

**UNIVERSIDADE FEDERAL DE SÃO CARLOS**  
**CENTRO DE CIÊNCIAS EXATAS E TECNOLOGIA**  
**PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA DE PRODUÇÃO**

**ANA PAULA DA SILVA PORTO**

**ENVOLVIMENTO DO USUÁRIO NA ELICITAÇÃO DE REQUISITOS EM  
PROJETOS ÁGEIS DE DESENVOLVIMENTO DE *SOFTWARES* EMPRESARIAIS  
SOB DEMANDA**

**SÃO CARLOS-SP**

**2021**

ANA PAULA DA SILVA PORTO

**ENVOLVIMENTO DO USUÁRIO NA ELICITAÇÃO DE REQUISITOS EM  
PROJETOS ÁGEIS DE DESENVOLVIMENTO DE *SOFTWARES* EMPRESARIAIS  
SOB DEMANDA**

Projeto de pesquisa apresentado ao Programa de Pós-Graduação em Engenharia de Produção da Universidade Federal de São Carlos (UFSCar) como requisito para o desenvolvimento da Dissertação.

**Orientador:** Prof. Dr. Sérgio Luis da Silva

**SÃO CARLOS-SP**

**2021**



# UNIVERSIDADE FEDERAL DE SÃO CARLOS

Centro de Ciências Exatas e de Tecnologia  
Programa de Pós-Graduação em Engenharia de Produção

---

## Folha de Aprovação

---

Defesa de Dissertação de Mestrado da candidata Ana Paula da Silva Porto, realizada em 31/08/2021.

### Comissão Julgadora:

Prof. Dr. Sergio Luis da Silva (UFSCar)

Profa. Dra. Fabiane Leticia Lizarelli (UFSCar)

Profa. Dra. Silvia Inês Dallavalle de Pádua (USP)

Profa. Dra. Camila de Araujo (UFU)

O presente trabalho foi realizado com apoio da Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) - Código de Financiamento 001.

O Relatório de Defesa assinado pelos membros da Comissão Julgadora encontra-se arquivado junto ao Programa de Pós-Graduação em Engenharia de Produção.

## RESUMO

Apesar da literatura apresentar um interesse crescente na utilização de metodologias ágeis como estratégia para minimizar problemas em relação às expectativas do cliente não atendidas, pouco ainda se sabe sobre como a engenharia de requisitos contribui para este propósito. Nesse sentido, o objetivo desta pesquisa é compreender como ocorre o envolvimento do usuário final no gerenciamento ágil do desenvolvimento de *softwares* empresariais sob demanda de modo a atender as expectativas dos clientes. E, para isso, lança o olhar para a atividade de elicitação de requisitos, que está relacionada ao esforço planejado em busca das reais necessidades e anseios para um projeto, sendo mais proativo que a simples coleta de informações. Por meio de revisões sistemáticas da literatura, este estudo analisou 25 estudos relevantes cuja síntese dos dados apontou que os teóricos dessas áreas propõem definições para visão do produto e métodos para concebê-la, no entanto, as definições se mostram parciais e possuem limitações como, por exemplo, alto nível de abstração o que dificulta a operacionalização do conceito. Já os métodos encontrados não são capazes de sistematizar o envolvimento do usuário no processo. Visando o alinhamento entre a prática e o objetivo da pesquisa, foi realizado um estudo de caso que acompanhou um projeto de uma empresa desenvolvedora de *softwares* e seu respectivo cliente, observando criteriosamente a fase de elicitação de requisitos ágeis e sua relação com o aceite do produto por parte dos usuários. Após a consolidação das informações, o estudo reforçou que o sucesso na elicitação dos requisitos partindo da interação com os usuários do *software* pode ser alcançado se o processo de comunicação for realizado com o mínimo de ruído. O estudo resultou ainda na necessidade de elicitar os requisitos a partir da visão do usuário, traduzindo-os em uma linguagem próxima aos desenvolvedores evitando dificuldade na compressão. Evidenciou-se que a participação ativa do cliente, anulando o usuário, cria barreiras à fase de elicitação dos requisitos. Um achado relevante está relacionado ao uso das práticas híbridas no caso, mesmo quando a empresa acredita usar, puramente, a abordagem ágil. Outra percepção relevante deste estudo se dá acerca do papel crítico do *Product Owner* sendo uma figura externa à equipe de desenvolvimento, impactando na compreensão clara por parte da equipe de desenvolvimento de quais recursos são necessários no produto.

**Palavras-chave:** Levantamento de requisitos. Desenvolvimento ágil de *software*. Usuário final. Clientes.

## ABSTRACT

Although the literature shows a growing interest in the use of agile methodologies as a strategy to minimize problems in relation to unmet customer expectations, little is still known about how requirements engineering contributes to this purpose. In this sense, the goal of this research is to understand how the involvement of the end user in the agile management of on-demand enterprise software development occurs in order to meet customer expectations. To do so, it looks at the activity of requirements elicitation, which is related to the planned effort in search of the real needs and desires for a project, being more proactive than the simple collection of information. Through systematic reviews of the literature, this study analyzed 25 relevant studies whose synthesis of the data indicated that the theorists of these areas propose definitions for product vision and methods to conceive it, however, the definitions are partial and have limitations such as, for example, high level of abstraction that makes it difficult to operationalize the concept. The methods found are not able to systematize the user's involvement in the process. In order to align the practice and the research objective, a case study was conducted that followed a project of a software development company and its respective customer, carefully observing the agile requirements elicitation phase and its relationship with the acceptance of the product by the users. After consolidating the information, the study reinforced that the success in eliciting requirements from the interaction with software users can be achieved if the communication process is carried out with a minimum of noise. The study also resulted in the need to elicit the requirements from the user's point of view, translating them into a language close to the developers, avoiding difficulties in compression. It became evident that the active participation of the customer, overriding the user, creates barriers to the requirements elicitation phase. A relevant finding is related to the use of hybrid practices in the case, even when the company believes to use, purely, the agile approach. Another relevant insight of this study is the critical role of the Product Owner being an external figure to the development team, impacting the development team's clear understanding of what features are needed in the product.

**Keywords:** Requirements elicitation. Agile software development. End user. Customers.

## LISTA DE SIGLAS

EAR – Engenharia de Requisitos Ágeis

ER – Engenharia de Requisitos

GAP – Gerenciamento Ágil de Projeto

GREM - *Gamified Requirements Engineering Model*

JAD – *Joint Policação Development*

NFR – Requisitos Não Funcionais

QA – Analista de Qualidade

RSD – (Especificação de Requisitos para Desenvolvedores

RSL – Revisão Sistemática da Literatura

UCD – *Design* Centrado no Usuário

## SUMÁRIO

<b>1 INTRODUÇÃO</b> .....	09
1.1 CONTEXTO E MOTIVAÇÕES .....	09
1.2 DELIMITAÇÃO DO PROBLEMA .....	12
1.3 OBJETIVO DE PESQUISA .....	13
<b>2 REFERENCIAL TEÓRICO PRELIMINAR</b> .....	14
2.1 GERENCIAMENTO DE PROJETOS ÁGIL .....	14
2.2 ENGENHARIA DE REQUISITOS .....	17
2.2.1 Elicitação de requisitos .....	19
2.2.2 Elicitação de requisitos na Engenharia de Requisitos ágil .....	20
<b>3 REVISÃO SISTEMÁTICA DA LITERATURA</b> .....	23
3.1 METODOLOGIA DA RSL .....	23
3.2 DEFINIÇÃO DO PROBLEMA, OBJETIVO E QUESTÃO DE PESQUISA .....	24
3.3 DEFINIÇÃO DAS FONTES PRIMÁRIAS E STRING DE PESQUISA .....	24
3.4 CRITÉRIOS DE INCLUSÃO .....	28
3.5 CRITÉRIOS DE QUALIFICAÇÃO .....	28
3.6 MÉTODOS E FERRAMENTAS .....	29
3.7 CONDUÇÃO DAS BUSCAS .....	30
3.8 RESULTADOS .....	31
3.9 CONCLUSÃO DA RSL .....	45
<b>4 MÉTODO DE PESQUISA</b> .....	47
4.1 PROPOSIÇÕES .....	48
4.2 PROTOCOLO DO ESTUDO DE CASO .....	49
4.2.1 Escolha do caso .....	49
<b>4.2.2 Escolha dos perfis entrevistados</b> .....	<b>50</b>
<b>4.2.3 Procedimento de coleta de dados</b> .....	<b>51</b>
<b>4.2.4 Procedimento de análise dos dados</b> .....	<b>52</b>
<b>4.2.3 Estrutura da entrevista</b> .....	<b>52</b>
<b>5 ESTUDO DE CASO</b> .....	<b>59</b>
5.1 A EMPRESA .....	59
5.1 ANÁLISE DE DOCUMENTOS .....	62
5.2 O CASO PARA OBSERVAÇÕES PARTICIPATIVAS E ENTREVISTAS .....	65
5.3 OBSERVAÇÕES PARTICIPATIVAS .....	66

<b>5.3.1 Observação das reuniões para elicitação de requisitos .....</b>	<b>66</b>
<b>5.3.2 Observação visita ao cliente para elicitação de requisitos .....</b>	<b>67</b>
<b>5.3.3 Observação das entrevistas para elicitação de requisitos .....</b>	<b>67</b>
<b>5.3.4 Observação nos grupos de discussão .....</b>	<b>68</b>
<b>5.4 ENTREVISTAS .....</b>	<b>70</b>
<b>5.4.1 Entrevistas – antes do início do projeto.....</b>	<b>71</b>
<b>5.4.2 Entrevistas – durante o desenvolvimento do projeto .....</b>	<b>73</b>
<b>5.4.3 Entrevistas – fase de homologação.....</b>	<b>76</b>
<b>6 RESULTADOS E DISCUSSÕES .....</b>	<b>81</b>
6.1 TÉCNICAS ELICITAÇÃO DE REQUISITOS E ENVOLVIMENTO DO USUÁRIO ...	81
6.2 A INDISPONIBILIDADE DO USUÁRIO E A AGILIDADE DO PROJETO.....	83
6.3 COMPREENSÃO DOS REQUISITOS ORIENTADOS AO USUÁRIO .....	83
6.4 PAPEL DO USUÁRIO ANULADO E/OU SUBESTIMADO PELO PAPEL DO CLIENTE NO PROCESSO DE ELICITAÇÃO DE REQUISITOS .....	84
<b>7 CONCLUSÕES, LIMITAÇÕES E RECOMENDAÇÕES .....</b>	<b>86</b>
7.1 CONTRIBUIÇÕES DO ESTUDO.....	86
7.2 LIMITAÇÕES DO ESTUDOS .....	88
7.3 SUGESTÕES PARA ESTUDOS FUTUROS.....	89
<b>REFERÊNCIAS .....</b>	<b>91</b>
<b>APÊNDICE A .....</b>	<b>96</b>
<b>APÊNDICE B.....</b>	<b>105</b>
<b>APÊNDICE C .....</b>	<b>106</b>

## 1. INTRODUÇÃO

Este Capítulo tem por finalidade sintetizar brevemente a evolução do gerenciamento de projetos, da teoria de gerenciamento ágil e suas relações com a fase de engenharia de requisitos. O texto aborda ainda a contextualização da pesquisa explicitando as motivações e justificativas para o estudo. Após a construção desse panorama, traz o problema de pesquisa, assim como os objetivos do estudo.

### 1.1 CONTEXTO E MOTIVAÇÕES

Foi na década de 1990 que os desafios encontrados na aplicação de métodos tradicionais para o gerenciamento de projetos em produtos de inovação, especialmente na área de desenvolvimento de *software*, se tornaram temas de pesquisas que envolveram muitos profissionais e pesquisadores (PERMINOVA et al., 2008). Um marco importante neste período foi a reunião de um conjunto de teóricos e práticos da área de desenvolvimento de *software* que culminou na criação, em 2001, de um documento nomeado como “Manifesto para Desenvolvimento Ágil de *Software*” (BECK et al., 2001). Nesta reunião, foram discutidos os desafios encontrados nas abordagens tradicionais e levaram estes estudiosos à busca por soluções que resultaram no desenvolvimento de abordagens alternativas como teorias com princípios, técnicas e ferramentas, posteriormente intituladas como Gerenciamento Ágil de Projetos – GAP (AMARAL et al., 2011).

No entanto, mesmo considerando as evoluções nas abordagens ágeis, vários desafios são constantemente apontados na literatura e, muitos deles, atribuídos à elicitación de requisitos como, por exemplo, assertividade na estimativa dos custos e assertividade na entrega do produto desenvolvido que, por vezes, não correspondem às reais necessidades dos usuários (LARMAN, 2007; SAEEDA et al. 2020).

Aqui vale apresentar a definição do termo “elicitar” que se distingue de “coletar”, onde, segundo Ferreira (2014), o último termo refere-se ao “ato de colher, recolher, arrecadar”, enquanto o primeiro refere-se ao “extrair, tirar de”. Logo, coleta é como pegar conchas na praia, você pega somente o que você consegue ver e é reativa, não proativa. Já a elicitación é como na arqueologia onde se planeja uma busca em um local de propósito, como que um ato de provocar uma resposta ou reação em algo ou alguém, sendo muito mais proativo e menos reativo. Neste sentido, é importante salientar que esta pesquisa busca compreender fenômenos envolvidos

com a elicitação de requisitos que, como explicado, se difere de outros termos comumente utilizados.

A fase responsável pela elicitação desses requisitos recebe o nome de análise ou engenharia de requisitos. Neste documento, o termo utilizado será “engenharia de requisitos” ou ER, como é comum na literatura. De acordo com Fowler (2005), a engenharia de requisitos é uma etapa primordial para a concepção de um novo sistema e faz parte da maioria dos modelos de processo de desenvolvimento de *software*, contudo, com enfoques diferenciados em cada um.

De acordo com Paetsch (2003), em se tratando de modelo tradicional e ágil, a principal distinção entre a Engenharia Ágil de Requisitos (EAR) e a ER tradicional é que a primeira acolhe requisitos de forma iterativa e que mudam rapidamente, mesmo tardiamente no processo de desenvolvimento de *software*, e a segunda reúne e especifica todos os requisitos antes do desenvolvimento de *software*.

Assim, as soluções para muitos problemas relacionados à engenharia de requisitos advindos dos modelos tradicionais têm sido buscadas nos métodos ágeis de desenvolvimento de *software* uma vez que estes foram desenvolvidos para se adaptar e prosperar na mudança (SAVOLAINEN, 2010; PAETSCH, et al. 2003). Todavia, a fase de ER na abordagem ágil não possui características prescritivas, ou seja, uma definição formal sobre como as atividades devem ser executadas e isto faz com que haja uma variedade de formas de se definir como os requisitos serão levantados, priorizados, especificados e validados. Isto conduz para questionar como a ER tem sido realizada em projetos ágeis para *softwares* empresariais sob demanda de forma a driblar os desafios decorrentes das abordagens tradicionais e enfrentar os desafios atuais do GAP como, por exemplo, os elucidados pela pesquisa de Saeeda, et al. (2020), envolvendo questões como elicitação de requisitos pouco assertiva, levantamento e documentação incompleta de requisitos e baixo ou nenhum envolvimento do usuário final.

Há um consenso na literatura, conforme apresentado no capítulo de revisão, de que muitas técnicas da ER são consideradas boas práticas dentro dos processos de elicitação e modelagem de requisitos e aumentam as chances de sucesso do projeto. Em sintonia com os trabalhos relacionados, embora existam muitas validações empíricas de métodos de elicitação de requisitos, apenas alguns poucos trabalhos (e.g., MARTINS et al., 2019; AINHOA et al., 2019; SAEEDA et al., 2020) fazem análises relacionadas ao envolvimento do usuário como fator de influência na aceitabilidade do produto desenvolvido.

Esta dificuldade de envolver o usuário para o entendimento das reais necessidades de um projeto de *software* foi inicialmente observada na prática, antes mesmo da presente

pesquisa, por meio da experiência profissional prévia da mestranda, que motivou a realização da presente pesquisa e os levantamentos e apontamentos iniciais da literatura. Na análise exploratória da literatura, evidenciou-se que este tema e sua importância têm chamado a atenção de alguns autores sobre essa temática, como os estudos de Schon et al. (2017), Martins et al. (2019) e Saeeda et al. (2020). Através destas leituras foram encontrados elementos teóricos para estruturar um trabalho de forma empírica de modo a aprofundar os estudos visando descrever como se dá o envolvimento dos usuários, além do cliente, na fase de elicitação de requisitos no contexto ágil uma vez que poucos estudos estão diretamente relacionados ao tema.

Outra motivação está na lacuna encontrada na literatura referente à tratativa dos termos “cliente” e “usuário” que por vezes são tratados como sinônimos e, portanto, não buscam compreender o envolvimento de cada um destes papéis de forma separada. Neste documento, as nomenclaturas “cliente” e “usuário” não são tratadas como sinônimos e devem ser entendidas separadamente. De acordo com Pressman (2016), em alguns casos, cliente e usuário são a mesma pessoa, mas para muitos projetos são figuras distintas. Cliente é a pessoas ou grupo que (1) originalmente requisita o *software* a ser construído, (2) define os objetivos gerais para o negócio do *software*, (3) fornece os requisitos básicos para o projeto e (4) coordena os recursos financeiros para o projeto. Usuário é uma pessoa ou um grupo que vai realmente usar o *software* construído para atingir algum propósito de negócio.

Ainda sobre nomenclaturas, é importante ressaltar que os termos “participação” e “envolvimento” dos usuários também não serão tratados como sinônimos. Dahlsten (2004) define o envolvimento como um processo de socialização no qual o conhecimento tácito do cliente é obtido pela empresa. Em suma o processo de socialização afirmado pelo mesmo autor seria em tese regido pelos procedimentos e/ou métodos de envolvimento propostos para este fim enquanto, a obtenção e representação do conhecimento tácito do cliente seria a função principal dos requisitos levantados. Ou seja, os requisitos seriam “o como” externalizar este conhecimento tácito do cliente para que a equipe de projeto possa compreender e utilizá-lo.

Para O'Brien (2006), mais importante que a participação é o envolvimento do usuário final no desenvolvimento de novos sistemas de informação para a redução do potencial de resistência do mesmo. Essa concepção de envolvimento ajuda a garantir que os usuários assumam a autoria e que o produto atenda a todas as suas necessidades. Assim, seja qual for o grau de elegância técnica e de eficácia no processamento de dados de um determinado sistema, ele não será efetivo se frustrar ou incomodar os seus usuários. Logo, é fundamental para o avanço da teoria e das discussões relativas a esse tema, que o estado da arte da experiência e

envolvimento do usuário no GAP seja entendido e sintetizado, de modo que possa embasar os debates e reunir as principais dificuldades e contribuições.

Sendo assim, para o campo empresarial, esse estudo possibilita as organizações refletirem e guiarem suas iniciativas de envolvimento do usuário sobre uma base mais sólida e robusta. Desta forma, elas seriam capazes de diagnosticar a situação atual desta participação e envolvimento dos usuários em seus projetos e identificar oportunidades de melhorias.

## 1.2 DELIMITAÇÃO DO PROBLEMA

Apesar da elicitação detalhada dos requisitos ser muito importante em cada projeto, a metodologia ágil se concentra em menos documentação e processamento rápido, o que pode levar a ignorar os requisitos necessários do usuário (SATRIA; SHÖN, 2017). Este desencontro de informações causa falhas que deverão ser posteriormente corrigidas e este ciclo torna o processo como um todo menos ágil (SFETSOS, 2016). Outros estudos ainda, como Medeiros et al. (2020), indicam que os requisitos na abordagem ágil (considerando o forte envolvimento do usuário) são quase sempre insuficientes e inadequados para serem usados pela equipe de desenvolvimento, tornando o desenvolvimento mais difícil em um estágio posterior.

Para os autores Sfetsos et al. (2016) e Saeeda et al. (2020), existem meios para reduzir a discrepância entre o que é necessário para o usuário e o produto desenvolvido, porém, concluem afirmando que outros estudos empíricos na academia e na indústria (mais pontos de dados) poderiam fornecer evidências mais confiáveis, reforçando estas descobertas.

Em sintonia com os trabalhos relacionados, embora existam muitas validações empíricas de métodos de elicitação de requisitos, apenas alguns poucos trabalhos (e.g., MARTINS et al., 2019; AINHOA et al., 2019; SAEEDA et al., 2020) fazem análises relacionadas ao envolvimento do usuário como fator de influência na aceitabilidade do produto desenvolvido. Esta lacuna é ainda maior ao levar em consideração se os estudos têm o cuidado de distinguir claramente os papéis de cliente e usuário no contexto avaliado.

Isto implica na necessidade de evoluir nos estudos empíricos sobre: quais as práticas efetivas ou potenciais de elicitação de requisitos que são utilizadas no GAP que possibilitam o envolvimento do usuário sem prejudicar a agilidade no processo? Estas práticas contribuem para a aceitabilidade do *software* por parte dos usuários finais?

Vale elucidar que o foco desta pesquisa se concentra exclusivamente na construção de *softwares* empresariais sob demanda, ou seja, quando uma determinada empresa solicita soluções de forma personalizada e os *softwares* são desenvolvidos com foco em requisitos

específicos para determinado cenário, modelo de trabalho ou estratégia de negócios. Focar nesta área possibilita a identificação dos papéis clientes e usuários finais desde as fases iniciais dos projetos e, com isso, estudar sobre o envolvimento de ambos na fase de elicitação dos requisitos e suas implicações.

Na sequência, foram traçados os objetivos, geral e específicos, com a intenção de guiar o estudo com o propósito de gerar contribuições acerca desta problematização.

### 1.3 OBJETIVO DA PESQUISA

Este trabalho tem por objetivo geral compreender como sucede o envolvimento do usuário no processo de elicitação de requisito no gerenciamento ágil do desenvolvimento de *softwares* empresariais sob demanda. E, tem por objetivos específicos:

- a) Descrever, na prática, como ocorre o envolvimento dos usuários na EAR (Engenharia Ágil de Requisitos);
- b) Avaliar se o papel do usuário é subestimado e/ou anulado pelo envolvimento do papel do cliente no processo de elicitação de requisitos e;
- c) Descrever a aceitabilidade do *software* por parte dos usuários finais.

Para atingir tais objetivos, e considerando a viabilidade de acesso, a parte de campo empírica do presente trabalho será realizada com o acompanhamento de um projeto de uma empresa, analisando como as diferentes dinâmicas no processo de elicitação de requisitos podem impactar na aceitação do produto pelo usuário final.

## 2 REFERENCIAL TEÓRICO PRELIMINAR

Nesta seção, será apresentado o arcabouço teórico conceitual que fornecerá contextualização acerca dos assuntos relacionados ao tema desta pesquisa. Serão apresentados conceitos referentes ao gerenciamento ágil de projetos, engenharia de requisitos, engenharia de requisitos no contexto ágil e, dentre suas atividades, de forma mais detalhada, a fase de elicitação de requisitos.

### 2.1 GERENCIAMENTO ÁGIL DE PROJETOS

A teoria de Gerenciamento Ágil de Projeto (GAP) expandiu-se a partir da assinatura do *Manifesto for Agile Software Development*, em 2001 (BECK et al., 2001). Este manifesto aconteceu a partir da reunião de um conjunto de teóricos e práticos da área de desenvolvimento de *software* que resultou na criação do documento que agrupou princípios advindos de metodologias leves, como eram rotulados os processos “ágeis” precedente à assinatura do manifesto. Como exemplo para estes métodos, temos: *Extreme Programming* (XP); *SCRUM*; *Dynamic Systems Development Method* (DSDM); *Adaptive Software Development* (ASD); *Feature-Driven Development* (FDD), e outros, de acordo ao que se lê no Manifesto. A exploração pela simplificação dos processos de desenvolvimento é congruente entre eles, assim como também, o uso de poucos padrões, interação no desenvolvimento combinadas com entregas de curto prazo, tentativas de eliminar tarefas que não agregam valor ao produto e foco na participação das pessoas, inclusive clientes.

Desta forma, por ter origem em princípios semelhantes e objetivos comuns acerca de lidar com os mesmos desafios, os métodos de gestão são agrupados sob um mesmo rótulo, o gerenciamento ágil de projetos (HODA; NOBLE; MARSHALL, 2012), tendo a seguinte definição:

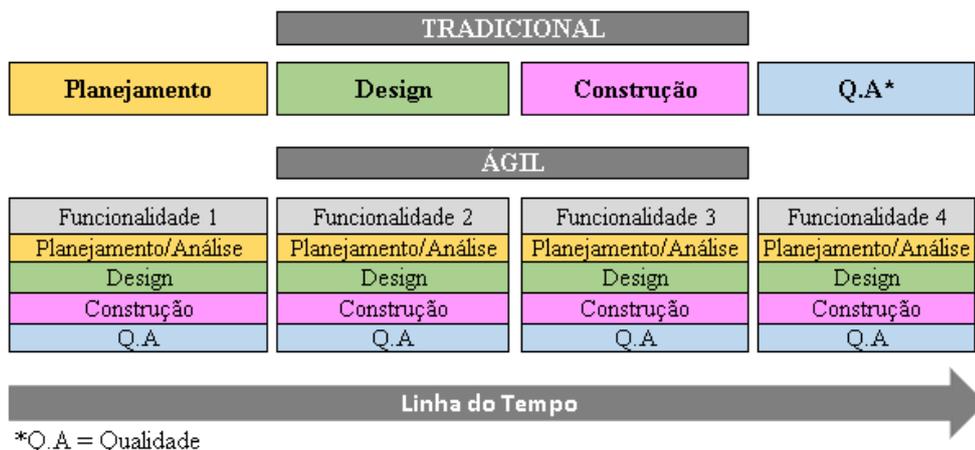
O gerenciamento ágil de projetos é uma abordagem fundamentada em um conjunto de princípios, cujo objetivo é tornar o processo de gerenciamento de projetos mais simples, flexível e iterativo, de forma a obter melhores resultados em desempenho (tempo, custo e qualidade), menos esforço em gerenciamento e maiores níveis de inovação e agregação de valor ao cliente (AMARAL et al. 2011).

Para melhor entendimento deste marco, é crucial buscar compreender as reais e principais distinções entre a teoria tradicional e ágil de gerenciamento de projetos que, segundo Amaral et al. (2011), são: autogestão, uso da visão no lugar do escopo, iteração e envolvimento do cliente e busca pela simplicidade.

Em se tratando de estrutura e fases, é muito comum a abordagem tradicional de gerenciamento de projetos fazer uso do modelo em cascata (em inglês “*waterfall*”), iniciado em meados da década de 70. Esta metodologia é caracterizada por fases que ocorrem dentro de uma sequência rígida, onde os conhecimentos necessários para a execução dos projetos podem ser antecipados e previamente adquiridos, com o início das atividades de uma etapa acontecendo imediatamente após o término daquela que a precedeu (AHERN; LEAVY; BYRNE, 2014). Já a abordagem ágil, segundo os mesmos autores, não se antecipa e propõe adquirir os conhecimentos a cada iteração por meio da experimentação no decorrer do projeto, onde os ciclos respondem a uma participação e interação contínua entre os participantes e fases do projeto, conforme ilustrado na Figura 1.

Muitos autores escreveram e escrevem relatos a respeito do gerenciamento ágil de projetos, todavia, dentre os principais autores, os que publicaram trabalhos mais robustos com definições para o termo “Gerenciamento Ágil de Projetos”, evidenciam-se Highsmith (2004) e Chin (2004). Para Highsmith (2004), o gerenciamento ágil de projetos pode ser entendido como: “[...] um conjunto de princípios, valores e práticas que auxiliam a equipe de projetos a entregar produtos ou serviços de valor em um ambiente de projetos desafiador”. Para Chin (2004) o gerenciamento ágil de projetos possui sua base na prática de uso de poucos padrões, equipes menores e interação no desenvolvimento. Baseado nestas definições e complementadas pela definição supracitada de Amaral et al. (2011), cria-se o pressuposto que a agilidade é alcançada, principalmente, à medida que a habilidade de se adaptar às mudanças é aplicada.

**Figura 1** – Ciclo de desenvolvimento de projetos tradicionais e ágeis



Fonte: Elaboração própria.

Embora existam muitas definições acerca do termo agilidade em projeto, como identificar o uso ou não da abordagem de gerenciamento ágil por uma organização? Para este

fim, o presente trabalho tomará por base a definição alcançada por Eder, S. et al. (2015) através do estudo que traz como conclusão a possibilidade de identificar o uso, ou não uso, da abordagem do gerenciamento ágil de projetos por meio da observação de seis características, conforme Quadro 1.

Embora os autores afirmem que tais diferenças devam ser tratadas ainda como uma hipótese a ser testada, esta forma de avaliação é válida uma vez que esta pesquisa a usará apenas para que, ao eleger o caso real para estudo, não seja necessário descrever totalmente as práticas realizadas pela empresa ou então confiar nos rótulos adotados pela empresa e/ou autodeclaração dos profissionais, enviesando a pesquisa.

**Quadro 1** - Características para identificação da abordagem de gerenciamento de projeto em uma organização.

<b>Característica</b>	<b>Abordagem de gerenciamento de projetos tradicional</b>	<b>Abordagem de gerenciamento ágil de projetos</b>
1) A forma de elaboração do plano do projeto	Há um único plano de projeto, que abrange o tempo total do projeto e contém os produtos, entregas, pacotes de trabalho e atividades.	Há dois planos de projeto: a) um plano geral que considera o tempo total de duração do projeto, mas que contém apenas os produtos principais do projeto; b) um plano de curto prazo (iteração) que é realizado sucessivas vezes e com um grau menor de detalhe por vez, buscando sempre as entregas mais importantes segundo o cliente/mercado por meio de iteração
2) A forma como se descreve o escopo do projeto	Descrição exata do resultado final por meio de texto, com normas do tipo contratuais, números objetivos e indicadores de desempenho.	Busca elaborar a visão do produto contendo uma descrição abrangente do projeto de forma macro e desafiadora, orientada para a resolução de problemas e geralmente contendo artefatos visuais
3) O nível de detalhe e padronização com que cada atividade do projeto é definida	As atividades são descritas de maneira padronizada e organizadas em listas do tipo WBS. Contêm códigos e são classificadas em conjuntos de pacotes de trabalho, entregas e produtos do projeto.	Não há um padrão para a descrição das atividades, que podem ser escritas na forma de histórias, problemas, ações ou entregas. E não há uma tentativa de organização, apenas a priorização do que deve ser executado no momento.
4) O horizonte de planejamento das atividades da equipe de projeto	As listas de atividades são válidas para o horizonte total do projeto.	O plano é detalhado em curto prazo por meio de iteração (semanal, quinzenal ou mensal), criado em conjunto com os membros da equipe de projeto, incluindo os clientes)
5) A estratégia utilizada para o controle do tempo do projeto	Empregam-se relatórios com indicadores de desempenho, documentos escritos, auditorias e análises de transições de fase. As reuniões da equipe não são frequentes.	As mudanças são identificadas por meio de controles visuais e reuniões diárias, face a face, entre os profissionais envolvidos no projeto. A medição de progresso é orientada para resultados tangíveis (protótipos, demonstrações, desenhos ou artefatos visuais)
6) A estratégia utilizada para a garantia do atingimento do escopo do projeto	O cliente avalia, prioriza, adiciona ou altera o produto final do projeto, conforme a experiência com os resultados alcançados. A equipe altera as atividades para obter os resultados propostos pelo cliente.	O <i>Product Owner</i> avalia, prioriza, adiciona ou altera as atividades do projeto para que os resultados estejam em conformidade com o escopo do projeto assinado com o cliente.

**Fonte:** Eder, S. et al. (2015)

Posto isso e com toda essa mudança acerca do gerenciamento de projetos rumo a um ganho em agilidade, o próximo item descreve o processo de engenharia de requisitos e sua evolução para o contexto ágil a fim de apresentar as modificações que devem ser consideradas na forma como a tradicional engenharia de requisitos de *softwares* é feita.

## 2.2 ENGENHARIA DE REQUISITOS

O ciclo de vida natural de desenvolvimento de um *software* abrange as fases: concepção ou criação; construção ou programação; implantação (disponibilização); adaptações (pequenos ajustes ou melhorias); maturidade (utilização plena do sistema); declínio, manutenção; morte ou descontinuidade. Quando as três primeiras fases são elaboradas de forma errada, a morte do sistema de informação é acelerada, principalmente se o *software* focar a gestão estratégica da organização (REZENDE, 2005). Neste sentido, a engenharia de requisitos se concentra na fase de concepção. De acordo com Sommerville (2003), a engenharia de requisitos é sintetizada como o conjunto das atividades:

- Elicitação de Requisitos: Compreende o conjunto de atividades que o analista de requisitos (ou um membro da equipe) emprega para elicitare ou descobrir as solicitações dos usuários, determinando as reais necessidades por trás das solicitações. Esta interação com os *stakeholders* pode dar-se através dos mais diversos meios, como entrevistas, dinâmicas participativas, questionários, entre outros.

- Análise e Negociação de Requisitos: Nesta etapa está incluída a análise de todas as necessidades e expectativas levantadas na atividade anterior, que agora serão compiladas e ponderadas pelo Analista de Requisitos. A determinação de quais requisitos serão realmente tratados no desenvolvimento deve ser estabelecida por meio de um consenso entre a equipe de desenvolvimento e os *stakeholders*, como consequência da negociação;

- Especificação ou Documentação de Requisitos: Compreende a atividade de especificação dos requisitos do sistema, ou seja, a “tradução” dos desejos e necessidades dos *stakeholders* para uma forma mais estruturada, que possa ser compreendida pelos demais participantes da equipe de desenvolvimento;

- Validação de Requisitos: Através desta atividade os requisitos previamente documentados ou especificados são validados pelos *stakeholders*, sob o ponto de vista de sua consistência e completude.

Observando os escritos de Pressman (2006), Sawyer e Kotonía (2001), Sommerville (2003) e Thayer e Dorfman (1997) é possível notar que, para as atividades da engenharia de

requisitos, não há um consenso ou uniformidade quanto as nomenclaturas usadas em seus textos. De forma resumida, o Quadro 2 apresenta algumas denominações para essas atividades, baseados no modelo tradicional de desenvolvimento de software.

**Quadro 2** – Atividades da engenharia de requisitos de acordo com diferentes autores

<b>PRESSMAN, 2006</b>	<b>SOMMERVILLE, 2003</b>	<b>SAWYER e KOTONIA, 2001</b>	<b>THAYER e DORFMAN, 1997</b>
Concepção	Viabilidade	Elicitação	Elicitação
Elicitação	Elicitação		
Elaboração	Análise	Análise	Análise
Negociação			
Especificação	Especificação	Especificação	Especificação
Validação	Validação	Validação	Validação
Gestão	Gerenciamento	Gerenciamento	Gestão

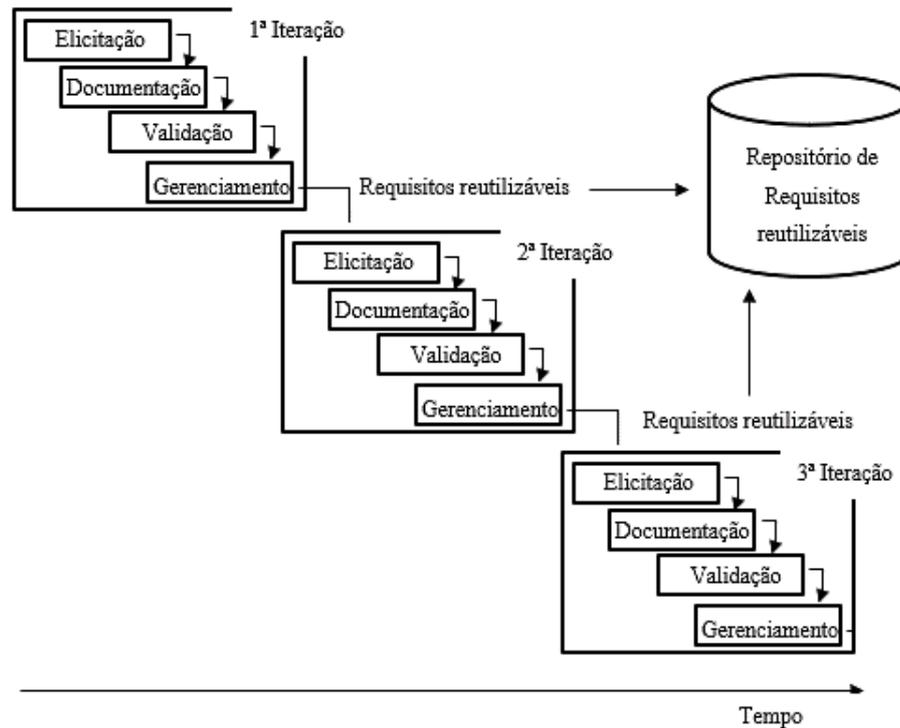
**Fonte:** Adaptado de Pressman (2006); Sommerville (2003); Sawyer e Kotonia (2001); Thayer e Dorfman (1997).

Ainda que com distintas nomenclaturas, as atividades de elicitação, análise, especificação, validação e gerenciamento de requisitos podem ser a síntese de todas as outras atividades (PRESSMAN, 2006).

Ainda que a engenharia de requisitos estivesse tradicionalmente bem definida e estruturada, o mundo dos negócios se tornou caracterizado pela complexidade, com as exigências mudando rapidamente e isto fez com que as empresas passassem a enfrentar o desafio de reduzir o tempo de colocação do produto no mercado enquanto entrega produtos inovadores que atenda e satisfaça o cliente. Para este fim, muitas empresas adotam métodos ágeis, onde a engenharia de requisitos promete benefícios como satisfação do cliente por meio de iterações curtas, realizadas de forma incremental e empírica (SCHÖN et al., 2017). Ainda de acordo com Schön et al. (2017), as principais atividades da engenharia de requisitos (elicitação, documentação, validação e gerenciamento) não são claramente separadas no contexto ágil. Elas são repetidas a cada iteração e as informações que podem ser reutilizadas são armazenadas em repositório para futuras referências, conforme exemplificado na Figura 2.

As metodologias ágeis (por exemplo, *Scrum* (SCHWABER, 2004), *Kanban* (ANDERSON, 2010) ou *Extreme Programming* (BECK, 2000)) fornecem um modelo de processo para desenvolver produtos, mas não definem uma estrutura prescritiva para a engenharia de requisitos.

**Figura 2** – Ciclo de iterações da Engenharia de Requisitos na abordagem ágil



**Fonte:** Elaboração própria

De acordo com Saeeda et al. (2020), os principais desafios para a engenharia de requisitos advindos destas abordagens flexíveis, inclui, mas não se limita a, identificação incompleta dos requisitos, implementação de requisitos vagos e comprometer o envolvimento dos usuários que negativamente impacta a funcionalidade e a qualidade do *software*. Ainda segundo o mesmo autor, para superar estes desafios, alguns dos recentes esforços de pesquisa têm se concentrado na gestão da fase de elicitação durante o desenvolvimento de *software* no contexto ágil, entretanto, há uma falta de soluções que fornecem aos engenheiros/operadores de *software* uma solução, conforme apresentado na sequência.

### 2.2.1 Elicitação de Requisitos

De acordo com o IEEE (1998), a elicitação de requisitos é um processo através do qual o cliente e o desenvolvedor de um sistema descobrem, avaliam, articulam, compreendem e documentam os requisitos do sistema e os processos do ciclo de vida. Já Pohl (2010) define a fase de elicitação como uma atividade central da engenharia de requisitos tradicional, considerando as atividades como se segue: (1) Adquirir conhecimento do projeto, (2)

Determinar as fontes de requisitos, (3) Definir a técnica de elicitação apropriada, (4) Identificar a requisitos dessas fontes (5) Documentar e (6) Refinar os requisitos.

Os requisitos elicitados ocupam-se de agrupar características do sistema, do projeto e dos requisitos de processo. Em geral, os requisitos são classificados como funcionais e não funcionais, e os métodos de elicitação se comportam de forma diferente dependendo do tipo de requisito a ser elicitado. Por exemplo, a revisão sistemática da literatura conduzida por Rodriguez et al. (2017) *apud* Ruedaa et al. (2020) comprova a existência dos muitos métodos de elicitação, incluindo JAD (*Joint Application Development*), entrevistas, questionários, análise de tarefas, entre outros. Os resultados mostram que ainda há importantes desafios para melhorar os métodos de elicitação de requisitos, inclusive compreender as necessidades para o projeto, além da solicitação do cliente. Outros trabalhos, como o de Carrizo et al. (2014) *apud* Ruedaa et al. (2020), concluem que o melhor método de elicitação de requisitos depende do contexto do projeto. Os métodos de elicitação baseados em entrevistas são provavelmente os mais tradicionais e comumente usados (SCHÖN et al., 2017).

No que diz respeito ao gerenciamento tradicional, a fase de elicitação acontece uma única vez no início do processo. Porém, no contexto ágil, esta fase de repete a cada interação, conforme Figura 2, e, portanto, tem suas particularidades (SCHÖN et al., 2017) apresentadas na sequência deste texto.

### **2.2.2 Elicitação de requisitos na Engenharia de Requisitos Ágil**

Em comparação à fase de elicitação de requisitos na abordagem tradicional, Schön et al (2017), descreve a engenharia de requisitos ágil acontecendo junto com o próprio desenvolvimento onde os requisitos são regularmente descritos a partir de uma perspectiva do cliente na forma de épicos e histórias de usuário (*User Stories*) em vez de criar um documento de requisitos (como Casos de Uso<sup>1</sup>, por exemplo).

A elicitação de requisitos na abordagem ágil, bem como na tradicional, inicia quando a necessidade de negócio é percebida dentro de uma organização e têm a pretensão de estabelecer o entendimento básico dos problemas, das pessoas que estão envolvidas com as soluções e da comunicação e colaboração efetiva entre clientes e desenvolvedores. A principal distinção está

---

<sup>1</sup> Casos de Uso: Documento elaborado pelo analista de requisitos para descrever o passo a passo de uma atividade do sistema a ser desenvolvido. Nele são apresentadas as regras de negócio, as funcionalidades e o diagrama de Caso de Uso. Neste último, são ilustrados os atores e suas atuações no sistema (em relação a atividade em questão).

na iteração das atividades de elicitação, pois, no contexto ágil, esta ocorrerá várias vezes durante os ciclos do projeto (PRESSMAN, 2006).

Dentro do contexto ágil, uma vez alocados os requisitos a uma iteração de desenvolvimento, estes passam a ser analisados, projetados, implementados, testados e implantados. Assim, na próxima iteração, um outro subconjunto dos requisitos (funcionalidades) será considerado para desenvolvimento, o que faz a equipe produzir uma nova versão (ou incremento) do sistema. Este incremento leva consigo extensões e refinamentos sobre a versão anterior. Com esses eventos cíclicos, o desenvolvimento passa a evoluir em versões, por meio da elicitação incremental e iterativa de novas funcionalidades até que o sistema completo esteja construído (BEZERRA, 2007), conforme ilustrado na Figura 2.

Ainda de acordo com Bezerra (2007), este modelo que incrementa e é iterativo contribui para o envolvimento e participação do cliente nas atividades de construção do *software*. Isso é possível uma vez que estes possuem, desde o início do projeto, uma versão do sistema para utilizar. É claro que as versões iniciais do sistema são desprovidas de grande parte das funcionalidades necessárias, no entanto, essa situação é muito melhor quando comparada àquela em que o usuário recebe o sistema todo de uma vez no final do projeto (conforme ocorre no modelo em cascata).

Segundo Rodriguez (2017), a elicitação é reconhecida como uma das atividades mais importantes em processo de desenvolvimento de *software*, pois tem impacto direto em seu sucesso. Segundo Pressman (2006), algumas técnicas comumente usadas na abordagem tradicional, podem ser otimizadas para apoiar o processo ágil, como: Entrevistas, Observação Direta, Protótipo, Análise de Documento, *Brainstorming* e JAD (*Joint Application Development*) (PRESSMAN, 2006).

E, embora exista estudos que fornecem novas práticas, dentro do contexto ágil, para melhorar esta tarefa, como Gamificação, *Design Thinking*, *Mind Mapping*, JRD (*Joint Requirement Document*), *Workshop*, UCD (*User-Centered Design*) e *Data driven* (HARBERS, et al., 2015; SFETSOS et al., 2016; MARTINS et al., 2019; SAEEDA et al.; ORIOL, et al., 2020), os autores concluem que ainda há questões a serem resolvidas, cuja as principais apontadas são a identificação dos requisitos não-funcionais, uso de artefatos que comprometem a agilidade e mudança frequente dos requisitos.

Após esta fase que descreve o arcabouço teórico conceitual, ficaram claras as necessidades e dificuldades do GAP em conquistar a aceitabilidade do produto desenvolvido por meio do envolvimento do usuário, distintamente do cliente, durante as atividades de elicitação. E então, surge a necessidade de ampliar este estudo e descobrir se existem práticas

efetivas de elicitación adotadas para projetos ágeis relatadas em outros estudos científicos com o propósito de alavancar o grau de aceite por parte dos usuários dos *softwares*. Este avanço no estudo é descrito no próximo capítulo por meio de uma Revisão Sistemática da Literatura.

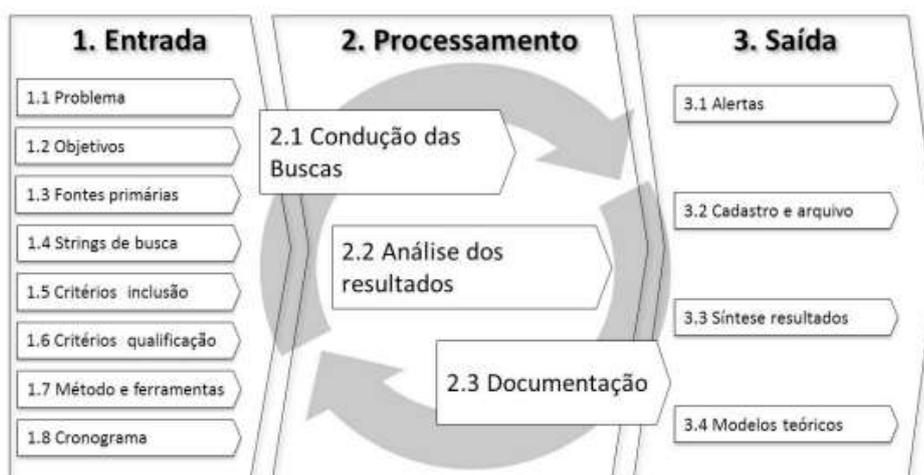
### 3 REVISÃO SISTEMÁTICA DA LITERATURA

Esta seção apresenta e discute pesquisas sobre a engenharia de requisitos na abordagem ágil e aprofunda a investigação a fim de verificar o estado da arte e, desta forma, dar sustentação à sistematização de propostas de solução para a lacuna apontada acerca da relação entre o envolvimento dos usuários nos processos de elicitação de requisitos sob a ótica da engenharia de *software* ágil e a aceitabilidade do produto desenvolvido.

#### 3.1 METODOLOGIA DA REVISÃO SISTEMÁTICA DA LITERATURA

Uma revisão sistemática da literatura (RSL) é um método científico que busca identificar, avaliar e interpretar resultados de pesquisas que sejam relevantes para uma determinada área da ciência (Kitchenham, 2004; Biolchini et al., 2007). Em razão do grande número de resultados que possam ser relevantes à questão de pesquisa, a RSL deve ser desenvolvida seguindo rigorosamente um método que esteja bem definido a fim de garantir a confiabilidade e a qualidade da síntese obtida. Foi por esta razão que Conforto, Amaral e Silva (2011) propuseram um roteiro aprimorando as propostas de Biolchini et al. (2007) e de Levy e Ellis (2006). Os autores criaram o roteiro seguindo o procedimento utilizado por Dyba e Dingsøyr (2008) e por meio de uma pesquisa-ação realizada em três pesquisas na área de gestão de operações. O roteiro recebeu o nome de *RBS Roadmap* e é organizado em fases e etapas, possuindo 15 etapas distribuídas em 3 fases (Entrada, Processamento e Saída). A Figura 3 ilustra as fases e etapas deste roteiro.

**Figura 3** – Representação do roteiro *RBS Roadmap*



**Fonte:** Conforto, Amaral e Silva (2011).

Na primeira fase, chamada de “Entrada”, são definidas as etapas: o problema (ponto de partida da RSL); os objetivos (claros, factíveis e alinhados com os objetivos do projeto de pesquisa); fontes primárias (artigos, periódicos ou bases de dados que serão úteis para a definição de palavras-chave, e identificação dos principais autores e artigos relevantes); *strings* de busca (palavras e termos referente ao tema de pesquisa); critérios de inclusão (caso os artigos não se enquadrarem nestes critérios, são excluídos no decorrer dos filtros de leitura); critérios de qualificação (avalia a importância do artigo para o estudo); método e ferramentas (define os filtros de busca, processo de busca nas bases de dados, armazenamento dos resultados, etc.) e o cronograma (é preciso estar atento para o prazo máximo viável para a condução da RSL).

Na segunda fase, chamada de “Processamento”, as etapas de busca, análise dos resultados e documentação seguem um processo iterativo contendo 7 passos. Esse processo está ilustrado na Figura 4. Estes passos compreendem as buscas, a leitura e análise dos resultados, a documentação e arquivamento dos artigos selecionados nos filtros, bem como os resultados das buscas e filtros de leitura. O Passo 1 consiste na busca por periódico, de forma individual, onde apenas o título, resumo e palavras-chave são lidos. Os artigos que estiverem alinhados com os objetivos da pesquisa e atenderem os critérios de inclusão serão selecionados para o próximo filtro (Passo 2). Os resultados são registrados individualmente, por periódico, utilizando um formulário (Form1). Este formulário documenta informações de busca, tais como: nome do periódico; Quantidade de artigos encontrados na busca; Quantidade de artigos selecionados (Filtro 1); Data da busca; Quantidade de artigos excluídos; % de aproveitamento.

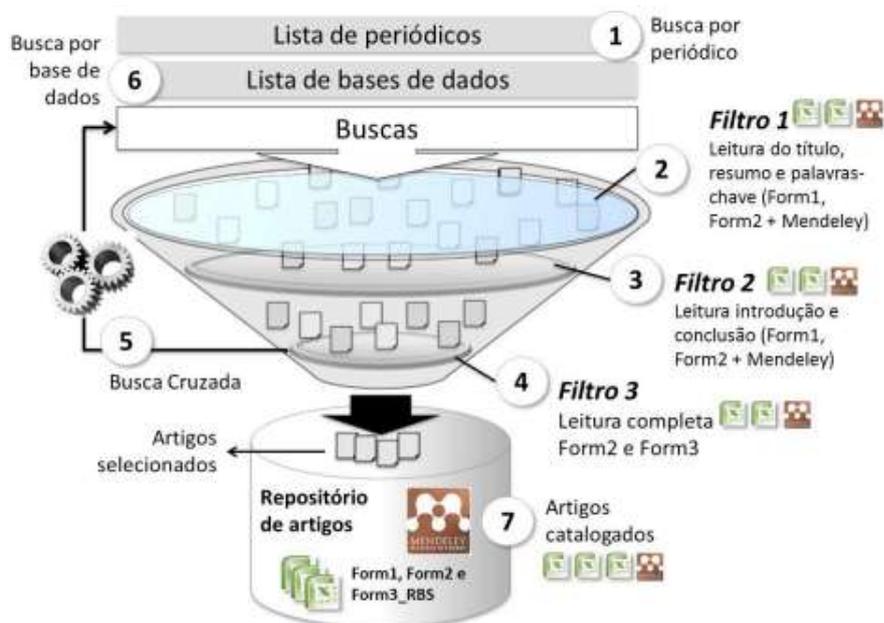
O Filtro 2 consiste na leitura da introdução e conclusão dos artigos, e repete a leitura do título, resumo e palavras-chave (Passo 3). Os artigos que não atendem os objetivos de busca e os critérios de inclusão são eliminados da revisão.

No Passo 4, é realizada a leitura completa (Filtro 3) dos artigos que passaram pelo Filtro 2. Os artigos que passam por este filtro são considerados relevantes para a pesquisa. É neste filtro ainda que se inicia o processo iterativo de busca cruzada (Passo 5), rastreando artigos por meio das citações dos autores. Nesse momento também são avaliados os critérios de qualificação do artigo.

No Passo 6 realiza-se a busca nas bases de dados. O principal objetivo é encontrar artigos recentes, publicados em anais de congressos e eventos científicos que estão indexados em bases de dados. Utilizou-se a *string* de busca para identificar um conjunto de artigos relevantes para o estudo.

Todos os resultados obtidos durante o processo são catalogados (Passo 7) a fim de que pudessem ser usados na fase final da revisão sistemática.

**Figura 4** – Procedimento iterativo da fase de processamento, RBS Roadmap.



**Fonte:** Conforto, Amaral e Da Silva (2011).

Na última fase, denominada “Saída”, incluindo 4 etapas, são inseridos os alertas nos principais periódicos identificados, realiza-se o arquivo (repositório) e escreve a síntese e resultados (relatório síntese da bibliografia estudada).

Cada etapa deste roteiro foi adaptada para esta RSL e suas fases e etapas são descritas no transcrito do texto que segue.

### 3.2 DEFINIÇÃO DO PROBLEMA, OBJETIVO E QUESTÕES DE PESQUISA

Alguns autores que se dedicaram a estudar as abordagens ágeis existentes alegam que a agilidade aplicada na fase de engenharia de requisitos é propensa a concentrar-se apenas na delimitação do contexto e na simplicidade e não na resolução e descoberta de problemas (Turk et al., 2014; Hosbond e Nielsen, 2008). Sendo, a fase de elicitação de requisitos, determinante para o bom êxito do projeto (Schön et al., 2017), propõe-se esta investigação com o propósito de aprofundar o conhecimento acerca das técnicas que promovam maior envolvimento do usuário nesta fase de modo a concentrar-se na resolução e descoberta de problemas, conquistando o aceite do usuário.

Com o propósito de identificar o estado da arte destes tópicos e a fim de facilitar a extração, análise e síntese dos resultados, foram definidas as seguintes questões específicas para a revisão da literatura:

- Q1: O que atualmente se sabe sobre os desafios e limitações das técnicas de engenharia de requisitos adotadas em projetos ágeis?
  - Q2: Quais técnicas estão sendo utilizadas para elicitação de requisitos em projetos que adotam metodologias ágeis?
  - Q3: Quais desafios encontrados para envolver o usuário no processo de elicitação de requisitos em projetos que adotam metodologias ágeis a fim de aumentar a aceitabilidade do produto?
  - Q4: Quais as implicações, para a indústria de *software* e para a comunidade acadêmica, dos atuais estudos que envolvem a ER em projetos ágeis?
- Estas questões serão apresentadas novamente no final deste capítulo organizando a síntese dos resultados provindos da RSL.

### 3.3 DEFINIÇÃO DAS FONTES PRIMÁRIAS E STRING DE PESQUISA

Após a definição dos objetivos e das questões que nortearão a revisão da literatura, foi necessário definir as fontes primárias. Para tanto, foram consultados especialistas e pesquisadores na área de engenharia de *software* a fim de receber indicações de artigos clássicos e periódicos relevantes para esta área de estudo, bem como se levou em consideração a experiência profissional da pesquisadora neste campo de estudo para encontrar fontes relevantes para o início de uma revisão bibliográfica preliminar exploratória, discutidas no capítulo anterior. O Quadro 3 apresenta as fontes primárias reunidas e informando: ano de publicação, autor(es) e títulos das obras.

**Quadro 3** – Fontes Primárias usadas na fase exploratória da RSL

Ano	Autor (es)	Título
1998	IEEE – <i>Institute of Electrical Electronics Engineers.</i>	<i>Guide for Developing System Requirements Specifications.</i>
2001	FOWLER, M.; HIGHSMITH, J.	<i>The agile manifesto</i>
2001	BECK, K., et al.	<i>Manifesto for agile software development.</i>
2004	HIGHSMITH, J.	<i>Agile Project Management: creating innovative products.</i>
2006	PRESSMAN, R. S.	Engenharia de <i>software</i> .
2011	AMARAL, D.C, et al.	Gerenciamento Ágil de Projetos – aplicação em produtos inovadores.
2017	SCHÖN, et al.	<i>Agile Requirements Engineering: A systematic literature review.</i>

**Fonte:** Elaboração própria.

Ainda nesta etapa, um conjunto de palavras-chave foi extraído dos textos cuja leitura fora realizada em caráter exploratório e, então, estas palavras foram combinadas com as questões de pesquisa. Em um segundo momento, as grafias e sinônimos alternativos foram identificados. Em se tratando de um processo de uma RSL, que é um aspecto crítico, as palavras-chave foram otimizadas de forma iterativa, ou seja, criadas e adaptadas na medida em que se acumulava conhecimento por meio das leituras dos textos identificados. Por fim, as palavras-chave foram definidas e, depois de testadas em vários bancos de dados e periódicos, estas foram finalmente refinadas. A lista final pode ser encontrada no Quadro 4.

**Quadro 4** – Palavras-chave usadas na pesquisa

<b>Categoria</b>	<b>Palavras-chave</b>
<i>Agile methodology</i>	<i>agile, agility, scrum, extreme programming, XP</i>
<i>User involvement</i>	<i>hci, usability, user, involvement, engagement, participation, customer</i>
<i>Requirements Engineering</i>	<i>requirements engineering, requirements, user Story, elicitation,</i>
<i>Improve</i>	<i>challenges, improve, improving, increase, increasing</i>
<i>Software</i>	<i>software, system</i>

**Fonte:** Elaboração própria.

Uma terceira fase compreendeu realizar uma conexão entre os termos (palavras-chave) com operadores booleanos (OR e AND) e, a partir disso, criar a *string* de pesquisa.

A aplicação da *string* buscou incluir periódicos e base de dados digitais contendo artigos relacionados à questão de pesquisa. Neste quesito, é importante lembrar que cada base de dados digital tem particularidades em relação ao mecanismo de busca e, em função disto, uma adaptação da *string* de pesquisa para cada base de dados foi requerida. Para as buscas, limitaram-se em todas as bases às áreas de pesquisa em Ciências da Computação, Projetos e Engenharia, uma vez que a problematização que envolve esta pesquisa diz respeito a projetos relacionados à estas áreas específicas. À medida que a *string* foi sendo aplicada nas bases de dados, o processo foi sendo documentado e o quadro 5 apresenta um extrato simplificado deste arquivo.

A princípio, os resultados totalizaram um volume 1.652 artigos relacionados à *string*. A partir deste ponto, iniciou-se um refinamento por meio dos critérios de inclusão/exclusão e análises a partir de leituras, conforme o fluxo demonstrado na Figura 5.

**Quadro 5** – Estratégias de busca

Base de dados	Estratégia de busca	Data da pesquisa	String	Resultado da busca
<i>Web of Science</i>	Tópico	15-01-2020		397
<i>Scopus</i>	Article title, Abstract and Keywords	20-01-2020	<i>(agile OR agility OR scrum OR extreme programming OR xp) AND (hci OR usability OR user OR involvement OR engagement OR participation OR customer) AND (“requirements engineering” OR requirements OR “user Story” OR elicitation) AND (software OR system)</i>	246
<i>IEEE Xplore</i>	All	22-01-2020		465
<i>Engineering Village</i>	Subject, Title and Abstract	20-01-2020		373
<i>Journal of Software</i>	Article title, Abstract and Keywords	15-02-2020		171

Fonte: Elaboração própria.

### 3.4 CRITÉRIOS DE INCLUSÃO

Foram considerados artigos publicados em revistas científicas e congressos, bem como escritos em inglês e português; publicados a partir do ano 2000<sup>2</sup>; artigos apresentando abordagens para envolver o usuário em processos de desenvolvimento ágil e artigos relacionados à fase de elicitação da engenharia de requisitos ágeis. Livros completos, estudos acadêmicos que tratam do ensino de metodologias ágeis, bem como artigos duplicados, foram descartados.

### 3.5 CRITÉRIOS DE QUALIFICAÇÃO

Para a avaliação dos artigos foi utilizado um questionário adaptado de Shön et al. (2017), observado a escala de três pontos de Likert (1932), como no Quadro 6.

**Quadro 6** – Questionário de qualidade aplicado nos artigos selecionados (continua).

	Questões	Nota	Descrição
<b>CQ1</b>	A proposta é validada?	0	Não é validada.
		0,5	É parcialmente validada em laboratórios.
		1	É validada através de Estudos de Casos.
<b>CQ2</b>	Existe uma descrição clara dos objetivos da pesquisa?	0	Não é possível identificar os objetivos.
		0,5	Os objetivos são definidos sem clareza.
		1	Os objetivos são claramente definidos.

<sup>2</sup> Ano 2000: Marcado pelo Manifesto Ágil, as empresas passaram a adotar metodologias ágeis em seus projetos.

	Questões	Nota	Descrição
CQ3	O estudo apresenta uma descrição pormenorizada da abordagem?	0	Não, faltam detalhes.
		0,5	Parcialmente, se quiser utilizar a abordagem, é necessário ler as referências.
		1	Sim, apresenta os detalhes com clareza.
CQ4	O estudo foi citado por outros autores?	0	Não, não há citações da pesquisa.
		0,5	Parcialmente. Entre 1-10 citações.
		1	Sim, mais de 10 estudos citam a pesquisa.
CQ5	Há uma descrição clara dos resultados?	0	Não é possível identificar os resultados.
		0,5	Os resultados são descritos sem clareza.
		1	Os resultados são claramente apresentados.
CQ6	O estudo apresenta um artigo isento de opinião ou um ponto de vista pessoal?	0	Não, o estudo apresenta opiniões e pontos de vistas pessoais.
		0,5	Parcialmente, o trabalho é explicado e inserido num contexto específico.
		1	Sim, o estudo é unicamente baseado na pesquisa.

Fonte: Adaptado de Shön et al. (2017).

Os artigos cuja soma classifica-se abaixo da faixa Média são desconsiderados e os demais, selecionados para análise e síntese dos resultados, conforme ilustrado no Quadro 7.

Quadro 7 – Pontuação por faixa de qualidade

Baixa	Média	Alta	Muito Alta
$0 \leq N \leq 2,5$	$3 \leq N \leq 3,5$	$3 \leq N \leq 4,5$	$5 \leq N \leq 6$

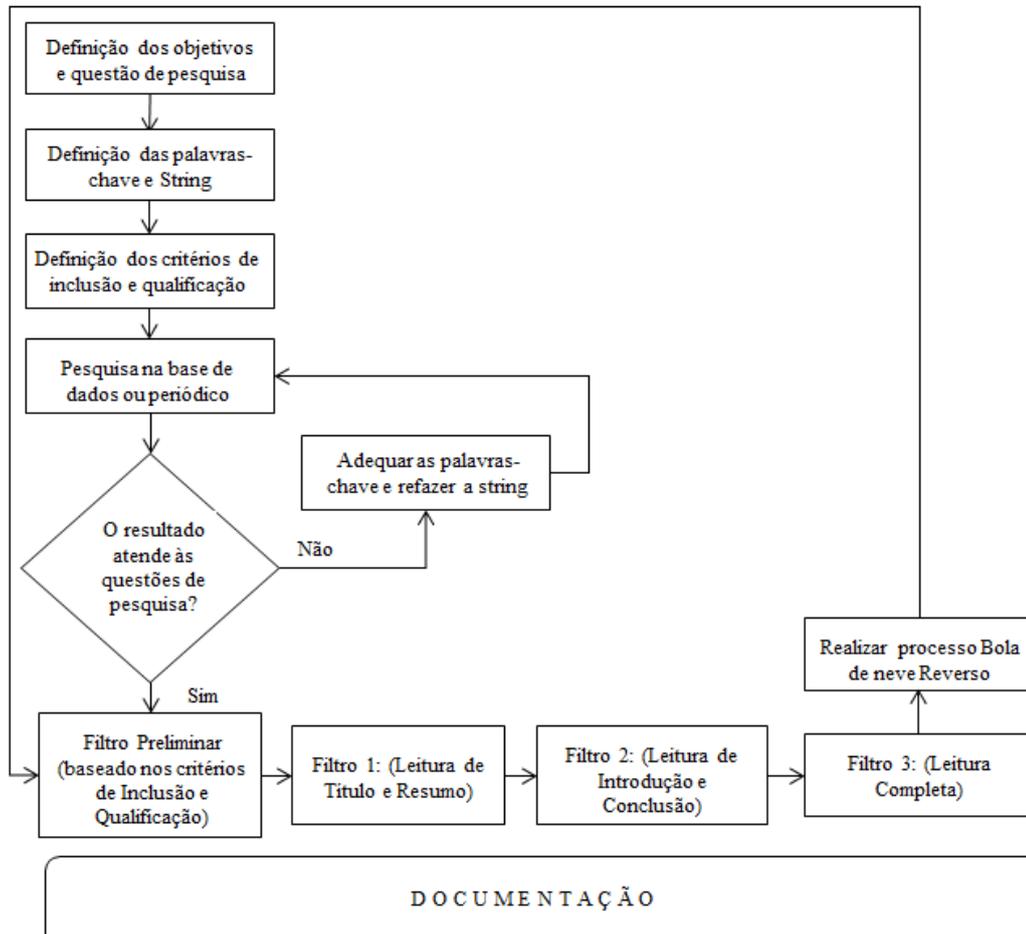
Fonte: Elaboração própria.

### 3.6 MÉTODOS E FERRAMENTAS

Cada ciclo do roteiro foi realizado de forma individual para cada base de dados e/ou periódico. A Figura 5 apresenta o roteiro que fora adaptado do modelo *RBS Roadmap*.

Para gerenciamento do resultado das pesquisas, fora utilizado o *software Mendeley* para marcar passagens de texto e classificações.

**Figura 5** – Fluxograma do roteiro adaptado de *RBS Roadmap*



**Fonte:** Elaboração própria.

### 3.7 CONDUÇÃO DAS BUSCAS

As buscas nas bases de dados e periódicos iniciaram-se em janeiro de 2020. Nesta fase, após as definições iniciais, iniciou-se a condução das buscas para extração dos dados dos artigos, onde a *string* fora aplicada nas bases de dados e periódicos. O Quadro 8 apresenta as informações sintetizadas desta planilha.

Foram extraídos inicialmente dados de publicação que permitiram o Filtro Preliminar visando aplicar os critérios de inclusão/exclusão previamente estabelecidos. Esta etapa iniciou com o total de 1.652 artigos e foram selecionados 252 artigos para compor a próxima fase. Para o Filtro 1, realizou-se a leitura de Título e Resumo a fim de identificar a aderência ao tema pesquisado e, selecionou-se o total de 95 artigos potencialmente relevantes para a próxima fase. No Filtro 2, com a leitura de Introdução e Conclusão, foram aplicados os critérios de inclusão e exclusão que, após investigação criteriosa, foram excluídos 66 artigos. Desta forma, o filtro 3

iniciou-se com 29 artigos para avaliação por meio da leitura completa dos materiais e aplicação dos critérios de qualificação. Nesta fase foram excluídos 9 artigos por não atenderem à qualidade esperada e por desviarem o foco do tema central da pesquisa. Contudo, ainda nesta fase (Filtro 3), tendo feito a triagem dos artigos, foi utilizada a técnica de *snowballing* (WOHLIN, 2014) para identificar artigos adicionais a partir das listas de referência dos artigos selecionados. Esta técnica possibilitou a inclusão de outros novos 5 artigos que, passaram pelos mesmos critérios de avaliação e foram inseridos na lista final de selecionados.

**Quadro 8** – Registro das buscas nas bases de dados e periódicos da RSL

Registro das buscas nas base de dados e periódicos									
Fonte	Data da Busca	Resultado	Filtro Preliminar	Filtro 1	Filtro 2	Filtro 3	<i>Snowballing</i>	% aproveitamento	Selecionados
<i>Web of Science</i>	15/01/2020	397	89	47	11	9	2	2,77%	25
<i>Scopus</i>	20/01/2020	246	46	15	5	3	1	1,63%	
<i>IEEE Xplore</i>	20/01/2020	465	61	18	6	4	1	1,08%	
<i>Engineering Village</i>	22/01/2020	373	25	6	3	2		0,54%	
<i>Journal of Software</i>	15/02/2020	171	31	9	4	2	1	1,75%	
<b>Totais</b>		<b>1652</b>	<b>252</b>	<b>95</b>	<b>29</b>	<b>20</b>	<b>5</b>	<b>7,76%</b>	

Filtro Preliminar: Critério de Inclusão

Filtro 1: Leitura de Título e Resumo

Filtro 2: Leitura de Introdução e Conclusão

Filtro 3: Leitura completa

**Fonte:** Elaboração própria.

Por fim, com um aproveitamento de quase 8%, a RSL fará síntese de 25 artigos considerados impactantes para o entendimento do estado da arte em relação ao objetivo de pesquisa e, por meio da análise destes materiais é elaborado o relatório síntese da revisão que contempla de forma organizada as contribuições dos autores para o tema abordado na pesquisa e, comparam-se as discrepâncias e/ou analisa as suas complementariedades.

### 3.8 RESULTADOS

Em primeiro lugar, serão descritas as características dos estudos selecionados por meio da apresentação de dados quantitativos. Após, serão apresentadas as percepções obtidas dos materiais selecionados e as conclusões relacionadas com as questões de pesquisa desta RSL.

### 3.8.1 Resumo dos Estudos

Em relação aos canais de publicação, os estudos selecionados são os publicados em anais de conferências ou em revistas científicas. Em comparação, 14 (56%) dos estudos incluídos foram publicados em anais de conferências e 11 artigos (44%) apareceram em revistas científicas.

O quadro 9 apresenta a distribuição dos estudos e seus métodos de investigação. Em resumo, 10 (40%) dos materiais selecionados são Estudos de Caso e, outros 9 (36%) realizaram uma Revisão Sistemática da Literatura, incluindo um estudo que realizou Mapeamento. Outros métodos encontrados foram: Delphi, *Design Science*, Experimento e um estudo empírico, cujo método não ficou evidente por meio da leitura.

**Quadro 9** – Distribuição de acordo com o método de pesquisa

Método de Pesquisa	Quantidade	Percentual
Estudo de Caso (único)	7	28%
Estudo de Caso (múltiplos casos)	3	12%
Revisão Sistemática da Literatura (incluindo Mapeamento)	9	36%
Pesquisa-Ação	2	8%
Experimento	1	4%
Design Science	1	4%
Método Delphi	1	4%
Estudo Empírico	1	4%
<b>Total</b>	<b>25</b>	<b>100%</b>

**Fonte:** Elaboração própria.

Em suma, pode afirmar-se que a RE em metodologias ágeis é frequentemente investigada no contexto da vida real, uma vez que quase metade (40%) dos estudos considerados buscaram imergir em um contexto prático e real denotando ainda que este campo de investigação está muito próximo das práticas de trabalho existentes uma vez que em todos estes estudos foi possível encontrar situações adequadas à pretensão dos estudos dentro das empresas.

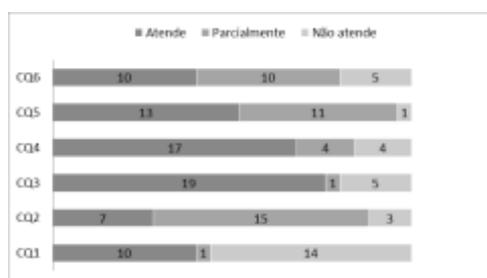
No que diz respeito a qualidade dos materiais selecionados, foram empregados os critérios de qualificação conforme supracitado e o resultado é apresentado no Quadro 10.

**Quadro 10** – Distribuição dos estudos com base nas notas dos critérios de qualificação.

Média	Alta	Muito Alta
$3 \leq N \leq 3,5$	$4 \leq N \leq 4,5$	$5 \leq N \leq 6$
4	14	7

Fonte: Elaboração própria.

Os resultados da avaliação da qualidade são apresentados de acordo com os critérios na Figura 6.

**Figura 6** – Distribuição dos estudos de acordo com os critérios de qualificação

Fonte: Elaboração própria.

O primeiro critério (CQ1) examina se a proposta é validada. Para 10 artigos, isto é verdade, uma vez que os autores fizeram uso de metodologias, como Estudo de Caso, *Design Science* e Pesquisa-Ação para validar a proposta do estudo. O estudo que utilizou o método Delphi para validar a pesquisa recebe nota 0,5 para este critério, uma vez que os especialistas usaram de suas experiências práticas para tratarem do assunto, no entanto, a teoria não foi comprovada no contexto prático.

No segundo critério (CQ2) foram testados se os objetivos do estudo foram incluídos de forma clara no artigo. Em apenas 7 dos trabalhos selecionados, os objetivos foram bem descritos e claros. No entanto, também apenas três artigos não apresentaram os objetivos, uma vez que estes foram descritos de forma muito pouco clara.

Com o terceiro critério (CQ3) é possível confirmar se o estudo em avaliação apresenta uma descrição pormenorizada da abordagem. Em 19 estudos, a abordagem é descrita com detalhes suficientes para o entendimento de modo que outros investigadores possam utilizar a mesma abordagem seguindo a escrita. Para um documento é necessário incluir a leitura das referências para entendimento e acompanhamento da abordagem. Os demais, 5, não trazem clareza suficiente na abordagem para total compreensão e reutilização.

O critério quatro (CQ4) questionou quantas vezes o estudo havia sido citado noutros artigos. Para este efeito, fora utilizado o número de citações do Google Acadêmico (data de

avaliação 02-06-2020). Nesta avaliação o resultado da qualificação apresentou 16 estudos com mais de dez citações em outros artigos, seis tinham sido citados entre 1-10 artigos e outros três artigos não tinham qualquer citação até à data da avaliação. Vale ressaltar que os resultados podem ser diferentes durante a leitura desta SLR, devido ao número de citações que pode variar em função da data em que se faz a leitura, principalmente por envolver 3 estudos que foram publicados nos primeiros trimestres de 2020 e, portanto, ainda passam por processo de reconhecimento da comunidade acadêmica.

Com o critério de número cinco (CQ5) foi possível avaliar a clareza de apresentação dos resultados da pesquisa. Neste sentido, 13 dos 25 trabalhos descrevem com clareza de definição os resultados apurados por meio da pesquisa, enquanto 11 declaram de forma subjetiva e apenas um não declara os resultados ou não foi possível identificar por meio da leitura.

O critério de qualificação seis (CQ6) avalia se o estudo fornece uma opinião ou um ponto de vista pessoal. Nesta avaliação, somaram-se 5 artigos cuja escrita não se isenta de opiniões e ponto de vista dos autores, de modo que a confiabilidade deles tendeu a ficar comprometida. Todavia, estes não deixaram de ser considerados nesta RSL por se tratar de especialistas e profissionais na área do estudo e, portanto, estavam imersos em um contexto prático para emissão de opiniões que puderam contribuir com a comunidade acadêmica e empresarial.

Em resumo, dos 25 estudos analisados, apenas 8 documentos preencheram todos os critérios de qualidade, apresentados no Quadro 11.

**Quadro 11** – Estudos que preencheram os 6 critérios de qualidade.

2014	INAYAT, S. et al.	<i>A systematic literature review on agile requirements engineering practices and challenges.</i>
2016	BJARNASON, et al.	<i>A multi-case study of agile requirements engineering and the use of test cases as requirements</i>
2017	SCHÖN, W. et al.	<i>Agile Requirements Engineering: A systematic literature review.</i>
2017	RODRIGUEZ, G., et al.	<i>Systematic Literature Review About Software Requirements Elicitation</i>
2019	AINHOA, A., et al.	<i>Leveraging creativity in requirements elicitation within agile software development: A systematic literature review</i>
2020	MEDEIROS, J. et al.	<i>Requirements specification for developers in agile projects: Evaluation by two industrial case studies</i>
2020	SAEEDA, et al.	<i>A proposed framework for improved software requirements elicitation process.</i>
2020	RUEDAA, S., et al.	<i>Requirements elicitation methods based on interviews in comparison: A family of experiments</i>

**Fonte:** Elaboração própria.

### 3.8.2 Mapeamento das evidências

Nesta seção são apresentados os resultados da revisão sistemática, alinhados de acordo com cada questão de pesquisa. Contudo, com o objetivo principal de identificar os desafios mais importantes em engenharia de requisitos ágil nas empresas e com a intenção de obter uma compreensão compartilhada sobre esses desafios entre os especialistas no campo da engenharia de requisitos ágil, o Apêndice A sintetiza uma visão geral das percepções dos principais estudos levantados<sup>3</sup> e seus respectivos métodos de pesquisa. Os estudos estão classificados em ordem crescente do ano de publicação.

Em síntese dos trabalhos analisados, é importante ressaltar que, embora o estudo de Paetsch et al. (2003), de Ramesh et al. (2010), bem como outros artigos selecionados, cuja pesquisa aconteceu há mais de 5 anos, tenham trazido resultados significativos sobre o tema, é preciso esclarecer se os desafios relatados, embora ainda relevantes atualmente, foram acrescidos e/ou alterados uma vez que esta área de desenvolvimento de *software* sofre mudanças significativas em curto espaço de tempo e o conjunto de conhecimentos na área evoluiu nos últimos anos.

Outra ressalva se encontra nos trabalhos, como o de Bjarnason et al. (2011), que retratam estudos de caso que foram realizados em uma única empresa, portanto, os resultados podem não ser aplicáveis a outras empresas e não podem ser generalizados.

E ainda, os estudos de Inayat et al. (2015), Heikkila et al. (2015), Soares et al. (2015), Shön et al. (2017), Satria et al. (2017), Martins et al. (2019) e Ainhua et al. (2019) tem por objetivo fazer uma revisão de estudos primários de forma a aplicar uma análise da literatura existente e identificar o estado atual da arte. Nesta abordagem, é possível obter uma impressão de aspectos relevantes do ponto de vista teórico, todavia, não é considerada uma abordagem apropriada quando a pretensão é analisar os desafios na prática.

#### **(Q1) – O que atualmente se sabe sobre os desafios e limitações das técnicas de engenharia de requisitos adotadas em projetos ágeis?**

Para o mapeamento dos desafios das atividades de requisitos em projetos ágeis, consideraram-se as recomendações de Dybå (2011) identificando-os por categorias. Após revisões e análises, de modo a eliminar duplicações e agrupar similaridades, chegou-se a um

---

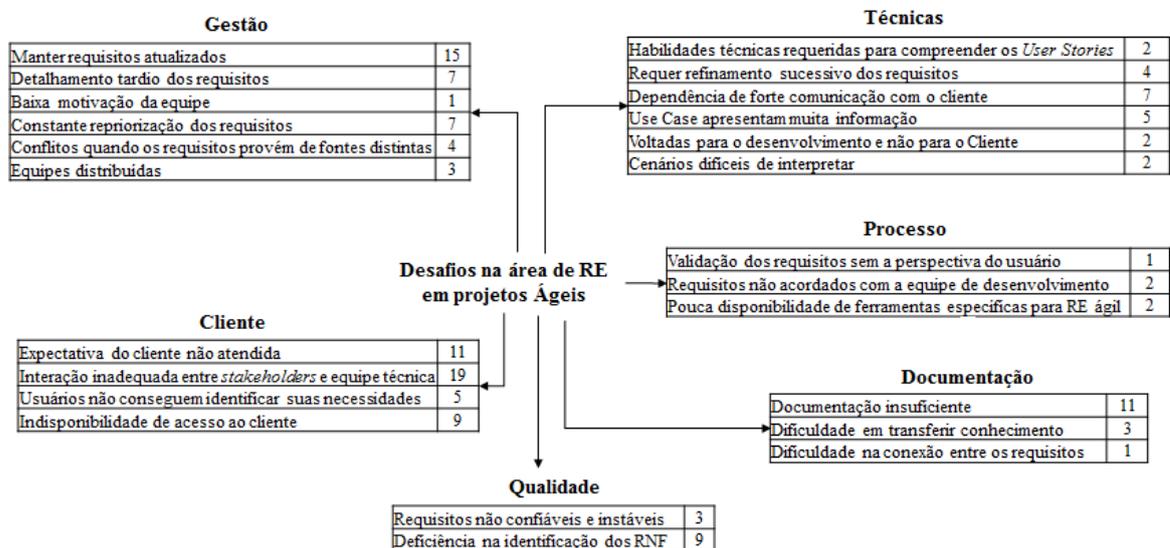
<sup>3</sup> Quadro com a visão sintetizada das percepções de todos os 25 estudos selecionados para a RSL.

total de 6 categorias, sendo estas: Gestão, Cliente, Técnicas, Processo, Documentação e Qualidade.

A Figura 7 ilustra os desafios que foram mapeados por categoria e seus respectivos temas, apresentando ainda a quantidade de ocorrências levantadas por estudo, considerando a lista daqueles que preencheram os 6 critérios de qualidade. Mais detalhes podem ser observados através da leitura do Apêndice B.

Analisando o total de problemas por Categorias, é possível constatar que os desafios relacionados à Cliente e Gestão foram os que apresentaram os maiores números de ocorrências de problemas – 44 e 37, respectivamente.

**Figura 7** – Desafios da ER na abordagem ágil agrupados por Categorias



**Fonte:** Elaboração própria.

É válido destacar os desafios “Interação inadequada entre *stakeholders* e equipe técnica”, com 19 ocorrências na categoria Cliente, e “Manter requisitos atualizados”, com 15 ocorrências na categoria Gestão. Esse fato aponta que os valores ágeis “Equipes se adaptam rapidamente às mudanças” e “Interação contínua com o cliente” não correspondem com a realidade apresentada pela literatura, uma vez que empresas que se adequam de forma ágil às mudanças têm potencial para manter seus artefatos atualizados, bem como aquelas com habilidade em interagir continuamente com o cliente não apresenta número elevado de ocorrências neste desafio.

De forma a analisar os demais desafios, é possível destacar ainda as ocorrências: “Expectativa do cliente não atendida”, “Documentação Insuficiente”, com 11 ocorrências; “Deficiência na identificação dos RNF” e “Indisponibilidade de acesso ao cliente”, com 9

ocorrências cada; “Constante priorização dos requisitos”, “Dependência de forte comunicação com o cliente” e “Detalhamento tardio dos requisitos” todos com 7 ocorrências cada.

Outros estudos desta RSL, além dos do Quadro 11, vão na mesma linha e retratam esta questão.

Um estudo empírico de Ramesh et al. (2010) com base na análise dos dados coletados em 16 organizações de desenvolvimento de *software* dos Estados Unidos revela em concordância, os sete desafios existentes na Engenharia de Requisitos em metodologias ágeis, sendo estes: (1) estimativa de custo e cronograma; (2) arquitetura inadequada; (3) negligência de requisitos não funcionais; (4) acesso e participação do cliente; (5) priorização em uma única dimensão; (6) verificação inadequada de requisitos e (7) documentação mínima. E, dos sete desafios, três estão diretamente relacionados com a falta de envolvimento do cliente no processo de levantamento de requisitos, os itens 1, 3 e 4.

Por sua vez, Bjarnason, Wnuk & Regnell (2011), por meio de um estudo de caso realizado em uma empresa, apresenta como desafios as ocorrências relacionadas ao planejamento da agilidade; fraca priorização de requisitos; problemas de qualidade; sistema concluído tardiamente; falta de requisitos documentados; função de cliente-representante e requisitos fracos nas primeiras iterações do projeto.

Já o estudo realizado por Anitha & Prabhu (2012), discute alguns dos conceitos errados e mitos que existem na indústria relativamente à Engenharia de Requisitos e como as necessidades do usuário normalmente escapam pelas fendas sob o nome de "requisitos não funcionais". Dois desafios apontados pelo estudo são: (1) Comunicação dos requisitos dos usuários num formato que os usuários possam facilmente validar; (2) Quebrar as necessidades dos usuários em termos tangíveis de concepção que o desenvolvimento possa facilmente implementar.

Ainda nesta análise com o propósito de identificar os desafios e limitações da RE na abordagem ágil, um mapeamento da literatura realizada por Heikkila et al. (2015), aponta os principais desafios: Problemas com clientes ou representantes de clientes; insuficiência de formato de *User Stories* e dificuldades para confiar no conhecimento de requisitos tácitos.

Uma revisão sistemática das práticas e desafios encontrados na engenharia de requisitos em metodologias de *software* ágil apresentadas por Inayat et al. (2014) relata que as práticas de engenharia de requisitos utilizadas nas metodologias ágeis de desenvolvimento de *software* podem compensar algumas das deficiências encontradas nas abordagens tradicionais. No entanto, esta abordagem flexível ainda enfrenta alguns desafios como a falta de envolvimento

dos usuários, a negligência dos requisitos não-funcionais e problemas com a estimativa do custo e do prazo, que estão diretamente relacionados com a fase de levantamento dos requisitos.

Já Soares et al. (2015) investigam a Engenharia de Requisitos com foco na fase de elicitação e apresenta nove desafios enfrentados durante esta fase, apresentados no Quadro 12.

Em um estudo sobre Especificação de Requisitos para desenvolvedores em projetos ágeis pesquisado por Medeiros et al. (2020), uma investigação apontou várias limitações relativas às suas atividades de engenharia de requisitos, como a Especificação de Requisitos de *Software* orientada para o cliente, a falta de informação necessária para realizar atividades de concepção e a baixa disponibilidade do cliente. Para ultrapassar estas questões, o estudo propôs uma abordagem nomeada de RSD (Especificação de Requisitos para Desenvolvedores) para criar especificações que forneça informação mais próxima das necessidades de desenvolvimento. Esta pesquisa foi estruturada através de dois estudos de casos industriais, focando os engenheiros de *software* como unidade de análise. A avaliação indicou como limitações: necessidade de priorização e categorização dos critérios de aceitação, uma ferramenta para criar, pesquisar e rastrear os artefatos, e obter testes de aceitação a partir dos critérios de aceitação.

**Quadro 12** - Desafios enfrentados durante a elicitação dos requisitos

ID	Desafios	Descrição
1	Priorização dos requisitos	A priorização dos requisitos com base apenas no valor que estes têm para o usuário também representa um risco para o projeto.
2	Identificação dos requisitos não-funcionais	A falta de especificação de requisitos não funcionais pode provocar problemas futuros.
3	Especificação dos requisitos	A metodologia ágil aborda geralmente user stories como um meio de especificar os requisitos. Muitas vezes apresentam baixo nível de detalhe, causando dificuldades para outras atividades de desenvolvimento.
4	Alteração dos requisitos	Há que prestar atenção às constantes mudanças de requisitos em projetos ágeis, avaliando os seus impactos e riscos.
5	Definição dos requisitos	As dificuldades que os usuários enfrentam para entender o que deve ser descrito pela equipe de desenvolvimento.
6	Conexão entre os requisitos	Dificuldade em lidar com a conexão entre requisitos, uma vez que no desenvolvimento ágil, os user stories podem ser desenvolvidos individualmente.
7	Envolvimento dos stakeholders	Os usuários nem sempre estão disponíveis para responder a perguntas sobre requisitos de software.
8	Comunicação com os stakeholders	Muitas vezes revela-se difícil comunicar com as partes interessadas de uma forma eficiente. Isto pode ter consequências indesejáveis, uma vez que os requisitos ágeis se baseiam na comunicação contínua e na cooperação com o stakeholder..
9	Validação dos requisitos	A validação dos requisitos pode ser deficiente devido ao baixo nível de especificação dos requisitos quando se utiliza uma metodologia ágil.

**Fonte:** Soares, et al. (2015)

Já Rodriguez et al. (2017), em uma RSL de 35 artigos considera não haver propostas consistentes sobre fatores correspondentes a "Determinar Fontes" e "Definir Técnica". Portanto, recomenda o aprofundamento nas pesquisas visando uma proposta que cubra as atividades no processo de levantamento de requisitos, tais como "Adquirir conhecimento de domínio", "Definir uma técnica adequada", "Determinar a fonte dos requisitos".

Para essa questão as respostas são mais consensuais e em síntese podemos dizer que os estudos apontam o envolvimento do cliente e/ou seu representante como um dos principais desafios a serem enfrentados pela fase de elicitação dos requisitos, bem como problemas com a linguagem escrita dos requisitos, ora voltada para o cliente, desencadeando dificuldade de compreensão por parte dos desenvolvedores, ora uma escrita técnica que dificulta a validação do cliente. As dificuldades para adquirir conhecimento sobre as tarefas onde o *software* irá atuar também mostrou ser um desafio encontrado amplamente nos estudos desta revisão.

### **(Q2) – Quais técnicas estão sendo utilizadas para elicitação de requisitos em projetos que adotam metodologias ágeis?**

O Quadro 13 mostra os resultados da avaliação dos estudos relacionados com a Q2. Por conseguinte, são apresentados os destaques encontrados nos estudos incluídos relativos a esta questão.

A síntese dos estudos resultou em um total de 6 estratégias para elicitar requisitos em projetos que adotam alguma metodologia ágil citados em mais de 20% dos estudos selecionados, conforme pode ser observado também no Quadro 13.

**Quadro 13** – Estratégias para elicitação de requisitos em projetos ágeis

ID	Técnicas	% de citação
1	Entrevistas	92%
2	<i>User Story</i>	84%
3	Protótipo	60%
4	<i>Design Thinking</i>	44%
5	UCD (Design Centrado no Usuário)	32%
6	<i>Story card</i>	20%
7	Questionários	16%
8	Caso de Uso / Cenários	8%
9	JRD ( <i>Joint Requirements Document</i> ) e Mind Mapping	8%
10	<i>Workhop</i>	4%
11	GREM ( <i>Gamified Requirements Engineering Model</i> )	4%

**Fonte:** Elaboração própria.

A realização de Entrevistas com o cliente é a técnica mais utilizada. Além desta, foram citadas JRD (*Joint Requirements Document*) e *Mind Mapping*, Questionários, *Design Thinking*, Protótipos, UCD, *Workshop* e GREM (*Gamified Requirements Engineering Model*) como técnicas mais atuais utilizadas e sugeridas para levantamento de requisitos.

Kautz (2010) estudou a forma como os clientes e os usuários participaram do desenvolvimento ágil por meio da contribuição da técnica de Design Participativo. A pesquisa fornece um estudo de caso de grande escala no âmbito de desenvolvimento ágil e concentra-se no papel dos clientes e na forma como estes foram envolvidos através de diferentes atividades de *design* e desenvolvimento. O projeto investigado utilizou o método *Extreme Programming* (XP) onde, Gamificação, *User Stories*, *Story Cards*, *Workshop* e testes de aceitação estruturaram o envolvimento do cliente. Os autores afirmam terem encontrado envolvimento do cliente na concepção das atividades sob a forma de participação direta e indireta no projeto de desenvolvimento ágil.

Da mesma forma, Bellucci et al. (2015) investigaram uma abordagem ao *design* com e pelo usuário, razão pela qual realizaram um estudo de campo para avaliar a forma como os usuários interagem com um protótipo. Com base nos seus resultados, o protótipo foi desenvolvido iterativamente através de loops de *feedback* constantes (semanalmente) emitidos pelos usuários. Os usuários principais escolhidos participaram ativamente no processo de inovação através da geração de ideias, bem como em todas as fases do processo de desenvolvimento através de ferramentas de *co-design*. A abordagem é chamada de *Co-design Extrem*, onde o *design* é desenvolvido em ciclos rápidos e permitem a exploração extensiva e vivência de cenários junto aos usuários.

Outra técnica que tem sido utilizada como forma de elicitação de requisitos e imersão nas áreas de processo é o *Design Thinking*. Com o uso da triangulação de dados, o artigo de Martins et al. (2019) traz uma revisão bibliográfica que recolhe os desafios na elicitação de requisitos de *software* em metodologias ágeis e o uso do *Design Thinking*. O resultado deu lugar a um estudo de caso em um projeto de organização pública brasileira, através de questionário de *workshop* de usuários com 20 itens. Os autores trazem como conclusão a afirmação de que, por meio da aplicação da abordagem de avaliação no caso estudado, há indícios de contribuição, sobre o uso de *Design Thinking*, para a maioria dos desafios de elicitação de requisitos em metodologia ágil.

Já Ainhoa et al. (2019) estudaram por meio de uma RSL como a técnica de *Design Thinking* pode contribuir para aumentar a criatividade no processo de elicitação de requisitos e enfrentar os desafios já relatados pela literatura atual. Os autores deste artigo sugerem um

aprofundamento no estudo para se obter uma visão detalhada da situação atual na indústria, a fim de analisar as circunstâncias específicas dos profissionais e dos projetos em termos de localização e recursos, bem como para contrastar os resultados da investigação com a situação atual da indústria.

Em um estudo de Lombriser, Lucassen & Brinkkemper (2016) apud Ainhoa et al. (2019), os autores apresentam uma técnica nomeada de GREM (*Gamified Requirements Engineering Model*), que integra as teorias da gamificação e do envolvimento no contexto do desempenho da engenharia de requisitos. Gamificação é o uso de mecânicas e características de jogos para engajar, motivar comportamentos e facilitar o aprendizado de pessoas em situações reais, normalmente não relacionados a jogos. Os resultados da experiência evidenciaram este aumento da criatividade e do desempenho em termos de quantidade e qualidade dos requisitos. No entanto, os elementos de jogos que parecem trazer mais criatividade aos requisitos podem não ser adequados em fases em que a cooperação e o intercâmbio de ideias são necessários. Como trabalho futuro, os autores alertam que são necessárias mais experiências com diferentes tamanhos de amostra e estrutura de jogos para generalizar os resultados e a aplicabilidade do GREM.

Em outro artigo recentemente publicado no *Journal of Software* por Saeeda, et al. (2020), lê-se uma proposta de um *framework* para melhorar os processos de elicitação dos requisitos de *software* no *Scrum* por meio de uma implementação em um projeto de TI. Os autores propõem um *framework* que unifica o *Joint Requirements Document (JRD)*, *Mind Mapping* e *Scrum* para incrementalmente identificar, implementar, evoluir e gerenciar os requisitos do sistema ao longo do ciclo de desenvolvimento de *software*. Para esta pesquisa foi utilizado a abordagem metodológica de pesquisa-ação. Vale ressaltar que este é o único estudo em que se observa a distinção clara entre os papéis de “cliente” e “usuário final”, principalmente ao sugerir que sessões interativas com clientes e usuários finais são benéficas para o reconhecimento rápido dos requisitos por meio da mistura de técnicas de *software* nas sessões JAD e JRD.

No estudo mais recente, escrito por Oriol et al. (2020), os autores estudam como executar a fase de elicitação dos requisitos em empresas ágeis com base em dados e ferramentas, chamados de “*Data-driven and tool-supported*”. A abordagem baseia-se na declaração de limites sobre questões relacionados à qualidade, cuja violação desencadeia alertas definidos pelo usuário. Os pesquisadores utilizaram instrumentos de medição padronizados para colher a percepção de 22 profissionais envolvidos na utilização da ferramenta. As análises quantitativas

e qualitativas produziram resultados positivos, ou seja, a percepção dos profissionais em relação à compreensibilidade, fiabilidade, utilidade e relevância da ferramenta foi positiva.

Ainda neste ano, foi publicado um experimento de Ruedaa et al. (2020) onde os autores encontraram algumas diferenças significativas nas técnicas de elicitação, apresentando que a Prototipagem produz os melhores resultados para obter o maior número possível de requisitos e é a mais adequada para atender aos requisitos funcionais, a técnica JAD requer o maior tempo para relatar os requisitos, é a mais adequada para requisitos não funcionais e evita duplicações, enquanto que as Entrevistas não estruturadas necessitam de menor tempo para relatar os requisitos, mas de baixa qualidade nos resultados produzindo maior sobreposição.

**(Q3) – Quais desafios encontrados para envolver o usuário no processo de elicitação de requisitos em projetos que adotam metodologias ágeis a fim de aumentar a aceitabilidade do produto?**

Poucos foram os estudos encontrados que procuram envolver o usuário no processo de elicitação de requisitos. Muitos deles mencionam a participação (não envolvimento) do cliente durante a fase de Engenharia de Requisitos de forma geral, sem especificar a fase de elicitação. Nota-se também nesta análise que muitos dos estudos não fazem menção à distinção dos termos cliente, *stakeholder* e usuário e, desta forma, não é possível compreender com clareza se o estudo se relaciona com o perfil cliente ou com perfil usuário que, para esta RSL, é critério de observação. Dos poucos estudos que diferenciam os termos supracitados, Paetsch et al. (2003), menciona os termos "*customer involvement*", "*user involvement*" e "*stakeholder involvement*" no mesmo parágrafo, parecendo tratar do mesmo papel.

Um estudo de caso realizado por Ramesh, Cao & Baskerville (2010) em 16 organizações de desenvolvimento de *software* nos Estados Unidos da América revelaram sete desafios encontrados na Engenharia de Requisitos usando abordagens ágeis para o desenvolvimento de *software*. Entre eles, um está diretamente relacionado com a atividade de envolver o usuário no processo de elicitação de requisitos. O desafio é chamado de “participação dos clientes” e menciona que a comunicação com as partes interessadas depende de muitos fatores, tais como disponibilidade, consenso e confiança, especialmente nas fases iniciais do projeto. Um desenvolvedor de *software* que participou deste estudo relatou que os usuários adequados (finais) estarão sempre ocupados com as suas atividades na organização ou o seu chefe não lhes permitirá juntar à equipe de desenvolvimento em tempo integral.

Outros estudos selecionados para a revisão apresentam meios pelos quais as empresas buscam enfrentar os desafios relatados na literatura, bem como, envolver os clientes no

processo de engenharia de requisitos, como é o caso de Bellucci et al. (2015) que combinam XP com sessões de *co-design*, a fim de desenvolver um produto com forte envolvimento do usuário. Com base em suas descobertas, o protótipo deve ser desenvolvido iterativamente. No entanto, este trabalho não foi selecionado como principal referência para este estudo por tratar de forma específica a prototipação de produtos tangíveis e, portanto, não aplicável a produtos como *softwares* que é objeto desta revisão de literatura.

Harbers et al. (2015) estudaram a aplicação de *workshop* para o processo de elicitação de *User Stories*. O objetivo deste *workshop* é inserir os valores das partes interessadas no processo de engenharia de requisitos. Os resultados mostram que *User Stories* baseados em valores são considerados valiosos para o usuário final, mas complicados para estimar e testar. Este documento conclui, portanto, que este tipo de *workshop* é um método promissor para a incorporação de valores no processo de Engenharia de Requisitos, mas que os *User Stories* baseados em valores precisam ser traduzidos em Casos de Uso para torná-los adequados ao planejamento e organização das atividades de desenvolvimento.

Soares et al. (2015), reforça os achados de Harbers et al. (2015) mostrando que requisitos levantados por meio de *User Stories* trazem desafios como, por exemplo, a escrita apresentando descrição de alto nível dos requisitos do *software*, visando o cliente. Como consequência, os desenvolvedores consideram que, muitas vezes, estes requisitos são breves, vagos, ambíguos e insuficientes para capturar as complexidades do *design* inicial.

O artigo de Adikari et al. (2016) apresenta os resultados de dois projetos ágeis para validar a proposição de que incorporando uma perspectiva de *Design Centrado no Usuário* (UCD) no Desenvolvimento Ágil de *Software* melhora a qualidade da usabilidade e aceitação dos sistemas. Por meio da abordagem metodológica “*Design Science*”, os autores desenvolveram uma abordagem para integrar a perspectiva UCD no processo de desenvolvimento. Como solução prática, os autores propõem uma abordagem chamada “*Little Design Up Front* (LDUF)” - que fornece apenas detalhes necessários das informações do UCD para apoiar a análise e o *design* em iterações ágeis de modo que a perspectiva do UCD possa ser considerada sem sobrecarregar as práticas ágeis existentes. Os resultados mostraram que os usuários acham os produtos desenvolvidos por meio dessa abordagem mais fáceis de aprender, mais fáceis de usar e requerem menos suporte para poderem usar.

Na proposta do estudo de Saeeda et al. (2020) que propõe o uso do *framework* JRD e *Mind Mapping* para a elicitação de requisitos (já mencionado para a Q2), os autores também mencionam a necessidade de se criar um ambiente amigável desde o início para desenvolver a confiança do usuário final em fornecer todos os requisitos funcionais necessários em forma de

fala, para que a equipe de desenvolvimento possa entender melhor as necessidades e desejos do usuário. Os resultados foram validados com base em análises estatísticas que refletiram a utilidade do *framework* proposto, provando que os problemas de elicitação de requisitos no desenvolvimento de *software* ágil podem ser aprimorados com a aplicação de abordagens não-ágeis.

Um ponto congruente a ser considerado, provindo da análise dos estudos revisados, é o desafio em praticar a comunicação e colaboração contínuas como abordagens na tentativa de envolver as partes interessadas (não necessariamente o usuário final) no processo de elicitação de requisito compatível com métodos ágeis de desenvolvimento de *software*. Os resultados da literatura apontam ainda que a variedade de aspectos e métodos relatados pelos estudos revisados não direcionam para um modelo de processo comum para o envolvimento de partes interessadas em um ambiente ágil.

**(Q4) – Quais as implicações, para a indústria de *software* e para a comunidade acadêmica, dos atuais estudos que envolvem a ER em projetos ágeis?**

Os resultados levantam algumas questões que a academia deve investigar para melhorar as atuais práticas de engenharia de requisitos quando aplicadas em projetos que adotam metodologias ágeis, estimulando que mais pesquisas sejam feitas nessa área. Por exemplo, quais ajustes precisam ser feitos nas atuais técnicas utilizadas para elicitar requisitos em projetos ágeis? A adoção de práticas de engenharia de requisitos em projetos ágeis compromete a agilidade do processo? A qualidade da especificação em projetos ágeis é menor do que nos projetos que utilizam metodologias tradicionais?

Outra questão que merece atenção da comunidade acadêmica é a baixa qualidade dos artigos selecionados. O total a ser utilizado para a extração de dados era, inicialmente, de 32 artigos. Entretanto, durante a etapa de análise da qualidade, aproximadamente 22% desses artigos (7) foram excluídos, devido à baixa qualidade. Dos 25 artigos restantes, apenas 6 foram considerados de qualidade muito alta. Além disso, outros 3 artigos, cujo método de pesquisa foi Estudo de Caso, não consideraram de forma adequada a relação entre o pesquisador e as outras pessoas envolvidas na pesquisa. Desta forma, identifica-se um ponto de atenção no que diz respeito a redução do viés nos estudos.

Em relação ao campo empresarial/industrial e tendo como base os resultados apresentados, percebe-se que, mesmo com a adoção das metodologias ágeis, as empresas ainda continuam apresentando diversos desafios a serem enfrentados, relacionados principalmente com a participação e envolvimento do cliente e com a gestão de requisitos. De um total de 135

ocorrências, 44 estão relacionadas a Clientes, enquanto outros 37 problemas estão relacionados a gestão de requisitos. Isso sinaliza que as empresas necessitam analisar seus atuais processos de desenvolvimento, procurando identificar os gargalos que comprometem principalmente a satisfação do cliente e a qualidade dos requisitos.

### 3.9 CONCLUSÃO DA RSL

Respondendo à questão de pesquisa principal (Como a engenharia de requisitos tem sido conduzida em projetos que adotam metodologias ágeis?), inicialmente foram registrados 25 artigos que investigaram um total de 13 empresas.

Os resultados da revisão mostram que a análise de requisito na metodologia ágil tem sido estudada dentro de várias áreas de pesquisa (por exemplo, engenharia de *software*, fatores humanos ou *Design Participativo*) e, a partir destas análises, foi possível perceber que este é um importante tema de pesquisa. Além disso, a heterogeneidade dos aspectos estudados nos artigos que foram revisados apresenta que esta é uma área de pesquisa complexa e com muitas influências interfuncionais distintas. Por fim, foi possível identificar também que este campo de pesquisa está muito próximo das práticas de trabalho existentes nas empresas, uma vez que grande parte dos artigos incluídos para estudo realiza análise dos aspectos em um contexto da vida real, ou seja, a maioria das pesquisas é realizada por meio de estudos de casos.

De acordo com os estudos analisados, a técnica de Entrevista é a mais utilizada para a fase de Elicitação de Requisitos e a categoria de Clientes foi a que apresentou a maior quantidade de problemas (44), o que pode ser justificado pelo baixo fornecimento de ferramentas adequadas para envolver o cliente e meios adequados para gerenciar sua participação. O uso dessas práticas só foi relatado em 9 artigos.

Alguns dos estudos observados abordam a perspectiva do usuário no método ágil de desenvolvimento de *software* e contribuem com alguns insights sobre quais metodologias são úteis para tornar esta metodologia ágil mais voltada ao usuário. Os métodos mais percebidos nos artigos revisados que contribuem com a questão de pesquisa foram “*Human-Centered Design*”, “*Design Thinking*” e “*Participatory Design*”. No entanto, dentre estes mesmos estudos também é possível notar que existem problemas com o compartilhamento de conhecimento entre os *stakeholders* e usuários referente aos requisitos do usuário e a responsabilidade pela usabilidade. Estes achados fortalecem a necessidade fundamental de encontrar metodologias que sejam apropriadas e que contribuam para a resolução destes problemas.

Um ponto de limitação observado em concordância nos estudos selecionados consiste no desafio de criar uma boa comunicação entre os desenvolvedores e as partes interessadas e isto fortaleceu o objetivo desta pesquisa cuja pretensão está em investigar a partir da análise de um caso quais abordagens são usadas para envolver os usuários no processo de levantamento de requisitos no desenvolvimento de *software* e quais metodologias são comumente usadas para apresentar as perspectivas dos usuários.

Diante do exposto, a lacuna teórica identificada nas atividades de elicitação de requisitos ágeis ficou mais evidente e comprovada. Portanto, na aplicação em campo seguinte, mais evidências serão pesquisadas para ilustrar os desafios relatados pela literatura em relação ao envolvimento dos usuários nas atividades de elicitação de requisitos num contexto ágil.

Para entender como o trabalho de campo exploratório (piloto) e como o estudo de caso serão realizados, o próximo capítulo discute os métodos de pesquisa que serão utilizados, assim como as etapas de decisões metodológicas tomadas.

## 4 MÉTODO DE PESQUISA

Essa pesquisa classifica-se como exploratória, dado que o objetivo não é descrever ou identificar e explicar relações, mas sim compreender um dado fenômeno (YIN, 2015). A abordagem do problema é classificada como qualitativa, cujo objetivo consiste em examinar aspectos, em relação à sua interpretação em contextos específicos de investigação (KETOKIVI; CHOI, 2014) e refere-se a uma generalização analítica (DUBOIS; ARAUJO, 2007).

Segundo Yin (2005, p.20), o estudo de caso tem por objetivo “contribuir com o conhecimento que temos dos fenômenos individuais, organizacionais, sociais, políticos e de grupo, além de outros fenômenos relacionados”. Ainda segundo este mesmo autor, os estudos de caso se tornam uma estratégia coerente quando o objetivo de pesquisa visa responder questões “como” e “por que”, quando o pesquisador tem pouco controle sobre os acontecimentos e quando estes estão inseridos em contextos da vida real.

Neste sentido, o estudo de caso foi escolhido como método de pesquisa para avaliar a abordagem ágil uma vez que o contexto real em prática é essencial no processo de avaliação, considerando a observação das técnicas de elicitação dos requisitos, as dificuldades para envolver o usuário final, as razões dos pedidos de mudanças nos requisitos e a aceitabilidade do produto pelo grupo de usuários. Estas são variáveis que requerem a observação do uso da abordagem ágil em um cenário do mundo real durante alguns meses e envolvendo vários sprints. Portanto, não se pode avaliar estas variáveis facilmente através de experimentos controlados, alicerçando, desta forma, a escolha do método.

Apesar de, teoricamente, ser possível a aplicação desta pesquisa em qualquer projeto que utilize a abordagem ágil, optou-se por restringir a área de *software* devido a maioria dos estudos avaliados na RSL terem sido realizados em empresas dessa área e apresentarem lacuna para a problematização a ser estudada nesta pesquisa. Outra motivação se dá pela acessibilidade da pesquisadora a este ramo de atuação.

O estudo foi desenvolvido através do estudo de um caso prático e em tempo real, sendo acompanhado desde sua concepção até a finalização por meio de observações participativas em todos os processos e reuniões que se relacionarem com o objetivo da pesquisa, como a fase de elicitação e teste de aceite por parte dos usuários (homologação). O estudo contempla ainda uma fase de análise de documentos de outros projetos da mesma empresa desenvolvedora que já foram finalizados, cujo propósito consiste no levantamento de dados para comparação com o estudo em tempo real na discussão dos resultados.

A escolha da empresa para acompanhamento e análise do caso foi intencional e se deu por se tratar de uma unidade de negócio com mais de 20 anos de atuação na área de desenvolvimento de *software* e por gerenciar projetos por meio da abordagem ágil há mais de 10 anos. Outra motivação secundária se dá pela facilidade ao acesso que a pesquisadora teve à empresa, aos colaboradores, aos documentos, aos clientes e usuários dos produtos desenvolvidos.

O estudo de caso proposto por esta pesquisa acompanhou em tempo real de execução um projeto desenvolvido sob demanda para atender a necessidade de gerenciamento de entrega de pedidos de uma multinacional. Esta escolha passou antes por uma criteriosa seleção, como apresentado no transcorrer deste capítulo.

E, de forma a contribuir como guia para a investigação do objeto de campo, as proposições, extraídas de todo o referencial teórico analisado, são apresentadas na sequência.

#### 4.1 PROPOSIÇÕES

As proposições partiram das interpretações e críticas da literatura, vide capítulo 3, onde os achados confirmam a importância do envolvimento do usuário durante as atividades de elicitação de requisitos ágeis e sua relação com a aceitabilidade do produto, bem como os desafios relatados na tentativa de envolver estes usuários. Logo, propõe-se investigar no estudo de caso, se:

P1: O papel do usuário é envolvido no processo de elicitação de requisitos com apoio de ferramentas em uma adaptação da abordagem ágil no desenvolvimento de *softwares* empresariais sob demanda (ADIKARI et al., (2009); CALEFATO et al., (2012); SFETSOS et al., (2016); ORIOL, M. et al., (2020) e SAEEDA et al., (2020));

P2: A indisponibilidade de acesso ao usuário no processo de elicitação de requisitos em uma abordagem ágil no desenvolvimento de *softwares* empresariais sob demanda compromete a agilidade do projeto, se comparada à projeção de prazo estipulada. (RAMESH et al., (2010);

P3: Os desenvolvedores de *softwares* empresariais sob demanda, em uma abordagem ágil, têm dificuldade de compreensão dos requisitos quando estes são orientados ao usuário. (ANITHA et al., (2012); SOARES et al., (2015); MEDEIROS et al., (2020)) e;

P4: O papel do usuário é anulado e/ou subestimado pelo papel do cliente no processo de elicitação de requisitos na abordagem ágil de desenvolvimento de *softwares* empresariais sob demanda.

Aqui vale destacar que, para a proposição P4, nada referente foi encontrado durante a revisão da literatura, no entanto, foi incluída como sugestão para observação cautelosa durante o estudo de caso em função da experiência profissional da pesquisadora.

Definidas as proposições, as atividades seguintes consistem em apresentar o protocolo de pesquisa que, segundo Yin (1994), além de guiar às entrevistas, a validade e a confiabilidade dos dados do estudo de caso são fortalecidas por um protocolo de pesquisa bem elaborado. Para tanto, este trabalho buscou descrever um protocolo minucioso e claro que engloba, inclusive, sua aplicação experimental em um projeto piloto (Vide Apêndice C), onde aconteceu o aperfeiçoamento dos métodos e instrumentos da pesquisa.

## 4.2 PROTOCOLO DO ESTUDO DE CASO

A estrutura do estudo de caso foi definida pelas fases propostas por Yin (2009) e Voss (2002): mapeamento da literatura, seleção do caso, planejamento do caso e condução do estudo. As orientações principais para o estudo de caso, como os casos, os perfis entrevistados, o estilo de análise dos dados e o roteiro de entrevistas são descritas neste capítulo que engloba as regras e os procedimentos a serem seguidos ao se utilizar este instrumento.

Em uma adaptação de Yin (2001), este estudo descreve como busca se ater às validades de modo a garantir o rigor e a robustez do estudo de caso:

- a) Validade Interna: Testada por meio da construção da narrativa, mostrando o grau com que o estudo mapeia o fenômeno estudado;
- b) Validade Externa: Testada através da análise de vários projetos e comparação dos resultados encontrados com casos descritos na teoria;
- c) Validade do construto: Testada por meio da revisão e aprovação dos relatórios sobre o estudo pelos entrevistados e análise de várias fontes de informação;
- d) Confiabilidade: Testada através da definição de um protocolo de estudo de caso e transcrição dos casos.

### 4.2.1 Escolha do caso

Uma vez escolhida a empresa mantenedora dos casos, o projeto para observância em tempo real foi eleito de forma a atender aos requisitos:

- Houver apenas a real possibilidade de uma proposta comercial de modo a garantir que nenhum processo de elicitação tenha sido iniciado;

- A realização do projeto deve acontecer por meio da abordagem ágil, de acordo com a definição apresentada no Quadro 1;
- Houver aceite por parte do cliente em relação ao acompanhamento do pesquisador durante o desenrolar do projeto;
- Houver distinção entre os papéis de cliente e usuário;
- Projeto com prazo de curta duração (até 2 meses), a fim de viabilizar a entrega do resultado desta pesquisa como parte da dissertação de mestrado.

### **4.2.3 Procedimento de coleta de dados**

A coleta de dados é então realizada durante a interação com o objeto de estudo por meio de entrevistas semiestruturadas, observações participativas e análise de documentos. Vale ressaltar que o acompanhamento se dá na forma de expectador, com a participação da pesquisadora nas discussões somente quando permitido. O desenho da pesquisa pode ser visualizado na Figura 9.

As entrevistas semiestruturadas são realizadas com os colaboradores designados pela empresa, satisfazendo os critérios especificados neste protocolo, tendo por objetivo a coleta de informações por meio de um roteiro pré-determinado, mas também se valendo de toda flexibilidade permitida pelo método de estudo de caso.

Além das entrevistas, são realizadas observações participativas (inclusive reuniões) e consulta a documentos da empresa (projeto de acompanhamento e projetos anteriores). Estas práticas aconteceram no período de novembro de 2020 a maio de 2021<sup>4</sup>.

As observações participativas ocorrem em todas as interações na fase de elicitação dos requisitos a fim de manter maior proximidade nesta etapa do projeto onde se concentra a problematização desta pesquisa. Outra observação participativa desenrola-se durante o Teste de Aceite, onde o cliente e usuário tem contato com o produto acabado para certificar se o que foi entregue atende as reais necessidades originadas e ainda, se estas necessidades foram percebidas e elicidadas pelos envolvidos na fase inicial.

No que diz respeito a análise documental, o acesso a requisitos do projeto em acompanhamento e de projetos anteriores contribui para o entendimento sobre a forma como as necessidades dos usuários são percebidas e documentadas e ainda, o acesso aos registros de reporte de falhas ou solicitações de mudanças provenientes da parte do cliente contribuem para

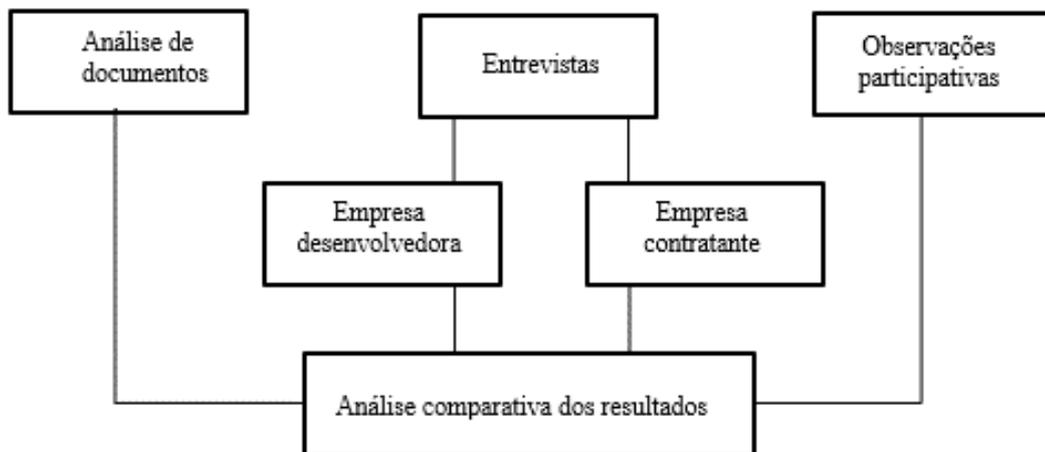
---

<sup>4</sup> Projeção para encerramento do projeto era abril/2021. No entanto, a data foi adiada em função das adversidades causadas pela pandemia da COVID-19.

entender o volume de discrepâncias entre a necessidade real dos usuários durante o uso do *software* no contexto do trabalho e a solução entregue pelo *software* desenvolvido pela empresa contratada. Para este instrumento, é necessário um tempo de estudo para seleção criteriosa dos documentos a serem considerados, bem como para o aprendizado de ferramentas usadas para o gerenciamento do projeto de modo a ganhar tempo durante a fase de execução em si.

Após a análise dos dados coletados, um relatório com as informações consolidadas é redigido e, posteriormente, aprovado pelos membros da organização estudada.

**Figura 8** – Desenho da pesquisa

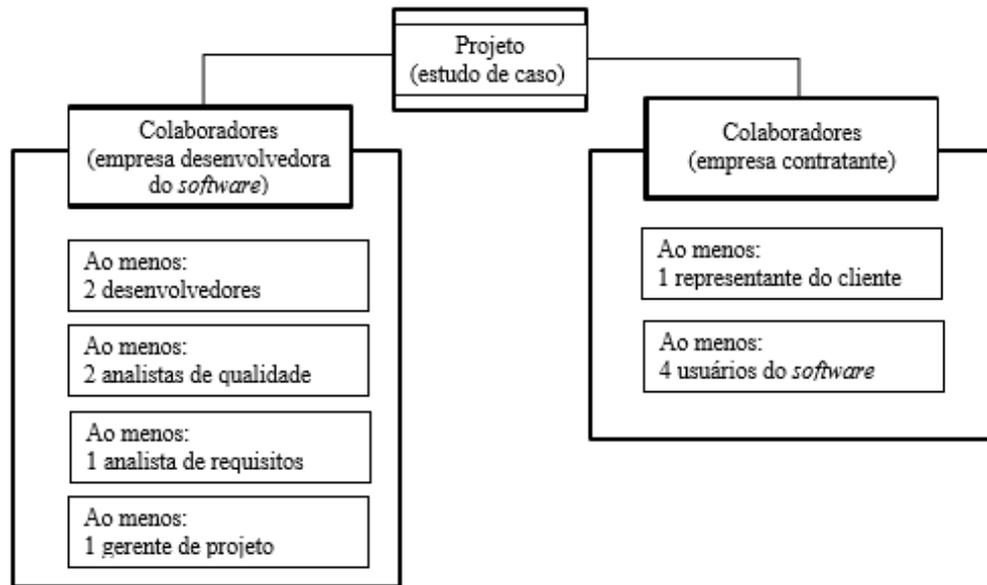


Fonte: Elaboração própria.

#### 4.2.3 Escolha dos perfis entrevistados

Primeiramente, realiza-se uma entrevista com os responsáveis pelo projeto da empresa e com os envolvidos nas atividades de elicitação de requisitos, a fim de entender os processos e práticas utilizadas pela organização. Posteriormente, com o caso de estudo já selecionado, as entrevistas semiestruturadas são conduzidas com múltiplos respondentes para levantamento dos dados do projeto, de forma a obter diferentes percepções acerca do fenômeno (Voss et al., 2002). Os respondentes são membros da equipe do projeto foco do estudo e membros da empresa contratante, como na Figura 8.

**Figura 9** – Estrutura do caso e amostra



Fonte: Elaboração própria.

Desta forma, pretende-se obter uma variedade de percepções que contribuem para o esclarecimento dos objetivos desta pesquisa e garante a validade do estudo.

#### 4.2.4 Procedimento de análise dos dados

Tendo finalizada as entrevistas, os dados coletados foram então transcritos em formato de uma narrativa, redigindo a história observada do caso e os pontos considerados relevantes. Por meio dessa transcrição é possível realizar a análise, em profundida, de cada fonte de informação e, posteriormente, a realização de uma análise comparativa entre elas. Essas análises são então confrontadas com a literatura referente às práticas e desafios da Elicitação de requisitos ágeis.

#### 4.2.5 Estrutura da entrevista

O roteiro para as entrevistas semiestruturadas está dividido em 7 partes.

Parte 1: Dados sobre o entrevistado e a entrevista.

O pesquisador preenche o máximo possível destas informações antes da entrevista, de modo a ganhar tempo durante a entrevista para abordar a questão principal. Os dados do entrevistado, da organização que ele pertence e algumas informações sobre a entrevista são registrados conforme no quadro 14.

Optou-se por manter a identidade do entrevistado em sigilo e, em momento algum este será associado nominalmente à pesquisa. Além da palavra do pesquisador, é fornecido ao entrevistado um termo de compromisso que formaliza este compromisso. Esta opção permite que o entrevistado se sinta em uma posição mais confortável e segura e isto deu a eles maior liberdade para falar.

**Quadro 14** – Dados da entrevista e do perfil do entrevistado

<b>Entrevistador:</b> Ana Paula Porto Serafim		
<b>Sobre o Entrevistado</b>		
<b>Nome:</b>	<b>Formação:</b>	<b>Cargo/Função:</b>
<b>Tempo na Função:</b>	<b>Tempo de trabalho com a Metodologia Ágil?:</b>	
<b>Sobre a Entrevista</b>		
<b>Local:</b>	<b>Tipo Entrevista:</b> Semiestruturada	<b>Data:</b>
<b>Duração Estimada:</b> 30 min	<b>Forma de Registro:</b> Anotação	

Fonte: Elaboração própria.

Parte 2: Informações apresentados ao entrevistado (sobre a pesquisa).

Nesta parte o entrevistado toma nota sobre a pesquisa de modo a ser contextualizado sobre os objetivos principais, bem como as pretensões de contribuições nos âmbitos acadêmicos e empresariais, conforme apresentado no Quadro 15.

Tendo preenchido o formulário, uma cópia é fornecida ao entrevistado que, posteriormente, pode entrar em contato com o pesquisador, por algum motivo, através dos dados inclusos nele.

**Quadro 15** – Dados da entrevista explicitadas ao entrevistado

<b>Sobre o Pesquisador</b>	
<b>Nome:</b> Ana Paula da Silva Porto	
<b>Formação:</b> Administração	<b>Contato:</b>
<b>Orientador:</b> Prof. Dr. Sérgio da Silva	Universidade Federal de São Carlos/Dep. Eng. Produção <a href="http://www.ppgep.dep.ufscar.br/">http://www.ppgep.dep.ufscar.br/</a> contato:ppgep@dep.ufscar.br
<b>Sobre a Pesquisa</b>	
<b>Linha de Pesquisa:</b> Qualidade	
<b>Objetivos da pesquisa:</b> Entender as técnicas de envolvimento do usuário no projeto.	
<b>Etapa atual:</b> Pesquisa de campo (estudo aprofundado por meio da avaliação de casos reais)	
<b>Contribuições da pesquisa:</b> Propostas de abordagens para maximizar a aceitabilidade dos <i>softwares</i> por parte dos usuários.	

Fonte: Elaboração própria.

Parte 3: Algumas informações no âmbito geral para orientar o pesquisador

Estas informações orientam o pesquisador antes, durante e depois da entrevista, como um preparo para as entrevistas por meio de um *checklist*, conforme mostra o Quadro 16.

**Quadro 16** – Orientações para a entrevista

<b>Antes da entrevista:</b>
Reservou tempo adequado para a entrevista (30 min aprox.)?
Disponibilizou link de reunião para o entrevistado?
Confirmou se o entrevistado tem acesso à ferramenta usada para a reunião online?
<b>Durante a entrevista:</b>
Não serão anotados apenas o que se ouve, mas também o que se vê, pois pode perceber informações não verbais como, por exemplo, nervosismo, dúvida etc.
Apresentar as orientações gerais ao entrevistado, bem como a definição de termos.
Estar atento no que se refere à imprecisão nas informações em função da memória falha do entrevistado ou ainda, respostas viesadas, discurso pronto etc.
Nos casos em que o entrevistado não quiser responder alguma questão, será solicitado para que tente uma resposta em âmbito geral para que não chegue a especificidade de questões sigilosas.
<b>Após a entrevista:</b>
Repassar toda a entrevista e fazer anotações adicionais e buscar já fazer interpretações sobre os dados coletados a fim de corrigir quaisquer erros.
Terminar de preencher o protocolo, caso houver pendências.
Transferir o conteúdo da entrevista para um formato digital.
Elaborar texto com as interpretações e enviar ao entrevistado para a avaliação final.
Orientar para caso, após a entrevista, o entrevistado queira fazer algum adendo em alguma de suas respostas, este poderá fazer por meio de contato telefônico ou e-mail.
Solicitar ao entrevistado críticas, sugestões ou comentários sobre a forma como a entrevista foi conduzida.
Solicitar que o entrevistado sugira outras pessoas (de dentro da sua organização ou não) que poderiam contribuir com a pesquisa.
Combinar uma avaliação posterior dos dados coletados na entrevista e, principalmente, das interpretações feitas pelo pesquisador.
Dar ao entrevistado uma estimativa sobre o encerramento da pesquisa e mencionar a forma como os dados serão divulgados (no caso, dissertação).

**Fonte:** Elaboração própria.

#### Parte 4: Orientações gerais ao entrevistado a respeito da entrevista

Nesta parte do protocolo são incluídas as orientações e regras que devem ser seguidas pelos entrevistados, vide quadro 17. No entanto, outras questões/dúvidas trazidas pelos entrevistados também podem ser esclarecidas antes da entrevista.

**Quadro 17** – Orientações gerais para o entrevistado antes da entrevista

<b>Orientações gerais ao entrevistado:</b>
Os questionários não possuem questões fechadas, ou seja, não oferecem alternativas como resposta.
Será bem aceita também a conversa de forma espontânea, com a liberdade para usar seu conhecimento, experiência e opinião para acrescentar tópicos que julgar importante.
Por se tratar de uma entrevista não gravada, o pesquisador poderá solicitar ao entrevistado que repita alguns trechos para que possa fazer a correta anotação.
A entrevista deve durar em média 30 minutos, mas se o entrevistado estiver de acordo, poderá haver prorrogações para melhor coleta das informações.

**Fonte:** Elaboração própria.

#### Parte 5: Apresentação dos termos utilizados nos questionários

Nesta parte, os termos contidos nas questões ou nas possíveis respostas que podem provocar dúvidas ao entrevistado são esclarecidos, pois acredita-se que os casos de dúvidas por

parte do entrevistado trazem situações indesejáveis como interrupções que podem ser evitadas, bem como afetar a linha de raciocínio de ambos; ou ainda, o entrevistado, mesmo com a dúvida, pode não questionar ao pesquisador e responder à questão de forma errada. Os termos e suas definições podem ser vistos no Quadro 18.

**Quadro 18** – Orientações gerais sobre os termos empregados na entrevista

<b>Termos:</b>
Usuário: Papel daquele que irá usufruir do serviço prestado, também rotulado como usuário final.
Cliente: Embora seja comumente confundido com usuário, este papel é bem diferente, pois exerce uma responsabilidade tática e estratégica dentro do processo de desenvolvimento de <i>software</i> . O cliente é quem compra o produto, negocia, realiza acordos e fornece os requisitos para o serviço que é entregue, assim como é responsável por avaliar os resultados do serviço.
Participação do usuário: Garante apenas que os usuários sejam acessados de forma consultiva para testar e validar protótipos em fases finais do desenvolvimento de <i>software</i> .
Envolvimento do usuário: Garante que os usuários estejam engajados nas atividades de desenvolvimento do novo <i>software</i> e assumam a autoria para que o produto atenda a todas as suas necessidades.
Elicitação: Compreende o conjunto de atividades que o analista de requisitos emprega para descobrir as solicitações dos usuários, determinando as reais necessidades por trás das solicitações.
Falha: É quando um <i>software</i> não cumpre seu objetivo em relação às necessidades originais.

**Fonte:** Elaboração própria.

#### Parte 6: Seção relacionada aos questionários

Esta seção apresenta o questionário guia a ser utilizado durante as entrevistas.

As questões foram concebidas de modo a englobar a observação das 4 proposições já descritas neste capítulo. A primeira observação visou identificar se o papel do usuário é envolvido no processo de elicitação de requisitos ágeis seguida da observação em relação a indisponibilidade de acesso ao usuário neste mesmo processo e sua relação com a agilidade do projeto. As questões visam ainda coletar informações referentes aos desenvolvedores e a dificuldade de compreensão dos requisitos quando estes são orientados ao usuário e, por fim, busca de forma cautelosa, pois não se encontrou evidência na literatura, entender se o papel do usuário é anulado e/ou subestimado pelo papel do cliente no processo de elicitação de requisitos ágeis.

As questões foram organizadas em função das fases do projeto, bem como de acordo com os perfis entrevistados. O Quadro 19 apresenta esta relação de questões x fases x perfis. Para complementar, o Quadro 20 apresenta a descrição das questões e suas conexões com as proposições a serem observadas.

**Quadro 19** – Mapeamento das questões para as entrevistas

	EMPRESA DESENVOLVEDORA				EMPRESA CLIENTE	
	DESENV	QA	LP	AR	CLIENTE	USUÁRIO
ANTES			Q1	Q1		Q3*
			Q2	Q2		
DURANTE	Q1	Q1	Q4	Q4	Q4	Q4
	Q4	Q4	Q5	Q5	Q5	Q5
	Q5	Q5	Q6	Q6	Q6	Q10
	Q7	Q7	Q7	Q7	Q7	
	Q8	Q8	Q8	Q9	Q9	
	Q9	Q9	Q9			
HOMOLOGAÇÃO	Q13	Q13	Q11	Q12	Q11	Q10
	Q14	Q14	Q12	Q13	Q12	Q13
			Q13	Q14	Q13	Q14
			Q14		Q14	Q15
						Q16
						Q17

\*Desconsiderar as questões Q4, Q5 e Q10 (durante), caso a resposta seja ‘Não’.

**Legenda:**

Desenv.: Analista desenvolvedor

QA: Analista de Qualidade

LP: Líder de Projeto, dependendo da metodologia ágil usada este papel pode receber títulos distintos.

AR: Analista de Requisito, principal responsável pela elicitação.

Cliente: Neste projeto, esta figura inclui também o P.O, já que um membro do cliente assumiu esta função.

Fonte: Elaboração própria

**Quadro 20** – Questionário a ser aplicado nas entrevistas (continua).

	QUESTÕES	PROPÓSITO
Q1	Já trabalhou com este cliente? Se sim, o projeto visava atender os mesmos usuários?	Busca observar/analisar se o envolvimento dos usuários está influenciado pelo histórico de projetos, impactando a P1.
Q2	Qual abordagem será usada para entender as necessidades deste projeto?	Em observação à P1, busca conhecer se fazem uso de ferramentas que apoiam o processo de elicitação.
Q3	Você tem conhecimento de um <i>software</i> que está sendo construído para auxiliar nas suas atividades atuais? Se sim, como tomou conhecimento do projeto do novo <i>software</i> ?	Questão de impacto para P1 e P4 trazendo informações sobre o grau de envolvimento dos usuários.
Q4	Descreva, em sua perspectiva, qual o objetivo do projeto e quais foram as motivações para o projeto acontecer?	Informação importante na análise comparativa a fim de identificar a conformidade ou discrepância entre as respostas e a real necessidade do projeto para o usuário (P1 e P4).

	QUESTÕES	PROPÓSITO
Q5	O projeto trata de novas funcionalidades ou atualização para as que já existem?	Busca compreender se o envolvimento dos usuários é anulado pelo cliente em função do tipo do projeto, impactando a P4.
Q6	Em algum momento o usuário foi consultado, mas estava indisponível/ocupado, dificultou o acesso ou demorou no retorno?	Informação de forte impacto para a P2, pois de acordo com Ramesh, Cao & Baskerville (2010) envolver os usuários na Engenharia de Requisitos usando abordagens ágeis para o desenvolvimento de <i>software</i> depende de muitos fatores, tais como disponibilidade destes, especialmente nas fases iniciais do projeto.
Q7	Quem serão os usuários do <i>software</i> ? Você os conhece?	Questão de impacto para P1 e P4 trazendo informações sobre o grau de envolvimento dos usuários.
Q8	Em algum requisito houve dificuldade para compreender o que se pedia?	Esta questão faz-se necessária para validação da P3 e confirmação dos achados na pesquisa de Medeiros, J. et al. (2020) indicando que os requisitos na abordagem ágil (considerando o forte envolvimento do usuário) são insuficientes e inadequados para serem usados pela equipe de desenvolvimento, tornando o desenvolvimento mais difícil em um estágio posterior.
Q9	Em situações de dúvidas/problemas em relação aos requisitos do projeto, qual meio usado para buscar resposta/solução?	Questão de impacto para P1, P2 e P4 trazendo informações sobre o grau de envolvimento dos usuários, bem como de sua disponibilidade de acesso.
Q10	Você foi consultado a respeito deste novo <i>software</i> ?	Questão de impacto para P1, P2 e P4 trazendo informações sobre o grau de envolvimento dos usuários, bem como de sua disponibilidade de acesso.
Q11	Como são colhidas e tratadas as percepções dos usuários?	Informação de forte impacto para a P4, pois busca identificar se os usuários são consultados ou apenas o cliente é contatado.
Q12	Os usuários parecem ter entendido a proposta do <i>software</i> ?	Questão de impacto para P1 e P4 uma vez que, segundo Adikari et al. (2016) e Steftos et al. (2016), a compreensão do <i>software</i> está relacionada ao grau de envolvimento destes no levantamento dos requisitos.
Q13	Você considera este um projeto de sucesso?	Questão de impacto para P1 e P4 uma vez que, segundo Adikari et al. (2016) e Steftos et al. (2016), o aceite do <i>software</i> está relacionado ao grau de envolvimento destes no levantamento dos requisitos.
Q14	Houve solicitações de adequação?	Questão de impacto para P1, P3 e P4 uma vez que um alto volume de adequações pode ser resultado do baixo envolvimento dos usuários e/ou na dificuldade de compreensão dos requisitos por parte dos desenvolvedores.
Q15	Está confortável com a proposta do novo <i>software</i> ?	Questão de impacto para P1 e P4 uma vez que, segundo Adikari et al. (2016) e Steftos et al. (2016), os usuários tendem a familiarizar com o <i>software</i> quando envolvidos no levantamento dos requisitos.
Q16	Você consegue utilizar o <i>software</i> por completo e este atende à todas as necessidades da tarefa?	Questão de impacto para P1 e P4 uma vez que o envolvimento dos usuários no levantamento dos requisitos contribui para que o <i>software</i> seja mais próximo de suas reais necessidades.
Q17	Haveria sugestão de melhoria/adequações?	Questão de impacto para P1, P3 e P4 uma vez que um alto volume de adequações pode ser resultados do baixo envolvimento dos usuários e/ou na dificuldade de compreensão dos requisitos por parte dos desenvolvedores.

Fonte: Elaboração própria

#### Parte 7: Termo de compromisso

Conforme mencionado na parte 1 deste protocolo, deverá ser entregue ao entrevistado um termo onde o pesquisador se compromete formalmente com o compromisso de manter em sigilo a identidade do entrevistado.

Termo de compromisso: “Eu, Ana Paula da Silva Porto, brasileira, casada, administradora, portador do RG n.º XX.XXX.XXX SSP/SP, CPF n.º XXX.XXX.XXX-XX, residente e domiciliado à Avenida Pedro II, 09, Jardim Santa Clara, na cidade de Mococa, SP, venho através do presente termo, comprometer-me a não associar ou relacionar, direta ou indiretamente, de forma escrita, verbal ou de qualquer outra forma, o nome ou a identidade de (nome completo do entrevistado) à minha pesquisa de mestrado iniciada em 2020 e ligada ao Programa de Pós-Graduação do Departamento de Engenharia de Produção da Universidade Federal de São Carlos, seja durante e/ou após a realização da mesma.”

Por se tratar de um método de pesquisa flexível, como é o estudo de caso em tempo real, o protocolo gerado não foi definitivo desde o início, isto é, foi modificado e reformulado com o decorrer da pesquisa, conforme prega a abordagem qualitativa adotada. O projeto piloto, descrito no Apêndice D, por meio da reflexão sobre a dinâmica da entrevista, contribuiu para a reorganização de algumas perguntas para que ela pudesse fluir melhor, e incorporou novos questionamentos considerados pertinentes.

## 5 ESTUDO DE CASO

O estudo contemplou o caso do desenvolvimento e implantação do sistema ‘Tracking de Pedidos’ elaborado pela empresa DESENV<sup>5</sup> para uma grande rede varejista que atua em âmbito nacional.

### 5.1 A EMPRESA

A empresa mantenedora do caso a ser estudado situa-se no interior de São Paulo, atuando no mercado de desenvolvimento de *softwares* há 25 anos e aplicando a abordagem ágil na gestão de seus projetos desde 2010. Já somaram mais de mil projetos implantados e mais de 350 clientes atendidos no Brasil e exterior.

Nesta empresa, encontrou-se espaço para estudar em tempo real um projeto e, com a intenção de conhecer sobre as práticas comumente utilizadas pela empresa em relação a fase de elicitação de requisitos, a pesquisadora então entrou em contato com os analistas de requisitos e obteve como respostas as técnicas pormenorizadas a seguir, no Quadro 21.

**Quadro 21** – Técnicas de elicitação utilizadas pela empresa desenvolvedora.

<b>Técnica</b>	<b>Descrição</b>
<b>Observação pessoal</b>	Constitui-se da observação, pelo analista, do local onde será implementado o <i>software</i> , ou que poderá ser afetado pelo seu produto. A observação é realizada com o intuito de obter informações sobre o ambiente, tornando possível determinar as rotinas de trabalho dos usuários, seus problemas e erros de procedimentos, entre outros aspectos. Mais que isso, a observação do ambiente de trabalho dos usuários do <i>software</i> a ser desenvolvido permite a explicitação de detalhes sobre procedimentos operacionais que não tenham sido informados pelos usuários quando da aplicação de questionários e entrevistas.
<b>Entrevista</b>	Utilizada pelos analistas com o objetivo de melhor entender as necessidades dos usuários. Para isso, o analista define os objetivos, o número de quesitos, a duração da entrevista, quem serão as pessoas entrevistadas (tentando sempre envolver o usuário, mas nem sempre é possível), o horário das entrevistas, quais serão os assuntos tratados, entre outras informações julgadas úteis.
<b>Seminário</b>	Utilizado para reunir informações a respeito do sistema solicitado. Normalmente, os analistas obtêm informações gerais reunindo pessoas chave em um mesmo ambiente e procuram, com a interação de todos, extrair delas suas necessidades.
<b>Pesquisa (busca)</b>	Busca reunir informações a respeito do sistema que será desenvolvido por meio de pesquisas por atividades e/ou processos passados. Tais buscas dão subsídios para que os analistas possam identificar volumes e ocorrências, bem como indicar prazos.
<b>Técnica mista</b>	Compreende um conjunto de técnicas empregadas concomitantemente com o objetivo de reunir as vantagens e eliminar as desvantagens de cada uma das técnicas das quais é composta.

**Fonte:** Elaboração própria.

<sup>5</sup> Nome fictício dado à empresa desenvolvedora do *software* que é o objeto deste estudo.

Tendo em mente as técnicas utilizadas pela empresa para elicitação de requisitos, a pesquisadora dá início à primeira fase da coleta de dados, ou seja, as entrevistas planejadas para o período ANTES do início do desenvolvimento e, simultaneamente, dá-se início à análise de documentos disponibilizados pela empresa DESENV relacionados à projetos encerrados antes do início deste estudo. Para tanto, as descrições pormenorizadas destas fases estão contidas no texto que segue detalhando o caso estudado.

## 5.2 ANÁLISE DE DOCUMENTOS

A fase de análise documental concentrou-se na investigação de documentos pertencentes à projetos de *software* sob demanda já finalizados pela empresa DESENV e que foram gerenciados sob a metodologia da abordagem ágil, respeitando a disponibilidade de acesso concedida pela empresa. Foram analisados, portanto, documentos de três projetos de clientes distintos, desenvolvidos entre os anos de 2018 e 2019. Neste estudo, os projetos receberam os nomes de Pj1, Pj2 e Pj3 para facilitar o agrupamento e discussão dos achados. Os documentos analisados, bem como as informações observadas em cada projeto estão reunidos no Quadro 22.

**Quadro 22** – Documentos e informações analisadas nos projetos.

<b>Documentos</b>	<b>Finalidade</b>
Atas de reunião com cliente para elicitação de requisitos (registradas via e-mail)	Verificar a frequência com que o perfil cliente foi consultado formal e informalmente para elicitação dos requisitos e esclarecimento de dúvidas.  Verificar a frequência com que os usuários estiveram em contato com os analistas de requisitos (para constar se estes foram envolvidos no processo de entendimento das necessidades).
Plataforma Jira	Verificar a quantidade de vezes que o serviço de manutenção foi acionado pelo cliente em período de homologação ou após a garantia.
E-mails trocados entre DESENV e Cliente pós-período de garantia	Verificar se o serviço de manutenção foi acionado após o período de garantia do produto.

**Fonte:** elaboração própria

Vale ressaltar que os e-mails trocados com o cliente por parte da equipe de analistas de requisitos estiveram disponíveis para consulta, uma vez que, no gerenciamento ágil nem sempre se formaliza uma reunião em atas. E, ainda neste contexto, os analistas também se disponibilizaram para contextualizar a pesquisadora acerca do projeto e responder a eventuais curiosidades.

As atas de reuniões não recebiam necessariamente esta nomenclatura nos e-mails, mas continham os ponto-chave das reuniões, ou seja, os envolvidos, os assuntos discutidos e as soluções acordadas, bem como, itens pendentes para outra reunião. Outros e-mails cujo remetente ou destinatário era o analista de requisitos também ficaram sob análise da pesquisadora por entender que se tratava de contatos para esclarecimento de dúvidas e, portanto, pertenciam à fase de elicitação. Com isso, foi possível observar a frequência de contato com o cliente na fase de elicitação e o grau de envolvimento do usuário nesta fase.

Em relação à plataforma Jira, foram observados os “chamados”, conceito este utilizado pela empresa para definir toda e qualquer solicitação de serviço de manutenção que pode ser de dois tipos: corretivas e evolutivas. Um chamado se classifica como manutenção corretiva quando se trata de um erro do sistema, ou seja, comportamentos inadequados do *software* desencadeados por erros na escrita do código ou por falta de entendimento dos requisitos escritos/repassados ao desenvolvimento. Já manutenção evolutiva corresponde a solicitação de desenvolvimento de novas funcionalidades, adequação de funcionalidades existentes ou implantação de melhorias ao sistema.

Os três projetos analisados foram gerenciados pela abordagem SCRUM e utilizaram técnicas mistas para a elicitação de requisitos, incorporando as observações pessoais por parte dos analistas e entrevistas, exceto o projeto Pj3 que utilizou exclusivamente a técnica de entrevistas, apenas. Estes processos geraram artefatos como: Visão ilustrada do produto, *Product Backlog*, *Backlog Sprints*, Especificações Funcionais (*Stories*) e Controle de Incidências (Jira).

O Projeto Pj1 tratou do desenvolvimento de *software* para o controle de viagens de funcionários de uma multinacional, onde controlava-se os reembolsos financeiros, tickets ofertados, descontos concedidos, hospedagens, passagens aéreas, trajetos (itinerário), períodos de viagens. Teve duração de 3 meses e 23 dias (desde o primeiro contato com cliente para fechamento da proposta até a entrega do produto para homologação final do cliente).

O Projeto Pj2 buscou desenvolver uma plataforma para oferta de produtos de uma seguradora (automóveis e residenciais) também multinacional envolvendo questões como cadastro e edição de categorias de seguro, franquias, prêmios, sinistros, beneficiários e rotina de cálculos.

Já o Projeto Pj3 desenvolveu um *software* para impressão de etiquetas para uma editora nacional, onde o objetivo consistiu em gerar etiquetas com informações completas configuradas pelo usuário para que pudessem ser transmitidas via *software* para a máquina de impressão que desencadeava o processo (maquinário) de etiquetagem.

Os projetos Pj2 e Pj3 tiveram, em média, a duração de 5 e 2 meses, respectivamente. Para estes casos, não foi possível mensurar o período exato, pois ambos tiveram seu início de forma não convencional, ou seja, as atividades iniciais foram executadas por desenvolvedores da DESENV que estavam alocados na empresa cliente e, estes, dividiam suas rotinas de trabalho entre atividades de projetos anteriores<sup>6</sup> e atividades do novo projeto aproveitando qualquer ócio na tentativa de ganhar vantagem em relação ao prazo estimado, portanto, a estimativa baseia-se nos primeiros e-mails encontrados entre desenvolvedor alocado e empresa cliente cujo assunto dizia respeito ao projeto em questão.

Para facilitar a compreensão dos achados destes três projetos, a Quadro 23 ilustra os dados organizados por projetos e suas categorias.

**Quadro 23** – Dados obtidos por meio da análise documental de projetos finalizados.

Projetos	Informações coletadas						
	Abordagem	Duração (aprox.)	Contexto	Qtde reunião c/ cliente	Qtde interação c/ usuário	Chamados ao Serviço de Manutenção	
						Evolutivas	Corretivas
Pj1	SCRUM	4 meses	Controle de viagens de colaboradores	8	5	1	10
Pj2		5 meses	Seguros residenciais e de automóveis	12	3	4	5
Pj3		2 meses	Impressão de Etiquetas - Editora	5	0	0	2

**Fonte:** Elaboração própria

Em síntese, o projeto Pj1 com duração de 4 meses, aproximadamente, teve reuniões semanais com os clientes para a elicitação dos requisitos, sendo que apenas nas três últimas os usuários não participaram e, nestas, foram discutidos temas como layout e design do *software*. Muito embora não seja possível uma análise aprofundada do caso, uma vez que não se obteve dados para esta amplitude de estudo, mas a única alteração solicitada pós homologação tratou-se exatamente de questões de usabilidade sugerindo que a participação dos usuários nestas fases finais poderia ter evitado estes desentendimentos quanto às reais necessidades neste quesito usabilidade.

Já para o projeto Pj2, este foi o que teve maior número de reuniões, no entanto sua duração foi de 5 meses e as reuniões aconteciam de forma mais espaçadas, em média a cada 15

<sup>6</sup> Clientes dos projetos P2 e P3 já haviam contratado a empresa DESENV em períodos anteriores para projetos distintos.

dias. Este projeto contou com a participação ativa do usuário apenas nas três primeiras reuniões. De acordo com a leitura de e-mails trocados entre DESENV e empresa Pj2, ficou evidenciado que a não participação dos usuários nas demais reuniões se dava pela indisponibilidade deles, uma vez que não podiam se ausentar das rotinas diárias do trabalho. Para este projeto foram solicitadas quatro alterações, sendo duas de baixa complexidade (relacionadas ao layout) e outras duas de alta complexidade relacionadas ao tratamento de exceções para determinados fluxos de contratação de seguro. Estas alterações foram levantadas pelos usuários após a primeira semana de uso do *software*, já instalado no cliente e dentro do prazo de garantia.

O projeto Pj3 foi o de menor duração e menor número de reuniões com o cliente, bem como nenhuma interação com o usuário. No entanto, não foram observados pedidos de alterações no *software* após a homologação ou em período de uso. Um fator importante observado para este caso é que o representante do cliente, o gerente de produção, antes de ocupar esta posição de gerência, foi executor no setor onde o *software* iria operacionalizar, sugerindo que esta situação possa ter beneficiado a compreensão das necessidades dos usuários, uma vez que conhecia as rotinas e particularidades da área.

As análises dos achados nesta fase estão contidas no capítulo de Resultados e Discussões cruzando-as com os dados coletados por meio das entrevistas e levantamentos extraídos da observação participativa da pesquisadora durante o acompanhamento em tempo real do desenvolvimento de um *software* empresarial sob demanda. Embora cada fase tenha seu tempo para execução determinado no planejamento desta pesquisa, as observações participativas aconteceram, inicialmente, em paralelo a fase de análise de documentos e todos os dados relevantes para este estudo estão descritos em detalhes no texto que segue.

### **5.3 O CASO PARA OBSERVAÇÕES PARTICIPATIVAS E ENTREVISTAS**

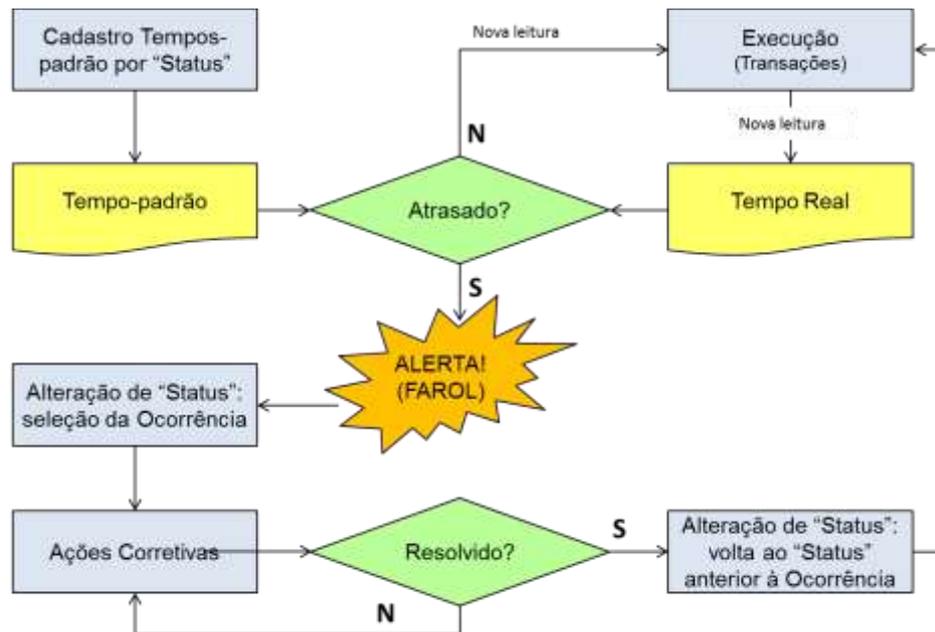
O caso estudado envolve o desenvolvimento de um *software* cujo objetivo concentrou-se em buscar melhorias no acompanhamento e gerenciamento de pedidos no processo logístico de:

- Distribuição dos produtos entre as lojas (via transferências entre Centros de Distribuição e lojas);
- Coleta (via devolução de clientes);
- Entregas de compras realizadas *online*.

Para todos os rastreios (processos logísticos) possíveis, as informações são advindas de transportadoras, Correios ou outros aplicativos da própria empresa (cujo nomes não podem ser

citados, pois fazem referência ao nome comercial da empresa que está sob sigilo nesta pesquisa). As informações geradas pelo *software* servirão de apoio gerencial e base de dados para pesquisa do SAC (Serviço de Atendimento ao Cliente). Um resumo, elaborado pela equipe de desenvolvimento, pode ser apreciado através da Figura 10.

**Figura 10** – Escopo do projeto (visão) desenhado pelos análises de requisitos do caso em estudo.



**Fonte:** Elaborado pela equipe do projeto (2021).

Na ocasião da escolha do caso, o projeto em questão ainda era uma solicitação de proposta comercial onde nenhum processo de elicitação havia sido iniciado. No entanto, pelo primeiro contato com o cliente já era possível identificar claramente a distinção entre os papéis de cliente e usuário de tal forma que dava condição ao planejado no protocolo.

Outro fato relevante para a adoção deste caso concentra-se na escolha, por parte da equipe, em gerenciar o projeto por meio da abordagem SCRUM<sup>7</sup> de gerenciamento ágil. Aqui vale ressaltar que a escolha da abordagem ágil foi fator de decisão para escolha do caso, todavia, a abordagem de gerenciamento foi sendo acompanhada no transcorrer do projeto de modo a certificar que se tratava, de fato, de um contexto ágil de construção de um projeto. Para o acompanhamento e certificação, foram utilizados como base os 6 critérios definidos e apresentados no Quadro 1, o resultado deste acompanhamento pode ser lido no Quadro 24.

<sup>7</sup> Scrum é um *framework* de gerenciamento de projetos, da organização ao desenvolvimento ágil de produtos complexos e adaptativos com o mais alto valor possível.

**Quadro 24** - Características do projeto (caso) identificando a abordagem ágil no gerenciamento de projeto.

Característica	Abordagem de gerenciamento ágil de projetos
1) A forma de elaboração do plano do projeto	Foi elaborado um plano geral contendo o prazo total de duração do projeto e apresentando apenas os produtos principais do projeto sem detalhes. Elaborou-se também os planos de curto prazo (iteração) que é realizado sucessivas vezes buscando sempre a cronologia das entregas mais importantes de acordo com as definições feitas pelo <i>Product Owner</i> (nomenclatura usada para se referir ao representante do cliente no projeto). Os planos de curta duração (iteração) também são chamados de “ <i>sprints</i> ” na abordagem Scrum.
2) A forma como se descreve o escopo do projeto	Os analistas de requisitos, junto ao <i>Product Owner</i> , elaboraram a Visão do produto contendo uma descrição abrangente do projeto e ilustraram a proposta, conforme mostra a Figura 10.
3) O nível de detalhe e padronização com que cada atividade do projeto é definida	As funcionalidades a serem implementadas no projeto foram mantidas em uma lista chamada de <i>Product Backlog</i> . No início de cada <i>Sprint</i> , realizou-se um <i>Sprint Planning Meeting</i> , ou seja, uma reunião de planejamento na qual o <i>Product Owner</i> prioriza os itens do <i>Product Backlog</i> e a equipe seleciona as atividades que ela será capaz de implementar durante o <i>Sprint</i> que se inicia. As tarefas alocadas em um <i>Sprint</i> são transferidas do <i>Product Backlog</i> para o <i>Sprint Backlog</i> .
4) O horizonte de planejamento das atividades da equipe.	Os planos de curto prazo eram detalhados pelos analistas a cada 15 dias em conjunto com dois representantes do cliente.
5) A estratégia utilizada para o controle do tempo do projeto	A cada dia de uma <i>sprint</i> , a equipe realizava uma breve reunião (primeiro horário, pela manhã), chamada <i>Daily Scrum</i> . Nesta reunião, falava-se sobre o que foi feito no dia anterior, identificando os impedimentos e priorizando o trabalho do dia que se inicia. Todas as atividades e quaisquer mudanças eram identificadas por meio de controles visuais ( <i>post-it</i> ) e a medição de progresso era orientada para o alcance da visão elaborada.
6) A estratégia utilizada para a garantia do atingimento do escopo do projeto	Ao final de um <i>sprint</i> , a equipe apresentava as funcionalidades implementadas em uma reunião denominada <i>Sprint Review Meeting</i> com a participação do <i>Product Owner</i> que avalia, prioriza, adiciona ou altera o produto final do projeto, conforme a experiência com os resultados alcançados. Nesta reunião (ou, em alguns casos, em reuniões posteriores), realizava-se ainda uma <i>Sprint Retrospective</i> para assimilar lições aprendidas e a equipe poder iniciar o planejamento do próximo <i>Sprint</i> . Desta forma, o ciclo se reiniciava.

Fonte: Elaboração própria.

A equipe do projeto se apresentava contendo o *Development Team*, um *Product Owner* (P.O) e um *Scrum Master*. Muito embora na abordagem ágil não sejam adotadas nomenclaturas para distinguir os papéis, neste projeto fora identificado o uso comum dos termos que definem as funções durante a execução, sendo estes, 4 desenvolvedores, 3 analistas de teste e 2 analistas de requisitos. Em relação ao P.O, este ocupava função externa à equipe de desenvolvimento (Coordenador de Logística do cliente) e esta escolha fora justificada pela experiência deste na função e pelo conhecimento técnico acerca das funcionalidades a serem automatizadas pelo *software*. E o *Scrum Master* foi um papel ocupado por um dos analistas de requisitos do projeto.

Ainda neste período, o *Product Owner* (representante do cliente - Coordenador de Logística) foi contatado pela pesquisadora onde foram apresentadas as propostas de acompanhamento do projeto em tempo real com intenção de formar um estudo de caso para pesquisa acadêmica em um projeto de mestrado e ele se mostrou satisfeito com o protocolo apresentado e deu seu aceite em participar, desde que o nome da empresa cliente se mantivesse

sob sigilo. A partir de então, a pesquisadora ganhou espaço para participar de todas as reuniões deste projeto que tinha prazo estipulado de 3 meses (entre fevereiro e abril de 2021) - prévia estimada por meio da proposta comercial.

Como instrumentos de pesquisa foram utilizados: análise de documentos, entrevistas semiestruturadas e observações participativas.

#### 5.4 OBSERVAÇÕES PARTICIPATIVAS

Além da análise de conteúdo em documentos de projetos já finalizados, foi realizada uma análise por meio de observações participativas em reuniões e encontros entre a empresa DESENV e cliente durante o desenrolar do projeto visando compreender aspectos referentes aos objetivos desta pesquisa. Tais objetivos versam as técnicas utilizadas para o levantamento de requisitos buscando compreender se ocorre o envolvimento dos usuários na EAR e como este processo se desenvolve. Busca avaliar a disponibilidade de acesso ao usuário por parte da equipe desenvolvedora e identificar se o papel do usuário é subestimado e/ou anulado pelo envolvimento do papel do cliente neste processo. Estas observações também buscaram levantar informações acerca do grau de dificuldade de compreensão dos requisitos por parte dos desenvolvedores e ainda, sobre o grau de aceitação do *software* por parte dos usuários finais.

As observações aconteceram após a aceitação da proposta comercial por parte do cliente até a fase de homologação, conforme ilustrado no Quadro 25.

**Quadro 25** – Fonte de informação para as observações participativas

Fonte	Quantidade	Participação da pesquisadora (Qtde)	Participação dos usuários
Reuniões formais	4	4	1
Grupos de discussão	12	6	0
Entrevistas	4	3	2
Visitas ao cliente	1 (3 dias)	1 (1 dia)	3

Fonte: elaboração própria.

Esta etapa consistiu em observar, portanto, reuniões formais entre DESENV e cliente com a participação dos analistas de requisitos; grupos de discussões envolvendo analistas de requisitos, desenvolvedores e/ou representantes do cliente; entrevistas com cliente e/ou usuários e, ainda, visitas do analista de requisitos ao ambiente onde o *software* será instalado (observando, além das técnicas, toda e qualquer conversa em que se utilizou a linguagem oral).

#### 5.4.1 Observação das reuniões para elicitação de requisitos

A primeira reunião formal ocorreu 5 dias após o fechamento do acordo comercial da proposta. No encontro, que ocorreu na sede da empresa cliente e teve duração de 4 horas (incluindo um almoço), estiveram presentes 2 analistas de requisitos da DESENV e o coordenador de logística (P.O), representando o cliente, além da pesquisadora que atuou passivamente observando as discussões e a forma como as informações eram coletadas e armazenadas. Esta reunião teve caráter formal, onde os analistas passaram a conhecer, pela visão do cliente, as necessidades para o desenvolvimento do *software*. Questões como: “Onde o *software* será implantado”, “Quem serão os usuários”, “Frequência de acesso”, “Como surgiu a demanda do *software*” foram abordadas neste primeiro contato entre empresas e os dados armazenados por meio da escrita digital em um dispositivo móvel pertencente a um dos analistas que, posteriormente, em ambiente da empresa DESENV, transcreveu suas percepções em um documento chamado Visão, ilustrando o produto a ser entregue ao cliente. Este documento foi revisado outras duas vezes por ocasião de outros encontros que provocaram uma melhor compreensão das necessidades.

O segundo encontro contou com a participação de líder do SAC (representando o usuário, embora não seja, pois não executa as tarefas, apenas coordena equipes) e ocorreu no intervalo de uma semana após a primeira reunião. Já a terceira reunião aconteceu na semana posterior e teve como objetivo principal adequar as datas de entrega em função da pandemia da COVID-19, no entanto, neste encontro foram discutidos ajustes no documento Visão para que ficasse condizentes com as expectativas do cliente.

A última reunião formal aconteceu com objetivo de liberar o *software* para a fase de homologação, onde o cliente (junto ao usuário) testa o produto entregue a fim de certificar se este corresponde às necessidades percebidas inicialmente e se adequa na rotina de trabalho do usuário. Nesta reunião participaram os analistas de requisitos, o *Scrum Master*, a analista de teste (qualidade) e o P.O (representante do cliente - coordenador de logística), além da pesquisadora.

#### 5.4.2 Observação da visita ao cliente para elicitação de requisitos

Tendo completado 5 dias após o segundo encontro, um dos analistas da DESENV se deslocou para a unidade da empresa cliente onde se localiza a área central do SAC para lá permanecer durante 3 dias, juntamente do *Product Owner*, em observação do cenário e

elicitação de parte dos requisitos para dar início a elaboração do *Product Backlog* (pelo P.O), seguido das especificações funcionais (por meio de *stories*) para desenvolvimento. Durante esta fase, por se tratar de um período restritivo em função da pandemia causada pela COVID-19, a pesquisadora não pode estar presente todos os dias e teve autorização da empresa cliente para permanência apenas no primeiro dia. Neste dia, portanto, o analista informou que a técnica utilizada para elicitação dos requisitos se tratava de métodos etnográficos. Todavia, os relatos observados pela pesquisadora não eram condizentes com atividades próprias da etnografia na sua natureza mais pura. Vale ressaltar que a etnografia, embora baseada no trabalho de campo, nunca é rígida em sua abordagem. Em vez disso, o analista deve ser o tempo todo informado pelas perspectivas, experiências e habilidades únicas de todos os envolvidos com observações inerentemente colaborativas. Ao contrário disto, o analista esteve sempre no papel de construtor de soluções ao invés de se ocupar construindo pesquisas (e, em última análise, soluções) sob medida para o problema em questão. Vale acrescentar ainda que, ao menos no primeiro dia, os usuários de mantiveram compenetrados em suas atividades rotineiras e não foram abordados pelo analista por meio de questionamentos. De acordo com o analista de requisitos, dois usuários foram entrevistados informalmente por ele no último dia da visita, mas para este cenário a pesquisadora não pode escrever relatos de observação uma vez que, conforme explicado, não pode estar presente.

#### **5.4.3 Observação das entrevistas para elicitação de requisitos**

As entrevistas foram abertas quanto ao questionário e respostas, onde o analista de requisitos (entrevistador) se norteava por uma ficha contendo os pontos centrais (chamados também de pontos de função) do projeto. O P.O, representado pelo cliente - coordenador de logística, mostrou-se entendido acerca do processo que envolve o *software* e, em duas ocasiões, dispensou a participação dos usuários convidados justificando não haver necessidade de desligamento destes de suas rotinas, uma vez que ele tinha para si conhecimentos suficientes acerca do trabalho destes usuários e saberia opinar. Ainda sob insistência do analista de requisitos, nestas duas ocasiões, os usuários foram dispensados.

Nesta observação ficou evidenciado a baixa participação do usuário para os esclarecimentos das dúvidas em relação ao produto a ser desenvolvido, muito embora, o cliente tenha conseguido esclarecer todas as dúvidas levantadas pelos analistas é sabido, por meio da literatura, que com o envolvimento do usuário provoca novos insights e contribui para maior abrangência do produto em suas rotinas diárias.

Aqui vale salientar que, na última entrevista, foram apresentadas algumas partes do projeto em protótipos para avaliação do cliente que, aproveitando a ocasião, sugeriu algumas adequações. Mais uma vez, o usuário não tomou conhecimento do protótipo para aprovação ou sugestões.

#### **5.4.4 Observação nos grupos de discussão**

Em razão da pandemia COVID-19, os grupos de discussões, inicialmente programados para ocorrer toda sexta-feira na sala de reunião da empresa DESENV, acabaram ocorrendo todos de modo virtual. No total foram 12 encontros para apresentar ideias, sugestões e esclarecer dúvidas sobre os requisitos levantados (em formato de *stories*). Destes, a pesquisadora pode participar de apenas 6 (4 oportunidades foram perdidas por questões técnicas e outras duas por coincidirem com horário de outros compromissos profissionais da pesquisadora).

Nestes encontros participaram o *Development Team*, o *Scrum Master*, o *P.O* e a pesquisadora (quando possível). As sugestões e dúvidas eram analisadas pelos analistas que respondiam de imediato quando tinham plena certeza acerca do assunto abordado ou estas eram levadas ao conhecimento do cliente através de e-mails redigidos por eles. Nas trocas de e-mail, o destinatário era sempre o coordenador de logística que respondia com prontidão (a maior demora registrada para a resposta do e-mail foi de 4 horas de espera). No total, foram enviados 10 e-mails oriundos destes grupos de discussão e, em apenas 1 destes, o cliente colocou o líder do SAC como cópia para leitura, mas não houve manifestação deste durante a troca dos e-mails.

Para facilitar a compreensão e ordem dos acontecimentos o Quadro 26 resume os fatos em ordem cronológica e lista os participantes de cada evento, bem como os objetivos principais de cada caso, com exceção dos grupos de discussão (12 no total) cujo propósito foi sempre o mesmo e as datas fixas (toda sexta-feira).

No total, foram oito encontros com o cliente (4 reuniões e 4 entrevistas) sendo apenas dois presenciais em virtude das medidas restritivas adotadas pelos municípios onde se situam as empresas (DESENV e cliente) em decorrência das altas taxas de transmissão do vírus causador da COVID-19. Quanto às reuniões online, todas foram realizadas por meio da plataforma *Google Meet* e gravadas, o que facilitou este estudo, pois o conteúdo pode ser revisado outras vezes para certificação dos fatos observados.

**Quadro 26** – Cronologia dos encontros observados pela pesquisadora

<b>Data</b>	<b>Encontro</b>	<b>Participantes</b>	<b>Objetivos principais</b>
18-2-2021	1ª reunião	Analistas de requisitos (2) Representante do cliente (P.O) Pesquisadora	Tomar conhecimento, pela visão do cliente, acerca das necessidades para o desenvolvimento do <i>software</i> .
25-2-2021	2ª reunião	Analistas de requisitos (2) Representante do cliente (P.O) Líder do SAC Pesquisadora	Apresentar o documento Visão para colher informações para possíveis ajustes
01-3-2021	1ª entrevista	Analistas de requisitos (1) Representante do cliente (P.O) Pesquisadora	Elicitação dos requisitos através da visão do coordenador de logística que representou o cliente durante todas as fases do projeto.
02-3-2021	Visita ao cliente	Analista de requisitos (1) Usuários do SAC Pesquisadora	Observar o cenário onde o <i>software</i> será inserido e elicitar os requisitos para elaboração do <i>Product Backlog</i> , seguido das especificações funcionais (por meio de <i>stories</i> ) para desenvolvimento.
08-3-2021	3ª reunião	Analistas de requisitos (2) Representante do cliente (P.O) Pesquisadora	Adequar as datas de entrega em função da pandemia da COVID-19 e apresentar o documento Visão com os ajustes realizados.
10-3-2021	2ª entrevista	Analistas de requisitos (1) Representante do cliente (P.O)	Ajustar e complementar a elicitação dos requisitos através da visão do coordenador de logística que representou o cliente durante todas as fases do projeto.
11-3-2021	3ª entrevista	Analistas de requisitos (1) Representante do cliente (P.O) Líder do SAC (usuário) Pesquisadora	Elicitação dos requisitos através da visão do líder do SAC que representava, na ocasião, o grupo de usuários do <i>software</i> .
12-3-2021	4ª entrevista	Analistas de requisitos (1) Representante do cliente (P.O) Usuários – SAC (2) Pesquisadora	Elicitação dos requisitos através da visão dos usuários do <i>software</i> .
03-5-2021	4ª reunião	Analistas de requisitos (2) Scrum Master Analista de Teste Representante do cliente (P.O) Pesquisadora	Liberar o <i>software</i> para a fase de homologação, onde o cliente (junto ao usuário) testa o produto entregue a fim de certificar se este corresponde às necessidades percebidas inicialmente.

**Fonte:** Elaboração própria

Ficou evidenciado durante as observações que a pandemia da COVID-19 dificultou o contato próximo do analista com o ambiente do usuário (não é possível julgar se estes momentos aconteceriam, de fato, em situação que não houvesse impedimentos). Ainda que os usuários tenham participado duas vezes em entrevistas virtuais, os analistas não conseguiram forte envolvimento por parte deles e, por consequência, o trabalho do produto se tornou um trabalho de estimativa e assimilação de conteúdo transmitido pelo cliente (seu representante). Os usuários se mantiveram calados nas reuniões e só respondiam quando indagados e, quase sempre, em concordância com seus líderes. Aqui vale ressaltar que as reuniões eram coletivas e não individuais.

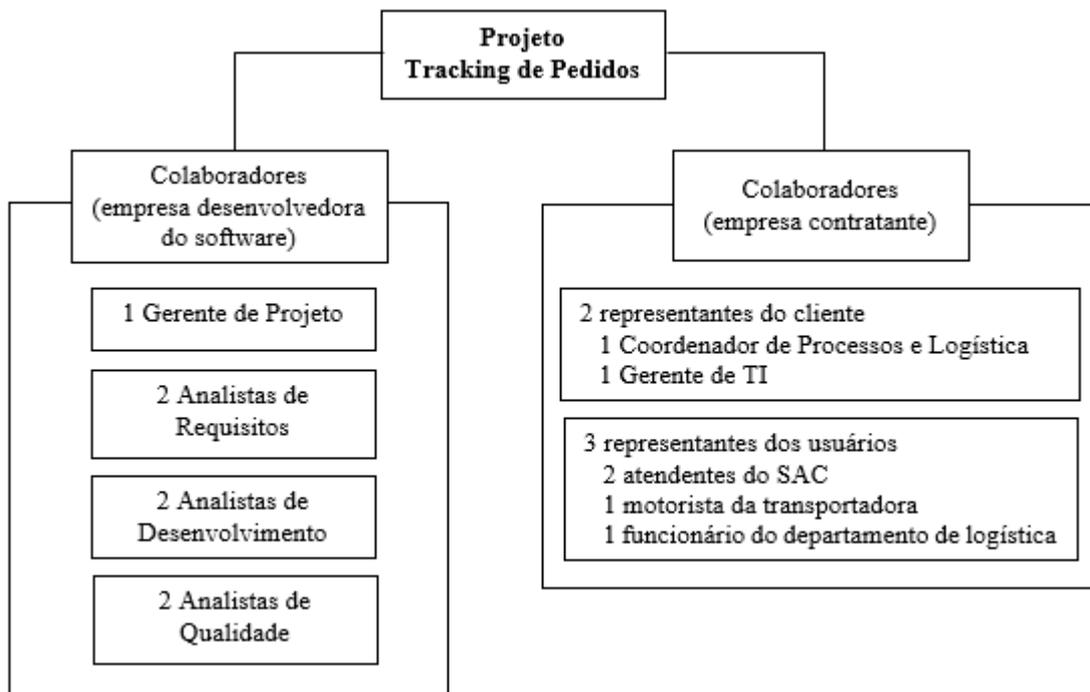
## 5.5 ENTREVISTAS

Para realização das entrevistas a amostra foi constituída pela equipe da empresa DESENV incluindo os envolvidos no processo de desenvolvimento do *software*, desde a elicitación de requisitos ao teste de aceite/homologação e os integrantes da empresa cliente, divididos em dois grupos: clientes e usuários, sendo o primeiro responsável por contatar a empresa e solicitar os requisitos e o segundo formado pelas pessoas que realmente utilizam o sistema. A amostragem é representada conforme a estrutura da Figura 11.

As entrevistas foram sempre agendadas com antecedência, com duração aproximada de 30 minutos. Foram realizadas *online*<sup>8</sup> via plataforma *Google Meet* em sala de reunião criada pelo responsável da empresa DESENV e não foram gravadas, conforme prescrito no protocolo de pesquisa.

As entrevistas partiram de questionamentos básicos, apoiados em teorias e proposições (vide Quadro 20) que ofereceram um amplo campo de informações dos indivíduos. A partir dele foi possível avaliar os níveis de comunicação dos usuários, clientes e desenvolvedores.

**Figura 11** – Amostra para coleta de dados do estudo de caso.



**Fonte:** Elaboração própria.

<sup>8</sup> Em respeito às medidas de isolamento social mediante à propagação do vírus causador da COVID-19

Os principais pontos observados durante as entrevistas (respostas e comportamentos) são descritos e separados por fase das entrevistas (antes do início do projeto, durante o projeto e homologação). Alguns trechos das entrevistas foram transcritos para ilustrar os achados e, para estes casos, a identificação do entrevistado se dará de forma anônima respeitando a identidade de cada um e, serão marcados como “A” para analistas, “D” para desenvolvedores, “AT” para analistas de teste, “SM” para *Scrum Master*, “C” para clientes e “U” para os usuários, como segue.

### 5.5.1 Entrevistas – Antes do início do projeto

As entrevistas que ocorreram antes de dar início à fase inicial do projeto foram realizadas com o *Scrum Master*, dois analistas de requisitos e com os quatro usuários selecionados, sendo um motorista da transportadora, o líder do SAC, um atendente do SAC e um analista de logística. Tiveram o propósito de observar/analisar o grau de envolvimento dos usuários e se este possível envolvimento poderia ser influenciado pelo histórico de projetos, ou seja, projetos anteriores podem ter concedido bagagem e aprendizagem para os envolvidos. Esta fase buscou conhecer ainda se os analistas fazem uso de ferramentas que apoiam o processo de elicitação e quais são elas.

A princípio, o *Scrum Master* e os Analistas de Requisitos mencionaram não haver trabalhado com este cliente em nenhuma outra ocasião e, portanto, desconheciam as atividades do trabalho dos usuários. Complementaram ainda relatando não terem antes desenvolvido projeto similar a esta requisição.

Em relação as técnicas para elicitação dos requisitos, os analistas relataram que as utilizam sempre no intuito de contribuir para melhorar a comunicação entre usuários e analistas, identificando com clareza as necessidades dos usuários. No entanto, além das técnicas escolhidas, os analistas entrevistados consideram que o protagonismo deve estar na ciência dos usuários sobre a necessidade da obtenção do maior número de informações, mantendo-os sempre informados a respeito da relevância dessas informações para o sucesso do produto final.

Segundo o *Scrum Master*, as técnicas utilizadas para elicitação dos requisitos deste projeto buscam conexão com o universo do usuário, mas não nomeou as técnicas ou ferramentas utilizadas, mesmo quando questionado diretamente. Já os analistas mencionaram técnicas como a etnografia (observação), prototipagem com *design* centrado no usuário (UCD) e entrevistas. No entanto, aqui vale destacar que os clientes não mencionaram em nenhum momento (demais fases de entrevistas) o fato de os protótipos estarem voltados para os requisitos não-funcionais

visando uma aproximação com o usuário, porém, é importante mencionar também que os mesmos clientes enfraqueceram esta possível aproximação quando validavam os documentos entregues (como protótipos, por exemplo) sem apresentar aos usuários para uma avaliação.

Em relação aos usuários, ao serem questionados sobre o novo *software* a ser desenvolvido (entrevistas realizadas antes do início do projeto), apenas um respondeu ter ouvido a respeito do assunto em caráter informal, denotando assim a falta de envolvimento destes nesta fase pré-projeto. Todavia, como os analistas ainda não haviam iniciado contato para elicitación, a questão não tem forte impacto para entender o grau de envolvimento dos usuários, haja vista que estes poderiam ser consultados em um momento qualquer após a entrevista.

Na sequência, algumas respostas aos questionamentos formulados:

É interessante que a gente consiga de alguma forma transmitir ao usuário a necessidade de estar levantando todas as informações e, por outro lado, a gente consiga explicar como é feita a utilização de cada uma das informações no processo e o nosso objetivo, aonde queremos chegar com tais informações. Isso é mais importante do que a escolha da técnica. (A1)

O analista deve passar as informações para o usuário, falar sobre as informações do andamento, das dificuldades, dos problemas que está tendo. Isso vale para qualquer técnica que esteja sendo aplicada para o levantamento de informações.

[...]

Vamos iniciar o processo de conhecimento dos requisitos com as entrevistas e depois, com o andamento do projeto, vamos incluir a etnografia e a prototipagem voltada para o usuário, tentando garantir maior usabilidade. (A2)

(A2)

Então, ouvi sobre mudanças no sistema que usamos atualmente através da “rádio peão”, nada formalizado pelos nossos superiores. Pelo menos não até agora. (U2)

### 5.5.2 Entrevistas – Durante o desenvolvimento do projeto

As entrevistas que ocorreram durante à construção do *software* foram realizadas com membros da empresa DESENV, sendo estes: o *Scrum Master*, dois analistas de requisitos, dois desenvolvedores e dois analistas de testes e com os membros da empresa cliente, sendo estes: o coordenador de logística e o gerente de TI, representando o cliente e quatro usuários selecionados, sendo um motorista da transportadora, o líder do SAC, um atendente do SAC e um analista de logística.

Estas entrevistas aconteceram em dias distintos, portanto, em diferentes fases da construção do *software* e tiveram o propósito de observar/analisar o grau de envolvimento dos usuários, bem como de sua disponibilidade de acesso. Esta fase buscou observar se o

envolvimento dos usuários é anulado pelo cliente em função do tipo do projeto (novas funcionalidades ou melhorias) e, observa ainda se a documentação para a construção do *software* se torna insuficiente e/ou inadequada para serem usados pela equipe de desenvolvimento, quando o usuário é envolvido nas fases de elicitação.

Durante as entrevistas com os integrantes da empresa desenvolvedora do *software*, foi reforçado o fato de nenhum destes ter trabalhado anteriormente com o cliente em questão e todos os envolvidos no projeto demonstraram ciência sobre a intenção do *software* em melhorar e adaptar uma rotina já existente na empresa cliente, ou seja, não se tratava de uma funcionalidade desconhecida pelos usuários que poderia prejudicar o repasse de informações pela falta de noção prévia da operacionalização do *software*.

Em relação ao objetivo do projeto, ao serem questionados, os clientes comungaram da mesma ideia em todos os detalhes, enquanto membros da empresa desenvolvedora divergiram em alguns detalhes entre si, como pode ser lido nos recortes das entrevistas transcritos. Os usuários falaram sobre o objetivo do projeto na expectativa de que o *software* preenchesse as necessidades atuais da rotina de trabalho, portanto, davam detalhes técnicos e não falavam de forma generalizada.

O objetivo visa trazer melhorias para o acompanhamento dos pedidos, durante as distribuições entre CD's ou durante as coletas e, principalmente para as entregas dos pedidos do e-commerce. (C1)

Bem, o propósito deste projeto é desenvolver um *software* que ajude o cliente no acompanhamento dos pedidos, inclusive tracking reverso, indicando status, prazos, meios de transporte e, o mais importante, seja um *software* com autonomia para tomar decisões em função de dados parametrizados, como por exemplo, disparo de e-mails para transportadoras ou clientes em determinadas situações. (A2)

Pelo que entendi até o momento, o objetivo deste *software* é gerenciar a entrega de pedidos online para ajudar a transportadora durante o rastreamento destas entregas e informar status para os clientes. (D1)

Este projeto busca ajudar os atendentes do SAC a ter informações mais precisas para conversarem com os clientes, assim as informações estarão sempre atualizadas e o cliente poderá confiar no atendimento SAC da empresa. (AT2)

Este novo sistema deve cuidar da transferência de status do pedido entregue pelo Correios de forma automática, porque isso não temos hoje. E, este projeto vai cuidar também para que os motoristas das transportadoras tenham condições de atualizar os status do pedido mesmo que estejam offline, para que ele não se esqueça de fazer isso em outro momento. (U2)

Ainda nesta busca em conhecer o grau de entendimento do objetivo por parte dos integrantes ficou evidenciado que cada qual definia o grupo de usuários de forma distinta. Por exemplo, desenvolvedores entrevistados acreditavam que o grupo de usuários seriam as

transportadoras e que o SAC poderia consultar o sistema se achasse pertinente. Já os analistas de Teste definiam o SAC como principal grupo de usuários. Analistas e *Scrum Master* definiram os usuários tal como o cliente, ou seja, SAC (Serviço de Atendimento ao Cliente), Transportadoras, Correios, Equipe de Logística dos CD's e Gerente de Logística (para fins consultivos de relatórios gerenciais). Vale destacar que a coleta destas informações se deu após 1 semana do início do desenvolvimento, indicando uma dificuldade de compreensão do projeto como um todo nesta fase inicial.

Em relação ao entendimento dos requisitos por parte dos desenvolvedores, as percepções sobre os aspectos envolvidos no processo de desenvolvimento foram mais técnicas, focadas nos requisitos dos *stories* (e *Product Backlog*). Com uma falta de atuação do P.O, as atividades dos desenvolvedores giravam em torno do que estava escrito nesses documentos e, neste cenário, foi constatada a existência de conflito com a equipe de desenvolvedores por compreensões distorcidas dos requisitos solicitados. Ao serem questionados, os desenvolvedores relatam que esta não é uma situação isolada deste projeto. Afirmam ser comum a dificuldade em compreender casos de *stories*, onde a linguagem é mais voltada ao usuário e de forma mais generalizada, sem muitos detalhes técnicos e regras de negócio. Alguns trechos que exemplificam estes achados:

[...] os dois (analistas e desenvolvedores) devem conhecer com o que vão trabalhar. E também é importante saber o que está sendo criado como um todo. Eu já vi bastante dificuldade quando a gente começa a fazer um projeto a gente não sabe a onde vai dar e não tem uma ideia do todo. A toda hora tem que se parar e fazer aquele balanço, traz o chefe e diz: “vem cá, mas e agora? e se acontecer isso?” Então, é importante que se tenha conhecimento de tudo para começar a desenvolver. (D1)

Um saber o que o outro realmente necessita para fazer o sistema funcionar porque às vezes a visão que o cara (analista) tem é um pouco vaga. (D2)

[...] então ele (o analisa) vai falar de um determinado assunto, a pessoa (desenvolvedor) tem que estar sabendo sobre aquele assunto em profundidade, senão não adianta. (AT1)

[...] o domínio do negócio. Na realidade, o analista começa a aprender aquele negócio. Então, é... digamos, do analista... ele domina as ferramentas de obtenção das informações, as técnicas de obtenção das informações. Isso é importantíssimo para o analista. Mas, se o desenvolvedor só conhecer partes do negócio, sempre haverá interpretações distorcidas dos requisitos a ele atribuído para codificação (D1)

Já em relação a disponibilidade e o envolvimento dos participantes, todos os entrevistados consideraram como parte fundamental para um perfeito entendimento das necessidades dos usuários na fase de levantamento de requisitos. No entanto, os analistas indicaram a ausência dos usuários e desenvolvedores em partes importantes da elicitação de requisitos. Isso é verificado em função dos argumentos que indicam a necessidade de existir, inclusive, uma aproximação entre as duas equipes, usuários e desenvolvedores. A seguir algumas indicações do achado:

Na etapa de elicitação de requisitos, o analista tem que adquirir a confiança do cliente do projeto de modo que este auxilie na obtenção das informações junto aos usuários. Nesta etapa seria interessante também permitir que o usuário fosse um membro de tempo integral na equipe do projeto. Mas, neste projeto, na maioria das reuniões o usuário não esteve presente e, durante a fase de observação, estes estavam ocupados com suas atividades e, ao ser consultado, respondiam com muita objetividade para não paralisar suas tarefas. (A2)

O envolvimento de todas as partes e, principalmente, a disponibilidade de todas as partes poderia dar maior garantia ao projeto. (SM)

Veja, na minha opinião, para que o usuário possa transmitir seu conhecimento sobre suas atividades...bem, eu acho que ele deveria ter disponibilidade de tempo, tipo, que ele fosse liberado do trabalho rotineiro e pudesse se comunicar com tranquilidade sem a correria de ter que voltar ao posto de trabalho dele e concluir suas tarefas. (A1)

O usuário não esteve até o momento envolvido fortemente neste projeto porque eu pude estar representando todos eles. Eu conheço todo o processo de trabalho deles porque já estive daquele lado. (C1)

Por último, o envolvimento do usuário não pode ser constatado ainda nesta fase de forma evidenciada, ou seja, o representante do cliente afirma ter consultado os usuários em todas as oportunidades que se fizeram necessárias. Aqui vale destacar que, de forma unânime, todos consultaram o representante do cliente em casos de dúvidas e dificuldades no desenrolar do projeto. Ainda para esta questão, os usuários relataram desconhecer detalhes do desenrolar do projeto e que todo conhecimento sobre o novo *software* fora obtido em: conversas informais nos corredores, como frases do tipo “em breve seus problemas acabarão” fazendo menção ao *software* que estava sendo produzido; nas poucas reuniões em que participaram, onde, segundo eles, não tiveram autonomia para indagar ou opinar e; na visita que o analista fez ao ambiente de trabalho, que, novamente, segundo eles, em função da rotina acelerada nas atividades não puderam se ater a detalhes através das poucas conversas com o analista .

### 5.5.3 Entrevistas – Fase de Homologação

As entrevistas que ocorreram durante a fase de homologação do *software* foram realizadas com todos os perfis selecionados e buscaram observar se o aceite do *software* está relacionado ao grau de envolvimento dos usuários no levantamento dos requisitos e a familiarização dos usuários com o *software* entregue, uma vez que a literatura retrata que o envolvimento dos usuários no levantamento dos requisitos contribui para que o *software* seja mais próximo de suas reais necessidades.

Até esta fase de homologação, os usuários mencionaram terem apenas visualizado uma parte de telas do *software* apresentadas pelo coordenador de logística para apreciação do novo *layout*, mas não houve interação com as telas e tampouco espaço para dúvidas e/ou sugestões. Para esta questão, um dos desenvolvedores entrevistados reconheceu a importância de uma maior aproximação e um contato mais próximo entre a equipe de desenvolvimento e os usuários e, todos os entrevistados, indicaram em algum momento que quanto mais próximo o contato entre usuários e analistas melhor é a transmissão de informações entre ambos e para a equipe como um todo. Outro fator destacado principalmente por analistas é que para toda técnica, o diferencial consiste em realizar um planejamento prévio do que será solicitado aos usuários durante o levantamento de requisitos. As seguintes respostas demonstram o que foi observado:

Para melhorar mesmo seria... a comunicação direta com o próprio usuário, com quem precisa, com o próprio utilizador. (A1)

Haveria a necessidade de reuniões com os analistas e todas as pessoas que vão trabalhar nesse sistema em um maior número de vezes. [...] se houvessem reuniões assim todas as dúvidas e arestas seriam colocadas nessas reuniões com maior frequência. Com certeza, reuniões frequentes ajudariam o *software* sair mais “redondo”. (U5)

Se o gestor passar para todas as pessoas envolvidas a importância do desenvolvimento daquele sistema, a importância do comprometimento de todas as áreas envolvidas no sistema, desde o gerente até o operacional, possivelmente haveria um engajamento maior e, com isso, uma melhor participação. (D2)

Antes do primeiro contato com o usuário, é importante que o analista estude as legislações internas envolvidas e procure entender melhor o negócio do cliente. (A1)

Foi feita uma lista de perguntas que poderiam ser feitas ao usuário no local da reunião. Nós já tínhamos um conhecimento anterior, antes da reunião com o usuário. Nós estávamos preparados. Fomos para a reunião já sabendo, em parte, o que o usuário solicitaria. Essa preparação prévia foi realizada considerando o processo utilizado pelo usuário que nos foi fornecido antecipadamente. Mas, encontramos poucas vezes o usuário nestas reuniões...mas, o cliente sempre soube dar o respaldo necessário para levantar os requisitos. (A4)

Para os analistas, a disponibilidade e o envolvimento de todos são fundamentais para uma perfeita aplicação da técnica de elicitação. Os analistas citam terem feito muitas adaptações nas técnicas e elas não podem, portanto, serem consideradas como “usadas na íntegra”. Aqui, novamente, fica demonstrada a necessidade da existência de um contato mais próximo entre usuários e analistas. A seguir algumas indicações do achado:

Na fase inicial do levantamento de requisitos, o analista deve adquirir a confiança do cliente para que este o auxilie na obtenção das informações junto aos usuários envolvidos. Nesta fase, também seria interessante transformar o usuário em um membro de tempo integral na equipe do projeto. Ai sim, daria certo. (A1)

Olha, eu acredito que para o usuário, o crucial para ele poder repassar o conhecimento que ele tem e as necessidades que ele tem para o sistema é... ele deve ter disponibilidade de tempo e de preferência que ele seja até liberado do trabalho corriqueiro dele para que ele possa se comunicar com tranquilidade, sem pressa em voltar para o trabalho dele. (A2)

Na sequência, o elemento clareza foi lembrado sendo destacada a importância de sua utilização como um fator que pode melhorar a aplicabilidade das técnicas de elicitação e então melhorar a comunicação entre analistas e usuários. Nesse aspecto foi ressaltada a necessidade do analista ser objetivo e de tentar não utilizar uma linguagem muito técnica na sua interação com o usuário e a necessidade de ambos serem claros e responsáveis ao fornecer informações um ao outro. Reproduzem-se abaixo algumas respostas ao questionamento formulado:

O analista não deve utilizar linguagem técnica com o usuário e deve ser claro e objetivo. (A1)

Se um precisa fornecer as informações e o outro precisa trabalhar essas informações, então, as informações precisam ser passadas de uma forma clara, de uma forma bem responsável. (A8)

O usuário deve realmente passar para o analista o que ele quer porque ele pede uma coisa e depois vê que não é aquilo que ele quer. Ele tem que passar realmente o que ele quer, o trabalho do dia a dia dele: “Eu preciso disso aqui”. O usuário tem que ser claro com o analista e o analista vai colocar dentro do sistema o dia a dia do usuário. Mas, nem sempre o usuário tem este conhecimento e, por isso, nem sempre é permitido consultá-lo (C1)

No que diz respeito à forma como o usuário foi consultado durante o transcorrer do projeto, de forma unânime, a resposta indicou que o cliente foi o canal de transmissão das dúvidas e dificuldades por parte da equipe de desenvolvimento. No entanto, o próprio cliente mencionou estar seguro quanto as informações que passou sem ter consultado os usuários, reforçando sua habilidade técnica com as questões operacionais por já ter executado estas funções dos usuários enquanto colaborador da empresa em tempos remotos.

Quanto aos usuários, estes foram entrevistados logo no dia em que foram apresentados ao sistema já finalizado e em execução (ambiente de homologação). Estes mencionaram a necessidade de um treinamento para compreenderem o funcionamento de todas as operações, pois o layout continha botões e campos com nomes diferentes dos utilizados no sistema atual e o *software* contemplava novas rotinas que necessitavam de parametrizações para o bom funcionamento. Todavia, em um contexto geral, os usuários entrevistados demonstraram-se com boas expectativas quanto à migração para este novo *software*, ou seja, não pareciam estar resistentes ao uso.

Em relação a satisfação dos usuários, estes descreveram que ela existiu somente após algumas adaptações, manutenções e revisões das rotinas do *software*. O *software* foi considerado complexo, porém completo, com boa usabilidade e com telas padronizadas. O sistema foi considerado bom, em relação ao uso, para quem compreende as rotinas exigidas aos sistemas de logística (vínculo com o negócio), porém não foi considerado confiável por fazer atualizações de informações apenas no modo online, o que, segundo os usuários, dificulta o comprometimento de atualização por parte dos motoristas<sup>9</sup> das transportadoras.

Foram relatados problemas de comunicação, falta de treinamento, desconhecimento em relação às funcionalidades e parametrizações do sistema. Ocorreram falhas de comunicação que ocasionaram distorções na compreensão dos desenvolvedores em relação às necessidades dos usuários, acarretando diversos retrabalhos (precisamente 8 pontos de melhorias/ajustes e 12 pontos de correções de falhas operacionais). Os prazos definidos previamente não foram cumpridos, mesmo assim a satisfação dos usuários sobre o sistema foi considerada boa.

A comunicação externa foi considerada boa, principalmente em nível de analista e cliente. Os clientes e usuários, porém, mencionaram a necessidade de retrabalhos por falhas no processo de comunicação. Para os desenvolvedores a comunicação externa não foi acessível e suas maiores dificuldades encontraram-se entre os setores da empresa, devido à existência de barreiras na comunicação, pois, em todas as circunstâncias, deviam passar pelo crivo do analista e isso prejudicava a agilidade do projeto. Isso aconteceu também entre clientes e usuários, já que os usuários consideraram não existir uma boa comunicação interna.

Ainda nos aspectos referentes à satisfação, os desenvolvedores mencionaram a complexidade do *software*, o fato dele não ser confiável e que após algumas adequações ele ficou satisfatório. Outro relato importante evidenciado nas entrevistas é que apresentar o sistema para o cliente à medida que cada módulo é finalizado, facilita na mudança de requisitos.

---

<sup>9</sup> Motoristas das transportadoras utilizam o sistema para atualizar o status dos pedidos. Em condições offline, ele deve aguardar o momento oportuno (online) para atualizar os pedidos.

No entanto, se apresentando apenas para o cliente, o usuário poderá perceber necessidade de mudanças apenas quando tiver acesso ao produto já finalizado, como foi registrado neste caso estudado. Seguem relatos destas percepções:

[...] a gente vai precisar da sua (do cliente) participação no desenvolvimento tantas e tantas vezes por semana para poder você verificar se o implementado está correto. Então, geralmente, a gente costuma fazer assim. (A2)

O usuário, eu acredito que ele tem que ter disposição e tempo para poder estar junto com o analista, para que esse trabalho seja feito com calma e não com pressa. (A1)

[...] é nas reuniões que ele (o analista) vai compreender o que realmente o usuário vai querer [...] se o motorista não puder atualizar as informações sempre que precisar, ele não o fará quando puder, porque ele se esquece ou porque não terá tempo hábil e isto precisa ser ajustado, não tem jeito. (U2)

[...] se a gente conhece os módulos de maneira superficial, poderemos fazer uma análise mais aprofundada do produto apenas na fase final e aí não adianta reclamar das mudanças e adaptações, por que ou é isso ou o *software* cai em desuso. (U1)

Outro aspecto importante que foi elucidado durante os questionamentos sobre a satisfação do projeto consiste na qualidade das interações entre usuários e analistas. É que, de acordo com os analistas, o usuário deve ter um bom conhecimento do negócio que será informatizado. Mas, sob a ótica dos usuários, os analistas é que deveriam possuir uma visão mais detalhada daquilo que estão analisando. Na sequência algumas indicações desse achado:

Então, para essa comunicação o usuário tem que poder passar todas as informações que ele tem sobre o negócio. (D2)

Conseguir identificar o que é realmente relevante dentro do contexto do trabalho que se está desenvolvendo, dentro do que se está mapeando, o que é requisito, o que é relevante para o sistema. Isso é necessário porque existem coisas que estão dentro do processo do usuário e não poderão ser implementadas, que não dependem da área de sistemas. E com tão pouco tempo de contato com o usuário não se pode perder tempo como aquilo que não é importante. (A2)

[...] se você vai falar de um assunto, é preciso acreditar que a pessoa tem que estar sabendo sobre aquele assunto em um certo nível, senão nem adianta. Por isso, muitas vezes preferi me calar por achar que não seria compreendido. (U1)

Se o usuário é comunicativo e sabe falar sobre sua rotina de trabalho, ele repassa estas informações para o analista de forma clara sabe, não fica de enrolação e nem medindo o que se pode falar porque compreende que se o analista tem muitas informações, poderá fazer um bom trabalho para ele. (A1)

Sempre um deve saber sobre o que o outro precisa para então fazer o *software* funcionar [...] sabe, as vezes a visão do cara (analista) é muito vaga. (U3)

Além de identificar claramente que a participação do cliente ocorreu de maneira ativa acompanhando toda a evolução do sistema, desde a solicitação à implantação, outras análises deste levantamento de informações podem ser apreciadas no capítulo que segue.

## 6 RESULTADOS E DISCUSSÕES

Para a triangulação das informações obtidas por meio da análise documental, observações participativas e entrevistas, antes foi necessário consolidar as respostas obtidas por meio das entrevistas. Desta vez não apenas identificadas no discurso dos respondentes, mas agrupadas de forma a compor um todo que possibilite um relacionamento entre elas, análise de documentos de projetos anteriores, a literatura existente sobre o assunto e as proposições levantadas neste estudo. Para facilitar o entendimento do leitor, os achados foram separados e estudados de acordo com o conjunto de respostas para cada uma das perguntas propostas e organizadas de modo a permitir analisar os impactos que cada qual poderia causar nas proposições deste estudo, conforme apresentado no Quadro 27.

**Quadro 27** – Relação das questões e suas respectivas categorias.

P1	P2	P3	P4
Q1, Q2, Q3, Q4, Q7, Q9, Q10, Q12, Q13, Q14, Q15, Q16 e Q17	Q6, Q9 e Q10	Q8, Q14 e Q17	Q3, Q4, Q5, Q7, Q9, Q10, Q12, Q13, Q14, Q15, Q16 e Q17

Onde:

P1: O papel do usuário é envolvido no processo de elicitação de requisitos com apoio de ferramentas em uma adaptação da abordagem ágil no desenvolvimento de *softwares* empresariais sob demanda.

P2: A indisponibilidade de acesso ao usuário no processo de elicitação de requisitos em uma abordagem ágil no desenvolvimento de *softwares* empresariais sob demanda compromete a agilidade do projeto.

P3: Os desenvolvedores de *softwares* empresariais sob demanda, em uma abordagem ágil, têm dificuldade de compreensão dos requisitos quando estes são orientados ao usuário.

P4: O papel do usuário é anulado e/ou subestimado pelo papel do cliente no processo de elicitação de requisitos na abordagem ágil de desenvolvimento de *softwares* empresariais sob demanda.

**Fonte:** Elaboração própria.

### 6.1 TÉCNICAS DE ELICITAÇÃO DE REQUISITOS E O ENVOLVIMENTO DO USUÁRIO

Na literatura poucos foram os estudos encontrados que buscaram envolver o usuário no processo de elicitação de requisitos. Muitos não fazem menção à distinção dos termos cliente, stakeholder e usuário e, desta forma, não é possível compreender com clareza a relação das técnicas estudadas com cada perfil.

Nas observações de projetos anteriores já finalizados, os relatos apresentam a utilização de técnicas como Observação (citadas nos documentos como Etnografia), Entrevistas, Protótipos e *User Story*. Por meio das análises documentais não foi possível identificar se houveram adaptações nas técnicas, mas ficou evidenciado a participação dos usuários de forma mais efetiva quando os analistas puderam fazer contato com estes desde os primeiros passos do projeto, sugerindo que o ganho de confiança dos usuários pode intensificar o envolvimento. Em um projeto específico, a única alteração solicitada pós homologação tratou-se exatamente de questões de usabilidade, assunto abordado na única reunião em que não houve participação do usuário. Esta nota sugere que a participação dos usuários nesta reunião poderia ter evitado este retrabalho.

Já em relação ao estudo de caso, neste foi possível identificar a possibilidade de envolver o usuário no processo de elicitação de requisitos com apoio de ferramentas adaptando a abordagem ágil no desenvolvimento de *softwares* empresariais sob demanda. Os analistas, inclusive, fizeram menção a esta prática em projetos anteriores considerados de sucesso, onde o envolvimento de todos foi mais natural pelo não enfrentamento de uma pandemia, como é o caso atual. No entanto, para este estudo de caso, algumas das técnicas mencionadas na literatura, como *design thinking*, UCD, JAD e GREM, não puderam ser identificadas na prática, enquanto que *User Story*, Protótipos, Observações e Entrevistas foram técnicas amplamente utilizadas e, em todas se notou adaptações, como pode ser lido no Quadro 28.

**Quadro 28** – Técnicas de elicitação e suas adaptações para o caso estudado.

Técnica	Adaptações
<i>User Story</i>	Após 3 entregas de módulos, os <i>user stories</i> foram adaptados para contemplar uma parte com escrita técnica, como sugerido na literatura por Medeiros et al. (2020), propondo uma abordagem nomeada de RSD (Especificação de Requisitos para Desenvolvedores) para criar especificações que forneça informação mais próxima das necessidades de desenvolvimento.
Prototipação	Não fora feito prototipação para todos os módulos (telas), apenas para aqueles cujo cliente indicou ter maiores chance de dificuldade quanto a usabilidade. Desta forma, pretendeu-se focar no que era importante para o usuário e ganhar em agilidade no projeto, como proposto na literatura por Bellucci et al. (2015), onde o cliente participa ativamente no processo de inovação através da geração de ideias. A abordagem é chamada de Co-design Extrem. No entanto, vale ressaltar que o estudo de Bellucci et al. (2015) utiliza o termo “cliente” e, no caso estudado, houve a pretensão de envolver o usuário, mas este não esteve envolvido em razão da barreira imposta pelo representante do cliente, a ser discutido mais adiante no texto.
Observação	Neste caso, as adaptações aconteceram pela situação desfavorável causada pelas restrições de acesso ao local de trabalho do usuário em função da pandemia, como detalhado no capítulo anterior. Assim, as observações passaram a ser convertidas em entrevistas e, mais uma vez, com poucas participações do usuário.
Entrevistas	Quando realizadas de modo virtual, criou-se o anseio em maior participação dos usuários pela facilidade de acesso, porém, não pode ser concretizado na prática.

**Fonte:** Elaboração própria.

Para todos os casos, o envolvimento do cliente se mostrou eficaz, indicando que as adaptações surtem efeito quanto ao envolvimento, porém, os usuários não tiveram contato próximo ao analista na aplicabilidade destas técnicas para melhores constatações.

## **6.2 A INDISPONIBILIDADE DE DO USUÁRIO E A AGILIDADE DO PROJETO**

Para a P2, de acordo com o caso estudado, não é possível fazer muitas constatações. Durante o desenrolar do projeto, os usuários apenas participaram de algumas reuniões e, para todos os outros tratamentos de elicitación e/ou dúvidas, o papel do cliente estava fortemente envolvido e criando barreira de acesso ao usuário.

A literatura indica, através do estudo de caso realizado por Ramesh e Baskerville (2010), que o desafio chamado de “participação dos clientes” depende de muitos fatores, tais como disponibilidade, consenso e confiança, especialmente nas fases iniciais do projeto. Um desenvolvedor de *software* que participou deste estudo relatou que os usuários adequados (finais) estarão sempre ocupados com as suas atividades na organização ou o seu chefe não lhes permitirá juntar à equipe de desenvolvimento em tempo integral. Esta afirmação pode ser notada na observação deste estudo de caso.

Apenas na fase de observação, onde o analista visitou o local de trabalho dos usuários, estes se mostraram ocupados em suas atividades e pouco participativos nas tentativas de interações por parte do analista e, para este cenário, o estudo apresenta que o não envolvimento dos usuários pode ter prejudicado a agilidade do projeto uma vez que a falta de clareza no entendimento do processo por parte do analista pode ter acarretado retrabalhos em fase de homologação.

Já em relação aos achados na fase de análise dos documentos de projetos anteriores, estes não podem ser base de discussão para esta questão, uma vez que, mesmo tendo percebido a indisponibilidade dos usuários nos projetos Pj2 e Pj3, ainda assim não foi possível avaliar a agilidade dos projetos em tempo de execução de modo a avaliar se houve prejuízo ou não.

## **6.3 COMPREENSÃO DOS REQUISITOS ORIENTADOS AO USUÁRIO**

Na literatura, através da leitura dos estudos de Harbers et al. (2015), é possível notar os resultados indicando que *User Stories* orientados aos usuários são complicados para o desenvolvimento e testes. Este estudo conclui que os *User Stories* baseados em valores precisam ser traduzidos em Casos de Uso para torná-los adequados ao planejamento e

organização das atividades de desenvolvimento. Ainda na literatura, Soares et al. (2015) reforçam os achados de Harbers et al. (2015) mostrando que requisitos levantados por meio de *User Stories* trazem desafios como, por exemplo, a escrita apresentando descrição de alto nível dos requisitos do *software*, visando o cliente e, muitas vezes, são breves, vagos, ambíguos e insuficientes para capturar as complexidades do design inicial.

Este é, portanto, um ponto congruente entre a literatura e o caso observado. Durante as entrevistas com equipe de desenvolvimento e teste ficou evidenciado que as distorções na compreensão em relação às necessidades dos usuários foram ocasionadas pelo tipo de linguagem oferecida nos *User Story* que prejudicou o entendimento técnico para a conversão da necessidade em lógica de programação. A afirmação pode ser verificada como verdade quando identificadas que as adaptações/melhorias requeridas pelos usuários estão, na grande maioria (7 de 8 no total), relacionadas aos módulos iniciais. Ou seja, após a adaptação da escrita dos *User Stories*, notou-se uma tendência na melhor compreensão dos requisitos por parte dos desenvolvedores.

As observações obtidas na análise documental de projetos anteriores não contribuíram para esta questão, pois não trouxeram informações referente ao nível de entendimentos dos desenvolvedores em função da técnica de escrita dos requisitos.

#### **6.4 PAPEL DO USUÁRIO ANULADO E/OU SUBESTIMADO PELO PAPEL DO CLIENTE NO PROCESSO DE ELICITAÇÃO DE REQUISITOS**

Como visto, a literatura investigada nesta pesquisa não apresenta relatos sobre este ponto de observação. A experiência técnica da pesquisadora trouxe esta proposição até aqui e, para este caso estudado, houve constatação do fato elucidado. Ou seja, houve pouca participação do usuário e, para todas as indisponibilidades deste perfil, o envolvimento do cliente supria, com restrições, as necessidades. E, justificando as indisponibilidades dos usuários, o representante do cliente fez menção a ‘não necessidade’ de participação destes por vários motivos, como:

1º - Serem representados pelo ‘cliente’, uma vez que este possuía conhecimento sobre a rotina operacional a ser conduzida no *software*;

2º - Não se ausentarem de suas atividades rotineiras;

3º - Por não conhecerem sobre a estratégia organizacional, poderiam dificultar a passagem das informações e tornar o processo lento levantando pontos de menor importância para o negócio.

Através das observações e entrevistas, constatou-se que o usuário envolvido no processo pode ser benéfico em vários aspectos, mas ao mesmo tempo pode representar problemas ou riscos. Através relatos ouvidos e percebidos em documentos, nem todos os usuários podem contribuir com o projeto da mesma forma. Então, subentende que os tipos de cliente variam de acordo com os estágios ou fases do projeto, como:

- Usuários que possuem alto conhecimento técnico, caracterizados pela participação intensa e alto engajamento (quando convidado) em projetos de desenvolvimento de novos produtos. São usuários que dão grandes contribuições relacionadas às funcionalidades do *software* e normalmente requerem algumas adaptações por parte da equipe de desenvolvimento.

- Usuários que fornecem inputs para o projeto por meio de relatos sobre as necessidades da rotina de trabalho diário. Este usuário não precisa ser envolvido frequentemente, mas o input deste tipo de usuário é valioso na fase de geração de ideias para novos projetos de *software* e para a fase de homologação, pois podem contestar a aplicabilidade do *software* construído na rotina de trabalho.

- Usuários que fornecem experiências de aplicação do *software*, caracterizados por estarem prontamente motivados para testar protótipos e compartilhar suas experiências com a empresa. Geralmente, este tipo de usuário pode esperar algum tipo de recompensa como fator motivacional para se engajar no projeto.

Em síntese, o não envolvimento do usuário prejudicou o conhecimento de domínio por parte dos analistas de requisitos e desencadeou atividades de retrabalho em período pós homologação. Esta sequência acabou por estender o prazo para o *software* estar operacionalizando na empresa cliente e prejudicou o nível de satisfação e confiança no *software* na visão de cliente e usuário.

## 7 CONCLUSÃO, LIMITAÇÕES E RECOMENDAÇÕES

Neste capítulo são feitas as considerações sobre os resultados encontrados e destacadas as principais conclusões, assim como evidenciadas as contribuições da pesquisa, tanto para o campo acadêmico quanto empresarial. Também são explicitadas as contribuições originais, limitações e sugestões para estudos futuros.

### 7.1 Contribuições do estudo

O estudo de caso acompanhado em tempo real, bem como a análise de documentos de projetos anteriores e revisão da literatura dessa dissertação permitiram avançar nas discussões e aumentar a base empírica acerca do envolvimento dos usuários na EAR, avaliando se o papel do usuário é subestimado e/ou anulado pelo envolvimento do papel do cliente neste processo e sobre a aceitabilidade do software por parte dos usuários finais. O estudo discorreu sobre a relação entre o sucesso na elicitação dos requisitos e o envolvimento dos usuários, justificando que a interação com os usuários do *software* poderia ser alcançada se o processo de comunicação fosse realizado com o mínimo de ruído, ou seja, um processo em que a comunicação entre os envolvidos ocorresse de maneira clara e constante.

O estudo realizado reforçou a importância do processo de comunicação entre usuários e analistas, indicando que este precisa, além da clareza já citada, de um grande comprometimento do cliente e/ou seu representante desde o início do projeto, ou seja, que uma vez definido uma requisição de projeto para o desenvolvimento de um novo *software* ou adaptações, tanto o cliente solicitante quanto analistas de requisitos deverão comprometer-se em buscar conhecer os anseios e necessidades do usuário na sua rotina diária. Ainda dentro deste contexto identificou-se que só a clareza e o comprometimento são insuficientes para a melhoria do processo de identificação das necessidades e expectativas dos usuários. Assim, identificou-se que a cooperação entre usuários e analistas de requisitos é extremamente necessária para compor um cenário no qual possa existir um completo compartilhamento do conhecimento, além da informação.

Em resumo, a partir da análise comparativa dos dados bibliográficos levantados na revisão sistemática da literatura e do estudo de caso, verificou-se que:

- Os requisitos dos usuários quando não conhecidos completamente pelo cliente e analista de requisitos desencadeiam solicitações de adequações após a entrega (pós fase de homologação), comprometendo o uso do *software* pelos usuários. Pelo estudo de caso esta

afirmação pode ser verificada nos casos em que houve a necessidade de adequações no sistema após sua implantação, o que está relacionado às dificuldades encontradas em funções específicas do *software* e ao surgimento de necessidades não cumpridas em relação a necessidade do usuário. Ainda que estes achados tenham sido percebidos nas validações de *Sprint Reviews*, algumas atividades precisaram ser reconduzidas, gerando atraso no projeto;

- A comunicação apresentou impactos significativos, ou seja, falhas de comunicação que acarretaram retrabalhos, devido à interpretação técnica dos desenvolvedores que inviabilizou alguns módulos do sistema na prática. Embora as cerimônias do Scrum tivessem sido realizadas zelando pela boa comunicação, estes impactos podem estar relacionados a escrita dos requisitos em formato de *user stories* orientados à linguagem do cliente. Outro fator pode estar relacionado aos baixos índices de participação do usuário no processo de desenvolvimento, que esteve presente em poucas fases de elicitação e desenvolvimento, pois foi representado pela figura do cliente que se responsabilizou por coletar as informações verbais e não verbais dos usuários e pelo repasse aos analistas de requisitos. Desta forma, as reais necessidades dos usuários podem não terem sido elicitadas corretamente.

- A satisfação está diretamente relacionada ao processo de comunicação que envolva os usuários do *software*. As ausências de diálogos ocorridas na fase de elicitação de requisitos do *software* ocasionaram problemas visíveis no produto afetando diretamente a satisfação dos usuários. Quanto maior for à qualidade e fluência da comunicação entre os envolvidos no processo de desenvolvimento de *software* e os usuários maior é a tendência de satisfação em relação ao sistema e menores serão os índices de retrabalho e adaptações.

- Um achado relevante está relacionado ao uso das práticas híbridas no caso. A disseminação das práticas do ágil dentro da empresa apresentou a combinação e adaptação dos modelos de gestão tradicionais. Embora a empresa não mencione utilizar abordagem híbrida, alguns pontos justificaram este achado, como, por exemplo, a distinção clara entre os papéis dentro do time de desenvolvimento ou ainda a anulação de algumas práticas de reuniões de validação do usuário das partes entregues do *software* produzido.

- Outra percepção deste estudo se dá acerca do papel crítico do P.O como “tradutor” da linguagem do usuário para a equipe de desenvolvimento. O P.O deve ser o elo entre a equipe de desenvolvimento e clientes. Este papel deve ter compreensão sobre o negócio e domínio em traduzir as necessidades aos desenvolvedores. No caso estudado, o P.O foi uma figura externa à equipe de desenvolvimento, era funcionário da empresa cliente. Este ponto acaba por impactar na compreensão clara por parte da equipe de desenvolvimento de quais recursos são necessários no produto, uma vez que o P.O não demonstrou o domínio na habilidade de tradutor.

Além dos fatores citados acima e identificados nas palavras dos entrevistados, verifica-se que, muitas vezes, o time de desenvolvimento (principalmente os perfis cuja principal atividade concentra-se na elicitação de requisitos) tendem a transferir aos usuários o insucesso de algum projeto. Verifica-se, nesse contexto, que embora possa existir alguma culpa dos usuários pelo naufrágio do projeto torna-se necessário enfatizar que todo o processo de levantamento das necessidades dos usuários e a conseqüente definição de todos os requisitos que devam ser considerados durante a especificação do produto de *software* é de responsabilidade do time de desenvolvimento do projeto.

Mesmo que se diga, em alguns casos, que o usuário contribuiu pouco ou não cooperou com a equipe de desenvolvimento com vistas ao sucesso do projeto, não se pode transferir integralmente ao usuário tal problemática. Durante a identificação das necessidades dos usuários, os analistas utilizam-se de processos dedutivos e ou inferenciais para suprir a falta de dados claros que deixam de ser fornecidos pelos usuários. A partir daí, é possível desenvolver estratégias para motivar os usuários no sentido de obter-se o comprometimento e a cooperação necessários para que o processo de comunicação flua de maneira clara e sem ruídos entre ambos.

Como pode ter sido visto pela leitura deste trabalho, não se objetivou aqui apresentar receitas ou moldes que direcionem o processo de análise de requisitos. Cada profissional apto a realizar o trabalho de análise de requisitos deve, com base em sua bagagem acadêmica e profissional, procurar identificar nas diversas metodologias disponíveis a mais adequada ao seu ambiente de trabalho. O presente trabalho de pesquisa, por ser de cunho teórico, poderá servir para que analistas percebam que a comunicação entre eles e os usuários durante o levantamento de requisitos poderá ser melhorada se eles conseguirem agregar ao seu cotidiano os elementos identificados neste estudo.

Por fim, o estudo permite as organizações o melhor entendimento da dinâmica das trocas de informações e aprendizados no GAP, incluindo o envolvimento dos usuários.

## **7.2 Limitações do estudo**

Pela utilização do estudo de caso como método de pesquisa, seus resultados não podem ser generalizados, devendo ser realizados novos estudos. Contudo, a discussão teórica em contraposição com os estudos de campo permitiu o melhor entendimento da situação e foi suficiente para verificar e discutir as proposições deste estudo, contribuindo e possibilitando futuras investigações, como descrito no próximo tópico.

Os casos estudados, em tempo real de acompanhamento e análises documentais de projetos anteriores, possuíam características importantes e permitiram reflexões, análises e aprendizados profundos, tanto pela qualidade das empresas e projetos estudados, quanto pelo acesso obtido pelo pesquisador. Contudo, todos os projetos analisados utilizavam modelos de gerenciamento de projetos que se aproximavam do Scrum, o que pode ter introduzido vieses no estudo, apesar do esforço durante a construção e análise para que isso não ocorresse.

Pode-se considerar ainda como limitação deste estudo, a dificuldade da pesquisadora em acompanhar, como expectadora, os encontros presenciais uma vez que foram realizados em período de pandemia da COVID-19, momento este em que grande parte das empresas restringe o acesso aos locais de trabalho. Considerando um período sem pandemia, a pesquisadora teria assistido mais encontros e estes, com maior duração, vislumbrando um maior número de informações levantadas para a discussão do caso.

### **7.3 Sugestões para estudos futuros**

Além de cumprir seu papel explicando e organizando a fase de elicitação de requisitos e o envolvimento do usuário no gerenciamento ágil de projetos de *software* sob demanda, durante a construção do estudo, foram levantados diversos questionamentos para estudos futuros, contribuindo também para a descoberta de novos problemas e questões de pesquisa.

Com relação à evolução do estudo, sugere-se que outros estudos de casos sejam feitos em projetos que adotem métodos ágeis diferentes dos dessa dissertação, por exemplo, XP e *Kanban*. Ainda neste sentido de evolução, sugere-se a investigação em empresas de ramos diferentes, visando compreender as similaridades e discrepâncias a partir deste estudo como base.

Além disso, uma *survey*, tendo como referência este estudo, possibilitaria o diagnóstico da realidade do envolvimento dos usuários na fase de elicitação de requisitos em projetos GAP em escala ampla e a melhor orientação de estudos futuros para suprir as dificuldades encontradas.

Propõe-se a realização de estudos que visem incorporar questões de "antropologia de desenvolvimento de *software*" e investigar se os analistas de requisitos que possuam uma formação mais humanista apresentam ou não maior vantagem na identificação das necessidades dos usuários em relação àqueles formados em cursos mais voltados à parte técnica, como os cursos de ciência da computação e de tecnologia da informação.

Sugere-se ainda que, um estudo futuro investigue se o tipo de metodologia utilizada no processo de desenvolvimento de *software* pode ou não influenciar na capacidade do analista em identificar as necessidades dos usuários.

## **REFERÊNCIAS**

- ADIKARI S., McDonald C., Campbell J. **Little Design Up-Front: A Design Science Approach to Integrating Usability into Agile Requirements Engineering**. Faculty of Information Sciences and Engineering, University of Canberra ACT 2601 Australia, 2009.
- ADIKARI, S., McDonald, C., Lynch, N. **Design Science-Oriented Usability Modelling for Software Requirements**. In: *Proceedings of HCI International*, pp. 373–382 (2007)
- AHERN, T.; LEAVY, B.; BYRNE, P. J. **Knowledge formation and learning in the management of projects: a problem solving perspective**. *International Journal of Project Management*, v. 32, pp. 1423-1431, 2014.
- ANDERSON, D. J. **Kanban – Successful Evolutionary Change for your Technology Business**, Blue Hole Press, 2010.
- ANITHA, P. C.; Prabhu, B. **Integrating requirements engineering and user experience design in Product life cycle Management**. 1st International *Workshop* on Usability and Accessibility Focused Requirements Engineering, UsARE 2012 - Proceedings.
- AMARAL, D.C, et al. **Gerenciamento Ágil de Projetos – aplicação em produtos inovadores**. São Paulo: Saraiva. 2011.
- BECK, K., et al. **Manifesto for agile software development**. 2001. Disponível em: <<http://www.agilemanifesto.org/>>. Acesso em: 21 jun. 2019.
- \_\_\_\_\_. **Extreme Programming Explained: Embrace Change**, Addison-Wesley, 2000.
- BELLUCCI, A, et al., **Extreme Co-design: Prototyping with and by the User for Appropriation of Web-connected Tags**, in: 2015: pp.109–124.
- BEZERRA, Eduardo. **Desenvolvimento incremental e iterativo**. Disponível em <http://www.mundooo.com.br/php/modules.php?name=MOOArtigos&pa=showpage&pid=20>. Acessado em julho de 2019.
- BJARNASON, E., Wnuk, K., Regnell, B.: **A case study on benefits and side-effects of agile practices in large-scale requirements engineering**. In: *Proceedings of the 1st Workshop on Agile Requirements Engineering - AREW*, pp. 1–5, New York: ACM Press, 2011.
- BJARNASON, E., et al. **A multi-case study of agile requirements engineering and the use of test cases as requirements**, *Information and Software Technology*, Volume 77, 2016, Pages 61-79, ISSN 0950-5849, <https://doi.org/10.1016/j.infsof.2016.03>
- CALEFATO, F., Damian, D. & Lanubile, F. **Computer-mediated communication to support distributed requirements elicitation and negotiations tasks**. *Empir Software Eng* 17, 640–674 (2012). <https://doi.org/10.1007/s10664-011-9179-3>
- CHIN, G. **Agile Project Management: how to succeed in the face of changing project requirements**. New York: Amacon. 2004.
- CONFORTO, E.; AMARAL, D.; DA SILVA, S.L. **Roteiro para Revisão Bibliográfica Sistemática: Aplicação no Desenvolvimento de Produtos e Gerenciamento de Projetos**. Congresso Brasileiro de Gestão de Desenvolvimento de Produtos – Porto Alegre – RS. 2011.
- DAHLSTEN, F. **Hollywood wives revisited: a study of customer involvement in the XC90 project at Volvo Cars**. *European Journal of Innovation Management*. Vol. 7, n. 2, pg. 141-149, 2004.
- DYBÅ, T., DINGSØYR, T. **Empirical studies of agile software development: a systematic review**. *Information and software technolog*. <https://doi.org/10.1016/j.infsof.2008.01.006>. 2011, 833–859.

- FERREIRA, Aurélio Buarque de Holanda. **Dicionário da língua portuguesa**. 5 ed. Curitiba: Positivo, 2014.
- EDER, S., et al. **Diferenciando as abordagens tradicional e ágil de gerenciamento de projetos**. *Production*. USP, São Carlos, 2016.
- FOX, J. D.; Sillito; Maurer, F. “**Agile Methods and *User-Centered Design* : How These Two Methodologies Are Being Successfully Integrated In Industry**” in *Agile*, 2008, pp. 63–72.
- FOWLER, M. **The new methodology**. 2005. Disponível em: <<http://www.martinfowler.com/articles/newMethodology.html>>. Acesso em: 21 julho de 2019.
- FOWLER, M.; HIGHSMITH, J. **The agile manifesto**. Technical report, Agile Alliance, 2001.
- GIL, A. C. **Metodologia do ensino superior**. 4 ed. São Paulo: Atlas, 2011.
- HASTIE, S.; WOJEWODA, S. Standish Group 2015 Chaos Report Q&A with Jennifer Lynch. 2015. Disponível em: <https://www.infoq.com/articles/standish-chaos-2015>. Acesso em 22 abr. 2020.
- HARBERS, M., Detweiler, C., Neerincx, M. A. **Embedding Stakeholder Values in the Requirements Engineering Process**. Springer International Publishing, Cham, 2015.
- HEIKKILA, V. T., et al. **A Mapping Study on Requirements Engineering in Agile Software Development**. *Proceedings - 41st Euromicro Conference on Software Engineering and Advanced Applications*, SEAA. 2015.
- HIGHSMITH, J. **Agile Project Management: creating innovative products**. Boston: Addison – Wesley. 2004.
- HODA, R.; NOBLE, J.; MARSHALL, S. **Developing a grounded theory to explain the practices of selforganizing Agile teams**. *Empirical Software Engineering*, v. 17, n. 6, pp. 609-639, 2012.
- HOSBOND, J.H., NIELSEN, P.A. **Misfit or misuse? lessons from implementation of scrum in radical product innovation**. In: *International Conference on Agile Processes and Extreme Programming in Software Engineering*. Springer, 2008. p. 21–31.
- IEEE – Institute of Electrical and Electronics Engineers. **Standards Glossary of Software Engineering Terminology**: Std 610.12, N.Y., 1990. 84p.
- IEEE – Institute of Electrical Electronics Engineers. **IEEE Guide for Developing System Requirements Specifications**. IEEE Std 1233-1998, New York, 1998b.
- INAYAT, I., et al. **A systematic literature review on agile requirements engineering practices and challenges**. *Computers in Human Behavior*. 2014
- International Organization for Standardization, **ISO 9241-210: 2010 - Ergonomics of human-system interaction – Part 210: Human-centred design for interactive systems**, 2010.
- KETOKIVI, M., & Choi, T. **Renaissance of case research as a scientific method**. *Journal of Operations Management*, 232-240. 2014
- LARMAN, C. **Utilizando uml e padrões: uma introdução à análise e ao projeto orientados a objetos e ao desenvolvimento iterativo**. 3. ed. São Paulo: Bookman, 2007.
- LIMA, Adilson da Silva. **Especificações Técnicas de Software**. São Paulo: Erica, 2012.

- LUCIA, A. De e Qusef, A. 2010. **Requirements Engineering in Agile Software Development**. Journal Of Emerging Technologies In Web Intelligence. 2010.
- MACKEY, A., & Gass, S. M. **Second language research: Methodology and design**. Lawrence Erlbaum Associates Publishers. 2005.
- MARTINS, R. A. Abordagens Quantitativa e Qualitativa. In: CAUCHICK, P. A. M., **Metodologia em pesquisa em engenharia de produção e gestão de operações**. Campus: São Paulo, 2010.
- MEDEIROS, J. D. R. V, et al. **Engenharia de requisitos em projetos ágeis: Uma revisão da literatura baseada em evidências da indústria**. Divulgação Científica e Tecnológica do IFBP, 28 edição especial. João Pessoa: Principia, Dez. 2015.
- MEDEIROS, J. D. R. V, et al. **Requirements specification for developers in agile projects: Evaluation by two industrial case studies**. Information and *Software Technology*. V. 117, 2020.
- O'BRIEN, James A. **Sistemas de Informação e as decisões gerenciais na era da internet**. São Paulo: Saraiva, 2 ed., 2006.
- ORIOLE, M., Martínez-Fernández, S., Behutiye, W. et al. **Data-driven and tool-supported elicitation of quality requirements in agile companies**. *Software Qual J* 28, 931–963 (2020). <https://doi.org/10.1007/s11219-020-09509-y>
- PAETSCH, F. et al. **Requirements engineering and agile software development**. Proceedings. 12th IEEE International *Workshops* on Enabling Technologies: Infrastructure for Collaborative Enterprises. 2003.
- PARMA, L. **Success factors in managing systems projects**. Data management, v.25, n.3, março de 1987.
- PERMINOVA, O.; GUSTAFSSON, M.; WIKSTRÖM, K. **Defining uncertainty in projects - a new perspective**. International Journal of Project Management, v.26, n.1, p.73-79, 2008.
- POHL, K. **Requirements engineering: Fundamentals, principles, and techniques**. Springer. Berlin, Germany. 2010.
- PRESSMAN, R. S. **Software Engineering: A Practitioner's Approach**. 4th. ed. [S.l.]: McGrawHill Higher Education, 1997.
- \_\_\_\_\_. **Engenharia de software**. 6. ed. São Paulo: McGraw Hill, 2006.
- \_\_\_\_\_. **Engenharia de software**. 8. ed. São Paulo: McGraw Hill, 2016.
- RAMESH, B., Cao, L., Baskerville, R.: **Agile requirements engineering practices and challenges: an empirical study**. *Inf. Syst. J.* 20, 449–480, 2010.
- REGNELL, B. et al. **“An Industrial Case Study on Distributed Prioritization in MarketDriven Requirements Engineering for Packaged Software”** *Requirements Eng. J.*, vol. 6, 2001, pp. 51–62.
- RODRIGUEZ, G., WONG, L., MAURICIO, D., **A Systematic Literature Review About Software Requirements Elicitation**. Vol. 12 296–317. 2017.
- RUEDAA, S. **Requirements elicitation methods based on interviews in comparison: A family of experiments**. Information and *Software Technology*. 126. 2020.

- SAEEDA H, Dong J, Wang Y, Abid MA. **A proposed framework for improved software requirements elicitation process in SCRUM: Implementation by a real-life Norway-based IT project.** J Softw Evol Proc. 2020; e2247. <https://doi.org/10.1002/smr.2247>
- SATRIA, D., Sensuse, D. I., & Noprisson, H. 2018. **A systematic literature review of the improved agile software development.** 2017. International Conference on Information Technology Systems and Innovation, ICITSI 2017 - Proceedings, 2018-January, 94–99.
- SAVOLAINEN, J; KUUSELA; VILAVAARA, A. **Transition to agile development - rediscovery of important requirements engineering practices.** In 18th IEEE International Requirements Engineering Conference (RE), pag. 289 –294, 2010.
- SAWYER, P.; KOTONYA, G. **Software requirements. In: Guide to the software engineerin body of knowledge, swebok,** 2001. Disponível em: <[http://www.swebok.org/stoneman/trial\\_1\\_00.htm](http://www.swebok.org/stoneman/trial_1_00.htm)>. Acesso em: 21 jun. 2019.
- SCHÖN, E.-M., Thomaschewski, J., & Escalona, M. J. **Agile Requirements Engineering: A systematic literature review.** Computer Standards and Interfaces. Elsevier. <https://doi.org/10.1016/j.csi.2016.08.011>. 2017
- Schön EM., Winter D., Escalona M.J., Thomaschewski J. (2017) **Key Challenges in Agile Requirements Engineering.** In: Baumeister H., Lichter H., Riebisch M. (eds) Agile Processes in *Software Engineering and Extreme Programming*. XP 2017. Lecture Notes in Business Information Processing, vol 283. Springer, Cham
- SCHWABER, K. **Agile Project Management with Scrum.** Microsoft, 2004.
- SFETSOS, P., Angelis, L., Stamelos, I., Raptis, P. **Integrating User-Centered Design Pratices into Agile Web Development: A Case Study.** Information, Intelligence, System and Aplications (IISA), 2016, p. 13-15. July 2016.
- SOARES, H.F., et al. **Investigating the link between user Stories and documentation debt on software projects.** In: Proceedings of the 12th International Conference on Information Technology - New Generations. IEEE, 2015.
- SOMMERVILLE, I. **Engenharia de Software.** 6. ed. São Paulo: Addison Wesley, 2003.
- SOMMERVILLE, I.: **Software Engineering,** p. 122. Pearson Addison-Wesley, England. 2004.
- THAYER, R., Dorfman M. **System and software requirements engineering.** IEEE Computer Society Press, 1990.
- TURK, D., FRANCE, R., RUMPE, B., 2014. **Limitations of agile software processes.** In: Third International Conference on Extreme Programming and Flexible Processes in *Software Engineering*, XP2002, Italy, pg. 43-46, 2002. arXiv:1409.6600.
- VASCONCELOS, Isabella F. F. G.; MASCARENHAS, André. **Paradoxos Organizacionais e Tecnologia da Informação: Uma Análise Crítica da Implantação de Sistemas de Autoatendimento na Área de Gestão de Pessoas da Souza Cruz.** In Anais: XXVII Encontro da Associação Nacional de Pós-Graduação e Pesquisa em Administração, Atibaia, 2003.
- VASCONCELOS, Isabella F. F. G.; et al. **O Lado Humano da Tecnologia: um Estudo Exploratório sobre os Paradoxos Organizacionais dos Sistemas de Informação.** In Anais: XXV Encontro da Associação Nacional de Pós-Graduação e Pesquisa em Administração, Campinas, 2001.

VERSIONONE. **The 13th Annual State of Agile Report**. 2019. Disponível em: <  
<https://www.stateofagile.com/#ufh-i-521251909-13th-annual-state-of-agile-report/473508>>.  
Acesso em: 5 mar. 2020

VOSS, C.; TSIKRIKTSIS, N.; FROHLICH, M. **Case research in operation management**.  
*International Journal of Operation & Production Management*, v.22, n. 2, p. 195-219, 2002.

YIN, R. K. **Case study research: *design and methods*** 4 ed. Trousand Oaks: SAGE  
Publications, Inc., 2009

YIN, R. K. **Estudo de Caso: planejamento e métodos**. Tradução de Daniel Grassi, 5. ed.  
Porto Alegre: Bookman, 2015.

**APÊNDICE A**  
**Síntese dos estudos da RSL**

ID	Ano	Autor(es)	Citado	Tema	Método Pesquisa	Fonte	Observações importantes
1	2003	PAETSCH, F., et al.	624	<i>Requirements engineering and agile software development.</i>	Estudo empírico	Conferência	<p>Compara as abordagens tradicionais de engenharia de requisitos e o desenvolvimento ágil de <i>software</i> e apresenta as possíveis formas de como a abordagem ágil pode se beneficiar dos métodos de engenharia de requisitos.</p> <p>O estudo menciona os termos "<i>customer involvement</i>", "<i>user involvement</i>" e "<i>stakeholder involvement</i>" no mesmo parágrafo parecendo tratar do mesmo papel. O estudo conclui ainda que mesmo que os requisitos sejam levantados em sessões de grupo (proposta do estudo), não é garantida a presença dos usuários. No entanto, enfatiza que a maioria dos requisitos não funcionais devem ser conhecidos no desenvolvimento, porque podem afetar a escolha da base de dados, da linguagem de programação ou do sistema operativo. Desta forma, o estudo propõe que os métodos ágeis precisam incluir mais explicitamente o tratamento de requisitos não funcionais.</p>
2	2008	DYBÅ, T., Dingsoyr, T	2.555	<i>Empirical studies of agile software development: a systematic review.</i>	Revisão Sistemática da Literatura	Revista Científica	<p>A principal implicação para a pesquisa é a necessidade de mais e melhores estudos empíricos de desenvolvimento ágil de <i>software</i>. Para os leitores industriais, a revisão fornece um mapa de resultados, de acordo com o tema, que pode ser comparado por relevância para os seus próprios cenários e situações.</p>
3	2009	ADIKARI, et al.	60	<i>Little Design Up-Front: A Design Science Approach to Integrating Usability into Agile Requirements Engineering.</i>	Design Science	Conferência	<p>Os autores desenvolveram um quadro baseado no <i>Design Thinking</i> para apoiar as atividades de um projeto de <i>software</i> ágil num contexto real. Apresenta os resultados de dois projetos ágeis para validar a proposição de que incorporando uma perspectiva de <i>Design Centrado no Usuário (UCD)</i> no Desenvolvimento Ágil de <i>Software</i> melhora a qualidade da usabilidade e aceitação dos sistemas. Os autores desenvolveram uma abordagem chamada "<i>Little Design Up Front</i>" para integrar a perspectiva UCD no processo de desenvolvimento.</p>

4	2010	KAUTZ, K.	23	<i>Participatory Design Activities and Agile Software Development</i>	Estudo de Caso	Conferência	A pesquisa contribui para os estudos das atividades de elicitação no desenvolvimento de <i>software</i> . Fornece um estudo de caso de grande escala no âmbito de desenvolvimento ágil e concentra-se na forma como os clientes e usuários participaram na prática em atividades de desenvolvimento e <i>design</i> ágeis. O projeto investigado utilizou o método eXtreme Programação (XP). Gamificação, <i>User Stories</i> , <i>Story Cards</i> , <i>Whorkhop</i> e testes de aceitação estruturaram o envolvimento do cliente e do usuário. Os autores afirmam terem encontrado envolvimento do cliente e do usuário na concepção das atividades sob a forma de participação direta e indireta no projeto de desenvolvimento ágil. Os representantes dos clientes envolvidos desempenharam um papel informativo, funções consultivas e participativas no projeto. Isto levou a que os mesmos desempenhassem funções de empoderamento. Já os usuários tiveram a possibilidade de realizar o seu trabalho para que houvesse satisfação e de uma forma eficaz, eficiente e económica.
5	2010	RAMESH, Cao & Baskerville	308	<i>Agile requirements engineering practices and challenges: an empirical study.</i>	Estudo de Caso (Múltiplos casos)	Revista Científica	Desafios percebidos pelos autores: estimativa de custo e cronograma; arquitetura inadequada; negligência de requisitos não funcionais; acesso e participação do cliente; priorização em uma única dimensão; verificação inadequada de requisitos; documentação mínima.
6	2011	BJARNASON, Wnuk & Regnell	90	<i>A case study on benefits and side-effects of agile practices in large-scale requirements engineering.</i>	Estudo de Caso	Conferência	Desafios relativos ao planeamento da agilidade; fraca priorização de requisitos; problemas de qualidade; sistema concluído tardiamente; falta de requisitos documentados; função de cliente-representante; requisitos fracos nas primeiras iterações do projeto.
7	2012	CALEFATO, F., Damian, D. & Lanubile, F.	49	<i>Computer-mediated communication to support distributed requirements elicitations and negotiations tasks</i>	Estudo empírico	Revista Científica	Analisa a eficácia da comunicação síncrona mediada por computador nas elicitações de requisitos e nas negociações. A investigação é orientada por um <i>framework</i> que considera a eficácia de um meio de comunicação em relação à informação das necessidades de elicitação de requisitos e negociação. As conclusões apresentam provas empíricas sobre a percepção e a adequação objetiva entre a tecnologia de comunicação síncrona e as atividades de requisitos. Em primeiro lugar, a comunicação presencial nem sempre é o meio preferido para as tarefas de requisitos. Em segundo lugar, verifica-se que o desempenho do grupo não é afetado pelo meio de comunicação. Em terceiro lugar, quando os grupos interagem apenas através da comunicação baseada em texto, a base comum nas negociações de requisitos demora mais tempo, indicando que a elicitação de requisitos é a tarefa em que as ferramentas de comunicação mediadas por computador têm mais oportunidades para uma aplicação bem sucedida.

8	2012	ANITHA, P. C.; Prabhu, B.	20	<i>Integrating requirements engineering and user experience design in Product life cycle Management.</i>	Estudo empírico	Conferência	Este documento discute alguns dos conceitos errados e mitos que existem na indústria relativamente à Engenharia de Requisitos (RE) e ao Projeto de Experiência do Utilizador (UXD), e como as necessidades do usuário normalmente escapam pelas fendas sob o nome de "requisitos não funcionais". O documento destaca a necessidade de focalização na RE e UXD durante o ciclo de vida global de desenvolvimento do produto. Dois desafios encontrados são: 1. Comunicação dos requisitos dos usuários num formato que os engenheiros de requisitos podem facilmente incorporar; 2. Quebrar as necessidades dos usuários em termos tangíveis de concepção que o desenvolvimento possa facilmente implementar.
9	2014	INAYAT, Salim, Marczak, Daneva & Shamshirband	308	<i>A systematic literature review on agile requirements engineering practices and challenges.</i>	Revisão Sistemática da Literatura	Revista Científica	Apresentaram práticas e desafios encontrados durante a elicitação dos requisitos na utilização de metodologias ágeis. Dificuldades relativas à documentação mínima; disponibilidade do cliente; negligência com requisitos não funcionais (NFR's); incapacidade e concordância do cliente; limitações contratuais; mudança de requisitos e sua avaliação.
10	2015	HEIKKILA, Damian, Lassenius & Paasivaara	66	<i>A Mapping Study on Requirements Engineering in Agile Software Development.</i>	Estudo de Mapeamento	Conferência	O estudo apresenta como resultado do mapeamento, os principais desafios: Problemas com clientes ou representantes de clientes; insuficiência de formato de <i>user stories</i> ; dificuldades para confiar no conhecimento de requisitos tácitos.
11	2015	SOARES, Alves, Mendes, Mendonca & Spinola	22	<i>Investigating the link between user stories and documentation debt on software projects.</i>	Revisão Sistemática da Literatura	Conferência	O resultado da investigação apresenta desafios na identificação de requisitos não funcionais; falta de informação; volatilidade de requisitos; definição de requisitos; dependência entre requisitos; previsão de impactos de mudanças; dependência do usuário; comunicação e colaboração com usuários; validação de requisitos.

12	2015	HARBERS, M., Detweiler, C., Neerinx, M. A	16	<i>Embedding Stakeholder Values in the Requirements Engineering Process.</i>	Estudo de Caso	Conferência	Estudaram a aplicação de <i>workshop</i> para o processo de elicitação de <i>user stories</i> . O objetivo deste <i>workshop</i> é inserir os valores das partes interessadas no processo de engenharia de requisitos. Portanto, partes interessadas diretas e indiretas devem ser identificadas e em uma segunda etapa, os valores de cada grupo de partes interessadas são revelados. Na sequência, uma situação para cada valor é fornecida e as necessidades das partes interessadas nessa situação são analisadas e descritas. Os resultados mostram que <i>user stories</i> baseados em valores são considerados valiosos para o usuário final, mas complicados para estimar e testar. Este documento conclui, portanto, que o este tipo de <i>workshop</i> é um método promissor para a incorporação de valores no processo de Engenharia de Requisitos, mas que os <i>user stories</i> baseados em valores precisam ser traduzidas em Casos de Uso para torná-los adequados ao planejamento e organização das atividades de desenvolvimento.
13	2015	BELLUCCI, A., et al.,	14	<i>Extreme Co- design: Prototyping with and by the User for Appropriation of Web- connected</i>	Estudo de Caso	Conferência	Os autores descrevem um projeto de protótipo de campo em que as ferramentas protótipo ligadas à web são co-desenhadas e programadas semanalmente com e pelo usuário. A abordagem é chamada de <i>Co-design</i> Extrem, onde a forma como o <i>design</i> é inscrito em sessões de Programação Extrema com ciclos rápidos de utilização, <i>design</i> e desenvolvimento permitem a exploração extensiva e vivência de cenários de apropriação. Os autores discutem esta abordagem de sucesso como uma tática de <i>design</i> para produtos inacabados, para promover a criatividade dos usuários finais através de uma utilização situada e mostrar como a metodologia ágil XP e a implantação dessa abordagem suportaram interações significativas entre o <i>designer</i> e o utilizador que resultaram no avanço da concepção inicial.

14	2016	BJARNASON, et al.	39	<i>A multi-case study of agile requirements engineering and the use of test cases as requirements</i>	Estudo de Caso (Múltiplos casos)	Revista Científica	<p>O estudo investigou a prática ágil de utilizar casos de teste como requisitos para compreender como os casos de teste podem suportar as principais atividades de requisitos, e como esta prática varia.</p> <p>Foi realizado um estudo de caso iterativo em três empresas e recolhido dados através de 14 entrevistas e dois grupos focais.</p> <p>A utilização de casos de teste como requisitos coloca tanto benefícios como desafios quando se trata de elucidar, validar, verificar e gerir requisitos. No entanto, a utilização dos casos de testes para elicitação de requisitos complexos, como a usabilidade e os requisitos de qualidade são um desafio encontrado para as três empresas avaliadas. Isto deve-se à complexidade destes requisitos e à necessidade da competência adicional dos papéis envolvidos. As três empresas descreveram os desafios da documentação de interações complexas, por exemplo, entre componentes, como casos de teste. Para a empresa A e C, o tempo (relacionado ao custo) e a dificuldade de expressão dificultam a elicitação por meio dos casos de teste. A empresa B também mencionou que os conhecimentos técnicos limitados afetam a capacidade do cliente para discutir os requisitos de qualidade por meio de casos de teste. Do mesmo modo, a captura de requisitos complexos com casos de teste de aceitação é um desafio, mencionado para as interações dos usuários e os requisitos de qualidade.</p>
15	2016	SFETSOS, P., et al.	6	<i>Integrating User-Centered Design Practices into Agile Web Development: A Case Study.</i>	Estudo de Caso	Conferência	<p>Esta investigação analisa uma forma para melhorar o produto do desenvolvimento. Fox, et. al também fez esta pesquisa que, posteriormente, foi conduzida por Sfetsos et al. O resultado propõe um método híbrido executado dentro do processo de sprint, mas não se centra-se unicamente na melhoria do sprint, mas sim no produto final. A ideia principal da melhoria é maximizar a usabilidade do produto. A ideia é incorporar a UCD no sprint do Agile. Nesta abordagem, o UCD centrou-se na experiência do usuário (UX) conduzida em cada sprint.</p>
16	2017	SCHÖN, Winter, Escalona & Thomaschewski	25	<i>Key Challenges in Agile Requirements Engineering.</i>	Método Delphi	Conferência	<p>O estudo apresenta alguns desafios, como: dificuldade em manter o foco no propósito do projeto após muitas iterações; desafios em escrever os requisitos de forma incremental; dificuldade de envolver os <i>stakeholders</i> e usuários em todas as iterações.</p>
17			144	<i>Agile Requirements Engineering: A systematic literature review.</i>	Revisão Sistemática da Literatura	Revista Científica	<p>RSL com uma avaliação extensiva da qualidade dos estudos incluídos. 27 artigos relevantes. Insights profundos sobre os seguintes aspectos do Agile RE: envolvimento de partes interessadas e usuários, elicitação, perspectiva do usuário, metodologias integradas, compreensão partilhada, documentação e Requisitos Não Funcionais (NFR). Agile RE é um campo de investigação complexo com influências inter-funcionais. Este estudo contribui para o corpo de conhecimento do desenvolvimento de <i>software</i>, avaliando o envolvimento das partes interessadas e dos usuários em ER ágeis, fornecendo metodologias que tornam a ASD mais centrada no ser humano e dando uma visão geral da gestão de requisitos na ASD.</p>

18	2017	SATRIA, D., Senseuse, D. I., & Noprisson, H.	3	<i>A systematic literature review of the improved agile software development.</i>	Revisão Sistemática da Literatura	Conferência	<p>Muitos documentos desta investigação centram-se na forma de melhorar o produto do desenvolvimento de <i>software</i>, incluindo o problema da usabilidade, que constitui uma desvantagem do desenvolvimento ágil de <i>software</i>.</p> <p>Outra desvantagem do desenvolvimento ágil de <i>software</i> é a falta de documentação. Esta falta de documentação por vezes fez com que o <i>software</i> não satisfizesse as expectativas dos usuários.</p> <p>Outro ponto fraco desta metodologia é criar uma boa comunicação entre os desenvolvedores e as partes interessadas, que se tornou um tópico de pesquisa para resolver este problema.</p>
19	2017	RODRIGUEZ, G., et al.	22	<i>Systematic Literature Review About Software Requirements Elicitation</i>	Revisão Sistemática da Literatura	Revista Científica	<p>Este documento propôs identificar a situação atual dos últimos pesquisas relacionadas à elicitação de requisitos de <i>software</i> através de estrutura para revisão de literatura, a fim de responder às seguintes questões: P1) Que aspectos foram cobertos por diferentes propostas de elicitação de requisitos? P2) Quais atividades da elicitação ? e P3) Que fatores influenciam a elicitação e como? Foi realizada uma análise cruzada do resultado. Uma RSL de 35 artigos considera não haver propostas consistentes sobre fatores correspondem a "Determinar Fontes" e "Definir Técnica". Portanto, recomenda o aprofundamento nas pesquisas visando uma proposta que cubra as atividades no processo de levantamento de requisitos, tais como "Adquirir conhecimento de domínio", "Definir uma técnica adequada", "Determinar a fonte dos requisitos".</p>
20	2019	MARTINS, H. F., et al.	1	<i>Design Thinking: Challenges for Software Requirements Elicitation</i>	Triangulação: Revisão Sistemática da Literatura; Observação; Questionário; e Entrevista.	Conferência	<p>Métodos ágeis se adaptam bem às equipes de desenvolvimento de <i>software</i> nas atividades de levantamento de requisitos. O <i>Design Thinking</i> tem sido utilizado como uma técnica de levantamento de requisitos e imersão nas áreas de processo, o que aproxima o cliente da equipe de projeto de <i>software</i> e permite a criação de melhores projetos. Com o uso da triangulação de dados, este artigo traz uma revisão bibliográfica que recolhe os desafios na elicitação de requisitos de <i>software</i> em metodologias ágeis e o uso do <i>Design Thinking</i>. O resultado deu lugar a um estudo de caso em um projeto de organização pública brasileira, através de questionário de <i>workshop</i> de usuários com 20 itens, aplicados durante o estudo, a fim de identificar a prática do <i>Design Thinking</i> neste contexto. O artigo propõe uma visão geral de 13 desafios estudados, dos quais oito apresentaram forte evidência de contribuição (envolvimento das partes interessadas, definição e validação de requisitos, cronograma, planejamento, detalhes de requisitos e priorização e interdependência), três apresentaram evidência parcial de contribuição e dois não foram elegíveis para conclusões (requisitos não-funcionais, uso de artefatos e mudança de requisitos). O principal resultado deste trabalho é apresentar uma análise da utilização do <i>Design Thinking</i> para ver se este se enquadra corretamente para ser utilizado como meio de resolver os desafios da elicitação dos requisitos de <i>software</i> quando se utilizam métodos ágeis.</p>

21	2019	AINHOA, A., et al.	1	<i>Leveraging creativity in requirements elicitation within agile software development: A systematic literature review</i>	Revisão Sistemática da Literatura	Revista Científica	<p>Revisões bibliográficas sistemáticas, com especial enfoque em engenharia de <i>software</i>, elicitação de requisitos e metodologias ágeis (fornecendo detalhes de <i>Scrum</i>, <i>Extreme Programming</i> e <i>Kanban</i>). Investigou-se também a criatividade, primeiro de um ponto de vista teórico, e em segundo lugar no que diz respeito à aplicação da criatividade no <i>design</i> (com especial enfoque no <i>Design Thinking</i>) e elicitação de requisitos.</p> <p>Sugere:</p> <ul style="list-style-type: none"> <li>- Utilizar outra combinação de palavras-chave, como as relacionadas com a inovação e o envolvimento dos usuários.</li> <li>- Para se obter uma visão detalhada da situação atual na indústria, deve ser criado um questionário a ser utilizado para analisar as circunstâncias específicas dos profissionais e dos projetos em termos de localização e recursos, bem como para contrastar os resultados desta investigação com a situação atual da indústria.</li> </ul>
22	2020	MEDEIROS, J. et al.	0	<i>Requirements specification for developers in agile projects: Evaluation by two industrial case studies</i>	Estudo de Caso (Múltiplos casos)	Revista Científica	<p>O estudo apresenta como resultado a conclusão de que uma atividade inadequada de especificação de requisitos atua como catalisador de outros problemas, como a baixa produtividade da equipe e a dificuldade na manutenção do <i>software</i>. Embora a abordagem ágil tenha crescido nos últimos anos, esta investigação apontou várias limitações relativas às suas atividades de engenharia de requisitos, como a Especificação de Requisitos de <i>Software</i> (SRS) fornecida em alto nível e orientada para o cliente, a falta de informação necessária para realizar atividades de concepção e a baixa disponibilidade do cliente. Para ultrapassar estas questões, foi proposta a abordagem RSD (Especificação de Requisitos para Desenvolvedores) para criar um SRS que forneça informação mais próxima das necessidades de desenvolvimento. Além disso, este estudo demonstra que as revisões bibliográficas existentes identificam uma procura de estudos mais empíricos sobre a atividade de especificação de requisitos.</p> <p>Esta avaliação foi realizada através de dois estudos de casos industriais realizados utilizando uma concepção de casos múltiplos, focando os engenheiros de <i>software</i> como unidade de análise. Os dados foram recolhidos durante 15 meses a partir de documentos, observações e entrevistas. Os dados foram triangulados, analisados e sintetizados. A avaliação indicou a necessidade de priorização e categorização dos critérios de aceitação, uma ferramenta para criar, pesquisar e rastrear os artefatos, e obter testes de aceitação a partir dos critérios de aceitação. Por outro lado, os resultados mostraram que as práticas utilizadas para especificar os requisitos utilizando a abordagem RSD têm potencial para produzir um SRS mais objetiva e adaptada à equipe de desenvolvimento.</p>
23	2020	SAEEDA, et al.	0	<i>A proposed framework for improved software requirements</i>	Pesquisa-Ação	Revista Científica	<p>Propõe um framework que unifica o <i>Joint Requirements Document</i> (JRD), MIND MAPPING e <i>SCRUM</i> para incrementalmente identificar, implementar, evoluir e gerenciar os requisitos do sistema ao longo do ciclo de desenvolvimento de <i>software</i>.</p>

				<i>elicitation process.</i>			
24	2020	RUEDAA, S., et al.		<i>Requirements elicitation methods based on interviews in comparison: A family of experiments</i>	Experimento	Revista Científica	<p>Contexto do estudo: Há vários métodos para obter requisitos através de entrevistas entre um usuário final e uma equipe de desenvolvedores de <i>software</i>. A escolha do melhor método neste contexto é geralmente sobre as preferências subjetivas dos desenvolvedores ao invés de razões objetivas. O trabalho foi desenvolvido em razão da falta de avaliações empíricas de métodos que ajudem a desenvolvedores a escolher o método mais adequado.</p> <p>Este estudo projeta e conduz uma família de experimentos para comparar três métodos para atender às necessidades: Entrevistas não estruturadas, onde não há protocolo ou artefatos específicos; JAD, onde cada membro da equipe de desenvolvimento tem um papel específico; Prototipação, onde os desenvolvedores contrastam os requisitos com o usuário final através de protótipos.</p> <p>A experiência consistiu e 4 rodadas de entrevistas entre estudantes que desempenham o papel de desenvolvedores e um instrutor que desempenha o papel de cliente. Tiveram que elaborar um documento de especificação de requisitos como resultado das entrevistas. Foram recrutados 167 sujeitos em 4 rodadas em 3 anos. Os assuntos foram reunidos em equipes de desenvolvimento de 6 desenvolvedores, no máximo, e cada equipe era uma unidade experimental.</p> <p>Resultados: Os autores encontraram algumas diferenças significativas. A Prototipagem produz os melhores resultados para obter o maior número possível de requisitos, o JAD requer o maior tempo para relatar os requisitos e o menor número de sobreposições, e Entrevistas não estruturadas produzem a maior sobreposição e o menor tempo para relatar as exigências.</p> <p>Conclusões: A Prototipagem é a mais adequada para elucidar as exigências funcionais, a JAD é a mais adequada para requisitos não funcionais e para evitar sobreposições, Entrevistas Não Estruturadas é a mais rápida, mas com pouca qualidade nos resultados.</p>

25	2020	Oriol, M. et al.	0	<i>Data-driven and tool-supported elicitation of quality requirements in agile companies</i>	Pesquisa-Ação	Conferência	Os requisitos de qualidade (QR) são um artefato-chave necessário para garantir a qualidade e o sucesso de um sistema de <i>software</i> . Apesar da sua importância, os QR raramente recebem o mesmo grau de atenção que os seus equivalentes funcionais em projetos de desenvolvimento de <i>software</i> ágil (ASD). Além disso, a fase de envolver o usuário não é totalmente explorada, ou é mesmo negligenciada, nas atividades de elicitação de QR. Neste trabalho, é apresentado uma abordagem baseada em dados para a geração e documentação semi-automática de QR no contexto da ASD. A abordagem baseia-se na declaração de limiares sobre questões relacionadas com a qualidade, cuja violação desencadeia alertas definidos pelo usuário. Estes alertas são utilizados para consultar um catálogo de padrões de QR que são apresentados à equipe da ASD através de um painel de controle que implementa várias técnicas de análise. Uma vez selecionados, os padrões geram os QR's, que são documentados e armazenados no <i>backlog</i> . Ao longo de 1 ano, quatro empresas de dimensão e perfil diferentes seguiram esta abordagem e implantaram a plataforma nas suas instalações para gerar QR's semi-automáticos em vários projetos. Foram utilizados instrumentos de medição padronizados para suscitar a percepção de 22 profissionais relacionados à utilização da ferramenta. As análises quantitativas e qualitativas produziram resultados positivos, ou seja, a percepção dos profissionais em relação à compreensibilidade, fiabilidade, utilidade e relevância da ferramenta foi positiva. A pesquisa conclui que os resultados mostram potencial para a adoção futura desta abordagem na elicitação dos QR nas empresas ágeis.
----	------	------------------	---	--	---------------	-------------	--

## APÊNDICE B

### Desafios do processo de elicitação (baseado na RSL)

Desafios	Ocorrências	Categoria	Qtde desafio pr Categoria
Expectativa do cliente não atendida	11	Cliente	44
Interação inadequada entre stakeholders e equipe técnica	19		
Usuários não conseguem identificar suas necessidades	5		
Indisponibilidade de acesso ao cliente	9		
Documentação insuficiente	11	Documentação	15
Dificuldade em transferir conhecimento	3		
Dificuldade na conexão entre os requisitos	1		
Validação dos requisitos sem a perspectiva do usuário	1	Processo	5
Requisitos não acordados com a equipe de desenvolvimento	2		
Pouca disponibilidade de ferramentas específicas para RE ágil	2		
Habilidades técnicas requeridas para compreender os User Stories	2	Técnicas	22
Requer refinamento sucessivo dos requisitos	4		
Dependência de forte comunicação com o cliente	7		
Use Case apresentam muita informação	5		
Voltadas para o desenvolvimento e não para o Cliente	2		
Cenários difíceis de interpretar	2		
Manter requisitos atualizados	15	Gestão	37
Detalhamento tardio dos requisitos	7		
Baixa motivação da equipe	1		
Constante repriorização dos requisitos	7		
Conflitos quando os requisitos provém de fontes distintas	4		
Equipes distribuídas	3		
Requisitos não confiáveis e instáveis	3	Qualidade	12
Deficiência na identificação dos RNF	9		

## APÊNDICE C

### PROJETO PILOTO

O projeto piloto tem a intenção de reproduzir em escala menor os métodos planejados para o estudo de caso e, por este meio, testar a necessidade de adequação dos instrumentos e procedimentos contidos no método de modo a calibrá-lo para a coleta de dados definitiva.

Para este estudo piloto, foi analisado um caso similar ao que se estudou na fase definitiva, todavia, não foi possível fazer o acompanhamento em tempo real, considerando a situação de pandemia de coronavírus (COVID19<sup>10</sup>) que nos acomete neste mesmo período de análises (março de 2020 a janeiro de 2021). Sendo assim, dos instrumentos de pesquisa definidos para a coleta definitiva, não se pode testar a “observações participativas” uma vez que se tratou de um projeto já finalizado. Portanto, foram consideradas as análises documentais e as entrevistas semiestruturadas.

#### 4.3.1 Participantes e contexto

O projeto piloto ocorreu dentro de uma empresa de tecnologia de informação produtora de *softwares* do interior do estado de São Paulo cujos projetos são voltados para a área empresarial. O projeto em questão abordou o estudo de um projeto de *software* desenvolvido em 2019 cujo objetivo foi criar uma plataforma para controle de cursos EaD fornecidos pela empresa contratante a seus funcionários como forma de educação continuada. A empresa cliente é uma multinacional no setor metalúrgico e a filial estudada situa-se em Santa Bárbara D’oeste, estado de São Paulo.

Quatro colaboradores da empresa de *software* que estiveram envolvidos no projeto em questão foram convidados e aceitaram participar do estudo piloto. Vale ressaltar que esta mesma empresa não foi escolhida para a coleta definitiva a fim de reduzir o viés da pesquisadora, pois esta possui vínculo de prestação de serviço com a mesma. Os colaboradores escolhidos trabalham em projetos ágeis, em média, há 10 anos, com idade entre 30 e 47 anos. Destes quatro colaboradores, 1 é analista de requisito, 1 é desenvolvedor, 1 é analista de qualidade e o outro, *Scrum Master*. Por parte da empresa cliente, aceitaram o convite apenas o Diretor de RH (figura que representou o cliente no projeto) e 2 usuários. Essa escolha foi baseada no fato de que o estudo piloto deve, preferencialmente, ocorrer com uma população semelhante à da pesquisa definitiva (MACKEY; GASS, 2005). Sendo assim, após aprovação do projeto de pesquisa pelo responsável

---

<sup>10</sup> Pandemia COVID-19: Declarada em 11 de março de 2020 pela Organização Mundial da Saúde. O vírus (SARS-CoV-2) causa síndrome respiratória aguda grave 2. A doença foi identificada pela primeira vez em Wuhan, na província de Hubei, República Popular da China, em 1 de dezembro de 2019. Fonte: OMS

pela empresa, a pesquisadora começou a acessar em março de 2020 os documentos do projeto, com a qual havia a intenção de conduzir o estudo piloto. Neste mesmo período iniciariam as entrevistas para a coleta dos dados, no entanto, em função da pandemia, estas atividades foram postergadas e retomaram em setembro do mesmo ano na modalidade *online por* meio da Plataforma *Google Meet*. Os colaboradores e a empresa cliente receberam informações, tanto na forma oral como escrita, relacionadas à pesquisa. Foi esclarecido que a participação era completamente voluntária, bem como não haveria remuneração para participação e que a pesquisa ocorreria no horário do trabalho em um único encontro. Após o esclarecimento de eventuais dúvidas, foi informado que as entrevistas não seriam gravadas e suas identidades seriam confidenciais. Como forma de incentivo para a participação ficou estabelecido que a pesquisadora retornaria no primeiro semestre de 2021 e proveria um *feedback* para as empresas participantes.

#### 4.3.2 Procedimentos para coleta de dados

Considerando a distinção na linha do tempo em relação ao estudo piloto (caso passado) e o estudo de caso (acompanhamento em tempo real), o desenho de coleta de dados do projeto piloto, ver Quadros 29 e 30, foi projetado para refletir uma situação do passado. Para a pesquisa definitiva, houve alterações que adaptaram as questões para o acompanhamento em tempo real do projeto, conforme Quadros 19 e 20.

**Quadro 29** – Mapeamento das questões para as entrevistas no projeto piloto

	EMPRESA DESENVOLVEDORA				EMPRESA CLIENTE	
	DESENV	QA	LP	AR	CLIENTE	USUÁRIO
ANTES			Q1	Q1		Q3
			Q2	Q2		
				Q3		
DURANTE	Q1	Q1	Q4	Q4	Q4	Q4
	Q4	Q4	Q5	Q5	Q5	Q5
	Q5	Q5	Q6	Q6	Q6	Q10
	Q7	Q7	Q7	Q7	Q7	
	Q8	Q8	Q8	Q9	Q9	
	Q9	Q9	Q9			
HOMOLOGAÇÃO	Q13	Q13	Q11	Q12	Q11	Q10
	Q14	Q14	Q12	Q13	Q12	Q13
			Q13	Q14	Q13	Q14
			Q14		Q14	Q15
						Q16

**Legenda:**

Desenv.: Analista desenvolvedor

QA: Analista de Qualidade

LP: Líder de Projeto, dependendo da metodologia ágil usada este papel pode receber títulos distintos.

AR: Analista de Requisito, principal responsável pela elicitação.

Fonte: Elaboração própria

**Quadro 30** – Questionário aplicado no projeto piloto (continua)

	QUESTÕES	PROPÓSITO
Q1	Já havia trabalhado com este cliente? Se sim, o projeto anterior visava atender os mesmos usuários?	Busca observar/analisar se o envolvimento dos usuários está influenciado pelo histórico de projetos, impactando a P1.
Q2	Qual abordagem foi usada para entender as necessidades deste projeto?	Em observação à P1, busca conhecer se fazem uso de ferramentas que apoiam o processo de elicitação.
Q3	Você teve conhecimento do <i>software</i> que construído antes de sua entrega final?	Questão de impacto para P1 e P4 trazendo informações sobre o grau de envolvimento dos usuários.
Q4	Descreva, em sua perspectiva, qual foi o objetivo do projeto e quais foram as motivações para o projeto acontecer?	Informação importante na análise comparativa a fim de identificar a conformidade ou discrepância entre as respostas e a real necessidade do projeto para o usuário (P1 e P4).
Q5	O projeto tratou de novas funcionalidades ou atualização para as que já existiam? Se já existiam, como funcionam antes?	Busca compreender se o envolvimento dos usuários é anulado pelo cliente em função do tipo do projeto, impactando a P4.
Q6	Em algum momento o usuário foi consultado, mas estava indisponível/ocupado, dificultou o acesso ou demorou no retorno?	Informação de forte impacto para a P2, pois de acordo com Ramesh, Cao & Baskerville (2010) envolver os usuários na Engenharia de Requisitos usando abordagens ágeis para o envolvimento de <i>software</i> depende de muitos fatores, tais como disponibilidade destes, especialmente nas fases iniciais do projeto.
Q7	Quem eram os usuários do <i>software</i> ? Você os conheceu?	Questão de impacto para P1 e P4 trazendo informações sobre o grau de envolvimento dos usuários.
Q8	Em algum requisito houve dificuldade para compreender o que se pedia?	Esta questão faz-se necessária para validação da P3 e confirmação dos achados na pesquisa de Medeiros, J. et al. (2020) indicando que os requisitos na abordagem ágil (considerando o forte envolvimento do usuário) são insuficientes e inadequados para serem usados pela equipe de desenvolvimento, tornando o desenvolvimento mais difícil em um estágio posterior.
Q9	Em situações de dúvidas/problemas em relação aos requisitos do projeto, qual meio usado para buscar resposta/solução?	Questão de impacto para P1, P2 e P4 trazendo informações sobre o grau de envolvimento dos usuários, bem como de sua disponibilidade de acesso.
Q10	Você foi consultado a respeito do novo <i>software</i> ?	Questão de impacto para P1, P2 e P4 trazendo informações sobre o grau de envolvimento dos usuários, bem como de sua disponibilidade de acesso.
Q11	Como foram colhidas e tratadas as percepções dos usuários?	Informação de forte impacto para a P4, pois busca identificar se os usuários são consultados ou apenas o cliente é contatado.

Q12	Os usuários parecem ter entendido a proposta do <i>software</i> ?	Questão de impacto para P1 e P4 uma vez que, segundo Adikari et al. (2016) e Steftos et al. (2016), a compreensão do <i>software</i> está relacionada ao grau de envolvimento destes no levantamento dos requisitos.
Q13	Você considera este um projeto de sucesso?	Questão de impacto para P1 e P4 uma vez que, segundo Adikari et al. (2016) e Steftos et al. (2016), o aceite do <i>software</i> está relacionado ao grau de envolvimento destes no levantamento dos requisitos.
Q14	Houve solicitações de adequação?	Questão de impacto para P1, P3 e P4 uma vez que um alto volume de adequações pode ser resultado do baixo envolvimento dos usuários e/ou na dificuldade de compreensão dos requisitos por parte dos desenvolvedores.
Q15	Está confortável com a proposta do novo <i>software</i> ?	Questão de impacto para P1 e P4 uma vez que, segundo Adikari et al. (2016) e Steftos et al. (2016), os usuários tendem a familiarizar com o <i>software</i> quando envolvidos no levantamento dos requisitos.
Q16	Você consegue utilizar o <i>software</i> por completo e este atende à todas as necessidades da tarefa?	Questão de impacto para P1 e P4 uma vez que o envolvimento dos usuários no levantamento dos requisitos contribui para que o <i>software</i> seja mais próximo de suas reais necessidades.
Q17	Haveria sugestão de melhoria/adequações?	Questão de impacto para P1, P3 e P4 uma vez que um alto volume de adequações pode ser resultados do baixo envolvimento dos usuários e/ou na dificuldade de compreensão dos requisitos por parte dos desenvolvedores.

### 4.3.3 Procedimentos de análise dos dados

Em função da pandemia, as entrevistas foram *online* e gravadas por meio da Plataforma *Google Meet* e, posteriormente, as mensagens foram transcritas para análise. Para o piloto, planejou-se que os dados coletados na pesquisa seriam analisados em termos do grau de conhecimento por parte da empresa desenvolvedora a respeito das tarefas dos usuários de modo a refletir com similaridade o que se buscou na pesquisa definitiva.

Já a análise de documentos buscou avaliar os artefatos que armazenaram os requisitos para o desenvolvimento do *software* e então verificar a possibilidade e viabilidade de comparar os achados com os resultados das entrevistas e com o *software* produzido.

### 4.3.4 Conclusão do projeto piloto

Os dados qualitativos provindos: (1) das respostas dos participantes aos questionários, (2) da análise dos documentos serviram para validar esses instrumentos, uma vez que as informações esperadas puderam ser apreendidas. Logo, para a pesquisa final estes foram mantidos. No entanto, algumas considerações provindas deste estudo piloto poderão refletir em adequações para a

pesquisa definitiva. Na sequência, apresenta-se um resumo das principais mudanças que o desenho da pesquisa definitiva sofreu a partir dos resultados do estudo piloto.

1) Em relação a análise de documentos:

Para este instrumento, ficou claro a necessidade de planejar um tempo para seleção criteriosa dos documentos a serem considerados, bem como para o aprendizado de ferramentas usadas para o gerenciamento do projeto. Em se tratando de projetos ágeis, as empresas tendem a não criar muitos documentos e, desta forma, algumas informações importantes podem ficar espalhadas entre as ferramentas gerenciais usadas nos projetos. Para este estudo piloto, por exemplo, muitas informações foram extraídas da ferramenta Jira que dá suporte, principalmente, para a rastreabilidade de *bugs*. Logo, é necessário repensar em um tempo para conhecer a organização dos documentos do projeto, bem como interagir com as ferramentas gerenciais e de apoio que serão utilizadas no estudo definitivo.

2) Em relação às instruções para entrevistas:

Apesar das instruções para os questionários explicitarem as definições dos termos, muitos dos participantes misturaram os conceitos de cliente e usuário, bem como pareceram não entender a distinção entre participação e envolvimento. Dessa forma, esse resultado foi positivo, pois serviu como aprendizado, já que os termos serão definidos usando palavras que sejam mais bem compreendidas pelos colaboradores e, principalmente, pela classe operária (que também poderá ser “usuário” no estudo definitivo).

3) Em relação à entrevista:

Com relação a análise das mensagens obtidas por meio das entrevistas, um fato não esperado que ficou perceptível é que, de uma maneira geral, as mensagens gravadas possuíam um baixo nível de complexidade. Esse fato provavelmente deve-se ao fato de que profissionais desta área (exatas/desenvolvimento de *software*) tendem a ser muito objetivos e constroem frases curtas para as respostas, como, por exemplo, na mensagem gravada pelo analista de qualidade como “Sim” e “Não” apenas. O mesmo ocorreu com os usuários que, neste estudo, se tratava de operários metalúrgicos e não pareciam conseguir estender as respostas.

A complexidade das respostas previa verificar o nível de envolvimento dos usuários no projeto, mas com base nas mensagens do estudo piloto ponderou-se que essa variável não apresentaria valor significativo se mantivessem respostas curtas e muito objetivas e, portanto, para o estudo definitivo, as questões deverão ser acrescidas de indagações que provoquem os

entrevistados (“como”, “por que”, “quando”) na tentativa de obter extensão na elaboração das respostas. As questões a serem adaptadas serão:

Ao usuário: “Você tem conhecimento de um *software* que está sendo construído para auxiliar nas suas atividades atuais?”. Para esta questão será acrescentado: “Se sim, como tomou conhecimento do projeto do novo *software*?”.

A todos: “Conhece quais foram as motivações para o projeto acontecer?”, será acrescentado: “ Se sim, quais foram?”.

A todos: “O projeto tratou de novas funcionalidades ou atualização para as que já existiam?”, será acrescentado: “Se já existiam, como funcionam antes?”.

#### 4) Em relação a manipulação das respostas:

Neste passo vale ressaltar que os objetivos da pesquisa (de entender como o usuário é envolvido no processo, bem como de identificar se o cliente anula o papel do usuário no processo de elicitação) não deveriam ter sido divulgados com tamanha clareza no início das entrevistas, pois através deste estudo piloto ficou evidente que, exceto por parte dos usuários, os entrevistados apresentaram mudança de comportamento na tentativa de qualificar o processo de elicitação. Esta percepção foi possível graças à comparação entre o retorno das entrevistas e as análises dos documentos do projeto, uma vez que a análise documental não refletia a veracidade das informações repassadas nas entrevistas, haja visto, por exemplo, o alto volume de solicitações de mudanças por parte da empresa cliente após o período de homologação.

Afora os já citados benefícios na afinação do método da pesquisa, um ponto de mudança positivo que ficou evidente com a condução do estudo piloto foi a experiência que esse procedimento possibilitou a uma pesquisadora iniciante. A possibilidade de efetuar uma pesquisa qualitativa como essa tem sido ao mesmo tempo um grande desafio e um enorme aprendizado. É por meio da busca pela excelência na condução desse projeto de pesquisa, seguindo parâmetros rigorosos, que ocorre a maturação de um pesquisador. Sendo assim, o estudo piloto extrapola a mera função de afinar instrumentos e método, agindo como catalisador para refinar não somente essas ferramentas, mas também prepara aquele/a que as conduz, já que o/a pesquisador/a terá mais confiança e experiência para conduzir a pesquisa definitiva, e, possivelmente, futuras pesquisas.

E ainda, além da calibragem do método de pesquisa e da aprendizagem acumulada para a pesquisadora, o estudo piloto trouxe alguns resultados preliminares importantes que valem ser considerados nesta fase, uma vez que não foram levantados na fase da RSL. Estes achados mostram que o desafio de envolver o usuário final acontece, principalmente, devido à falta de processos, como:

- processos de aquisição de conhecimento e partilha de conhecimentos que permitem a exploração das necessidades de todas as partes interessadas, incluindo usuário final;
- processos iterativos que permitem a reformulação deste entendimento ao longo de todo o projeto; e
- processos eficazes de comunicação e coordenação que suportam os dois primeiros tipos de processos supracitados.

De um lado, a interação entre os desenvolvedores e analistas de requisitos, do outro, os interessados que conhecem os requisitos do sistema (por exemplo, a comunidade empresarial, incluindo usuários finais), ambos sofrem de problemas comuns de elicitação de requisitos.

Em primeiro lugar, os analistas de requisitos têm menos oportunidades de elicitar conhecimentos relevantes devido às múltiplas camadas de interessados. Como mostra a Figura 12, a comunicação entre analistas de requisitos e usuários finais acontece, geralmente, por canais intermediários (no gerenciamento ágil, por meio do *Product Owner*<sup>11</sup>, e este, acessa os usuários por outros canais ainda). Além disso, os membros da equipe de desenvolvimento (como *Scrum Master*<sup>12</sup>, Arquitetos, Devs<sup>13</sup> e QA<sup>14</sup>) têm uma chance muito maior de introduzir interpretações errôneas das necessidades das partes interessadas em cada nível de comunicação devido aos ruídos que permeiam as transferências das mensagens (como fatores culturais, linguagem ou estilos de comunicação).

Em segundo lugar, a partilha e integração de conhecimentos entre locais e grupos funcionais podem tornar-se problemáticas. Os usuários e supervisores podem não partilhar informação ricas em detalhes (como padrões e políticas da empresa, por exemplo) por considerar informações usuais e não relevantes para o sistema.

---

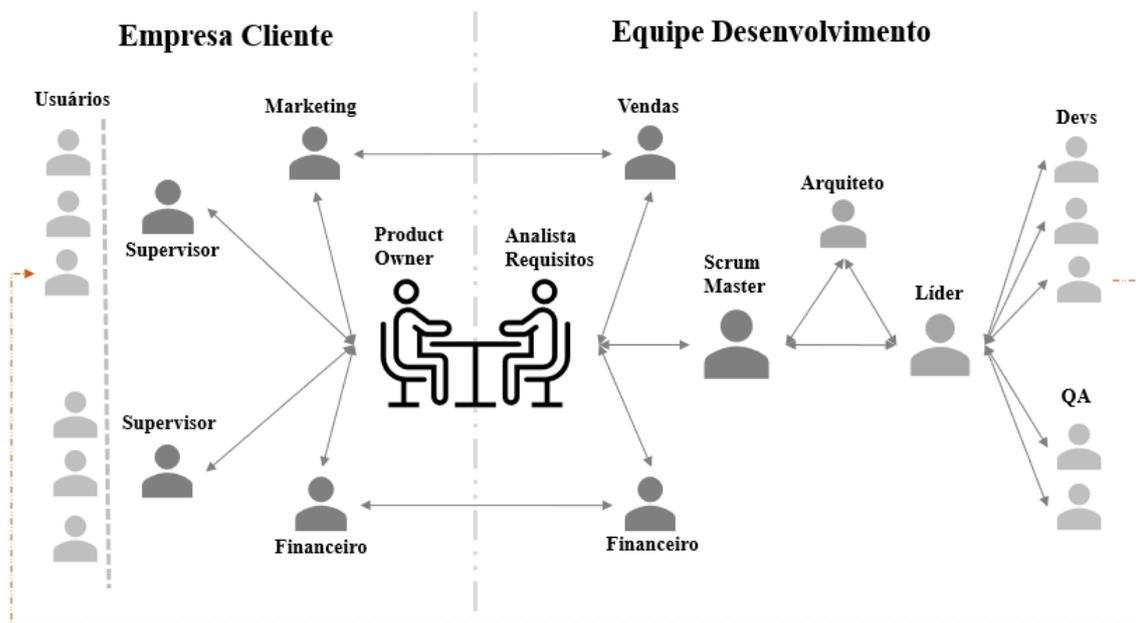
<sup>11</sup> *Product Owner*: membro de uma equipe de desenvolvimento ágil (*Scrum* ou alguma técnica similar) para definir e priorizar as funcionalidades de um produto ou projeto. Ele é responsável por manter a integridade conceitual das novas funcionalidades, bugs ou melhorias, para que essas sigam uma visão definida para o produto ou projeto. Além disso, ele também é responsável pela qualidade final das entregas, sendo o único que deve ter poder de aceitar funcionalidades como concluídas.

<sup>12</sup> *Scrum Master*: Responsável por fazer a ligação entre a equipe e o Analista de Requisitos e/ou *Product Owner*, de forma a resolver quaisquer empecilhos no processo.

<sup>13</sup> Devs: Equipe de desenvolvedores do produto.

<sup>14</sup> QA: Analistas de qualidade que certificam o produto em suas funcionalidades e aspectos não-funcionais, bem como a aderência do produto ao requisitado.

**Figura 12** – Ciclo de comunicação para elicitação de requisitos no desenvolvimento ágil



**Fonte:** Elaboração própria.

Neste sentido, Land (1982), antes mesmo do manifesto ágil, reforça o fato de que usuários e analistas possuem diferentes ideologias, perspectivas, objetivos e interesses e, além disso, há também a dificuldade em motivar os usuários e o conflito entre eles próprios, uma vez que possuem objetivos diferentes entre si e, sempre que ajudam a definir sistemas, procuram defender seus interesses. Isto provoca também a omissão de informações com vistas a garantir o status próprio (a citar, o medo do usuário com o novo sistema).