

CLAUDIO AUGUSTO PEREIRA ROLIM

**Análise da Integração do Design Centrado no Usuário com Métodos  
Ágeis em Projetos de Software**

São Paulo  
2025

CLAUDIO AUGUSTO PEREIRA ROLIM

**Análise da Integração do Design Centrado no Usuário com Métodos  
Ágeis em Projetos de Software**

**Versão Original**

Monografia apresentada ao PECE – Programa de Educação Continuada em Engenharia da Escola Politécnica da Universidade de São Paulo como parte dos requisitos para a conclusão do curso de MBA em Engenharia de Software.

Área de Concentração: Engenharia de Software

Orientador(a): Prof.<sup>a</sup> Dr.<sup>a</sup> Tânia Letícia dos Santos

São Paulo  
2025

Autorizo a reprodução e divulgação total ou parcial deste trabalho, por qualquer meio convencional ou eletrônico, para fins de estudo e pesquisa, desde que citada a fonte.

#### FICHA CATALOGRÁFICA

[Colocar na versão final do trabalho. Obter em:  
<https://www.poli.usp.br/bibliotecas/servicos/catalogacao-na-publicacao>]

Nome: ROLIM, Claudio Augusto Pereira

Título: Análise da Integração do Design Centrado no Usuário com Métodos Ágeis em Projetos de Software

Monografia apresentada ao PECE – Programa de Educação Continuada em Engenharia da Escola Politécnica da Universidade de São Paulo como parte dos requisitos para a conclusão do curso de MBA em Engenharia de Software.

Aprovado em: / /

Banca Examinadora

Prof(a). Dr(a). \_\_\_\_\_

Instituição: \_\_\_\_\_

Julgamento: \_\_\_\_\_

Prof(a). Dr(a). \_\_\_\_\_

Instituição: \_\_\_\_\_

Julgamento: \_\_\_\_\_

Prof(a). Dr(a). \_\_\_\_\_

Instituição: \_\_\_\_\_

Julgamento: \_\_\_\_\_

## DEDICATÓRIA

*Dedico este trabalho ao meu querido  
pai, em memória, cujo exemplo,  
ensinamentos e dedicação à minha  
educação foram a base de todas as  
minhas conquistas; e à minha esposa  
Raíza, pela maravilhosa pessoa que  
ela é.*

## **AGRADECIMENTOS**

À Universidade de São Paulo – USP – onde tive a honra de me graduar e a oportunidade de realizar esta pós-graduação em Engenharia de Software, experiência pela qual sou profundamente grato.

À Escola Politécnica da Universidade de São Paulo e ao PECE pela estrutura disponibilizada e pelo suporte oferecido, que tornaram possível a realização deste curso presencial ao longo destes dois anos.

A todos os professores, cuja generosidade ao compartilhar conhecimentos foi essencial para o meu aprendizado e crescimento. Em especial, à minha orientadora, cuja orientação e apoio foram fundamentais para a concretização deste trabalho.

Ao meu ex-gestor Diego Ferrante, pelo incentivo para fazer esse curso, incluindo um incentivo financeiro generosamente oferecido, gesto que pretendo retribuir no futuro, auxiliando de forma semelhante outra pessoa a transformar sua trajetória por meio do estudo.

E, por fim, à minha família, sem a qual nada seria possível.

## RESUMO

ROLIM, C. A. P. **Análise da Integração do Design Centrado no Usuário com Métodos Ágeis em Projetos de Software**. 2025. 77 páginas. Monografia (MBA em Engenharia de Software). Programa de Educação Continuada em Engenharia da Escola Politécnica da Universidade de São Paulo. São Paulo. 2025.

Esta monografia analisa a integração do Design Centrado no Usuário (UCD) com o Desenvolvimento Ágil de Software (ASD) em projetos recentes de desenvolvimento de software, com foco em experiências práticas documentadas nos últimos cinco anos, entre 2020 e 2024. A pesquisa é conduzida por meio de uma Revisão Sistemática da Literatura (RSL) que examina estudos empíricos recentes, extraídos de artigos revisados por pares, para identificar pontos fortes e fracos dessa integração. Três objetivos específicos orientam o estudo: primeiramente, uma análise comparativa entre UCD e ASD para destacar convergências e divergências, fornecendo uma base teórica para a integração das duas abordagens; em seguida, a elaboração de um processo sistemático de revisão para buscar e selecionar os estudos empíricos relevantes; e, finalmente, uma análise detalhada dos resultados, identificando técnicas e métodos empregados para integração e os benefícios e problemas observados. Os resultados apontam como a integração entre o UCD e o ASD ocorreu na prática nos estudos analisados, além de vantagens, como maior foco na experiência do usuário, e desafios, tais como a adaptação do processo de design em ciclos ágeis. Adicionalmente, com base nos resultados da RSL, esta monografia propõe diretrizes que visam facilitar a aplicação conjunta de UCD e ASD.

Palavras-chave: Design Centrado no Usuário, Desenvolvimento Ágil de Software, Métodos Ágeis, Revisão Sistemática da Literatura, Integração, Experiência do Usuário, UX

## ABSTRACT

ROLIM, C. A. P. **Analysis of the Integration of User-Centered Design with Agile Methods in Software Projects**. 2025. 77 pages. Monografia (MBA em Engenharia de Software). Programa de Educação Continuada em Engenharia da Escola Politécnica da Universidade de São Paulo. São Paulo. 2025.

This academic work analyses the integration of User-Centered Design (UCD) with Agile Software Development (ASD) in recent software development projects, focusing on practical experiences documented in the last five years, between 2020 and 2024. The research is conducted through a Systematic Literature Review (SLR) that examines recent empirical studies, extracted from peer-reviewed articles, to identify strengths and weaknesses of this integration. Three specific objectives guide the study: first, a comparative analysis between UCD and ASD to highlight convergences and divergences, providing a theoretical basis for the integration of the two approaches; then, the elaboration of a systematic review process to search for and select relevant empirical studies; and, finally, a detailed analysis of the results, identifying techniques and methods employed for integration and the benefits and problems observed. The results indicate how the integration between UCD and ASD occurred in practice in the studies analysed, in addition to advantages, such as greater focus on user experience, and challenges, such as adapting the design process in agile cycles. Additionally, based on the results of the RSL, this work proposes guidelines that aim to facilitate the joint application of UCD and ASD.

**Keywords:** User-Centered Design, Agile Software Development, Agile Methods, Systematic Literature Review, Integration, User Experience, UX

## **LISTA DE QUADROS**

Pág.

Quadro 1 – Principais Métodos e técnicas do Design Centrado no Usuário.....	23
Quadro 2 – Princípios do "Manifesto para o Desenvolvimento Ágil de Software" .....	27
Quadro 3 – Convergências entre o UCD e o ASD.....	39
Quadro 4 – Divergências entre o UCD e o ASD.....	40
Quadro 5 – Critérios de PICOC.....	44
Quadro 6 – Strings de Busca.....	44
Quadro 7 – Critérios de Inclusão.....	47
Quadro 8 – Critérios de Exclusão.....	47
Quadro 9 – Resultado do Processo de Seleção.....	48
Quadro 10 – Como a integração ocorreu na prática.....	51
Quadro 11 – Diretrizes para a Integração entre o UCD e o ASD.....	58

## **LISTA DE ILUSTRAÇÕES**

Pág.

Figura 1 – Distribuição dos Artigos por Ano de Publicação .....	46
Figura 2 – Resultado do Processo de Seleção .....	48

## **LISTA DE ABREVIATURAS E SIGLAS**

ASD	Desenvolvimento Ágil de Software (do inglês: “ <i>Agile Software Development</i> ”)
PICOC	População, Intervenção, Comparaçāo, Resultados e Contexto (do inglês: “ <i>Population, Intervention, Comparison, Outcome and Context</i> ”)
RSL	Revisão Sistemática da Literatura
UCD	Design Centrado no Usuário (do inglês: “ <i>User-Centered Design</i> ”)
UX	Experiência do Usuário (do inglês: “ <i>User Experience</i> ”)
XP	Programação Extrema (do inglês: “ <i>Extreme Programming</i> ”)

## SUMÁRIO

Pág.

<b>1. INTRODUÇÃO.....</b>	<b>14</b>
1.1 Contexto.....	14
1.2 Objetivo.....	15
1.3 Justificativas.....	16
1.4 Método de Pesquisa.....	17
1.5 Estrutura do Trabalho.....	19
<b>2. REFERENCIAL TEÓRICO.....</b>	<b>20</b>
2.1 Design Centrado no Usuário.....	20
2.1.1. Conceitos e Princípios.....	20
2.1.2. Processos de Design.....	21
2.1.3. Métodos e Técnicas.....	22
2.1.4. Benefícios e Desafios.....	25
2.2 Desenvolvimento Ágil de Software.....	25
2.2.1 Origem e Princípios.....	26
2.2.2 Principais Métodos Ágeis.....	27
2.3 Considerações do Capítulo.....	30
<b>3. ANÁLISE COMPARATIVA ENTRE O UCD E O ASD.....</b>	<b>31</b>
3.1 Convergências e Similaridades.....	31
3.2 Divergências e Conflitos.....	34
3.3 Síntese dos Resultados.....	38
3.4 Considerações do Capítulo.....	41
<b>4. PROCESSO DE REVISÃO SISTEMÁTICA.....</b>	<b>42</b>
4.1 Questões de Pesquisa.....	42
4.2 Processo de Busca.....	43
4.3 Processo de Seleção.....	46

4.4 Processo de Análise.....	49
4.5 Considerações do Capítulo.....	49
<b>5. ANÁLISE E DISCUSSÃO.....</b>	<b>50</b>
5.1 QP1 – Como a Integração entre o UCD e o ASD ocorreu na prática?.....	50
5.2 QP2 – Pontos Fortes na Integração – Benefícios e Vantagens.....	53
5.3 QP3 – Pontos Fracos na Integração – Desafios e Problemas.....	55
5.4 Diretrizes para a Integração entre o UCD e o ASD.....	57
5.5 Considerações do Capítulo.....	60
<b>6. CONSIDERAÇÕES FINAIS.....</b>	<b>61</b>
6.1 Conclusões.....	61
6.2 Trabalhos Futuros.....	62
<b>REFERÊNCIAS.....</b>	<b>63</b>
<b>Apêndice A – Planilha de apoio para seleção dos 351 artigos inicialmente buscados na RSL.....</b>	<b>68</b>

# 1. INTRODUÇÃO

O desenvolvimento de software tem se tornado cada vez mais complexo e desafiador, demandando abordagens que conciliem a eficiência produtiva e a qualidade na experiência do usuário (SZABÓ; HERCEGFI, 2022). Nesse cenário, o Design Centrado no Usuário (UCD, do inglês User-Centered Design), e o Desenvolvimento Ágil de Software (ASD, do inglês Agile Software Development), têm se destacado como abordagens complementares, cada uma com características e princípios específicos, para responder a diferentes demandas do processo de criação de software (RAMJAUN; ABDULLAH; RANA, 2023).

O UCD coloca o usuário no centro do processo de design, buscando entender suas necessidades, comportamentos e preferências para criar produtos que ofereçam uma experiência satisfatória e intuitiva (NORMAN, 2024). Isso é especialmente relevante em um contexto no qual a usabilidade e a satisfação do usuário final são fatores cruciais para o sucesso de um produto (OMERKHEL; YUSOP; ISMAIL; AZMI; 2023). Por outro lado, o ASD enfatiza a flexibilidade, a resposta rápida a mudanças e o desenvolvimento incremental, proporcionando ciclos de entrega mais curtos e adaptáveis às demandas do mercado (AOUNI; MOUMANE; IDRI; NAJIB; JAN, 2024). As referidas abordagens possuem convergências e divergências entre si, ocasionando vantagens e dificuldades para a sua integração em projetos de desenvolvimento de *software* (SZABÓ; HERCEGFI, 2022).

## 1.1 Contexto

Apesar dos benefícios individuais tanto do Design Centrado no Usuário (UCD), quanto do Desenvolvimento Ágil de Software (ASD), a integração das referidas abordagens apresenta desafios específicos (DURAK; GULTEKIN; HARALAMPOPOULOS; SKOULIDAS; CIRAULO; VIPPARLA; KORCZUKOWSKI, 2024). O ritmo mais acelerado dos métodos ágeis, com seus ciclos curtos e foco em entregas rápidas, muitas vezes entra em conflito com as etapas mais longas de pesquisa e validação do UCD (AL-RAZGAN; ALDOSSARI; ALBESHER; ALSHAMMARI; ALOTAIBI; ALANQUARY; ALSAHRANI; ALSAYKHAN, 2022).

Essas e outras divergências entre as abordagens geram a necessidade de adaptações e ajustes para harmonizar o foco no usuário, com a agilidade e eficiência do desenvolvimento (SZABÓ; HERCEGFI, 2022).

Entretanto, recentemente, diversos estudos empíricos têm investigado essa integração na prática, documentando tanto os benefícios quanto as dificuldades enfrentadas (AL-RAZGAN et al., 2022). Essas pesquisas apontam que, embora a combinação entre UCD e ASD possa melhorar a qualidade dos produtos e aumentar a satisfação do usuário, também surgem desafios relacionados ao alinhamento entre equipes de UX e equipes de desenvolvimento de software, além da necessidade de adaptação das práticas do UCD para ciclos ágeis (DURAK et al., 2024).

Diante do exposto, compreender como as práticas de UCD e ASD podem ser integradas em projetos de desenvolvimento de software é um passo fundamental para orientar a aplicação conjunta dessas abordagens (OMERKHEL et al., 2023). Nesse cenário, a análise de estudos empíricos recentes sobre essa integração permite identificar práticas bem-sucedidas, desafios frequentes e propor diretrizes para facilitar essa combinação, visando o desenvolvimento de produtos de software que sejam tanto funcionais, quanto centrados na experiência do usuário (AL-RAZGAN et al., 2022; DURAK et al., 2024; OMERKHEL et al., 2023).

## 1.2 Objetivo

Com base no contexto apresentado, o objetivo geral desta monografia é: Compreender como o Design Centrado no Usuário (UCD) e o Desenvolvimento Ágil de Software (ASD) podem ser integrados, analisando suas convergências e divergências, e investigando como essa integração pode ocorrer na prática. E, com base na investigação de como a integração ocorre na prática, identificar pontos fortes e fracos dessa integração e propor diretrizes que orientem a aplicação conjunta dessas abordagens em projetos de desenvolvimento de software.

Para alcançar o objetivo geral, foram definidos os seguintes objetivos específicos:

1. Realizar uma análise comparativa entre o UCD e o ASD:
  - Identificar convergências e divergências entre as duas abordagens, de forma a construir uma fundamentação teórica para sua integração.
2. Conduzir uma Revisão Sistemática da Literatura (RSL) sobre a integração entre UCD e ASD:
  - Detalhar o processo sistemático de busca, seleção e análise de estudos empíricos recentes.
3. A partir dos estudos empíricos selecionados, analisar e discutir os resultados obtidos:
  - Identificar como a integração do UCD e ASD em projetos de software ocorre na prática;
  - Investigar os principais pontos fortes, incluindo benefícios e vantagens, da integração;
  - Investigar os principais pontos fracos, tais como desafios e problemas enfrentados; e
  - Desenvolver diretrizes baseadas nos resultados, para orientar a integração das duas abordagens.

Os objetivos desse estudo foram definidos para oferecer uma visão abrangente da integração entre UCD e ASD, fornecendo um suporte teórico e prático para pesquisadores e profissionais da Engenharia de Software que desejam alinhar agilidade e foco no usuário em seus projetos.

### **1.3 Justificativas**

A integração entre o Design Centrado no Usuário (UCD) e o Desenvolvimento Ágil de Software (ASD) possui grande importância no contexto atual de desenvolvimento de software (DURAK et al., 2024; RAMJAUN et al., 2023). Com a crescente demanda por produtos que não apenas atendam às funcionalidades esperadas, mas que também proporcionem uma experiência de usuário satisfatória, torna-se fundamental explorar como essas duas abordagens podem ser combinadas (SZABÓ; HERCEGFI, 2022). Nesse contexto, a satisfação do usuário final é um

diferencial competitivo crucial, influenciando a adoção e o sucesso de produtos no mercado (NORMAN, 2024).

Adicionalmente, apesar de ambos, UCD e ASD, serem amplamente utilizados individualmente, há lacunas na literatura quanto a diretrizes que auxiliem na integração dessas abordagens na prática (MOREIRA et al., 2023). Complementarmente, muitas organizações enfrentam desafios ao tentar harmonizar o foco na usabilidade do UCD com a rapidez e flexibilidade do ASD (SZABÓ; HERCEGFI, 2022). Portanto, este estudo se justifica pela necessidade de buscar perspectivas e orientações que possam ajudar a superar esses obstáculos, melhorando a eficiência e a qualidade dos produtos de software.

Por fim, ao investigar as práticas, benefícios e desafios da combinação entre UCD e ASD, esta monografia contribui para o avanço do conhecimento na área de Engenharia de Software. Ao propor diretrizes para essa integração, essa monografia proporciona contribuições para profissionais que buscam alinhar a abordagem ágil e o foco no usuário em seus projetos, com vistas à criação de produtos mais alinhados às necessidades dos usuários e consequente aumento da competitividade dos produtos no mercado.

## 1.4 Método de Pesquisa

Com base em Gil (2022), é possível afirmar que uma pesquisa científica pode ser classificada em diferentes categorias, a partir dos aspectos:

- i) Natureza dos dados: pode ser qualitativa ou quantitativa;
- ii) Objetivos: pode ser exploratória, descritiva ou explicativa;
- iii) Finalidade: pode ser aplicada ou básica;

Conforme a classificação de Gil (2022), a pesquisa desta monografia adota um método qualitativo, descritivo e básico. É uma pesquisa qualitativa por valorizar descrições detalhadas e análises contextuais em vez de dados numéricos ou medições estatísticas, e descritiva por buscar analisar e expor características do fenômeno estudado (GIL, 2022). Por fim, a pesquisa é classificada como básica,

pois tem como objetivo ampliar o conhecimento científico com base em estudos já existentes (GIL, 2022).

Para atender os objetivos propostos, esta monografia adotou uma abordagem metodológica estruturada em três etapas:

- Etapa 1: inicialmente, foi realizada uma análise comparativa entre os conceitos de Design Centrado no Usuário (UCD) e de Desenvolvimento Ágil de Software (ASD). Essa análise se baseou em uma revisão de literatura (GIL, 2022), com base tanto em trabalhos seminais quanto em trabalhos recentes, envolvendo livros e artigos acadêmicos especializados nas áreas de UCD e ASD.
- Etapa 2: em seguida, conduziu-se uma Revisão Sistemática da Literatura (RSL) sobre a integração entre UCD e ASD, com base em métodos propostos por Kitchenham e Charters (2007). Para manter a atualidade e relevância dos estudos, a referida RSL se baseia em artigos científicos empíricos recentes, publicados nos anos de 2020 a 2024, aceitos nos principais *journals* e disponibilizados em portais acadêmicos que contém estudos relevantes na área de Engenharia de Software (KITCHENHAM; CHARTERS, 2007). Descrição mais detalhada sobre os procedimentos metodológicos da RSL são apresentadas no quarto capítulo desta monografia (sob o título “Processo de Revisão Sistemática”).
- Etapa 3: a terceira etapa envolveu a análise dos resultados da RSL, destacando métodos, técnicas, benefícios e desafios da integração entre UCD e ASD. Essa avaliação buscou a compreensão sobre como a referida integração ocorre na prática e identificar os elementos-chave para sua eficácia (KITCHENHAM; CHARTERS, 2007).

Com base nos resultados obtidos, são sugeridas diretrizes para orientar a aplicação conjunta do UCD e do ASD, em projetos de desenvolvimento de software.

## 1.5 Estrutura do Trabalho

O Capítulo 1 INTRODUÇÃO apresenta o contexto da pesquisa, os seus objetivos, as justificativas para o estudo, o método utilizado e a estrutura do trabalho.

O Capítulo 2 REFERENCIAL TEÓRICO aborda os fundamentos do Design Centrado no Usuário (UCD) e do Desenvolvimento Ágil de Software (ASD).

O Capítulo 3 ANÁLISE COMPARATIVA ENTRE O UCD E O ASD discute as convergências e divergências entre essas abordagens, fornecendo uma síntese dos resultados obtidos.

O Capítulo 4 PROCESSO DE REVISÃO SISTEMÁTICA descreve as etapas do processo de Revisão Sistemática da Literatura (RSL), incluindo a formulação de questões de pesquisa, os critérios de busca, seleção e análise dos estudos.

O Capítulo 5 ANÁLISE E DISCUSSÃO descreve como a integração entre UCD e ASD ocorreu na prática nos estudos selecionados, os pontos fortes e fracos da integração, e sugere diretrizes para a integração entre o UCD e o ASD.

O Capítulo 6 CONSIDERAÇÕES FINAIS, apresenta as conclusões do trabalho e sugestões para trabalhos futuros.

## 2. REFERENCIAL TEÓRICO

Este capítulo discute os fundamentos do Design Centrado no Usuário (UCD, do inglês User-Centered Design) e do Desenvolvimento Ágil de Software (ASD, do inglês Agile Software Development).

A compreensão dos conceitos UCD e ASD é um passo fundamental para a análise de suas convergências, divergências e, por fim, para a análise de sua integração em projetos de software. Portanto, essa compreensão é essencial para o alcance dos objetivos desta monografia.

### 2.1 Design Centrado no Usuário

O Design Centrado no Usuário, em inglês User Centered Design (UCD), tem como objetivo fundamental desenvolver produtos e serviços que atendam eficazmente às necessidades, expectativas e limitações dos usuários finais, colocando-os no centro de todo o processo de design (SHARP; PREECE; ROGERS, 2019).

Originado nos anos 1980, o UCD foi amplamente influenciado pelo trabalho de Don Norman, que destacou a importância da usabilidade e da experiência do usuário na interação homem-computador (ALYAHYA; ALMUGHRAM, 2020; NORMAN, 2024). Essa abordagem evoluiu junto com o design de interfaces e experiências, enfatizando a criação de soluções intuitivas e eficientes que melhoram a satisfação e o engajamento dos usuários (NORMAN, 2024; SZABÓ; HERCEGFI, 2022).

#### 2.1.1. Conceitos e Princípios

O Design Centrado no Usuário (UCD) é uma abordagem de design que coloca as necessidades, preferências e limitações do usuário no centro do processo de desenvolvimento (SHARP et al., 2019). Ao focar diretamente nos usuários finais, o UCD busca criar produtos e serviços que sejam não apenas funcionais, mas também intuitivos e satisfatórios para quem os utiliza (SZABÓ; HERCEGFI, 2022). O UCD destaca a importância de entender profundamente o público-alvo para

assegurar que o resultado final atenda às suas expectativas reais (SHARP et al., 2019).

Segundo Sharp et al. (2019), o Design Centrado no Usuário (UCD) baseia-se em três princípios fundamentais: i) foco contínuo nas necessidades dos usuários; ii) avaliação empírica da solução de design; e iii) um processo iterativo de desenvolvimento do design. A avaliação empírica, de acordo com os referidos autores, envolve observar e medir as reações e o desempenho dos usuários desde as etapas iniciais do desenvolvimento, garantindo que o design atenda às suas necessidades. Já o caráter iterativo do UCD destaca a importância de coletar e incorporar feedback dos usuários finais de forma contínua, permitindo refinamentos e melhorias com base nas experiências reais dos usuários (SHARP et al., 2019). Diferentemente dos princípios de outras abordagens de desenvolvimento de produtos ou serviços, que podem priorizar requisitos técnicos ou de negócio, o UCD se diferencia por priorizar a experiência do usuário, resultando em soluções mais eficazes e centradas nas pessoas (SZABÓ; HERCEGFI, 2022).

### **2.1.2. Processos de Design**

De acordo com Salinas, Cueva e Paz (2020), o Design Centrado no Usuário (UCD) é composto por quatro atividades principais: i) compreender e especificar o contexto de uso; ii) definir os requisitos dos usuários; iii) desenvolver soluções de design; e iv) avaliar os designs com base nos requisitos estabelecidos. Essas quatro atividades são detalhadas nos parágrafos subsequentes.

A primeira atividade do UCD, a compreensão do contexto de uso, envolve entender o ambiente e as condições nas quais o usuário interage com o produto ou serviço (SHARP et al., 2019). Esse entendimento inicial permite identificar fatores que impactam a usabilidade e a experiência do usuário, tais como as tarefas que ele realiza, o ambiente físico e social e as ferramentas que utiliza (SHARP et al., 2019). Essa atividade é fundamental para alinhar o design às reais necessidades e expectativas dos usuários, assegurando que o produto seja relevante e eficaz em seu contexto de aplicação (SZABÓ; HERCEGFI, 2022).

Na atividade de definir os requisitos dos usuários do UCD identifica-se as necessidades e os objetivos específicos dos usuários que o produto ou serviço precisa atender (SHARP et al., 2019). A definição dos requisitos envolve a coleta de dados a partir de entrevistas, questionários, grupos focais e observações de comportamento (SZABÓ; HERCEGFI, 2022). Essa atividade assegura que o design final incorpore os requisitos essenciais para a experiência do usuário, permitindo a criação de um produto que esteja em sintonia com os desejos e limitações do público-alvo (SHARP et al., 2019).

Enquanto a atividade de desenvolvimento de soluções de design do UCD abrange a criação de protótipos e representações visuais do produto, que permitem que designers e *stakeholders* visualizem o conceito e avaliem a viabilidade das soluções propostas (SZABÓ; HERCEGFI, 2022). Essa atividade é iterativa e promove a exploração criativa, possibilitando que múltiplas opções sejam consideradas antes de se avançar para um conceito final (SHARP et al., 2019).

Por fim, a avaliação do design, no processo de UCD, é uma atividade contínua em que as soluções de design são testadas e refinadas com base no feedback dos usuários (SHARP et al., 2019). Essa atividade busca assegurar que o design atenda efetivamente aos requisitos definidos e ofereça uma boa experiência ao usuário (SHARP et al., 2019). A avaliação pode incluir testes de usabilidade, pesquisa com usuários e avaliação heurística, ajudando a identificar pontos de melhoria para refinar o design até que atinja um nível apropriado de usabilidade e satisfação para o usuário final (SALINAS et al., 2020).

### **2.1.3. Métodos e Técnicas**

Os métodos e técnicas do Design Centrado no Usuário buscam a compreensão e melhoria da interação entre usuários e sistemas, visando otimizar a usabilidade, eficiência, eficácia e satisfação do usuário final (SZABÓ; HERCEGFI, 2022).

Alguns dos principais métodos e técnicas do Design Centrado no Usuário são resumidos no Quadro 1, destacando suas características e objetivos (SALINAS et al., 2020; SZABÓ; HERCEGFI, 2022):

Quadro 1 – Principais Métodos e técnicas do Design Centrado no Usuário.

Métodos e Técnicas	Descrição	Objetivo	Referência
<b>Avaliação Heurística</b>	Avaliação das interfaces de um produto de software com base em regras predefinidas.	Detectar problemas de usabilidade nas interfaces do sistema.	Salinas et al. (2020)
<b>Entrevistas</b>	Coleta de dados qualitativos por meio de perguntas abertas com usuários reais.	Compreender necessidades, percepções e motivação dos usuários.	Salinas et al. (2020); Szabó e Hercegfi (2022)
<b>Grupos Focais</b>	Discussão em grupo moderada para explorar atitudes e percepções sobre o produto.	Obter insights coletivos sobre a experiência e expectativas dos usuários.	Salinas et al. (2020); Szabó e Hercegfi (2022)
<b>Identificação de stakeholders e usuários</b>	Processo de mapeamento e identificação dos stakeholders e usuários finais envolvidos no projeto.	Entender quem são os usuários e partes interessadas para assegurar que o design atenda às suas necessidades.	Salinas et al. (2020)
<b>Jornada do Usuário</b>	Mapeamento visual das etapas e pontos de contato da experiência do usuário com o produto.	Identificar pontos críticos e oportunidades de melhoria no fluxo do usuário.	Szabó e Hercegfi (2022)
<b>Observação</b>	Acompanhamento das interações dos usuários com o produto no seu contexto de uso.	Entender o comportamento e contexto do usuário na prática.	Szabó e Hercegfi (2022)
<b>Questionários</b>	Questionários quantitativos para mensurar atitudes e satisfação dos usuários ou aspectos da usabilidade.	Coletar dados estruturados sobre satisfação, preferências ou aspectos de usabilidade.	Salinas et al. (2020); Szabó e Hercegfi (2022)
<b>Personas</b>	Criação de arquétipos baseados em dados reais de usuários representativos.	Manter o foco nas características e necessidades do público-alvo.	Szabó e Hercegfi (2022)
<b>Prototipagem</b>	Design de uma versão inicial ou “rascunho” do sistema para explorar ideias e apresentá-las aos usuários finais.	Facilitar a comunicação, exploração, refinamento e avaliação das ideias de design antes do desenvolvimento.	Salinas et al. (2020)
<b>Testes de Usabilidade</b>	Observação e avaliação de usuários interagindo com o produto em ambiente controlado.	Avaliar a eficácia, eficiência e satisfação com o design.	Salinas et al. (2020); Szabó e Hercegfi (2022)

Fonte: elaborado pelo autor com base em Salinas et al. (2020); Szabó e Hercegfi (2022).

Adicionalmente, vale ressaltar que esses métodos e técnicas podem ser utilizados em mais de uma das quatro atividades principais do processo de design do UCD anteriormente mencionadas: contexto de uso, requisitos do usuário, soluções de design e avaliação do design (SALINAS et al., 2020).

Especificamente quanto aos métodos e técnicas do UCD utilizados na atividade de avaliação do design, Szabó e Hercegfi (2022) os dividem em duas categorias principais: i) analíticos; e ii) empíricos. Enquanto Salinas et al. (2020) os dividem em três categorias: i) inspeção; ii) teste; e iii) investigação. Os referidos autores destacam suas principais características de cada categoria (SALINAS et al., 2020; SZABÓ; HERCEGFI, 2022):

- Métodos analíticos ou de inspeção: buscam avaliar a experiência do usuário identificando problemas de usabilidade, comparando a interface com padrões estabelecidos ou simulando a atividade esperada do usuário, sem a necessidade de interação direta dos usuários finais; como, por exemplo, a avaliação heurística ou o Cognitive Walkthrough (SALINAS et al., 2020; SZABÓ; HERCEGFI, 2022). Segundo Szabó e Hercegfi (2022), esses métodos podem ser realizados de forma mais econômica, pois não exigem a participação direta dos usuários finais no processo de avaliação.
- Métodos empíricos ou de teste: buscam avaliar a experiência do usuário ao envolver os usuários finais diretamente no processo, permitindo que realizem tarefas no sistema, para identificar problemas de usabilidade, observados em situações reais de uso (SALINAS et al., 2020). De acordo com Szabó e Hercegfi (2022), por um lado esses métodos permitem um estudo mais profundo das interações por envolver diretamente os usuários finais, mas, por outro lado, podem ser um processo mais custoso ou demorado, requerendo mais recursos, planejamento e organização prévia.
- Métodos de investigação (“*inquiry methods*”): buscam avaliar a experiência do usuário por meio da obtenção de informações da interação do usuário com o sistema, utilizando entrevistas, observações e pesquisas, possibilitando uma avaliação sobre as percepções e dificuldades dos usuários (SALINAS et al., 2020).

#### 2.1.4. Benefícios e Desafios

Os principais benefícios do Design Centrado no Usuário (UCD) estão relacionados ao aumento da usabilidade e satisfação do usuário, resultando em produtos que atendem de maneira eficaz e intuitiva às necessidades reais dos usuários (SHARP et al., 2019). Ao envolver os usuários em todas as fases do desenvolvimento, o UCD possibilita a criação de interfaces e funcionalidades que refletem diretamente as expectativas e preferências do público-alvo (NORMAN, 2024). Esse foco no usuário final contribui para a redução de erros e da necessidade de suporte pós-lançamento, além de melhorar a retenção e a aceitação do produto no mercado, uma vez que o usuário percebe o valor e a praticidade no uso (SHARP et al., 2019).

No entanto, a aplicação do UCD apresenta desafios (AL-RAZGAN et al., 2022). O processo pode ser demorado e custoso, especialmente devido à necessidade de pesquisas com usuários, testes contínuos e iterações frequentes no design (SZABÓ; HERCEGFI, 2022). Esses aspectos podem dificultar a adoção do UCD em ambientes com restrições de tempo e orçamento, tal como ocorre em alguns projetos ágeis (AL-RAZGAN et al., 2022). Adicionalmente, em organizações com culturas menos flexíveis, pode haver resistência cultural à adoção de práticas de design que priorizam o usuário (SZABÓ; HERCEGFI, 2022). Outro desafio é a necessidade de equipes com habilidades específicas, tais como pesquisa e design da experiência do usuário, que nem sempre estão disponíveis em todos os contextos (AL-RAZGAN et al., 2022). Assim, esses fatores podem dificultar a aplicação completa do UCD, especialmente em ambientes profissionais com resistência a mudanças ou projetos com restrições de recursos (SALMAN et al., 2023).

## 2.2 Desenvolvimento Ágil de Software

O Desenvolvimento Ágil de Software, em inglês “Agile Software Development” (ASD), com foco em adaptabilidade, colaboração, e entrega incremental, remodelou a indústria de software (AOUNI; MOUMANE; IDRI; NAJIB; JAN, 2024). Isso ocorreu porque o desenvolvimento ágil é diferente de outros métodos tradicionais de

desenvolvimento, que costumam seguir um plano mais rígido e linear, com etapas bem definidas (SOMMERVILLE, 2019). O desenvolvimento ágil é caracterizado pelo desenvolvimento iterativo e responsivo a mudanças, trabalhando com o feedback contínuo dos *stakeholders*, características que são críticas nos ambientes de rápido desenvolvimento da atualidade (PRESSMAN; MAXIM, 2021).

Em contraste com os métodos tradicionais orientados por planos, o desenvolvimento ágil integra as atividades de especificação, design e implementação de software em um único ciclo, comumente chamado de "*sprint*" (SOMMERVILLE, 2019). Adicionalmente, prioriza interações frequentes com o cliente e entregas incrementais a cada poucas semanas (WAJA et al., 2021). Os métodos ágeis, tais como Scrum e o Extreme Programming (XP), minimizam a documentação formal e favorecem a comunicação informal, visando entregar um produto funcional o mais rápido possível (PRESSMAN; MAXIM, 2021). Por meio da implementação de métodos ágeis, uma equipe estará em posição de responder positivamente às mudanças nos requisitos, ao mesmo tempo em que contribui para a geração de valor ao entregar incrementos reais ao produto com frequência (SOMMERVILLE, 2019).

### **2.2.1 Origem e Princípios**

O Desenvolvimento Ágil de Software surgiu em resposta às limitações dos métodos tradicionais de desenvolvimento, que frequentemente eram inflexíveis e não respondiam bem às mudanças nas necessidades dos clientes (PRESSMAN; MAXIM, 2021). Em 2001, Kent Beck e outros dezesseis desenvolvedores, escritores e consultores de software, se reuniram para discutir essas dificuldades e buscar uma abordagem que valorizasse a adaptação e a entrega contínua de valor aos clientes (BECK; BEEDLE; BENNEKUM; COCKBURN; CUNNINGHAM; FOWLER; GRENNING; HIGHSMITH; HUNT; JEFFRIES; KERN; MARICK; MARTIN; MELLOR; SCHWABER; SUTHERLAND; THOMAS, 2001). Esse grupo formalizou suas ideias e deu origem ao "Manifesto para o Desenvolvimento Ágil de Software" (PRESSMAN; MAXIM, 2021).

O Manifesto Ágil refletia a experiência prática dos seus fundadores e estabeleceu os doze princípios ágeis, conforme apresentado no Quadro 2 (BECK et al., 2001):

Quadro 2 – Princípios do "Manifesto para o Desenvolvimento Ágil de Software"

Princípio	Descrição
<b>1. Satisfação do cliente</b>	Entregar software com valor rapidamente e continuamente, visando a satisfação do cliente.
<b>2. Aceitar mudanças de requisitos</b>	Adaptar-se a mudanças, mesmo em fases avançadas como vantagem competitiva.
<b>3. Entregas frequentes</b>	Lançar versões funcionais de software regularmente, em intervalos de semanas ou meses.
<b>4. Colaboração entre negócios e desenvolvimento</b>	Manter a comunicação próxima entre o time de desenvolvimento e o cliente.
<b>5. Apoio e confiança à equipe</b>	Fornecer um bom ambiente para que o time alcance os objetivos.
<b>6. Comunicação face a face</b>	Preferir comunicação presencial e direta, que é mais eficaz para transmitir informações.
<b>7. Software funcional como medida de progresso</b>	Avaliar o progresso com base em funcionalidades entregues, em vez de documentações.
<b>8. Ritmo sustentável</b>	Manter um ritmo constante e sustentável para o desenvolvimento, evitando sobrecarga.
<b>9. Excelência técnica e design</b>	Focar na qualidade técnica e no bom design para aumentar a agilidade do produto.
<b>10. Simplicidade</b>	Maximizar o trabalho que não precisa ser feito, mantendo soluções simples e eficazes.
<b>11 Auto-organização das equipes</b>	As melhores arquiteturas, requisitos e designs surgem de equipes auto-organizadas.
<b>12. Reflexão constante</b>	Reavaliar o trabalho periodicamente para melhorar processos e torná-los mais eficientes

Fonte: adaptado de Beck et al. (2001).

A partir do Manifesto Ágil foram definidos quatro valores fundamentais para o Desenvolvimento Ágil de Software (BECK et al., 2001, n.p.):

1. Indivíduos e interações mais que processos e ferramentas.
2. Software em funcionamento mais que documentação abrangente;
3. Colaboração com o cliente mais que negociação de contratos; e
4. Responder a mudanças mais que seguir um plano.

## 2.2.2 Principais Métodos Ágeis

Duas décadas após o surgimento do Manifesto Ágil, há diversas propostas de métodos para o Desenvolvimento Ágil de Software (PRESSMAN; MAXIM, 2021).

Dentre os principais, pode-se destacar:

- Scrum, Kanban e Extreme Programming, métodos ágeis mais adotados (AOUNI et al., 2024);
- SAFe para contextos de grandes escalas, ou seja, para projetos muito grandes e com muitas equipes, especialmente em grandes organizações (ROHATGI, 2023); e
- Modelos híbridos, que combinam os métodos ágeis com os métodos tradicionais de desenvolvimento de software (BIANCHI; CONFORTO; REBENTISCH; AMARAL; REZENDE; PÁDUA, 2021).

O Scrum, reconhecido como um dos métodos ágeis mais utilizados, é definido por ciclos curtos de desenvolvimento denominados *sprints*, reuniões diárias (*dailies*) e revisões das *sprints* (AOUNI et al., 2024). A forma como ele é estruturado é adequada no contexto de desenvolvimento de software, pois permite que as equipes dividam o trabalho em tarefas menores e mais manejáveis, permitindo que equipes diversas colaborem e trabalhem em tarefas muito mais rápido (AOUNI et al., 2024). Putta, Uludağ, Hong, Paasivaara e Lassenius (2021), destacaram a vantagem do Scrum para projetos pequenos e médios, entretanto, apontam eventuais problemas em situações específicas de escalabilidade, nas quais os projetos são muitos grandes ou complexos.

Outro método ágil com grande adoção é o Kanban, focado em visualizar o fluxo de trabalho para melhorar a eficiência e gerenciar tarefas (ALAIDAROS; OMAR; ROMLI, 2021). Introduzido na Toyota na década de 1940, e inspirado no sistema Just In Time, o Kanban permite que as equipes priorizem e ajustem a carga de trabalho com flexibilidade, possibilitando entregas contínuas e atendendo melhor as demandas (ALAIDAROS; OMAR; ROMLI, 2021). O Kanban inclui um quadro visual no qual as tarefas são movidas entre colunas que representam etapas do processo, tais como “a fazer”, “em andamento” e “concluída” (ALAIDAROS; OMAR; ROMLI, 2021). Dessa forma, o referido método facilita o acompanhamento do progresso e a identificação de gargalos (ALAIDAROS; OMAR; ROMLI, 2021). O Kanban também pode ser combinado com outros métodos ágeis, tais como o Scrum, para oferecer mais controle e agilidade no desenvolvimento de projetos (ALAIDAROS; OMAR; ROMLI, 2021).

O Extreme Programming (XP) é um método ágil focado na entrega contínua e na adaptação às mudanças, especialmente útil para equipes médias ou pequenas, ou projetos de escopo reduzido (AKHTAR; BAKHTAWAR; AKHTAR, 2022). Criado por Kent Beck em 1996, o XP incorpora práticas como desenvolvimento orientado a testes, programação em par e feedback frequente do cliente (AKHTAR; BAKHTAWAR; AKHTAR, 2022). O referido método valoriza a simplicidade, a comunicação ativa e a satisfação do cliente (SOMMERVILLE, 2019). E, assim como o Scrum, o XP se organiza em fases iterativas, facilitando ajustes durante o desenvolvimento e minimizando erros (AKHTAR; BAKHTAWAR; AKHTAR, 2022).

Entretanto, um dos desafios frequentemente associados aos métodos ágeis é a escalabilidade, especialmente em grandes organizações que precisam coordenar diversas equipes, conforme apontado por Putta et al. (2021). Segundo Rohatgi (2023), para enfrentar esse desafio, existem abordagens ágeis especificamente desenvolvidas para lidar com a escalabilidade, tais como o Scaled Agile Framework (SAFe). O SAFe foi criado para expandir as melhores práticas ágeis para grandes projetos, introduzindo técnicas que alinham múltiplas equipes aos objetivos organizacionais (ROHATGI, 2023). Dessa forma, o SAFe se apresenta como uma solução adequada para organizações que desejam implementar métodos ágeis, mas que também necessitam de um método eficaz para gerenciar operações complexas e de grande escala, caracterizadas por múltiplas dependências entre projetos e a necessidade de integração entre as equipes (ROHATGI, 2023).

Adicionalmente, Bianchi et al. (2021) destacam os modelos híbridos de gerenciamento do desenvolvimento de software, que combinam métodos ágeis com métodos tradicionais de desenvolvimento de software. Segundo os referidos autores, os modelos híbridos são particularmente demandados em setores específicos, tais como ambientes tradicionais ou altamente regulamentados, nos quais as organizações necessitam de padrões mais previsíveis. Nesse contexto, as abordagens híbridas permitem combinar a flexibilidade dos métodos ágeis com o planejamento rigoroso e a previsibilidade do modelo tradicional, também conhecido como “cascata” (BIANCHI et al., 2021). O estudo dos referidos autores revelou que

as organizações que implementaram modelos híbridos conseguiram gerenciar riscos de forma eficaz e ganharam maior flexibilidade, ao mesmo tempo em que alcançaram conformidade com os requisitos específicos de seus setores, especialmente aquelas que atuam em contextos fortemente regulamentados.

### **2.3 Considerações do Capítulo**

Esse capítulo destacou o foco do Design Centrado no Usuário (UCD) em atender às necessidades reais dos usuários, proporcionando maior usabilidade e satisfação, embora enfrente desafios tais como custos e necessidade de equipes especializadas (SHARP et al., 2019; SZABÓ; HERCEGFI, 2022).

O desenvolvimento ágil de Software (ASD), englobando métodos como Scrum, Kanban e Extreme Programming, por sua vez, valoriza a flexibilidade, o feedback contínuo e as entregas rápidas (SOMMERVILLE, 2019). O ASD pode ser adaptado a diferentes contextos, incluindo o uso do método ágil SAFe para grandes projetos ou o uso dos modelos híbridos em setores altamente regulamentados (BIANCHI et al., 2021; ROHATGI, 2023).

O próximo capítulo aborda as convergências e divergências entre UCD e ASD, com o propósito de construir uma fundamentação teórica para sua integração, um dos objetivos desta monografia.

### 3. ANÁLISE COMPARATIVA ENTRE O UCD E O ASD

Este capítulo realiza uma análise comparativa entre o Design Centrado no Usuário (UCD) e o Desenvolvimento Ágil de Software (ASD), com a finalidade de identificar as convergências, similaridades, divergências e conflitos entre essas duas abordagens.

O objetivo desta análise é construir uma fundamentação para a compreensão de como UCD e ASD podem ser integrados de forma eficaz, em projetos de desenvolvimento de software.

#### 3.1 Convergências e Similaridades

Embora o Design Centrado no Usuário (UCD) e o Desenvolvimento Ágil de Software (ASD) tenham origens e focos distintos, ambas as abordagens compartilham princípios e valores que as tornam complementares em diversos aspectos (SZABÓ; HERCEGFI, 2022). Identificar essas convergências é fundamental para entender como as duas metodologias podem se complementar e contribuir para o desenvolvimento de produtos de software que sejam ao mesmo tempo funcionais, usáveis e alinhados às necessidades dos usuários (YERRAM et al., 2019).

Um primeiro aspecto de convergência entre o UCD e o ASD é a adoção de processos iterativos como princípio fundamental para alcançar seus objetivos (PRESSMAN; MAXIM, 2021; SHARP et al., 2019; SZABÓ; HERCEGFI, 2022). De acordo com Sharp et al. (2019), o processo iterativo é um dos três princípios fundamentais do UCD, enquanto o "Manifesto para Desenvolvimento Ágil de Software" também valoriza a iteração, ao priorizar a entrega contínua de software funcional, como um dos doze princípios fundamentais do ASD (PRESSMAN; MAXIM, 2021). Szabó e Hercegfi (2022) reforçam essa convergência, destacando que ambas as metodologias utilizam ciclos iterativos para incorporar mudanças e realizar melhorias ao longo do desenvolvimento. No UCD, esses ciclos incluem etapas de prototipagem, testes e avaliações com usuários, permitindo um refinamento constante do design (NORMAN, 2024). No ASD, os ciclos iterativos,

comumente chamados de *sprints*, focam no desenvolvimento e entrega de incrementos funcionais do software, possibilitando uma rápida adaptação a mudanças nos requisitos ou no mercado, além de garantir melhorias contínuas (PRESSMAN; MAXIM, 2021). Assim, conforme os referidos autores, ambas as abordagens convergem ao adotar o processo iterativo como um de seus pilares centrais.

Outro aspecto de convergência entre o UCD e o ASD é a ênfase no feedback contínuo (BAI et al., 2018; SHARP et al., 2019; STURM; TSCHOLL, 2019; WAJA et al., 2021). No UCD, conforme destacado por Sturm e Tscholl (2019), o feedback é obtido principalmente dos usuários finais. A usabilidade do software é avaliada qualitativamente por meio desse feedback constante, permitindo que o design seja ajustado para atender efetivamente às necessidades do público-alvo (STURM; TSCHOLL, 2019). No caso do ASD, Waja et al. (2021) apontam que o feedback contínuo desempenha um papel central no processo ágil. Esse feedback é coletado do cliente ao final de cada ciclo (*sprint*) e orienta as próximas etapas do desenvolvimento, permitindo adaptações rápidas a mudanças de requisitos, e melhorando a qualidade do produto, para também atender às necessidades do público-alvo (WAJA et al., 2021). Sharp et al. (2019) reforçam que o UCD prioriza a coleta contínua de feedback para assegurar que o design final atenda às expectativas dos usuários. Da mesma forma, Bai et al. (2018) destacam que o ASD utiliza o feedback contínuo para ajustar processos e produtos de forma iterativa, assegurando que as entregas estejam alinhadas com as necessidades em constante evolução de usuários e clientes. Assim, ambas as abordagens, como observado pelos autores mencionados, reconhecem o feedback contínuo como um componente essencial: no UCD, para o sucesso do design, e no ASD, para o sucesso do desenvolvimento de software.

Outra convergência entre o UCD e o ASD é a flexibilidade e a aceitação de mudanças (PRESSMAN; MAXIM, 2021; SHARP et al., 2019). Conforme observado nos estudos de Bianchi et al. (2022) e Sasmito e Fikri Hidayattullah (2021), a flexibilidade é um aspecto comum em ambas as abordagens, permitindo ajustar os processos para responder a mudanças. No UCD, isso se refere a alterações no

design da solução, enquanto no ASD está relacionado a mudanças na funcionalidade do software (BIANCHI et al., 2022; SASMITO; FIKRI HIDAYATTULLAH, 2021). Ambas as abordagens reconhecem que os requisitos e necessidades dos usuários podem evoluir e, portanto, incorporam a flexibilidade diante de mudanças (PRESSMAN; MAXIM, 2021; SHARP et al., 2019). No UCD, o feedback contínuo e as novas descobertas sobre os usuários permitem mudanças e refinamentos no design, assegurando que o produto permaneça alinhado às expectativas dos usuários finais (SHARP et al., 2019). De maneira semelhante, o ASD utiliza ciclos iterativos e o feedback constante para adaptar-se a mudanças nos requisitos, mesmo em estágios avançados do projeto, possibilitando a reavaliação frequente das prioridades (PRESSMAN; MAXIM, 2021). Assim, tanto o UCD quanto o ASD aceitam as mudanças como oportunidades para aprimorar o produto, ao invés de tratá-las como obstáculos (PRESSMAN; MAXIM, 2021; SHARP et al., 2019).

Outro ponto de convergência entre o UCD e o ASD é o compromisso com a satisfação dos *stakeholders*. Ambas as abordagens enfatizam a importância de compreender e atender às necessidades e expectativas de seus *stakeholders*, tais como clientes ou usuários finais (PRESSMAN; MAXIM, 2021; SHARP et al., 2019). No UCD, essa satisfação é alcançada por meio de uma imersão profunda nas experiências dos usuários, assegurando que o produto final seja intuitivo, eficiente e alinhado às suas demandas reais (SHARP et al., 2019). No ASD, a satisfação dos *stakeholders* é obtida por meio de entregas frequentes e incrementais de software funcional, permitindo feedback contínuo e ajustes rápidos às mudanças nos requisitos (SOMMERVILLE, 2019). Embora o ASD priorize o valor comercial e a velocidade de desenvolvimento, ele também considera as necessidades dos usuários finais, alinhando-se ao objetivo do UCD de favorecer a experiência do usuário (OMERKHEL et al., 2023). Assim, tanto o UCD quanto o ASD promovem uma colaboração próxima com os *stakeholders*, visando entregar produtos que não apenas atendam, mas superem suas expectativas (PRESSMAN; MAXIM, 2021; SHARP et al., 2019).

O UCD e o ASD também convergem na ênfase que ambos colocam na colaboração e comunicação eficaz entre as equipes de desenvolvimento e os *stakeholders* (SHARP et al., 2019; SOMMERVILLE, 2019; SZABÓ; HERCEGFI, 2022). No UCD, a colaboração ativa entre desenvolvedores, designers e usuários finais é fundamental para compreender as necessidades dos usuários e incorporar feedback ao longo do processo de design (SHARP et al., 2019). De maneira semelhante, o ASD promove uma comunicação contínua entre desenvolvedores, clientes e outros *stakeholders*, coletando feedback, facilitando a adaptação rápida a mudanças e assegurando que o produto atenda às expectativas (PRESSMAN; MAXIM, 2021). Ambas as abordagens valorizam equipes multidisciplinares que trabalham em conjunto de forma transparente, compartilhando informações e decisões para alcançar resultados de maior qualidade e relevância para o usuário final (PRESSMAN; MAXIM, 2021; SHARP et al., 2019).

Adicionalmente, outra similaridade entre as abordagens, é que tanto o UCD quanto o ASD, muitas vezes erroneamente compreendidos como metodologias rígidas ou sequências pré-definidas de etapas, que garantem automaticamente o sucesso de um projeto, são, na verdade, filosofias orientadoras que guiam o desenvolvimento, e não metodologias específicas (DWIVEDI; UPADHYAY; TRIPATHI, 2012; GHEORGHE; GHEORGHE; IATAN, 2020; NORMAN, 2024; ZAITSEV; GAL; TAN, 2018). Essas filosofias podem ser aplicadas por meio de diferentes técnicas e métodos (DWIVEDI et al., 2012; ZAITSEV et al., 2018). No caso do ASD, destacam-se métodos como o Scrum e o Extreme Programming (GHEORGHE et al., 2020). Já no UCD, incluem-se técnicas tais como mapas de empatia, criação de personas, prototipagem e testes de usabilidade (SHARP et al., 2019). Dessa forma, a escolha e a adaptação dos métodos e técnicas mais adequados se tornam responsabilidade das equipes e gestores, que devem ajustar cada abordagem às necessidades e ao contexto específico de cada projeto (SHARP et al., 2019; ZAITSEV et al., 2018).

### **3.2 Divergências e Conflitos**

Embora o UCD e o ASD compartilhem princípios e valores que facilitam sua integração, existem divergências importantes que podem gerar conflitos durante a

implementação conjunta dessas abordagens (SZABÓ; HERCEGFI, 2022). Essas diferenças podem gerar conflitos que, se não forem gerenciados adequadamente, podem comprometer a eficácia do processo de desenvolvimento e a qualidade do produto final (YERRAM et al., 2019). Assim, compreender essas diferenças é essencial para desenvolver estratégias eficazes que mitiguem os desafios e potencializem os benefícios da integração (SZABÓ; HERCEGFI, 2022).

Um primeiro aspecto de divergência entre o UCD e o ASD reside na priorização técnica (YERRAM et al., 2019). Com base em Sharp et al. (2019), no UCD há uma forte prioridade técnica na usabilidade e na experiência do usuário, focando em atender às necessidades, expectativas e desejos dos usuários finais. O objetivo é desenvolver produtos intuitivos, eficientes e satisfatórios, mesmo que isso demande mais tempo e recursos (SHARP et al., 2019). Em contrapartida, Gheorghe et al. (2020) afirmam que o ASD prioriza a entrega rápida de funcionalidades que agreguem valor de negócio imediato. Waja et al. (2021) complementam que, no ASD, a prioridade está no desenvolvimento e implementação de funcionalidades técnicas definidas no *backlog*, atendendo aos objetivos estratégicos do negócio e respondendo rapidamente às mudanças nos requisitos. Essa diferença de priorização pode levar a conflitos, já que as necessidades dos usuários identificadas pelo UCD podem não se alinhar com as prioridades de negócio estabelecidas no ASD, resultando na desconsideração de aspectos essenciais da experiência do usuário em favor de entregas rápidas ou demandas imediatas, comprometendo a satisfação do usuário e a usabilidade do produto (YERRAM et al., 2019).

A diferença no ritmo de trabalho e tempo de desenvolvimento é outra divergência importante entre o UCD e o ASD (AL-RAZGAN et al., 2022). O UCD tende a demandar mais tempo, pois envolve etapas detalhadas, tais como pesquisas aprofundadas, testes de usabilidade e iterações baseadas no feedback dos usuários (SHARP et al., 2019). Embora essas atividades sejam fundamentais para garantir que o produto atenda às necessidades dos usuários, elas podem prolongar o processo de desenvolvimento (AL-RAZGAN et al., 2022). Em contrapartida, Waja et al. (2021) afirmam que o ASD opera em ciclos curtos chamados *sprints*, geralmente de duas a quatro semanas, com entregas frequentes de incrementos funcionais do

software. A metodologia ágil prioriza a velocidade e a eficiência, buscando responder rapidamente às mudanças e entregar valor ao cliente o mais cedo possível (MATHARU et al., 2015). A discrepância entre os ritmos de trabalho pode gerar conflitos nos cronogramas, em que as atividades detalhadas do UCD são vistas como obstáculos ao progresso rápido promovido pelo ASD (AL-RAZGAN et al., 2022). Além disso, a pressão para cumprir prazos apertados pode limitar a capacidade das equipes de dedicar tempo suficiente às práticas de UCD, resultando em produtos menos refinados em termos de usabilidade e experiência do usuário (AL-RAZGAN et al., 2022).

Outra divergência está no nível de envolvimento dos usuários finais ao longo do processo de desenvolvimento (SIGNORETTI et al., 2020). No UCD, conforme destacado por Sharp et al. (2019), há uma forte prioridade na participação ativa e contínua dos usuários em todas as etapas do desenvolvimento. Essa participação inclui a coleta de feedback, realização de testes de usabilidade e validação de hipóteses de design, essenciais para garantir que o produto atenda às necessidades e preferências reais dos usuários (SZABÓ; HERCEGFI, 2022). Em contraste, Signoretti et al. (2020) apontam que, no ASD, o envolvimento dos usuários finais tende a ser menos frequente ou indireto. Segundo Omerkhel et al. (2023), o foco no ASD geralmente está no cliente ou em um representante do cliente, o Product Owner, que prioriza o *backlog* e fornece feedback às equipes de desenvolvimento. Contudo, essa abordagem do ASD pode limitar a compreensão profunda das necessidades dos usuários finais, já que o Product Owner nem sempre representa a diversidade de perspectivas dos usuários (OMERKHEL et al., 2023). Essa divergência no nível de envolvimento pode resultar em soluções que não atendem adequadamente às expectativas dos usuários, comprometendo tanto a usabilidade quanto a satisfação do usuário com o produto (OMERKHEL et al., 2023; SIGNORETTI et al., 2020).

A definição de papéis e responsabilidades na equipe é outro aspecto de divergência entre o UCD e o ASD (AL-RAZGAN et al., 2022; SZABÓ; HERCEGFI, 2022). No UCD, os papéis são claramente definidos, tais como designers de UX, pesquisadores de usabilidade e usuários finais, cada um com responsabilidades

específicas no processo de design (SHARP et al., 2019). Essa estrutura clara no UCD facilita a atribuição de tarefas e a especialização, garantindo que as atividades relacionadas à usabilidade e à experiência do usuário sejam tratadas adequadamente (SHARP et al., 2019). Por outro lado, o ASD adota equipes auto-organizadas, nas quais os membros podem assumir múltiplas funções e os papéis não são rigidamente estabelecidos (SOMMERVILLE, 2019; SZABÓ; HERCEGFI, 2022). Essa flexibilidade, embora favoreça a adaptação, pode resultar em falta de clareza sobre as responsabilidades relacionadas ao UX, levando a lacunas ou sobreposições de atividades (AL-RAZGAN et al., 2022). A ausência de papéis claramente definidos para o Design Centrado no Usuário, no contexto ágil, pode dificultar a integração efetiva das práticas de UCD, fazendo com que essas atividades sejam subvalorizadas ou negligenciadas (AL-RAZGAN et al., 2022; SZABÓ; HERCEGFI, 2022).

Outra divergência entre o UCD e ASD diz respeito à documentação gerada durante o processo de desenvolvimento (AL-RAZGAN et al., 2022; SHARP et al., 2019). O UCD demanda documentação detalhada para capturar *insights* de pesquisa, desenvolver personas, mapear jornadas do usuário, criar protótipos e registrar os resultados dos testes de usabilidade (SHARP et al., 2019). Essa documentação é fundamental para comunicar as descobertas, justificar decisões de design e garantir a consistência ao longo do desenvolvimento (SHARP et al., 2019). Por outro lado, o ASD valoriza o software funcional acima da documentação extensa, seguindo o princípio ágil de valorizar interações e indivíduos mais que ferramentas e processos (BECK et al., 2001). No ASD a documentação é mantida no mínimo necessário para não atrasar o desenvolvimento, e manter a equipe focada na entrega de valor (PRESSMAN; MAXIM, 2021). Essa diferença pode gerar conflitos, pois a documentação detalhada do UCD pode ser considerada burocrática no contexto ágil, enquanto a abordagem do ASD, com documentação reduzida, pode afetar a qualidade do design e dificultar a comunicação entre as equipes (AL-RAZGAN et al., 2022; SHARP et al., 2019).

Adicionalmente, há divergências nas técnicas específicas utilizadas por cada abordagem (SHARP et al., 2019; SOMMERVILLE, 2019; SZABÓ; HERCEGFI;

2022). O UCD emprega uma variedade de métodos centrados no usuário, tais como testes de usabilidade, entrevistas em profundidade e prototipação de alta-fidelidade, que são fundamentais para compreender as necessidades dos usuários e validar as soluções de design (SHARP et al., 2019). Essas técnicas demandam tempo, recursos e conhecimentos especializados (SHARP et al., 2019; SZABÓ; HERCEGFI; 2022). Em contraste, o ASD não incorpora formalmente essas técnicas em sua estrutura, focando na entrega de funcionalidades conforme priorizado no *backlog*, e utilizando métodos como reuniões diárias, planejamento e retrospectivas de *sprints* para gerenciar o processo de desenvolvimento (WAJA et al., 2021). A ausência de práticas específicas de UCD no ASD pode resultar em uma falta de compreensão profunda das necessidades do usuário, limitando a eficácia do produto em atender às expectativas dos usuários finais (SZABÓ; HERCEGFI; 2022). Adicionalmente, essa divergência nas técnicas utilizadas pode dificultar a integração das práticas de UCD no processo ágil, demandando adaptações de processos e capacitação das equipes para incorporar efetivamente essas práticas (SZABÓ; HERCEGFI; 2022).

### 3.3 Síntese dos Resultados

A análise comparativa entre o Design Centrado no Usuário (UCD) e o Desenvolvimento Ágil de Software (ASD) deste capítulo revelou uma série de convergências e divergências relevantes. Compreender esses aspectos é uma base importante para identificar oportunidades e desafios na integração dessas duas abordagens, em projetos de desenvolvimento de software (SZABÓ; HERCEGFI, 2022). Após a análise detalhada das convergências e divergências entre UCD e o ASD, este tópico apresenta uma síntese dos resultados obtidos.

A síntese apresentada nesse tópico é importante para a visualização de forma consolidada dos pontos de alinhamento e de antagonismo existentes entre as duas abordagens. A seguir, o Quadro 3 sintetiza as principais convergências analisadas e o Quadro 4 sintetiza as principais divergências entre as duas abordagens analisadas.

Quadro 3 – Convergências entre o UCD e o ASD.

Aspecto	UCD	ASD	Convergência	Referências
<b>Processos Iterativos</b>	Utiliza processos iterativos que incluem prototipagem, testes e avaliações com usuários para refinamento constante do design.	Utiliza ciclos iterativos chamados <i>sprints</i> , focando no desenvolvimento e entrega de incrementos funcionais do software.	Ambas adotam o processo iterativo como princípio fundamental para incorporar mudanças e melhorias ao longo do desenvolvimento.	Norman (2024); Pressman e Maxim (2021); Sharp et al. (2019); Szabo e Hercegfi (2022);
<b>Ênfase no Feedback Contínuo</b>	O feedback é obtido principalmente dos usuários finais para avaliar a usabilidade e ajustar o design conforme necessário.	Coleta feedback ao final de cada <i>sprint</i> para orientar as próximas etapas do desenvolvimento.	Reconhecem o feedback contínuo como essencial para ajustar processos e produtos de forma iterativa, alinhando-se às necessidades dos usuários e clientes.	Bai et al. (2018); Sharp et al. (2019); Sturm e Tscholl (2019); Waja et al. (2021).
<b>Flexibilidade Diante de Mudanças</b>	Permite ajustes no design com base em feedback contínuo e novas descobertas sobre os usuários.	Adapta-se a mudanças nos requisitos, mesmo em estágios avançados do projeto.	Ambas incorporam flexibilidade diante de mudanças, as vendo como oportunidades para aprimorar o produto.	Bianch et al. (2022); Pressman e Maxim (2021); Sasmito e Fikri Hidayattullah (2021); Sharp et al. (2019).
<b>Foco na Satisfação do Stakeholder</b>	Busca compreender e atender às necessidades dos usuários finais, garantindo um produto intuitivo e eficiente.	Alcança satisfação por meio de entregas frequentes e incrementais, permitindo feedback contínuo e ajustes rápidos.	Promovem colaboração próxima com <i>stakeholders</i> para entregar produtos que atendam ou superem suas expectativas.	Omerkhan et al., (2023); Pressman e Maxim (2021); Sharp et al. (2019); Sommerville (2019).
<b>Colaboração e Comunicação Eficaz</b>	Colaboração ativa entre desenvolvedores, designers e usuários finais é fundamental para o processo de design.	Promove comunicação contínua entre desenvolvedores, clientes e outros <i>stakeholders</i> para adaptação rápida a mudanças.	Valorizam equipes multidisciplinares que trabalham juntas de forma transparente, compartilhando informações e decisões.	Pressman e Maxim (2021); Sharp et al. (2019); Sommerville (2019); Szabo e Hercegfi (2022).
<b>Abordagem Filosófica</b>	É uma abordagem que orienta o desenvolvimento do design, e não um processo rígido. Aplicável por meio de diferentes técnicas e métodos.	Também é uma abordagem que orienta o desenvolvimento e não um processo específico. Aplicável a partir de métodos como o Scrum e o XP.	Ambas são abordagens filosóficas e não métodos rígidos. Compartilhando processos iterativos que permitem mudanças com base em feedback.	Dwivedi et al. (2012); Gheorghe et al. (2021); Norman (2024); Sharp et al. (2019); Szabo e Hercegfi (2022); Zaitsev; Gal; Tan (2018).

Fonte: elaborado pelo autor (2025).

Quadro 4 – Divergências entre o UCD e o ASD

Aspecto	UCD	ASD	Divergência	Referências
<b>Priorização técnica</b>	Prioriza usabilidade e experiência do usuário, focando nas necessidades e desejos dos usuários finais.	Prioriza a entrega rápida de funcionalidades que agregam valor de negócio imediato.	Necessidades dos usuários podem não se alinhar com prioridades de negócio, levando a possíveis conflitos	Gheorghe et al., (2020); Sharp et al. (2019); Waja et al. (2021); Yerram et al., (2019).
<b>Ritmo de Trabalho e Tempo de Desenvolvimento</b>	Envolve etapas detalhadas como pesquisas aprofundadas, testes de usabilidade e iterações longas, demandando mais tempo.	Opera em ciclos curtos ( <i>sprints</i> ), com entregas frequentes, priorizando velocidade e eficiência.	As atividades detalhadas do UCD podem ser vistas como obstáculos ao progresso rápido do ASD, gerando conflitos nos cronogramas.	Al-Razgan et al., 2022); Matharu et al. (2015); Sharp et al. (2019); Waja et al. (2021).
<b>Envolvimento dos Usuários Finais</b>	Participação ativa e contínua dos usuários finais em todas as etapas do desenvolvimento.	Envolvimento indireto ou menos frequente do usuário final; foco no cliente ou Product Owner.	O ASD pode limitar a compreensão profunda das necessidades dos usuários finais, afetando a usabilidade do produto.	Sharp et al. (2019); Signoretti et al. (2020); Szabó; Hercegfi (2022); Omerkhel et al. (2023).
<b>Papéis e Responsabilidades na Equipe</b>	Papéis claramente definidos (designers, pesquisadores) com responsabilidades específicas.	Equipes multifuncionais e auto-organizadas; papéis não rigidamente estabelecidos.	Falta de clareza sobre responsabilidades de UX no ASD, pode dificultar a integração entre as duas metodologias.	Al-Razgan et al. (2022); Sharp et al. (2019); Sommerville (2019); Szabó; Hercegfi (2022).
<b>Documentação</b>	Requer documentação detalhada para comunicar descobertas e justificar decisões de design.	Valoriza software funcional acima de documentação extensa; mantém documentação mínima.	Documentação detalhada pode ser vista como burocrática no ASD; documentação reduzida pode afetar qualidade do design e comunicação no UCD.	Al-Razgan et al. (2022); Beck et al. (2001); Pressman e Maxim (2021); Sharp et al. (2019).
<b>Técnicas Utilizadas</b>	Usa métodos centrados no usuário, tais como: testes de usabilidade, entrevistas e prototipação.	Foca na entrega de funcionalidades conforme o <i>backlog</i> , usando <i>sprints</i> e reuniões; não incorpora formalmente técnicas de UCD.	Ausência de práticas de UCD no ASD pode limitar a compreensão das necessidades dos usuários e dificultar a integração entre as metodologias.	Sharp et al. (2019); Sommerville (2019); Szabo e Hercegfi (2022); Waja et al. (2021).

Fonte: elaborado pelo autor (2025).

### **3.4 Considerações do Capítulo**

A análise comparativa deste capítulo demonstra que, embora o UCD e o ASD compartilhem princípios comuns que facilitam sua integração, diferenças importantes, tais como ritmo de trabalho, prioridades de negócio, estrutura de equipe e principais técnicas utilizadas, podem gerar desafios se não forem gerenciadas adequadamente. Reconhecer tanto as semelhanças quanto as divergências é fundamental para o desenvolvimento de estratégias que promovam uma integração harmoniosa, aproveitando os aspectos em comum e ajustando os processos nos aspectos mais conflitantes.

No próximo capítulo, é detalhado o processo de Revisão Sistemática da Literatura, que busca identificar estudos empíricos e evidências práticas sobre a integração entre o UCD e o ASD, aprofundando a compreensão sobre como superar os desafios identificados e potencializar os benefícios da integração das referidas abordagens.

## 4. PROCESSO DE REVISÃO SISTEMÁTICA

Neste capítulo, é apresentado o protocolo adotado para a condução do processo de Revisão Sistemática da Literatura (RSL) sobre a integração do Design Centrado no Usuário (UCD) com o Desenvolvimento Ágil de Software (ASD), alinhados aos objetivos desta monografia. A RSL realizada para este estudo envolve a busca de estudos empíricos recentes nos principais portais acadêmicos, com conteúdo especializado em Engenharia de Software, visando a abrangência, a atualidade e a relevância dos estudos selecionados.

No processo de revisão sistemática adotou-se como referência o trabalho de Kitchenham e Charters (2007). Essa escolha se deve ao fato de os autores terem desenvolvido um processo de RSL específico para orientar pesquisas no contexto da Engenharia de Software, intitulado: “Guidelines for Performing Systematic Literature Reviews in Software Engineering” (KITCHENHAM; CHARTERS, 2007). Adicionalmente, a relevância do trabalho de Kitchenham e Charters é evidenciada pelo fato de ele ter sido referência para mais de dez mil pesquisas desde sua publicação (RESEARCH GATE, 2024).

### 4.1 Questões de Pesquisa

A definição das questões de pesquisa é uma etapa fundamental na RSL, pois orienta todo o processo da pesquisa (KITCHENHAM; CHARTERS, 2007).

Para alcançar os objetivos gerais e específicos desta monografia, foram formuladas as seguintes questões de pesquisa para a RSL:

QP1 – Como a integração do UCD e ASD em projetos de software ocorreu na prática?

QP2 – Quais pontos fortes, tais como benefício e vantagens, foram identificados para integração do UCD e ASD em projetos de software na prática?

QP3 – Quais pontos fracos, tais como desafios e problemas enfrentados, foram identificados para integração do UCD e ASD em projetos de software na prática?

## 4.2 Processo de Busca

Para definir o processo de busca da RSL, é necessário delimitar as bases de dados e quais os termos de busca (*strings* de busca) serão utilizados para localizar as publicações científicas relevantes (KITCHENHAM; CHARTERS, 2007).

A base de dados escolhida para conduzir a RSL foi a Scopus, por ser um dos maiores repositórios de pesquisas científicas do mundo (LEVINE-CLARK; GIL, 2021). A Scopus oferece uma ampla cobertura de publicações acadêmicas, incluindo aquelas voltadas para o UCD e para a Engenharia de Software, o que a torna particularmente adequada para os objetivos deste estudo (LEVINE-CLARK; GIL, 2021; MARTÍNEZ-FERNÁNDEZ et al., 2022).

Adicionalmente, entre os portais acessados por meio da Scopus, destaca-se o IEEE Xplore, reconhecido como uma das principais fontes de publicações científicas nas áreas de computação e Engenharia de Software (MARTÍNEZ-FERNÁNDEZ et al., 2022; TOMASZEWSKI, 2021). Além do IEEE Xplore, por meio da Scopus são consultados indiretamente os principais portais acadêmicos que incluem conteúdos especializados em Engenharia de Software, tais como ScienceDirect, ACM Digital Library e Springer (MARTÍNEZ-FERNÁNDEZ et al., 2022).

Após a definição da base de dados, é necessário definir as “*strings* de busca”, combinações de palavras-chave e operadores lógicos (*boolean*), utilizados para localizar artigos relevantes em bases de dados científicas (KITCHENHAM; CHARTERS, 2007).

As *strings* de busca desta RSL foram estruturadas de acordo com os critérios de PICOC (*Population, Intervention, Comparison, Outcome, Context*), conforme sugerido por Kitchenham e Charters (2007).

De acordo com as orientações de Kitchenham e Charters (2007), os critérios PICOC, originalmente desenvolvidos para a área médica, podem ser adaptados para a Engenharia de Software da seguinte forma:

- **População (Population):** refere-se ao grupo ou contexto de interesse em Engenharia de Software, abrangendo categorias de profissionais, sistemas ou tipos de projetos de software;
- **Intervenção (Intervention):** trata das ferramentas, métodos, práticas ou técnicas que estão sendo avaliados no estudo;
- **Comparação (Comparison):** representa uma alternativa com a qual a intervenção está sendo comparada;
- **Resultados (Outcomes):** envolve os efeitos esperados da intervenção; e
- **Contexto (Context):** representa o contexto em que o experimento é realizado.

Os critérios de PICOC desenvolvidos para este estudo, são detalhados no Quadro 5:

Quadro 5 – Critérios de PICOC

Critério	Descrição
<b>População</b>	Projetos baseados no Desenvolvimento Ágil de Software (ASD).
<b>Intervenção</b>	Adoção e integração de práticas do Design Centrado no Usuário (UCD).
<b>Comparação</b>	Não se aplica um sistema de comparação a RSL deste estudo.
<b>Resultados</b>	Compreensão de como o UCD pode ser integrado com o ASD em projetos de desenvolvimento de software.
<b>Contexto</b>	Estudos empíricos que relatam experimentos de software, tais como estudos de caso ou relatos técnicos, conduzidos em ambientes profissionais e que envolvam qualquer tipo de stakeholder.

Fonte: elaborado pelo autor, com base em Kitchenham e Charters (2007).

Com base nos critérios de PICOC e nos objetivos desta pesquisa, foram elaboradas as *strings* de busca que são apresentados no Quadro 6:

Quadro 6 – Strings de Busca

Conceito	Strings de busca (palavras-chave e operadores lógicos)
<b>Desenvolvimento Ágil de Software</b>	“agile” OR “scrum” OR “extreme programming” OR “XP” OR “kanban”
<b>Design Centrado no Usuário</b>	“user-centered” OR “user centered” OR “*centered design” OR “UCD” OR “user experience” OR “UX” OR “usability” OR “human-computer interaction” OR “HCI”
<b>Desenvolvimento de Software</b>	“development” OR “software” OR “system*” OR “app*” OR “application*”
<b>Estudos Empíricos</b>	“empirical study” OR “case study” OR “experience report”

Fonte: elaborado pelo autor, com base em Kitchenham e Charters (2007).

Nas *strings* de busca, o termo “integration” não foi incluído como palavra-chave, pois, conforme observado em uma busca preliminar, durante uma pesquisa piloto, sua utilização poderia levar à exclusão de artigos potencialmente relevantes. Dessa forma, a avaliação dos artigos que efetivamente abordam a integração entre UCD e ASD foi realizada somente em uma etapa posterior, por meio da análise individual de cada artigo durante o processo de seleção, o qual será detalhado posteriormente.

Para maior relevância dos resultados obtidos, a busca foi realizada nas três seções principais dos artigos: título, resumo (*abstract*) e palavras-chave.

Dessa forma, a *string* de busca final na base Scopus foi definido como:

( TITLE-ABS-KEY ( "agile" OR "scrum" OR "extreme programming" OR "xp" OR "kanban" ) AND TITLE-ABS-KEY ( "user-centered" OR "user centered" OR "centered design" OR "UCD" OR "user experience" OR "UX" OR "usability" OR "human-computer interaction" OR "HCI" ) AND TITLE-ABS-KEY ( "development" OR "software" OR "system\*" OR "app\*" OR "application\*" ) AND TITLE-ABS-KEY ( "Empirical study" OR "case study" OR "experience report" ) )

Como resultado do processo de busca, foram identificados 351 artigos que atendiam aos critérios iniciais.

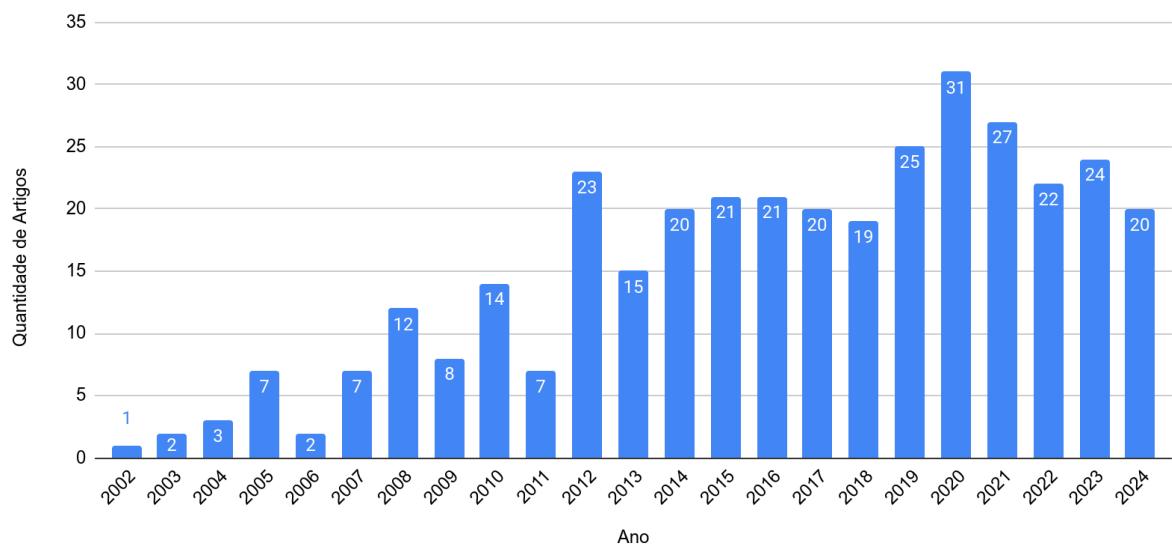
Vale ressaltar que esse resultado não contempla artigos publicados no mês de dezembro de 2024, porque a busca foi conduzida no mês de novembro de 2024.

A busca realizada pode ser observada no seguinte link (importante ressaltar que a busca para esta monografia contempla artigos disponibilizados até a data de 30 de novembro de 2024):

<https://www.scopus.com/results/results.uri?sort=r-f&src=s&sid=491cc851049bc23bac57849468796146&sot=a&sdt=a&sl=459&s=%28+TITLE-ABS-KEY+%28+%22agile%22+OR+%22scrum%22+OR+%22extreme+programming%22+OR+%22xp%22+OR+%22kanban%22+%29+AND+TITLE-ABS-KEY+%28+%22user-centered%22+OR+%22user+centered%22+OR+%22centered+design%22+OR+%22UCD%22+OR+%22user+experience%22+OR+%22UX%22+OR+%22usability%22+OR+%22human+computer+interaction%22+OR+%22human+computer+interaction%22+OR+%22HCl%22+%29+AND+TITLE-ABS-KEY+%28+%22development%22+OR+%22software%22+OR+%22system%22+OR+%22app%22+OR+%22application%22+%29+AND+TITLE-ABS-KEY+%28+%22Empirical+study%22+OR+%22case+study%22+OR+%22experience+report%22+%29+OR+%22origin=searchadvanced&editSaveSearch=&txGid=719ee1fa124928e56a0727e9866bdac3&sessionId=491cc851049bc23bac57849468796146&limit=200>

Adicionalmente, a distribuição por ano de publicação desses 351 artigos é detalhada na Figura 1.

Figura 1 – Distribuição dos Artigos por Ano de Publicação



Fonte: elaborado pelo autor, com base nos resultados da busca (2025).

### 4.3 Processo de Seleção

Para definir o processo de seleção da RSL, é necessário delimitar os critérios de inclusão e exclusão, conforme recomendado por Kitchenham e Charters (2007). Esses critérios são aplicados para filtrar e selecionar as publicações relevantes dentre os 351 artigos inicialmente identificados no processo de busca.

Os critérios de inclusão e exclusão são parâmetros pré-definidos que determinam quais estudos são considerados potencialmente relevantes (inclusão) ou desconsiderados (exclusão), com base na relevância para as questões de pesquisa (KITCHENHAM; CHARTERS, 2007).

Assim, para selecionar apenas os estudos empíricos relevantes para os objetivos deste estudo, foram definidos os critérios de inclusão e de exclusão, apresentados respectivamente nos Quadros 7 e 8.

Cada critério foi legendado com um código para referência no texto desta seção:

Quadro 7 – Critérios de Inclusão

Código	Critérios de inclusão
I1	Estudos de caso ou relato de experiência que focam na integração entre o UCD e o ASD, em projetos de desenvolvimento de software.
I2	Pesquisas realizadas em contextos profissionais reais e baseadas em projetos reais de software, desconsiderando aqueles que se caracterizam como experimentais, pilotos ou de caráter acadêmico.
I3	Artigos com acesso integral disponível para leitura e download.

Fonte: elaborado pelo autor (2025).

Quadro 8 – Critérios de Exclusão

Código	Critérios de exclusão
E1	Artigos publicados há mais de cinco anos (anteriores a 2020).
E2	Artigos não disponíveis em inglês ou português, ou sem tradução para esses idiomas.
E3	Artigos que não sejam estudos primários, como artigos teóricos ou revisões.

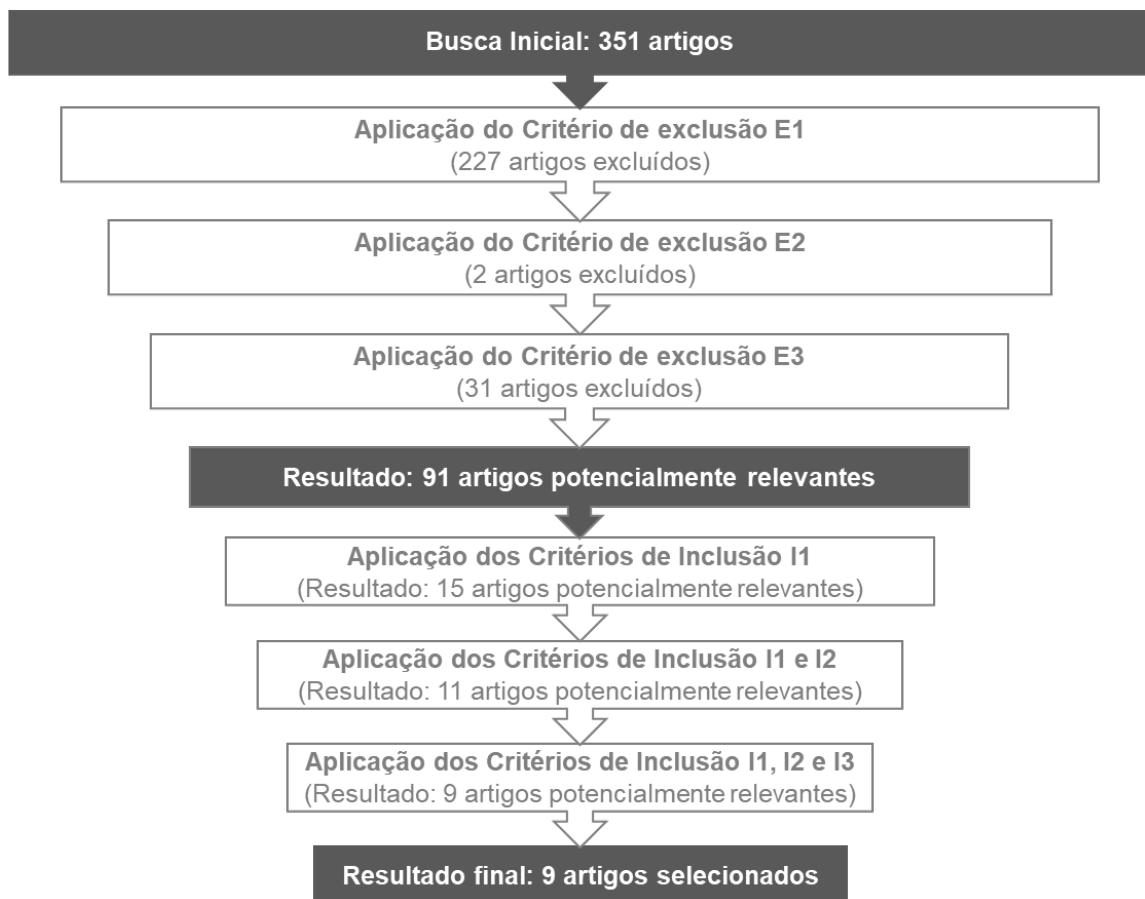
Fonte: elaborado pelo autor (2025).

Na seleção considerou-se apenas artigos que não infringiram nenhum dos critérios de exclusão e atenderam integralmente a todos os critérios de inclusão. Os critérios de seleção foram aplicados na seguinte ordem durante o processo de seleção: E1; E2; E3; I1; I2 e I3.

Com aplicação do critério de exclusão E1 (atualidade do artigo), 227 artigos foram excluídos dos 351 artigos iniciais, permanecendo 124 resultados. Após a aplicação de E2 (idioma do artigo), 2 artigos foram excluídos (um estava escrito em alemão e o outro em espanhol), permanecendo 122 resultados. Após a aplicação do critério E3(tipo de estudo), outros 31 artigos foram excluídos, e 91 artigos potencialmente relevantes permaneceram para análise a partir dos critérios de inclusão. Após a leitura detalhada dos resumos (*abstracts*) dos 91 artigos restantes, 15 artigos atenderam o critério de inclusão I1, e 11 artigos atenderam concomitantemente os critérios I1 e I2. Por fim, 2 artigos não foram possíveis de serem obtidos em função da restrição de acesso, permanecendo assim 9 artigos que atenderam concomitantemente os critérios de inclusão I1, I2 e I3.

Dessa forma, o processo de seleção é detalhado na Figura 2:

Figura 2 – Resultado do Processo de Seleção.



Fonte: elaborado pelo autor, com base no processo de seleção (2025).

Dessa forma, os nove artigos que foram considerados na seleção final são apresentados no Quadro 9.

Quadro 9 – Resultado do Processo de Seleção.

Nº	Artigos Selecionados	Referência
1	Managing User-Centered Design Activities in Distributed Agile Development	Alyahya e Almughram (2020)
2	Combining User-Centered Design and Lean Startup with Agile Software Development: A Case Study of Two Agile Teams	Signoretti et al. (2020)
3	A Case Study on Combining Agile and User-Centered Design	Pakhtusova et al. (2021)
4	Investigating Problem Definition and End-User Involvement in Agile Projects that Use Lean Inceptions	Ferreira et al. (2022)
5	Agile software development and UX design: A case study of integration by mutual adjustment	Persson et al. (2022)
6	Improving Agile Software Development using User-Centered Design and Lean Startup	Zorzetti et al. (2022)
7	The Gap Between UX Literacy and UX Practices in Agile-UX Settings: A Case Study	Azevedo et al. (2023)
8	How Good UX Design enhances the value of Quality Products: An	Moreira et al. (2023)

	Experience Report on applying Integrated Ux Lean In Scrum	
9	Exploring Priority of UCD in an Agile Development Environment: A Case Study in Iraq	Salman et al. (2023)

Fonte: elaborado pelo autor (2025)

Adicionalmente, todos os 351 artigos avaliados no processo de seleção estão listados na seção de “Apêndice” ao final desta monografia, sob o título “Planilha de apoio para seleção dos 351 artigos inicialmente buscados na RSL”. Nessa planilha, encontram-se informações detalhadas desses artigos e o modo como os critérios de exclusão e inclusão foram aplicados individualmente em cada estudo.

#### 4.4 Processo de Análise

Por fim, os 9 artigos selecionados foram utilizados como base do processo de análise, no qual são analisados e discutidos os resultados obtidos, conforme os objetivos definidos para esta monografia.

O processo de análise foi realizado por meio de uma leitura detalhada de cada artigo selecionado, onde cada estudo é avaliado de forma independente e também comparados entre si, considerando suas especificidades e os principais aspectos e contribuições relacionados às questões de pesquisa (KITCHENHAM; CHARTERS, 2007).

#### 4.5 Considerações do Capítulo

Neste capítulo, é apresentado o protocolo adotado para a condução da Revisão Sistemática da Literatura (RSL) sobre a integração do UCD com o ASD. Incluindo a definição das questões de pesquisa e o processo sistemático de busca, seleção e análise de estudos empíricos relevantes.

No próximo capítulo, são apresentados os resultados da revisão sistemática, incluindo a descrição de como a integração entre as abordagens ocorreu na prática e a análise detalhada dos benefícios e desafios identificados. E, com base nesses resultados, são propostas as diretrizes para a integração entre o UCD e o ASD.

## 5. ANÁLISE E DISCUSSÃO

Este capítulo apresenta os resultados da análise dos estudos selecionados na Revisão Sistemática da Literatura, conforme descrito no capítulo anterior. O objetivo é responder às questões de pesquisa e, com base nos resultados, propor diretrizes para a integração entre o UCD e o ASD.

### 5.1 QP1 – Como a Integração entre o UCD e o ASD ocorreu na prática?

Os estudos analisados exploraram a integração entre o Design Centrado no Usuário (UCD) e o Desenvolvimento Ágil de Software (ASD) em contextos variados, de ambientes distribuídos a organizações multinacionais e projetos locais em pequenas empresas, refletindo um amplo espectro de abordagens. Por exemplo, Alyahya e Almughram (2020) exploraram a integração em ambientes geograficamente distribuídos, enfrentando desafios como barreiras culturais e coordenação de equipes por meio de ferramentas digitais. Adicionalmente, estudos como os de Signoretti et al. (2020) e Zorzetti et al. (2022) investigaram organizações multinacionais que adotaram abordagens combinadas de UCD, Lean Startup e métodos ágeis, focando na adaptação organizacional e mudança de paradigmas. O estudo de Salman et al. (2023), por sua vez, destacou o contexto de pequenas empresas no Iraque, evidenciando os desafios específicos de incorporar o UCD em um cenário de desenvolvimento ágil local.

Na prática, a integração entre o UCD e o ASD foi implementada por meio de abordagens variadas, com destaque para a realização de atividades de design antecipado (*upfront* design), o trabalho em paralelo entre designers e desenvolvedores com comunicação contínua (“trilhas paralelas” ou “parallel tracks”) e o uso de ciclos iterativos e colaborativos (AZEVEDO ET AL., 2023; MOREIRA ET AL, 2023; PAKHTUSOVA ET AL., 2021; PERSSON ET AL., 2022). Algumas organizações utilizam ferramentas digitais para coordenar atividades entre equipes remotas de design e desenvolvimento (ALYAHYA; ALMUGHRAM, 2020). Além disso, foram destacados abordagens específicas, como o uso de métodos enxutos,

tais como Lean UX, Lean Startup e Lean Inception (FERREIRA et al., 2022; MOREIRA et al., 2023; SIGNORETTI et al., 2020; ZORZETTI et al., 2022), o uso do Diamante Duplo (SIGNORETTI et al., 2020; ZORZETTI et al., 2022) e construção de MVPs (FERREIRA et al., 2022; PAKHTUSOVA et al., 2021). Mas apesar da variedade de métodos existentes e os benefícios da integração, os estudos apontam para dificuldades recorrentes, como os desafios da sincronização entre designers e desenvolvedores e restrições de tempo e recursos para atividades do UCD em projetos ágeis (ALYAHYA; ALMUGHRAM, 2020; AZEVEDO ET AL., 2023; SALMAN et al., 2023; ZORZETTI et al., 2022).

A forma como cada estudo individualmente abordou na prática a integração entre o UCD e o ASD, é sumarizada no Quadro 10.

Quadro 10 – Como a integração ocorreu na prática

Nº	Artigo	Contexto	Como a integração ocorreu na prática	Técnicas e Métodos Utilizados
1	Alyahya e Almughram (2020)	Empresas de TI com equipes que operam em contextos distribuídos, localizadas em diferentes países (principalmente Oriente Médio) e com experiência prévia na integração de UCD e métodos ágeis.	Atividades de Design Centrado no Usuário foram adaptadas ao contexto de equipes distribuídas com uso de ferramentas digitais e desenvolvimento de novas ferramentas para coordenação entre as equipes ágeis e de design.	Design participativo, <i>mockups</i> , personas, protótipos e reuniões de alinhamento remoto. Adicionalmente, foi desenvolvido um sistema digital para melhor integrar e coordenar as equipes distribuídas.
2	Signoretti et al. (2020)	Duas equipes ágeis da área financeira, localizadas no Brasil, que desenvolveram softwares internos com foco nos usuários e usando métodos de UX e UCD.	A integração combinou UCD, métodos ágeis e uma abordagem de negócios chamada “Lean Startup”. As equipes adotaram ciclos de “construir, medir e aprender” e o conceito de “pivotar” do “Lean Startup”.	Ciclos de “construir, medir e aprender” da abordagem “Lean Startup”. Uso do XP para o desenvolvimento ágil de software. E adoção do processo do diamante duplo no caso do UCD, baseado na proposta da empresa “Pivotal Labs”.
3	Pakhtusova et al. (2021)	Projeto de pequeno porte desenvolvido na Rússia com o objetivo de facilitar a busca por cursos <i>online</i> de plataformas acadêmicas diversas. A equipe de desenvolvimento era composta por dois desenvolvedores e um designer de UX.	A integração ocorreu em duas etapas: uma etapa longa de design adiantado (chamada “Big Upfront Design”), seguida por uma etapa de Desenvolvimento Ágil Iterativo onde equipes ágeis e de UX trabalharam em paralelo, com colaboração e comunicação contínua.	Esboços de papel ( <i>paper sketch</i> ), prototipagem rápida de baixa fidelidade, prototipagem de alta-fidelidade no Figma, desenvolvimento de MVP, sessões de Ideação e testes de usabilidade.

Nº	Artigo	Contexto	Como a integração ocorreu na prática	Técnicas e Métodos Utilizados
4	Ferreira et al. (2022)	Projeto em uma grande empresa do setor de petróleo e energia no Brasil, envolvendo o desenvolvimento de ferramentas digitais para otimização de processos de refinaria.	Utilizou a abordagem "Lean Inception" (que combina Design Thinking e Lean Startup). As equipes ágeis integraram métodos de UCD no início do desenvolvimento para definir problemas e envolver usuários finais desde as fases mais iniciais do projeto.	Workshops, personas, jornadas do usuário, <i>brainstorming</i> de funcionalidades, construção de MVP e testes de usabilidade.
5	Persson et al. (2022)	Projeto industrial de uma empresa de software dinamarquesa que integrou design de UX e desenvolvimento ágil de software a partir de ajustes mútuos e contínuos entre ambas as equipes.	A integração envolveu a colaboração contínua e informal entre designers de UX e desenvolvedores, equilibrando atividades de design antecipado com trabalhos paralelos para atender às demandas de agilidade e qualidade nos projetos.	Criação de <i>wireframes</i> e protótipos no design inicial e uso de Jira e colaboração contínua entre equipes de design e equipes ágeis durante os trabalhos em paralelo.
6	Zorzetti et al. (2022)	Equipes multifuncionais (desenvolvedores de software, designers de UX e gerente de produto) de uma multinacional com sede no Brasil, desenvolvendo produtos de software em um laboratório de inovação localizado em uma universidade.	A integração envolveu UCD, Lean Startup e métodos ágeis baseando-se em 3 fases iterativas: definir o escopo e recursos ("Scoping"); explorar e moldar a solução junto aos usuários ("Discovery & framing"); e iterar com entregas ágeis constantes ("Iterations").	Uso do Diâmetro Duplo para explorar problemas e definir soluções no UCD. Ciclos de "construir, medir e aprender" do Lean Startup para validar hipóteses continuamente. E desenvolvimento de Software ágil baseado no XP.
7	Azevedo et al. (2023)	Duas empresas, uma de médio porte e outra de grande porte no setor automotivo, que desejam integrar atividades de UX no Desenvolvimento Ágil de Software e possuem diferentes níveis de compreensão em UX. O país não foi especificado.	Os designers de UX e desenvolvedores de software trabalharam de forma paralela ("trilhas paralelas"), mas interligado a partir de comunicação e colaboração contínuas entre ambas as equipes. Com testes de usabilidade sendo realizados no final de cada iteração.	Entrevistas com <i>stakeholders</i> e usuários, personas, protótipos de baixa, média e alta-fidelidade, testes de usabilidade. Treinamentos de equipes para fazer com que o conhecimento teórico de UX seja realmente aplicado na prática.
8	Moreira et al. (2023)	Projeto em uma empresa de pesquisa tecnológica localizada na região amazônica brasileira, voltado para a automação de processos industriais e integração de unidades globais.	O método Scrum foi adaptado com a utilização do método de Lean UX antes e depois do ciclo de vida da sprint, para auxiliar na construção da aplicação, e também entender os requisitos e as funcionalidades construídas.	Técnicas de "Design Upfront" com um projeto de "kick-off" para entender o problema e elaborar o projeto como base no UCD. E protótipos de baixa e alta-fidelidade para avaliar as soluções desenvolvidas.
9	Salman et al. (2023)	Pequenas empresas de software no Iraque, com foco em contextos profissionais reais de desenvolvimento ágil de software. Incluindo também o setor industrial de Bagdá, capital do Iraque.	Engenheiros de software iraquianos integraram métodos de UCD em contextos profissionais reais de desenvolvimento ágil de software, incluindo participação ativa de usuários finais.	Os métodos de UCD mais utilizados na prática nos projetos avaliados foram entrevistas e prototipação de baixa e alta-fidelidade. Seguido pelo método de observação, "survey" e grupos focais.

Fonte: elaborado pelo autor (2025)

## 5.2 QP2 – Pontos Fortes na Integração – Benefícios e Vantagens

Com base na análise de como a integração entre o UCD e o ASD ocorreu na prática, os principais pontos fortes identificados pela integração foram: i) melhoria na experiência do usuário; ii) maior colaboração e comunicação entre as equipes; iii) maior obtenção e utilização de feedback contínuo; iv) redução de retrabalho e maior eficiência a longo prazo; v) conquista de vantagem competitiva.

Um dos benefícios mais evidentes da integração entre UCD e ASD está relacionado uma compreensão mais profunda das necessidades e preferências dos usuários, que resultou na melhoria da experiência do usuário nos produtos desenvolvidos (AZEVEDO et al., 2023; SALMAN et al., 2023; ZORZETTI et al., 2022). Conforme foi observado nas experiências práticas dos artigos analisados, quando métodos de UCD são aplicados ao longo do processo de desenvolvimento, as equipes conseguem compreender com maior profundidade as reais necessidades e os comportamentos dos usuários, com impacto positivo para qualidade do produto final (AZEVEDO et al., 2023; ZORZETTI et al., 2022). Ao incorporar atividades de UCD durante o processo de desenvolvimento, a equipe alcança um entendimento mais robusto dos problemas e das demandas (SALMAN et al., 2023). Assim, os produtos resultantes tendem a oferecer interfaces mais intuitivas, maior satisfação ao usuário e menos ajustes tardios, elevando o grau de qualidade final e a melhora na experiência do usuário (AZEVEDO et al., 2023; SALMAN et al., 2023).

Além da melhoria na experiência do usuário, um segundo ponto forte observado da integração na prática é o fortalecimento da colaboração e da comunicação entre as equipes (SIGNORETTI et al., 2020; ZORZETTI et al., 2022). O ASD incentiva interações frequentes e trocas rápidas de informação, enquanto o UCD demanda envolvimento constante de designers, desenvolvedores, *stakeholders* e usuários finais (SIGNORETTI et al., 2020; ZORZETTI et al., 2022). O cenário de integração entre o UCD e o ASD permite a todos compartilhar a mesma visão de produto, compreendendo o “porquê” das decisões e assumindo responsabilidades que outrora ficavam restritas a papéis específicos (SIGNORETTI et al., 2020; ZORZETTI et al., 2022). Essa integração promove a sinergia entre os profissionais, resultando

em decisões mais acertadas e na resolução mais ágil de problemas (SIGNORETTI et al., 2020; ZORZETTI et al., 2022).

Um terceiro ponto forte fundamental na integração entre o UCD e o ASD é a obtenção e utilização contínua de feedback (AZEVEDO et al., 2023; SIGNORETTI et al., 2020). Tanto o ASD quanto o UCD enfatizam a importância de validar ideias e soluções de forma recorrente, seja por meio de revisões de *sprint*, testes de usabilidade ou interações diretas com os usuários (AZEVEDO et al., 2023; SIGNORETTI et al., 2020). Ao incorporar momentos de coleta de feedback, as equipes conseguem verificar hipóteses, ajustar escopos e adaptar soluções ainda nos estágios iniciais do desenvolvimento (AZEVEDO et al., 2023; SIGNORETTI et al., 2020). Esse feedback frequente não só reduz os riscos de construir funcionalidades desnecessárias, como também permite maior aderência às demandas reais, aumentando a chance de sucesso do produto no mercado (AZEVEDO et al., 2023; SIGNORETTI et al., 2020).

Um quarto ponto forte observado na integração entre o UCD e o ASD na prática é o aumento da eficiência a longo prazo e à redução de retrabalho (AZEVEDO et al., 2023; SIGNORETTI et al., 2020; ZORZETTI et al., 2022). Quando há uma compreensão aprofundada das necessidades do usuário desde o início, a equipe evita investir tempo e recursos em soluções que não atendam ao propósito ou que precisem de modificações significativas em fases mais avançadas (AZEVEDO et al., 2023; ZORZETTI et al., 2022). A abordagem iterativa característica do ASD, combinada à experimentação e ao envolvimento constante de usuários incentivados pelo UCD, permite identificar mais rapidamente suposições equivocadas e corrigi-las antes de causar grandes impactos (SIGNORETTI et al., 2020). Esse processo diminui a probabilidade de mudanças onerosas de última hora, otimiza o uso do tempo e facilita entregas mais constantes de valor (AZEVEDO et al., 2023; SIGNORETTI et al., 2020; ZORZETTI et al., 2022).

Por fim, um quinto ponto forte crucial, evidenciado pelos relatos e estudos empíricos analisados sobre essa integração na prática, é a conquista de vantagem competitiva e maior valor de negócio, uma vez que sinergia entre UCD e o ASD contribui para

entregas capazes de atender com mais eficácia tanto às expectativas do usuário quanto às exigências do mercado (AZEVEDO et al., 2023; MOREIRA et al., 2023; ZORZETTI et al., 2022). A colaboração multidisciplinar, o envolvimento constante de usuários e *stakeholders* e a mentalidade orientada a experimentação e melhoria contínua resultam em produtos mais atrativos, de maior valor agregado e que promovem a satisfação do usuário final (MOREIRA et al., 2023; ZORZETTI et al., 2022). Dessa forma, evidencia-se que o investimento em práticas de UX é compensado por benefícios, tais como redução de retrabalho, maior competitividade e produtos mais alinhados às necessidades reais do público-alvo (AZEVEDO et al., 2023; ZORZETTI et al., 2022).

### **5.3 QP3 – Pontos Fracos na Integração – Desafios e Problemas**

Com base na análise de como a integração entre o UCD e o ASD ocorreu na prática, os principais pontos fracos identificados pela integração foram: i) resistência cultural à integração; ii) limitação de recursos e orçamento para atividades de UCD; iii) dificuldades de sincronizar atividades de UCD e ASD; iv) dificuldade de envolver efetivamente o usuário final; v) falta de capacitação em UX em equipes ágeis.

O primeiro ponto fraco observado na prática para integração entre o ASD e o UCD é a resistência cultural e organizacional para adoção das práticas de UX em contextos ágeis (AZEVEDO et al., 2023; SALMAN et al., 2023; SIGNORETTI et al., 2020; ZORZETTI et al., 2022). Quando as equipes de desenvolvimento já possuem uma cultura enraizada em métodos ágeis consolidados ou estritamente técnicos, há uma dificuldade em aceitar práticas de UCD que priorizam pesquisa com usuários, testes de usabilidade e iterações centradas na experiência do usuário (AZEVEDO et al., 2023; SALMAN et al., 2023). Alguns profissionais resistem a mudanças, preferindo manter métodos conhecidos e fluxos de trabalho consolidados (SALMAN et al., 2023). Nessas circunstâncias, a falta de suporte ou de decisão assertiva por parte da alta gestão intensifica o problema, levando a conflitos internos e a uma possível pressão para que os métodos de UCD sejam abandonados em favor de ciclos mais curtos e entregas mais rápidas (AZEVEDO et al., 2023; SIGNORETTI et al., 2020; ZORZETTI et al., 2022).

Um segundo ponto fraco observado foi a limitação de recursos e orçamentos para adoção de métodos de UCD em contextos de desenvolvimento ágil (AZEVEDO et al., 2023; SALMAN et al., 2023). Inserir etapas de pesquisa, prototipação e testes de usabilidade na rotina de desenvolvimento demanda tempo, ferramentas adequadas e pessoal especializado (SALMAN et al., 2023). Em muitos casos, as organizações contam com equipes de UX reduzidas, que precisam se dividir em diversos projetos ao mesmo tempo, o que resulta na execução apressada ou até mesmo na exclusão de determinadas etapas de validação com usuários (AZEVEDO et al., 2023). Há situações em que os prazos e os custos apertados levam a decisões que priorizam a codificação imediata em detrimento de investigações sobre as reais necessidades e preferências de quem vai utilizar o *software*, comprometendo a qualidade das soluções criadas (AZEVEDO et al., 2023; SALMAN et al., 2023).

A pressão por entregas rápidas, intrínseca ao Desenvolvimento Ágil, também acaba gerando conflitos de agenda entre designers e desenvolvedores, configurando o terceiro ponto fraco observado na prática dessa integração, que é a dificuldade de sincronizar atividades do UCD e do ASD (ALYAHYA; ALMUGHRAM, 2020; AZEVEDO et al., 2023; SALMAN et al., 2023). Em *sprints* ágeis curtos, é possível que a equipe de desenvolvimento se veja na obrigação de iniciar a codificação antes mesmo de o design estar concluído ou validado com os usuários, o que, por sua vez, pode acarretar retrabalho e desalinhamento entre o que foi projetado e o que realmente está sendo construído (ALYAHYA; ALMUGHRAM, 2020). A carência de tempo para testes e ajustes de usabilidade dentro de cada ciclo de desenvolvimento representa um problema recorrente, dificultando a efetiva incorporação das recomendações de UX (SALMAN et al., 2023). Quando tais processos são atropelados pela necessidade de cumprir *sprints* e manter ritmo de entrega, atividades cruciais de UX são postergadas ou executadas às pressas, resultando em retrabalho e na possibilidade de soluções que não atendam plenamente às expectativas dos usuários (ALYAHYA; ALMUGHRAM, 2020; AZEVEDO et al., 2023).

O quarto ponto fraco diz respeito à dificuldade de envolver efetivamente os usuários em todas as etapas do processo de desenvolvimento (ALYAHYA; ALMUGHRAM,

2020; AZEVEDO et al., 2023; SALMAN et al., 2023). Embora o UCD dependa fortemente de feedback real, há ambientes em que a disponibilidade de usuários é limitada ou quase inexistente (ALYAHYA; ALMUGHRAM, 2020; SALMAN et al., 2023). Não raro, a equipe de UX acaba realizando testes sem a devida participação de pessoas que realmente utilizariam o sistema, ou se baseia em pressupostos e orientações genéricas (ALYAHYA; ALMUGHRAM, 2020). Essa falta de envolvimento pode ser resultado tanto da indisponibilidade do público-alvo quanto de uma cultura organizacional que não valoriza a aproximação e a empatia com o usuário final (AZEVEDO et al., 2023; SALMAN et al., 2023). Em algumas situações, a ausência de clareza nos requisitos e o subaproveitamento do feedback dos usuários geram produtos que não correspondem às necessidades e expectativas reais, o que vai contra os princípios fundamentais de um processo centrado no usuário (AZEVEDO et al., 2023; SALMAN et al., 2023).

Por fim, o desconhecimento ou falta de capacitação em UX por parte de desenvolvedores e *stakeholders* configuram um obstáculo crítico para a consolidação dessa abordagem em ambientes ágeis (AZEVEDO et al., 2023; SALMAN et al., 2023). Muitos profissionais ainda confundem UX com simples ajustes de interface ou se limitam a avaliações superficiais de usabilidade (AZEVEDO et al., 2023). A falta de treinamento e conscientização impede que a equipe de desenvolvimento compreenda a real importância de conduzir atividades de UCD desde os estágios iniciais do projeto (AZEVEDO et al., 2023). Essa lacuna de conhecimento, somada às restrições de cultura, recursos, prazos e envolvimento do usuário, pode resultar em uma integração frágil entre UCD e o ASD (AZEVEDO et al., 2023; SALMAN et al., 2023).

#### **5.4 Diretrizes para a Integração entre o UCD e o ASD**

Com base na análise da RSL de como a integração entre o UCD e o ASD ocorreu na prática e na teoria apresentada, este estudo sugere orientações para a integração entre o Design Centrado no Usuário (UCD) e o Desenvolvimento Ágil de Software (ASD). As diretrizes são apresentadas no Quadro 11, detalhando sua descrição, justificativa e formas de implementação.

Quadro 11 – Diretrizes para a Integração entre o UCD e o ASD

Nº	Diretriz	Descrição	Justificativa	Implementação	Referências
1	Investir em conscientização e treinamento em UX.	Promover capacitações regulares sobre princípios, processos e benefícios do UX para todos os envolvidos no projeto.	Aumentar a compreensão sobre processos e benefícios do UX e diminuir a resistência cultural e organizacional à integração do UCD com o ASD.	Realizar treinamentos e conscientização sobre UX que envolvam todos os <i>stakeholders</i> , incluindo gerentes e desenvolvedores. Importante também o apoio da alta gestão.	Azevedo et al. (2023); Salman et al. (2023); Signoretti et al. (2020); Zorzetti et al. (2022).
2	Promover a Colaboração Interdisciplinar entre todas as partes envolvidas no projeto de desenvolvimento.	Promover colaboração entre designers, desenvolvedores e <i>stakeholders</i> com o apoio de métodos interdisciplinares e ferramentas colaborativas.	Permitir alinhamento e comunicação eficaz entre todas as partes envolvidas no projeto. A colaboração interdisciplinar é um valor importante tanto para o UCD quanto para o ASD.	Utilizar ferramentas como Figma, Miro, Jira e realizar reuniões interdisciplinares e sessões de design colaborativo, como workshops de cocriação.	Alyahya e Almughram (2020); Moreira et al. (2023); Pakhtusova et al. (2021); Salman et al. (2023); Signoretti et al. (2020).
3	Estabelecer canais de comunicação entre as equipes.	Estabelecer canais de comunicação claros e regulares entre designers de UX, desenvolvedores e outras partes interessadas.	A comunicação eficaz reduz mal-entendidos, promove colaboração e garante que <i>insights</i> de UX sejam considerados em decisões de design e desenvolvimento.	Implementar ferramentas de comunicação, reuniões de alinhamento e canais formais e informais para priorizar melhorias.	Alyahya e Almughram (2020); Azevedo et al. (2023).
4	Organizar atividades de design antecipado antes dos <i>sprints</i> ágeis.	Planejar <i>sprints</i> antecipados de UX para pesquisa e definição de requisitos antes dos <i>sprints</i> ágeis.	O design antecipado de UX ajuda a reduzir o conflito de sincronização entre o tempo maior necessário da pesquisa aprofundada do UCD com os ciclos curtos do ASD.	Realizar antecipadamente algumas atividades de pesquisa e design de UX, como entrevistas com usuários, personas e protótipos mais elaborados.	Moreira et al. (2023); Pakhtusova et al. (2021); Persson et al. (2022).
5	Criar processos de trabalho em paralelo, entre designers de UX e desenvolvedores de software.	Sincronizar atividades de UX e desenvolvimento ágil em “trilhas paralelas” com entregas iterativas e incrementais.	Essa abordagem facilita a adaptação de mudanças para priorização do UCD ao mesmo tempo que mantém o foco em entregas rápidas e funcionais do ASD.	Criar ciclos intercalados onde o design anteceda parcialmente o desenvolvimento, envolvendo avaliações constantes de usabilidade e feedback de usuários.	Azevedo et al. (2023); Pakhtusova et al. (2021); Persson et al. (2022).
6	Assegurar priorização de tempo e recursos para as atividades de UCD.	Garantir a alocação adequada de orçamento, pessoal especializado e cronograma para execução das práticas de UX/UCD em cada <i>sprint</i> ou ciclo.	As práticas de UCD exigem tempo e investimento em pesquisa. Conforme observado na análise dos pontos fracos, a limitação de tempo, recursos e orçamento pode comprometer a qualidade do UX.	Realizar planejamento de tempo e orçamento para atividades de UX. Definir times multidisciplinares, incluindo profissionais dedicados ao UX. Assegurar o apoio da alta gestão para priorização do UCD.	Azevedo et al. (2023); Salman et al. (2023).

Nº	Diretriz	Descrição	Justificativa	Implementação	Referências
7	Envolver o usuário final ao longo de todo o processo de desenvolvimento.	Envolver o usuário ao longo de todo o processo, não apenas na validação final. E coletar feedback frequente do usuário.	O UCD depende fortemente do feedback de usuários reais ao longo de todo o ciclo de desenvolvimento. A falta de usuários reais pode comprometer a qualidade do UX.	Manter contato direto com usuários e stakeholders e incorporar feedback no <i>backlog</i> . Promover uma cultura organizacional que valorize a empatia com o usuário final. Realizar sessões de avaliação de usabilidade em cada iteração.	Alyahya e Almughram (2020); Azevedo et al. (2023); Moreira et al. (2023); Salman et al. (2023); Signoretti et al. (2020).
8	Dedicar tempo para validação contínua com os Usuários.	Realizar sessões regulares de validação com stakeholders e usuários no final de cada sprint para coletar feedback sobre as entregas e priorizar melhorias.	Permite que o produto atenda às expectativas e necessidades reais dos usuários, aumentando a satisfação do cliente e reduzindo mudanças tardias.	Dedicar tempo na <i>sprint</i> para validação de design e incluir feedback diretamente no <i>backlog</i> para implementação na próxima <i>sprint</i> .	Moreira et al. (2023); Pakhtusova et al. (2021); Persson et al. (2022).
9	Investir em ferramentas rápidas como Prototipagens Rápidas e Teste de usabilidade simplificados.	Elaborar <i>sketches</i> (desenhos rápidos) e protótipos simples para validar fluxos de navegação e conceitos de design com o Product Owner e usuários reais antes de investir em alta-fidelidade.	Evitar desperdício de tempo e recursos em alternativas de design que podem não atender às necessidades do projeto, além de facilitar a sincronização do UCD com o ritmo de entregas rápidas do ASD.	Utilizar protótipos de baixa fidelidade e outros métodos menos onerosos para realizar testes rápidos em pequena escala. Como wireframes, protótipos em papel ou ferramentas de desenhos rápidos.	Azevedo et al. (2023); Ferreira et al. (2022); Moreira et al. (2023); Pakhtusova et al. (2021); Signoretti et al. (2020).
10	Equilibrar Papéis e Responsabilidades no projeto de desenvolvimento.	Manter a composição da equipe com papéis relacionados ao design, ao software e ao negócio (Designer de UX, Gerente do Produto e Desenvolvedores), permitindo que todos participem de discussões de design, estratégicas e técnicas.	Facilitar a integração das diferentes disciplinas, promovendo uma colaboração eficaz e assegurando que todas as áreas críticas (design, desenvolvimento e gestão de produto) sejam atendidas de forma equilibrada.	Designar os papéis durante a formação da equipe e assegurar que cada membro entenda suas responsabilidades. Realizar treinamentos interdisciplinares para promover a compreensão mútua entre os diferentes papéis.	Salman et al. (2023); Signoretti et al. (2020); Zorzetti et al. (2022).
11	Incentivar uma mentalidade orientada a problemas entre as partes envolvidas no projeto.	Encorajar toda a equipe a adotar uma abordagem orientada ao problema, envolvendo-se ativamente na compreensão das necessidades do usuário antes do desenvolvimento propriamente dito.	Ao focar no problema real do usuário, evita-se desenvolver funcionalidades irrelevantes, aumenta-se a empatia e garante-se que o produto tenha valor real.	Treinar equipes para investigar problemas, envolver stakeholders em entrevistas e pesquisas, e promover engajamento em atividades de exploração inicial.	Signoretti et al. (2020); Zorzetti et al. (2022).

Fonte: elaborado pelo autor (2025)

É importante ressaltar que a aplicação na prática de recomendações e diretrizes para integrar o UCD ao ASD requer adaptações específicas, levando em consideração o contexto único de cada empresa, projeto ou equipe (AZEVEDO et al., 2023). Por fim, essas orientações têm como objetivo facilitar a integração dessas abordagens, possibilitando que o produto final atenda às necessidades dos usuários sem comprometer os valores do desenvolvimento ágil.

## 5.5 Considerações do Capítulo

Este capítulo apresentou uma análise dos estudos selecionados na RSL, respondendo às questões de pesquisa estabelecidas e identificando os principais pontos fortes e fracos na integração entre UCD e ASD. Por fim, foram sugeridas diretrizes para a integração do design centrado no usuário com métodos ágeis.

No próximo capítulo, serão apresentadas as conclusões do trabalho, bem como sugestões para pesquisas futuras que possam aprofundar o entendimento e a aplicação da integração entre UCD e ASD.

## 6. CONSIDERAÇÕES FINAIS

Esta monografia analisou a integração entre o Design Centrado no Usuário (UCD) e o Desenvolvimento Ágil de Software (ASD), com o propósito de identificar os pontos fortes e fracos dessa combinação e sugerir diretrizes para sua aplicação. A pesquisa, baseada em uma Revisão Sistemática da Literatura (RSL) que abrangeu estudos empíricos publicados entre 2020 e 2024, proporcionou informações relevantes sobre os métodos utilizados, os benefícios alcançados e os desafios enfrentados pelas equipes ao combinar essas abordagens.

### 6.1 Conclusões

Entre os principais resultados, destaca-se que UCD e ASD compartilham valores fundamentais, tais como o foco na satisfação dos *stakeholders*, a iteratividade e a ênfase no feedback. Ambos valorizam a colaboração e a comunicação eficiente entre os membros da equipe e com os *stakeholders*. No entanto, foram identificadas diferenças relevantes nos ritmos de trabalho e na priorização técnica de cada abordagem. O UCD enfatiza a importância de uma pesquisa detalhada e um design cuidadoso, enquanto o ASD privilegia entregas rápidas e incrementais, o que pode gerar conflitos na alocação de tempo e recursos para atividades de design.

Este estudo também identificou técnicas úteis para integrar as duas abordagens, tais como prototipagens rápidas, sessões de ideação, design antecipado, trabalho em paralelo com comunicação contínua (“trilhas paralelas”) e colaboração interdisciplinar. Essas técnicas possibilitam que as equipes mantenham o foco no usuário sem comprometer os valores do desenvolvimento ágil.

Entre as vantagens identificadas, a integração entre UCD e ASD possibilita o desenvolvimento de produtos que atendam tanto aos requisitos funcionais quanto às expectativas dos usuários, proporcionando o ganho de vantagem competitiva. Essa integração oferece benefícios importantes, como melhoria na experiência do usuário, conquista de vantagem competitiva, maior satisfação dos *stakeholders* e redução de retrabalho.

Por outro lado, foram identificados problemas e desafios importantes para a integração, incluindo o descompasso entre os ritmos de trabalho de UCD e ASD, a priorização de funcionalidades técnicas em detrimento do design, resistência cultural à integração, dificuldades para envolver efetivamente o usuário final e a falta de recursos e/ou capacitação em UCD dentro de equipes ágeis. Esses desafios podem comprometer a eficácia da integração caso não sejam adequadamente abordados e gerenciados.

Com base nesses resultados, diretrizes foram propostas para facilitar a integração entre o UCD e o ASD. Essas orientações oferecem suporte às equipes de desenvolvimento para equilibrar as demandas por entregas rápidas com a necessidade de métodos centrados no usuário, promovendo um ambiente de trabalho mais alinhado às necessidades do projeto.

Portanto, conclui-se que a integração entre o Design Centrado no Usuário e o Desenvolvimento Ágil de Software é não apenas viável, mas também benéfica para o desenvolvimento de produtos de software. A adoção das orientações propostas pode auxiliar as equipes a enfrentarem os desafios inerentes a essa integração, promovendo um desenvolvimento mais eficiente, alinhado tanto às demandas do mercado quanto às necessidades e preferências dos usuários finais.

## 6.2 Trabalhos Futuros

Como trabalhos futuros, sugere-se a adoção das diretrizes propostas em diferentes circunstâncias de desenvolvimento de software para avaliar sua adaptabilidade em diferentes contextos e identificar possíveis melhorias e aprimoramentos. Adicionalmente, seria relevante investigar como a integração entre UCD e ASD pode ser adaptada a diferentes tamanhos de equipe, indústrias e complexidades de projeto, bem como desenvolver ferramentas específicas para facilitar essa colaboração. Ampliar a compreensão sobre os desafios culturais e organizacionais envolvidos nessa integração também é uma oportunidade importante para pesquisas futuras.

## REFERÊNCIAS

AKHTAR, A.; BAKHTAWAR, B.; AKHTAR, S. EXTREME PROGRAMMING VS SCRUM: A COMPARISON OF AGILE MODELS. **International Journal of Technology, Innovation and Management (IJTIM)**, v. 2, n. 2, 27 out. 2022.

AL-RAZGAN, M. et al. Challenges of Integrating Agile and UX/UCD: Systematic Literature Review. **International Journal of Computer Applications**, v. 184, n. 33, p. 40–58, 20 out. 2022.

ALAIDAROS, H.; OMAR, M.; ROMLI, R. The state of the art of agile kanban method: challenges and opportunities. **Independent Journal of Management & Production**, v. 12, n. 8, p. 2535–2550, 1 dez. 2021.

ALYAHYA, S.; ALMUGHRAM, O. Managing User-Centered Design Activities in Distributed Agile Development. **Interacting with Computers**, v. 32, n. 5-6, p. 548–568, set. 2020.

AOUNI, F. E. et al. A systematic literature review on Agile, Cloud, and DevOps integration: Challenges, benefits. **Information and Software Technology**, p. 107569–107569, 1 set. 2024.

AZEVEDO, D.; LUKA RUKONIĆ; KIEFFER, S. The Gap Between UX Literacy and UX Practices in Agile-UX Settings: A Case Study. **Lecture notes in computer science**, p. 436–457, 1 jan. 2023.

BAI, X. et al. Continuous delivery of personalized assessment and feedback in agile software engineering projects. **International Conference on Software Engineering**, 27 mai. 2018.

BECK, K. et al. **Manifesto for Agile Software Development**. Agile Manifesto, 2001. Disponível em: <<https://agilemanifesto.org/iso/ptbr/manifesto.html>>. Acesso em: 20 de outubro de 2024.

BIANCHI, M. J. et al. Recommendation of Project Management Practices: A Contribution to Hybrid Models. **IEEE Transactions on Engineering Management**, v. 69, n. 6, p. 1–14, 2021.

BIANCHI, M.; MARZI, G.; DABIC, M. Guest Editorial: Agile Beyond Software—In Search of Flexibility in a Wide Range of Innovation Projects and Industries. **IEEE Transactions on Engineering Management**, v. 69, n. 6, p. 3454–3458, dez. 2022.

DWIVEDI, S. et al. A working Framework for the User-Centered Design Approach and a Survey of the available Methods. **International Journal of Scientific and Research Publications**, v. 2, n. 4, 2012.

Durak, H. et al. User-centered design in agile: Integrating psychological principles for enhanced user experience. **International Journal of Recent Advances in Multidisciplinary Research**, v. 11, n. 5, 2024

FERREIRA, B. et al. Investigating Problem Definition and End-User Involvement in Agile Projects that Use Lean Inceptions. **XX Brazilian Symposium on Software Quality**, 8 nov. 2022.

GIL, A. C. **Como elaborar projetos de pesquisa**. 7. ed. Barueri: Atlas, 2022.

GHEORGHE, A.-M.; GHEORGHE, I. D.; IATAN, I. L. Agile Software Development. **Informatica Economica**, v. 24, n. 2/2020, p. 90–100, 30 jun. 2020.

KITCHENHAM, B.; CHARTERS, S. **Guidelines for performing systematic literature reviews in software engineering**. Technical Report EBSE 2007-001, Keele University and Durham University Joint Report, 2007.

LEVINE-CLARK, M.; GIL, E. L. A new comparative citation analysis: Google Scholar, Microsoft Academic, Scopus, and Web of Science. **Journal of Business and Finance Librarianship**, v. 26, n. 1–2, p. 145–163, 2021.

MARTÍNEZ-FERNÁNDEZ, S. et al. Software Engineering for AI-Based Systems: A Survey. **ACM Transactions on Software Engineering and Methodology**, v. 31, n. 2, p. 1–59, 30 abr. 2022.

MATHARU, G. S. et al. Empirical Study of Agile Software Development Methodologies. **ACM SIGSOFT Software Engineering Notes**, v. 40, n. 1, p. 1–6, 6 fev. 2015.

MOREIRA, S. M. V. et al. How Good UX Design enhances the value of Quality Products: An Experience Report on applying Integrated UX Lean In Scrum. **International Conferences on Applied Computing 2023**. Anais...Brazil: IADIS Press, 2023.

NORMAN, D. A. **O design do dia a dia: Edição revista e ampliada**. 1. ed. Rio de Janeiro: Rocco, 2024.

OMERKHEL, Q. et al. USER INVOLVEMENT APPROACH IN AGILE SOFTWARE DEVELOPMENT: A SYSTEMATIC LITERATURE REVIEW. **Journal of Information System and Technology Management**, v. 8, n. 32, p. 01-20, 15 set. 2023.

PAKHTUSOVA, Y.; MEGHA, S.; ASKARBEKULY, N. A Case Study on Combining Agile and User-Centered Design. **Communications in Computer and Information Science**, p. 47–62, 2021.

PERSSON, J. S. et al. Agile software development and UX design: A case study of integration by mutual adjustment. **Information and Software Technology**, v. 152, p. 107059, dez. 2022.

PRESSMAN, R. S.; MAXIM, B. R. **Engenharia de Software: Uma abordagem profissional**. 9. ed. Porto Alegre: AMGH Editora Ltda, 2021.

PUTTA, A. et al. Why Do Organizations Adopt Agile Scaling Frameworks? **Proceedings of the 15th ACM / IEEE International Symposium on Empirical Software Engineering and Measurement (ESEM)**, 11 out. 2021.

RAMJAUN, U. S.; ABDULLAH, K. H.; RANA, M. E. User Centered Agile Software Development: Emerging Themes and Future Trends. p. 652–655, 25 out. 2023.

RESEARCH GATE. Research gate: Guidelines for performing Systematic Literature Reviews in Software Engineering. Citations (9,970). Disponível em: <[https://www.researchgate.net/publication/302924724\\_Guidelines\\_for\\_performing\\_Systematic\\_Literature\\_Reviews\\_in\\_Software\\_Engineering#citations](https://www.researchgate.net/publication/302924724_Guidelines_for_performing_Systematic_Literature_Reviews_in_Software_Engineering#citations)>. Acesso em: 01 de nov. de 2024.

ROHATGI, G. Agile Transformation: A Roadmap with the Scaled Agile Framework as a Guide. **Journal of Technological Innovations**, v. 4, n. 3, 2023.

SALINAS, E.; CUEVA, R.; PAZ, F. A Systematic Review of User-Centered Design Techniques. **Design, User Experience, and Usability. Interaction Design**, p. 253–267, 2020.

SALMAN, F.; BALUCH, B.; ABU BAKAR, Z. EXPLORING PRIORITY OF UCD IN AN AGILE DEVELOPMENT ENVIRONMENT: A CASE STUDY IN IRAQ. **Journal of Theoretical and Applied Information Technology**, v. 31, p. 24, 2023.

SASMITO, G. W.; FIKRI HIDAYATTULLAH, M. The Implementation of User Centered Design Methods on Public Service Mapping Websites. **IOP Conference Series: Materials Science and Engineering**, v. 1077, n. 1, p. 012022, 1 fev. 2021.

SHARP, H.; PREECE, J.; ROGERS, Y. **Interaction Design: Beyond Human-Computer Interaction**. 5. ed. Indianapolis, Indiana, USA: John Wiley & Sons, 2019.

SIGNORETTI, I. et al. Combining User-Centered Design and Lean Startup with Agile Software Development: A Case Study of Two Agile Teams. **Lecture Notes in Business Information Processing**, p. 39–55, 2020.

SOMMERVILLE, I. **Engenharia de Software: 10ª edição**. 10. ed. São Paulo: Pearson Education do Brasil, 2019.

STURM, U.; TSCHOLL, M. The role of digital user feedback in a user-centred development process in citizen science. **Journal of Science Communication**, v. 18, n. 01, 17 jan. 2019.

SZABÓ, B.; HERCEGFI, K. User-centered approaches in software development processes: Qualitative research into the practice of Hungarian companies. **Journal of Software: Evolution and Process**, v. 35, n. 2, 25 ago. 2022.

TOMASZEWSKI, R. A study of citations to STEM databases: ACM Digital Library, Engineering Village, IEEE Xplore, and MathSciNet. **Scientometrics**, v. 126, n. 2, p. 1797–1811, 1 fev. 2021.

WAJA, G.; SHAH, J.; NANAVATI, P. AGILE SOFTWARE DEVELOPMENT. **Journal of Engineering Applied Sciences and Technology**, v. 5, n. 12, 1 abr. 2021.

YERRAM, S. R. et al. Human-Centered Software Development: Integrating User Experience (UX) Design and Agile Methodologies for Enhanced Product Quality. **Asian journal of humanity, art and literature**, v. 6, n. 2, p. 203–218, 31 dez. 2019.

ZAITSEV, A.; GAL, U.; TAN, B. Reviewing the Role of the Agile Manifesto and Agile Methods in Literature. **THE AMERICAS CONFERENCE ON INFORMATION SYSTEMS**, 16 ago. 2018.

ZORZETTI, M. et al. Improving Agile Software Development using User-Centered Design and Lean Startup. **Information and Software Technology**, v. 141, p. 106718, ago. 2021.

## APÊNDICE A – Planilha de apoio para seleção dos 351 artigos inicialmente buscados na RSL

Título do artigo	Autores	Ano	Critérios		Resultado
			de Exclusão	de Inclusão	
Quantified extreme scenario based design approach	Abdallah A. et al.	2013	E1		Excluído
Agile & user centric SOA based service design framework applied in disaster management	Abdelouhab K.A. et al.	2014	E1		Excluído
OGITO, an Open Geospatial Interactive Tool to support collaborative spatial planning with a maptable	Aguilar R. et al.	2021			Não incluído
Usability and Accessibility Based Quality Evaluation of Apparel Websites in Pakistan	Ahmad M. et al.	2020			Não incluído
Reflections on Agile Software Development: A Conversion Master Case study	Ahmed A. et al.	2022			Não incluído
Managing the social aspects of software development ecosystems: An industrial case study on personality	Akarsu Z.; Yilmaz M.	2020			Não incluído
Agile beeswax: Mobile app development process and empirical study in real environment	Al-Rabaiah H.A.; Medina-Medina N.	2021			Não incluído
Integrating User Experience into Agile : An Experience Report on Lean UX and Scrum	Alhammad M.M.; Moreno A.M.	2022		I1;	Não incluído
Steel threads: Software engineering constructs for defining, designing and developing software system architecture	Alkobaisi S. et al.	2012	E1		Excluído
A Model for the Definition, Prioritization and Optimization of Indicators	Almeida F.V. et al.	2022	E3		Excluído
Agile project management: Better deliveries to the end user in software projects with a management model by Scrum	Almeida H.; Correia W.	2019	E1		Excluído
A hybrid framework HWF-PR in software engineering: MySpot-PR an application	Alvear A. et al.	2019	E1		Excluído
Managing User-Centered Design Activities in Distributed Agile Development	Alyahya S.; Almughram O.	2020		I1;I2;I3;	Incluído
Deployment and Validation of the CS-AWARE Solution at Two Pilot Sites: A Combined Agile Software Development and Design-Based Research Approach	Andriessen J. et al.	2022			Não incluído
User-centered design practices in serum development process: A distinctive advantage?	Anwar S. et al.	2014	E1		Excluído
Applying user-centered design to business modeling: CBM.next as a case study	Arar R. et al.	2018	E1		Excluído
Integrating a SCRUM-Based Process with Human Centred Design: An Experience from an Action Research Study	Ardito C. et al.	2017	E1		Excluído
The application of agile techniques for manufacturing flexibility	Arokiam I. et al.	2005	E1		Excluído
The Gap Between UX Literacy and UX Practices in Agile-UX Settings: A Case Study	Azevedo D. et al.	2023		I1;I2;I3;	Incluído
New Trends in Mobile Technologies Education in Slovakia: An Empirical Study	Babič F. et al.	2018	E1		Excluído
A Scrum-Based Development Process to Support Co-creation with Elders in the eHealth Domain	Barambones J. et al.	2020		I1;	Não incluído
Introducing usability concerns early in the DSL development cycle: FlowSL experience report	Barišić A. et al.	2014	E1		Excluído
Concept mapping in agile usability: A case study	Barksdale J.T.; McCrickard D.S.	2010	E1		Excluído
Exploring the 3-dimensional variability of websites' user-stories using triadic concept analysis	Bazin A. et al.	2024	E3		Excluído
AN OPEN SOURCE GEOVISUAL ANALYTICS TOOLBOX for MULTIVARIATE SPATIO-TEMPORAL DATA in ENVIRONMENTAL CHANGE MODELLING	Bernasocchi M. et al.	2012	E1		Excluído
Eliciting requirements for digital systems using activity and task analysis	Bezerra de Mello A.C.; Marques das Neves A.M.	2018	E1		Excluído
The HURIE method: A case study combining requirements gathering and user interface evaluation	Bias R.G. et al.	2007	E1		Excluído
Development of a Virtual Teaching Assistant System applying Agile methodology	Biswas P. et al.	2012	E1		Excluído
Development of a project tracking information system using agile methodology	Biswas P.; Schultz S.R.	2019	E1		Excluído
Designing experiential training in lean product development: A collaboration between industry & academia	Blanco E. et al.	2019	E1		Excluído
Virtual reality for prototyping service journeys	Boletsis C.	2018	E1		Excluído

User-centered design in practice: The brown university experience	Bordac S.; Rainwater J.	2008	E1		<b>Excluído</b>
Focal points for a more user-centred agile development	Bordin S.; De Angeli A.	2016	E1		<b>Excluído</b>
Inoculating an agile company with user-centred design: An empirical study	Bordin S.; De Angeli A.	2017	E1		<b>Excluído</b>
Blind search and flexible product visions: the sociotechnical shaping of generative music engines	Bown O.	2024			<b>Não incluído</b>
Flexibility during the covid-19 pandemic response: Healthcare facility assessment tools for resilient evaluation	Brambilla A. et al.	2021			<b>Não incluído</b>
My reviewers: Participatory design & crowd-sourced usability processes	Branham C. et al.	2015	E1		<b>Excluído</b>
Design Thinking and Scrum in Software Requirements Elicitation: A Case Study	Braz R.S. et al.	2019	E1		<b>Excluído</b>
Integrating design thinking and agile approaches in analytics development: The case of Aginic	Brea E. et al.	2024		I1;	<b>Não incluído</b>
Designing for participation in local social ridesharing networks - Grass roots prototyping of IT systems	Brereton M.; Ghelawat S.	2010	E1		<b>Excluído</b>
Extreme Ungrading: Rewilding the Classroom through Human-Centered Design	Brewer J.	2024			<b>Não incluído</b>
Evolving interactions: Agile design for networked media performance	Brown A.R. et al.	2009	E1		<b>Excluído</b>
Agile User Experience Design	Brown D.	2013	E1		<b>Excluído</b>
Agile User Experience Design: A Practitioner's Guide to Making It Work	Brown D.D.M.	2012	E1		<b>Excluído</b>
The role of UX professionals in agile development: A case study from industry	Bruun A. et al.	2018	E1		<b>Excluído</b>
When user experience met agile: A case study	Budwig M. et al.	2009	E1		<b>Excluído</b>
Increasing the UX maturity level of clients: A study of best practices in an agile environment	Buis E.E.G. et al.	2023			<b>Não incluído</b>
MUSE: Scaffolding metacognitive reflection in design-based research	Cabales V.	2019	E1		<b>Excluído</b>
An experience report: Using mobile development to teach software design	Campbell J.; Tafliovich A.	2015	E1		<b>Excluído</b>
Agile development of interactive installations: Two case studies	Campos P.	2010	E1		<b>Excluído</b>
Agile software development process applied to the serious games development for children from 7 to 10 years old	Cano S.P. et al.	2015	E1		<b>Excluído</b>
Focusing Extreme Programming on Usability	Carbon R. et al.	2004	E1		<b>Excluído</b>
A multipath methodology to promote ergonomics, safety and efficiency in agile factories	Ceccacci S. et al.	2019	E1		<b>Excluído</b>
Communication of design decisions and usability issues: A protocol based on personas and nielsen's heuristics	Choma J. et al.	2015	E1		<b>Excluído</b>
Understanding usability: Investigating an integrated design environment and management system	Chong Lee J. et al.	2007	E1		<b>Excluído</b>
REACT-M: The report of a case study on the application of an agile method for software requirements management; [REACT-M: O Relato de um Estudo de Caso de Aplicação de um Método Ágil para Gerência de Requisitos de Software]	da Silva B.W.F.V.; Oliveira S.R.B.	2020			<b>Não incluído</b>
Usability evaluation practices within agile development	Da Silva T.S. et al.	2015	E1		<b>Excluído</b>
Combining interaction design and agile methods for better project management	Damkul N. et al.	2009	E1		<b>Excluído</b>
Agile and UCD integration based on pre-development usability evaluations: An experience report	de Carvalho J.M.I. et al.	2016	E1		<b>Excluído</b>
AGILUS: A method for integrating usability evaluations on agile software development	de Freitas R.C. et al.	2016	E1		<b>Excluído</b>
DevOpsUse for rapid training of agile practices within undergraduate and startup communities	de Lange P. et al.	2016	E1		<b>Excluído</b>
The story map of Evandro case – development and creation of an interactive cartographic narrative	de Lima T.F. et al.	2024	E3		<b>Excluído</b>
In search of margin for business networks: The European Patent Office	Delporte-Vermeiren D. et al.	2004	E1		<b>Excluído</b>
Literature Review1	Dhandapani S.	2016	E1		<b>Excluído</b>
MetaMorph: formalization of domain-specific conceptual modeling methods—an evaluative case study, juxtaposition and empirical assessment	Döller V. et al.	2023	E3		<b>Excluído</b>
Using AI to Advance Factory Planning: A Case Study to Identify Success Factors of Implementing an AI-Based Demand Planning Solution	Dowie U.; Grothmann R.	2021			<b>Não incluído</b>
Integrating user research into an agile project at nhs digital: A case study	Duda S.; Chearman T.	2021		I1;	<b>Não incluído</b>
Lean Software Systems Engineering for Developers: Managing Requirements, Complexity, Teams, and Change Like a Champ	Durham D.; Michel C.	2021	E3		<b>Excluído</b>

Empirical studies of agile software development: A systematic review	Dybå T.; Dingsøyr T.	2008	E1		Excluído
The Implications of a Co-Created Software Solution for Mobility in Rural Areas	Eichholz L.	2023			Não incluído
An exploratory study on usage of process mining in agile software development	Erdem S.; Demirörs O.	2017	E1		Excluído
A comparative study of users' experiences with Microsoft Windows Vista and Windows XP: A case study of University of Ibadan	Esse U.C.	2013	E1		Excluído
Association rule mining for finding usability problem patterns: A case study on StackOverflow	Etemadi V. et al.	2017	E1		Excluído
UI/UX for Aerospace Qualification Business Processes	Eyisi K. et al.	2024			Não incluído
OpenUP/MMU-ISO 9241-210. Process for the human centered development of software solutions	Farinango C.D. et al.	2015	E1		Excluído
A case study of using HCI methods to improve tools for programmers	Faulring A. et al.	2012	E1		Excluído
Investigating Problem Definition and End-User Involvement in Agile Projects that Use Lean Inceptions	Ferreira B. et al.	2022		I1;I2;I3;	Incluído
Lean communication-centered design: A lightweight design process	Ferreira D.V.C.; Barbosa S.D.J.	2016	E1		Excluído
Interaction designers on eXtreme Programming teams: Two case studies from the real world	Ferreira J. et al.	2007	E1		Excluído
Design sprint in classroom: exploring new active learning tools for project-based learning approach	Ferreira V.G.; Canedo E.D.	2020			Não incluído
Navigational distances between UX information and user stories in agile virtual environments	Filho A.C.; Zaina L.A.M.	2020			Não incluído
Innovation through Universal Design in Agile UX Software Development Teams A Collaborative Case Study of an under Graduate AR Tourist Guide Project	Finn E.; Kuusinen J.	2021			Não incluído
Concurrent scheduling and layout of virtual manufacturing cells considering assembly aspects	Forghani K. et al.	2021	E3		Excluído
The right metric for the right stakeholder: A case study of improving product usability	Friedman A.; Flaounas I.	2018	E1		Excluído
Presumptive Design: Design Provocations for Innovation	Frishberg L.; Lambdin C.	2015	E1		Excluído
An experience report on using LEGO-based activities in a software engineering course	Gama K.	2019	E1		Excluído
Immersive virtual reality deployment in a lean manufacturing environment	Gamlin A. et al.	2014	E1		Excluído
Capturing & measuring emotions in UX	Garcia S.E.; Hammond L.M.	2016	E1		Excluído
UX-Painter: Fostering UX Improvement in an Agile Setting	Gardey J.C. et al.	2023			Não incluído
The development of a clinical outcomes survey research application: Assessment centerSM	Gershon R. et al.	2010	E1		Excluído
Reactive variability management in agile software development	Ghanam Y. et al.	2010	E1		Excluído
Agile Practices for Global Challenges	Ghosh B.	2023			Não incluído
Human-Centered Design Approaches to Industry 4.0 Challenges: Directions for Future Research	Gomide P. et al.	2024			Não incluído
The usefulness of usability and user experience evaluation methods on an e-Learning platform development from a developer's perspective: A case study	Gordillo A. et al.	2015	E1		Excluído
Measureability of Functional Size in Agile Software Projects: Multiple Case Studies with COSMIC FSM	Hacaloglu T.; Demirors O.	2019	E1		Excluído
Applying Stepped Task in Remote Unmoderated Test: A Case Report	He S.	2021			Não incluído
Study of factors influencing the adoption of agile processes when using Wikis	Heredia A. et al.	2014	E1		Excluído
Human-computer-machine interaction for the supervision of flexible manufacturing systems: A case study	Hernandez J.D. et al.	2020			Não incluído
Understanding information needs of agile teams to improve requirements communication	Hess A. et al.	2019	E1		Excluído
UX Poker: Estimating the Influence of User Stories on User Experience in Early Stage of Agile Development	Hinderks A. et al.	2022			Não incluído
Systematic literature reviews in agile software development: A tertiary study	Hoda R. et al.	2017	E1		Excluído
Experiences integrating sophisticated user experience design practices into Agile processes	Hodgetts P.	2005	E1		Excluído
Contextual Design: Design for Life, Second Edition	Holtzblatt K.; Beyer H.	2016	E1		Excluído
Certification and regulatory challenges in medical device software development	Hrgarek N.	2012	E1		Excluído
User evaluation support through development environment for agile software teams	Humayoun S.R. et al.	2014	E1		Excluído
A model-based approach to ongoing product evaluation	Humayoun S.R. et al.	2012	E1		Excluído

5 users every friday: A case study in applied research	Illmensee T.; Muff A.	2009	E1		<b>Excluído</b>
A Product Enhancement Methodology for Advancing Interactive Animatronic Teddy Bear	Ionica A.; Leba M.	2024			<b>Não incluído</b>
User experience design goes agile in lean transformation - A case study	Isomursu M. et al.	2012	E1		<b>Excluído</b>
Combining User Experience and Learning Efficacy in Design and Redesign	Jahnke I. et al.	2023			<b>Não incluído</b>
Using the Design Sprint process to enhance and accelerate behavioral medicine progress: a case study and guidance	Jake-Schoffman D.E.; McVay M.A.	2021			<b>Não incluído</b>
Navigating Design Science Research in mHealth Applications: A Guide to Best Practices	Jat A.S. et al.	2024			<b>Não incluído</b>
The changing ways of computer science & engineering education: A suitable pedagogy to adapt better	Jayadevappa S.; Shankar R.	2009	E1		<b>Excluído</b>
Component-based Development Using Moodle As Alternative for E-learning Software Development	Jingga K.; Sunindyo W.D.	2020			<b>Não incluído</b>
Security and human computer interfaces	Johnston J. et al.	2003	E1		<b>Excluído</b>
Orchestrating mobile applications: A software engineering view	Juárez-Ramírez R. et al.	2012	E1		<b>Excluído</b>
Integrating agile and user-centered design: A systematic mapping and review of evaluation and validation studies of agile-UX	Jurca G. et al.	2014	E1		<b>Excluído</b>
Equality = inequality: Probing equality-centric design and development methodologies	Khaled R.	2011	E1		<b>Excluído</b>
An Enhanced Agile V-Model: Conformance to regulatory bodies and experiences from model's adoption to medical device development.	Khan A.A. et al.	2024			<b>Não incluído</b>
The agile UX development lifecycle: Combining formative usability and agile methods	Kieffer S. et al.	2017	E1		<b>Excluído</b>
Contextualizing user centered design with agile methods in Ethiopia	Kifle M. et al.	2017	E1		<b>Excluído</b>
Software Development: Past, Present, and Future	Kiswani J.H. et al.	2022	E3		<b>Excluído</b>
Using an ontology for systematic practice adoption in agile methods: Expert system and practitioners-based validation	Kiv S. et al.	2022			<b>Não incluído</b>
Towards Super User-Centred Continuous Delivery: A Case Study	Klemets J.; Storholmen T.C.B.	2020			<b>Não incluído</b>
Data visualization for psychotherapy progress tracking	Koerner K. et al.	2012	E1		<b>Excluído</b>
Procuring interoperability at the expense of usability: A case study of UK National Programme for IT assurance process	Krause P.; De Lusignan S.	2010	E1		<b>Excluído</b>
Design and assessment of energetic agility measures in factories based on multivariate linear regression	Kuhlmann T.; Sauer A.	2019	E1		<b>Excluído</b>
Guidelines are a tool: Building a design knowledge management system for programmers	Kuniavsky M.; Raghavan S.	2005	E1		<b>Excluído</b>
Extended reality prototyping for transdisciplinary collaboration in product development	Küntzer L. et al.	2022	E3		<b>Excluído</b>
Animated and Unmoderated: A Usability Evaluation Case Study of A/B Designs with SAS Analytics Customers	Kurian S. et al.	2022	E3		<b>Excluído</b>
An Initial Usability Testing for Improving Acceptance Criteria in A Scrum Project: An Angkasa LMS case study	Kusumo D.S. et al.	2022			<b>Não incluído</b>
Possibilities for user-centric and participatory design in modular health care facilities	Lahtinen M. et al.	2020			<b>Não incluído</b>
A comparative study of digital human modelling simulation results and their outcomes in reality: A case study within manual assembly of automobiles	Lämkull D. et al.	2009	E1		<b>Excluído</b>
dmBridge: Building a collaborative solution for streamlined digital library design and development	Lampert C. et al.	2010	E1		<b>Excluído</b>
Tailored Design Thinking Approach - A Shortcut for Agile Teams	Lang D. et al.	2021			<b>Não incluído</b>
UX in agile before and during development	Larusdottir M.K. et al.	2018	E1		<b>Excluído</b>
A usability-pattern-based requirements-analysis method to bridge the gap between user tasks and application features	Lee S.-H. et al.	2010	E1		<b>Excluído</b>
Service Design Handover to user experience design – a systematic literature review	Leinonen A.; Roto V.	2023	E3		<b>Excluído</b>
Does maturity level influence the use of Agile UX methods by digital startups? Evaluating design thinking, lean startup, and lean user experience	Lermen F.H. et al.	2023			<b>Não incluído</b>
Introduction to the Special Issue on Usability and User Experience: Methodological Evolution	Lewis J.R.	2015	E1		<b>Excluído</b>
The role of the interaction designer in an agile software development process	Lievesley M.A.; Yee J.S.R.	2006	E1		<b>Excluído</b>
Product family design through ontology-based faceted component analysis, selection, and optimization	Liu Y. et al.	2013	E1		<b>Excluído</b>
Integrating usability evaluations into Scrum: A case study based on remote synchronous user testing	Lizano F. et al.	2014	E1		<b>Excluído</b>
Using UxD artefacts to support the writing of user stories: Findings of	Lopes L.A. et al.	2018	E1		<b>Excluído</b>

an empirical study with agile developers				
Are Virtual Reality Serious Video Games More Effective Than Web Video Games?	Lopez-Fernandez D. et al.	2023		Não incluído
Learning and Motivational Impact of Using a Virtual Reality Serious Video Game to Learn Scrum	Lopez-Fernandez D. et al.	2023		Não incluído
A guide to agile development of interactive software with a "user Objectives"-driven methodology	Losada B. et al.	2013	E1	Excluído
Applying usability engineering in InterMod agile development methodology. A case study in a mobile application	Losada B. et al.	2013	E1	Excluído
Specifying modernization into service-oriented SaaS system in a case of public transport document generator	Mahfudhi M.G.; Dias T.G.	2016	E1	Excluído
Measuring progress of scrum-based software projects	Mahnic V.; Zabkar N.	2012	E1	Excluído
Requirement changes and project success: The moderating effects of agile approaches in system engineering projects	Maierhofer S. et al.	2010	E1	Excluído
Bridging the gap between academic research and pragmatic needs in usability: A hybrid approach to usability evaluation of health care information systems	Mann D.M. et al.	2018	E1	Excluído
Redesigning educational dashboards for shifting user contexts	Manwaring E. et al.	2017	E1	Excluído
Challenges in developing software for the swedish healthcare sector	Maqbool B.; Herold S.	2021		Não incluído
Enriching user stories with usability features in a remote agile project: A case study	Marques A.B. et al.	2022		Não incluído
Do usability and agility combine? Investigating the adoption of usability modeling in an agile software project in the industry	Marques A.B. et al.	2018	E1	Excluído
How does a (Smart) age-friendly ecosystem look in a post-pandemic society?	Marston H.R. et al.	2020		Não incluído
UX Requirements Matters: Guidelines to Support Software Teams on the Writing of Acceptance Criteria	Martinelli S. et al.	2022		Não incluído
Analysis of user interaction in the schedule of consultancies and exams via smartphone; [Análise da Interação do Utilizador no Agendamento de Consultas e Exames via Smartphone]	Mártires I. et al.	2019	E1	Excluído
Correspondence of OOP Refactoring Techniques in PL\SQL Environment: A Case Study of Agile Project Code	Mashaqbeh K. et al.	2023		Não incluído
Applying lean startup: An experience report - Lean & lean UX by a UX veteran: Lessons learned in creating & launching a complex consumer app	May B.	2012	E1	Excluído
CMMI the agile way in constrained and regulated environments	McMahon P.E.	2016	E1	Excluído
Using "rapid experimentation" to inform customer service experience design	Meckem S.; Carlson J.L.	2010	E1	Excluído
Agile Requirements Engineering and Software Planning for a Digital Health Platform to Engage the Effects of Isolation Caused by Social Distancing: Case Study	Meinert E. et al.	2020		Não incluído
Analysis of pervasive games based learning systems requirements using game theory	Mejbri Y. et al.	2016	E1	Excluído
The Use of Data-Driven Personas to Facilitate Organizational Adoption–A Case Study	Miaskiewicz T.; Luxmoore C.	2017	E1	Excluído
Case study of customer input for a successful product	Miller L.	2005	E1	Excluído
A human-centred context-aware approach to develop open-standard agile ridesharing using mobile social networks	Mirisaei S.H.	2010	E1	Excluído
Decreasing rework in video games development from a software engineering perspective	Mitre-Hernández H.A. et al.	2016	E1	Excluído
On the Mapping of Underlying Concepts of a Combined Use of Lean and User-Centered Design with Agile Development: The Case Study of the Transformation Process of an IT Company	Morales C. et al.	2019	E1	Excluído
HOW GOOD UX DESIGN ENHANCES THE VALUE OF QUALITY PRODUCTS: AN EXPERIENCE REPORT ON APPLYING INTEGRATED UX LEAN IN SCRUM	Moreira S.M.V. et al.	2023	I1;I2;I3;	Incluído
A study of accessibility requirements for media players on the web	Moreno L. et al.	2011	E1	Excluído
MIDOAA: Inclusive model of development of accessible learning objects	Mourao A.B.; Netto J.F.M.	2018	E1	Excluído
We'll Take it from Here: Letting the Users Take Charge of the Evaluation and Why that Turned Out Well	Munteanu C. et al.	2013	E1	Excluído
The Impact of Scrum-XP Hybrid Methodology on Quality in Web Development with Distributed Teamwork	Mustafa N. et al.	2023		Não incluído
Developing a Desktop-based Offline Quiz Application	Mutiawani V. et al.	2020		Não incluído
Two case studies of user experience design and agile development	Najafi M.; Toyoshiba L.	2008	E1	Excluído
On designing a usable interactive system to support transplant nursing	Narasimhadhevara A. et al.	2008	E1	Excluído
An Exploratory Study about Non-functional Requirements Documentation Practices in Agile Teams	Nasir S. et al.	2023		Não incluído

A decade of veteran voices: Examining patient portal enhancements through the lens of user-centered design	Nazi K.M. et al.	2018	E1		<b>Excluído</b>
MODERN ACADEMIC LIBRARIES REGULATION: THE CASE STUDY IN THE EMERGING COUNTRY	Ncube E.D. et al.	2022			<b>Não incluído</b>
A case study in open source innovation: Developing the Tidepool Platform for interoperability in type 1 diabetes management	Neinstein A. et al.	2016	E1		<b>Excluído</b>
Integration of software engineering techniques through the use of architecture, process, and people management: An experience report	Nelson C.; Kim J.S.	2005	E1		<b>Excluído</b>
Usability in B2B Software Development	Nguyen A.; Weber C.M.	2024			<b>Não incluído</b>
Understanding Users Through Three Types of Personas	Nielsen L. et al.	2021	E3		<b>Excluído</b>
The usability expert's fear of agility - An empirical study of global trends and emerging practices	Nielsen L.; Madsen S.	2012	E1		<b>Excluído</b>
User Experience in Large-Scale Robot Development: A Case Study of Mechanical and Software Teams	Nielsen S. et al.	2023			<b>Não incluído</b>
IUCP: Estimating interactive-software project size with enhanced use-case points	Nunes N. et al.	2011	E1		<b>Excluído</b>
Learning Design Meets Service Design for Innovation in Online Learning at Scale	O'Donnell M.; Schulz L.	2020			<b>Não incluído</b>
A Systematic Mapping Study of HCI Practice Research	Ogunyemi A.A. et al.	2019	E1		<b>Excluído</b>
Data-Driven Design Process in Adoption of Marking Menus for Large Scale Software	Oh J.-Y.; Uggirala A.	2013	E1		<b>Excluído</b>
WEB APPLICATION EVALUATION AND REFACTORING: A QUALITY-ORIENTED IMPROVEMENT APPROACH	Olsina L. et al.	2008	E1		<b>Excluído</b>
Incremental quality improvement in Web applications using Web model refactoring	Olsina L. et al.	2007	E1		<b>Excluído</b>
Experiences from training agile software developers in focused workshops	Øvad T.; Larsen L.B.	2014	E1		<b>Excluído</b>
A Case Study on Combining Agile and User-Centered Design	Pakhtusova Y. et al.	2021		I1;I2;I3;	<b>Incluído</b>
Resourcing Small Indigenous Languages in the Field	Panayiotou A. et al.	2023			<b>Não incluído</b>
Software development methodologies, agile development and usability engineering	Parsons D. et al.	2007	E1		<b>Excluído</b>
Finding the forest in the trees	Patton J.	2005	E1		<b>Excluído</b>
WebTraceSense—A Framework for the Visualization of User Log Interactions	Paulino D. et al.	2024	E3		<b>Excluído</b>
StatBase: library statistics made easy	Payne A.; Curtis J.	2014	E1		<b>Excluído</b>
Proposal of a practice to conceive a minimum viable multimedia system	Pelaez Ayala C.A. et al.	2021	E2		<b>Excluído</b>
Towards a Pattern Language to Embed UX Information in Agile Software Requirements	Pereira A. et al.	2021			<b>Não incluído</b>
Towards a framework that promotes integration between the UX design and SCRUM, Aligned to CMMI	Peres A.L.; Meira S.L.	2015	E1		<b>Excluído</b>
The impact of user research on product design case study: Accessibility ecosystem for windows vista	Perkins A.; Cohene T.	2006	E1		<b>Excluído</b>
Requirements communication and balancing in large-scale software-intensive product development	Pernstål J. et al.	2015	E1		<b>Excluído</b>
Agile software development and UX design: A case study of integration by mutual adjustment	Persson J.S. et al.	2022		I1;I2;I3;	<b>Incluído</b>
Human-Centred Design Thinking Using the Intelligence Amplification Design Canvas and the Adaptive Integrated Digital Architecture Framework	Piest J.P.S. et al.	2023			<b>Não incluído</b>
Chaste: Using agile programming techniques to develop computational biology software	Pitt-Francis J. et al.	2008	E1		<b>Excluído</b>
UX design in agile: A DSDM case study	Plonka L. et al.	2014	E1		<b>Excluído</b>
A method to integrate UCD process into agile development approach	Prabhala S.; Chouinard T.	2019	E1		<b>Excluído</b>
ARP-GWO: an efficient approach for prioritization of risks in agile software development	Prakash B.; Viswanathan V.	2021			<b>Não incluído</b>
Challenges in the Development of Mobile Online Services in the Automotive Industry - A Case Study	Prenner N. et al.	2021			<b>Não incluído</b>
Combining lean and agile characteristics: Creation of virtual groups by enhanced production flow analysis	Prince J.; Kay J.M.	2003	E1		<b>Excluído</b>
Agile project management: A case study of a Virtual Research Environment development project	Procter R. et al.	2011	E1		<b>Excluído</b>
Validating mobile designs with agile testing in China: Based on baidu map for mobile	Qu J.; Zhang J.	2016	E1		<b>Excluído</b>
Design, development and validation of a mobile application for goal setting and self-monitoring of dietary behaviors	Quesada-Lopez C. et al.	2016	E1		<b>Excluído</b>
Making the whole product agile - A product owners perspective	Raithatha D.	2007	E1		<b>Excluído</b>

Tablet-based well-being check for the elderly: Development and evaluation of usability and acceptability	Ray P. et al.	2017	E1		<b>Excluído</b>
Application of a Ludic Video Game as an Alternative Resource in the Classroom. A Case Study	Rea-Peña X.M. et al.	2020			<b>Não incluído</b>
Development of a situational development methodology - A hybrid approach for agile transformation in technical design; [Entwicklung einer situativen entwicklungsmethodik - Ein hybrider Ansatz zur agilen transformation im technischen design]	Reichelt F. et al.	2021	E2		<b>Excluído</b>
Mdax: Agile generation of collaborative mdao workflows for complex systems	Risueño A.P. et al.	2020			<b>Não incluído</b>
Developing a conversational agent with a globally distributed team: An experience report	Ruane E. et al.	2020			<b>Não incluído</b>
An Active Learning Didactic Proposal with Human-Computer Interaction in Engineering Education: A Direct Current Motor Case Study	Said A. et al.	2022	E3		<b>Excluído</b>
Integrating agile development processes and user centred design - a place for usability maturity models?	Salah D. et al.	2014	E1		<b>Excluído</b>
A practitioner perspective on integrating agile and user centred design	Salah D. et al.	2014	E1		<b>Excluído</b>
Patterns for integrating agile development processes and user centred design	Salah D. et al.	2015	E1		<b>Excluído</b>
Observations on utilising usability maturity model-human centredness scale in integrating agile development processes and user centred design	Salah D. et al.	2015	E1		<b>Excluído</b>
Current Practices and Required Knowledge of Non-Functional Requirements Elicitation in Agile ContextKey Results of an Empirical Study	Saleh M. et al.	2024			<b>Não incluído</b>
EXPLORING PRIORITY OF UCD IN AN AGILE DEVELOPMENT ENVIRONMENT: A CASE STUDY IN IRAQ	Salman F.A. et al.	2023		I1;I2;I3;	<b>Incluído</b>
The value of T-GIUESE-supporting usability evaluation practices during development product: A controlled experiment	Salman F.A. et al.	2017	E1		<b>Excluído</b>
A systematic review of usability techniques in agile methodologies	Salvador C. et al.	2014	E1		<b>Excluído</b>
Um estudo de caso de aplicação de um método ágil para desenvolvimento de requisitos de software: O react	Santos K.B.C.; Oliveira S.R.B.	2018	E1		<b>Excluído</b>
Development of novel eHealth services for citizen use - Current system engineering vs. best practice in HCI	Scandurra I. et al.	2013	E1		<b>Excluído</b>
Enterprise experience into the integration of human-centered design and Kanban	Schön E.-M. et al.	2016	E1		<b>Excluído</b>
Strike while the iron is hot: User centricity adapted to the agile innovation development process	Schulz J. et al.	2017	E1		<b>Excluído</b>
The indicator browser: A web-based interface for visualizing UrbanSim simulation results	Schwartzman Y.; Borning A.	2007	E1		<b>Excluído</b>
Software Development for Text Processing in Mapping Architecturally Significant Requirements Towards Quality Attributes	Seba M.D. et al.	2024			<b>Não incluído</b>
Enterprise software experience design: Journey and lessons	Sekar B.	2017	E1		<b>Excluído</b>
A review of quantitative empirical approaches in human-computer interaction	Serrano J.F. et al.	2014	E1		<b>Excluído</b>
Integrating user-centered design practices into agile Web development: A case study	Sfetsos P. et al.	2016	E1		<b>Excluído</b>
Agile 5 using high maturity CMMI practices to improve agile processes and achieve predictable results	Sharma D. et al.	2016	E1		<b>Excluído</b>
The role of physical artefacts in agile software development: Two complementary perspectives	Sharp H. et al.	2009	E1		<b>Excluído</b>
Business process driven trust-based task scheduling	Shu J. et al.	2019	E1		<b>Excluído</b>
Boosting Agile by Using User-Centered Design and Lean Startup: A Case Study of the Adoption of the Combined Approach in Software Development	Signoretti I. et al.	2019	E1		<b>Excluído</b>
Combining User-Centered Design and Lean Startup with Agile Software Development: A Case Study of Two Agile Teams	Signoretti I. et al.	2020		I1;I2;I3;	<b>Incluído</b>
Success and Failure Factors for Adopting a Combined Approach: A Case Study of Two Software Development Teams	Signoretti I. et al.	2020		I1;I2;	<b>Não incluído</b>
Challenges and recommendations for eHealth usability evaluation with elderly users: systematic review and case study	Sinabell I.; Ammenwerth E.	2024			<b>Não incluído</b>
Agile eHealth Usability Evaluation: A Triangulative Approach to Promoting the Usability of eHealth Systems	Sinabell I.; Ammenwerth E.	2024			<b>Não incluído</b>
U-SCRUM: An agile methodology for promoting usability	Singh M.	2008	E1		<b>Excluído</b>
A case study of a framework for user-centered hardware development based on the aCar electric vehicle project	Soltes M. et al.	2018	E1		<b>Excluído</b>
HUMAN FACTORS AT THE SPEED OF RELEVANCE FOR AGILE ENGINEERING	Steinberg R.	2022			<b>Não incluído</b>
A daily dose of DSL MDE micro injections in practice	Stieglbauer G. et al.	2018	E1		<b>Excluído</b>
Empowering agile project members with accessibility testing tools: A	Stray V. et al.	2019	E1		<b>Excluído</b>

case study					
Mapping participatory design methods to the cognitive process of creativity to facilitate requirements engineering	Sulmon N. et al.	2012	E1		<b>Excluído</b>
Intelligent RAN Automation for 5G and Beyond	Sun Q. et al.	2024	E3		<b>Excluído</b>
Intuitive voting: A case study in rapid research & prototype design	Tannen R.	2015	E1		<b>Excluído</b>
A UI/UX Evaluation Framework for Blockchain-Based Applications	Tharani J.S. et al.	2022			<b>Não incluído</b>
Making sense of purpose, direction and innovation: An embedded design led innovation case study in the Australian mining industry	Townson P. et al.	2014	E1		<b>Excluído</b>
Usability testing of educational computer games on the topic safe internet	Tuparova D. et al.	2020			<b>Não incluído</b>
Design and development in the "agile room": Trialing Scrum at a Digital Agency	Tzanidou K.; Ferreira J.	2010	E1		<b>Excluído</b>
Agile user centered design: Enter the design studio - A case study	Ungar J.M.; White J.A.	2008	E1		<b>Excluído</b>
Empirical study on the documentation phase in the human-centered design process	Viana G. et al.	2019	E1		<b>Excluído</b>
The development of a platform to ensure an integrated care plan for older adults with complex care needs living at home	Villa-García L. et al.	2022			<b>Não incluído</b>
On the development of strategic games based on a Semiotic analysis: A case study of an optimized Tic-Tac-Toe	Villacís C. et al.	2016	E1		<b>Excluído</b>
The Open Data Potential for the Geospatial Characterisation of Building Stock on an Urban Scale: Methodology and Implementation in a Case Study †	Villanueva-Díaz C. et al.	2024			<b>Não incluído</b>
SCRUMIA - An educational game for teaching SCRUM in computing courses	Von Wangenheim C.G. et al.	2013	E1		<b>Excluído</b>
A Case Study in Digital Government: Developing and Applying UrbanSim, a System for Simulating Urban Land Use, Transportation, and Environmental Impacts	Waddell P.; Bornig A.	2004	E1		<b>Excluído</b>
Apathy Towards the Integration of Usability Work: A Case of System Justification	Wale-Kolade A.; Nielsen P.A.	2016	E1		<b>Excluído</b>
Integrating usability work into a large inter-organisational agile development project: Tactics developed by usability designers	Wale-Kolade A.Y.	2015	E1		<b>Excluído</b>
Integrating usability practices into agile development: A case study	Wale-Kolade A.Y. et al.	2014	E1		<b>Excluído</b>
Approaches to cross-cultural design: Two case studies with UX web-surveys	Walsh T.; Nurkka P.	2012	E1		<b>Excluído</b>
Evaluation of BehaviorMap: A user-centered behavior language	Wanderley F. et al.	2015	E1		<b>Excluído</b>
SnapMind: A framework to support consistency and validation of model-based requirements in agile development	Wanderley F. et al.	2014	E1		<b>Excluído</b>
User-centric immersive virtual reality development framework for data visualization and decision-making in infrastructure remote inspections	Wang Z. et al.	2023			<b>Não incluído</b>
Data-Driven Persona Retrospective Based on Persona Significance Index in B-to-B Software Development	Watanabe Y. et al.	2021			<b>Não incluído</b>
Methodical Framework and Case Study for Empowering Customer-Centricity in an E-Commerce Agency—The Experience Logic as Key Component of User Experience Practices Within Agile IT Project Teams	Weber B. et al.	2021			<b>Não incluído</b>
A Case Study on Implementing Agile Techniques and Practices: Rationale, Benefits, Barriers and Business Implications for Hardware Development	Weichbroth P.	2022			<b>Não incluído</b>
Meeting organisational needs and quality assurance through balancing agile and formal usability testing results	Winter J. et al.	2011	E1		<b>Excluído</b>
Agile Development of Cross-University Digital Education Ecosystems	Wolff C. et al.	2021			<b>Não incluído</b>
OJAX: A case study in agile Web 2.0 open source development	Wusteman J.	2009	E1		<b>Excluído</b>
Software project management combining Agile, Lean startup and design thinking	Ximenes B.H. et al.	2015	E1		<b>Excluído</b>
A new combined method for UCD and software development and case study	Xiong Y.; Wang A.	2010	E1		<b>Excluído</b>
A comprehensive evaluation of agile maturity self-assessment surveys	Yürüm O.R. et al.	2018	E1		<b>Excluído</b>
Usability of a Robot Avatar Designed for the Real World: The Alter-Ego X Case Study	Zambella G. et al.	2024			<b>Não incluído</b>
Transparency by Default: GDPR Patterns for Agile Development	Zieni B. et al.	2021			<b>Não incluído</b>
The use of prototypes within agile product development explorative case study of a Makeathon	Zink L. et al.	2017	E1		<b>Excluído</b>
An Empirical-informed Work Process Model for a Combined Approach of Agile, User-Centered Design, and Lean Startup	Zorzetti M. et al.	2020		I1;I2;	<b>Não incluído</b>
A Practice-Informed Conceptual Model for a Combined Approach of Agile, User-Centered Design, and Lean Startup	Zorzetti M. et al.	2020			<b>Não incluído</b>
Improving Agile Software Development using User-Centered Design and Lean Startup	Zorzetti M. et al.	2022		I1;I2;I3;	<b>Incluído</b>

JointCalc: A web-based personalised patient decision support tool for joint replacement	Zotov E. et al.	2020		Não incluído
ChildProgramming evolution, a method to increase the computational thinking skills in school	Zúñiga Muñoz R.F. et al.	2019	E1	Excluído
Proceedings - 31st EUROMICRO Conference on Software Engineering and Advanced Applications - EUROMICRO-SEAA 2005	[No Authors Found]	2005	E1	Excluído
Proceedings - Agile 2008 Conference	[No Authors Found]	2008	E1	Excluído
SAICSIT 2008: riding the wave of technology - Proceedings of the 2008 Annual Research Conference of the South African Institute of Computer Scientists and Information Technologists	[No Authors Found]	2008	E1	Excluído
Proceedings of the 2008 International Workshop on Cooperative and Human Aspects of Software Engineering, CHASE '08, Co-located with the 13th International Conference on Software Engineering, ICSE 2008	[No Authors Found]	2008	E1	Excluído
GAS'11 - Proceedings of the 1st International Workshop on Games and Software Engineering, Co-located with ICSE 2011	[No Authors Found]	2011	E1	Excluído
Proceedings - 2011 Agile Conference, Agile 2011	[No Authors Found]	2011	E1	Excluído
Proceedings - 2012 Agile Conference, Agile 2012	[No Authors Found]	2012	E1	Excluído
IASTED Multiconferences - Proceedings of the IASTED International Conference on Software Engineering, SE 2013	[No Authors Found]	2013	E1	Excluído
18th Americas Conference on Information Systems 2012, AMCIS 2012, Volume 4	[No Authors Found]	2012	E1	Excluído
18th Americas Conference on Information Systems 2012, AMCIS 2012, Volume 1	[No Authors Found]	2012	E1	Excluído
18th Americas Conference on Information Systems 2012, AMCIS 2012, Volume 5	[No Authors Found]	2012	E1	Excluído
18th Americas Conference on Information Systems 2012, AMCIS 2012, Volume 3	[No Authors Found]	2012	E1	Excluído
18th Americas Conference on Information Systems 2012, AMCIS 2012, Volume 6	[No Authors Found]	2012	E1	Excluído
18th Americas Conference on Information Systems 2012, AMCIS 2012, Volume 2	[No Authors Found]	2012	E1	Excluído
40th International Conference on Environmental Systems	[No Authors Found]	2010	E1	Excluído
Proceedings of the 13th International Workshop on Exploring Modeling Methods for Systems Analysis and Design, EMMSAD 2008 - Held in Conjunction with the CAiSE 2008 Conference	[No Authors Found]	2008	E1	Excluído
15th Ibero-American Conference on Software Engineering, CIBSE 2012	[No Authors Found]	2012	E1	Excluído
Proceedings of the 2013 10th International Conference on Information Technology: New Generations, ITNG 2013	[No Authors Found]	2013	E1	Excluído
SIGDOC 2013 - Proceedings of the 31st ACM International Conference on Design of Communication	[No Authors Found]	2013	E1	Excluído
Proceedings - 2013 ACM / IEEE International Symposium on Empirical Software Engineering and Measurement, ESEM 2013	[No Authors Found]	2013	E1	Excluído
Proceedings of the IASTED International Conference on Software Engineering, SE 2014	[No Authors Found]	2014	E1	Excluído
15th International Conference on Agile Software Development, XP 2014	[No Authors Found]	2014	E1	Excluído
11th IFIP WG 2.13 International Conference on Open Source Systems, OSS 2015	[No Authors Found]	2015	E1	Excluído
15th International Conference on Software Process Improvement and Capability Determination, SPICE 2015	[No Authors Found]	2015	E1	Excluído
2nd Extreme Programming Universe and 1st Agile Universe Conference, XP/Agile Universe 2002	[No Authors Found]	2002	E1	Excluído
17th International Conference on Human-Computer Interaction, HCI International 2015	[No Authors Found]	2015	E1	Excluído
Proceedings - 2015 Agile Conference, Agile 2015	[No Authors Found]	2015	E1	Excluído
16th International Conference on Software Process Improvement and Capability Determination, SPICE 2016	[No Authors Found]	2016	E1	Excluído
15th International Conference on Human-Computer Interaction, HCI International 2013	[No Authors Found]	2013	E1	Excluído
18th International Conference on Agile Software Development, XP 2017	[No Authors Found]	2017	E1	Excluído
Proceedings - 2017 IEEE/ACM 5th International Workshop on Conducting Empirical Studies in Industry, CESI 2017	[No Authors Found]	2017	E1	Excluído
XII Jornadas Iberoamericanas de Ingeniería de Software e Ingeniería del Conocimiento 2017, JIISIC 2017 - Held Jointly with the Ecuadorian Conference on Software Engineering, CEIS 2017 and the Conference on Software Engineering Applied to Control and Automation Systems, ISASCA 2017	[No Authors Found]	2017	E1	Excluído
Proceedings - 2017 IEEE 25th International Requirements Engineering Conference, RE 2017	[No Authors Found]	2017	E1	Excluído

18th International Conference on Product-Focused Software Process Improvement, PROFES 2017	[No Authors Found]	2017	E1		<b>Excluído</b>
CEUR Workshop Proceedings	[No Authors Found]	2017	E1		<b>Excluído</b>
24th International Working Conference on Requirements Engineering Foundation for Software Quality, REFSQ 2018	[No Authors Found]	2018	E1		<b>Excluído</b>
Confederated International International Workshop on Enterprise Integration, Interoperability and Networking, EI2N 2017, Fact Based Modeling, FBM 2017, Industry Case Studies Program, ICSP 2017, International Workshop on Methods, Evaluation, Tools and Applications for the Creation and Consumption of Structured Data for the e-Society, Meta4eS 2017, OnTheMove Academy, OTMA 2017 and ODBASE posters 2017, held as part of OTM 2017	[No Authors Found]	2018	E1		<b>Excluído</b>
Joint Proceedings of the AHFE 2018 International Conference on Human Factors in Artificial Intelligence and Social Computing, Software and Systems Engineering, The Human Side of Service Engineering and Human Factors in Energy, 2018	[No Authors Found]	2019	E1		<b>Excluído</b>
22nd AGILE Conference on Geographic Information Science, 2019	[No Authors Found]	2020	E3		<b>Excluído</b>
20th International Conference on Agile Software Development, XP 2019	[No Authors Found]	2019	E1		<b>Excluído</b>
8th International Conference on Design, User Experience, and Usability, DUXU 2019, held as part of the 21st International Conference on Human-Computer Interaction, HCI International 2019	[No Authors Found]	2019	E1		<b>Excluído</b>
Proceedings - 2018 44th Latin American Computing Conference, CLEI 2018	[No Authors Found]	2018	E1		<b>Excluído</b>
Proceedings - 2019 IEEE/ACM 41st International Conference on Software Engineering: Software Engineering Education and Training, ICSE-SEET 2019	[No Authors Found]	2019	E1		<b>Excluído</b>
10th Brazilian Workshop on Agile Methods, WBMA 2019	[No Authors Found]	2019	E1		<b>Excluído</b>
21st International Conference on Agile Software Development, XP 2020	[No Authors Found]	2020	E3		<b>Excluído</b>
12th International Conference on Social Computing and Social Media, SCSM 2020, held as part of the 22nd International Conference on Human-Computer Interaction, HCII 2020	[No Authors Found]	2020	E3		<b>Excluído</b>
8th IFIP WG 13.2 International Conference on Human-Centered Software Engineering, HCSE 2020	[No Authors Found]	2020	E3		<b>Excluído</b>
21st International Conference on Product-Focused Software Process Improvement, PROFES 2020	[No Authors Found]	2020	E3		<b>Excluído</b>
6th Ibero-American Conference on Human-Computer Interaction, HCI-Collab 2020	[No Authors Found]	2020	E3		<b>Excluído</b>
International Workshops associated with 33rd International Conference on Advanced Information Systems Engineering, CaSE 2021	[No Authors Found]	2021	E3		<b>Excluído</b>
22nd International Conference on Agile Software Development, XP 2021	[No Authors Found]	2021	E3		<b>Excluído</b>
AHFE Conferences on Human Factors in Software and Systems Engineering, Artificial Intelligence and Social Computing, and Energy, 2021	[No Authors Found]	2021	E3		<b>Excluído</b>
12th International Conference on Software Business, ICSOB 2021	[No Authors Found]	2021	E3		<b>Excluído</b>
SQBS 2020 - Proceedings of the 19th Brazilian Symposium on Software Quality	[No Authors Found]	2020	E3		<b>Excluído</b>
8th Changeable, Agile, Reconfigurable and Virtual Production Conference, CARV 2021 and 10th World Mass Customization and Personalization Conference, MCPC 2021	[No Authors Found]	2022	E3		<b>Excluído</b>
11th Brazilian Workshop on Agile Methods, WBMA 2021	[No Authors Found]	2023	E3		<b>Excluído</b>
Production Processes and Product Evolution in the Age of Disruption - Proceedings of the 9th Changeable, Agile, Reconfigurable and Virtual Production Conference CARV2023 and the 11th World Mass Customization and Personalization Conference MCPC2023	[No Authors Found]	2023	E3		<b>Excluído</b>
International KES Conference on Human Centred Intelligent Systems, KES HCIS 2023	[No Authors Found]	2023	E3		<b>Excluído</b>
workshops presented at 23rd International Conferences on Agile Software Development, XP 2022 and 24th International Conferences on Agile Software Development, XP 2023	[No Authors Found]	2024	E3		<b>Excluído</b>
ICISE 2023 - 8th International Conference on Information Systems Engineering	[No Authors Found]	2023	E3		<b>Excluído</b>