestão da dívida - Identificação</i> ^	
<i>Prática - Avaliação do cliente</i>	▼
<i>Prática - Ações preventivas</i>	▼

Fonte: Elaborado pelo autor (2023)

A Figura 22 apresenta os elementos que detalham a prática. Os elementos associados aos ícones de cor amarela e branca, são inspirados na Linguagem Essence (OMG 2018) e a sua organização na Biblioteca de Práticas (JACOBSON 2018). O contexto da ReD apresenta as causas e dívidas de requisitos que podem ser mitigados pela prática.

Figura 22 – Protótipo da solução para o passo 5



Fonte: Elaborado pelo autor (2023)

7 CONSIDERAÇÕES

Considerando a proposta da tese, até o momento do exame de qualificação, alguns objetivos foram atendidos e são destacados a seguir.

Os elementos (consequências, causas e práticas) que envolvem a dívida de requisitos em organizações ágeis foram identificados por meio do estado da prática e do estado da arte. O estado da prática ocorreu com a participação de profissionais de organizações ágeis e foi apoiado pela análise de dados *grounded theory*. Ao final do estudo foi apresentado um modelo teórico representando o resultado das consequências das dívidas de requisitos, as suas causas e as práticas que podem mitigá-las. Neste estudo foi possível observar a falta de compreensão sobre o termo, a falta de conhecimento sobre a existência dessas dívidas e, consequentemente, as consequências causadas por elas. O estado da prática resultou em uma publicação¹⁵ no 34th *International Conference on Software Engineering & Knowledge Engineering* (Seke, 2022), que aconteceu em Pittsburgh, USA.

O estado da arte foi apoiado por dois estudos sistemáticos: o mapeamento sistemático da literatura e o estudo terciário. O mapeamento sistemático da literatura investigou as consequências, as causas das dívidas de requisitos e a relação entre elas, bem como as estratégias ou práticas que podem mitigá-las ou preveni-las. O resultado foi obtido por meio de 21 estudos que comprovaram as informações identificadas no estado da prática e contribuíram para a ampliação das evidências já encontradas anteriormente. Detectou-se, o desconhecimento do termo dívida técnica, bem como, o termo dívida de requisitos por parte dos profissionais, e alguns estudos destacaram que embora alguns profissionais conheçam o termo, não compreendem o significado. Constatou-se ainda, a carência de estudos voltados especificamente para dívidas de requisitos, pois os estudos selecionados para o MSL abordam a dívida técnica de maneira geral e destacam a dívida de requisitos e outros tipos de dívida dentro do contexto geral da DT. Essa carência confirma a necessidade de estudos dedicados à dívida de requisitos para que as organizações ágeis possam compreender melhor essa área e ter condições de buscar alternativas para prevenção e mitigação, no caso da existência da dívida.

O estudo terciário foi proposto para compreender os aspectos da gestão da dívida de requisitos e as práticas associadas a ela. Neste estudo, observou-se que a gestão da dívida não trata especialmente a dívida de requisitos, mas a dívida técnica de modo geral. O estudo terciário também demonstrou que não existe um consenso entre os estudos que padronizam as atividades ou áreas que fazem parte da gestão da dívida técnica. Ao final do estudo, foram

¹⁵ DOI: 10.18293/SEKE2022-114

propostas áreas focadas na gestão da dívida de requisitos que podem auxiliar as organizações ágeis a gerenciarem essas dívidas por meio das evidências encontradas, quando ocorrem.

Finalizados os estudos do estado da prática e do estado da arte, as evidências encontradas foram unificadas, contemplando e fundamentando o resultado desses estudos. Essa unificação sintetiza as consequências das dívidas de requisitos e as causas que as ocasionam, as práticas que podem mitigar e prevenir as causas e consequências das dívidas, assim como as áreas de gestão propostas, associadas às práticas encontradas e aquelas que foram obtidas no estudo terciário.

Para organizar e disponibilizar essas informações, está sendo desenvolvida (em fase de população dos dados) uma ontologia, a OntoReD, que reúne todos os elementos que compõem a dívida de requisitos e que é o núcleo de apoio à abordagem proposta. Para o desenvolvimento da ontologia, foi utilizada a ferramenta Protégé.

Considerando as etapas realizadas, foi proposta uma abordagem, denominada PracticesforReD que auxilia as organizações ágeis compreenderem a existência de dívida de requisitos no seu produto de software e consiste na recomendação de práticas para mitigá-las e preveni-las, conforme seu cenário atual. A abordagem é apoiada pela OntoReD, biblioteca de práticas (Jacobson, 2018), Linguagem Essence (OMG, 2018), *Product Scoping Canvas* (Pereira-Vale; Ochoa; Perovich, 2023), e etapas da *gap analysis* (Marra *et al.* 2018).

Para permitir o acesso à abordagem pelos profissionais das organizações ágeis, está sendo desenvolvida uma solução computacional que apoia a PracticesforReD, na qual os profissionais poderão obter a recomendação das práticas e o seu detalhamento conforme as informações que contemplam o cenário atual.

Pretende-se avaliar a abordagem proposta e a sua solução computacional, por meio do paradigma GQM (Basili; Caldiera; Rombach, 1994), assim que o MVP (produto mínimo viável) estiver disponível.

7.1 ATIVIDADES FUTURAS

As atividades a serem desenvolvidas até a defesa da tese de doutorado concentram-se em:

- Concluir as instâncias na OntoReD;
- Realizar Snowballing voltada a dívida de requisitos;
- Finalizar o desenvolvimento e descrever a solução computacional proposta;
- Apresentar a solução computacional proposta;

- Elaborar o processo de avaliação, baseado em GQM, da Abordagem e solução propostas;
- Realizar a avaliação da Abordagem e solução propostas;
- Elaboração de artigos: essa atividade consiste em obter o retorno da comunidade acadêmica e divulgar os resultados do trabalho.
- Defesa da Tese de Doutorado: evento previsto no programa de Pós-graduação em Ciência da Computação. Para a defesa do doutorado, as atividades pendentes e as sugestões da Banca do Exame de Qualificação serão realizadas.

O Quadro 18 apresenta o cronograma das atividades futuras previstas para esta pesquisa.

Quadro 18 – Cronograma da pesquisa para a tese de Doutorado

Atividade	2023.2	2024.1	2024.2	2025.1	2025.2
Adequações conforme sugestões da banca na qualificação					
Concluir as instâncias na OntoReD;					
Elaborar o processo de avaliação da abordagem e solução					
Realizar <i>Snowballing</i> voltada a dívida de requisitos					
Solução computacional que contempla os passos da abordagem proposta					
Modelar e descrever a solução					
Apresentar a solução					
Avaliar o estudo					
Elaboração de artigos					
Defesa da Tese de Doutorado					
Escrita, ajustes e Defesa da Tese					

Fonte: Elaborado pelo autor (2023)

REFERÊNCIAS

- ABAD, Zahra Shakeri Hossein; RUHE, Guenther. **Using Real Options to Manage Technical Debt in Requirements Engineering**. IEEE International Requirements Engineering Conference, 2015. DOI: 10.1109/RE.2015.7320428. Disponível em: <https://ieeexplore.ieee.org/document/7320428>. Acesso em: Jul. 2021.
- ALBUQUERQUE, Danyllo; GUIMARAES, Everton; TONIN, Graziela; PERKUSICH, Mirko; ALMEIDA, Hyggo; PERKUSICH, Angelo. **Comprehending the Use of Intelligent Techniques to Support Technical Debt Management**. IEEE/ACM International Conference on Technical Debt (TechDebt). 2022. DOI: 10.1145/3524843.3528097. Disponível em: <https://ieeexplore.ieee.org/document/9804504>. Acesso em: Jan. 2023.
- ALFAYEZ, Reem; ALWEHAIBI, Wesam; WINN, Robert; VENSON, Elaine; BOEHM, Barry. **A systematic literature review of technical debt prioritization**: In: Proceedings of the 3rd International Conference on Technical Debt. p. 1–10, 2020. Disponível em: <https://dl.acm.org/doi/abs/10.1145/3387906.3388630>. Acesso em: Mai. 2021.
- ALI, Naveed; LAI, Richard. **Method of software requirements specification and validation for global software development**. Requirements Engineering. vol. 22, pp. 191–214. 2017. Disponível em: <https://link.springer.com/article/10.1007/s00766-015-0240-4> . Acesso em: Dez. de 2022.
- ALVES, Nicolli S.R.; MENDES, Thiago S.; MENDONÇA, Manoel G. de; SPÍNOLA, Rodrigo O.; SHULL, Forrest; SEAMAN, Carolyn. **Identification and management of technical debt: A systematic mapping study**. Information and Software Technology. Vol. 70. p. 100-121. 2016. DOI:<https://doi.org/10.1016/j.infsof.2015.10.008>. Disponível em: <https://www.sciencedirect.com/science/article/abs/pii/S0950584915001743>. Acesso em: Jan. 2023.
- ALVES, Rosangela; KINCHESCKI, Geovana Fritzen; SILVA, Viviane Regina; VECCHIO, Humberto Pereira; OLIVEIRA, Cláudio Ladeira; CANCELIER, Mikhail Vieira de Lorenzi. **Aplicabilidade da Matriz GUT para identificação dos processos críticos: O estudo de caso do departamento de direito da Universidade Federal de Santo Catarina**. In: XVII Colóquio Internacional de Gestão Universitária. 2017, Mar del Plata, Argentina. Disponível em: https://repositorio.ufsc.br/xmlui/bitstream/handle/123456789/181033/101_00160.pdf?sequence=1&isAllowed=y . Acesso em: Jul. 2023.
- AMPATZOGLOU, Areti; AMPATZOGLOU, Apostolos; CHATZIGEORGIOU, Alexander; AVGERIOU, Paris. **The financial aspect of managing technical debt: A systematic literature review**. Information and Software Technology. Vol 64. p. 52-73. 2015. DOI: <https://doi.org/10.1016/j.infsof.2015.04.001>. Disponível em: <https://www.sciencedirect.com/science/article/abs/pii/S0950584915000762>. Acesso em: Jan. 2023.
- AMPATZOGLOU, Apostolos; BIBI, Stamatia; AVGERIOU, Paris; VERBEEK, Marijin; CHATZIGEORGIOU, Alexander. **Identifying, categorizing and mitigating threats to validity in Software Engineering**. secondary studies. Information and Software Technology. Vol. 106. p. 201-230. 2019. DOI: <https://doi.org/10.1016/j.infsof.2018.10.006>.

AVGERIOU, Paris; KRUCHTEN, Philippe; OZKAYA, Ipek, and SEAMAN, Carolyn. **Managing Technical Debt in Software Engineering**. In: Dagstuhl Seminar 16162. DOI: 10.4230 / DagRep.6.4.110, 2016. Disponível em <https://www.dagstuhl.de/16162>. Acesso em: Mar. 2020.

BEHUTIYE, Woubshet; RODRÍGUEZ, Pilar; OIVO, Markku; AARAMAA, Sanja; PARTANEN, Jari; ABHERVÉ, Antonin. **How agile software development practitioners perceive the need for documenting quality requirements: a multiple case study**. 2020. Disponível em: <https://ieeexplore.ieee.org/document/9226305/>. Acesso em: Ago. 2021.

BEHUTIYE, Woubshet Nema; RODRÍGUEZ, Pilar; OIVO, Markku; TOSUN, Ayşe. **Analyzing the concept of technical debt in the context of agile software development: A systematic literature review**. In: Information and Software Technology. vol. 82. p. 139-158, 2017. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0950584916302890>. Acesso em: Jul. 2021.

BESKER, Terese; MARTINI, Antonio; BOSCH, Jan. **Technical Debt Triage in Backlog Management**. In: IEEE/ACM International Conference on Technical Debt (TechDebt), 2019. DOI: DOI 10.1109/TechDebt.2019.00010. Acesso em: Ago. 2021.

BJARNASON Elizabeth; WNUK, Krzysztof; REGNELL Björn. **A case study on benefits and side-effects of agile practices in large-scale requirements engineering**. AREW '11: Proceedings of the 1st Workshop on Agile Requirements Engineering, pp. 1-5, July 2011. DOI: <https://doi.org/10.1145/2068783.2068786>. Disponível em: <https://dl.acm.org/doi/10.1145/2068783.2068786>. Acesso em: Dez. 2021.

BRANCO FILHO, Gil. **A organização, o planejamento e o controle da manutenção**. Rio de Janeiro, Editora Ciência Moderna Ltda., 2008.

CHARMAZ, Kathy. **Constructing Grounded Theory: A practical guide through qualitative analysis**, 2006. London. ISBN 2005928035.

CHARMAZ, Kathy. **Constructing Grounded Theory**, 2nd ed., 2014.

DARE Centre for Reviews and Dissemination, What are the Criteria for the Inclusion of Reviews on DARE? 2007, <<http://www.york.ac.uk/inst/crd/faq4.htm>

CENTRE FOR REVIEWS AND DISSEMINATION. **Database of Abstracts of Reviews of Effects (DARE): Quality-assessed Reviews**. Bookshelf ID: NBK285222. University of York. 2014. Disponível em: <https://www.ncbi.nlm.nih.gov/books/NBK285222/> Acesso em: Jan 2023.

CODABUX, Zadia; WILLIAMS, Byron J. **Technical Debt Prioritization Using Predictive Analytics**. In: IEEE/ACM International Conference on Software Engineering Companion (ICSE-C), 38th ed., 2016. Disponível em: <https://ieeexplore.ieee.org/abstract/document/7883378>. Acesso em: Ago. 2021.

CURCIO, Karina; NAVARRO, Tiago; MALUCELLI, Andreia; REINEHR, Sheila. **Requirements engineering: A systematic mapping study in agile software development**. In: Journal of Systems and Software, vol. 139, pag. 32-50, 2018. DOI:

<https://doi.org/10.1016/j.jss.2018.01.036>. Disponível em:
<https://www.sciencedirect.com/science/article/pii/S0164121218300141#bib0003>. Acesso em:
 Jan. 2022.

CUNNINGHAM, Ward. **The WyCash Portfolio Management System**. Proc. OOPSLA, ACM, 1992. Disponível em: <https://dl.acm.org/doi/pdf/10.1145/157710.157715>. Acesso em:
 Set. 2020.

DALPIAZ, Fabiano; BRINKKEMPER, Sjaak. **Agile requirements engineering with user stories**. In: IEEE 26th International Requirements Engineering Conference (RE). 2018. DOI: 10.1109/RE.2018.00075. Disponível em: <https://ieeexplore.ieee.org/document/8491182>. Acesso em: Fev 2023.

DARWISH, Nagy Ramadan; MEGAHED, Salwa. **Requirements Engineering in Scrum Framework**. In: International Journal of Computer Applications (0975 – 8887), vol. 149, Nº 8, 2016. Disponível em:
https://www.researchgate.net/publication/308186449_Requirements_Engineering_in_Scrum_Framework. Acesso em: Jan. 2022.

DRESCH, Aline; LACERDA, Daniel Pacheco; ANTUNES, José Antonio Valle Júnior. **Design science research: método de pesquisa para avanço da ciência e tecnologia**. Ed. Bookman, E-pub. ISBN 978-85-8260-553-0. 2015.

DINGSØYR, Torgeir; NERUR, Sridhar; BALIJEPALLY, VenuGopal; MOE, Nils Brede. **A decade of agile methodologies: Towards explaining agile software development**. Journal of Systems and Software. Vol. 85, pp 1213 – 1221, 2012. DOI:
<https://doi.org/10.1016/j.jss.2012.02.033>. Disponível em:
<https://www.sciencedirect.com/science/article/pii/S0164121212000532>. Acesso em: Nov. 2021.

DYBÅ, Tore; DINGSØYR, Torgeir. **Empirical studies of agile software development: A systematic review**. In: Information and Software Technology. Vol. 50. Pp. 833 – 859. 2008 DOI: <https://doi.org/10.1016/j.infsof.2008.01.006>. Disponível em:
<https://dl.acm.org/doi/10.1016/j.infsof.2008.01.006>. Acesso em Dez. 2021.

EASTERBROOK, Steve; SINGER, Janice; STOREY, Margaret-Anne; DAMIAN Daniela. **Selecting Empirical Methods for Software Engineering Research**. Guide to Advanced Empirical Software Engineering - Chapter 11. ISBN-13: 78-1-84800-043-8. Springer, 2008.

ELGHARIANI, Kaiss; KAMA, Nazri. **Review on Agile requirements engineering challenges**. In: 3rd International Conference on Computer and Information Sciences (ICCOINS). DOI: 10.1109/ICCOINS.2016.7783267, 2016. Disponível em:
<https://ieeexplore.ieee.org/abstract/document/7783267>. Acesso em Jan. 2022.

ERNST, Neil A. **On the Role of Requirements in Understanding and Managing Technical Debt**. In: Third International Workshop on Managing Technical Debt (MTD). DOI: 10.1109/MTD.2012.6226002, 2012. Disponível em:
<https://ieeexplore.ieee.org/document/6226002>. Acesso em: Jul. 2020.

FERNÁNDEZ D. Méndez; WAGNER S.; FELDERER, KALINOWSKI, M.; MAFRA, M.; VETRÒ, P., A.; CONTE, T., M.; CHRISTIANSSON, T.; GREER, D.; LASSENIUS, C.;

MÄNNISTÖ, T.; NAYABI, M.; OIVO, M., PENZENSTADLER, B.; PFAHL, D.; PRIKLADNICKI, R.; RUHE, G.; SCHEKELMANN, A.; SEM, S.; SPINOLA, R.; TUZCU, DE LA VARA, A.; J. L.; WIERINGA, R.. **Naming the pain in requirements engineering**. In: Empirical Software Engineering, vol. 22, 2017. Disponível em: <https://link.springer.com/article/10.1007/s10664-016-9451-7> Acesso em: Ago. 2021.

FOWLER, Martin. **Dívida Técnica**. 2003. Disponível em: <http://www.martinfowler.com/bliki/TechnicalDebt.html>. Acesso em: Jun. 2020.

FREIRE, Sávio; RIOS, Nicolly; MENDONÇA, Manoel; FALESSI, Davide; SEAMAN, Carolyn; IZURIETA, Clemente; SPÍNOLA, Rodrigo O. **Actions and impediments for technical debt prevention: results from a global family of industrial surveys**. SAC '20: The 35th ACM/SIGAPP Symposium on Applied Computing, p. 1548–1555, 2020. DOI: <https://doi.org/10.1145/3341105.3373912>. Disponível em: <https://dl.acm.org/doi/10.1145/3341105.3373912>. Acesso em: Ago. 2021.

FREIRE, Savio; RIOS, Nicoli; PÉREZ, Boris; CASTELLANOS, Camilo; CORREAL, Dario; RAMAČ, Robert; TAUŠAN, Nebojsa; MANDIĆ, Vladimir; HERNANDEZ, Alexia Pacheco; LOPEZ, Gustavo; MENDONÇA, Manoel; IZURIETA, Clemente; FALESSI, Davide; SEAMAN, Carolyn B.; SPINOLA, Rodrigo Oliveira. **Pitfalls and Solutions for Technical Debt Management in Agile Software Projects**. 2021a. Disponível em: <https://ieeexplore.ieee.org/document/9507075/>. Acesso em: Ago. 2021.

FREIRE, Savio; RIOS, Nicoli; PÉREZ, Boris; CASTELLANOS, Camilo; CORREAL, Dario; RAMAČ, Robert; MANDIĆ, Vladimir; TAUŠAN, Nebojsa; LOPEZ, Gustavo; PACHECO, Alexia; FALESSI, Davide; MENDONÇA, Manoel; IZURIETA, Clemente; SEAMAN, Carolyn; SPINOLA, Rodrigo. **How Experience Impacts Practitioners' Perception of Causes and Effects of Technical Debt**. 2021b. Disponível em: <https://ieeexplore.ieee.org/document/9463240/>. Acesso em: Ago. 2021.

GHANBARI, Hadi. **Seeking Technical Debt in Critical Software Development Projects: An Exploratory Field Study**. 2016. Disponível em: <https://ieeexplore.ieee.org/document/7427856/>. Acesso em: Ago. 2021.

GRIVA, Anastasia; BYRNE, Stephen; DENNEHY, Denis; CONBOY, Kieran. **Software Requirements Quality: Using Analytics to Challenge Assumptions at Intel**. In: IEEE Software. Vol: 39. Pp. 80 – 88. 2022. DOI: 10.1109/MS.2020.3043868 Disponível em: <https://ieeexplore.ieee.org/document/9289010>. Acesso em: Ago. 2022.

GUO, Yuepu; SPÍNOLA, Rodrigo Oliveira and SEAMAN, Carolyn. **Exploring the costs of technical debt management – a case study**. In: Empirical Software Engineering, v. 21. ed. 1. p. 159-182, 2016. DOI: 10.1007/s10664-014-9351-7. INSS: 1573-7616. Disponível em: <https://doi.org/10.1007/s10664-014-9351-7>. Acesso em: Jun. 2021.

HEATON, Dustin; CARVER, C. Jeffrey. Claims about the use of software engineering practices in science: A systematic literature review. In: Information and Software Technology, vol. 67, p. 207-219, 2015. DOI: <https://doi.org/10.1016/j.infsof.2015.07.011>. Disponível em: <https://www.sciencedirect.com/science/article/abs/pii/S0950584915001342>. Acesso em: Ago. 2021.

HEEAGER, Lise T.; NIELSEN, Peter Axel. **A conceptual model of agile software development in a safety-critical context: A systematic literature review**. In: Information and Software Technology, vol. 103, p. 22-39, 2018. DOI: <https://doi.org/10.1016/j.infsof.2018.06.004>. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0950584918301125?via%3Dihub>. Acesso em: Jul. 2021.

HERDIKA, Hana Rizky; BUDIARDIO, Eko K. **Variability and Commonality Requirement Specification on Agile Software Development: Scrum, XP, Lean, and Kanban**. In: 3rd International Conference on Computer and Informatics Engineering (IC2IE). 2020. DOI: 10.1109/IC2IE50715.2020.9274564. Disponível em: <https://ieeexplore.ieee.org/abstract/document/9274564>. Acesso em: Ago. 2023.

HEVNER, Alan; CHATTERJEE, Samir. **Design Research in Information Systems - Theory and Practice**. Introduction to Design Science Research, Ebook, pag. 1-8. 2010. Disponível em: https://link.springer.com/chapter/10.1007/978-1-4419-5653-8_1. Acesso em: Jan. 2022.

HODA, Rashina; NOBLE, James. **Becoming agile: A grounded theory of agile transitions in practice**. In: 2017 IEEE/ACM 39th International Conference on Software Engineering (ICSE), 2017. DOI: 10.1109/ICSE.2017.21. Disponível em: <https://ieeexplore.ieee.org/document/7985657>. Acesso em: 15 ago. 2021.

HODA, Rashina; NOBLE, James; MARSHALL, Stuart. **Using Grounded Theory to study the human aspects of Software Engineering**. 2010. DOI: 10.1145/1938595.1938605. Disponível em: <https://www.researchgate.net/publication/228410715>. Acesso em: Out. 2019.

HODA, Rashina; NOBLE, James; MARSHALL, Stuart. **Developing a grounded theory to explain the practices of self-organizing Agile teams**. In: Empirical Software Engineering, vol. 17, 2012. Disponível em: <https://link.springer.com/article/10.1007/s10664-011-9161-0>. Acesso em: Ago. 2019.

HOLVITIE, Johannes; LEPPANEN, Ville; HYRYNSALMI, Sami. **Technical Debt and the Effect of Agile Software Development Practices on It – An Industry Practitioner Survey**. In: 6th IEEE International Workshop on Managing Technical Debt, 2014. DOI: 10.1109/MTD.2014.8. Acesso em: Ago. 2021.

INAYAT, Irum; SALIM, Siti Salwah; MARCZAK, Sabrina; DANEVA, Maya; SHAMSHIRBAND, Shahaboddin. **A systematic literature review on agile requirements engineering practices and challenges**. Computers in Human Behavior. Vol. 51, parte B, pg. 915-929. 2015. DOI: <https://doi.org/10.1016/j.chb.2014.10.046>. Disponível em: <https://www.sciencedirect.com/science/article/pii/S074756321400569X>. Acesso em Ago. 2023.

INTERNATIONAL STANDARD. ISO/IEC/ IEEE 29148 Reference number **ISO/IEC/IEEE 29148:2018** - Systems and software engineering — Life cycle processes. ISO/IEC/IEEE, 29148, 2018.

JACOBSON, Ivar. **Practice Library**. 2018. Disponível em <https://practicelibrary.ivarjacobson.com/>. Acesso: 26 jun. de 2021.

JACOBSON, Ivar; STIMSON, Roly. **The Essence of Software Engineering: Escaping Method Prison – On the Road to Real Software Engineering**. p. 37-58, 2017. Disponível em: https://link.springer.com/chapter/10.1007/978-3-319-73897-0_3. Acesso em: Abr. 2021.

JERONIMO, Helvio Junior; TRAVASSOS, Guilherme Horta. **Consolidating a common perspective on Technical Debt and its Management through a Tertiary Study**. Information and Software Technology. vOL 142. 2022. doi: <https://doi.org/10.1016/j.infsof.2022.106964>. Disponível em: <https://www.sciencedirect.com/science/article/abs/pii/S0950584922001057>. Acesso em: Jan. 2023.

JIA, Jingdong; YANG, Xiaoving, Yang; ZHANG, Rong; LIU, Xi. **Understanding software developers' cognition in agile requirements engineering**. Science of Computer Programming. Vol. 178, Pg. 1-19. 2019. DOI: <https://doi.org/10.1016/j.scico.2019.03.005>. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0167642318302764#br0030>. Acesso em: Set. 2023.

JOVANOVIĆ, Miloš; MAS, Antonia; MESQUIDA, Antoni-Lluís; LALÍC, Bojan. **Transition of organizational roles in Agile transformation process: A grounded theory approach**. In: Journal of Systems and Software, vol. 133, p. 174-194, 2017. DOI: <https://doi.org/10.1016/j.jss.2017.07.008>. Acesso em: Mar. 2021.

KALINOWSKI, Marcos; SPÍNOLA, Rodrigo Oliveira; CONTE, Tayana; PRIKLADNICKI, Rafael; MÉNDEZ, Fernández, Daniel; WAGNER, Stefan. **Towards building knowledge on causes of critical requirements engineering problems**. In: 27th International Conference on Software Engineering and Knowledge Engineering (SEKE). DOI: <http://dx.doi.org/10.18419/opus-3539>. 2015. Disponível em: <https://elib.uni-stuttgart.de/handle/11682/3556>. Acesso em: Jan. 2022.

KÄPYAHO, Marja; KAUPPINEN, Marjo. **Agile requirements engineering with prototyping: A case study**. Ieee 23rd International Requirements Engineering Conference (re). 2015. DOI: 10.1109/re.2015.7320450. Disponível em: <https://ieeexplore.ieee.org/document/7320450>. Acesso em: Dez. 2022.

KARHAPÄÄ, Pertti; BEHUTIYE, Woubshet; RODRÍGUEZ, Pilar; OIVO, Markku; COSTAL Dolors; FRANCH, Xavier; AARAMAA, Sanja; CHORAŚ, Michał; PARTANEN, Jari; ABHERVE, Antonin. **Strategies to manage quality requirements in agile software development: a multiple case study**. In: Empirical Software Engineering, vol. 26, 2021. Disponível em: <https://link.springer.com/article/10.1007/s10664-020-09903-x>. Acesso em: Ago. 2021.

KITCHENHAM, Bárbara; CHARTERS, Stuart. **Guidelines for performing systematic literature reviews in software engineering**, 2007.

KITCHENHAM, Barbara; BRERETON, O. Pearl; BUDGEN, David; TURNER, Mark; BAILEY, John; LINKMAN, Stephen. **Systematic literature reviews in software engineering – A systematic literature review**. Information and Software Technology. Vol. 51. p. 7-15. 2009. DOI: <https://doi.org/10.1016/j.infsof.2008.09.009>. Disponível em: <https://www.sciencedirect.com/science/article/abs/pii/S0950584908001390>. Acesso em: Jan 2023.

KRUCHTEN, Philippe; NORD, Robert L.; OZKAYA, Ipek; VISSER, Joost. **Technical Debt in Software Development: from Metaphor to Theory** - Report on the Third International Workshop on Managing Technical Debt. In: ACM SIGSOFT Software Engineering, v. 37, n. 5, 2012a. Disponível em: <http://doi.acm.org/10.1145/2347696.2347698>. Acesso em: Mar. 2020.

KRUCHTEN, Philippe; NORD, Robert L.; OZKAYA, Ipek. **Technical Debt: From Metaphor to Theory and Practice**. In: IEEE SOFTWARE, v. 29, n. 6, 2012b. DOI: 10.1109/MS.2012.167. Disponível em: <https://ieeexplore.ieee.org/document/6336722>. Acesso em: Mar. 2020.

KUHLEN, D.; SPECK, A. The potentials of a code generator which faces the stress ratio of requirements engineering processes in agile development projects. Series of the Gesellschaft für Informatik (GI). vol. P 255, p. 87, 2016. Acesso em: Ago. 2021.

LACERDA, Daniel Pacheco; DRESCH, Aline; PROENÇA, Adriano; ANTUNES, José Antonio Valle Júnior. **Design Science Research: a research method to production engineering**. DOI: <https://doi.org/10.1590/S0104-530X2013005000014>. 2013. Disponível em: <https://www.scielo.br/j/gp/a/3CZmL4JJxLmxCv6b3pnQ8pq/?lang=pt#>. Acesso em: Jan. 2022.

LAPLANTE, Phillip A. **Requirements Engineering for Software and Systems**. 3th ed. ISBN 9781138196117, 2018.

LENARDUZZI, Valentina; ORAVA, Teemu; SAARIMÄKI, Nyyti; SYSTA, Kari; TAIBI, Davide. **An Empirical Study on Technical Debt in a Finnish SME**. 2019. Disponível em: <https://ieeexplore.ieee.org/document/8870169/>. Acesso em: 26 ago. 2021.

LENARDUZZI, V.; BESKER, T.; TAIBI, D.; MARTINI, A; FONTANA, F. A. **A systematic literature review on Technical Debt prioritization: Strategies**, processes, factors, and tools. In: The Journal of Systems & Software vol.171, 2021. Acesso em: Jul 2021.

LENARDUZZI, Valentina; FUCCI, Davide. **Towards a Holistic Definition of Requirements Debt**. ACM/IEEE International Symposium on Empirical Software Engineering and Measurement (ESEM). vol1, p. 1-5, 2019. DOI: 10.1109/ESEM.2019.8870159. Disponível em: <https://www.computer.org/csdl/proceedings-article/esem/2019/08870159/1ecCOxHWORy>. Acesso em: Jul de 2021.

LI, Zengyang; AVGERIOU, Paris; LIANG Peng. **A systematic mapping study on technical debt and its management**. In: Elsevier: Journal of Systems and Software, vol. 101, p. 193-220, 2015. DOI: <https://doi.org/10.1016/j.jss.2014.12.027>. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0164121214002854?via%3Dihub>. Acesso em: Mar. 2020.

LIM, Erin; TAKSANDE, Nitin; SEAMAN, Carolyn. **A Balancing Act: What Software Practitioners Have to Say about Technical Debt**. In: IEEE Software. vol. 29, 2012. DOI: 10.1109/MS.2012.130. Disponível em: <https://ieeexplore.ieee.org/document/6280547>. Acesso em: 30 de mai. 2021.

LUCASSEN, Garm; DALPIAZ, Fabiano; WERF, Jan Martijn E.M. Van Der; BRINKKEMPER, Sjaak; Zowghi, Didar. **Behavior-Driven Requirements Traceability Via Automated Acceptance Tests**. IEEE 25th International Requirements Engineering Conference Workshops (Rew). 2017. DOI: 10.1109/REW.2017.84. Disponível em: <https://ieeexplore.ieee.org/document/8054891>. Acesso em: Dez. 2022.

MANDIĆ, Vladimir; TAUŠAN, Nebojsa; RAMAČ, Robert; FREIRE, Savio; RIOS, Nicolli; PEREZ, Boris; CASTELLANOS, Camilo; CORREAL, Dario; HERNANDEZ, Alexia Pacheco; LOPEZ, Gustavo; IZURIETA, Clemente; FALESSI, Davide; SEAMAN, Carolyn B.; SPINOLA, Rodrigo Oliveira. **Technical and Non-Technical Prioritization Schema for Technical Debt: Voice of TD-Experienced Practitioners**. 2021. Disponível em: <https://ieeexplore.ieee.org/document/9508982/>. Acesso em: Ago. 2021.

MANDIĆ, Vladimir; TAUŠAN, Nebojšav; Serbia Robert, RAMAČ. **The Prevalence of the Technical Debt Concept in Serbian IT Industry: Results of a National-Wide Survey**. In: 2020 IEEE/ACM International Conference on Technical Debt (TechDebt), 2020. Disponível em: <https://dl.acm.org/doi/pdf/10.1145/3387906.3388622>. Acesso em: Ago. 2021.

MARRA, Manuela; BICCARI, Carla Di; LAZOI, Mariangela; CORALLO, Angelo. **Gap Analysis Methodology for Product Lifecycle Management Assessment**. In: IEEE Transactions on Engineering Management. vol: 65, 2018. DOI: 10.1109/TEM.2017.2762401. Disponível em: <https://ieeexplore.ieee.org/document/8097037>. Acesso em Ago 2023.

MCCONNELL, Steve. **Managing Technical Debt**. Construx Software, v. 1, 2008. Disponível em: <https://www.construx.com/resources/whitepaper-managing-technical-debt/>. Acesso em: Jul. 2021.

MELO, Ana; FAGUNDES, Roberta; LENARDUZZI, Valentina; SANTOS, Wylliams. **Identification and Measurement of Technical Debt Requirements in Software development: a Systematic Literature Review**. Journal of Systems and Software. Vol 194, 2022. DOI: <https://doi.org/10.1016/j.jss.2022.111483>. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0164121222001650>. Acesso em: Jan 2023.

MELEGATI, Jorge; GOLDMAN, Alfredo; KON, Fabio; WANG, Xiaofeng. **A model of requirements engineering in software startups**. In: Information and Software Technology, vol. 109, p. 92-107, 2019. DOI: <https://doi.org/10.1016/j.infsof.2019.02.001>, 2019. Disponível em: <https://www.sciencedirect.com/science/article/abs/pii/S095058491930028X>. Acesso em: Mar. 2021.

MENDES, T.S.; DE FARIAS, M.A.F.; MENDONÇA, M.; (...), KALINOWSKI, M.; SPÍNOLA, R.O. **Impacts of agile requirements documentation debt on software projects: A retrospective study**. In: ACM Symposium on Applied Computing, vol. 04, p. 1290 – 12954, 2016. Acesso em: Ago. 2021.

MUMTAZ, Haris; SINGH, Paramvir; BLINCOE, Kelly. **A systematic mapping study on architectural smells detection**. Journal of Systems and Software. Vol. 173. 2021. DOI: <https://doi.org/10.1016/j.jss.2020.110885>. Disponível em: <https://www.sciencedirect.com/science/article/abs/pii/S0164121220302752> aCESSO EM: Jan. 2023.

NASCIMENTO, Rafael; Aranha, Eduardo; KULESZA, Uirá; LUCENA, Marcia.

Requirements Smells como indicadores de má qualidade na especificação de requisitos:

Um Mapeamento Sistemático da Literatura. DOI:10.17771/PUCRio.wer.inf2018-40, 2018.

Disponível em:

http://wer.inf.puc-rio.br/WERpapers/artigos/artigos_WER18/WER_2018_paper_40.pdf.

Acesso em: Jul. 2021.

NIELSEN, Mille, Edith; MADSEN, Christian Østergaard; Lungu Mircea Filip. **Technical Debt management: A Systematic Literature Review and Research Agenda for Digital Government**, International Conference on Electronic Government: EGOV: Electronic Government, p.

121-137, 2020. Disponível em:

https://link.springer.com/chapter/10.1007/978-3-030-57599-1_10. Acesso em: Jul. 2021.

OMG, OBJECT MANAGEMENT GROUP. **Essence – Kernel and Language for Software Engineering Methods**. Versão 1.2. 2018. Disponível em:

<https://www.omg.org/spec/Essence/20150601/Essence.xmi>. Acesso em: Nov. 2022.

OCHODEK, Mirosław; KOPCZYŃSKA, Sylwia. **Perceived importance of agile requirements engineering practices – A survey**. In: Journal of Systems and Software, vol. 143, pag. 29-43, 2018. DOI: Disponível em:

<https://www.sciencedirect.com/science/article/pii/S0164121218300955#!>. Acesso em: Jan. 2022.

OLIVEIRA, Gabriel; MARCZAK, Sabrina. **On the Empirical Evaluation of BDD Scenarios Quality: Preliminary Findings of an Empirical Study**. IEEE 25th International

Requirements Engineering Conference Workshops (REW). 2017. DOI: 10.1109/REW.2017.62.

Disponível em: <https://ieeexplore.ieee.org/document/8054868>. Acesso em: Dez. de 2022.

ORDÓÑEZ, Hugo; VILLADA, Andrés Felipe Escobar; VANEGAS, Diana Lorena Velandia; COBOS, Carlos; ORDÓÑEZ, Armando; SEGOVIA, Rocio. **An impact study of business process models for requirements elicitation in XP**.

International Conference on Computational Science and Its Applications, 9155, Springer Verlag (2015), pp. 298-312. Acesso em: Dez 2022.

PEFFERS Ken; TUUNANEN, Tuure; ROTHENBERGER, Marcus A.; CHATTERJEE, Samir.

A Design Science Research Methodology for Information Systems Research. DOI:

<https://doi.org/10.2753/MIS0742-1222240302>. In: Journal of Management Information Systems. 2014. Acesso em: Jan. 2022.

PERIARD, Gustavo. **Matriz GUT: guia completo**. Disponível:

<http://www.sobreadministracao.com/matrizgut-guia-completo>. Acesso em 28 abr. 2019.

PESTANA, Marcelo Diniz; VERAS, Gabriela Pinheiro; FERREIRA, Maria Tereza Matos;

SILVA, Abraao Ramos. **Aplicação integrada da matriz GUT e da matriz da qualidade em uma empresa de consultoria ambiental - Um estudo de caso para elaboração de propostas de melhorias**. In: ENEGEP: XXXVI Encontro Nacional de Engenharia de Produção. 2016,

João Pessoa, PB. Disponível em: https://abepro.org.br/biblioteca/tn_stp_227_329_30428.pdf.

Acesso em: Jul. de 2023.

PRESSMAN, Roger R.; MAXIM, Bruce S. **Engenharia de Software**. Uma abordagem profissional. Tradução: João Eduardo Nóbrega Tortello; revisão técnica: Reginaldo Arakaki, Julio Arakaki, Renato Manzan de Andrade. 8ª edição. Porto Alegre : AMGH, 2016.

PMBOK: Project Management Institute. **Um Guia do Conhecimento em Gerenciamento de Projetos** (Guia PMBOK) 6ª edição, em Português (Brasil). ISBN 9781628251845, 2017.

RAMAČ, Robert; MANDIĆ, Vladimir; TAUŠAN, Nebojsa; RIOS, Nicolli; NETO, Manoel G. de Mendonça; SEAMAN, Carolyn; SPINOLA, Rodrigo Oliveira. **Common Causes and Effects of Technical Debt in Serbian IT: InsignTD Survey Replication**. 2020. Disponível em: <https://ieeexplore.ieee.org/document/9226276/>. Acesso em: Ago. 2021.

RASHEED, Aqsa; ZAFAR, Bushra; SHEHRYAR, Tehmina; ASLAM, Naila Aiman; SAJID, Muhammad; ALI Nouman; DAR, Saadat Hanif; KHALID, Samina. Requirement Engineering Challenges in Agile Software Development. In: Mathematical Problems in Engineering. 2021. DOI: <https://doi.org/10.1155/2021/6696695>. Disponível em: <https://www.hindawi.com/journals/mpe/2021/6696695/>. Acesso em Mar. 2023.

REINEHR, Sheila. **Engenharia de Requisitos**. ISBN 978-65-5690-067-4. Porto Alegre: SAGAH, 2020.

RINDELL, Kalle; BERNSMED, Karin; JAATUN, Martin Gilje. **Managing Security in Software: Or: How I Learned to Stop Worrying and Manage the Security Technical Debt**. In: ARES '19: 14th International Conference on Availability, Reliability and Security, p. 1-8, 2019. DOI: <https://doi.org/10.1145/3339252.3340338>. Disponível em: <https://dl.acm.org/doi/10.1145/3339252.3340338>. Acesso em: Ago. 2021.

RIOS, Nicolli; SPINOLA, Rodrigo Oliveira; MENDONÇA, Manoel G. Neto; SEAMAN, Carolyn. **A Study of Factors that Lead Development Teams to Incur Technical Debt in Software Projects**. 2018a. Disponível em: <https://ieeexplore.ieee.org/document/8498243/>. Acesso em: Ago. 2021.

RIOS, Nicolli; MENDONÇA, Manoel Gomes Neto; SPÍNOLA, Rodrigo Oliveira; **A tertiary study on technical debt: Types, management strategies, research trends, and base information for practitioners**. In: Information and Software Technology, vol. 102, p. 117–145, 2018. DOI: <https://doi.org/10.1016/j.infsof.2018.05.010>. Acesso em: Ago. 2021.

RIOS, Nicolli; MENDONÇA, Manoel; SPÍNOLA, Rodrigo Oliveira; SEAMAN, Carolyn. **Causes and effects of the presence of technical debt in agile software projects**. In: Twenty-fifth Americas Conference on Information Systems, AMCIS, 2019.

RIOS, Nicolli; SPÍNOLA, Rodrigo Oliveira; MENDONÇA, Manoel; SEAMAN, Carolyn. **The practitioners' point of view on the concept of technical debt and its causes and consequences: a design for a global family of industrial surveys and its first results from Brazil**. In: Empirical Software Engineering, vol. 25, p. 3216–3287, 2020. DOI: <https://doi.org/10.1007/s10664-020-09832-9>. Disponível em: <https://link.springer.com/content/pdf/10.1007/s10664-020-09832-9.pdf> Acesso em: Ago. 2021.

RIOS, N.; MENDES, L.; CERDEIRAL, C., (...); SANTOS, G.; SPÍNOLA, R. O. **Hearing the Voice of Software Practitioners on Causes, Effects, and Practices to Deal with**

Documentation Debt. In: 26th International Working Conference on Requirements Engineering: Foundation for Software Quality, 2020. Acesso em: Ago. 2021.

RIOS, Nicolli; MENDONÇA, Manoel Gomes de Neto; SPÍNOLA, Rodrigo Oliveira. **A tertiary study on technical debt: Types, management strategies, research trends, and base information for practitioners.** Information and Software Technology Vol. 102. 2018. DOI: <https://doi.org/10.1016/j.infsof.2018.05.010>. Disponível em: <https://www.sciencedirect.com/science/article/abs/pii/S0950584918300946>. Acesso em: jan. 2023.

RIOS, Nicolli; RIBEIRO, Leilane F.; CAIRES, Vivyane; MENDES, Thiago S.; SPÍNOLA, Rodrigo O. **Towards an Ontology of Terms on Technical Debt.** In: Sixth International Workshop on Managing Technical Debt. 2014. DOI: 10.1109/MTD.2014.9. Disponível em: <https://ieeexplore.ieee.org/document/6974882>. Acesso em: Jan. 2023.

ROBIOLO, G.; SCOTT, E.; MATALONGA, S.; FELDERER, M. **Technical Debt and Waste in Non-functional Requirements Documentation: An Exploratory Study.** In: 20th International Conference on Product-Focused Software Process Improvement, 2019. Acesso em: Ago. 2021.

ROSSER, Larri Ann; NORTON, John H. **A Systems Perspective on Technical Debt.** 2021. Disponível em: <https://ieeexplore.ieee.org/document/9438359/>. Acesso em: Ago. 2021.
SALEH, Mohammed; BAHAROM, Fauziah; MOHAMED, Shafinah Farvin Packeer. **Critical Success Factors and Challenges in Agile Requirements Engineering.** Turkish Journal of Computer and Mathematics Education. Vol.12. N.3. 2021. 1670-1682. Disponível em: <https://www.turcomat.org/index.php/turkbilmat/article/view/989/776>. Acesso em: Ago 2023.

SIGG, Maria de León; REYES, Sodel Vázquez; ÁVILA, Daniel Rodríguez. **Towards the Use of a Framework to Make Technical Debt Visible.** 2020. Disponível em: <https://ieeexplore.ieee.org/document/9307774/>. Acesso em: Ago. 2021.

SILVA, Edna Lúcia; MENEZES, Estera Muszkat. **Metodologia da pesquisa e elaboração de dissertação.** 4. ed. rev. atual. – Florianópolis: UFSC, 2005.

SIMON, Herbert A. **The Sciences of the Artificial.** Third edition. ISBN eBook 9780585360102. 1996.

SOARES, Henrique F.; ALVES, Nicolli S. R.; MENDES, Thiago S.; MENDONÇA, Manoel; SPÍNOLA, Rodrigo O. **Investigating the link between user stories and documentation debt on software projects.** In: International Conference on Information Technology - New Generations, 2015. DOI: 10.1109/ITNG.2015.68. Disponível em: <https://ieeexplore.ieee.org/abstract/document/7113503>. Acesso em: Ago. 2021.

SOUZA, José Orlando de Lima. **Matriz de GUT – Descubra como tomar melhores decisões.** Ebook - Amazon, 1ª ed. ASIN: B09TQ6TTJR. 2022.

SWEBOK. **Guide to the Software Engineering.** Version 3.0. IEEE Computer Society. Editors: Bourque, Pierre; Fairley, Richard E, 2014.

SCHÖN, Eva-Maria; THOMASCHEWSKI Jörg; ESCALONA, María José. **Agile Requirements Engineering: A systematic literature review.** In: Computer Standards &

Interfaces, vol. 49, p. 79-91, 2017. DOI: <https://doi.org/10.1016/j.csi.2016.08.011>. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0920548916300708?via%3Dihub>. Acesso em: Jul. 2021.

SCHÖN, Eva-Maria; WINTER, Dominique; THOMASCHEWSKI Jörg; ESCALONA, María José. **Key challenges in agile requirements engineering**. In: International Conference on Agile Software Development: Agile Processes in Software Engineering and Extreme Programming, pp. 37-51, April 2017. Disponível em: https://link.springer.com/chapter/10.1007/978-3-319-57633-6_3. Acesso em Jan. 2022.

TONG, Haonan, LIU, Bin, WANG, Shihai. **Software defect prediction using stacked denoising autoencoders and two-stage ensemble learning**. In: Science Direct - Information and Software Technology, v. 96, p. 94-111, 2018. DOI: <https://doi.org/10.1016/j.infsof.2017.11.008>. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0950584917300113>. Acesso em: Set. 2019.

TRKMAN, Marina; MENDLING, Jan; TRKMAN, Peter; KRISPER, Marjan. **Impact of the conceptual model's representation format on identifying and understanding user stories**. Vol. 116, 2019. DOI: 10.1016/j.infsof.2019.08.001. Disponível em: <https://www.sciencedirect.com/science/article/pii/S095058491930165X>. Acesso em: dez. 2022.

VARALDA, Wagner; VEJA, Ítalo Santiago. **Hermeneutical Elicitation of Requirements: A Technical Perspective to Improve the Conception of the Software Requirements**. In: Journal of Computer and Communications. Vol. 6. Pg. 132-149. 2018. Disponível em: https://www.scirp.org/pdf/JCC_2018092715375517.pdf. Acesso em Set. 2023.

YLI-HUUMO, Jesse; MAGLYAS, Andrey; SMOLANDER, Kari. **How do software development teams manage technical debt? – An empirical study**. Journal of Systems and Software. Vol. 120. p 195 – 218. 2016. DOI: <https://doi.org/10.1016/j.jss.2016.05.018>. Disponível em: <https://www.sciencedirect.com/science/article/pii/S016412121630053X>. Acesso em: Jan 2023

WAINER, J. **Métodos de pesquisa quantitativa e qualitativa para a Ciência da Computação**. Instituto de Computação - Universidade de Campinas (UNICAMP), 2017. Disponível em: <https://www.researchgate.net/publication/237601058>. Acesso em: Out. 2019.

WANG, Qianneng; HUANG, Yujie. **Identification and Management of Requirements Debt Systematic Mapping Study and Survey**. DiVA - Digitala Vetenskapliga Arkivet, 2020. Disponível em: <https://www.diva-portal.org/smash/record.jsf?pid=diva2%3A1492963&dswid=-2317>. Acesso em: Ago. 2021.

WATTANAKRIENGKRAI, Supatsara; MAIPRADITY, Rungroj; HATAY, Hideki.; CHOETKIERTIKUL, Morakot; SUNETNANTA, Thanwadee; MATSUMOTOY, Kenichi. **Identifying Design and Requirement Self-Admitted Technical Debt using N-gram IDF**. In: 9th International Workshop on Empirical Software Engineering in Practice (IWESEP), 2018. DOI: 10.1109/IWESEP.2018.00010. Disponível em: <https://ieeexplore.ieee.org/document/8661216>. Acesso em: Set. 2020.

WIEGERS, Karl; BEATTY, Joy. **Software Requirements**. Third Edition ISBN: 978-0-7356-7966-5. Redmond – Washington. ISBN: 978-0-7356-7966-5, 2013.

WILLIAMS, Laurie. A (Partial) **Introduction to Software Engineering Practices and Methods**. 2009. Disponível em: <https://sdc.csc.ncsu.edu/files/resources/williams-software-engineering-2011.pdf>. Acesso em: Ago. 2021.

APÊNDICES

APÊNDICE A – ROTEIRO DA ENTREVISTA SEMIESTRUTURADA

Seção I - Engenharia de Requisitos	
Como ocorre o processo que envolve requisitos?	Questões gerais pertencentes ao Roteiro
Como as demandas chegam, quem as recebe, quem as especifica/documenta, ferramentas?	
Utilizam algum template, padrão para registrar esses requisitos? Usam ferramentas, quais?	
Como vocês monitoram os requisitos?	
Quais as dificuldades percebidas durante o processo que envolve requisitos?	
Por que essas dificuldades acontecem?	
Quem recebe as demandas? É o PO?	Exemplos de Questões que surgiram conforme a evolução da entrevista.
Em relação ao que foi especificado e o que foi desenvolvido, existe muita inconsistência de funcionalidades?	
O que é produzido no documento de requisitos? Esse documento é o detalhamento dos requisitos ou é o documento de visão?	
Tudo que está no backlog será desenvolvido?	
Seção II - Dívida Técnica (Requisitos)	
O que entendem por dívida técnica?	Questões gerais pertencentes ao Roteiro
Relate, por favor, situações que na sua concepção, caracteriza DT em requisitos.	
Como a DT é documentada?	
Como a DT é monitorada?	
Quais são as dificuldades enfrentadas em pagar a DT?	
Quais fatores acredita-se serem as causas das DT?	
Percebe as consequências ocasionadas pela DT em requisitos?	Exemplos de Questões que surgiram conforme a evolução da entrevista.
Qual a frequência da DT?	
Utilizam ferramentas para controlar e monitorar DT?	

Fonte: Elaborado pelo autor (2021)

APÊNDICE B - PERFIL COMPLETO DOS PARTICIPANTES DO ESTADO DA PRÁTICA

Empresas Nacionais		
Empresa/Participante	Localização/Função na empresa	Segmento/Experiência na função
Empresa A	Chapecó (SC)	Software para gestão de provedores de Internet
PA1	Gerente de Projetos	Experiente
PA2	Engenheiro de Requisitos	Experiente
Empresa B	Chapecó (SC)	Logística e Transporte
PB1	Gerente de Projetos	Novato
PB2	Gerente de Projetos	Novato
Empresa C	Chapecó (SC)	Soluções para logística
PC1	Scrum Master	Experiente
PC2	Analista de Negócios	Novato
Empresa D	Chapecó (SC)	Gestão de Hotelaria, Fast/Restaurantes e postos de combustíveis
PD1	Gestor de Processos	Experiente
PD2	Gerente de Projetos	Experiente
PD3	Gerente de Projetos	Experiente
Empresa E	São Paulo (SP)/Passo Fundo (RS)	Transformação digital
PE1	Engenheiro de Requisitos	Experiente
PE2	Business Analytics	Experiente
PE3	Engenheiro de Requisitos	Novato
Empresa F	Florianópolis (SC)	Soluções para Ministério da Saúde
PF1	Gerente de Projetos	Experiente
PF2	Engenheiro de Requisitos	Experiente
Empresa G	São Paulo (SP)	Tecnologia para setor automotivo
PG	Gerente de Produto	Experiente
Empresa H	Florianópolis (SC)	Não informado
PH	Gerente de Software	Experiente

Empresa I	São Paulo (SP) Maringá (PR)	Produto para Instituições Financeiras
PI	Agile Coach	Experiente
Empresas Internacionais		
Empresa J	Bruxelas (Bélgica)	Tecnologia para setor público
PJ	Software Engineer	Experiente
Empresa K	Nice (França)	Soluções para empresas aéreas
PK	Líder Técnico	Experiente
Empresa L	Abu Dhabi (Emirados Árabes)	Soluções para setor público
PL	Líder Técnico	Experiente
Empresa M	Beja (Portugal)	Soluções para segurança da informação
PM	CEO	Experiente

Fonte: Elaborado pelo autor (2021)

APÊNDICE C - EXEMPLOS DE SEGMENTOS CODIFICADOS E RESPECTIVOS CÓDIGOS

Código Inicial	Segmentos da entrevista dos participantes	Participante
Requisitos	“Esse tipo de situação da falta de requisitos gera falta de aderência num projeto que foi desenvolvido e aprovado e que vai retornar depois para corrigir, vai gerar retrabalho, vai gerar DT e vai gerar uma complexidade acima do que deveria ser”.	PC1
Dívida Técnica	“Quando acontecem os DT de negócios acredito que aconteçam por uma falta de entendimento, ou falta de alinhamento inicial, algo que não foi mapeado muito bem quando a solicitação chegou, e foi visto após uma entrega que deveria ter isso incluído”.	PE1
Práticas	“Nós fazemos referências com base em tags – por exemplo: workspace, documents, notification. Conseguimos então fazer uma ligação de tudo, como uma matriz”.	PJ
Problemas	“Esse processo de conversar com o cliente para entender os requisitos é uma coisa que não está mil maravilhas, porque é uma coisa difícil. Digamos assim: isso é uma coisa do projeto. Nós não temos uma pessoa responsável por uma funcionalidade (dificuldade de comunicação com cliente)”.	PF1
Documentação	“Esse documento de visão, se a gente olhar ali para o conteúdo engenharia de software, é um requisito que vai ter os requisitos funcionais e não funcionais. O que a gente chama de documento de visão, que para alguns eu acho que seria aquele primeiro documento lá, inicial. A gente inverte. O nosso documento de visão já é mais detalhado”.	PI

Fonte: Elaborado pelo autor (2021)

APÊNDICE D - CATEGORIAS E SUBCATEGORIAS DA CODIFICAÇÃO FOCALIZADA

Categorias do fenômeno: Fatores que impactam na Engenharia de Requisitos	
Categoria	Subcategorias
Causas que geram DT em Requisitos	<ul style="list-style-type: none"> ☹️ Falha no recebimento dos requisitos_DT_Elicitação de Requisitos ☹️ Falha no refinamento dos requisitos_DT_Análise de Requisitos ☹️ Inconsistência na Especificação de Requisitos ☹️ Falta validarem antes de desenvolverem_DT_Validação ☹️ Ausência de monitoramento dos Req DT_Gerência de Requisitos
Consequências que caracterizam DT em Requisitos	<ul style="list-style-type: none"> ☹️ Inconsistência nos RF/RN_DT_Análise de Requisitos ☹️ Deixaram de atender RF/RN devido má especificação_DT_ER ☹️ Requisitos não atendidos observados na entrega_DT_Validação ☹️ Não entregam tudo que foi solicitado_DT_Gerência de Requisitos
Práticas que podem reduzir e/ou tratar DT Requisitos	<ul style="list-style-type: none"> ☺️ Minimizar ou tratar a DT em requisitos ☺️ Atender a Elicitação de Requisitos ☺️ Auxiliar na Análise de Requisitos ☺️ Apoiar a implementação da ESR ☺️ Implementar a Validação de Requisitos ☺️ Auxiliar na Gerência de Requisitos

Fonte: Elaborado pelo autor na ferramenta MaxQda (2021)

APÊNDICE E – CODIFICAÇÃO FOCALIZADA



Observação: As denominações das categorias e subcategorias que constam na versão final da codificação focalizada foram preservadas

Categoria central – Fenômeno: Fatores que impactam na Engenharia de Requisitos		
Categorias	Subcategorias	Evidências
Causas que geram DT em Requisitos	Falha no recebimento dos requisitos_DT_Elicitação de Requisitos	Falta esclarecer melhor a demanda recebida inicialmente cliente
	Falha no refinamento dos requisitos_DT_Análise de Requisitos	Ausência de Requisitos
		Atrasam o desenvolvimento de algumas demandas/Requisitos
	Inconsistência na Especificação de Requisitos	Desenvolvem sem ter especificado os requisitos
		Falha na especificação dos requisitos
	Falta validarem antes de desenvolverem_DT_Validação	Não validam antes de desenvolver - não era conforme esperado
	Ausência de monitoramento dos Req DT_Gerência de Requisitos	Falta de monitoramento_rastreabilidade nos requisitos
		Backlog muito extenso difícil de gerenciar - implica tempo
Consequências que caracterizam DT em Requisitos	Inconsistência nos RF/RN_DT_Análise de Requisitos	Inconsistência dos Requisitos/Ausência (RF e RN)
	Deixaram de atender RF/RN devido má especificação_DT_ER	Faltaram funcionalidades e só perceberam com feedback do cliente após entrega
		Falta atender requisitos devido à má especificação
	Requisitos não atendidos observados na entrega_DT_Validação	Requisitos não atendidos, percebidos após a entrega
	Não entregam tudo que foi solicitado_DT_Gerência de Requisitos	Não desenvolvem tudo - Acúmulo/excesso do backlog com tempo
		Requisitos/demandas são descartadas/devido mudança de escopo
		Não entregam tudo que foi solicitado - Falta rastreabilidade nos req





Práticas que podem reduzir e/ou tratar DT	Minimizar ou tratar a DT em requisitos	DT avaliam impacto, costumam pagar nas próximas sprints
	Atender a Elicitação de Requisitos	Registrar todos os requis/demandas/histórias no Backlog
		Construir fluxo do processo/BDD para entender a demanda
		Realizar levantamento de REQ através de workshops multidisciplinares
	Auxiliar a Análise de Requisitos	Realizar refin requis/demandas - sprint/release - p técnica UX
		Definir e validam o que será priorizado
		Realizar cerimônias (iniciais, planning sprint, pós refin)
		Registrar e classificar RF e RN – utilizam Checklist para RN
		Seguir um processo requisitos
	Apoiar a implementação da ESR	Manter e descrever com clareza espec requisitos
		Desenvolver protótipos parte da especificação de requ
		Elaborar diagramas (DS, DUC, modelo do banco de dados)
	Implementar a Validação de Requisitos	Validar protótipos após especificação requisitos - cliente
		Realizar experimentações/protótipos/MVP pesquisa usuários
	Auxiliar na Gerência de Requisitos	Gerenciar os Requisitos/entregas/atrasos/backlog/andamento
		Realizar rastreabilidade dos requisitos

Fonte: Elaborado pelo autor (2021)

APÊNDICE F – EVIDÊNCIAS DAS SUBCATEGORIAS DA CATEGORIA “CAUSAS QUE GERAM DT EM REQUISITOS”

Causas que geram DT em Requisitos	Evidências - Memorandos ¹⁶
Falha no recebimento dos requisitos_DT_Elicitação de Requisitos	 Falta esclarecer melhor a demanda recebida inicialmente cliente 
Como as evidências se manifestaram nas entrevistas	
<p>A subcategoria presente nas causas da DT denominada de “falta esclarecer melhor a demanda recebida inicialmente cliente” pode ocorrer por vários fatores, como: (ia) a dificuldade de percepção inicial que pode ser resultante da pouca experiência do Product Owner; (ii) porque nem sempre o cliente sabe o que ele de fato precisa; e (iii) devido às solicitações que chegam pelo suporte e que não foram bem compreendidas. Em contrapartida, essa condição pode levar a uma falta de entendimento do que o cliente necessita, gerando uma falha na concepção dos requisitos e na sua especificação, causando inconsistência entre o que o cliente gostaria e o que foi entregue ou apresentado a ele na validação.</p>	





Fonte: Elaborado pelo autor considerando os relatos dos participantes da pesquisa (2021)

Causas que geram DT em Requisitos	Evidências - Memorandos
Falha no refinamento dos requisitos_DT_Análise de Requisitos	 Ausência de Requisitos  Atrasam o desenvolvimento de algumas demandas/Requisitos  
Como as evidências se manifestaram nas entrevistas	
<p>Uma causa de DT relevante para o estudo caracteriza-se pela “ausência de requisitos” percebidos durante o refinamento das demandas, e um dos problemas que dão origem a essa causa é quando o suporte (entrada de novas solicitações, além do recebimento por parte do PO) recebe a demanda e tem dificuldade de absorver requisitos fundamentais para o negócio. Outro fator que pode originar a ausência de requisitos é a falta de entendimento ou falta de alinhamento inicial, ou seja, algo que não foi mapeado e refinado após o recebimento da solicitação, e foi percebido apenas após uma entrega que determinada solicitação deveria ser melhor refinada durante a análise para evitar que essa ausência fosse identificada apenas somente a entrega.</p> <p>Outra situação que caracteriza uma causa de DT está relacionada ao atraso do desenvolvimento de algumas demandas/requisitos solicitados, ocorrendo devido à uma estimativa realizada inicialmente. No momento do refinamento dos requisitos, percebe-se que não será possível desenvolver essa demanda/requisitos no tempo estimado, sendo viável desenvolver apenas parte do que foi acordado conforme o prazo determinado. O restante do desenvolvimento da demanda será realizado posteriormente, incorrendo numa DT no que foi solicitado.</p>	


Fonte: Elaborado pelo autor considerando os relatos dos participantes da pesquisa (2021)

Causas que geram DT em Requisitos	Evidências - Memorandos
-----------------------------------	-------------------------



¹⁶ Anotações registradas durante a codificação de dados para ajudar o pesquisador a fundamentar seu modelo teórico.

Inconsistência na Especificação de Requisitos	na de	 Desenvolvem sem ter especificado os requisitos - gera retrabalh  Falha na especificação de requisitos	 
Como as evidências se manifestaram nas entrevistas			
<p>Uma das causas da DT condicionada à especificação de requisitos caracteriza-se pelo fato de desenvolver as demandas/requisitos sem ter-se especificado os requisitos, o que pode comprometer o que está sendo entregue, gerando inconsistências entre o que o cliente gostaria e o que de fato foi entregue, contraindo uma dívida técnica em requisitos devido à ausência de requisitos na versão.</p> <p>A evidência “falha na especificação de requisitos” caracteriza-se pelo fato de o requisito ter sido entendido de uma maneira equivocada e foi especificado dessa forma.</p>			

Fonte: Elaborado pelo autor considerando os relatos dos participantes da pesquisa (2021)



Causas que geram DT em Requisitos	Evidências - Memorando
Falta validarem antes de desenvolverem_DT_Validação	 Não validam antes de desenvolver - não era conforme esperado
Como as evidências se manifestaram nas entrevistas	
<p>A evidência voltada à ausência de validação antes do desenvolvimento pode ocorrer devido ao grande volume de solicitações recebidas associadas ao tempo programado para as regras, impossibilitando, muitas vezes, realizar a validação com o cliente antes de ser desenvolvido. Quando essa solicitação é entregue, o cliente informa que não está de acordo com o que havia requerido, gerando retrabalho, além da dívida contraída.</p>	

Fonte: Elaborado pelo autor considerando os relatos dos participantes da pesquisa (2021)





Causas que geram DT em Requisitos	Evidências - Memorandos
Ausência de monitoramento nos Req DT_Gerência de Requisitos	 Falta de monitoramento_rastreabilidade nos requisitos  Backlog muito extenso difícil de gerenciar - implica tempo
Como as evidências se manifestaram nas entrevistas	
<p>Uma evidência de destaque na subcategoria em questão é a falta de monitoramento e rastreabilidade nos requisitos, os quais precisam ser desenvolvidos, aqueles que foram desenvolvidos e aqueles que estão pendentes. Muitas vezes, a ausência desse tratamento e cuidado com os requisitos pode ocasionar inconsistência nos requisitos considerando o que foi solicitado e o que de fato foi entregue, ocorrendo porque determinadas funcionalidades foram esquecidas ou passaram despercebidas, gerando, assim, uma DT em requisitos. Muitas vezes, isso é identificado apenas após a implantação da versão/produto de software no stakeholder.</p> <p>Outro elemento resultante da categoria "Causas da DT referente à Gerência de Requisitos" está relacionado ao Backlog muito extenso, que ocorre devido ao volume de solicitações recebidas e que estão na fila de prioridades para serem desenvolvidas. No entanto, devido a esse volume, acabam contraindo uma dívida técnica em requisitos, prejudicando o tempo estimado para atender a essas solicitações, afetando, até mesmo, outros projetos que dependem do desenvolvimento dessas solicitações, dificultando, assim, o gerenciamento desse backlog.</p>	

Fonte: Elaborado pelo autor considerando os relatos dos participantes da pesquisa (2021)



**APÊNDICE G – EVIDÊNCIAS DAS SUBCATEGORIAS DA CATEGORIA
“CONSEQUÊNCIAS QUE CARCATERIZAM DT EM REQUISITOS”**

Consequências que caracterizam DT em Requisitos	Evidências - Memorandos
Inconsistência RF/RN_DT_Análise Requisitos nos de	 Inconsistência dos Requisitos/Ausência (RF e RN) 
Como as evidências se manifestaram nas entrevistas	
<p>As inconsistências dos requisitos caracterizam a ausência de requisitos funcionais e não funcionais durante o refinamento das solicitações recebidas e acontecem porque não perceberam essa necessidade durante o processo de refinamento. Essas inconsistências caracterizam retrabalho para o time, pois, quando essas forem percebidas, o desenvolvimento pode estar bem evoluído e isso implica em retornar a atividades anteriores para corrigir a falha.</p>	

Fonte: Elaborado pelo autor considerando os relatos dos participantes da pesquisa (2021)







Consequências que caracterizam DT em Requisitos	Evidências - Memorandos
Deixaram de atender RF/RN devido má especificação	 Faltaram func e só perceberam feedback do cliente após entrega  Falta atender requisitos devido a má especificação  
Como as evidências se manifestaram nas entrevistas	
<p>A evidência “faltaram funcionalidades e só perceberam com feedback do cliente após a entrega” ocorre porque possivelmente não estava contemplado na especificação de requisitos, passou despercebido e só foi observado com feedback do cliente, porém, nesse cenário, o relato aponta para a realização de uma força tarefa para “pagar” essa pendência.</p> <p>A falta de atendimento aos requisitos também é uma realidade presente em determinadas organizações, acontecendo devido a uma especificação inadequada, prejudicando o desenvolvimento do que foi solicitado, pois a especificação de requisitos não acontece de maneira adequada, com falhas nas informações dificultando a compreensão dos requisitos.</p>	

Fonte: Elaborado pelo autor considerando os relatos dos participantes da pesquisa (2021)

Consequências que caracterizam DT em requisitos	Evidências - Memorandos
Requisitos não atendidos observados na entrega_DT_validação	 Requisitos não atendidos percebidos após a entrega 
Como as evidências se manifestaram nas entrevistas	



A evidência “requisitos não atendidos percebidos após a entrega” sinaliza a necessidade de manter técnicas e artefatos que conduzem a aplicação da validação dos requisitos durante o desenvolvimento das solicitações/demandas recebidas, pois evitaria retrabalho antes de entregar a versão ou o produto para o stakeholder.

Fonte: Elaborado pelo autor considerando os relatos dos participantes da pesquisa (2021)







Consequências que caracterizam DT em requisitos	Evidências - Memorandos
Não entregam tudo que foi solicitado_DT_Gerência de Requisitos	 Não desenvolvem tudo - Acúmulo/excesso do backlog com tempo   Requisitos/demandas são descartadas/devido mudança de escopo   Não entregam tudo que foi solicitado - Falta rastreab nos req 
Como as evidências se manifestaram nas entrevistas	
<p>O acúmulo/excesso do backlog contraído ao longo do tempo dificulta o gerenciamento e prejudica o que precisa ser atendido, pois, muitas vezes, não se adota nenhuma prática que facilitaria esse monitoramento. Esse acúmulo ocorre devido às mudanças de prioridades, ocasionando que determinados requisitos podem ser desconsiderados e nunca desenvolvidos, com o backlog cada vez mais aumentando de forma desproporcional ao que está sendo desenvolvido.</p> <p>A evidência “requisitos/demandas são descartados devido a mudança de escopo” é ocasionada pela ausência de gerenciamento dos requisitos/demandas solicitadas, pois há requisitos que permanecem por muito tempo no backlog e podem nunca ser desenvolvidos, sendo descartados porque ocorreu uma alteração no escopo ou porque sugeriram evoluções do produto.</p> <p>"Não entregam tudo o que foi acordado" é um problema recorrente em várias organizações, ocorrendo devido à ausência de rastreabilidade dos requisitos, pois as empresas não possuem nenhuma prática para permitir o rastreamento dos requisitos, observando o que foi desenvolvido e o que ainda está pendente conforme as solicitações iniciais.</p>	

Fonte: Elaborado pelo autor considerando os relatos dos participantes da pesquisa (2021)

APÊNDICE H – EVIDÊNCIAS DAS SUBCATEGORIAS DA CATEGORIA “PRÁTICAS QUE PODEM REDUZIR E/OU TRATAR DT REQUISITOS”

Práticas que podem reduzir e/ou tratar DT requisitos	Evidências-Memorandos
Minimizar ou tratar a DT em requisitos	 DT avaliar impacto, costumam pagar nas próximas sprints 
Como as evidências se manifestaram nas entrevistas	
<p>Antes do pagamento da DT, deve-se avaliar o impacto dessa DT para não prejudicar o restante do projeto e não interferir na versão do cliente final. Ainda, quando ocorrer o pagamento, deve-se estar preparado para esses impactos e resolvê-los. Após análise do impacto da DT com o time, dependendo do grau de prioridade da DT, estas são pagas imediatamente, ou na próxima sprint, destacando que uma sugestão é aumentar a cobertura dos testes, porém, não é uma garantia que as DT em requisitos não vão mais ocorrer, mas pode ser uma alternativa para minimizá-la.</p>	

Fonte: Elaborado pelo autor considerando os relatos dos participantes da pesquisa (2021)

Práticas que podem reduzir e/ou tratar DT requisitos	Evidências-Memorandos
Atender a Elicitação de Requisitos	 Registrar todos os requis/demandas/historias no Backlog  Construir fluxo do processo/BDD para entender a demanda  Realizar levant de REQ através de workshops multidisciplinares   
Como as evidências se manifestaram nas entrevistas	
<p>Uma das evidências importantes no processo de elicitação é o registro das demandas e/ou requisitos/histórias que chegam do PO ou através do cliente final. Normalmente, são registradas numa ferramenta de gestão de projetos (como o Redmine ou Jira, por exemplo) formando o Backlog, para, posteriormente, dar os encaminhamentos necessários, pois, documentando as solicitações que chegam, haverá condições de amparar as próximas etapas da Engenharia de Requisitos. As demandas podem ser internas ou externas: (i) as externas chegam pelo suporte ou PO e são analisadas, aprovadas ou não, e, no caso de serem aprovadas, entram no backlog para refinamento posterior, após serem priorizadas na etapa seguinte; (ii) as demandas internas normalmente são sugestões de melhorias e chegam por diferentes setores e papéis na empresa (como diretoria ou <i>business consultant</i>) ou até mesmo como proposta a partir de uma análise do mercado observando novas oportunidades e se transformando em um projeto inovador e promissor, podendo estar no roadmap, e após aprovadas, entram para o backlog.</p> <p>Algumas empresas relataram que é necessário elaborar o fluxograma do processo, demonstrando as funcionalidades e regras do negócio para ter compreensão maior sobre a demanda/requisito solicitada, podendo-se utilizar <i>Business Process Model and Notation</i> (BPNM) ou formato BDD (<i>Behavior Driven Development</i>), por exemplo.</p> <p>Uma proposta para elicitação de requisitos é através de workshops multidisciplinares com sessão de <i>design thinking</i> para encontrar/confirmar soluções específicas.</p>	

Práticas que podem reduzir e/ou tratar DT requisitos	Evidências - Memorandos
Auxiliar na Análise de Requisitos	<ul style="list-style-type: none"> Realizar refin requis/demandas - sprint/release - p técnica UX Definir e validar o que será priorizado Realizar cerimônias (iniciais, planning sprint, pós refin) Registrar e classificar RF e RN - utilizam Checklist para RN Seguir um processo requisitos
Como as evidências se manifestaram nas entrevistas	
<p>Durante o refinamento dos requisitos, realiza-se a identificação dos requisitos, os quais devem ser classificados em funcionais e não funcionais. Para melhor compreensão, e para evitar DT em requisitos, pode ser utilizado um checklist para conferir os requisitos não funcionais necessários, pois, dependendo do produto, o requisito não funcional é essencial. Esse checklist orienta os colaboradores sobre as restrições importantes que devem ser seguidas.</p> <p>Na Análise de requisitos, é importante manter algumas cerimônias que ocorrem para evitar ao máximo alguma incoerência no entendimento da demanda e dos requisitos que a compõem, através de reuniões adotadas pelas empresas, as quais acontecem para apresentar a demanda para os membros do time e dar sequência ao refinamento do que será priorizado; cerimônias para esclarecer alguma dúvida referente à demanda antes do refinamento dos requisitos e do planejamento da sprint e da sua especificação através de brainstorming; realizam algumas reuniões pós-refinamento através de sessões de <i>design thinking</i>.</p> <p>Todas as demandas devem ser validadas e priorizadas durante a realização de reuniões para entrarem nas sprints ou releases e normalmente as priorizações ocorrem conforme a orientação do PO, de acordo com a ordem de grandeza/urgência e, a partir dessas priorizações, define-se a ordem de execução dessas demandas nas sprints ou releases.</p> <p>Outra prática adotada na análise de requisitos é manter um processo de requisitos definido, porém, pode ser adaptado conforme a demanda/requisitos que chegam, pois, conforme a complexidade da demanda, é importante que haja determinadas práticas para evitar inconsistências nos requisitos. Nesse contexto, apresenta-se alguns aspectos comuns a diferentes processos adotados por algumas empresas e que podem auxiliar a minimizar a ocorrência de dívidas: 1. possuem documento inicial com a visão do processo; 2. elaboram documento técnico; 3. pode ou não ter aprovação do cliente; 4. documento é refinado e quebrado em partes(tarefas); 5. tarefas validadas pelo time, estima-se o tempo de desenvolvimento.</p>	

Fonte: Elaborado pelo autor considerando os relatos dos participantes da pesquisa (2021)

Práticas que podem reduzir e/ou tratar DT requisitos	Evidências - Memorandos
Apoiar a implementação da ESR	<ul style="list-style-type: none"> Manter e descrever com clareza documento de espec requisitos Desenvolver protótipos parte do documento especificação de requ Elaborar diagramas (DS, DUC, modelo do banco de dados)
Como as evidências se manifestaram nas entrevistas	
<p>Uma necessidade no processo de requisitos é manter um registro das funcionalidades previstas na demanda/projeto/produto de software, classificando em requisitos funcionais e não funcionais preferencialmente, o qual deve manter uma linguagem clara para facilitar ao máximo a compressão do restante do time. Esse registro é necessário justamente porque os devs, por exemplo, podem ter acesso a ele, o</p>	

qual também é possível estar associado a outras informações mais técnicas que podem auxiliar no entendimento dos devs, como, por exemplo, as dependências de outros sistemas.

Para fortalecer a Especificação de Requisitos, algumas empresas elaboram artefatos, como, por exemplo, diagrama de casos de uso, diagrama de sequência, bem como, um modelo preliminar do banco de dados, necessários para visualizar o todo e não uma parte do que precisa ser desenvolvido, sempre que forem necessários, dependendo da complexidade e impacto das novas demandas/requisitos. Ainda, em algumas situações, também são desenvolvidos protótipos e incorporados ao ESR para dar clareza maior sobre o que deve ser desenvolvido.

Fonte: Elaborado pelo autor considerando os relatos dos participantes da pesquisa (2021)

Práticas que podem reduzir e/ou tratar DT requisitos	Evidências - Memorandos
Implementar a Validação de Requisitos	<ul style="list-style-type: none"> Validar protótipos após especificação requisitos - cliente Realizar experimentações/protótipos/MVP pesquisa usuários
Como as evidências se manifestaram nas entrevistas	
<p>A validação de requisitos através de protótipo deveria ser uma prática frequente nas empresas, pois poderia evitar muita incoerência nos requisitos. A prototipação tipicamente ocorre após a especificação dos requisitos, sendo, em muitos casos, desenvolvida pelos profissionais da área de UX/IU da empresa, acompanhados pelo restante do time, permitindo aos stakeholders uma representação visual muito próxima do que será desenvolvido, o que facilita muito a aprovação ou não do que está sendo apresentado.</p> <p>Uma prática utilizada pelas empresas durante a validação é a realização de experimentos através de protótipos ou MVP evidenciando pesquisas com usuários sobre o que será desenvolvido. Essa prática é interessante especialmente quando são solicitações que podem causar impacto no restante do produto/projeto, absorvendo diferentes percepções de uma amostra de stakeholders, procurando manter a essência e coerência das demandas/requisitos solicitados. No entanto, é uma sugestão que pode ser aplicada apenas quando a demanda for complexa ou pode causar impacto em outras funcionalidades.</p>	

Fonte: Elaborado pelo autor considerando os relatos dos participantes da pesquisa (2021)

Propondo práticas que podem reduzir e/ou tratar DT requisitos	Evidências – Memorandos
Auxiliar na Gerência de Requisitos	<ul style="list-style-type: none"> Gerenciar os Requisitos/entregas/atrasos/backlog/andamento Realizar rastreabilidade dos requisitos
Como as evidências se manifestaram nas entrevistas	
<p>Com a Gerência de Requisitos, é possível controlar e monitorar o andamento dos requisitos, através de seu status, como, por exemplo, se já foi atendido, se precisa ser desenvolvido e se está atrasado. Também é possível acompanhar o que está no backlog e o tempo que tal demanda/requisitos permanece lá. O histórico alimentado na gerência de requisitos permite identificar os requisitos que ainda estão aguardando para serem desenvolvidos. Havendo algo urgente, pode ser ajustado na próxima sprint. Para gerenciar os requisitos, pode-se utilizar ferramentas e, a partir delas, graficamente, consegue-se explorar a visualização do status de cada funcionalidade/requisitos através de gráficos representativos.</p>	

A rastreabilidade nos requisitos é importante e indicada porque permite manter uma referência/caminho de determinada funcionalidade e isso justifica sua necessidade, pois, através da rastreabilidade, é possível manter um histórico das funcionalidades, a relação entre funcionalidades, demandas e de tudo o que será e foi desenvolvido.

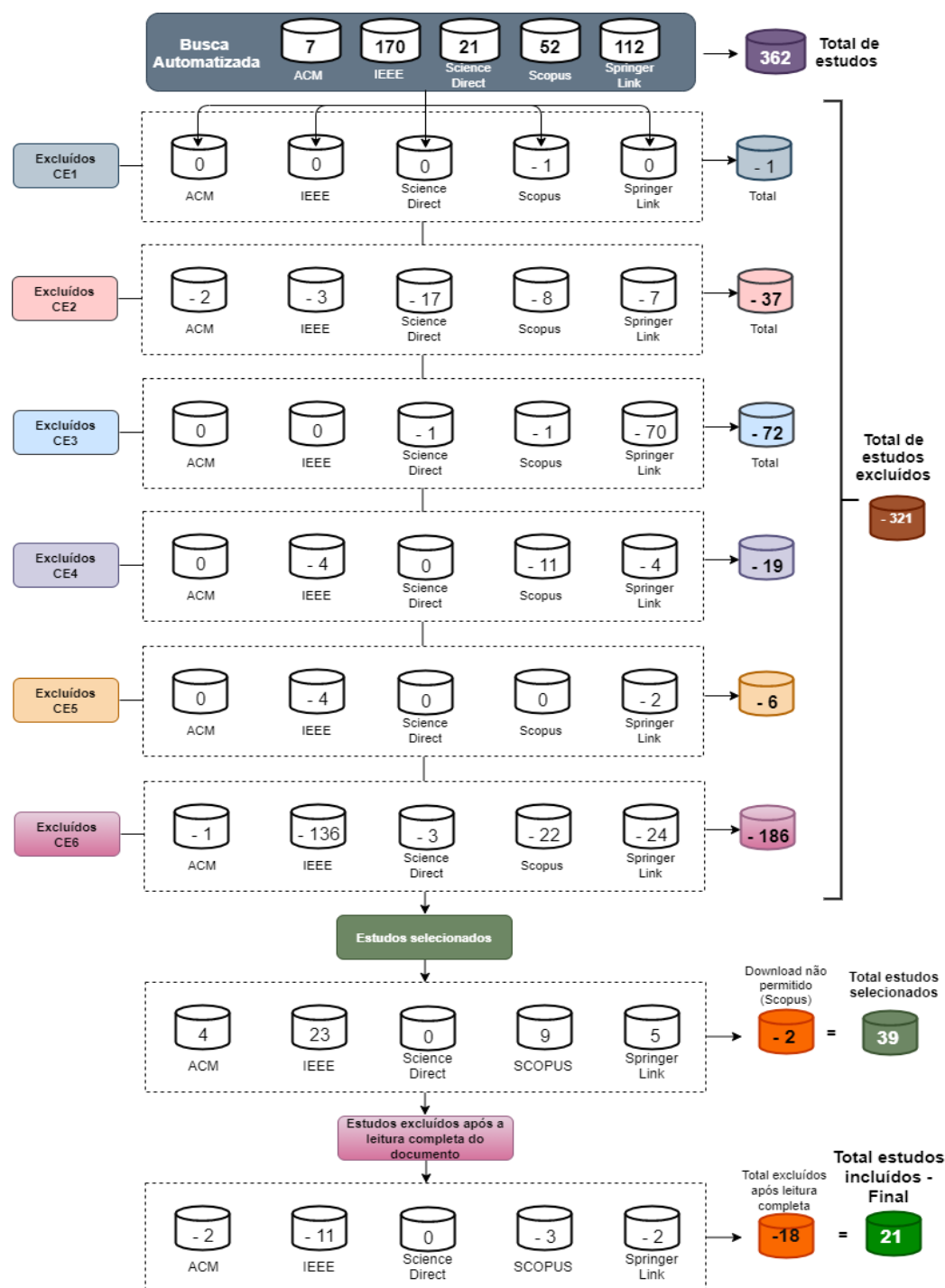
Fonte: Elaborado pelo autor considerando os relatos dos participantes da pesquisa (2021)

**APÊNDICE I – STRINGS DE BUSCA X MECANISMOS DE BUSCA –
MAPEAMENTO SISTEMÁTICO DA LITERATURA**

Mecanismos de busca	String de busca	#
ACM Digital Library	[[Publication Title: "technical debt"] OR [Publication Title: "requirements debt"]] AND [[Full Text: "requirements engineering"] OR [Full Text: "software requirements"]] AND [[Full Text: agile] OR [Full Text: kanban] OR [Full Text: scrum] OR [Full Text: "extreme programming"] OR [Full Text: lean]] AND [Publication Date: (01/01/2016 TO 12/31/2021)]	7
IEEEExplore	("Document Title":"technical debt" OR "Document Title":"requirements debt") OR ("Abstract":"technical debt" OR "Abstract":"requirements debt") AND ("Full Text Only":"requirements engineering" OR "Full Text Only":"software requirements") AND ("Full Text Only":agile OR "Full Text Only":scrum OR "Full Text Only":"extreme programming" OR "Full Text Only":kanban OR "Full Text Only":lean)	170
Science Direct	Title, abstract, keywords: ("technical debt" OR "requirements debt") AND ("software requirements" OR "requirements engineering") AND ("agile" OR "kanban" OR "scrum" OR "extreme programming" OR "lean")	21
Springer Link	("technical debt" OR "requirements debt") AND ("software requirements" OR "requirements engineering") AND ("agile" OR "kanban" OR "scrum" OR "extreme programming" OR "lean")	112
Scopus	(TITLE-ABS-KEY ("technical debt" OR "requirements engineering" OR "software requirements") AND ALL ("scrum" OR "lean" OR "agile" OR "extreme programming" OR "kanban") AND PUBYEAR > 2015	52
Total		362

Fonte: Elaborado pelo autor (2021)

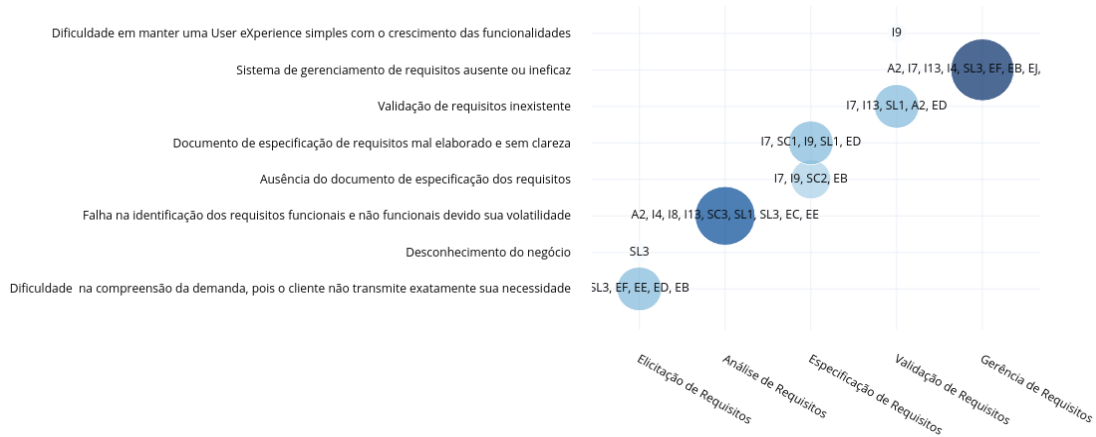
APÊNDICE J – PROCESSO DE SELEÇÃO DOS ESTUDOS



Fonte: Elaborado pelo autor (2021)

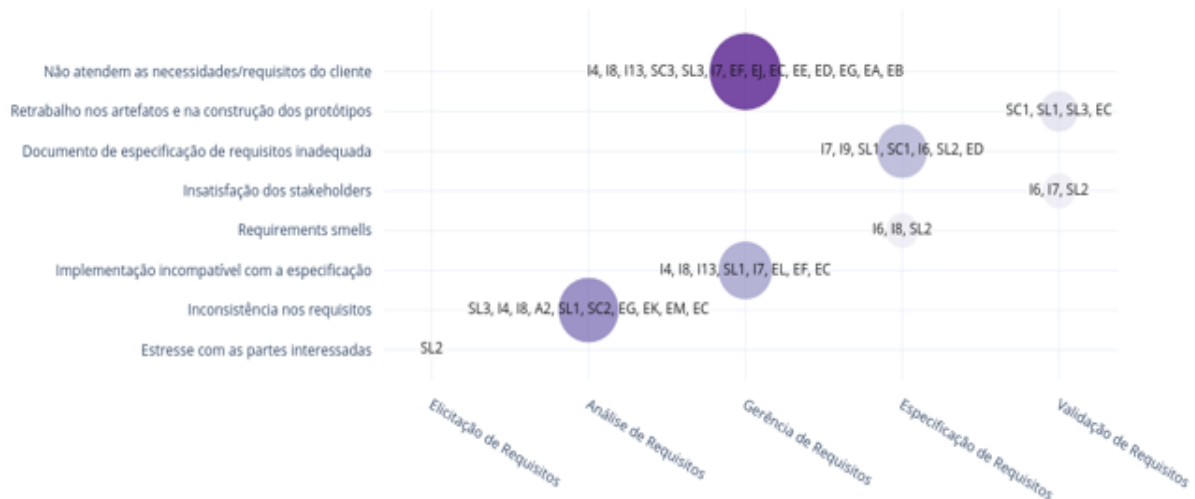
APÊNDICE K - UNIFICAÇÃO DAS EVIDÊNCIAS DA VISÃO PRÁTICA E DO MSL – CAUSAS DAS RED, SUAS CONSEQUÊNCIAS E PRÁTICAS PARA MITIGÁ-LAS

Unificação das evidências da visão prática e do MSL – Causas que ocasionam dívidas de requisitos



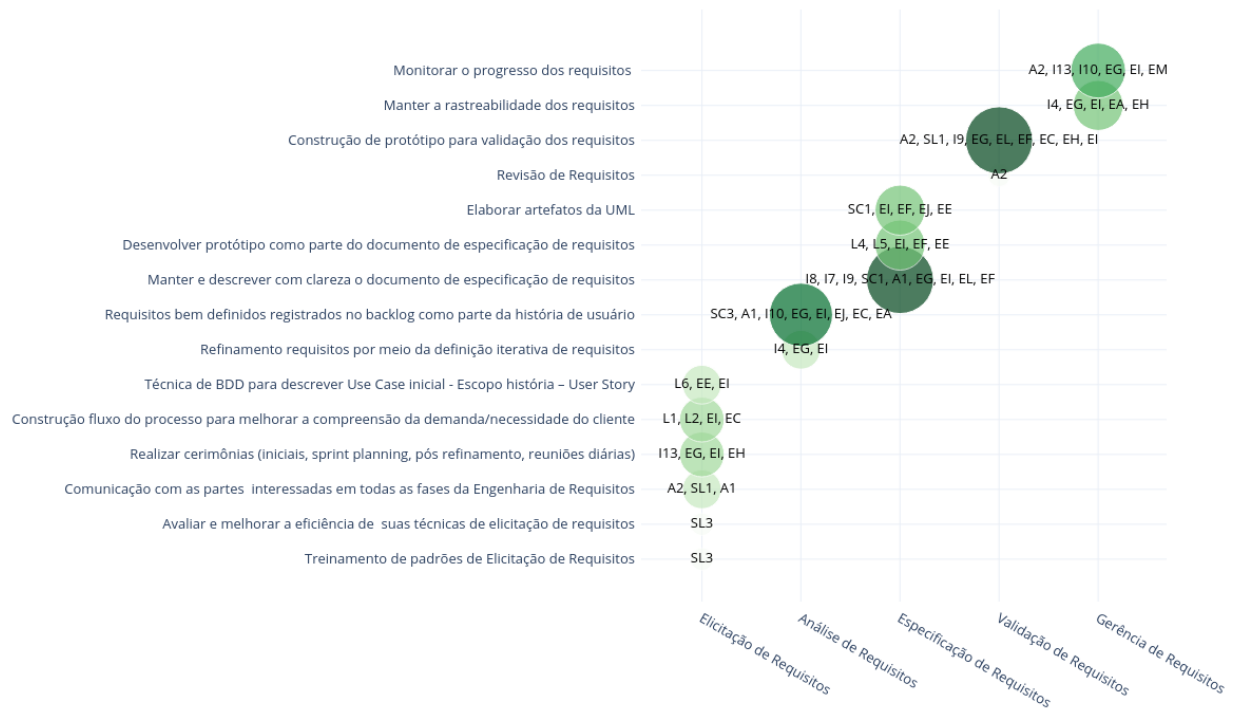
Fonte: Elaborado pelo autor (2022)

Unificação das evidências da visão prática e do MSL - Consequências das dívidas de requisitos



Fonte: Elaborado pelo autor (2022)

Unificação das evidências da visão prática e do MSL – Práticas para mitigar as causas das dividas de requisitos



Fonte: Elaborado pelo autor (2022)

APÊNDICE L – RESULTADO DA UNIFICAÇÃO DAS DÍVIDAS DE REQUISITOS E DAS SUAS CAUSAS

Resultados das dívidas – consequências das dívidas de requisitos	Evidência da Literatura	Evidência na empresa	Resultado das causas das dívidas	Evidência da Literatura	Evidência na empresa
Inconsistência nos requisitos	SL3, I4, I8, A2, SL1, (SC2 ¹⁷).	EG, EK, EM, EC	Dificuldade na compreensão da demanda, pois o cliente não consegue transmitir exatamente sua necessidade	SL3	EF, EE, ED, EB
			Falha na identificação dos requisitos funcionais e de qualidade (RNF) devido sua volatilidade (requisitos mal formulados, incompletos, com pouca clareza ou ambíguo)	A2, I4, I8, SL1, SL3, (SC3, I13)	EC, EE
			Sistema de gerenciamento de requisitos ausente ou ineficaz	A2, I4, SL3, (I7, I13)	EF, EB, EJ, EC, EE
Implementação incompatível com a especificação tratei como gerência	I4, I8, I13, SL1, I7	EL, EF, EC	Dificuldade na compreensão da demanda, pois o cliente não consegue transmitir exatamente sua necessidade	SL3	EF, EE, ED, EB
			Falha na identificação dos requisitos funcionais e de qualidade (RNF) devido sua volatilidade (requisitos mal formulados, incompletos, com pouca clareza ou ambíguo)	I4, I8, I13, SL1, (A2, SL3, SC3)	EC, EE
			Validação de requisitos inexistente	I13, SL1, I7	ED
			Sistema de gerenciamento de requisitos ausente ou ineficaz	I13, I4, I7, (A2, SL3)	EF, EB, EJ, EC, EE

¹⁷ As evidências das consequências e das causas que estão em parênteses sinalizam que o estudo apontou a evidência, que se caracterizou como consequência por exemplo, mas não mencionou nenhuma evidência que fosse possível caracterizar como causa, e o inverso também ocorreu.

Documento de especificação de requisitos inadequada	I7, I9, SL1, SC1, SL2 I6	ED	Ausência do documento de especificação dos requisitos (RF e RNF)	I7, I9, (SC2)	ED, EB
			Dificuldade em manter uma User eXperience (UX) simples com o crescimento das funcionalidades	I9	-
			Documento de especificação de requisitos mal elaborado e sem clareza	SC1, SL1, I9, I7, I6, SL2	ED
Requirements smells	I8, (I6, SL2)	-	Falha na identificação dos requisitos funcionais e de qualidade (RNF) devido sua volatilidade (requisitos mal formulados, incompletos, com pouca clareza ou ambíguo)	I8, (A2, I4, I13, SC3, SL1, SL3)	EC, EE
Retrabalho de artefatos e na construção dos protótipos	SL1, SL3, (SC1)	EC	Dificuldade na compreensão da demanda, pois o cliente não consegue transmitir exatamente sua necessidade	SL3	EF, EE, ED, EB
			Falha na identificação dos requisitos funcionais e de qualidade (RNF) devido sua volatilidade (requisitos mal formulados, incompletos, com pouca clareza ou ambíguo)	SL1, SL3, (A2, I4, I8, I13, SC3)	EC, EE
			Sistema de gerenciamento de requisitos ausente ou ineficaz	SL3, (A2, I7, I4, I13)	EF, EB, EJ, EC, EE
Insatisfação dos stakeholders	I7, (I6, SL2)	-	Ausência do documento de especificação dos requisitos (RF e RNF)	I7, (I9, SC2)	ED, EB
			Validação de requisitos inexistente	I7, (I13, SL1)	ED

			Sistema de gerenciamento de requisitos ausente ou ineficaz	I7, (A2, I13, I4, SL3)	EF, EB, EJ, EC, EE
Não atendem as necessidades/requisitos do cliente	I4, I8, I13, SC3, SL3, I7	EF, EJ, EC, EE, ED, EG, EA, EB	Dificuldade na compreensão da demanda, pois o cliente não consegue transmitir exatamente sua necessidade	SL3	EF, EE, ED, EB
			Falha na identificação dos requisitos funcionais e de qualidade (RNF) devido sua volatilidade (requisitos mal formulados, incompletos, com pouca clareza ou ambíguo)	I4, I8, I13, SC3, SL3, (A2, SL1)	EC, EE
			Ausência do documento de especificação dos requisitos (RF e RNF)	I7, (I9, SC2)	EB
			Documento de especificação de requisitos mal elaborado e sem clareza	I7, (SC1, I9, SL1)	ED
			Validação de requisitos inexistente	I13 (I7, SL1)	ED
			Sistema de gerenciamento de requisitos ausente ou ineficaz	I13, SL3, (A2, I7, I4)	EF, EB, EJ, EC, EE

Fonte: Elaborado pelo autor (2023)

APÊNDICE M - RESULTADO DA UNIFICAÇÃO DAS CAUSAS DAS DÍVIDAS DE REQUISITOS E DAS PRÁTICAS QUE PODERÃO MITIGÁ-LAS – ESTADO DA PRÁTICA E MAPEAMENTO SISTEMÁTICO DA LITERATURA

Resultados das causas obtidos no Mapeamento Sistemático da Literatura (MSL)/Visão prática (empresa)	Evidência Literatura (MSL)	Evidência a empresa	Resultados das práticas obtidos no Mapeamento Sistemático da Literatura (MSL)/Visão prática (empresa)	Evidência Literatura (MSL)	Evidência a empresa	Evidência Literatura
Dificuldade na compreensão da demanda, pois o cliente não consegue transmitir exatamente sua necessidade	SL3	EF, EE, ED, EB	Treinamento de padrões de elicitação de requisitos que enfatizem a forma como os requisitos devem ser elicitados e especificados	SL3	-	-
			Avaliar e melhorar a eficiência de suas técnicas de elicitação	SL3	-	-
Desconhecimento do negócio	SL3	-	Treinamento de padrões de elicitação de requisitos que enfatizem a forma como os requisitos devem ser elicitados e especificados	SL3	-	-
			Avaliar e melhorar a eficiência de suas técnicas de elicitação	SL3	-	-
Falha na identificação dos requisitos funcionais e de qualidade (RNF) devido sua volatilidade (requisitos mal formulados, incompletos, com pouca clareza ou ambíguo)	A2, I4, I8, I13, SC3, SL1, SL3	EC, EE	Comunicação com as partes interessadas em todas as fases da Engenharia de Requisitos	A2, SL1, (A1)	-	-
			Monitorar o progresso dos requisitos, por meio do gerenciamento dos requisitos, estabelecendo uma conexão entre os itens pendentes da dívida com os requisitos	A2, I13 (I10)	EG, EI, EM	-

			Construção de protótipo para validação dos requisitos	A2, SL1, (I9)	EG, EL, EF, EC, EH, EI	-
			Realizar cerimônias (iniciais, sprint planning, pós refinamento, reuniões diárias)	I13	EG, EI, EH	-
			Revisão de Requisitos	A2	-	-
			Refinamento requisitos por meio da definição iterativa de requisitos (trabalham com sprints/releases e técnicas de UX)	I4	EG, EI	-
			Manter e descrever com clareza o documento de especificação de requisitos	I8, (I7, I9, SC1, A1)	EG, EI, EL, EF	-
			Requisitos bem definidos registrados no backlog por meio da existência de regra de negócio como parte da história de usuário baseada nos requisitos do cliente, e com clareza na declaração do escopo.	SC3, (A1, I10)	EG, EI, EJ, EC, EA	-
			Treinamento de padrões de elicitação de requisitos que enfatizem a forma como os requisitos devem ser elicitados e especificados.	SL3	-	-

			Avaliar e melhorar a eficiência de suas técnicas de elicitação	SL3	-	-
			Construção fluxo do processo para melhorar a compreensão da demanda/necessidade do cliente.	-	EI, EC	L1, L2
Ausência do documento de especificação dos requisitos (RF e RNF) – (desenvolvem os requisitos sem apoio do documento de especificação)	I7, I9, SC2	EB	Manter e descrever com clareza o documento de especificação de requisitos	I7, I9, (I8, SC1, A1)	EG, EI, EL, EF	-
			Construção de protótipo para validação dos requisitos	I9 (A2, SL1)	EG, EL, EF, EC, EH, EI	-
			Desenvolver protótipo como parte do documento de especificação de requisitos	-	EI, EF, EE	L4, L5
Documento de especificação de requisitos mal elaborado e sem clareza	I7, C1, I9, SL1	ED	Manter e descrever com clareza o documento de especificação de requisitos	I7, I9, (I8, A1, SC1)	EG, EI, EL, EF	-

			Elaborar artefatos da UML	SC1	EI, EF, EJ, EE	-
			Construção de protótipo para validação dos requisitos	SL1, I9 (A2)	EG, EL, EF, EC, EH, EI	-
			Técnica de BDD para descrever Use Case inicial - Escopo história – User Story	-	EE, EI	L6
			Desenvolver protótipo como parte do documento de especificação de requisitos.	-	EI, EF, EE	L4, L5
Validação de requisitos inexistente – (desenvolvem sem validar os requisitos)	I7, I13, SL1, A2	ED	Manter e descrever com clareza o documento de especificação de requisitos	I7, (A1, I8, I9, SC1)	EG, EI, EL, EF	-
			Realizar cerimônias (iniciais, sprint planning, pós refinamento, reuniões diárias)	I13	EG, EI, EH	-
			Revisão de Requisitos	A2	-	-

			Monitoram o progresso dos requisitos, por meio do Gerenciamento dos requisitos, estabelecendo uma conexão entre os itens pendentes da dívida com os requisitos.	I13, (I10, A2)	EG, EI, EM	-
			Construção de protótipo para validação dos requisitos	SL1, (I9, A2)	EG, EL, EF, EC, EH, EI	-
Sistema de gerenciamento de requisitos ausente ou ineficaz (Não atende totalmente os requisitos solicitados – falta de gerenciamento (NAR) /* Falta de monitoramento – rastreabilidade nos requisitos P e Mudança na necessidade do negócio L SL3)	A2, I7, I13, I4, SL3	EF, EB, EJ, EC, EE	Comunicação com as partes interessadas em todas as fases da Engenharia de Requisitos	A2, (SL1, A1)	-	-
			Monitorar o progresso dos requisitos, por meio do gerenciamento dos requisitos, estabelecendo uma conexão entre os itens pendentes da dívida com os requisitos	A2, I13, (I10)	EG, EI, EM	-
			Construção de protótipo para validação dos requisitos	A2, (SL1, I9)	EG, EL, EF, EC, EH, EI	-

			Realizar cerimônias (iniciais, sprint planning, pós refinamento, reuniões diárias)	I13	EG, EI, EH	-
			Revisão de Requisitos	A2	-	-
			Manter a rastreabilidade dos requisitos	I4	EG, EI, EA, EH	-
			Manter e descrever com clareza o documento de especificação de requisitos	I7, (A1, I8, I9, SC1)	EG, EI, EL, EF	-
			Treinamento de padrões de ER que enfatizem a forma como os requisitos devem ser elicitados e especificados	SL3	-	-
			Avaliar e melhorar a eficiência de suas técnicas de elicitação	SL3	-	-
			Técnica de BDD para descrever Use Case inicial - Escopo história – User Story	-	EI, EE	L6
Dificuldade em manter uma User eXperience (UX) simples com o crescimento das funcionalidades	I9	-	Mantém e descrevem com clareza o documento de especificação de requisitos	I9, (I7, I8, SC1, A1)	EG, EI, EL, EF	-

			Construção de protótipo para validação dos requisitos	I9, (SL1, A2)	EG, EL, EF, EC, EH, EI	-

Fonte: Elaborado pelo autor (2023)

**APÊNDICE N – ESTUDOS PRIMÁRIOS ACRESCENTADOS NO MAPEAMENTO
DAS CAUSAS DE DÍVIDAS DE REQUISITOS VERSUS PRÁTICAS**

Identificação	Referência do estudo
L1	ORDÓÑEZ, Hugo; VILLADA, Andrés Felipe Escobar; VANEGAS, Diana Lorena Velandia; COBOS, Carlos; ORDÓÑEZ, Armando; SEGOVIA, Rocio. An impact study of business process models for requirements elicitation in XP International Conference on Computational Science and Its Applications, 9155, Springer Verlag (2015), pp. 298-312.
L2	TRKMAN, Marina; MENDLING, Jan; TRKMAN, Peter; KRISPER, Marjan.. Impact of the conceptual model's representation format on identifying and understanding user stories. Vol. 116 , 2019. DOI: 10.1016/j.infsof.2019.08.001. Disponível em: https://www.sciencedirect.com/science/article/pii/S095058491930165X . Acesso em: dez. 2022.
L3	OLIVEIRA, Gabriel; MARCZAK, Sabrina. On the Empirical Evaluation of BDD Scenarios Quality: Preliminary Findings of an Empirical Study . IEEE 25th International Requirements Engineering Conference Workshops (REW). 2017. DOI: 10.1109/REW.2017.62. Disponível em: https://ieeexplore.ieee.org/document/8054868 . Acesso em: Dez. de 2022.
L4	KÄPYAHO, Marja; KAUPPINEN, Marjo. Agile requirements engineering with prototyping: A Case Study . IEEE 23rd International Requirements Engineering Conference (RE). 2015. DOI: 10.1109/re.2015.7320450. Disponível em: https://ieeexplore.ieee.org/document/7320450 . Acesso em: Dez. 2022.
L5	ALI, Naveed; LAI, Richard. Method of software requirements specification and validation for global software development . Requirements Engineering, vol. 22, pp. 191–214. 2017. Disponível em: https://link.springer.com/article/10.1007/s00766-015-0240-4 . Acesso em: Dez. de 2022.
L6	LUCASSEN, Garm; DALPIAZ, Fabiano; WERF, Jan Martijn e.m. van der; BRINKKEMPER, Sjaak; ZOWGHI, Didar. Behavior-driven requirements traceability via automated acceptance tests . IEEE 25th International Requirements Engineering Conference Workshops. 2017. DOI: 10.1109/rew.2017.84. Disponível em: https://ieeexplore.ieee.org/document/8054891 . Acesso em: dez.2022.er agil

Fonte: Elaborado pelo autor (2023)

APÊNDICE O – STRINGS DE BUSCA X MECANISMOS DE BUSCA ESTUDO TERCIÁRIO

String de busca		#
Mecanismos de Busca Automatizada		
ACM Digital Library	[[Title: "technical debt" OR "requirements debt"] AND [[Title: "manage"] OR [Title: "managing"] OR [Title: "management"] OR [Title: "monitoring"] OR [Title: "practices"] OR [Title: "techniques"]]] AND [[Title: "systematic literature review"] OR [Title: "systematic mapping"] OR [Title: "systematic study"]]]	01
IEEEExplore	("All Metadata": "technical debt" OR "All Metadata": "requirements debt") AND ("All Metadata": "manage" OR "All Metadata": "managing" OR "All Metadata": "management" OR "All Metadata": "monitoring" OR "All Metadata": "practices" OR "All Metadata": "techniques") AND ("All Metadata": "Systematic literature review" OR "All Metadata": "systematic Mapping" OR "All Metadata": "Systematic study")	10
Science Direct	Title, abstract, keywords: ("technical debt" OR "requirements debt") AND ("manage" OR "managing" OR "management" OR "monitoring") AND ("systematic literature review" OR "systematic Mapping " OR "systematic study")	69
Springer Link	("technical debt" OR "requirements debt") AND ("manage" OR "managing" OR "management" OR "monitoring" OR "practices" OR "techniques") AND ("systematic literature review" OR "systematic mapping" OR "systematic study")	61
Scopus	(TITLE-ABS-KEY ("technical debt" OR "requirements debt") AND ALL ("manage" OR "managing" OR "management" OR "monitoring" OR "practices" OR "techniques") AND ALL ("systematic literature review" OR "systematic mapping" OR "systematic study"))	0
Total Parcial: Busca Automatizada		141
Busca Manual		
Tech Debt ¹⁸ 2022	Busca de estudos secundários voltados à gestão da dívida técnica ou gestão da dívida de requisitos e dívida técnica ou dívida de requisitos.	01
Tech Debt 2021		0
Tech Debt 2020		02
Tech Debt 2019		0
Tech Debt 2018		01
MTD ¹⁹ 2016	Busca de estudos secundários voltados à gestão da dívida técnica ou gestão da dívida de requisitos e dívida técnica ou dívida de requisitos.	0
MTD 2015		01
MTD 2014		0
MTD 2013		0
Total Parcial: Busca Manual		05
Total		146

Fonte: Elaborado pelo autor (2023)

¹⁸ Tech Debt: International Conference on Technical Debt

¹⁹ MTD: International Workshop on Managing Technical Debt

APÊNDICE P – CRITÉRIOS DE QUALIDADE

Critérios	Descrição do Critério de Qualidade	Pontuação ²⁰
CQ1	Os critérios de inclusão (CI) e exclusão (CE) foram definidos adequadamente?	S - 1 pt: CI e CE estão definidos de forma clara e explícita; P – 0,5 pt: CI e CE não estão explícitos; N – 0 pt: CI e CE não estão definidos.
CQ2	A busca automatizada e manual identificou todos os estudos relevantes?	S - 1 pt: Busca foi apoiada em 4 ou mais bibliotecas digitais e estratégias de pesquisa adicionais; P – 0,5 pt: A busca adotou 3 ou 4 bibliotecas digitais sem estratégias de pesquisa extras, ou pesquisas restritas em periódicos e anais de conferências; N – 0 pt – Busca apoiada em até 2 bibliotecas digitais ou um conjunto extremamente restrito de periódicos.
CQ3	Os estudos incluídos foram devidamente sintetizados?	S - 1 pt: Os CQ foram definidos explicitamente; P – 0,5 pt: a questão de pesquisa envolve questões de qualidade que são abordadas pelo estudo; N – 0 pt: nenhuma avaliação de qualidade explícita foi realizada.
CQ4	A qualidade dos estudos incluídos foi avaliada?	S - 1 pt: Os CQ foram definidos explicitamente; P – 0,5 pt: a questão de pesquisa envolve questões de qualidade que são abordadas pelo estudo; N – 0 pt: nenhuma avaliação de qualidade explícita foi realizada.
CQ5	São apresentados detalhes suficientes sobre os estudos individuais incluídos?	S - 1 pt: são apresentadas informações sobre cada artigo como locais de origem E os principais resultados/contribuições E a avaliação da qualidade; P - 0,5 pt: o estudo secundário apresenta informações generalizadas das fontes sobre: os tópicos ou resultados abordados OU a avaliação da qualidade OU classificação das fontes – ou seja, tipos de estudos; N - 0 pt: sem informações sobre as fontes incluídas.

Fonte: Elaborado pelo autor (2023)

²⁰ S = Sim; P = Parcialmente; N = Não.

APÊNDICE Q – RESULTADO DAS QUESTÕES DE PESQUISA DO ESTUDO TERCIÁRIO

QP 1: Quais são as atividades ou áreas da gestão da dívida técnica/gestão da dívida de requisitos? Do estudo terciário

Gestão da Dívida de Requisitos		
Área	Estudos	Descrição
Identificação	SD1, SD3, SD5, SD6, SD7, I1	Visualização da dívida, identificando suas causas e outros aspectos presentes no desenvolvimento de software que levaram à sua existência (SD1, SD7, I1).
Mensuração	SD1, SD3, SD5, SD6, SD7, I1	Responsável por analisar e quantificar os custos e esforços necessários para auxiliar na tomada de decisão referente ao ressarcimento da dívida (SD1, SD7, I1).
Priorização	SD1, SD3, SD6, SD7, I1	Visa organizar o pagamento das dívidas técnicas por ordem de importância, analisando fatores como questões técnicas e implicações financeiras (SD1, SD7, I1).
Reembolso/Pagamento	SD1, SD3, SD5, SD6, SD7, I1	Consiste no pagamento parcial ou total da dívida técnica, evitando o seu adiamento caso afete negativamente o projeto (SD1, SD7, I1).
Monitoramento	SD1, SD3, SD5, SD6, SD7, I1	Consiste em validar se a dívida técnica está sendo reduzida, atrasada ou se permanece gerando custos (SD1, SD7, I1).
Comunicação	SD6, SD7, I1	Responsável por manter a dívida identificada, visível às partes interessadas, para que possa ser discutida e gerenciada posteriormente (SD7, I1)
Documentação	SD6, SD7, I1	Fornecer uma maneira de representar e documentar a dívida de maneira uniforme, abordando as preocupações de partes interessadas específicas (SD7, I1).
Prevenção	SD7, I1	Objetiva verificar os artefatos existentes do projeto com o objetivo de evitar que possíveis dívidas de requisitos sejam incorridas (SD7, I1).

Fonte: Elaborado pelo autor (2023)

QP 2. Quais as práticas sugeridas para as atividades ou áreas da gestão da dívida de requisitos? (Área, práticas gerais e sua descrição)

(Descrição das práticas da gestão da dívida técnica do estudo terciário)

Áreas da Gestão da Dívida Técnica	Práticas	Descrição
Identificação	Avaliação do cliente	O sucesso do trabalho com requisitos de qualidade consiste em envolver as partes interessadas progressivamente, desenvolver listas de requisitos sustentáveis e registrar as pendências existentes. Portanto, revisar os requisitos com o cliente, incluindo a equipe de desenvolvimento, deve ser considerado essencial para toda a gestão. As empresas podem ajustar o produto e os requisitos especificados com base no feedback do cliente para identificar DTs com mais eficiência.
	Comunicação face a face	Em geral, a comunicação sobre os requisitos é hierárquica e baseada em e-mails e documentos. No entanto, é recomendável que os profissionais de software se comuniquem diretamente com seus pares e partes interessadas. A comunicação face a face é eficiente na troca de informações entre diferentes stakeholders, auxiliando na identificação das dívidas técnicas. Portanto, a comunicação face a face deve ser considerada essencial.
	Modelo de documentação	É um modelo que é preenchido no documento DT e contém diversos dados, principalmente referentes à medição. O modelo de documentação da dívida técnica proposta por Seaman e Guo (2011).
	Revisão por pares	Benefícios na revisão de requisitos são destacados na literatura, especialmente na identificação de defeitos e DTs. Dentre as formas de revisão, destaca-se a revisão por pares. Consiste no analista conduzir a entrevista com o cliente e gravar o áudio do diálogo. O áudio é revisado por outro analista (revisor), que anota as ambiguidades e lista as perguntas que faria se fosse o analista. As perguntas são utilizadas para esclarecimentos em futuras interações com o cliente.
	Análise simples de custo-benefício	É criada uma lista com itens de DT, onde cada um representa uma tarefa que ficou por fazer, mas que corre o risco de causar problemas futuros.
	Abordagem de quantificação	Utilizada para quantificar DT e interesse técnico, mas para isso seria necessário responder questões sobre investimentos, como (1) Qual o tamanho do meu DT? (2) Quanto de juros estou pagando pelo DT? (3) A dívida está crescendo? e quanto rápido? (4) Qual será a consequência de manter este DT para manutenção futura? Por fim, a Dívida Técnica pode ser

		quantificada considerando a relação entre as necessidades do usuário já eliciadas e todas as possíveis necessidades do usuário, inclusive as negligenciadas.
	Abordagem do vizinho mais próximo	Essa abordagem aproveita a experiência adquirida em DT previamente resolvidos e conta com o esforço necessário para corrigir esses débitos já identificados. A intuição é que o tempo médio para converter um DT é semelhante à correção de dívidas anteriores no projeto.
	Diagrama de causa e efeito	Serve para organizar as causas que levaram à incidência de um DT, ajudando a identificá-lo rapidamente, levando em consideração que os dados estão previamente descritos.
	Ações preventivas	Estudos primários relatam que prevenir a ocorrência de um DT pode ser mais barato do que o seu pagamento. Ações preventivas apoiam profissionais e equipes de software na aplicação de práticas aceitáveis que minimizem sua ocorrência. As ações preventivas relacionadas aos requisitos de DT são: controlar e negociar os requisitos de software; requisitos bem definidos; boa comunicação entre as partes interessadas; declaração de escopo bem definida; rastreamento de mudanças de requisitos e compromisso com o cliente.
	Mapa de pagamento	Os profissionais de software podem consultar este mapa para orientar suas decisões sobre a eliminação de DT em seus projetos. Como guia, o mapa pode informar um conjunto de práticas em resposta à necessidade de pagamento de DT. Além disso, pode ser usado como um dispositivo de comunicação para apoiar as equipes a comunicar efetivamente os DT existentes aos gerentes e decidir melhor sobre o pagamento dos requisitos do DT, por exemplo.
	Identificação através da ISO/IEC/IEEE 29148:2018	Descreve a qualidade dos requisitos desde a especificação até outras atividades de engenharia de requisitos no ciclo de vida do desenvolvimento de software. Ele fornece detalhes para criar requisitos textuais consistentes, incluindo características e atributos e critérios de idioma dos requisitos. A partir do momento que esses padrões de qualidade são violados, podem surgir cheiros de requisitos e dívidas técnicas. Analisar este documento de acordo com os projetos especificados e os requisitos documentados ajuda a identificar inconsistências, baixos níveis de qualidade e violações e, conseqüentemente, identificar DT.
	Análise de código	Analisar código-fonte para identificar violações de regras de codificação, falta de testes; calcular métricas de software com base no código-fonte para identificar problemas de design ou arquitetura.
	Análise de dependência	Analisar dependências entre diferentes tipos de elementos de software (por exemplo, componentes, módulos).
	Lista de controle	Verifica uma lista de cenários predefinidos em que ocorre.

Mensuração	Calcular dívida técnica por meio de fórmulas matemáticas ou modelos ou métricas de código	Calcular a dívida por meio de fórmulas matemáticas ou modelos ou métricas de código.
	Estimar dívida técnica (requisitos)	Estimar dívida técnica de acordo com a experiência e especialização.
	Categorização de custo	Estimar vários tipos de custo de manuseio da dívida técnica incorrido.
	Indicar dívida técnica usando métricas de qualidade	Indicar dívida técnica usando métricas de qualidade da operação do produto.
	Comparação de soluções	Comparação de soluções por meio de cálculo da distância entre a solução real e a solução ótima.
Priorização	Análise de custo-benefício	Se a resolução de um item DT puder gerar um benefício maior do que o custo, esse item DT deve ser reembolsado. Itens de DT com maior relação custo/benefício de reembolso devem ser pagos primeiro.
	Alto custo de correção primeiro	Itens de DT que são mais caros para resolver devem ser reembolsados primeiro.
	Abordagem de portfólio	Considera itens de DT juntamente com outras novas funcionalidades e bugs como riscos e oportunidades de investimento (ou seja, ativos).
	Alto interesse primeiro	Itens de DT que incorrem em juros mais altos devem ser reembolsados primeiro.
Pagamento	Reestruturação de código e projeto	Fazer alterações no código, design ou arquitetura de um sistema de software sem alterar os comportamentos externos do sistema de software, a fim de melhorar a qualidade interna.
	Refatoração de Código	Reescrever o código que contém dívida técnica.
	Automação	Automatizar o trabalho repetido manualmente, por exemplo, testes manuais, construções manuais e implantação manual.
	Reengenharia	Evolua o software existente para exibir novos comportamentos, recursos e qualidade operacional.
	Otimização	Agrupar módulos coesos com dependências gerenciáveis para simplificar o código.
	Corrigir erro	Resolver bugs conhecidos.
	Tolerância ao erro	Coloque estrategicamente exceções de tempo de execução onde está a dívida.
Monitoramento	Abordagem baseada em limiar	Definir limites para métricas de qualidade relacionadas a DT e emitir avisos se os limites não forem atingidos.

	Rastrear a propagação de DT	Rastreie as influências de DT por meio de dependências entre outras partes de um sistema e as partes do sistema que contém DT.
	Conferir o planejado	Medir regularmente a DT identificado e rastrear a mudança da DT.
	Plote várias medidas agregadas da dívida de requisitos	Plote várias medidas agregadas da dívida de requisitos ao longo do tempo e observe a forma da curva para observar as tendências.
Comunicação	Painel com exibição dos itens da dívida de requisitos	Manter um painel com a exibição de itens, tipos e valores de DT para informar todas as partes interessadas sobre a existência da DT.
	Listar as pendências como itens de DT identificados	Todos os itens de DT identificados, bem como qualquer coisa a ser resolvida no desenvolvimento, são colocados no backlog do projeto de software, para que os itens de DT possam ser tratados tão importantes quanto bugs conhecidos e recursos e funcionalidades planejados não implementados.
	Visualização de dependência indesejáveis	Visualizar as dependências indesejáveis (por exemplo, dependências excessivamente complexas) entre elementos de software (por exemplo, componentes e pacotes).
	Visualização de métricas de código	Visualizar métricas de código em algumas ferramentas, como mapas de código, e destaque os elementos de software com má qualidade medida (por exemplo, complexidade do código).
	Visualização de propagação DT	Mostrar as conexões entre diferentes itens da DT e como um item do DT afeta e é afetado por outros itens do DT
Documentação	Identificador exclusivo	Identificador exclusivo para um item de dívida técnica.
	Localização	A localização do item da DT identificado.
	Responsável/autor	Pessoa responsável por reembolsar o item da DT.
	Tipo da dívida técnica	Tipo: O tipo de DT em que este item de TD é classificado, por exemplo, dívida de requisitos.
	Descrição	Informações gerais sobre o item DT.
	Data/hora	A data ou hora em que o item DT é identificado.
	Custo estimado	Custo estimado para reembolsar o item dívida técnica.
	Montante de juros	O custo extra estimado de tolerar o item DT.

	Probabilidade de juros	A probabilidade de que os juros deste item da DT precisem ser reembolsados.
	Desvio padrão de juros	A diferença estimada entre o valor estimado dos juros e o valor real (futuro) dos juros.
	Correlações com outros itens da dívida	Relacionamentos entre este item DT e outros itens DT.
	Contexto	Um determinado contexto de implementação (por exemplo, linguagem de programação usada) de um tipo específico de DT em um item de DT.
	Regras de propagação	Como este item de DT afeta as partes relacionadas do sistema de software.
	Informar se foi intencional ou não	Informar se a dívida contraída foi intencional ou não.
Prevenção	Melhorar os processos de desenvolvimento atuais	Melhorar os processos de desenvolvimento atuais para prevenir a ocorrência de certos tipos de DT.
	Apoio à decisão de arquitetura, avaliando o potencial DT	Apoio à decisão de arquitetura, avaliando o potencial DT causado por diferentes opções de projeto de arquitetura e, em seguida, escolha a opção com menos potencial DT.
	Planejamento de custos do ciclo de vida	Desenvolver planos econômicos que observem o sistema durante todo o ciclo de vida para minimizar o DT geral do sistema.
	Analisar os fatores humanos, mantendo uma cultura que minimize o DT não intencional	Cultivar uma cultura que minimize o DT não intencional causado por fatores humanos, por exemplo, indiferença e ignorância.

Fonte: Elaborado pelo autor (2023)

QP 2. Quais as práticas sugeridas para as atividades ou áreas da gestão da dívida de requisitos? (Área, práticas gerais e suas práticas comuns/específicas)

(Descrição das práticas da gestão da dívida técnica do estudo terciário)

Áreas/Fases Gestão da Dívida Técnica	Práticas gerais	Práticas direcionadas para as práticas gerais	Estudos
Identificação	Avaliação do cliente	Revisão dos Requisitos	SD1
	Comunicação face a face	Comunicação por e-mail	SD1
		Comunicação por documentos	
		Comunicação transparente com os stakeholders	SD5
	Modelo de documentação	Modelo de documentação	SD1
	Revisão por pares	Registrar os diálogos com clientes	SD1
	Análise simples de custo-benefício	Criar lista de itens da dívida de requisitos	SD1
	Abordagem de quantificação	Informar consequência de manter a dívida para manutenção futuro	SD1
		Quantificar o juro da dívida	
		Quantificar o tamanho da dívida	
		Status do crescimento da dívida	
	Abordagem do vizinho mais próximo	Aproveitar experiências das dívidas resolvidas	SD1
	Diagrama de causa e efeito	Identificar as causas que levaram à incidência de um DT	SD1
	Ações preventivas	Compromisso com o cliente	SD1
		Controlar e negociar requisitos	SD1
		Declaração do escopo bem definido	SD1
		Manter a rastreabilidade dos requisitos	SD1
		Requisitos bem definidos	SD1

		Comunicação com as partes interessadas (stakeholders) em todas as fases da Engenharia de Requisitos	SD1
	Mapa de pagamento	Manter mapa de pagamento para orientar decisões sobre a eliminação da dívida	SD1
		Mapa como um guia para manter um conjunto de práticas conforme a necessidade de pagamento da dívida	
	Identificação através da ISO/IEC/IEEE 29148:2018	Confrontar com a norma os projetos especificados e os requisitos documentados	SD1
	Análise de código	Calcular métricas de software para identificar problemas de design e arquitetura	SD7, SD5
	Análise de dependência	Analisar as dependências entre os elementos do software	SD7
	Lista de controle	Verificar a lista de cenários em que a dívida ocorre	SD7
Mensuração	Calcular a dívida técnica	Calcular a dívida por meio de fórmula matemática ou modelos	SD7
		Calcular a dívida por meio de métricas do código fonte	
	Estimar dívida técnica de acordo com a experiência	Realizar estimativa de custo item da dívida de requisitos	SD7
	Categorização de custo	Estimar os tipos de custo de manuseio da dívida	SD7
	Indicar dívida técnica por meio de métricas	Indicar a dívida técnica usando métricas de qualidade da operação de produto	SD7
	Comparação de soluções	Cálculo entre a solução real e a solução ótima	SD7, SD6
Priorização	Análise de custo-benefício	Dívidas com maior relação custo-benefício de reembolso possuem prioridade	SD7, SD6
	Alto custo de correção primeiro	Itens de dívidas mais caras devem ser reembolsadas primeiro	SD7
	Abordagem de portfólio	Considerar os itens das dívidas como funcionalidades com bugs ativos	SD7, SD6
	Alto interesse primeiro	Priorizar dívidas com altos juros	SD7, SD6, SD5

Pagamento	Reestruturação de código e projeto	Realizar melhorias no código, design ou arquitetura	SD7
	Refatoração de Código	Refatorar o código que contém a dívida técnica	SD7, SD5
	Automação	Automatizar atividades manuais como testes e implantação	SD7, SD5
	Reengenharia	Evoluir o software	SD7
	Otimizar código	Otimizar o código simplificando-o	SD7
	Corrigir erro	Corrigir bugs	SD7
	Tolerância ao erro	Inserir estratégias de exceção de tempo e execução local da dívida	SD7
Monitoramento	Abordagem baseada em limiar	Definir limites para métricas de qualidade	SD7
		Emitir notificações de limites não alcançados	
	Rastrear a propagação da dívida de requisitos	Monitorar a influência da dívida de requisitos nas partes do sistema que não contém a dívida	SD7, SD5
		Monitorar a propagação da dívida de requisitos por meio de atributos de qualidade	SD7
	Conferir o planejado	Medir regularmente da dívida de dívida	SD7, SD5
		Rastrear mudança na dívida de requisitos	SD7
	Plotar várias medidas agregadas da dívida de requisitos	Plotar medidas de dívidas de requisitos para observar a curva para identificar tendências	SD7
Comunicação	Painel com exibição dos itens da dívida de requisitos	Manter itens da dívida de requisitos visíveis	SD7
	Listar as pendências como itens de dívidas identificados	Registrar no backlog itens da dívida de requisitos	SD7
	Visualização de dependência indesejáveis	Permitir visualizar pendências indesejáveis e complexas	SD7
	Visualização de métricas de código	Utilizar ferramentas para visualizar métricas de código	SD7
	Visualização de propagação TD	Disponibilizar a visualização da propagação da dívida de requisitos	SD7, SD5
		Mostrar conexão de um item da dívida de requisitos com outros itens da dívida	SD7

Documentação	Identificador exclusivo para um item de dívida	Apontar identificador exclusivo cada item da dívida de requisitos	SD7
	Localização do item da dívida	Registrar a localização do item da dívida de requisitos	SD7
	Responsável/autor	Informar o responsável pelo pagamento da dívida de requisitos	SD7
	Tipo da dívida técnica	Registrar para cada item da dívida o tipo de dívida técnica (por exemplo: Dívida de Requisitos)	SD7
	Descrição do item da dívida	Registrar informações gerais sobre o item da dívida de requisitos	SD7
	Data e hora da identificação da dívida	Informar data e hora da identificação da dívida de requisitos	SD7
	Custo estimado para reembolsar o item dívida	Registrar estimativa de custo do item da dívida de requisitos	SD7
	Montante de juros	Montante de juros extra da dívida de requisitos	SD7
	Probabilidade de juros	Informar a probabilidade que os juros do item da dívida de requisitos devem ser reembolsados	SD7
	Desvio padrão de juros	Informar a diferença entre o valor estimado de juros e o valor real dos juros	SD7
	Correlações com outros itens da dívida	Mostrar conexão de um item da dívida de requisitos com outros itens da dívida	SD7
	Contexto da dívida	Definir o contexto da dívida	SD7
	Regras de propagação	Estabelecer regras de propagação da dívida de requisitos	SD7
	Informar se foi intencional ou não	Informar se a dívida de requisitos foi intencional ou não intencional	SD7
Prevenção	Melhorar os processos de desenvolvimento atuais para prevenir a ocorrência de certos tipos de dívidas técnicas	Melhorar o processo de desenvolvimento de software	SD7
	Apoio à decisão de arquitetura, avaliando o potencial da dívida	Avaliar o potencial da dívida e escolher a dívida de menor potencial	SD7
	Planejamento de custos do ciclo de vida	Desenvolver planos econômicos para minimizar a dívida de requisitos	SD7
	Analisar os fatores humanos, mantendo uma cultura que minimize a dívida	Manter a cultura de práticas que minimize a dívida de requisitos	SD7

Fonte: Elaborado pelo autor (2023)

QP 3. Quais as limitações enfrentadas na implementação das atividades ou áreas da gestão da dívida de requisitos?

Limitações	Estudos
Gerenciar DT com eficiência	SD1
Falta de acesso a ferramentas	SD1
Entendendo que DT é um problema	SD1
Medir DT	SD1, SD7
Alocar mais tempo na elicitação de requisitos	SD1
Envolver a equipe no processo de gerenciamento de DT	SD1
Equilibrar os benefícios de pagar a DT com os custos desse processo	SD1
Gerenciar DT intencional e não intencional	SD1, SD7
Moral da equipe	SD1
Adaptar a equipe ao processo de gestão de DT	SD1
Gerenciamento automatizado	SD1
Colaboração do cliente neste processo	SD1
Cultura ou sentimentos pessoais	SD1
Gerenciar DTs mais antigos	SD1
Objetivos conflitantes	SD1
Restrições organizacionais	SD1
Tradição	SD1
Desafios na gestão de DT induzidos e não intencionais	SD7
Dificuldades na transformação do valor empresarial e econômico	SD7
Desafios na medição de DT	SD7
Desafios na prevenção de DT	SD7
Falta de uma teoria e modelos subjacentes para ajudar na identificação e medição de DT	SD7

Elaborado pelo autor (2023)

QP 4. Quais as ferramentas de apoio para gestão de dívida de requisitos?

Ferramentas	Descrição	Estudos
DebtFlag	Captura, rastreia e resolve TD no desenvolvimento de software. Composta principalmente por um plug-in do Eclipse usado para capturar, monitorar e gerenciar DT durante o desenvolvimento	SD5, SD7
Kit de Ferramentas de Análise de Software SIG	Calcula as propriedades do código	SD7
Design TD: codes smells; iPlasma;	Calcular métricas de código	SD7
Código: Métricas do Eclipse	Calcular métricas de código	SD7
Rational AppScan	Identificar falhas de segurança no código-fonte	SD7
PMD	Procurar problemas potenciais no código-fonte OSS	SD7
PHPMD	Detecta confusão (por exemplo, expressões supercomplicadas) no código PHP	SD7
NDepend	Calcula métricas de código .NET	SD7
NCapa	Analisa a cobertura de código para .NET	SD7
FxCop	Analisa o código gerenciado para identificar problemas de conformidade com as diretrizes de programação .NET	SD7
CodeXpert	Automatiza revisões de padrões e qualidade de código	SD7
PL/SQL	Cobertura Analisando a cobertura de código para Java OSS	SD7
Estilo de verificação	Verifica o código Java em relação aos padrões de codificação	SD7
Ferramenta de mapas de software	Ferramenta de mapas de software: Visualizando a qualidade do código dos arquivos de código-fonte	SD7
RE-KOMBINE (REQUISITOS)	Identifica e mede dívida de requisitos	SD7
Código Árvores de Natal	Visualiza a complexidade e cobertura do código	SD7
Plataforma de Inteligência de Aplicações de Software da CAST	Identifica violações no código-fonte e categorizar as violações por atributos de qualidade;	SD7
Plug-in de Avaliação de Dívida Técnica (SQALE) para SonarQube	Analisa, mede, visualiza e prioriza DT com base no modelo de qualidade SQALE;	SD7
STAN	Calcula as métricas de qualidade da estrutura de sistemas Java	SD7
Métricas Padrão de Recursos	Calcula métricas de código-fonte e analisando a qualidade do código para encontrar violações de estilo e problemas lógicos;	SD7
Verificador de conformidade RBML	Calcula a distância entre a realização de um padrão de projeto e o projeto pretendido	SD7

Uma ferramenta para identificar dependências ruins	Identifica dependências intra e inter módulos ruins	SD7
Plug-in Sonar DT	Identifica e mede DT na forma de baixa cobertura de código	SD7
SonarQubeGenericName	Plataforma aberta para gerenciar a qualidade do código	SD7
Plug-in SonarQube COBOL	Realiza revisões de código COBOL objetivas e automatizadas em relação às melhores práticas de codificação	SD7
CLIO	Identifica violações de modularidade	SD7
CodeVizardName	Identifica cheiros de código	SD7
FindBugs	Identifica problemas de análise estática automática	SD7

Fonte: Elaborado pelo autor (2023)

**APÊNDICE R – RESULTADO DA UNIFICAÇÃO DAS ÁREAS DA DÍVIDA TÉCNICA, PRÁTICAS GERAIS E PRÁTICAS
ESPECÍFICAS E COMUNS – ESTUDO TERCIÁRIO – MSL – ESTADO DA PRÁTICA**

Áreas	Prática Geral	Práticas Comuns e Específicas
Identificação	Avaliação do cliente	Revisão de Requisitos
	Comunicação face a face	Comunicação transparente com as partes interessadas (stakeholders) em todas as fases da Engenharia de Requisitos por meio de e-mail, documentos
		Realizar cerimônias (iniciais, sprint planning, pós refinamento, reuniões diárias)
	Modelo de documentação	Modelo de documentação
	Revisão por pares	Registrar os diálogos com clientes
	Análise simples de custo-benefício	Criar lista de itens da dívida de requisitos
	Abordagem de quantificação	Informar consequência de manter a dívida para manutenção futuro
		Quantificar o juro da dívida
		Quantificar o tamanho da dívida
		Status do crescimento da dívida
	Abordagem do vizinho mais próximo	Aproveitar experiências das dívidas resolvidas
	Diagrama de causa e efeito	Identificar as causas que levaram à incidência de uma dívida técnica
	Ações preventivas	Compromisso com o cliente
		Controlar e negociar requisitos
		Declaração do escopo bem definido
		Manter a rastreabilidade dos requisitos
		Requisitos bem definidos registrados no backlog por meio da existência de regra de negócio como parte da história de usuário baseada nos requisitos do cliente, e com clareza na declaração do escopo. EXEMPLO IMPLEMENTA

		Comunicação transparente com as partes interessadas (stakeholders) em todas as fases da Engenharia de Requisitos por meio de e-mail, documentos
	Mapa de pagamento	Manter mapa de pagamento para orientar decisões sobre a eliminação da dívida
		Mapa como um guia para manter um conjunto de práticas conforme a necessidade de pagamento da dívida
	Identificação através da ISO/IEC/IEEE 29148:2018	Confrontar com a norma os projetos especificados e os requisitos documentados
	Análise de código	Calcular métricas de software para identificar problemas de design e arquitetura
	Análise de dependência	Analisar as dependências entre os elementos do software
	Lista de controle	Verificar a lista de cenários em que a dívida ocorre
Mensuração	Calcular a dívida técnica	Calcular a dívida por meio de fórmula matemática ou modelos
		Calcular a dívida por meio de métricas do código fonte
	Estimar dívida técnica de acordo com a experiência	Realizar estimativa de custo item da dívida de requisitos
	Categorização de custo	Estimar os tipos de custo de manuseio da dívida
	Indicar dívida técnica por meio de métricas	Indicar a dívida técnica usando métricas de qualidade da operação de produto
	Comparação de soluções	Cálculo entre a solução real e a solução ótima
Priorização	Análise de custo-benefício	Dívidas com maior relação custo-benefício de reembolso possuem prioridade
	Priorizar correção de alto custo	Itens de dívidas mais caras devem ser reembolsadas primeiro
	Abordagem de portfólio	Considerar os itens das dívidas como funcionalidades com bugs ativos
	Priorizar alto interesse	Priorizar dívidas com altos juros
Pagamento	Reestruturação de código e projeto	Realizar melhorias no código, design ou arquitetura
	Refatoração de Código	Refatorar o código que contém a dívida técnica
	Automação	Automatizar atividades manuais como testes e implantação
	Reengenharia	Evoluir o software

	Otimizar código	Otimizar o código simplificando-o
	Corrigir erro	Corrigir bugs
	Tolerância ao erro	Inserir estratégias de exceção de tempo e execução local da dívida
Monitoramento	Abordagem baseada em limiar	Definir limites para métricas de qualidade
		Emitir notificações de limites não alcançados
	Rastrear a propagação da dívida de requisitos	Monitorar a influência da dívida de requisitos nas partes do sistema que não contém a dívida
		Monitorar a propagação da dívida de requisitos por meio de atributos de qualidade
		Monitorar o progresso dos requisitos, por meio do gerenciamento dos requisitos, estabelecendo uma conexão entre os itens pendentes da dívida com os requisitos
	Conferir o planejado	Medir regularmente a dívida
		Rastrear mudança na dívida de requisitos
	Plotar várias medidas agregadas da dívida de requisitos	Plotar medidas de dívidas de requisitos para observar a curva para identificar tendências
Comunicação	Painel com itens da dívida de requisitos	Manter itens da dívida de requisitos visíveis
	Listar as pendências como itens de dívidas identificados	Registrar no backlog itens da dívida de requisitos
	Visualização de dependência indesejáveis	Permitir visualizar pendências indesejáveis e complexas
	Visualização de métricas de código	Utilizar ferramentas para visualizar métricas de código
	Visualização de propagação da dívida técnica	Disponibilizar a visualização da propagação da dívida de requisitos
		Mostrar conexão de um item da dívida de requisitos com outros itens da dívida
Documentação	Identificador exclusivo para um item de dívida	Apontar identificador exclusivo cada item da dívida de requisitos

	Localização do item da dívida	Registrar a localização do item da dívida de requisitos
	Responsável/autor	Informar o responsável pelo pagamento da dívida de requisitos
	Tipo da dívida técnica	Registrar para cada item da dívida o tipo de dívida técnica
	Descrição do item da dívida	Registrar informações gerais sobre o item da dívida de requisitos
	Data e hora da identificação da dívida	Informar data e hora da identificação da dívida de requisitos
	Custo estimado para reembolsar o item dívida	Registrar estimativa de custo do item da dívida de requisitos
	Montante de juros	Montante de juros extra da dívida de requisitos
	Probabilidade de juros	Informar a probabilidade que os juros do item da dívida de requisitos devem ser reembolsados
	Desvio padrão de juros	Informar a diferença entre o valor estimado de juros e o valor real dos juros
	Correlações com outros itens da dívida	Mostrar conexão de um item da dívida de requisitos com outros itens da dívida
	Contexto da dívida	Definir o contexto da dívida
	Regras de propagação	Estabelecer regras de propagação da dívida de requisitos
	Informar se foi intencional ou não	Informar se a dívida de requisitos foi intencional ou não intencional
Prevenção	Melhorar os processos de desenvolvimento atuais para prevenir a ocorrência de certos tipos de dívidas técnicas	Treinamento de padrões de elicitação de requisitos que enfatizem a forma como os requisitos devem ser elicitados e especificados
		Avaliar e melhorar a eficiência de suas técnicas de elicitação de requisitos
		Construção fluxo do processo para melhorar a compreensão da demanda/necessidade do cliente
		Requisitos bem definidos registrados no backlog por meio da existência de regra de negócio como parte da história de usuário baseada nos requisitos do cliente, e com clareza na declaração do escopo.
		Refinamento requisitos por meio da definição iterativa de requisitos (trabalham com sprints/releases e técnicas de UX)
		Elaborar artefatos da UML
		Técnica de BDD para descrever Use Case inicial - Escopo história – User Story

		Desenvolver protótipo como parte do documento de especificação de requisitos
		Manter e descrever com clareza o documento de especificação de requisitos
		Construção de protótipo para validação dos requisitos
		Revisão de Requisitos
		Manter a rastreabilidade dos requisitos
	Apoio à decisão de arquitetura, avaliando o potencial da dívida	Avaliar o potencial da dívida e escolher a dívida de menor potencial
	Planejamento de custos do ciclo de vida	Desenvolver planos econômicos para minimizar a dívida de requisitos
	Analisar os fatores humanos, mantendo uma cultura que minimize a dívida	Manter a cultura de práticas que minimize a dívida de requisitos

Fonte: Elaborado pelo autor (2023)

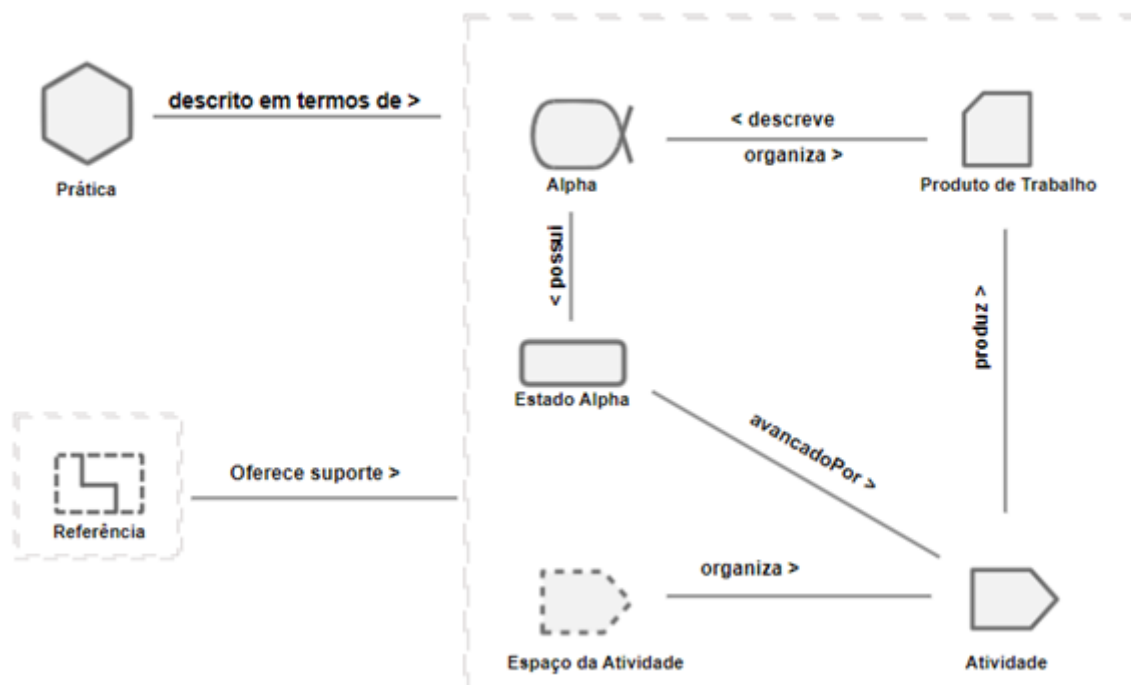
APÊNDICE S – GLOSSÁRIO DA ONTORED

Nome	Termo/Sigla	Descrição	Tipo
Ontologia voltada à dívida de requisitos	ONTOREd	Ontologia que representa o conhecimento voltado à dívida de requisitos	Conceito
Dívida de Requisitos	ReD	"refere-se à distância entre a especificação de requisitos desejada e a real implementação do sistema, sob suposições e restrições de domínio" (ERNST 2012).	Conceito
Fase da Engenharia de Requisitos	FaseER	Fases que compõem a Engenharia de Requisitos.	Conceito
Causa da dívida de requisitos	CausaReD	Fator eficaz que dá existência ao que não existia, ou seja, a origem das consequências das dívidas de requisitos.	Conceito
Consequência da dívida de requisitos	ConsequenciaReD	Problemas relacionados ao processo da Engenharia de Requisitos resultantes das causas, e que se manifestam no processo de desenvolvimento.	Conceito
Área da gestão da dívida de requisitos	AreaGestaoReD	A área da gestão da dívida consiste em fundamentar as atividades que apoiam especialmente o monitoramento, priorização, prevenção e pagamento da dívida. Áreas se estendem às dívidas de Requisitos (Albuquerque et al. 2023).	Conceito
Prática para mitigar a dívida	PraticaDívida	“Uma prática é uma abordagem repetível para fazer algo com um objetivo específico em mente. Uma prática descreve como lidar com um aspecto específico de um empreendimento de engenharia de software, incluindo as descrições de todos os elementos relevantes necessários para expressar a orientação de trabalho desejado para atingir o objetivo da prática. Uma prática pode ser definida como uma composição de outras práticas” (JACOBSON 2018).	Conceito
Nome	nome	Denominação de todos os elementos do domínio.	Atributo de instância
Descrição	descricao	Apresenta a descrição dos elementos da Ontologia	Atributo de instância
Classificação da Prática	classificacaoPratica	Define a classificação da prática da dívida: geral, comum genérica –(quando sugerida para a todas as causas), e específica (voltada à algumas causas).	Atributo de instância
Apresentação da prática	apresentacaoPratica	Apresenta uma visão geral sobre a Prática.	Atributo de instância
Produto de trabalho	produtoTrabalho	Artefato que compõe uma prática	Atributo de instância
Atividade	atividade	Ação que atende a elaboração ou parte de um artefato	Atributo de instância
Está associada a	estaAssociadaA	Categoriza a causa da dívida de requisitos e consequência, conforme a fase correspondente da Engenharia de Requisitos.	Relação
É evidenciada por	eEvidenciadaPor-	Consiste em evidenciar as causas e consequências associadas às fases da Engenharia de Requisitos.	Relação
Ocasiona	ocasiona	Motiva o acontecimento de algo, por exemplo: Uma causa ocasiona uma ou várias consequências.	Relação
Ocasionada por	ocasionadaPor	Representa a relação consequência -> causa.	Relação

Pode mitigar	podeMitigar	Representa que algo pode ser minimizado, por exemplo, causa que pode ser minimizada com alguma ação.	Relação
Mitigado por	mitigadoPor	Consiste em causas que podem ser mitigadas por determinadas práticas.	Relação
É constituída de	eConstituidaDe	Condiciona a colaboração de vários elementos para alcançar determinado resultado, por exemplo:	Relação
É Implementada Por	implementadaPor	Garante a realização de algo.	Relação

Fonte: Elaborado pelo autor (2023)

APÊNDICE T – ESTRUTURA DOS ELEMENTOS DA PRÁTICA



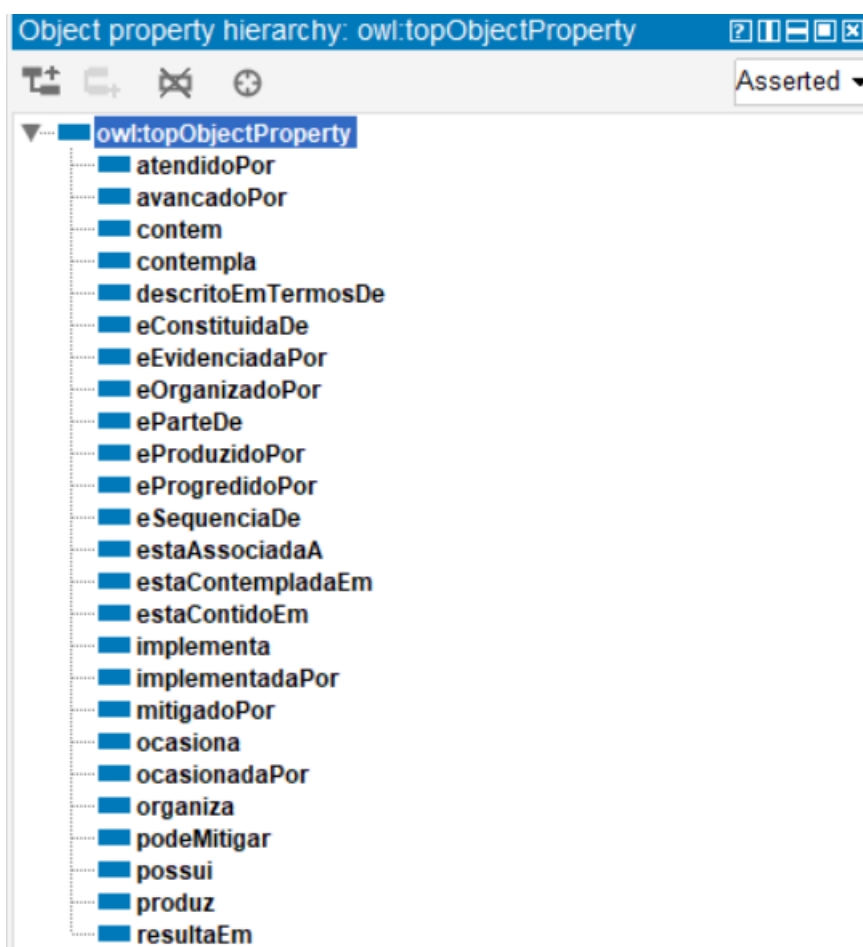
Fonte: Adaptado da Linguagem Essence (OMG 2018)

APÊNDICE U – CARACTERÍSTICAS DAS PROPRIEDADES DOS OBJETOS DEFINIDAS NA ONTORED

Característica - Object Property	Descrição
Functional	Se uma propriedade é definida como funcional para um determinado indivíduo “A”, então, pode existir, até, no máximo, um indivíduo “B” relacionado ao indivíduo “A” via essa propriedade.
Transitive	Uma propriedade P é transitiva quando a propriedade relaciona o indivíduo "A" ao indivíduo "B" e, também um indivíduo "B" ao indivíduo "C", então, infere-se que o indivíduo "A" está relacionado ao indivíduo "C", por meio da propriedade P.
Asymmetric	A propriedade P é definida como assimétrica e relaciona o indivíduo “A” ao indivíduo “B”, porém, o indivíduo “B” não pode ser relacionado ao indivíduo “A” por meio de uma propriedade individual P.
Reflexive	Uma propriedade P é caracterizada como reflexiva quando ela relaciona o indivíduo “A” a ele mesmo.

Fonte: (HORRIDGE; BRANDT 2011)

APÊNDICE V – OBJETO PROPERTIES DA ONTOREDO



Fonte: Capturada do Protégé – OntoReD (2023)

APÊNDICE X – DOMAIN, OBJETO, PROPRIEDADES, RANGE

Domain	Propriedade	Range
EstadoAlpha	avancadoPor	AtividadeProdutoTrabalho
AtividadeProdutoTrabalho	resultaEm	EstadoAlpha
FaseEngenhariaRequisitos	eEvidenciadaPor	ConsequenciaReD
FaseEngenhariaRequisitos	eEvidenciadaPor	CausaReD
ConsequenciaReD	estaAssociadaA	FaseEngenhariaRequisitos
CausaReD	estaAssociadaA	FaseEngenhariaRequisitos
ProdutoTrabalhoPratica	eOrganizadoPor	Alpha
Alpha	organiza	ProdutoTrabalhoPratica
ProdutoTrabalhoPratica	eProduzidoPor	AtividadeProdutoTrabalho
AtividadeProdutoTrabalho	produz	ProdutoTrabalhoPratica
AtividadeProdutoTrabalho	estaContempladaEm	EspacoAtividade
EspacoAtividade	contempla	AtividadeProdutoTrabalho
PraticaDivida	implementa	PraticaDivida
PraticaDivida	implementadaPor	PraticaDivida
CausaReD	mitigadoPor	PraticaDivida
PraticaDivida	podeMitigar	CausaReD
CausaReD	ocasiona	ConsequenciaReD
ConsequenciaReD	ocasionadaPor	CausaReD
Alpha	possui	EstadoAlpha
EstadoAlpha	eSequenciaDe	Alpha
PraticaDivida	descritoEmTermosDe	Alpha
PraticaDivida	descritoEmTermosDe	EstadoAlpha
PraticaDivida	descritoEmTermosDe	AtividadeProdutoTrabalho
PraticaDivida	descritoEmTermosDe	EspacoAtividade
PraticaDivida	descritoEmTermosDe	ProdutoTrabalhoPratica
AreaGestaoReD	eConstituidaDe	PraticaDivida
PraticaDivida	eParteDe	AreaGestaoReD
Alpha	progredidoPor	AtividadeProdutoTrabalho
PassosElementosPratica	atendidoPor	AtividadeProdutoTrabalho
ProdutoTrabalhoPratica	contem	PassosElementosPratica
PassosElementosPratica	estaContidoEm	ProdutoTrabalhoPratica

Fonte: Elaborado pelo autor (2023)

APÊNDICE Z - MATRIZ DE GUT ADAPTADA

		Grau de Relevância		
Frequência	Frequentemente	Média	Alta	Alta
	Parcialmente	Baixa	Média	Alta
	Eventualmente	Baixa	Baixa	Média
		Leve	Moderado	Crítico
		Impacto		

Fonte: Elaborada pelo autor (2023)