

Universidade de Brasília – UnB
Faculdade UnB Gama – FGA
Engenharia de Software

Desenvolvimento Centrado no Usuário para um processo Scrumban: um estudo de caso

Autores: Thiago Luiz de Souza Gomes e Victor Samuel dos Santos Lucas

Orientadora: Prof^a. Dr^a. Rejane Maria da Costa Figueiredo

Brasília, DF

2023



Thiago Luiz de Souza Gomes e Victor Samuel dos Santos Lucas

Desenvolvimento Centrado no Usuário para um processo Scrumban: um estudo de caso

Monografia submetida ao curso de graduação em Engenharia de Software da Universidade de Brasília, como requisito parcial para obtenção do Título de Bacharel em Engenharia de Software.

Universidade de Brasília – UnB

Faculdade UnB Gama – FGA

Orientador: Prof^a. Dr^a. Rejane Maria da Costa Figueiredo

Brasília, DF

2023

Thiago Luiz de Souza Gomes e Victor Samuel dos Santos Lucas

Desenvolvimento Centrado no Usuário para um processo Scrumban: um estudo de caso

Monografia submetida ao curso de graduação em Engenharia de Software da Universidade de Brasília, como requisito parcial para obtenção do Título de Bacharel em Engenharia de Software.

Brasília, DF

2023

Agradecimentos

Eu, Victor Samuel dos Santos Lucas, agradeço primeiramente a Deus, pois ao longo deste intenso processo, me deu forças a todo instante e permitiu que esse momento de grande felicidade fosse possível. Agradeço aos meus pais, Antônio Lucas e Gerônima Simão, por todo apoio, cuidados e amor incondicional, o suporte familiar foi fundamental para que eu acreditasse que esse sonho era possível de ser realizado. Não posso deixar de agradecer aos meus irmãos, amigos e a minha namorada por todo o apoio, neste e em todos os anos dentro da minha graduação. Por fim, mas não menos importante, agradeço a minha orientadora, Prof^a. Dr^a. Rejane Maria da Costa Figueiredo por aceitar me conduzir neste importante trabalho de pesquisa, e a todos os professores do curso pela excelência e todo o conhecimento compartilhado.

Eu, Thiago Luiz de Souza Gomes, agradeço primeiramente a Deus, pois ao longo de todo esse processo desafiador e de muitas descobertas, nunca deixou de me dá animo e curiosidade para aprender mais. Agradeço meus pais, Eduardo Luiz e Miriam de Souza, por sempre acreditar e incentivar os meus sonhos. Agradeço a minha namorada Catarina Gregório, que ao longo de todo este curso me ajudou de todas as formas possíveis a superar meus obstáculos. Não posso deixar de agradecer aos meus amigos por todo o apoio, companheirismo, suporte e todo o tipo de ajuda que sem os quais eu não estaria onde estou hoje. Por fim, agradeço a minha Orientadora Prof^a. Dr^a. Rejane Maria da Costa Figueiredo por aceitar este desafio e nos ter guiado rumo a excelência.

*"Se fiz descobertas valiosas, foi mais por ter paciência do que qualquer outro talento."
(Isaac Newton)*

Resumo

O Desenvolvimento Ágil de *Software* Centrado no Usuário refere-se à incorporação de diretrizes de *Design* Centrado no Usuário (*User-Centered Design - UCD*) em processos de Desenvolvimento Ágil de *Software*. Recentemente, essa integração ganhou popularidade e se tornou uma tendência entre as organizações que estão se esforçando progressivamente para alcançar o melhor dos dois mundos: entregar projetos dentro do prazo, sem ultrapassar limites de orçamento, atendendo aos requisitos do negócio e, simultaneamente, criar produtos e serviços utilizáveis, úteis e desejáveis para o usuário final. A incorporação dessas áreas de conhecimento é uma tarefa complexa e desafiadora, em relação à sua aplicabilidade técnica em projetos de *software*. A literatura aponta o surgimento de *frameworks* que compartilham o foco no cliente e no usuário, encorajando a integração entre eles, é o caso do *Scrumban-UCD*, que propõe um desenvolvimento fundamentado em princípios ágeis com foco no UCD. O objeto de pesquisa deste trabalho é uma *startup* cujo projeto, Groovoo, tem desafios quanto a segurança tecnológica em simultaneidade a satisfação dos usuários da aplicação com a experiência e usabilidade. O objetivo deste trabalho é propor a incorporação de diretrizes de UCD ao atual processo de desenvolvimento de *software* do projeto Groovoo. Empregando a técnica de estudo de caso, foram realizadas entrevistas com representantes do negócio, com abordagem qualitativa dos dados. Os dados coletados resultaram em uma adaptação do processo de desenvolvimento de *software* do projeto Groovoo com a incorporação de metodologias e técnicas de UCD, em conformidade ao cenário, as limitações da empresa e das equipe envolvidas. O processo adaptado possibilitou identificar os pontos de atenção e de possíveis melhorias da estratégia de tratamento e identificação das necessidades dos usuários da *startup*. Como trabalhos futuros, é possível aplicar e adaptar o processo nos demais projetos da *startup* para obtenção de uma visão do ecossistema de identificação das necessidades dos usuários.

Palavras-chave: Desenvolvimento Ágil de Software; Design Centrado no Usuário; UCD; *Scrumban-UCD*.

Abstract

The User-Centered Design (UCD) Agile Software Development refers to the integration of User-Centered Design guidelines into agile software development processes. Recently, this integration has gained popularity and become a trend among organizations striving to achieve the best of both worlds: delivering projects on time, within budget constraints, meeting business requirements, and simultaneously creating usable, useful, and desirable products and services for end users. Incorporating these knowledge areas is a complex and challenging task concerning their technical applicability in software projects. The literature highlights the emergence of frameworks that share the focus on the customer and the user, encouraging their integration. One such framework is Scrumban-UCD, which proposes a development approach based on agile principles with a UCD focus. The research object of this study is a startup whose project, Groovoo, faces challenges regarding technological security while ensuring user satisfaction with the application's experience and usability. The aim of this study is to propose the incorporation of UCD guidelines into the current software development process of the Groovoo project. To achieve this goal, qualitative data was collected through interviews with stakeholders using a case study approach. The collected data resulted in an adaptation of the Groovoo project's software development process, incorporating UCD methodologies and techniques, considering the company's scenario and limitations of the teams involved. The adapted process helped identify areas of concern and potential improvements in the strategy for identifying and addressing the needs of the startup's users.

As future work, it is possible to apply and adapt the process to other projects within the startup to gain a comprehensive understanding of the user needs identification ecosystem.

Keywords: Agile Software Development; User Center Desing; UCD; Scrumban-UCD.

Lista de ilustrações

Figura 1 – Exemplo do Quadro Scrumban	21
Figura 2 – Exemplo de alerta ao ultrapassar o limite de WIP	22
Figura 3 – Exemplo da aplicação do Scrumban	24
Figura 4 – Etapas do UCD	32
Figura 5 – Relação entre UCD e UX (Adaptada)	33
Figura 6 – Etapa Iniciação	43
Figura 7 – Etapa Iniciação Continuação	44
Figura 8 – Etapa Planejamento	45
Figura 9 – Etapa Planejamento Continuação	45
Figura 10 – Etapa de Implementação	46
Figura 11 – Etapa de Implementação Continuação	47
Figura 12 – Cronograma do projeto de pesquisa	51
Figura 13 – Processo de desenvolvimento de <i>software</i> do Projeto Groovoo	58
Figura 14 – Proposta do novo modelo Kanban	70
Figura 15 – Visão ampla do Modelo Scrumban-UCD	73
Figura 16 – Resposta do Questionário	76
Figura 17 – Resposta do Questionário	76
Figura 18 – Resposta do Questionário	77
Figura 19 – Resposta do Questionário	77
Figura 20 – Resposta do Questionário	78
Figura 21 – Resposta do Questionário	78
Figura 22 – Resposta do Questionário	79
Figura 23 – Resposta do Questionário	79
Figura 24 – Resposta do Questionário	80
Figura 25 – Resposta do Questionário	80
Figura 26 – Resposta do Questionário	81
Figura 27 – Resposta do Questionário	81
Figura 28 – Resposta do Questionário	81
Figura 29 – Carta de solicitação de acesso e citação	91
Figura 30 – Formulário do Grupo Focal	92
Figura 31 – Formulário Entrevista	95
Figura 32 – Questionário <i>Thinking Aloud</i>	99
Figura 33 – Exemplo de <i>User Flow</i>	100

Lista de tabelas

Tabela 1 – Análise comparativa entre Scrum, Kanban e Scrumban	20
Tabela 2 – Análise comparativa das técnicas (Adaptada)	31
Tabela 3 – Análise comparativa das metodologias (Adaptada)	36
Tabela 4 – Análise comparativa das técnicas UCD para Etapa Inicial (Adaptada) .	39
Tabela 5 – Análise comparativa das técnicas UCD para Etapa de Planejamento (Adaptada)	40
Tabela 6 – Análise comparativa das técnicas UCD para Etapa de Planejamento (Adaptada)	41
Tabela 7 – Análise comparativa das técnicas UCD para etapa de Implementação (Adaptada)	41
Tabela 8 – Protocolo de Estudo de Caso	49
Tabela 9 – Análise comparativa das técnicas do Scrumban selecionadas.	61
Tabela 10 – Análise comparativa das técnicas do UCD selecionadas.	63

Lista de abreviaturas e siglas

UnB	Universidade de Brasília
UCD	User-Centered Design
UX	User Experience
ROI	Return Over Investment
WIP	Work-in-Progress
CJM	Customer Journey Map
PO	Product Owner

Sumário

1	INTRODUÇÃO	13
1.1	Contexto	13
1.2	Problema	15
1.3	Objetivos	16
1.3.1	Objetivo Geral	16
1.3.2	Objetivos Específicos	16
1.4	Metodologia	16
1.4.1	Classificação Metodológica	16
1.4.2	Plano Metodológico	17
1.5	Organização do Trabalho	17
2	REFERENCIAL TEÓRICO	19
2.1	Considerações Iniciais do Capítulo	19
2.2	Scrumban	19
2.2.1	Princípios e Técnicas	21
2.2.2	Aplicação	23
2.3	<i>User-Centered Design</i> - UCD	25
2.3.1	Etapas UCD	26
2.3.2	UCD e <i>User Experience</i> - UX	32
2.3.3	UCD e <i>Agile</i>	34
2.3.4	Como um Design Centrado na Experiência do Usuário afeta positivamente os negócios?	37
2.4	Scrum-UCD	38
2.4.1	Análise comparativa entre as técnicas UCD	38
2.4.2	Metodologias Selecionadas por Argumanis, Moquilaza e Paz	42
2.4.3	Técnicas Selecionadas por Argumanis, Moquilaza e Paz	42
2.4.4	Etapas do Scrum-UCD	43
2.5	Considerações Finais do Capítulo	47
3	MATERIAIS E MÉTODOS	48
3.1	Considerações Iniciais do Capítulo	48
3.2	Planejamento de Pesquisa	48
3.2.1	<i>Background</i>	49
3.2.2	<i>Design</i>	49
3.2.3	Seleção de Caso	49
3.2.4	Procedimentos e papéis do estudo de caso	50

3.2.5	Coleta de Dados	50
3.2.6	Análise	50
3.2.7	Validade do plano	50
3.2.8	Limitações do Estudo	51
3.2.9	Relatório	51
3.2.10	Apêndices	51
3.2.11	Cronograma	51
3.3	Considerações Finais do Capítulo	52
4	ATENA SOLUTIONS	53
4.1	Considerações Iniciais do Capítulo	53
4.2	Atena Solutions	53
4.2.1	Groovoo	53
4.3	Scrumban Groovoo	53
4.3.1	Time de Produto	54
4.3.2	Scrum Master	54
4.3.3	Time de desenvolvimento	55
4.3.4	Time de Qualidade	55
4.3.5	Time de infraestrutura	55
4.3.6	Time de Design	55
4.3.7	Processos da Groovoo	55
4.4	Considerações Finais do Capítulo	58
5	DIRETRIZES DE UCD PARA O PROCESSO GROOVOO	59
5.1	Considerações Iniciais do Capítulo	59
5.2	Análise e seleção de diretrizes	59
5.2.1	Técnicas Seleccionadas Scrumban	59
5.2.2	Técnicas Seleccionadas UCD	61
5.2.3	Metodologia Seleccionada	63
5.3	Identificação de oportunidades de inserção	64
5.3.1	Inserção Scrumban	65
5.3.1.1	<i>Ready Queue</i>	65
5.3.1.2	Limite de Trabalho em Andamento (<i>Work-in-Progress - WIP</i>)	65
5.3.1.3	Políticas de Trabalho Explícitas	65
5.3.2	Inserção das Diretrizes do UCD	66
5.3.2.1	Grupo Focal	66
5.3.2.2	<i>Carding Sorting</i> ou Classificação de Cartas	67
5.3.2.3	<i>Design</i> Participativo	68
5.3.2.4	<i>Thinking Aloud</i> , ou Pensar em voz alta	69
5.3.2.5	<i>Design</i> em Paralelo ao <i>Sprint</i>	69

5.3.3	Proposta de adaptação do Processo	70
5.3.3.1	Quadro Scrumban-UCD	70
5.3.3.2	Fluxograma Scrumban-UCD	72
5.4	Avaliação, Validação e Consolidação de possibilidades de inserção .	74
5.4.1	Entrevista Semi-estruturada	74
5.4.2	Resultados	76
5.5	Considerações Finais do Capítulo	82
6	CONCLUSÃO	83
	REFERÊNCIAS	85
	APÊNDICES	90
	APÊNDICE A – CARTA DE SOLICITAÇÃO	91
	APÊNDICE B – QUESTIONÁRIO GRUPO FOCAL	92
	APÊNDICE C – QUESTIONÁRIO DA ENTREVISTA GROOVOO	95
	APÊNDICE D – QUESTIONARIO E <i>USER FLOW THINKING ALOUD</i>	99

1 Introdução

1.1 Contexto

O Manifesto Ágil no ano de 2001 - há pouco mais de duas décadas - apresentou uma mentalidade inovadora na entrega de valores e colaboração entre equipes de projetos, transformando o desenvolvimento de *software*, conduzindo o enfoque nas pessoas e não em processos ou algoritmos (SOARES, 2004). O foco do Ágil é fornecer uma abordagem mais leve ao desenvolvimento de *software*, em contradição ao tradicional "*software* pesado orientado a documentação e processos de desenvolvimento" (PATTERSON et al., 2017).

Vários *frameworks* executam a Metodologia Ágil, sendo o *Scrum* um dos mais populares (SCHWABER, 1997). O *Scrum* apresenta uma abordagem mais flexível para os membros da equipe em ambientes sujeitos a mudanças constantes. Esse *framework* atua na gestão de projetos e desenvolvimento, possibilitando aos colaboradores transparência de processos, acompanhamento e adaptação a possíveis mudanças de escopo, além de princípios de flexibilidade, adaptabilidade e produtividade em projetos, cujos requisitos de desenvolvimento sejam flexíveis. O *Scrum* é uma estrutura de processo usada para melhorar continuamente os recursos do produto, a eficiência da equipe e o ambiente de trabalho (FUIOR, 2019). Segundo Schwaber (1997), o resultado do processo deve ser um *software* que seja realmente útil para o cliente.

Já o *Kanban* é fundamentado em limitar e visualizar o trabalho em andamento, além de gerenciar o fluxo, tornando as políticas do processo explícitas pela implementação de *loops de feedback* (MIRCEA, 2019). Utilizando um sistema de visualização de todo ciclo de vida de um projeto, este *framework* tem como ênfase o *não sobrecarregamento* de membros que compõe uma equipe. Os principais fundamentos do *Kanban* são a visualização do fluxo de trabalho usando cartões, limitação do progresso do trabalho e medição do *lead time* (NIKITINA; KAJKO-MATTSSON; STRALE, 2012). Essas instruções orientam a equipe envolvida para iniciar uma nova atividade somente quando o item atual estiver concluído, e a equipe deve consentir mudanças incrementais e evolutivas, estimuladas a respeitar os atuais processos, papéis e responsabilidades.

Fundamentado nesses dois métodos, surge a metodologia híbrida *Scrumban*, do qual são obtidas as características básicas do *Scrum*, e do *Kanban* o desejo de melhoria e flexibilidade (FUIOR, 2019). Esse método incorpora aspectos dos dois *frameworks*, constituindo uma equipe formada por um time semelhante ao *Scrum*, que possui *Project Owner (PO)*, *Scrum Master* e *Scrum Team* (SENSUSE et al., 2017), e um fluxograma baseado na melhoria de processos do *Kanban*, por meio do quadro de tarefas. O método *Scrum-*

ban fornece uma abordagem por meio da divisão dos processos e atribuindo ao membro da equipe responsabilidades específicas dentro do projeto (SENSUSE et al., 2017). Dessa forma, esse método é uma combinação de práticas de *Kanban* e *Scrum* para gerenciar o desenvolvimento de *software* com base em diferentes situações de projeto (ALQUDAH; RAZALI, 2018).

A área e os profissionais de TI estão começando a observar o valor de envolver os usuários no desenvolvimento de seus produtos de *software* (PATTERSON et al., 2017). Isso tem sido reforçado por processos e métodos, como o *Design* Centrado no Usuário (*User Centered Design - UCD*) (GULLIKSEN et al., 2003). Com o emprego do UCD, busca-se desenvolver um produto (processo ou serviço) ideal, com base na compreensão das necessidades do usuário, em vez de forçá-lo a se envolver com os novos recursos do produto (ALMUGHARAM; ALYAHYA, 2017).

No entanto, o UCD não define como as equipes de desenvolvimento de *software* devem trabalhar ou como elas devem ser organizadas para alcançar os recursos de *software* prometidos (ARGUMANIS; MOQUILLAZA; PAZ, 2021). Baseado no *framework Scrum* e no UCD, o *framework Scrumban-UCD* propõe um desenvolvimento focado no *Design* Centrado no Usuário e em princípios ágeis. O *framework* busca fornecer uma metodologia de desenvolvimento incremental tratando da necessidade do usuário, colocando-o no centro do processo. Esse é o desafio do *design* e desenvolvimento de *software*: entregar produtos significativos para as pessoas, isso é, que corresponda às necessidades do usuário final (CECIL, 2006).

No contexto de UCD, o interesse na experiência do usuário tem crescido constantemente nos últimos anos. A Experiência do Usuário (*User Experience - UX*) é um campo de estudo destinado a colaborar para que um usuário percorra um processo ou produto digital com o mínimo de esforço e o máximo de valor. Enquanto o UCD se refere ao processo ou estratégia aplicada para projetar experiências, a UX lida diretamente com a experiência vivenciada pelos usuários no uso de produtos e serviços (JUSTINMIND, 2018).

Este trabalho tem como cenário a *Startup* norte-americana *Atena Solutions*, cujo objeto de estudo é o *Projeto Groovoo*. O Groovoo é uma plataforma de venda e emissão de ingressos para eventos sediados em estados norte-americanos, cujo processo de desenvolvimento é fundamentado nas metodologias ágeis *Scrum*, *Kanban* e *Scrumban*.

A *Atena Solutions* é uma *Startup* com foco no desenvolvimento de *software* e *cloud computing*, com operações em São Francisco (CA), Austin (TX), Nova York (NY), São Paulo (SP), Brasília (DF), Manaus (AM) e Vancouver (BC). Atualmente a empresa possui cerca de 17 colaboradores divididos em diversos projetos internos e de empresas parceiras. O Groovoo, selecionado para o estudo deste trabalho, é uma plataforma de venda e emissão

de ingressos para eventos sediados no estados norte-americanos, e disponível em ambientes *Web* e *Mobile*, que já processou mais de 10 milhões de reais em 18 meses.

No campo de negócios, visando competitividade no cenário norte-americano e a melhor oferta dos serviços prestados, no último semestre do ano de 2022, o Groovoo se tornou uma das primeiras empresas atuantes no ramo de venda de *tickets* e gestão de eventos na internet a implementar as tecnologias de *Tokens* não-intercambiáveis (*Non-fungible Token - NFT*) e Comunicação de Campo Próximo (*Near Field Communication - NFC*) em seus ingressos (ALGER, 2022).

Essas tecnologias surgiram para solucionar problemas de fraudes, relatados pelos *Stakeholders*. De acordo com Leonhardt (2018), cerca de 12% dos adultos na América enfrentam golpes na compra de ingressos online. Isso significa que aproximadamente 15 milhões de americanos foram vítimas de um golpe de ingresso online durante o segundo semestre de 2022, com base em cerca de 126 milhões de americanos que frequentam eventos musicais durante o verão e o outono.

1.2 Problema

Desde a idealização do Groovoo em meados de 2021, a metodologia *Scrumban* empregada pela *Atena* possibilitou uma boa gerência, visualização e comunicação entre os colaboradores. Mas se observa que o ciclo de desenvolvimento iterativo do Ágil gera alguns prazos curtos, reduzindo a quantidade de estudos necessários de *design* inicial, não garantindo que o produto desenvolvido corresponda às necessidades dos usuários, dada a falta de conhecimentos relacionados ao usuário e à sua jornada.

Dada a necessidade de procedimentos rigorosos de segurança para minimizar as fraudes no ato de aquisição de *tickets*, esses geram processos mais longos e burocráticos, e muitas vezes, dificultam a acessibilidade dos usuários finais no momento em que buscam adquirir os *tickets*. Para a *Startup*, o público alvo é, na sua maioria, brasileiros jovens ou da terceira idade, residentes nos Estados Unidos, cuja documentação americana pode não estar completa. Neste contexto, entender o cenário dos usuários, suas necessidades e desejos é crucial para que a oferta das tecnologias proporcione a melhor experiência e usabilidade, evitando que o *software* desenvolvido gere dúvidas, comportamentos inesperados e até, no pior dos casos, a não adaptação dos usuários as evoluções implementadas.

Os executivos da *Startup* estão sensíveis e favoráveis à introdução de processos e atividades iterativas fundamentadas no UCD, que possibilitem identificar se o produto desenvolvido atende as necessidades e as expectativas dos usuários, garantindo maior nível de satisfação e experiência ao interagir com a plataforma.

Neste cenário, a pergunta de pesquisa deste trabalho é: “*Como introduzir no processo de desenvolvimento de software do Groovoo as práticas do framework Scrumban-UCD para melhor reconhecer a jornada do usuário?*”

1.3 Objetivos

1.3.1 Objetivo Geral

O objetivo deste trabalho é *propor diretrizes de UCD para o processo de desenvolvimento de software do projeto Groovoo da Startup Atena Solutions.*

1.3.2 Objetivos Específicos

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

- Analisar e selecionar diretrizes empregadas pelo *framework Scrum-UCD* em relação ao *Scrumban* e diretrizes UCD da literatura;
- Identificar no processo de desenvolvimento de software Groovoo oportunidades de inserção de diretrizes do UCD;
- Avaliar e validar possibilidades de inserção de diretrizes UCD no *Processo de desenvolvimento de Software do Groovoo.*

1.4 Metodologia

A metodologia de desenvolvimento deste trabalho foi definida e classificada e um plano metodológico foi estruturado.

1.4.1 Classificação Metodológica

Dado que o objetivo deste trabalho é propor, baseado na literatura, diretrizes para um processo já existente, esta pesquisa é de natureza aplicada, pois visa se concentrar em fornecer soluções práticas para a resolução de um problema levantado.

A abordagem é qualitativa, pois os resultados obtidos partem das percepções dos indivíduos envolvidos, dos conflitos observados em campo e dos aspectos subjetivos identificados. E não fará uso de dados estatísticos.

A tipologia é classificada como descritiva, pois é fundamentada na descrição de um fenômeno, com métodos e características de um objeto.

A técnica empregada é o Estudo de caso, apoiado nas técnicas de revisão bibliográfica e documental.

1.4.2 Plano Metodológico

O plano metodológico é baseado em [Brereton et al. \(2008\)](#), que apresenta um *Protocolo de Estudo de Caso*, com vários itens a serem desenvolvidos, e esses distribuídos em quatro grandes fases: Planejamento; Coleta de Dados; Análise de Dados; e Relatórios.

Essas fases compreendem, de forma resumida:

- **Planejamento da Pesquisa:** nessa fase apresenta-se o contexto do pesquisa com a revisão bibliográfica, pergunta de pesquisa, definição dos objetivos do trabalho e a definição de um plano metodológico, com as escolhas metodológicas;
- **Coleta dos Dados:** essa fase compreende o levantamento e a aplicação das técnicas de coleta de dados como revisão documental, revisão bibliográfica e estudo de caso;
- **Análise dos Dados:** nessa fase realizam-se a interpretação e análise dos dados coletados, assim como a análise da validade do trabalho, cuja meta é avaliar a percepção dos gestores da empresa em relação as diretrizes propostas;
- **Relatório:** por fim, o relatório é constituído pelos resultados deste trabalho e caracterizado por esta monografia.

O planejamento metodológico do *Protocolo de Estudo de Caso* ([BRERETON et al., 2008](#)) é apresentado no Capítulo Proposta (REVISAR - ALINHAR COM NOVA ESTRUTURA).

1.5 Organização do Trabalho

O trabalho está disposto em 4 capítulos. Neste, **Capítulo 1 - Introdução**, compreende a introdução do trabalho, com a contextualização junto a revisão bibliográfica, o problema, objetivos de pesquisa, e a metodologia aplicada. Ademais, os restantes estão estruturados no seguinte formato:

- **Capítulo 2 - Referencial Teórico:** Descreve conceitos acerca de Metodologias Ágeis junto a alguns dos seus principais métodos, bem como os potenciais benefícios e obstáculos relacionados à sua aplicação. O capítulo é subdividido nas seções *Scrumban*, *Design Centrado no Usuário (User-Centered Design - UCD)*, e o *framework Scrum-UCD*;
- **Capítulo 3 - Materiais e Métodos:** Neste capítulo é retomado o objetivo e o plano metodológico adotado, onde são apresentados os métodos e materiais selecionados, juntamente com o detalhamento de cada componente do Protocolo de Estudo de Caso adotado nesta pesquisa;

- **Capítulo 4 - Atena Solutions:** Neste capítulo é apresentado a *Startup Atena Solutions* e o projeto Groovoo. Além disso, é apresentada divisão de colaboradores que estruturam o projeto e seu processo de desenvolvimento de software fundamentado em metodologias ágeis.
- **Capítulo 5 - Diretrizes de UCD para o processo Groovoo:** Neste capítulo é apresentado as diretrizes UCD selecionadas, com base na literatura, para o processo de desenvolvimento do Projeto Groovoo da *Startup Atena*, suas oportunidades de inserção e a consolidação e validação da proposição destas diretrizes por meio de questionário e entrevista.
- **Capítulo 6 - Conclusão:** Neste capítulo são retomados os objetivos específicos deste relatório com seus respectivos resultados, e por fim é apresentada uma conclusão sobre esta monografia.
- **Apêndices:** Os documentos produzidos pelos autores deste trabalho, como a carta de apresentação à empresa e os formulários utilizados, constam como apêndices.
 - Apêndice *A* - *Carta de Solicitação*
 - Apêndice *B* - *Questionário Grupo Focal*
 - Apêndice *C* - *Questionário da Entrevista Groovoo*
 - Apêndice *D* - *Questionario e User Flow Thinking Aloud*

2 Referencial Teórico

2.1 Considerações Iniciais do Capítulo

Neste capítulo são apresentadas as bases conceituais deste trabalho de pesquisa. O *Scrumban*, *framework* adotado pelo *Startup Atena* para o processo de desenvolvimento de *software* do Groovoo, objeto de estudo deste trabalho. Em seguida, uma seção sobre o tema *Design Centrado no Usuário (User Centered Design - UCD)*, bem como os potenciais benefícios e desafios relacionados a sua adoção. Por fim, ocorre um detalhamento do *framework Scrum-UCD*, para a proposição de uma abordagem fundamentada na Experiência do Usuário (*User Experience - UX*) ao processo de desenvolvimento do Groovoo.

2.2 Scrumban

O *Scrumban* é a combinação do *Framework Scrum* e do Método *Kanban*, reunindo recursos de ambos ao desenvolvimento de *software*. Por um lado, utiliza o caráter prescritivo do *Scrum* para ser ágil; por outro, incentiva a melhoria do processo do *Kanban* para permitir que as equipes melhorem continuamente seus processos (KHAN, 2014).

O autor Khan (2014) apresentou uma comparação entre o *Scrum*, *Kanban*, *Scrumban*. Na Tabela 1 apresenta-se uma evolução dessa comparação, com outras fontes e novos dados. Conforme a tabela, o *Scrumban* é semelhante ao *Kanban* nos critérios de visualização, *backlog*, mudanças no escopo, equipe, quadro, trabalho em equipe e impedimentos. Enquanto os tópicos: papéis, interação, cerimônias, seguem semelhantes ao *Scrum*.

É observado acerca da tabela de comparação, que o *Scrum* é rígido por não aderir tarefas de altas prioridades que não foram decididas para aquele *Sprint*. E que o *Kanban*, embora sozinho possa ajudar a equipe a gerenciar o fluxo de trabalho definindo limites de trabalho em andamento e melhorar ainda mais a colaboração da equipe definindo as políticas de trabalho explícita, ainda não é o suficiente para abranger todos os problemas da equipe.

O time precisa de um processo orientado à iteração para que os lançamentos da *Sprint* estejam sincronizados com outras equipes, com o suporte das cerimônias do *Scrum* para maior envolvimento da equipe, por isso o *Scrumban* une de forma flexível estes dois *frameworks* para melhor desenvolvimento do projeto e da equipe (KHAN, 2014).

Tabela 1 – Análise comparativa entre Scrum, Kanban e Scrumban

	Scrum	Kanban	Scrumban
Vista do fluxo de trabalho	Backlog, Progresso das tarefas e Feito (CAGLEY, 2013)	Visualização total (CAGLEY, 2013)	Visualização total (CAGLEY, 2013)
Backlog	Sprint e <i>Product</i> (KHAN, 2014)	<i>Backlog</i> com limites (KHAN, 2014)	<i>Backlog</i> com limites (KHAN, 2014)
Limite de WIPs	WIPs são definidas indiretamente pelas sprints (KNIBERG, 2009)	WIPs são limitadas diretamente pelo estágio do fluxo de trabalho (KNIBERG, 2009)	WIPs são limitadas diretamente pelo estágio do fluxo de trabalho (KNIBERG, 2009)
Mudanças no escopo	Próxima sprint (KHAN, 2014)	Adicione conforme a necessidade (RADICS, 2013)	Adicione conforme a necessidade (RADICS, 2013)
Papéis	<i>Product Owners, Scrum Masters, Team</i> (KHAN, 2014)	Conforme a necessidade (RADICS, 2013)	Times do Scrum mais os papéis conforme as necessidades (PAHUJA, 2012)
Equipe	Multifuncional (PAHUJA, 2012)	Pode ser multifuncional ou especializada (RADICS, 2013)	Pode ser multifuncional ou especializada (RADICS, 2013)
Estimação	Com <i>story points</i> ou dias (KHAN, 2014)	Não utiliza (KHAN, 2014)	Opcional (KNIBERG, 2009)
Quadro	<i>Quadro Burn-down</i> (KHAN, 2014)	Quadro mapeado com o processo (RADICS, 2013)	Quadro mapeado com o processo (RADICS, 2013)
Iteração	Com sprints (KHAN, 2014)	Não, deve seguir o fluxo de trabalho contínuo (RADICS, 2013)	Com sprints (KHAN, 2014)
Trabalho em equipe	Colaboração conforme as necessidades da tarefa (KHAN, 2014)	<i>Swarming</i> para alcançar os objetivos da equipe (RADICS, 2013)	<i>Swarming</i> para alcançar os objetivos da equipe (RADICS, 2013)
Impedimentos	Informar a equipe imediatamente (KHAN, 2014)	Informar a equipe e realizar o <i>Swarm</i> para resolver o impedimento (RADICS, 2013)	Informar a equipe e realizar o <i>Swarm</i> para resolver o impedimento (RADICS, 2013)
Priorização	PO prepara o <i>backlog</i> (KHAN, 2014)	Priorização é opcional (KNIBERG, 2009)	Fora do processo, deve ocorrer a priorização do <i>backlog</i> (RADICS, 2013)
Cerimonias	<i>Daily, Planning, Review, Restrospective</i> (KHAN, 2014)	Não é necessário (RADICS, 2013)	<i>Daily, Planning, Retro</i> conforme a necessidade (PAHUJA, 2012)
Métricas	Usa o <i>velocity</i> (KNIBERG, 2009)	Usa o <i>lead time</i> (KNIBERG, 2009)	<i>Velocity</i> é opcional Usa o <i>lead time</i> (KHAN, 2014)
Quadro Scrum	Recomeça após cada sprint (CAGLEY, 2013)	O quadro permanece (KNIBERG, 2009)	Depende da decisão do time (CAGLEY, 2013)

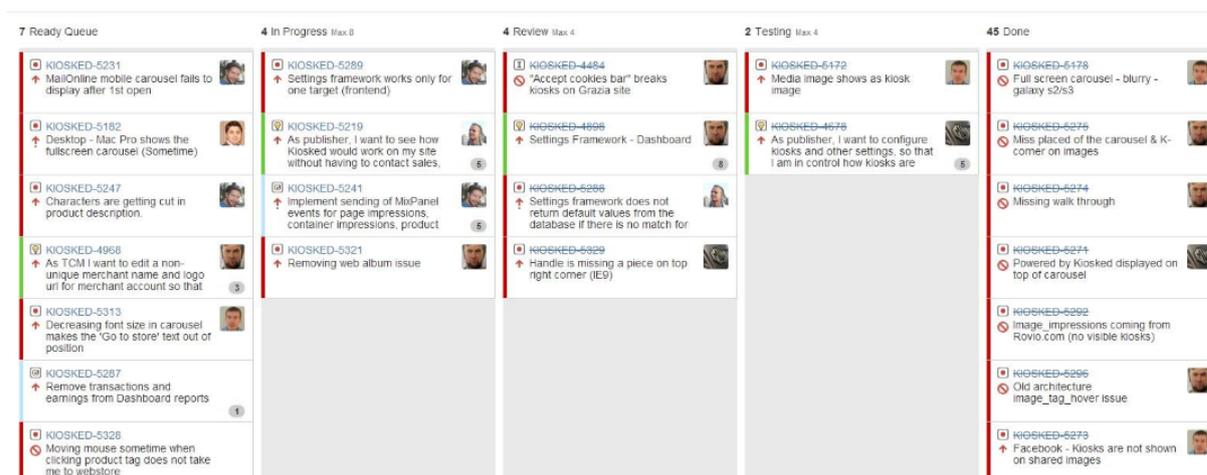
Fonte: (KHAN, 2014)

2.2.1 Princípios e Técnicas

Os princípios do *Scrumban* são adaptações de diretrizes oriundas dos *frameworks* *Scrum* e *Kanban* e suas combinações. De acordo com Khan (2014), existem sete princípios do *Scrumban*:

- **Visualizar o fluxo de trabalho:** No *Scrumban* a ideia é visualizar o fluxo de trabalho dentro e fora do *Sprint* (YERET, 2012). Uma vez que a visualização do *workflow* é alcançada, a inclusão dos *Product Owners* no time é facilitada, e eles, reconhecem as zonas de gargalos. Além disso, a visualização ajuda a conhecer quem está responsável por cada tarefa, e qual é o estado de progresso de diferentes histórias em um determinado ponto do tempo. A Figura 1 mostra um exemplo do quadro *Scrumban*

Figura 1 – Exemplo do Quadro *Scrumban*



Fonte: (KHAN, 2014)

- **Livre Movimentação de Tarefas:** No *Scrumban*, o trabalho é movimentado como e quando necessário para uma sessão. Porém, na transição de uma etapa para outra a tarefa só é movida para o próximo estágio se a equipe possuir a capacidade para a realiza.

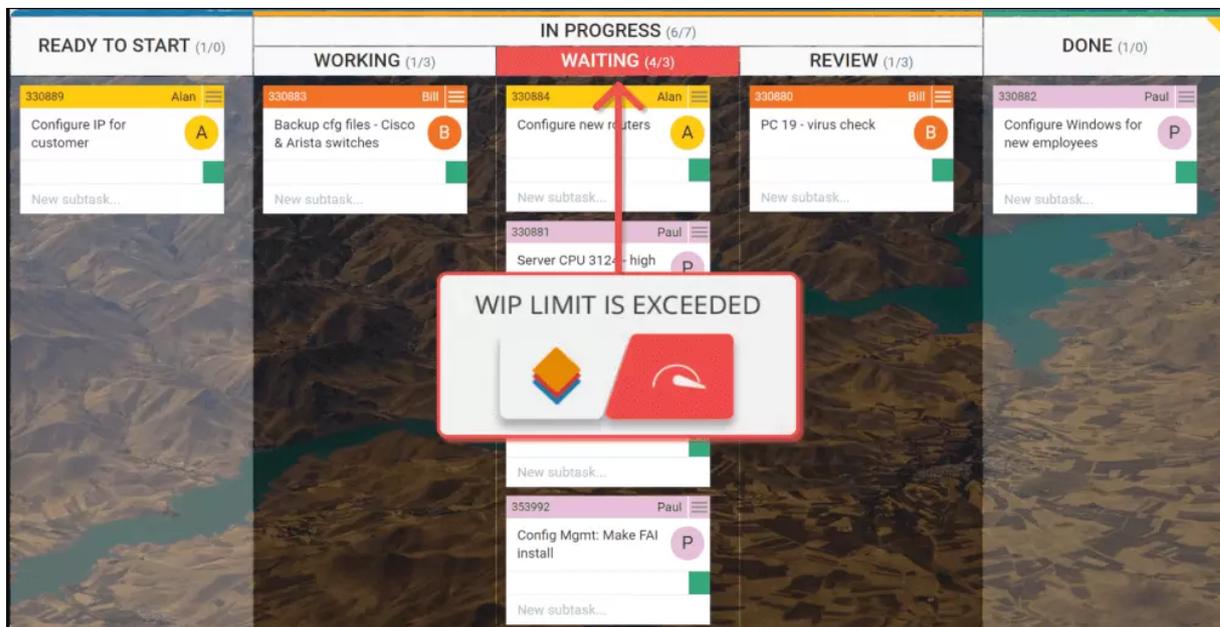
Neste contexto, Ladas (2008) propôs uma melhoria ao quadro de tarefas, o qual consiste na desvinculação do processo de **atribuição** de trabalho do processo de **priorização** de trabalho, e este aprimoramento seria adicionar uma sessão *Ready Queue* entre o *Backlog* e o *In Progress*, como observado na Figura 1.

- *Ready Queue*: Contém itens que estão pendentes do *backlog*, mas têm alta prioridade. Nesta fila, nenhuma tarefa é vinculada a nenhum indivíduo, mas assim que alguém estiver disponível, ele deve assumir uma dessas tarefas em vez de escolher algo do *backlog* geral

- **Limite de Trabalho em Andamento (*Work-in-Progress - WIP*):** WIP é a quantidade de tarefas na sessão *In Progress*, e um dos principais aspectos do *Scrumban* é impor um determinado limite de WIP a cada sprint. O objetivo é concentrar o foco da equipe em finalizar todo o trabalho atual ao invés de começar novas tarefas. Isso é benéfico para o time, pois agrega qualidades como:
 - Maior colaboração e evolução da equipe, uma vez que alguém finaliza uma tarefa ele deve auxiliar outro colega do time que ainda está com a tarefa em desenvolvimento;
 - Fluxo de trabalho mais agradável, pois o limite de WIP é pensado com base na capacidade da equipe e tende a fazer com que cada membro da equipe, não necessariamente todos, tenham uma tarefa por *Sprint*;
 - Entregas de qualidade, visto que a colaboração de todo o time em terminar as tarefas somente do *Sprint* atual promove um grau de detalhamento maior do colaborador, pois o mesmo não precisa ficar preocupado com as próximas etapas;

A Figura 2 demonstra uma situação de quando o limite de WIP é excedido

Figura 2 – Exemplo de alerta ao ultrapassar o limite de WIP



Fonte: (KHAN, 2014)

- **Políticas de Trabalho Explícitas:** Antes de começar o processo de desenvolvimento do projeto, a equipe deve definir regras, e políticas internas e externas ao time, como:
 - Definição do que é considerado "pronto";
 - Como proceder quando a equipe encontra um gargalo;
 - Regras de comunicação da equipe como ferramentas para comunicação, horários para o envio de mensagens e ligações, quem pode falar com quem, etc;
 - Políticas de boa convivência;
 - Definir como serão calculados a quantidade de WIPs por *Sprint*;

As políticas adotadas precisam ser continuamente lembradas no *Sprint Planning* e devem seguir durante todas as fases do projeto.

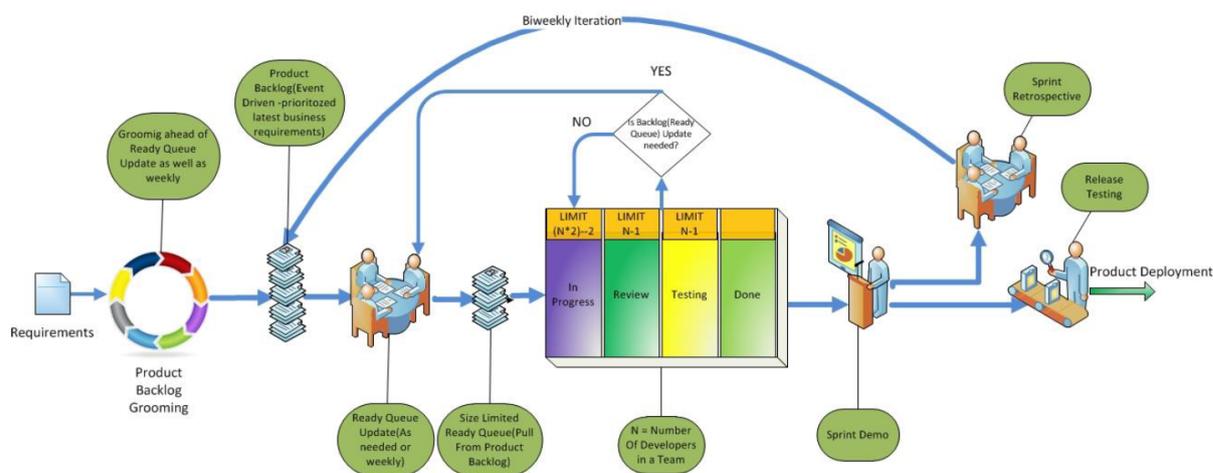
- **Reuniões de Planejamento Objetivas:** O *Scrumban* é uma estrutura que visa fornecer à equipe de desenvolvimento os melhores trabalho a serem feitos a seguir, não mais e não menos (LADAS, 2008). Portanto, não há necessidades de longas reuniões de planejamento, pois deve ser entregue à equipe de desenvolvimento todas as tarefas e com elas suas informações e insumos, evitando assim desnecessárias discussões sobre o que será feito.
- **Cerimônias do Scrum:** *Review*, *Retrospectives*, *Daily Stand-up meetings* são ritos advindos do *Scrum* que permanecem no *Scrumban*, pois são de suma importância para o desenvolvimento contínuo do projeto.
- **Estimação:** É um método que utiliza de métricas como o *velocity*, *story points*, quantidade de tarefas concluídas em um *Sprint* entre outros, para calcular a produtividade da equipe, embora ele seja uma forma do *Product Owner* visualizar o andamento do projeto e a contribuição de cada integrante o *Scrumban* o considera de caráter opcional, ficando a cargo dos devidos responsáveis se desejam ou não utilizá-lo.

2.2.2 Aplicação

A Figura 3 é uma proposta feita por Khan (2014) para a aplicação do *Scrumban* em uma determinada empresa, qual era o seu objeto de estudo.

Com a aplicação deste processo, certas dificuldades da equipe de desenvolvimento foram amenizadas, como: mudanças frequentes de prioridade, situações de pressão, falta de colaboração e comunicação da equipe, tarefas incompletas em sprints, constantes mudanças no processo, tais dificuldades levaram a atrasos na entrega de *software*, aumento dos custos e baixa motivação da equipe.

Figura 3 – Exemplo da aplicação do Scrumban



Fonte: (KHAN, 2014)

Abaixo são abordados os principais conceitos ilustrados na Figura 3:

1. Definir regras e políticas da equipe

- a) Plano de Comunicação: É necessário estabelecer quais ferramentas serão utilizadas para realizar contato, horários de disponibilidade para de cada membro da equipe e deve ser definido também quem tem acesso a quem e quais situações cada pessoa atende;
 - b) Regras de convivência: Geralmente estabelecidas pela seção de recursos humanos da empresa com base nos valores e políticas da empresa;
 - c) Solução de gargalos: Aplicação da técnica *Swarming* consiste na colaboração da equipe de desenvolvimento, no caso se um desenvolvedor está com um impedimento os outros colegas da equipe ao finalizarem suas tarefas devem ajudá-lo ao invés de começar uma próxima tarefa, exceto se houver alguma tarefa de alta prioridade na sessão *Ready Queue*.
 - d) Definição do que é Pronto: A equipe deve como determinar que uma tarefa está realizada, como, por exemplo, só poder ser considerada pronta após uma validação do PO, ou quando o cliente testa e retorna com um parecer positivo.
2. **Como será feito o cálculo de limite de WIP:** A equipe deve ter determinado antes de começar a sprint como será a política da quantidade de tarefas em progresso que pode ter simultaneamente.
 3. **Quais cerimônias do *Scrum* serão adotadas:** Deve definir como a equipe e os ritos ágeis serão adotados, e suas respectivas datas e horários.

4. **Construção do quadro Kanban:** Devem ser criadas sessões do *Kanban* com a fileira da *Ready Queue* e explicar para toda a equipe como ocorre todo o funcionamento do fluxo de trabalho.
5. **Levantamento dos requisitos:** Os requisitos devem ser levantados com o cliente para servir de insumo na construção das tarefas.
6. **Construção do *Backlog* pelo PO:** O *Product Owner*, ao receber os requisitos, deve construir as tarefas que irão compor o *Backlog* e suas respectivas prioridades.
7. **Ready Queue:** É uma sessão que fica entre o *Backlog* e o *In Progress*, nela está contida todas as tarefas de alta prioridade, depois do PO realizar a priorização ele precisa alocar na *Ready Queue* o que deverá ser feito durante o *Sprint*. Caso surja alguma demanda de alta urgência, cria-se uma tarefa nesta fileira e assim que um membro da equipe terminar o seu trabalho ele deve começá-la, invés de esperar a próxima *Sprint Planning* para receber uma nova atribuição.
8. ***Sprint Planning*:** Nesta cerimônia do *Scrum*, é distribuído entre os membros da equipe as tarefas da *Ready Queue* a serem realizadas durante o *Sprint*, respeitando o limite de WIP estabelecido e o princípio de reuniões pequenas onde o desenvolvedor recebe o que deve ser feito não mais e não menos (LADAS, 2008).
9. **Fechamento da *Sprint*:** O encerramento ocorre somente quando todas as tarefas estabelecidas para aquele *Sprint* estão completamente finalizadas com o critério de *Pronto*, anteriormente estabelecido, cumprido. Após esta etapa ocorre novamente o *Sprint Planning* e o começo de um novo *Sprint*.

2.3 *User-Centered Design* - UCD

O termo *User-Centered Design* - UCD foi popularizado pelo especialista em ciência cognitiva e engenharia de usabilidade, Don Norman, graças a trabalhos como *User Centered System Design: New Perspectives on Human-Computer Interaction* (NORMAN; DRAPER, 1986).

O UCD é uma coleção de processos iterativos que se concentram em colocar as necessidades do usuário no centro do *design* e em cada estágio de desenvolvimento, buscando obter uma compreensão profunda de quem usará o produto, entendendo os problemas enfrentados pelos usuários e a melhor forma que o *software* desenvolvido pode ajudá-lo.

O processo de *Design* Centrado no Usuário responde perguntas cruciais sobre aqueles que utilizam o produto. As seguintes perguntas são normalmente feitas durante o processo de UCD: *Quem usa seu produto? Quais são seus objetivos? O que os usuários*

estão procurando? Em que eles estão interessados? Como seus usuários veem o processo de conclusão de uma tarefa? E muitas outras (NOVOSELTSEVA, 2017).

O UCD possibilita o desenvolvimento a partir da perspectiva de como ele será compreendido pelo usuário, ao contrário de fazer com que os usuários adéquem seus comportamentos para usar um produto. No UCD, o desenvolvimento prossegue com o usuário dentro do foco (RUBIN; WILEY; SONS, 1984).

A ISO 13407 (1999) descreve que o *Design* Centrado no Usuário possui alguns princípios fundamentais que podem ser implementados ao processo de *design*:

- **Usuários envolvidos em todo processo de *design*:** A experiência final do usuário implica diretamente em decisões críticas de *design*.
- **Esclarecimento de requisitos:** Os requisitos de negócios são fundamentados nas necessidades do usuário.
- **Ciclo de *feedback* do usuário com ciclo de vida do produto:** O *feedback* dos usuários é analisado regularmente e auxilia a equipe de produto a tomar decisões mais focadas na experiência e necessidades do usuário.
- **Processo de *design* iterativo:** A equipe de produto visa melhorar constantemente a experiência do usuário; introduz mudanças gradativamente à medida que obtém mais compreensão sobre seu público-alvo.

Esses princípios possibilitam que o UCD seja uma metodologia empática, colaborativa, otimista e experimental, pois foca nas necessidades do usuário, envolvendo o consumidor em todo processo de desenvolvimento e auxiliando uma mudança valorizada pelo aprendizado, na prática.

2.3.1 Etapas UCD

O UCD possui uma série de técnicas que podem ser incorporadas ao processo iterativo, visando posicionar o usuário ao centro das decisões de *design*, e podem ser divididas em quatro etapas (ARGUMANIS; MOQUILLAZA; PAZ, 2021) (NORMAN, 2013):

- **Observação:** Nesta a etapa busca-se compreender a natureza do problema por meio da identificação do público alvo. Por meio de uma série de atividades fundamentadas em técnicas de levantamento qualitativo e quantitativo de dados, o pesquisador irá observar os potenciais clientes visando obter informações suficientes para entender seus interesses, necessidades, valores, motivações e dificuldades. Abaixo são citadas algumas técnicas de elicitación de requisitos.

- Investigação Contextual: É um estudo de como os usuários interagem com determinado equipamento ou ambiente que estão inseridos, visando garantir que as tecnologias incorporadas atendam suas necessidades. Nessa técnica, o pesquisador coleta e analisa dados substanciais em conjuntura com análises em tempo real, em substituição de entrevistas e análises isoladas.
- Grupo Focal: Este método é uma forma de pesquisa qualitativa baseada em reunir participantes em uma entrevista, a fim de coletar opiniões acerca de produtos ou serviços. O Grupo Focal deve encorajar os participantes a compartilhar pensamentos, sentimentos, atitudes e ideias sobre um determinado assunto. A organização de grupos focais possibilita obter adesão para um projeto e pode ser usado como entrada para a etapa de *design*.
- Personas: A técnica de Personas possibilita a representação do cliente ideal para um negócio por meio de um subconjunto de usuários fictícios. É fundamentada em dados realísticos sobre comportamento, e características demográficas e sociais. Na Engenharia de *Software*, essa técnica possibilita um melhor entendimento sobre as necessidades, experiências, comportamentos e objetivos dos usuários em uma ferramenta ou aplicação. É descrito na literatura que Personas é uma técnica útil para manter os desenvolvedores focados nos usuários e em suas necessidades (NIELSEN, 2004) (COOPER; REIMAN, 2003).

Ao final dessa etapa espera-se um bom entendimento sobre o problema a ser tratado.

- **Design(Ideação):** Também nominada de ideação, ou em português, projeto. Posteriormente à determinação dos requisitos, durante a fase atual, a busca e produção de potenciais soluções são o foco principal. É nesta etapa que os *designers* identificam quaisquer requisitos de negócios ou metas de usuário que devem ser atendidas para que o produto seja bem-sucedido (GUIMARÃES, 2019). Neste contexto, abaixo são listadas algumas técnicas para a fase de *design*.
 - *Brainstorming*: É uma técnica que, por meio do compartilhamento espontâneo e criativo de ideias, possibilita encontrar soluções para um problema. É recomendado que seja realizada em uma reunião, com pessoas ativas na empresa e com perspectivas diferentes. O *brainstorming* deve ser livre de críticas, pois mesmo as ideias que parecem ineficazes devem ser analisadas, afinal, elas podem ser o ponto inicial para construção de discussões mais profundas.
 - *Card Sorting*, ou Classificação de Cartas: No campo da Engenharia de Requisitos, a classificação de cartas é descrita como o método mais eficaz para elicitar domínios de problemas (NURMULIANI; ZOWGHI; WILLIAMS, 2004). Este método é usado para sugerir comportamentos intuitivos por meio da elicitação

de requisitos, é regularmente usado para registrar dados sobre diferentes formas de representar o conhecimento de domínio. Desta maneira, a classificação de cartas ajuda a entender as expectativas dos usuários e a compreensão do ambiente que estão inseridos, oferecendo mais informações sobre o público alvo, distinguindo complicações de baixo e alto nível, por um custo nominal, flexível e de fácil manuseio pelo pesquisador. É uma fonte valiosa de informações para priorizar solicitações de mudança e avaliar seu impacto (NURMULIANI; ZOWGHI; WILLIAMS, 2004).

- *Customer journey map* - CJM, ou Mapeamento da jornada do cliente - O Mapeamento da Jornada do Cliente (CJM) é uma ferramenta de gerenciamento estratégico cada vez mais popularizada por acadêmicos e profissionais por sua utilidade na compreensão de experiências do cliente (ROSENBAUM; OTÁLORA; RAMIREZ, 2016). O mapeamento da jornada do cliente é um diagrama que descreve os estágios pelos quais os clientes passam ao interagir com um produto de software, desde etapas básicas até processos complexos. O CJM visa listar todos os possíveis casos de uso que os clientes possam realizar durante a utilização de algum serviço. Para cumprir esses requisitos, os mapas de jornada devem estar fundamentados nos dados coletados na etapa de pesquisa. Ao compreender claramente os pontos de contato com o cliente, a gerência pode trabalhar com membros da equipe multifuncional para empregar táticas que promovam a inovação do serviço. O objetivo dessas táticas é aprimorar as interações do provedor de atendimento ao cliente, melhorando a experiência do cliente associada a cada ponto de contato (ROSENBAUM; OTÁLORA; RAMIREZ, 2016).
- *Little Design Up-Front (LDUF)* ou *Sprint 0*: É usada para criar a estrutura básica de um projeto para que os *sprints* futuros possam ser realmente incrementados (ALMUGHARAM; ALYAHYA, 2017). O LDUF possibilita que os *UX-Designers* estejam uma ou duas sprints a frente, projetando uma estrutura suficientemente flexível ao time de desenvolvedores *front-end*, antes mesmo deles implementarem suas linhas de código.

Independente do método utilizado, Norman (2013) sugere a possibilidade de aplicação de duas regras principais acompanhadas por uma terceira complementar.

- Gerar ideias: A geração de várias ideias é incentivada, pois possibilita um olhar mais amplo ao processo e produto desenvolvido, evitando o enraizamento prematuro em pensamentos iniciais.
- Analisar ideias: As ideias geradas devem ser analisadas com olhar crítico, mas não excludente. A etapa anterior pode ocasionar o surgimento de ideias utópicas ou que não se aplicam ao cenário existente, porém devem ser analisadas

pacientemente, pois podem conter percepções inovadoras e criativas que posteriormente podem ser extraídas e bem utilizadas na seleção final.

- Questionar: Questionar ideias levantadas possibilita um olhar mais amplo relativamente à solução proposta. Mesmo a pergunta mais simples, ou até categorizada como óbvia, pode conter questionamentos válidos que após uma cautelosa análise, pode ter instruções para resolução de um determinado problema.
- **Prototipagem:** O UCD é tendencioso para a prototipagem. Esta etapa do processo possibilita que a equipe de *design* implemente ideias coletadas em fases anteriores por meio de protótipos, ocasionando um melhor entendimento das necessidades do usuário e garantindo que as lacunas levantadas foram compreendidas. Esta parte do processo pode ser feita em etapas, construindo a partir de um conceito geral para um design completo (GUIMARÃES, 2019). Abaixo são citadas algumas técnicas de prototipagem.
 - Design em pares: Como o nome já sugere, o *pair Design* - ou *Design Pairing* é um método em que dois *designers* trabalham juntos no mesmo problema. Esta técnica colaborativa pode ser aplicada em diversos estágios do desenvolvimento e possibilita a análise de diferentes perspectivas, ocasionando em soluções concretas de design. A técnica mais comum do *Design* em pares possui os papéis de *Observador* e *Sintetizador*. O papel do Gerador é apresentar ideias para resolver um problema pré-definido, enquanto o Sintetizador complementa este trabalho por meio da análise e testes, tendo sempre em mente o objetivo e uma visão clara do resultado. O objetivo deste processo é dedicar-se a uma ideia que possa resolver um obstáculo de design enquanto atende aos objetivos de negócio. É um trabalho intenso que exige foco, colaboração e harmonia de ambas as funções.
 - Design Participativo: Este método é baseado em incluir os usuários ao processo de design, por meio de coleta de opiniões e até, envolvimento ativo na prototipação. A aplicação do *Design* Participativo é comumente em pequenos projetos que possibilitam a criação de protótipos que podem ser incrementados ao processo de design de um projeto maior. É normalmente usado juntamente a outros métodos de UCD (BAXEVANIS, 2006).
 - Protótipo de Papel: A prototipagem de papel é uma técnica que possibilita a criação e testes de interfaces de usuário de maneira rápida e barata. O principal foco desta técnica é mapear a estrutura de informações e o fluxo de interação dos usuários, testando a experiência do usuário em alto nível, e auxiliando a identificar possíveis problemas de usabilidade, evitando custo de alteração após a implementação. O protótipo de papel pode ser fundamentado em baixa fidelidade, *wireframes*, ou alta fidelidade, *mockups* ou *layouts*, e conforme mais

próximos forem da aparência final, maior qualidade terão os dados obtidos nos testes.

- **Testes:** Os protótipos criados devem ser testados e avaliados. Esta etapa visa realizar uma avaliação integral do produto por meio de testes de usabilidade com grupos de usuários reais. Fundamentado em técnicas de avaliação heurística, enquetes, observação, casos de uso, entre outras, essa etapa coloca os usuários em confronto com as interfaces desenvolvidas. Os testes possibilitam entender quais mudanças devem ser realizadas para melhorar a experiência do usuário, a fim de criar um produto final com a menor quantidade de imperfeições possíveis. À medida que um grupo maior de usuários começa a usar o modelo, necessidades, desafios novos ou imprevistos podem surgir e outras mudanças ou ajustes devem ser feitos. Algumas técnicas de testes são citados abaixo.
 - Avaliação heurística: A avaliação heurística é uma técnica de inspeção de usabilidade executada por examinadores que seguem um conjunto de princípios de usabilidade, as heurísticas, e avaliam todos os elementos de interface com o usuário, visando encontrar falhas de usabilidade (NIELSEN; MOLICH, 1990). O foco das heurísticas é realizar uma análise da qualidade da interface em relação à sua usabilidade, e assim, identificar precocemente problemas desta.
 - *Thinking aloud*, ou Pensar em voz alta: Este método está fundamentado em estimular um usuário a falar em voz alta durante a solução de algum problema ou execução de uma tarefa, possibilitando uma maior compreensão sobre sua experiência. É uma alternativa barata, robusta, flexível, convincente e com fácil aprendizado, possibilitando várias vantagens. Sendo uma das mais relevantes, a possibilidade de servir como uma janela para a alma, permitindo que você descubra o que os usuários realmente pensam sobre seu *design* (NILSEN, 2012).
 - Testes de Usabilidade: Este método é comumente usado para coletar dados enquanto usuários reais usam uma aplicação ou produto. Os usuários podem ser solicitados a seguir um protocolo e expor o que estão fazendo e por qual motivo. Normalmente, é necessário o tempo de dois especialistas por sessão – um para moderar, outro para anotar problemas, elevando o custo de aplicação desde método (BAXEVANIS, 2006).

A Tabela 2 sintetiza a aplicação dessas técnicas, abordando os custos necessários, os tipos de dados obtidos, podendo ser estatísticos ou não, o tamanho da amostra, além da etapa de aplicação.

Tabela 2 – Análise comparativa das técnicas (Adaptada)

	Custo	Saída	Tamanho da Amostra	Etapa	Autor(es)
Investigação Contextual	Mediano	-	-	Observação	(ARGUMANIS; MOQUILLAZA; PAZ, 2021)
Grupo Focal	Baixo	Não estatístico	Baixo	Observação	(BAXEVANIS, 2006)
Personas	Baixo	-	-	Observação	(ARGUMANIS; MOQUILLAZA; PAZ, 2021)
<i>Brainstorming</i>	Baixo	Não estatístico	Baixo	<i>Design</i> (Ideação)	(NORMAN, 2013)
Classificação de Cartas	Alto	Estatístico	Alto	<i>Design</i> (Ideação)	(BAXEVANIS, 2006)
Mapeamento da jornada do cliente	Mediano	-	-	<i>Design</i> (Ideação)	(ARGUMANIS; MOQUILLAZA; PAZ, 2021)
LDUF (ou Sprint 0)	Baixo	-	-	<i>Design</i> (Ideação)	(ARGUMANIS; MOQUILLAZA; PAZ, 2021)
Design em pares	Baixo	-	-	Prototipagem	(ARGUMANIS; MOQUILLAZA; PAZ, 2021)
Design participativo	Baixo	Estatístico	Baixo	Prototipagem	(BAXEVANIS, 2006)
Protótipo de papel	Baixo	-	-	Prototipagem	(ARGUMANIS; MOQUILLAZA; PAZ, 2021)
Avaliação Heurística	Alto	-	-	Testes	(ARGUMANIS; MOQUILLAZA; PAZ, 2021)
<i>Thinking Aloud</i>	Baixo	Não Estatístico	Baixo	Testes	(NILSEN, 2012)
Testes de Usabilidades	Alto	Estatísticos e não Estatísticos	Baixo	Prototipagem e Testes	(BAXEVANIS, 2006)

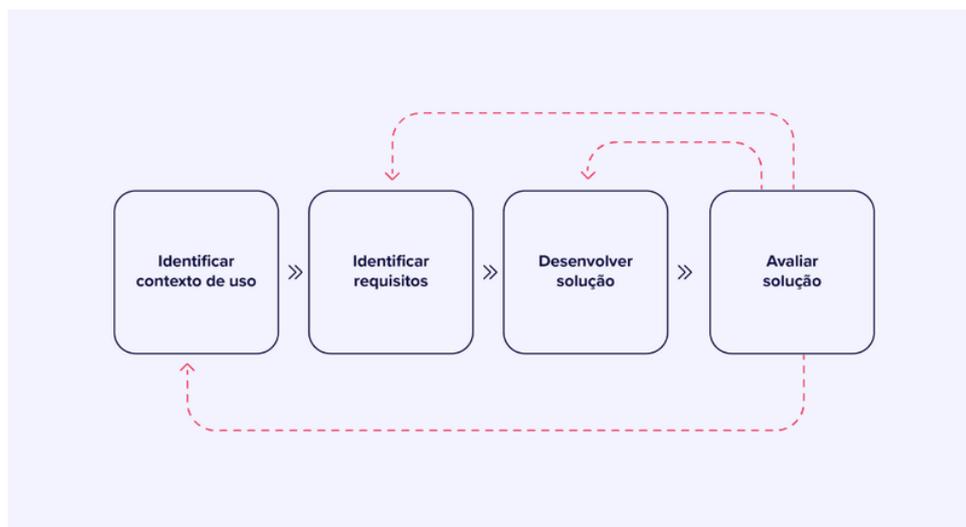
Fonte: (BAXEVANIS, 2006).

O UCD é um processo iterativo e incremental, a literatura sugere que após a finalização da etapa de testes é completada uma iteração e retorna-se à etapa de pesquisa. Nesse contexto, é possibilitado um refinamento e melhoria contínua do produto em seu

desenvolvimento. Em iterações futuras as etapas de prototipação e testes também podem ser aplicadas na fase de pesquisa, possibilitando maior certeza de que o problema enfrentado foi resolvido. As iterações devem ser realizadas até que os resultados sejam satisfatórios (FUNDATION, 2019).

A Figura 4 sintetiza o processo UCD iterativo abordado por (NORMAN, 2013).

Figura 4 – Etapas do UCD



Fonte: (FUNDATION, 2019)

2.3.2 UCD e *User Experience* - UX

User Experience - UX, ou Experiência do Usuário é um campo de estudo voltado a compreender a relação entre usuário e um produto ou serviço. O UX contempla todas as etapas do processo que o usuário está inserido, desde o interesse, as pesquisas e a compra de um determinado produto. Neste contexto, o principal percurso desta forma de pensamento é o *UX Design*. Este profissional deve estar atento com cada etapa a qual o usuário irá interagir com o produto ou serviço; e proporcionar que essa interação ocorra da maneira mais satisfatória possível.

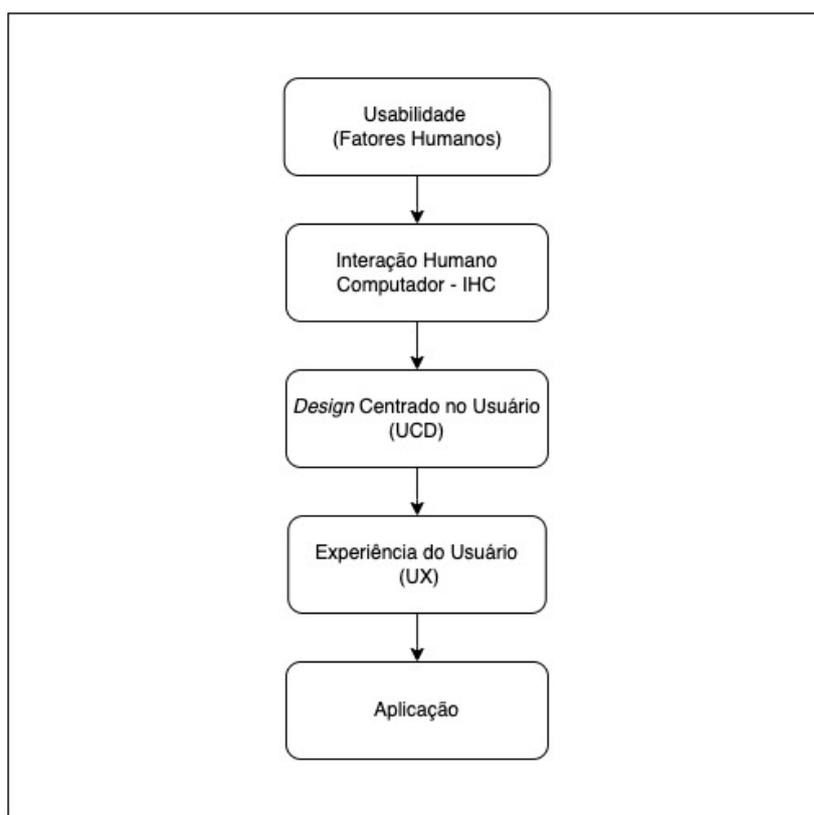
Pode parecer que *User Experience* - UX e *User-Centered Design* - UCD são termos muito semelhantes, quase equivalentes (HASSENZAHN, 2006) (MARCUS; GUNTHER; SIEFFERT, 2009). Ambas as abordagens são fundamentadas na identificação da prática de envolver os usuários finais em todo o processo de *design*, e não apenas nos estágios iniciais (VENTURI; TROOST; JOKELA, 2006). A Experiência do Usuário é processo para alcançar objetivos pragmáticos e hedônicos, empregando o *Design* Centrado no Usuário *User Centered Design* - UCD, na prática (HUSSEIN; MAHMUD; TAP, 2019).

Na Engenharia de *Software*, um produto visualmente atraente não é sinônimo de boa experiência para os usuários, se sua navegação não for intuitiva e oferecer uma boa

navegação, o desempenho geral não será bom. A UX possibilita a projeção de produtos digitais da forma que os usuários considerem útil, intuitivo e agradável de interagir. Trata-se de garantir que eles encontrem valor nisso e possam encontrar o que procuram sem esforço.

A Figura 5 simplifica a relação entre a metodologias UCD e a abordagem UX, abordando desde estudos iniciais de fatores humanos, até a aplicação.

Figura 5 – Relação entre UCD e UX (Adaptada)



Fonte: (LOWDERMILK, 2013)

Segundo os autores, o *design UX* está contido no escopo do *design UCD*. O UCD não é apenas sobre UX e *User Interface - UI*, mas sim uma metodologia para envolver o usuário em cada fase do processo de *design*. A UX é um dos muitos focos da UCD, incluindo toda a experiência do usuário com o produto, compondo reações físicas e emocionais (LOWDERMILK, 2013).

O UCD visa incorporar os usuários nos estágios iniciais de *design*. Ele explora quem usará o sistema e o *design* com base na idade, cultura e localização geográfica do usuário. De acordo com *designers* de produtos experientes, uma boa coleta de dados realizada na etapa de Observação proposta por Norman (2013) orienta as decisões de *design*, e possibilita economizar muito tempo e dinheiro a longo prazo (GOEL; TANWAR; SHARMA, 2022).

Após a obtenção desses dados, na etapa de *Design*, o UX pode ser incorporado ao UCD por meio da adição de um profissional de UX. Durante esta fase, os *designers* de UX e as partes interessadas fazem um *brainstorm* no mais alto nível sobre o produto (essencialmente, o conceito do produto) (GOEL; TANWAR; SHARMA, 2022).

As demais atividades de *design*, prototipagem e testes, incorporadas com a presença de um profissional de UX, devem incluir os usuários em atividades que busquem compreender e coletar dados acerca das suas experiências, dificuldades e expectativas com o produto desenvolvido, de modo a gerar conhecimento suficiente para confeccionar um produto que atenda ou supere as expectativas dos usuários. A abordagem UCD deve ser conduzida pelos *UX-designers* com sua perspectiva de usabilidade (BLUESTONE, 2015)

2.3.3 UCD e Agile

Os processos ágeis de *software* compõem um desenvolvimento iterativo e incremental, onde os requisitos são mutáveis conforme as necessidades do cliente (SHARMA; SARKAR; GUPTA, 2012). Estes métodos visam produzir *software* de alta qualidade por meio do uso de comunicação frequente entre os membros da equipe, entrega contínua de *software* funcional ao cliente e resposta rápida às rápidas mudanças solicitadas pelos *stakeholders* (ALYAHYA; IVINS; GRAY, 2012).

Embora a abordagem ágil seja iterativa e se concentre em usuários reais durante o seu desenvolvimento, ainda carece da percepção da importância da usabilidade e das interfaces de usuário, pecando na criação de uma experiência atrativa para os usuários finais.

O processo ágil e a abordagem de *Design Centrado no Usuário* (UCD) são dois campos com grandes oportunidades benéficas quando integrados. Ambos são baseados em *design* mínimo inicial e *feedback* frequente dos usuários, e ambos são centrados no ser humano. No entanto, uma das principais preocupações em integrá-los é como coordenar o trabalho adequadamente, especialmente em projetos distribuídos (ALMUGHRAM; ALYAHYA, 2017).

Metodologias ágeis e o processo UCD compartilham o foco no cliente e no usuário, encorajando a integração entre eles (ALMUGHRAM; ALYAHYA, 2017). A lacuna para integração dessas duas tecnologias é vista na abordagem de levantamento de requisitos. O ágil implementa testes de aceitação somente após o desenvolvimento dos recursos no final da iteração (ciclo de lançamento). O UCD já incorpora os requisitos nas fases iniciais, incluindo o teste de usabilidade antes do desenvolvimento do recurso.

Para obter benefícios de ambas as metodologias, Silva et al. (2012) e Beyer, Holtzblatt e Baker (2004) aconselham enfatizar os requisitos previamente reunidos na compreensão do problema e, em seguida, usar as diretrizes do UCD para os recursos que

estão sendo desenvolvidos. A seguir [Argumanis, Moquillaza e Paz \(2021\)](#) e [Almughram e Alyahya \(2017\)](#) listam metodologias usadas para integrar UCD no processo de desenvolvimento de projetos ágeis:

- **Design em paralelo ao Sprint:** A especialista em UX [Sy e Miller \(2008\)](#) enfatiza que *UX-Designers* e desenvolvedores devem colaborar para criar uma UX eficaz, evidenciando atividades com foco em UX em todo o projeto. Esta técnica possibilita que o time de desenvolvimento e o time de *design* trabalhem em duas trilhas paralelas, iterando o *design* e a implementação separados, mas de maneira simultânea. A autora [Sy \(2007\)](#) afirma que, "Um princípio chave do processo UCD da Equipe de Experiência do Usuário é a iteração do design; precisamos conseguir detectar falhas de *design* com antecedência, alterar os designs quantas vezes forem necessárias e, em seguida, incorporar as correções de *design*." Os autores [Almughram e Alyahya \(2017\)](#) abordam que o *design* paralelo pode ser implementado por meio de três abordagens.
 - Artefatos de requisitos amigáveis ao UX: Em um *backlog*, os artefatos de UX (como wireframes, maquetes, protótipos etc.) serão incluídos como histórias de usuários. Os requisitos de experiência do usuário, de acordo com [Cline \(2015\)](#), são histórias de usuários detalhadas ou casos de uso associados a uma imagem, mapa de navegação ou qualquer outro documento que descreva a interação do usuário. Estes artefatos devem ter objetividade e clareza, pois fazem parte de um requisito funcional, usado como uma ponte entre o projeto e os requisitos.
 - *Just in Time (JIT)*: Esta técnica ajuda no gerenciamento de processos por meio da limitação do escopo, produzindo a quantidade exata para atender a demanda. De acordo com [Argumanis, Moquillaza e Paz \(2021\)](#), na engenharia de *software*, com essa abordagem, a equipe identifica um problema que precisa ser resolvido, alguns membros serão responsáveis por explorar o novo problema e o restante continuará com as demandas anteriores.
 - Pico de Design, (ou *Design Spike*): De acordo com [Ogle \(2014\)](#), na área de desenvolvimento, um "pico" é um termo para um exercício rápido que permite explorar soluções sem o fardo de escrever um bom código e, em seguida, jogar tudo fora, para ser posteriormente realizado da maneira certa com confiança. Na UX, os "picos" são os períodos usados para exploração, pesquisa e, às vezes, prototipagem, ajudando equipes de *design* a levantar questões complexas em um processo de desenvolvimento ágil ([ALMUGHGRAM; ALYAHYA, 2017](#)).
- **Design dentro do Sprint:** Para [Argumanis, Moquillaza e Paz \(2021\)](#), o *Design* dentro do Sprint surge para solucionar problemas de comunicação ocasionados pela divisão de times no *Design* paralelo ao Sprint. Essa metodologia propõe que o time

de *design* trabalhe junto ao time de desenvolvimento, evitando problemas de comunicação.

- **Lean UX:** O *Lean UX* propõe um modo de trabalho com intuito de tornar os processos mais eficientes e ágeis, diminuindo riscos de desperdícios de recursos e tempo, por meio do lançamento de MVP's. De acordo com [Argumanis, Moquillaza e Paz \(2021\)](#) o *Lean UX* possibilita que os protótipos sejam testados pelos usuários sem sua fase inicial, permitindo rapidamente as alterações pelo time de *design*, reduzindo desperdício de recursos.
- **Design Thinking (DT):** De acordo com [Fundation \(2023\)](#), o *Design Thinking* é um processo iterativo e não linear, que as equipes usam para entender os usuários, desafiar suposições, redefinir problemas e criar soluções inovadoras para prototipação e testes. O *Design Thinking* é um método para desenvolver produtos e serviços com foco nas necessidades, benefícios e limitações do usuário. O principal objetivo é transformar dificuldades e limitações em benefícios para o cliente, e valor comercial para o seu negócio.

A Tabela 3 sintetiza o pensamento dos autores sobre a aplicação dessas metodologias, abordando as vantagens, desvantagens, e quando usar ou não.

Tabela 3 – Análise comparativa das metodologias (Adaptada)

	<i>Desing</i> em paralelo ao Sprint	<i>Desing</i> dentro da Sprint	<i>Lean UX</i>	<i>Design Thinking (DT)</i>
Vantagem	Facilita o planejamento do projeto	<i>Design</i> é validado tanto pelo time de desenvolvimento quanto pelo time de <i>design</i>	Diminui o desperdício de recursos	Maior entendimento das necessidades dos usuários e clientes
Desvantagem	Dificulta a comunicação	Dificulta a sincronização de atividades	Alto custo	Viável somente para inovação
Quando usar	Grandes projetos	Projetos pequenos	Apenas projetos novos	Projetos inovadores
Quando não usar	Pequenos projetos	Projetos grandes	Redesenho ou melhoria de projeto	Redesenho ou melhoria de projeto

Fonte: ([ARGUMANIS; MOQUILLAZA; PAZ, 2021](#)).

O *Design* em paralelo é uma técnica que facilita o planejamento das iterações, facilitando suas implementações em grandes projetos. Todavia, precisam estar fortemente alinhadas aos princípios e cerimônias do *Scrum* para mitigar a dificuldade de comunicação devido ao paralelismo de equipes.

A técnica de *Design* dentro do *Sprint* soluciona o problema de comunicação entre as equipes, onde designs e desenvolvedores trabalham simultaneamente dentro da mesma iteração, porém não é recomendada para projetos grandes, devido à dificuldade de sincronizar o maior número de atividades que acontecem em simultâneo.

O *Lean UX* possui um alto custo e sua implementação requer iterações maiores, devido à possibilidade de validação de MVP's com clientes e usuários. O risco de desperdícios é menor, porém é necessário avaliar os cronogramas e orçamentos que a empresa está suscetível para sua melhor adaptação.

Já o *Design Thinking - DT*, uma das metodologias mais conhecidas e utilizadas atualmente, fornece um maior entendimento das necessidades do cliente por aplicar diretamente diretrizes de UCD, buscando compreender problemas complexos de maneira centrada no usuário. Porém, essa metodologia é econômica e tecnicamente viável, somente para projetos inovadores, não sendo recomendada para melhoria de projetos.

2.3.4 Como um Design Centrado na Experiência do Usuário afeta positivamente os negócios?

Em um relatório de 2016, publicado na revista *Forrester Research* e intitulado de "*The Six Steps for Justifying Better UX*", os autores [Hogan et al. \(2016\)](#) verificaram que as empresas que investem em UX possuem menores custos de aquisição de clientes, custos de manutenção mais baixos, melhor captação de clientes e maior participação de mercado do que aquelas que não investem em UX.

O fato é que, quando um produto de software é intuitivo, fácil de usar, fácil de navegar e fornece um processo de *checkout* impecável, toda a jornada de satisfação-conversão-retenção do cliente é suportada, aumentando o tempo de permanência dos usuários, que consequentemente irá aumentar o número de vendas e a receita.

O retorno sobre o investimento (*Return Over Investment - ROI*) trata-se de um indicador gerencial de desempenho usado para avaliar a eficiência ou lucratividade de um investimento. O ROI tenta medir diretamente a quantidade de retorno de um determinado investimento, em relação ao seu custo.

Ainda em seu relatório, a *Forrester* aborda que um UX bem projetado pode resultar em um incrível aumento de 400% na taxa de conversão de um site, gerando um ROI significativo para a empresa: a *Forrester* afirma que cada dólar gasto em *design* UX retorna US\$ 100.

2.4 Scrum-UCD

O *framework Scrum-UCD* visa criar uma estrutura que combina as práticas e princípios da metodologia ágil *Scrum*, com diretrizes do Design Centrado no Usuário (*User Centered Design - UCD*). O objetivo é implementar uma metodologia que seja ágil enquanto esteja focada no usuário, possibilitando que o software seja projetado conforme as necessidades e expectativas dos usuários.

O *Scrum-UCD* permite que as equipes trabalhem de forma colaborativa e ágil, envolvendo os usuários desde o início do processo de design até a entrega final do software. Isso ajuda a garantir que o software seja desenvolvido conforme as necessidades reais dos usuários, resultando em soluções mais eficientes e satisfatórias.

2.4.1 Análise comparativa entre as técnicas UCD

De acordo com [Argumanis, Moquillaza e Paz \(2021\)](#), as quatro etapas propostas por [Norman \(2013\)](#) ao UCD, ao serem incorporadas ao *framework* ágil *Scrum*, podem ser sintetizadas em 3 etapas: Inicial, Planejamento e Implementação.

A etapa Inicial, proposta por [Argumanis, Moquillaza e Paz \(2021\)](#), está alinhada à etapa de Observação, abordando métodos que busquem compreender a natureza do problema, por meio da identificação do público alvo. O autor propõe que nesta etapa também ocorra um entendimento mais profundo sobre o produto desenvolvido, por meio do aprimoramento do documento de visão que deve ser realizado pelo *Product Owner*.

A etapa de Planejamento visa gerar um melhor entendimento do produto por meio do planejamento do projeto a partir da definição de detalhes do fluxo de trabalho, além de gerar uma visão de *design* inicial do produto por meio da criação de protótipos de baixa fidelidade. Esta fase aborda a etapa de *Design* (Ideação), além da prototipagem de baixo nível abordada na etapa de Prototipagem.

A etapa de Implementação visa aperfeiçoar os protótipos desenvolvidos na etapa de Planejamento, por meio da validação com o usuário. Esta etapa seria equivalente à junção do *design* de alto nível da etapa de Prototipagem, junto a etapa de Testes.

Para que a integração do UCD ao *Scrum* fosse possível, [Argumanis, Moquillaza e Paz \(2021\)](#) sintetizam a aplicação das técnicas levantadas na Tabela 2, abordando vantagem, desvantagem, onde usar e onde não usar.

A Tabela 4 é uma adaptação da tabela proposta por [Argumanis, Moquillaza e Paz \(2021\)](#) e sintetiza comparativamente os métodos propostos pelos autores para a etapa Inicial.

Tabela 4 – Análise comparativa das técnicas UCD para Etapa Inicial (Adaptada)

Método	Investigação Contextual	Grupo Focal	Personas
Vantagem	Melhora o entendimento sobre o contexto dos usuários	Possibilita reunir informações detalhadas sobre um domínio específico	Possibilita um bom entendimento sobre as necessidades dos usuários
Desvantagem	Dependência de terceiros	Participantes podem não expressar honestamente suas opiniões sobre os assunto	Não é aplicável em todos os tipos de projetos
Custo	Mediano	Baixo	Baixo
Estágio do projeto	Inicial	Inicial	Inicial
Onde usar	Qualquer projeto	Qualquer projeto	Projetos com usuários desconhecidos
Onde não usar	-	-	Projeto com usuários conhecidos

Fonte: ([ARGUMANIS; MOQUILLAZA; PAZ, 2021](#)) e ([BAXEVANIS, 2006](#)).

As Tabelas 5 e 6 são uma adaptação da tabela proposta por [Argumanis, Moquillaza e Paz \(2021\)](#) e sintetizam comparativamente os métodos propostos pelos autores para a etapa de Planejamento.

Tabela 5 – Análise comparativa das técnicas UCD para Etapa de Planejamento (Adaptada)

Método	<i>Brainstorming</i>	Classificação de Cartas	<i>Mapeamento da jornada do cliente</i>	Sprint 0
Vantagem	Possibilita encontrar soluções por meio do compartilhamento espontâneo de ideias	Fornecer conhecimento sobre o modo de pensar dos usuários	Fornecer conhecimento sobre o processo dos usuários para realizarem tarefas	Possibilita um bom espaço de tempo para ocorrer a investigação do usuário e a definição da visão do produto
Desvantagem	Desentendimentos surgem com mais frequência e algumas pessoas ficam desanimadas e acabam participando menos	É difícil realizar a classificação de muitos cartões	Não fornece informações sobre funcionalidades	Desnecessária se as decisões não forem implementadas
Custo	Baixo	Alto	Mediano	Mediano
Estágio do projeto	Planejamento	Planejamento	Planejamento	Planejamento
Onde usar	Qualquer projeto	Projetos com muitas informações estruturadas	Projetos com baixa visão inicial do produto	Qualquer projeto
Onde não usar	-	Projetos com pouca informação estruturada	Projetos com alta visão inicial do produto	-

Fonte: (ARGUMANIS; MOQUILLAZA; PAZ, 2021) e (BAXEVANIS, 2006).

Tabela 6 – Análise comparativa das técnicas UCD para Etapa de Planejamento (Adaptada)

Método	Protótipo de papel	<i>Design</i> em pares	<i>Design</i> participativo
Vantagem	Fornece uma visão geral do <i>design</i> do produto de maneira rápida e econômica	Fornece aos <i>designers</i> uma ideia clara sobre as restrições técnicas	Fornece uma visão geral sobre as necessidades e desejos dos usuários
Desvantagem	<i>Designers</i> são abstratos	Baixa eficiência em pequenas tarefas	Necessita da participação de terceiros
Custo	Baixo	Baixo	Baixo
Estágio do projeto	Planejamento	Planejamento	Planejamento
Onde usar	Qualquer projeto	Projetos complexos	Projetos complexos
Onde não usar	-	Projetos pequenos	Projetos pequeno

Fonte: (ARGUMANIS; MOQUILLAZA; PAZ, 2021) e (BAXEVANIS, 2006).

A Tabela 7 é uma adaptação da tabela proposta por Argumanis, Moquillaza e Paz (2021) e sintetiza comparativamente os métodos propostos pelos autores para a etapa de Implementação.

Tabela 7 – Análise comparativa das técnicas UCD para etapa de Implementação (Adaptada)

Método	Avaliação Heurística	<i>Thinking Aloud</i>	Testes de Usabilidade
Vantagem	Possibilita a identificação de problemas de design	Identifica os sentimentos dos usuários ao interagirem com alguma tela	Possibilita a identificação de problemas de design
Desvantagem	Alto custo	Necessita de participação de terceiros	Podem surgir testes incertos devido a usuários fadigados, principalmente se já tiveram dificuldades na interação com o produto
Custo	Alto	Baixo	Alto
Estágio do projeto	Implementação	Implementação	Implementação
Onde usar	Redesign de projetos ou melhorias	Redesign de projetos ou melhorias	Redesign de projetos ou melhorias
Onde não usar	Projetos novos	Projetos novos	Projetos novos

Fontes: (ARGUMANIS; MOQUILLAZA; PAZ, 2021) e (BAXEVANIS, 2006).

2.4.2 Metodologias Seleccionadas por Argumanis, Moquillaza e Paz

Com base nas metodologias abordadas por [Argumanis, Moquillaza e Paz \(2021\)](#) na Tabela 3, o *Design* em paralelo ao *Sprint* foi a metodologia selecionada. Tendo potencial para ser implementada em grandes projetos, essa técnica facilita o planejamento das iterações e possibilita que o time de desenvolvimento trabalhe em paralelo ao time de *design*. Falhas são detectadas com antecedência, permitindo que os *designers* realizem alterações e validem com técnicas específicas. Todavia, para evitar problemas de comunicação ocasionados pelo paralelismo das equipes, a equipe ou o principal membro do time de desenvolvimento deve participar da validação dos protótipos junto aos times de produto e *design*, e que ambas as equipes devem participar juntas em todas as cerimônias do *Scrum*. Isso é fundamental para aprimorar a comunicação entre ambas as partes ([ARGUMANIS; MOQUILLAZA; PAZ, 2021](#)).

2.4.3 Técnicas Seleccionadas por Argumanis, Moquillaza e Paz

A partir das análises comparativas entre as técnicas, [Argumanis, Moquillaza e Paz \(2021\)](#) selecionaram 9 técnicas que podem ser implementadas nas etapas Inicial, Planejamento e Implementação.

- **Protótipo de Papel:** Foi selecionada por possibilitar uma fácil visualização e avaliação do *design* por um baixo custo.
- **Personas:** Foi selecionado para garantir que toda a equipe tenha uma ideia clara dos usuários.
- **Sprint 0:** Foi escolhido devido à necessidade dos designers de um espaço para entender as necessidades do usuário e definir uma visão global das interfaces.
- **Design em pares:** Foi escolhido devido à necessidade dos *designers* de um espaço para entender as necessidades do usuário e definir uma visão global das interfaces.
- **Classificação de Cartas:** Foi selecionado somente para projetos que necessitam de informações estruturadas, pois facilita o processo de definição da estrutura.
- **Avaliação Heurística:** Foi selecionado apenas para projetos de redesenho, pois permitir que o especialistas em *design* identifiquem os problemas já existentes.
- **Investigação contextual:** Foi selecionado para obter conhecimento das atividades do usuário, o qual é importante para o processo de definição de requisitos.
- **Thinking Aloud:** Foi selecionado porque é eficaz para obter informações sobre a experiência do usuário ao usar o sistema.

- **Mapa da Jornada do Usuário:** Foi selecionado porque é eficaz para obter uma ideia do processo de negócios que os usuários seguirão e, portanto, identificar detalhadamente as necessidades dos usuários.

2.4.4 Etapas do Scrum-UCD

A partir das análise comparativa, [Argumanis, Moquillaza e Paz \(2021\)](#) propõe a inserção das técnicas selecionadas nas etapas Inicial, de Planejamento e de Implementação.

- **Etapa Inicial (Sprint 0):**

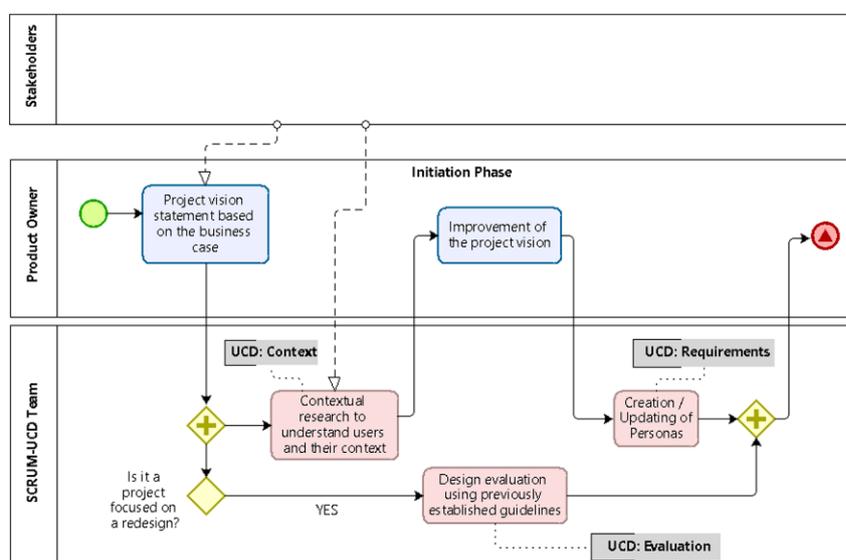
A primeira fase do *framework* começa com o documento de visão do projeto criado pelo *Product Owner*, o qual é aprimorado após um processo de investigação contextual que permite ao time *Scrum-UCD* conhecer os usuários reais e seus contextos, para poder identificar suas necessidades de forma transparente ([ARGUMANIS; MOQUILLAZA; PAZ, 2021](#)).

Em seguida é aplicada a técnica de Personas, elaborando os perfis de usuários com as informações coletadas durante a investigação contextual. E por fim o *backlog* do produto é criado com as listas dos requisitos do projeto, os quais são validados e aprimorados iterativamente pelos usuários.

Se o projeto for voltado para um *redesign*, a avaliação heurística deve ser executada paralelamente à investigação contextual e à criação ou atualização de Personas ([ARGUMANIS; MOQUILLAZA; PAZ, 2021](#)).

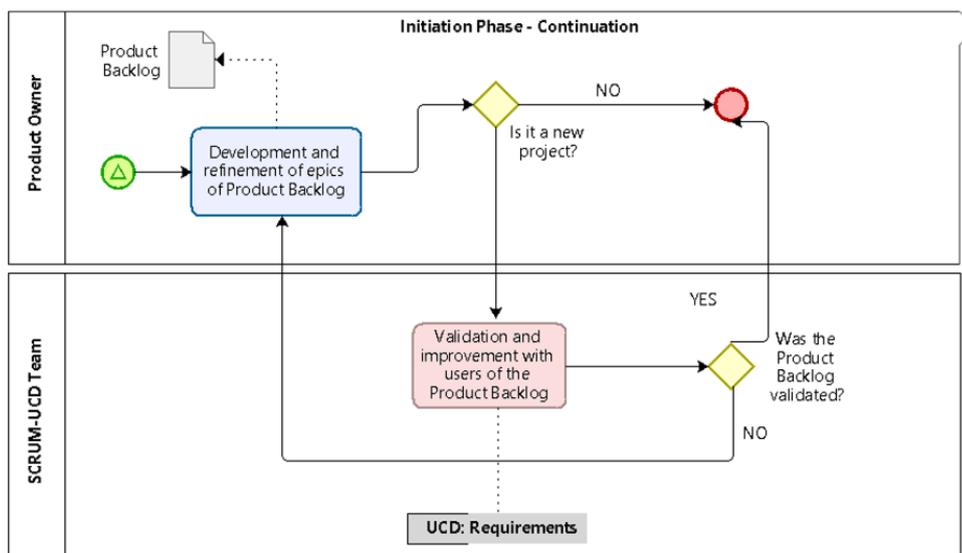
As Figuras 6 e 7 ilustram o processo da Etapa Inicial descrito acima.

Figura 6 – Etapa Iniciação



Fonte: ([ARGUMANIS; MOQUILLAZA; PAZ, 2021](#))

Figura 7 – Etapa Iniciação Continuação



Fonte: (ARGUMANIS; MOQUILLAZA; PAZ, 2021)

- **Etapa de Planejamento (Sprint 0):**

Na fase de planejamento é utilizado a técnica de classificação de cartas com os usuários para melhor entendimento do projeto, logo após é desenvolvido o mapa de jornada de usuário com o intuito de detalhar o fluxo de trabalho que os usuários seguirão ao usar o sistema (ARGUMANIS; MOQUILLAZA; PAZ, 2021).

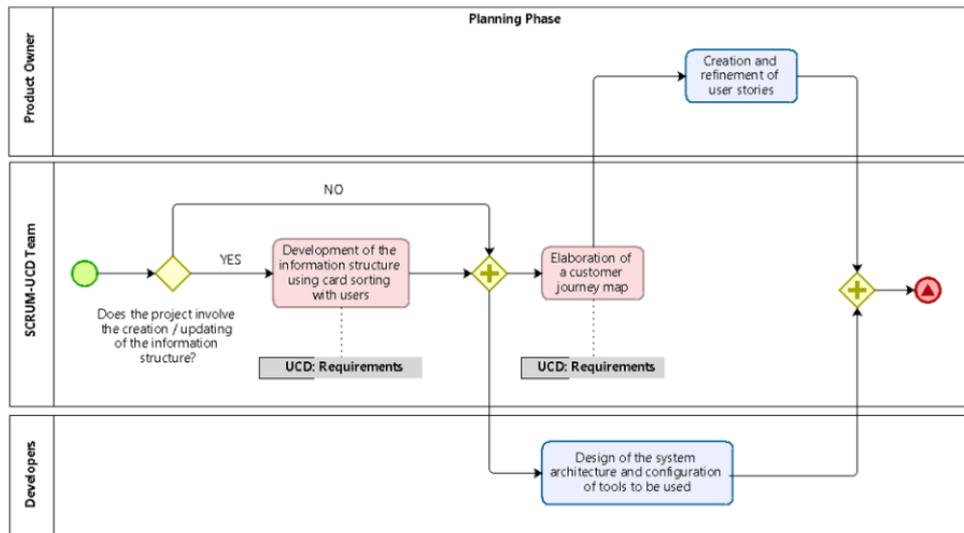
Com a ajuda dessas técnicas, as histórias de usuários são criadas e estimadas, inicia-se então o *design* de protótipos de papel para a definição de uma visão global do sistema. Emfim, as tarefas são identificadas e incluídas no *Sprint backlog*, criado para os *Sprints* de *design* e desenvolvimento (ARGUMANIS; MOQUILLAZA; PAZ, 2021).

A equipe de desenvolvimento deve definir a arquitetura do sistema e configurar as ferramentas de desenvolvimento em paralelo à elaboração do mapa da jornada do usuário e à criação e refinamento de histórias de usuários (ARGUMANIS; MOQUILLAZA; PAZ, 2021).

Antes do início da fase de implementação, os desenvolvedores devem explicar as restrições técnicas do projeto para serem consideradas durante a concepção dos protótipos, restrições tais como tempo, custo, nível técnico da equipe e limitações do ambiente de desenvolvimento (ARGUMANIS; MOQUILLAZA; PAZ, 2021).

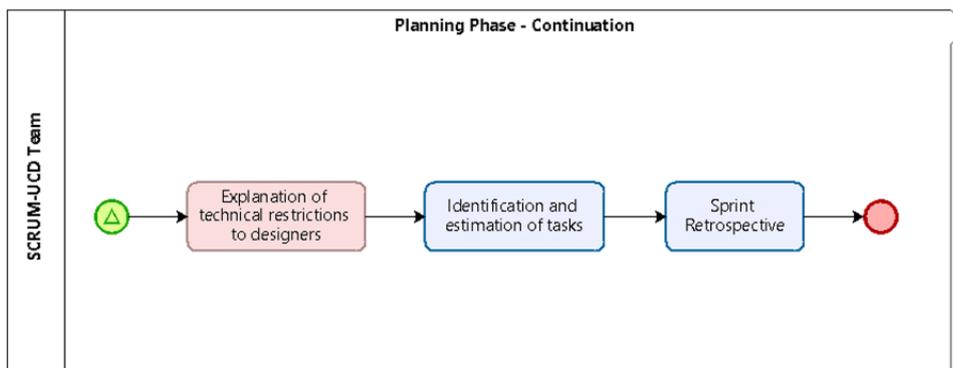
As Figuras 8 e 9 ilustram o processo da Etapa de Planejamento descrito acima.

Figura 8 – Etapa Planejamento



Fonte: (ARGUMANIS; MOQUILLAZA; PAZ, 2021)

Figura 9 – Etapa Planejamento Continuação



Fonte: (ARGUMANIS; MOQUILLAZA; PAZ, 2021)

• **Etapa de Implementação:**

A etapa de implementação começa com o emprego da técnica *thinking aloud*, para a validação dos protótipos de papel. Quanto menor o projeto, menor o risco de desperdício de recursos caso uma mudança seja solicitada em fases posteriores da implementação, então pequenos projetos podem executar esta validação apenas uma vez e começar a desenvolver os protótipos de alta fidelidade, considerando as observações do usuário, os quais foram apontados durante o teste do protótipo de baixa fidelidade (ARGUMANIS; MOQUILLAZA; PAZ, 2021).

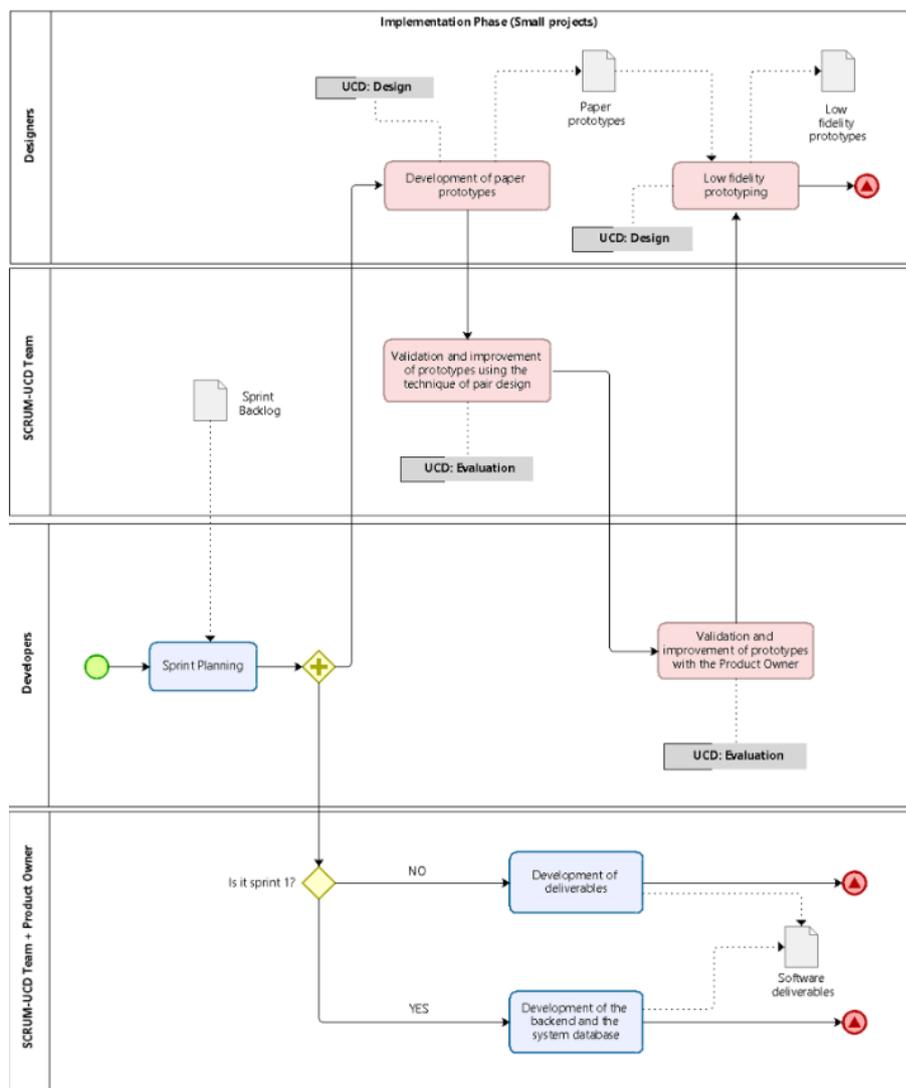
Paralelamente aos *Sprints de design*, são executados os *Sprints de desenvolvimento*. Os desenvolvedores aproveitam o primeiro *Sprint* para definir a arquitetura do sistema e

as ferramentas que utilizarão (*frameworks*, ambiente de desenvolvimento, bancos de dados, etc.), enquanto os *designs* são iniciados. Durante os próximos *Sprints* (*Sprint-i*), os desenvolvedores focam no desenvolvimento das entregas, implementando os protótipos desenhados nos *Sprints* anteriores (*Sprint i-1*). Portanto, os *Sprints* de *design* estão sempre à frente dos *Sprints* de desenvolvimento em uma iteração (ARGUMANIS; MOQUILLAZA; PAZ, 2021).

Ao final do *Sprint*, as entregas dos protótipos de alta fidelidade e do *software* implementado são validadas com o *Product Owner*, e passam por uma última validação do usuário. Os resultados ajudam a identificar as mudanças que o time precisa fazer, e essas mudanças são incluídas na nova versão do *Sprint backlog*. Por fim, o *Sprint* termina com a cerimônia da retrospectiva. (ARGUMANIS; MOQUILLAZA; PAZ, 2021).

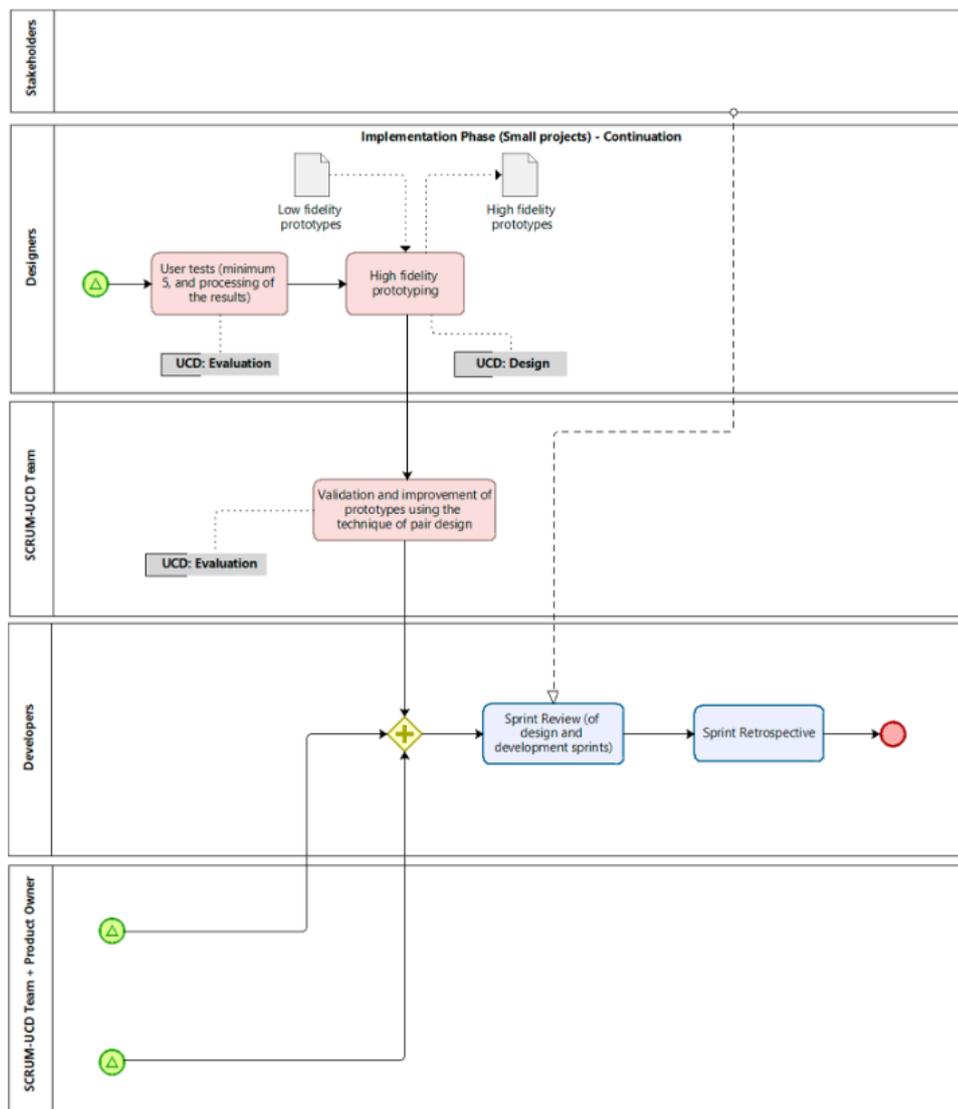
As Figuras 10 e 11 ilustram o processo da Etapa de Implementação descrito acima.

Figura 10 – Etapa de Implementação



Fonte: (ARGUMANIS; MOQUILLAZA; PAZ, 2021)

Figura 11 – Etapa de Implementação Continuação



Fonte: (ARGUMANIS; MOQUILLAZA; PAZ, 2021)

2.5 Considerações Finais do Capítulo

Neste capítulo foi apresentado o *framework* ágil *Scrumban*, o processo de *Design* Centrado no Usuário - UCD, e o *Scrum-UCD*, que juntos fundamentam uma visão ampla sobre como combinar as melhores práticas de gerenciamento de projetos com abordagens centrada no usuário.

A partir deste estudo, foram selecionadas técnicas apresentadas na seção 2.2.1 - Princípios e Técnicas para o processo ágil Scrumban. Para o UCD e Scrum-UCD foi escolhida a metodologia de *Design* em paralelo ao Sprint abordada na seção 2.4.2 - Metodologias Selecionadas, e as técnicas selecionadas na seção 2.4.3 - Técnicas Selecionadas.

3 Materiais e Métodos

3.1 Considerações Iniciais do Capítulo

Neste capítulo retoma-se o Plano Metodológico, apresentado no *Capítulo 1*, constituído de 4 grandes fases: Planejamento; Coleta de Dados; Análise de Dados; e Relatórios. Em seguida, apresenta-se um detalhamento dos itens do Protocolo de Estudo de Caso.

3.2 Planejamento de Pesquisa

Esta fase compreende o planejamento, isso é, dado o tema de pesquisa e o contexto, foi levantada a questão de pesquisa e definido o objetivo a ser atingido, assim como o Plano de Pesquisa para alcançá-lo.

O Capítulo 1 contempla esta fase. Foi selecionada a técnica de Estudo de Caso, dado que este trabalho é a comparação de um processo de uma *startup* com o *framework Scrumban-UCD* e demais abordagens.

Para a execução do Estudo de Caso adotou-se o Protocolo proposto por [Brereton et al. \(2008\)](#), no qual os pesquisadores puderam se orientar no planejamento e na execução da técnica de estudo de caso, compreendendo onze itens, conforme Tabela 8.

O planejamento detalhado de cada um dos itens é apresentado nas subseções seguintes.

Tabela 8 – Protocolo de Estudo de Caso

Etapa	Descrição
<i>Background</i>	Identificar pesquisas anteriores sobre o tópico Definir a principal questão de pesquisa que está sendo abordada por este estudo Identificar quaisquer questões de pesquisa adicionais que serão abordadas
<i>Design</i>	Identificar se é um projeto de caso único ou caso múltiplo Descrever o objeto de estudo Identificar propostas de cada questão de pesquisa
Seleção de Caso	Critérios para seleção de casos
Procedimentos e papéis do estudo	Procedimentos que regem os procedimentos de campo Papéis dos membros da equipe de pesquisa do estudo de caso
Coleta de Dados	Identificar os dados a serem coletados Definir um plano de coleta de dados Definir como os dados serão armazenados
Análise	Identificar os critérios para interpretar os resultados do estudo de caso Definir quais dados serão utilizados para a resposta da pesquisa Considerar a gama de possibilidades e suas explicações A análise deve ocorrer à medida que a tarefa do estudo de caso progride
Validade do plano	Verificar a validade do plano em relação aos aspectos gerais, validade interna, externa e da validade do constructo
Limitações do Estudo	Especificar questões de validade residual, incluindo potenciais conflitos de interesse (ou seja, inerentes ao problema, em vez de decorrentes do plano).
Relatórios	Identificar público-alvo e o relacionamento com outros estudos
Cronograma	Fornecer estimativas de tempo para todas as etapas principais: planejamento, coleta de dados, análise de dados, geração de relatórios
Apêndices	Registrar informações complementares para compreensão do estudo de caso

Fonte: (BRERETON et al., 2008)

3.2.1 *Background*

Este item do Protocolo, *background*, é resultante da *Pesquisa Bibliográfica e Pesquisa Documental*, que possibilitou a contextualização deste trabalho, com as buscas nas principais bases científicas voltadas para metodologias ágeis, como o *Scrum* e *Kanban*, Experiência de Usuário (UX) e no *Design Centrado no Usuário (UCD)*.

Os resultados constituem o *Capítulo 1* e o *Capítulo 2*.

3.2.2 *Design*

Como *design*, este estudo se caracteriza como caso único, uma vez que a pesquisa envolve o projeto Groovoo da *Atena Solutions*.

3.2.3 Seleção de Caso

A seleção de caso se caracteriza pelo *Projeto Groovoo* da *Startup Atena Solutions*.

A descrição deste objeto de estudo, com um detalhamento do processo de desenvolvimento de *software*, é apresentada no *Capítulo 4 - Atena Solutions*

3.2.4 Procedimentos e papéis do estudo de caso

Quanto aos papéis, os autores pesquisadores deste trabalho têm vínculo com a *Startup*. Um como desenvolvedor líder técnico e outro, agora ex-colaborador. E a pesquisa tem o patrocínio do CEO da *Startup* (Sr. Frederico Castro) para a execução deste trabalho de pesquisa, conforme carta de apresentação à empresa, Apêndice - A.

Quanto aos procedimentos, esses foram definidos alinhados aos objetivos específicos, isto é, foram planejadas 3 atividades:

- I. Analisar e selecionar diretrizes empregadas pelo *framework Scrum-UCD* em relação ao Scrum e diretrizes UCD da literatura;
- II. Identificar no processo de desenvolvimento de *software Groovoo* oportunidades de inserção de diretrizes do UCD;
- III. Avaliar e apontar possibilidades de inserção de diretrizes UCD no *Processo de desenvolvimento de Software do Groovoo*.

3.2.5 Coleta de Dados

O item Coleta de Dados do Protocolo constitui o emprego de atividades para o alcance dos três objetivos específicos.

A descrição desta coleta é apresentada no *Capítulo 5 - Diretrizes de UCD para o processo Groovoo*.

3.2.6 Análise

Neste trabalho, a *Análise dos Dados* foi realizada em paralelo com a *Coleta de Dados*, pois optou-se por uma abordagem flexível, em que os pesquisadores, ao coletar dados e propor diretrizes as avaliassem.

Contudo, no final da coleta, as diretrizes apontadas foram apresentadas aos diretores da startup *Atena Solutions*.

A descrição da Coleta assim como da Análise de Dados é apresentada no *Capítulo 5 - Diretrizes de UCD para o processo Groovoo*

3.2.7 Validade do plano

O objetivo é identificar o domínio para o qual os achados do estudo podem ser generalizados. Inicialmente o foco é o *Projeto Groovoo* da *Atena Solutions*, mas espera-se que a abordagem possa ser empregada nos demais projetos da *Atena Solutions*.

3.2.8 Limitações do Estudo

O escopo desta pesquisa está delimitado a um processo de um projeto, *Projeto Groovoo*, da *Atena Solutions*, restringindo a generalização dos resultados.

3.2.9 Relatório

O Relatório desta pesquisa se materializa como sendo esta monografia de TCC.

3.2.10 Apêndices

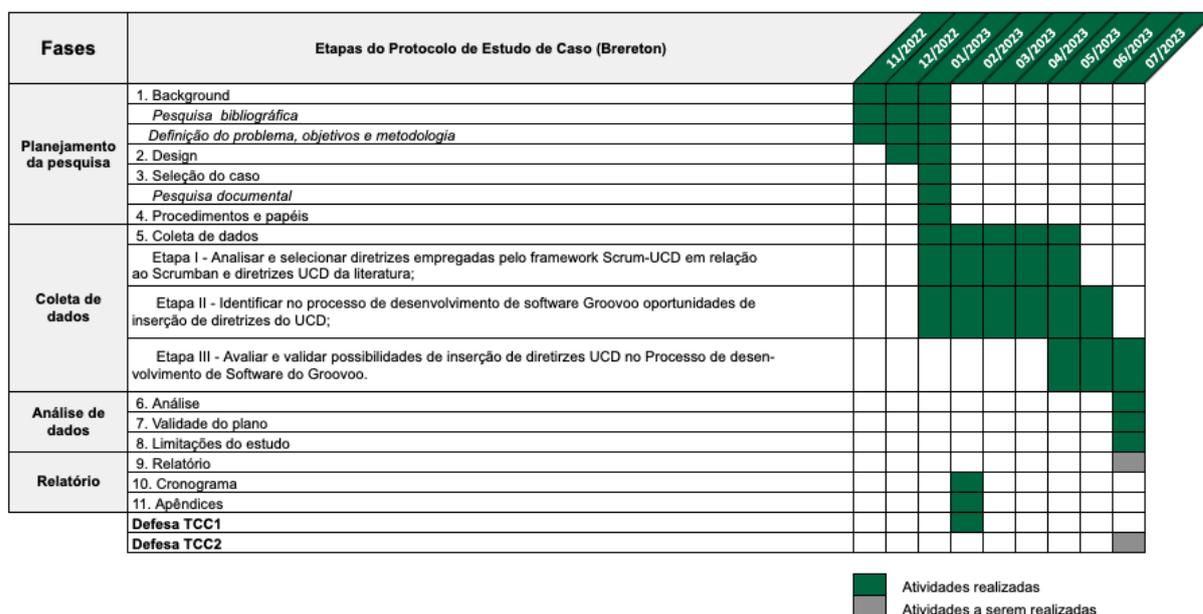
Os documentos produzidos pelos autores deste trabalho, como a carta de apresentação à empresa e os formulários utilizados constam como apêndices.

- Apêndice *A* - *Carta de Solicitação*
- Apêndice *B* - *Questionário Grupo Focal*
- Apêndice *C* - *Questionário da Entrevista Groovoo*
- Apêndice *D* - *Questionario e User Flow Thinking Aloud*

3.2.11 Cronograma

O cronograma proposto para esta pesquisa encontra-se detalhado na Figura 12. Apresentam-se as atividades realizadas conforme a metodologia empregada.

Figura 12 – Cronograma do projeto de pesquisa



Fonte: Os autores

3.3 Considerações Finais do Capítulo

No presente Capítulo, apresentou-se o plano metodológico planejado e empregado e as técnicas e instrumentos utilizados, visando contribuir para inclusão de diretrizes de Design Centrado no Usuário no processo de desenvolvimento do projeto Groovoo. Nos capítulos seguintes apresenta-se o objeto de estudo deste trabalho, a empresa Atena Solutions, seguido do capítulo de Coleta e Análise de Dados.

4 Atena Solutions

4.1 Considerações Iniciais do Capítulo

Neste capítulo apresenta-se a caracterização da *Startup Atena*, com a descrição do modelo de processo de *software* empregado, baseado no *framework Scrumban*, seguido da caracterização do *Projeto Groovoo*, objeto de estudo deste trabalho.

4.2 Atena Solutions

Fundada em 25 de julho de 2006 por Frederico Castro e Arthur Morales, com operações em São Francisco (CA), Austin (TX), Nova York (NY), São Paulo (SP), Brasília (DF), Manaus (AM) e Vancouver (BC), a *Atena Solutions* é uma *startup* focada no desenvolvimento de *software* e na computação em nuvem, baseados no modelo de sistema *Software as a Service* (SaaS) - ou Software como Serviço, em português.

A Atena possui clientes como: *Caixa Econômica Federal; Infraero Aeroportos; Kikkin Internet Banking; MetaLives; Rybena Acessibilidade Web; Sodexo; Tunad*, e a *Groovoo*, objeto de estudo deste trabalho.

Nas seções seguintes apresenta-se uma síntese dos processos empregados para o desenvolvimento de aplicações pela empresa *Startup Atena Solutions*, com origem nos *frameworks* Ágeis *Scrum*, *Kanban* e *Scrumban* na plataforma *Groovoo*, um portal de venda de ingressos para eventos sediados em estados norte-americanos.

4.2.1 Groovoo

Idealizado em meados de 2021, a *Groovoo* é um aplicativo de rede social e um portal de emissão de ingressos para eventos localizados em estados norte-americanos, que visa permitir que os usuários interajam entre si com interesses variados, como: relacionamentos, criação de laços de amizade, encontros por diversão ou até mesmo negócios.

4.3 Scrumban Groovoo

A *Atena Solutions* implementa para o projeto *Groovoo* a combinação dos *frameworks* ágeis *Scrum* e *Kanban*, o *Scrumban*. Decerto, o *Scrumban* é usado para harmonizar as melhores características de ambos os métodos, concentrando a natureza normativa do método *Scrum*, juntamente a capacidade de melhoria e visualização de processos do método *Kanban*. A equipe *Groovoo* implementa essa metodologia com uma equipe definida

basicamente em seis áreas de atuação, sendo elas: Time de Produto; *Scrum Master*; Time de Infraestrutura; Time de *Design*; Time de Qualidade, e o Time de Desenvolvimento, este que, é composto pelos desenvolvedores e equipe de teste.

4.3.1 Time de Produto

O Time de Produto visa a gerência do artefato desenvolvido, visando maximizar o valor do produto, levantando requisitos e direcionando-os ao público adequado.

- Shareholders

Os *Shareholders* são um grupo formado por três indivíduos que possuem ações majoritárias da Groovoo, eles buscam por melhorias no sistema, afim de atrair mais stakeholders e aumentar o valor do produto.

- Stakeholders

Os *Stakeholders* são um grupo variado de indivíduos interessados e impactados pelo valor gerado pela Groovoo, sendo eles: organizadores de eventos e investidores.

- Product Manager

O *Product Manager* está atrelado a todas as áreas do desenvolvimento do produto como negócios, *marketing* e desenvolvimento. Tem como função manter o ritmo das entregas orquestrando todos o setores, participando como um facilitador para os impedimentos e garantindo o alinhamento da visão do produto com as necessidades do cliente. De modo geral, ele é responsável pela fluidez da construção do produto.

- Product Owner

O *Product Owner* (PO) está atrelado ao time de desenvolvimento e o cliente, ele tem como função transformar as necessidades dos usuários em funcionalidades a serem priorizadas no *backlog* conforme o valor que elas agregarão ao produto. É responsável também pelos critérios de aceitação da funcionalidade desenvolvida ao final de cada história.

4.3.2 Scrum Master

O *Scrum Master* participa das *daily*s de todas as equipes, atuando como um facilitador para os impedimentos, preocupando-se com o desenvolvimento interpessoal da equipe e satisfação pessoal de cada desenvolvedor acerca da empresa e o seu papel dentro dela.

4.3.3 Time de desenvolvimento

- Líder Técnico

O Líder Técnico da Groovoo, é um desenvolvedor que pode propor mudanças na arquitetura e tecnologias utilizadas na confecção do software. As demandas mais complexas ficam a cargo dele e é sua responsabilidade auxiliar e destravar impedimentos técnicos dos demais desenvolvedores da equipe.

- Desenvolvedor

Os Desenvolvedores atuam de maneira *full stack* nas demandas priorizadas pelo PO no *Sprint backlog*, ou seja, são responsáveis pelo desenvolvimento mais técnico, trabalhando explicitamente no código-fonte, sendo ele tanto no *backend*, como no *frontend*.

4.3.4 Time de Qualidade

O Time de qualidade é composto por profissionais responsáveis por realizar testes de usabilidade na aplicação a partir do momento que as alterações são levadas ao ambiente de homologação. Desse modo, esse time visa analisar o ponto de vista do consumidor, provocando diferentes casos de uso e analisando a experiência do usuário no uso da aplicação.

4.3.5 Time de infraestrutura

O Time de infraestrutura é formado por engenheiros *DevOps* e eles atuam na montagem do ambiente de todos os projetos da empresa, são responsáveis pela manutenção, suporte e evolução dos sistemas conforme as necessidades do produto.

4.3.6 Time de Design

O Time de *design* é responsável por criar protótipos de alta fidelidade fundamentados pelas especificações do Time de produto, visando proporcionar a melhor experiência de uso para os usuários. Desse modo, a partir prototipação, a amostra criada é apresentada ao Time de produto e ao Líder técnico do time de desenvolvimento, de modo a, realizar ajustes e criar uma versão final que será implementada ao *backlog* e posteriormente concretizada em um *Sprint*.

4.3.7 Processos da Groovoo

O processo de desenvolvimento do Projeto Groovoo é baseado no *framework Scrum-ban*, com foco em uma estrutura ágil para que as equipes trabalhem juntas de maneira

eficaz. Fundamentando-se nesse princípio, essa metodologia propõe que um projeto seja ramificado em diferentes ciclos de atividades, que devem ser apresentados em um *Timebox*, cujo conjunto de atividades especificadas deve ser executadas. Desse modo, esse período conta com frequentes reuniões, as chamadas *daily*s, que objetivam o alinhamento dos colaboradores envolvidos em cada ciclo, tipicamente espaçados por semanas ou até meses, dependendo, é claro, com a conformidade das demandas.

Dessa forma, o processo inicia-se com as demandas da área solicitante especificadas no *Product Backlog* pelo Time de Produto. Nesse contexto, elas são levadas em uma reunião de planejamento entre o Time de Produto e o Time de Desenvolvimento, que ocorre de maneira oral e demonstrativa, abordando quais entregáveis devem ser realizados durante cada ciclo iniciado, a priorização de cada um deles, juntamente às análises da equipe sobre tempo, esforço e possíveis dificuldades que poderão ser encontradas, criando assim, os chamados *Sprints plannings*.

Após esse planejamento, as tarefas levantadas para cada *Sprint planning*, são incorporadas ao sistema de controle e gestão de fluxo, o *Kanban*. Esse método é utilizado para a melhoria visual das demandas solicitadas, por meio da incorporação de *cards* em fluxos de trabalhos, categorizados por:

1. **Backlog:** O *Backlog* consiste em uma categoria responsável por armazenar tudo aquilo que foi especificado pelo time de produto e será posteriormente levado a um *Sprint*.
2. **Sprint:** Normalmente numeradas, a categoria *Sprint* é responsável por registrar as demandas que deverão ser realizadas no decorrer daquele intervalo de tempo.
3. **In Progress:** Essa categoria é responsável por conter as demandas que já estão sendo realizadas, e posteriormente serão incorporadas ao servidor de homologação para a realização de testes de usabilidade pelo Time de Qualidade.
4. **Testing:** Essa categoria é responsável por abranger as demandas que já foram implementadas pelo Time de Desenvolvimento e estão disponíveis para testes pelo Time de Qualidade.
5. **To Deploy:** Essa categoria é responsável por abranger a última etapa do processo de categorização da metodologia *Kanban*, responsável por comportar as demandas que já foram testadas em ambiente de homologação e já estão disponíveis para os usuários em ambiente de produção.

Durante cada ciclo *Sprint*, são realizadas as chamadas, *Dailys*, um rito ágil fundamentado em reuniões diárias ocorridas manhã, com duração de aproximadamente 15 minutos e a presença do *Product Manager*, *Product Owner*, Time de Desenvolvimento,

Time de Qualidade e o Time de *Design*, como o objetivo disseminar o conhecimento sobre o que foi feito no dia anterior, identificar impedimentos e abordar o trabalho que será realizado no dia atual.

Ainda durante o ciclo *Sprint*, caso seja necessário, devido à dificuldade de alguma demanda, a equipe de desenvolvimento emprega o procedimento *Pair Programming*, uma prática que consiste na programação compartilhada entre colaboradores. Além disso, dependendo de como o *Sprint* esteja se encaminhando, são realizadas reuniões semanais com membros do Time de Produto, a fim de, esclarecer dúvidas sobre alguma especificação e eliminar possíveis travamentos.

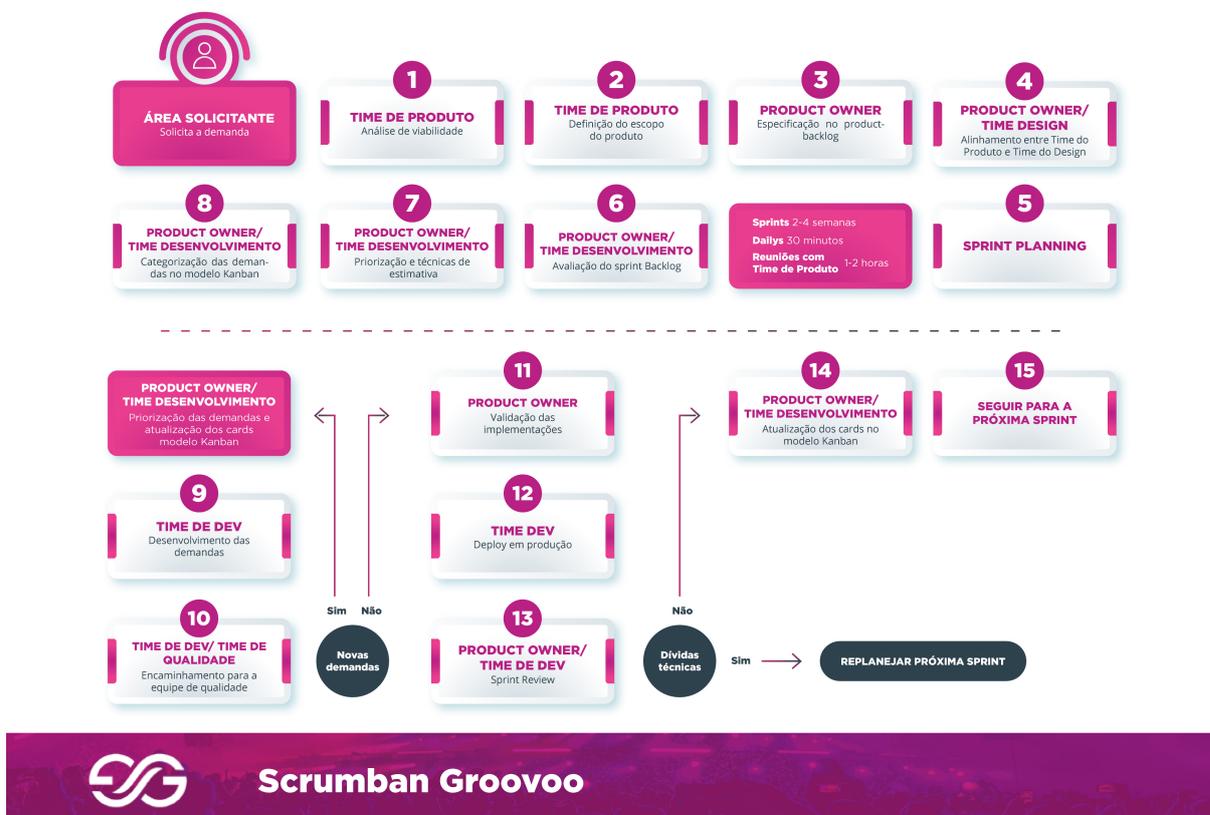
Após realizadas, o Time de Desenvolvimento encaminha as demandas para o ambiente de homologação. Nessa etapa o Time de Qualidade entra como principal ativo, responsável por avaliar a qualidade das demandas, verificando os critérios de aceitação em acordo com as funcionalidades implementadas, realizando testes de usabilidade, segurança e qualidade. Desse modo, esse ambiente é responsável por conter uma versão de fidelidade alta da aplicação, que será posteriormente enviada para o ambiente de Produção e estará disponível para os usuários.

Posteriormente aos testes, o *Deploy* é feito para o ambiente de produção de forma automatizada, com a *pipeline* construída pelo Time de Infraestrutura, utilizando tecnologias que garantem a integridade e confiabilidade do sistema. Desse modo, a nova funcionalidade fica disponível para o usuário.

Por fim, é realizado um *Sprint Review*, onde os times da Groovoo se reúnem e apresentam como foi o andamento daquele período que está se encerrando. O Time de Desenvolvimento apresenta ao Time de Produto as funcionalidades implementadas, abordando pontos positivos, dificuldades encontradas e como foram contornadas, além de possíveis implementações que não foram concretizadas, ocasionando assim o replanejamento da próxima *Sprint* com a adição dessas demandas como dívidas técnicas, e assim o ciclo se inicia novamente.

Descrito as características dos times acima, a Figura 13 exemplifica o processo de desenvolvimento do software do projeto Groovoo.

Figura 13 – Processo de desenvolvimento de *software* do Projeto Groovoo



Fonte: Os autores

No projeto Groovoo, esse ciclo de desenvolvimento ágil de *software* proposto pelo *Scrumban* oferece uma boa gestão de tempo em relação às demandas solicitadas, mas por conta de curtos prazos ainda carece da percepção das reais necessidades dos usuários. Dada a carência de procedimentos que envolvam o usuário no ciclo de desenvolvimento de *software*, problemas de aceitação dos usuários em relação à implementação de novas funcionalidades pode ser algo frequente.

Neste contexto, a integração do UCD ao *Scrumban* propõe diminuir a distância entre as expectativas e necessidades dos usuários, e a solução final oferecida pela empresa, evitando que o *software* desenvolvido gere dúvidas, comportamentos inesperados e até, no pior dos casos, a não adaptação às evoluções implementadas.

4.4 Considerações Finais do Capítulo

Neste capítulo foi abordado o processo da Groovoo para o desenvolvimento de um produto de *software*, considerando toda a organização da equipe e suas responsabilidades em suas respectivas etapas de confecção do produto, desde do cliente até o usuário final.

5 Diretrizes de UCD para o processo Groovoo

5.1 Considerações Iniciais do Capítulo

Neste capítulo retoma-se os *Objetivos Específicos* deste trabalho apresentados no Capítulo 1. Inicialmente, é apresentada a análise e seleção de diretrizes do UCD com base na literatura e quanto às possibilidades de inserção no processo de desenvolvimento do Projeto Groovoo da *Startup* Atena. Em seguida, apresentam-se as oportunidades de inserção das diretrizes selecionadas ao processo de desenvolvimento. Por último, a consolidação da inserção das diretrizes selecionadas.

5.2 Análise e seleção de diretrizes

Esta seção apresenta os procedimentos e etapas adotadas para a produção do 1.º objetivo específico:

- *Analisar e selecionar diretrizes empregadas pelo framework Scrum-UCD em relação ao Scrumban e diretrizes UCD da literatura.*

A *Startup* Atena, detentora do Projeto Groovoo, possui limitações geográficas, financeiras e de tempo, que impactam significativamente na escolha de diretrizes ofertadas pelo UCD e o *framework Scrumban*.

Atualmente os colaboradores residem em diferentes países, o que impacta ativamente em dinâmicas presenciais recomendadas pelo UCD. Além disso, a quantidade de recursos financeiros e de equipe também afeta na escolha de técnicas e metodologias.

Desse modo, também considerando que o Projeto Groovoo está em estágio de redesenho, na análise e seleção, foram identificadas diretrizes que sejam viáveis e agreguem real valor ao atual processo de desenvolvimento de software, considerando os curtos prazos de entrega do projeto.

5.2.1 Técnicas Selecionadas Scrumban

A empresa já possui um processo de desenvolvimento fundamentado no *framework Scrumban*, porém não faz uso de algumas técnicas essenciais do Kanban.

Neste trabalho foram selecionadas algumas técnicas para proposição de inserção ao atual processo:

- **Ready Queue:** Esta técnica foi selecionada por gerenciar demandas de alta prioridade em todo o momento, a *Ready Queue* é uma seção do quadro Kanban, que fica entre o *Backlog* e o *In Progress* e contém as tarefas priorizadas do *backlog*, nesta fileira as atividades não estão atreladas a ninguém, porém assim que alguém tiver disponibilidade deve escolher uma das opções que estão na *Ready Queue* para trabalhar, invés de alguma tarefa do *backlog* geral. Esta seção pode ser alterada ao longo de todo o projeto independente de qual parte da *sprint* está, ou seja, se surgir uma demanda de alta prioridade quando todos já estão trabalhando em suas respectivas atividades o PO (*Product Owner*) pode colocar essa tarefa na *Ready Queue* e quando alguém da equipe finalizar o seu trabalho deve iniciá-la imediatamente sem esperar a próxima *Sprint Planning* ou aprovação do PO (*Product Owner*).
- **Limite de Trabalho em Andamento (*Work-in-Progress* - WIP):** Esta técnica foi escolhida por ser uma ferramenta para um maior controle de qualidade, levando em consideração a produtividade e a qualidade de vida da equipe. WIP é a quantidade de tarefas que estão em andamento e o objetivo em limitá-la é aumentar o foco da equipe somente nas demandas designadas para aquela *sprint*, portanto se chegar uma atividade nova e o limite de WIP estiver esgotado, então essa tarefa deve entrar no *backlog* e se for de alta urgência ela é alocada para a *Ready Queue*, mas só poderá entrar em progresso depois de alguma atividade que está em *In Progress* sair desta fileira.
- **Políticas de Trabalho Explícitas:** Esta abordagem foi selecionada por se tratar de uma forma prática e eficiente de implementar novas metodologias ao fluxo de trabalho. Nisto é necessário definir as regras e explicá-las no começo de cada projeto isto inclui tanto regras de convivência e cultura da empresa e do cliente, como até mesmo como será realizado todo o processo do fluxo de trabalho. Este conjunto de regras, instruções e planejamentos devem ser revisitados ao longo do projeto é até mesmo recomendado que eles estejam escritos em algum ambiente que todos do projeto visitem, por exemplo, no quadro de tarefas.

No Capítulo 2, Referencial Teórico, foram apresentadas diversas técnicas do Scrum-ban levantadas por autores.

Ao se analisar o cenário e limitações da *Startup* Atena e do Projeto Groovoo, as técnicas foram selecionadas por oferecer um alto valor de retorno por um baixo custo financeiro e de aplicação.

Na Tabela 9 apresenta-se o conjunto de técnicas selecionadas para o Processo da empresa, assim como uma comparação entre elas. Algumas, os autores denominam métodos em detrimento de técnicas.

Tabela 9 – Análise comparativa das técnicas do Scrumban selecionadas.

Técnica /Método	<i>Ready Queue</i>	Limite de WIP (Work-in-Progress)	<i>Políticas de Trabalho Explícitas</i>
Vantagem	Possibilita uma melhor gestão das tarefas com alta prioridade ao longo de todo o projeto	Limita a quantidade de trabalho na <i>sprint</i> proporcionando um foco maior nas tarefas atuais e uma melhor qualidade nos entregáveis	Implementar novas metodologias ao fluxo de trabalho de forma mais branda a equipe.
Desvantagem	Demandas de alta prioridades não são imediatamente resolvidas	Precisa ter mais colaboradores para poder ter mais tarefas em progresso simultaneamente	-
Custo	Baixo	Baixo	Baixo
Onde usar	Qualquer projeto	Qualquer projeto	Qualquer projeto
Onde não usar	-	-	-

Fonte: Os autores.

5.2.2 Técnicas Selecionadas UCD

A empresa não apresenta diretrizes de *Design Centrado no Usuário* implementadas ao atual processo de desenvolvimento de *software*.

Neste trabalho foram selecionadas quatro técnicas em acordo com as etapas de *Observação, Ideação, Prototipagem e Testes*, propostas por Norman (2013):

- **Observação:** Nesta etapa, o Time de Produto irá identificar e compreender a natureza do problema por meio da identificação do público alvo, observando de maneira direta e sistemática como os usuários atuais e futuros se comportam com as funcionalidades disponibilizadas. As informações coletadas devem ser suficientes para identificar comportamentos, dificuldades e necessidades.
 - Grupo Focal: A técnica de Grupo Focal foi selecionada por ter um baixo custo e um alto valor de retorno em informações sobre os usuários. Embora envolva encontros presenciais, é possível adaptá-la para ser realizada de maneira remota, especialmente quando os colaboradores de uma empresa trabalham remotamente, que é o caso do projeto Groovoo. É importante garantir a confidencialidade das informações compartilhadas durante o grupo focal remoto. Os participantes devem ser informados sobre a natureza confidencial da discussão e sua permissão para uso e análise dos dados coletados.

- **Ideação:** Nesta etapa, o Time de Produto junto ao Time de *Design* irão analisar os dados coletados na etapa anterior e transformá-los em requisitos funcionais que devem ser elicitados para que o produto atenda as necessidades dos usuários.
 - *Card Sorting*, ou Classificação de Cartas: Embora a técnica de Classificação de Cartas tenha um alto custo, ela foi selecionada por ser um método eficaz na elicitação de requisitos em projetos com muitas regras de negócio estruturadas. Que é o caso do Groovoo, um projeto dividido em três aplicações, que além de ser uma plataforma de venda de ingressos, enfrenta questões sobre como manter a segurança sem dificultar o acesso dos usuários, sejam eles imigrantes com documentação irregular, ou adultos de terceira idade que não são familiarizados com tecnologias atuais.

- **Prototipagem:** Nesta etapa, o Time de *Design* irá implementar os dados coletados em etapas anteriores por meio de protótipos, proporcionando um melhor entendimento das necessidades do usuário e validando as funcionalidades que se encaixem no projeto.
 - *Design Participativo*: Embora a técnica de *Design Participativo* recomende a participação de terceiros, ela foi selecionada, pois no Projeto Groovoo o Time de Produto tem contato direto com clientes, facilitando essa iteração e diminuindo a lacuna. Além disso, essa técnica fornece uma visão geral sobre as necessidades dos usuários por um baixo custo em projetos complexos. Por fim, ao conduzir usuários reais a etapa de prototipagem, o Time de *Design*, já com as informações coletadas anteriormente, conseguirá validar as necessidades dos usuários transformando-as em protótipos reais.

- **Testes:** Por fim, nesta etapa o Time de Produto, Time de *Design* e usuários reais se alinham para realizar uma avaliação integral do produto gerado nas etapas anteriores, colhendo *feedbacks* e entendendo quais mudanças devem ser realizadas para melhorar a experiência do usuário.
 - *Thinking Aloud*, ou Pensar em voz alta: Embora também seja tradicionalmente realizada de maneira presencial, a técnica de *Thinking Aloud* pode ser adaptada para o cenário remoto que os colaboradores do Projeto Groovoo se encontram. Assim como na técnica de Grupo Focal, os participantes devem ser encorajados pelos facilitadores a expressar opiniões na realização de determinadas tarefas, expressando sentimentos, dificuldades e melhorias. Ela foi selecionada por ser oferecer um alto retorno por um baixo custo financeiro em grandes projetos que se encontram em etapas de redesenho ou melhoria.

O Capítulo 2 aborda técnicas levantadas por autores, porém ao analisar o cenário e limitações da *Startup* Atena e do Projeto Groovoo, as técnicas propostas foram selecionadas por oferecer um alto valor de retorno por um baixo custo financeiro e de aplicação. As técnicas selecionadas compõem as quatro etapas propostas por Norman (2013) e a literatura sugere que após a finalização da etapa de Testes, o Time de Produto deve retornar à etapa de Observação, possibilitando o refinamento do produto até que os resultados da avaliação sejam satisfatórios.

Na Tabela 10 apresenta-se o conjunto de técnicas selecionadas para o Processo da empresa, assim como uma comparação entre elas. Algumas, os autores denominam métodos em detrimento de técnicas.

Tabela 10 – Análise comparativa das técnicas do UCD selecionadas.

Técnica/ Método	Grupo Focal	Classificação de Cartas	Design Participativo	Pensar em voz alta
Etapa	Observação	<i>Design</i> (Ideação)	Prototipagem	Testes
Vantagem	Possibilita reunir informações detalhadas sobre um domínio específico	Fornecer conhecimentos sobre o modo de pensar dos usuários	Fornecer uma visão geral sobre as necessidades e desejos dos usuários	Identificar os sentimentos dos usuários ao interagirem com alguma tela
Desvantagem	Participantes podem não expressar honestamente suas opiniões sobre o assunto	É difícil realizar a classificação de muitos cartões	Necessita da participação de terceiros	Necessita de participação de terceiros
Custo	Baixo	Alto	Baixo	Baixo
Onde usar	Qualquer projeto	Projetos com muitas informações estruturadas	Projetos Complexos	Redesign de projetos ou melhorias
Onde não usar	-	Projetos com pouca informação estruturada	Projetos Pequenos	Projetos novos

Fonte: Os autores

5.2.3 Metodologia Selecionada

Como a empresa já utiliza em seu processo de desenvolvimento de *Software* uma metodologia Ágil, e em acordo com a seleção de Argumanis, Moquillaza e Paz (2021) abordada no Capítulo 2, a metodologia selecionada para a aplicação do UCD ao atual

processo de desenvolvimento de *software* do Projeto Groovoo foi o *Design em Paralelo ao Sprint*.

Essa metodologia foi selecionada dada a facilidade do planejamento de funcionalidades em redesenho, possibilitando que o Time de Desenvolvimento trabalhe em paralelo ao Time de Produto e Time de Design, enquanto eles estão iterando sobre quatro etapas do UCD e aplicando as técnicas propostas.

A metodologia selecionada permite a detecção de falhas de *design* com antecedência, evitando gastos com manutenção.

5.3 Identificação de oportunidades de inserção

Esta seção apresenta os procedimentos e etapas adotadas para a produção do 2.º objetivo específico:

- *Identificar no processo de desenvolvimento de software Groovoo oportunidades de inserção de diretrizes do UCD.*

A união do *framework* ágil *Scrumban* às diretrizes do *Design Centrado no Usuário - UCD* é uma tarefa que exige ponderada adaptação da equipe, dos processos e das entregas. Em um processo ágil de desenvolvimento de *software*, onde o produto é diferente após cada *sprint*, construir um ciclo de *feedback* é fundamental para o sucesso do produto, necessitando não apenas de conhecimentos técnicos dos colaboradores, mas também de adaptações cognitivas e trabalho em equipe.

Devido às variações entre as abordagens de cada equipe, descobrir o método adequado para o seu time pode envolver um processo de teste e adaptação. A autora [Littman \(2014\)](#) descreve algumas boas práticas que podem ser incorporadas ao time de desenvolvimento, para facilitar a adaptação dos colaboradores, à incorporação do UCD a um *framework* ágil.

- Construa uma equipe de pessoas colaborativas e multifuncionais;
- Traga pesquisadores e designers de UX para a equipe desde o início do projeto;
- Adote uma abordagem colaborativa, não isolada, para fazer o trabalho;
- Encoraje a experimentação;
- Encontre o ritmo de *sprint* ideal para sua equipe;
- Certifique-se de que as atividades de experiência do usuário sejam focadas e acionáveis;

- Teste com qualidade apropriada os materiais de estudo obtidos pela experiência do usuário.

5.3.1 Inserção Scrumban

Foram analisadas as técnicas selecionadas do Scrumban e como inserí-las no Processo Groovoo.

5.3.1.1 *Ready Queue*

Ao examinar o fluxo de trabalho empregado no projeto Groovoo, observa-se que a presença de tarefas de alta urgência durante o desenvolvimento da *sprint* torna desafiadora a priorização e organização das demandas. É nesse contexto que surge a possibilidade de aplicar a técnica da *Ready Queue*.

Essa coluna deve ser posicionada entre o *Backlog* e a seção *In Progress*. Durante o decorrer da *sprint*, caso novas demandas sejam recebidas, elas serão priorizadas inicialmente enquanto os desenvolvedores estão em progresso com suas tarefas. Após essa priorização, será determinado se a tarefa é urgente ou não. Se for considerada de alta prioridade, será inserida na *Ready Queue*. Assim que um colaborador concluir sua tarefa e o limite de WIP for liberado, a atividade será movida para a seção *In Progress*, independentemente do estágio em que a *sprint* se encontra.

5.3.1.2 Limite de Trabalho em Andamento (*Work-in-Progress - WIP*)

O processo de desenvolvimento do projeto deve possuir um controle da quantidade de tarefas que seja diretamente proporcional à quantidade de colaboradores de forma que não aconteça sobrecarga por parte dos desenvolvedores e nem entregas de baixa qualidade que não atendam as expectativas do cliente.

O limite de WIP deve ser determinado em conjunto com a equipe levando em consideração o tamanho do time de desenvolvimento, recomenda-se que cada membro da equipe tenha no máximo uma tarefa por *sprint* e caso necessário apenas um integrante da equipe poderá ter duas tarefas.

No projeto Groovoo, este limite de trabalho em andamento é essencial para um desenvolvimento mais focado e controlado, uma vez que tanto a parte dos desenvolvedores quanto a do cliente é levado em consideração, tais práticas agregam para um sistema de gestão mais preciso e controlado.

5.3.1.3 Políticas de Trabalho Explícitas

Esta técnica pode ser aplicada ao longo de todo o projeto, mas recomenda-se que seja no começo de cada projeto ou *feature*, ela consiste em uma conversa facilitadora do

desenvolvimento através da determinação de algumas regras e instruções bem definidas, como:

- **Plano de Comunicação:** A fim de facilitar a comunicação efetiva, é essencial estabelecer quais ferramentas serão empregadas para o contato, determinar os horários de disponibilidade de cada membro da equipe e definir as permissões de acesso entre os membros. Além disso, é importante identificar quais situações cada pessoa é responsável por atender.
- **Regras de convivência:** Normalmente já é estabelecida pela própria cultura da empresa e a forma profissional de se lidar com as pessoas.
- **Solução de gargalos com *Swarming*:** A implementação da técnica *Swarming* envolve a colaboração da equipe de desenvolvimento, na qual, caso um desenvolvedor enfrente algum impedimento, os outros membros da equipe devem auxiliá-lo ao invés de iniciar uma nova tarefa, a menos que haja alguma tarefa de alta prioridade na *Ready Queue*.
- **Definição do que é Pronto:** A equipe deve estabelecer critérios para determinar quando uma tarefa está concluída, por exemplo, somente considerando-a pronta após validação do PO (*Product Owner*) ou quando o cliente realiza testes e fornece um parecer positivo.

Estas medidas descritas acima, devem ser tomadas em conjunto com a equipe do projeto Groovoo em uma reunião dedicada somente ao estabelecimento destas regras, e deve ser disponibilizado em um local de fácil acesso aos membros como o quadro Kanban, pois é necessário que elas estejam a todo momento sendo lembradas pelo time.

5.3.2 Inserção das Diretrizes do UCD

Foram analisadas as técnicas selecionadas do UCD e como inserí-las no Processo Groovoo.

5.3.2.1 Grupo Focal

O processo de desenvolvimento de *software* deve incorporar técnica de Grupo Focal antes da definição de cada História de Usuário, para que os dados coletados sejam incorporados a etapa de Ideação e os critérios de aceitação sejam identificados e validados por meio da identificação das necessidades do usuário.

Os curtos prazos do ágil recomendam que o Time de Produto deve realizar rápidas reuniões com uma amostra de usuários que representem as diferentes personas que usam o produto. Caso a amostra de usuários seja grande, o Time de Produto deve dividir os

entrevistados em pequenos grupos e realizar as reuniões separadamente. Um plano de entrevista deve ser estruturado para facilitar a coleta de informações dos entrevistados, que deve envolver a imersão na temática por meio de uma conversa e um formulário com perguntas objetivas de múltipla escolha e de escala linear para facilitar a análise do Time de Produto com uso gráficos.

Os dados coletados devem ser suficientes para obter conhecimento de como uma funcionalidade será usada pelos diferentes tipos de usuários. Por terem uma limitação geográfica, os colaboradores devem realizar as reuniões com o Grupo Focal com a maior interação possível, com câmeras e áudios ligados, para obter uma melhor identificação das reações dos entrevistados.

5.3.2.2 *Carding Sorting* ou Classificação de Cartas

A técnica de Classificação de Cartas deve ser aplicada após a coleta de dados realizada na etapa de Observação em uma reunião online com o Time de Produto usando ferramentas que facilitem a visualização e definição de Épicas. É compreensível que essa etapa dure mais que a anterior, pois é aqui que o Time de Produto irá debater as regras de negócio que satisfaçam as necessidades dos usuários, atendendo limitações de escopo, equipe e prazos.

A abordagem deve ser realizada com um olhar centrado no usuário desde o começo. Utilizando a Classificação de Cartas como uma solução eficiente e acessível para o design, o Time de Produto do Projeto Groovoo obterá um Mínimo Produto Viável (MVP) que melhor atenda às exigências e desejos dos usuários, de forma ágil e econômica.

No entanto, na etapa de *Design*(Ideação) do protótipo é onde ocorrerá o aprimoramento necessário. Isso significa que o Card Sorting pode ser empregado em duas fases desse processo. Em outras palavras, é viável iniciar com o Card Sorting em uma abordagem aberta e, posteriormente, migrar para uma abordagem fechada à medida que o desenvolvimento do produto avança.

- **Card Sorting Aberto:** É utilizado quando é preciso entender quais categorias macro fazem mais sentido para as pessoas, considerando um conjunto de conteúdos.
- **Card Sorting Fechado:** Os cartões já têm conteúdo e categorias dados. Os participantes devem agrupar o conteúdo e eleger a categoria na qual mais se encaixam.

Desse modo, a técnica de Classificação de Cartas deve ser utilizada em toda funcionalidade que exija uma estruturação geral de informações, sugestões de navegação, menus e regras de negócio. O seu resultado será um insumo para chegar em uma solução mais completa e centrada no usuário, gerando Histórias de Usuários e seus critérios de aceitação.

5.3.2.3 Design Participativo

O *Design Participativo* deve ser empregado com o Time de Design a uma amostra significativa dos usuários do produto. Também realizada de maneira online com ferramentas que promovam a participação ativa dos colaboradores, essa técnica deve promover a co-criação e a tomada de decisões conjuntas. Ao seguir essas diretrizes, mesmo com uma equipe limitada, o Time de Design poderá envolver efetivamente o cliente no processo de prototipação e obter resultados mais alinhados com suas necessidades.

Os usuários participantes devem ser estimulados a expressarem seus sentimentos em relação a como os critérios de aceitação das Histórias de usuários podem ser expressos por meio de protótipos, de formas variadas, como:

- **Política e Sociológica:** No contexto do desenvolvimento de produtos de design, é proposto que indivíduos, especialmente aqueles com acesso limitado a oportunidades, sejam encorajados a participar ativamente e expressar seus sentimentos. É recomendado que equipes de design adotem abordagens inclusivas, como a realização de pesquisas e entrevistas, a condução de workshops participativos e a promoção de canais abertos de comunicação. Essas estratégias visam assegurar a inclusão de todas as perspectivas durante o processo de design. Ao estimular a expressão desses sentimentos, busca-se garantir que as soluções desenvolvidas sejam verdadeiramente representativas, inclusivas e alinhadas com as necessidades e desejos dos envolvidos em seu respectivo contexto.
- **Geográfica e Contextual:** É fundamental estimular ativamente a expressão dos membros da população local, considerando suas diversas culturas e experiências prévias. Reconhecendo que essas perspectivas podem enriquecer o projeto, é recomendado que o Time de design crie um ambiente inclusivo e receptivo, onde as pessoas se sintam encorajadas a compartilhar seus sentimentos, conhecimentos e ideias, é o caso dos usuários que podem se sentir coagidos por estarem com documentação irregular. Isso pode ser alcançado por meio de estratégias como a realização de grupos de discussão, sessões de brainstorming colaborativas e atividades participativas. Ao valorizar e incorporar as contribuições da população local, o Time de design promove a co-criação de soluções mais relevantes, culturalmente sensíveis e significativas para o contexto que se encontra o Projeto Groovoo.
- **Prática:** No contexto prático e usual, o Time de design deve promover ativamente a participação das pessoas, levando em conta suas diversas experiências, conhecimentos prévios e suas ações. Reconhecendo que essas perspectivas podem trazer benefícios ao projeto, o Time de design poderá obter soluções mais robustas, relevantes e alinhadas com as necessidades e expectativas dos usuários.

5.3.2.4 *Thinking Aloud*, ou Pensar em voz alta

O Time de Produto junto ao Time de *Design* deve implementar o método de *Thinking Aloud* ao final do ciclo iterativo para coletar dados em testes de usabilidade. Não deve ser revelado aos participantes que eles estão sobre um processo de testes, essa palavra pode assustá-los influenciando que seu comportamento não seja fiel ao habitual.

Esse método pode ser utilizado tanto individualmente como em grupo, onde o Time de *Design* realiza uma coleta sistemática de declarações dos usuários e posteriormente realiza uma análise, de modo a obter possíveis problemas de usabilidade no produto.

Os facilitadores da reunião devem pedir aos participantes expressem livremente seus pensamentos durante a conclusão da tarefa, compartilhando o que estão observando, pensando, fazendo e sentindo. Isso permite que os observadores obtenham *insights* sobre os processos cognitivos dos participantes, indo além do resultado. O objetivo é tornar os processos de pensamento o mais explícitos possível durante o desempenho da tarefa, que pode ser dividida em duas etapas:

- **Coleta de Declarações:** Os participantes devem ser encorajados a realizar tarefas cotidianas com o produto, narrando em voz alta as ações realizadas, decisões tomadas, opiniões e sentimentos enquanto interagem com o protótipo.
- **Análise das Declarações:** O Time de Produto junto ao Time de *Design* deve analisar as declarações e verificar quais possíveis mudanças no protótipo são necessárias.

Após a finalização, caso seja necessário, o Time de Produto realiza mais ciclos de modo a aprimorar os dados coletados e prototipar um *design* mais fiel às necessidades dos usuários.

5.3.2.5 *Design* em Paralelo ao *Sprint*

Essa metodologia de desenvolvimento deve ser incorporado ao atual processo de desenvolvimento do Projeto Groovoo, pois possibilita que o Time de Desenvolvimento e o Time de *Design* trabalhem em duas trilhas paralelas, iterando o *design* e a implementação separados, mas de maneira simultânea. Desta forma, é possível verificar falhas de *design* com antecedência e alterar quantas vezes for necessário.

O atual modelo Kanban do processo será adaptado para a inserção de uma coluna definida como UCD, que será a coluna inicial do novo processo, antes da coluna de Backlog, e será responsável por conter as quatro etapas do modelo de *Design* Centrado no Usuário, aplicando as diretrizes selecionadas e validando-as com os usuários.

A metodologia será aplicada com a abordagem *Just in Time - JIT*, que ajuda no gerenciamento de processos evitando gargalos, produzindo a quantidade necessária para atender uma demanda. Dessa forma, quando o Time de Produto identificar um problema que precisa ser resolvido ou funcionalidade nova, eles serão responsáveis por explorar o novo problema, enquanto o Time de Desenvolvimento continuará com as demandas anteriores.

Por se tratar de uma abordagem de fácil aplicação com um grande valor e um baixo custo, ela se torna uma opção interessante no processo de trabalho do projeto Groovoo, uma vez que adiciona na metodologia ágil já utilizada uma forma prática de incluir o processo de *desing* e UX.

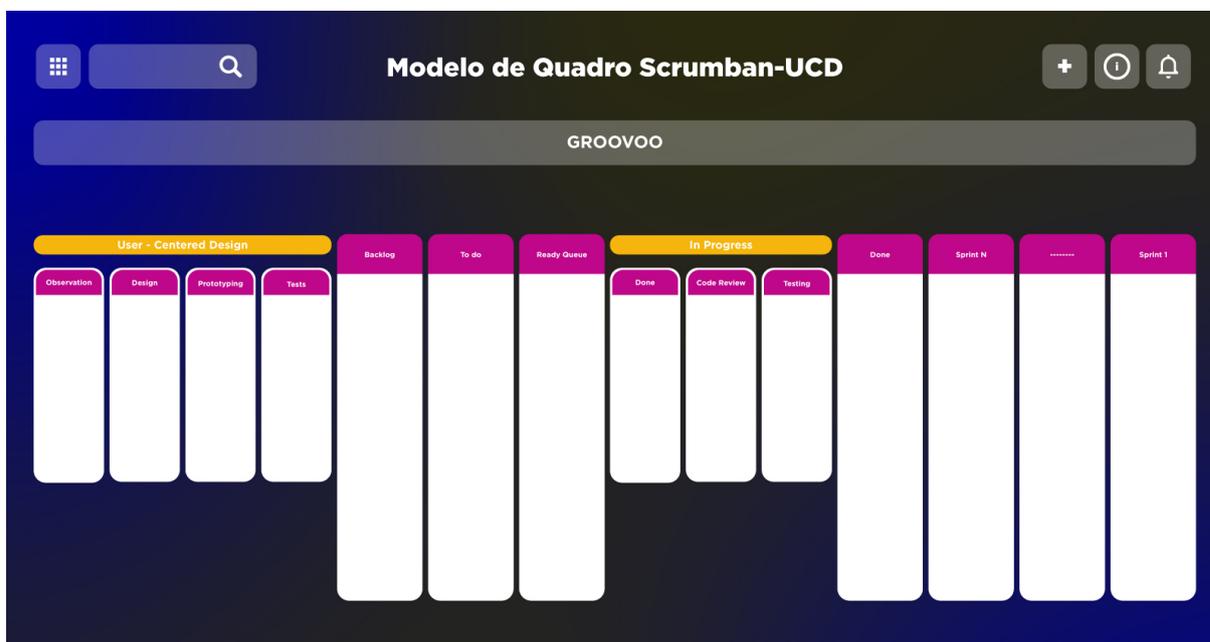
5.3.3 Proposta de adaptação do Processo

A proposta envolve adaptar o processo de Desenvolvimento de Software já existente com as diretrizes selecionadas do UCD e do modelo Kanban.

5.3.3.1 Quadro Scrumban-UCD

Na Figura 14 apresenta-se a adaptação do Quadro Kanban pela inserção das Diretrizes e Técnicas do Kanban selecionadas.

Figura 14 – Proposta do novo modelo Kanban



Fonte: Os autores

O *User-Centered Desgin* será responsável por iniciar o desenvolvimento de uma nova demanda. O Time de Produto vai aplicar as técnicas selecionadas nas 4 etapas de modo a compreender o cenário dos usuários em relação à inserção de uma nova funciona-

lidade. Por se tratar de uma inserção do UCD em um modelo de desenvolvimento ágil, o tempo dedicado a essa etapa pode variar entre funcionalidades, mas a proposta é que não se estenda por mais de uma semana, o que pode ser um desafio para os responsáveis de produto da empresa. Abaixo apresenta-se um resumo de como as etapas devem ser incorporadas.

- **Observação:** Para minimizar o tempo e aumentar a qualidade dos dados coletados, o Apêndice B demonstra um exemplo de formulário que pode ser aplicado nas entrevistas usando a Técnica de Grupo Focal.
- **Design (Ideação):** Nessa etapa o Time de Produto pode usar ferramentas que facilitem a visualização e organização de *cards* que auxiliem a definição de requisitos, como a plataforma Miro.
- **Prototipagem:** Nessa etapa deve ser realizada em uma reunião online entre Time de *Design* e a amostra de usuários que participou do levantamento de dados na etapa de Observação. O Time de *Design* deve estimular os entrevistados a expressar suas opiniões sobre como a funcionalidade pode ser implementada visualmente, e que será posteriormente aprimorada e levada a etapa de Testes.
- **Testes:** Nessa etapa o Time de *Design* e o Time de Produto devem verificar como o usuário se sente com a proposta implementada pelos estudos realizados e expor seus comentários. Para minimizar o tempo e aumentar a qualidade dos dados coletados, o Apêndice D demonstra um exemplo de fluxo e questionário que podem ser aplicados nas reuniões usando a Técnica de *Thinking Aloud*.

Após finalizado o ciclo iterativo do UCD, a funcionalidade é colocada na fileira de *Backlog* para ser priorizada e granularizada em tarefas técnicas com critérios de aceitação, e após, seguir para as demais fileiras do Quadro Kanban.

- **Backlog:** Nessa etapa ocorre primeiramente a especificação das histórias de usuários com a definição dos critérios de aceitação, esta fileira será incrementada conforme as novas demandas vão saindo do ciclo UCD, ou seja, ela é oposta ao *backlog* do *Scrum* que é fixo ao longo de todo o projeto. Esta seção é o momento em que deve ocorrer o contato do time de produto com os desenvolvedores.
- **To Do:** Nessa etapa é realizada uma reunião com o time de produto em que é definido as tarefas prioritárias do *backlog* para aquela *sprint*, desse modo as histórias de usuários são granularizadas em tarefas técnicas pelo *Tech Lead* e distribuídas entre a equipe de desenvolvimento e alocadas nessa seção enquanto não são iniciadas.

- **Ready Queue:** Esta seção será utilizada quando surgirem demandas de alta urgência durante o andamento da *sprint*, ou seja, os desenvolvedores estarão com suas respectivas tarefas em andamento e por isso estas solicitações ficarão alocadas na fileira até o limite de WIP ser liberado e ela ser movida para a seção *Doing*, esta etapa permite que a demanda não passe pelas seções anteriores e seja apenas especificada para o time de desenvolvimento.

A etapa *In Progress* possui 3 fileiras sendo elas: *Doing*, *Code Review* e *Testing*, todas as tarefas que estão nestas seções são contabilizadas para o limite de WIP, ou seja, se o limite é 3 e existe uma tarefa em cada uma delas então não poderá entrar nenhuma nova demanda nessa etapa, pois o limite já está no máximo.

- **Doing:** Nessa seção estarão alocados todas as tarefas que estão em desenvolvimento, ou seja, as demandas especificadas e priorizadas na seção *To Do* ou na *Ready Queue*.
- **Code Review:** Nesta seção estará todas as tarefas que saíram da etapa *Doing* e agora necessitam de uma revisão técnica realizada pelo *Teach Lead*, para possíveis ajustes de integração e padronização de código-fonte, caso necessite de ajustes a tarefa retorna para a etapa anterior.
- **Testing:** Nesta seção estará alocada todas as tarefas que estão prontas para serem testadas pelo PO (*Product Owner*), caso necessite de ajustes a tarefa retorna para a seção *Doing*, se não a demanda estará concluída e pronta para ser colocada em produção.

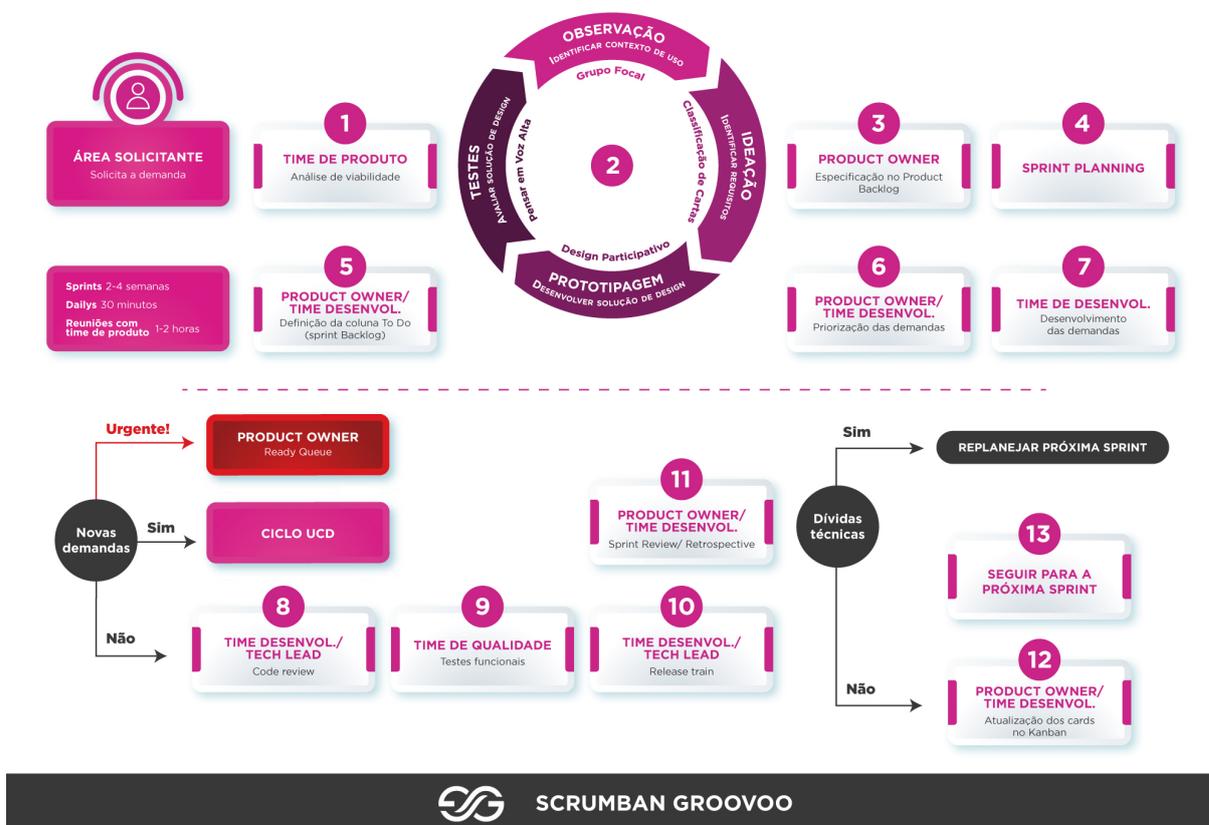
Após a finalização de Etapa *In Progress*, a funcionalidade chega ao estágio final do Quadro Kanban, onde estão contidas as seguintes etapas:

- **Done:** Responsável por conter todas as tarefas concluídas, com os critérios de pronto atendidos, na *sprint* atual.
- **Sprint - N:** Irá conter todo o histórico de tarefas concluídas em cada *sprint* para o acompanhamento da velocidade do time por meio de uma análise temporal.

5.3.3.2 Fluxograma Scrumban-UCD

Na Figura 15 apresenta-se uma visão ampla da proposta de adaptação do Fluxograma apresentado no Capítulo 4 para a inserção das Diretrizes UCD.

Figura 15 – Visão ampla do Modelo Scrumban-UCD



Fonte: Os autores

Ao surgir uma nova demanda da Área Solicitante, que pode estar relacionada a necessidades encontradas pelos *Stakeholders* ou *Shareholders*, ela é encaminhada ao Time de Produto que realiza a análise de viabilidade, verificando se ela realmente pode ser implementada analisando limitações de prazo, escopo e recursos.

Após a análise de viabilidade, a demanda é direcionada ao Ciclo Iterativo do UCD. Nessa Etapa o Time de Produto e o Time de *Design* serão os agentes responsáveis por identificar o contexto dos usuários, identificar requisitos, desenvolver soluções de *design* e implementar avaliações das soluções de *design*.

Após a validação, a demanda é direcionada ao *Product Owner* para ocorrer a especificação e categorização no *Product Backlog* do Quadro Kanban. Em seguida, ocorre a cerimônia de *Sprint Planning*, onde o Time de Produto e o Time de Desenvolvimento planejam a próxima sprint com as demandas contidas no *Product Backlog*, com duração aproximada de duas a quatro semanas, *dailys* de 30 minutos e reuniões dos desenvolvedores com o Time de Produto para esclarecimentos de dúvidas pontuais. Ainda na *Sprint Planning*, o *Product Owner* define quais tarefas serão implementadas no *sprint* atual, e realiza a categorização e priorização na coluna *To Do* do quadro Scrumban-UCD.

Em seguida o Time de Desenvolvimento entra na *sprint* com a implementação das demandas. Na hipótese de uma nova demanda no decorrer do *sprint*, o Time de Produto

deve ser responsável por identificar se essa demanda é urgente ou não. Caso ela tenha alta urgência, o P.O deve encaminhá-la a fileira *Ready Queue*, para poder ser desenvolvida sem passar pelas etapas anteriores. Caso não tenha alta urgência, a demanda é direcionada ao início do fluxograma onde deve ser submetida ao Ciclo UCD para validação juntos aos usuários.

Com o desenvolvimento das demandas finalizado, o *Tech Lead* deverá aplicar políticas de controle de qualidade de código com revisões de escrita e implementação. Em seguida, elas são encaminhadas ao Time de Qualidade que deve realizar testes funcionais para verificar a qualidade das implementações, validando a ocorrência da cerimônia de *Release Train* que adiciona as novas funcionalidades no ambiente de produção para estarem disponíveis para os usuários.

Por fim, o *Product Owner* junto ao Time de Desenvolvimento realizam a cerimônia de *Sprint Review/ Retrospective* que analisa como ocorreu a *sprint*, verificando ocorrências de dívidas técnicas e dificuldades. Após isso ocorre o planejamento da próxima *sprint* e atualização das demandas no quadro Kanban.

5.4 Avaliação, Validação e Consolidação de possibilidades de inserção

Esta seção apresenta os procedimentos e etapas adotadas para a produção do 3.º. objetivo específico.

- *Avaliar, validar e consolidar possibilidades de inserção de diretrizes UCD no Processo de desenvolvimento de Software do Groovoo.*

5.4.1 Entrevista Semi-estruturada

Uma única entrevista coletiva foi conduzida com todos os participantes do projeto Groovoo, a reunião teve um total de 2 horas e 30 minutos de duração, divididas em 3 momentos, sendo o primeiro uma discussão com a opinião e os sentimentos de cada colaborador acerca do atual processo de desenvolvimento da empresa, no segundo aconteceu a proposição do novo processo descrito na Seção 5.3.3 por meio de uma apresentação, e por fim, todos tiveram um espaço para expressar suas opiniões sobre a viabilidade do novo processo e sua eficiência quanto ao custo/benefício.

O time em sua maioria demonstrou descontentamento frente aos problemas enfrentados no atual processo:

- A falta de estudos iniciais ocasionaram problemas de comunicação;

- O Time de Produto não conseguia acompanhar se as demandas estavam atendendo as necessidades dos usuários;
- As cerimônias do *Scrum* adotadas não estavam surgindo efeito, uma vez que as constantes demandas de alta urgência apareciam com a *sprint* em andamento e os desenvolvedores tinham que parar suas atividades para atender estes pedidos sem especificações.
- As características da metodologia ágil eram deixadas para trás, ocasionando com que a equipe realizasse entregas que não atendiam as reais necessidades dos usuários, uma vez que, as demandas surgiam sem estudo prévio daqueles que utilizam o sistema diariamente.

A apresentação do novo processo para o time do projeto Groovoo, consistiu em seis etapas sendo elas:

- **Atual Processo de Desenvolvimento:** Foi mostrado de forma sucinta como estava ocorrendo o atual processo de desenvolvimento.
- **Oportunidades de Melhoria:** Foram apresentadas vulnerabilidades no atual processo, que por conta dos prazos de Desenvolvimento Ágil a busca da real compreensão das necessidades do usuário fica em segundo plano.
- **Design Centrado no Usuário - UCD:** Foi explicado o que é o UCD, sua importância e o seu impacto dentro do desenvolvimento do produto.
- **Etapas e Técnicas Selecionadas:** Foram apresentadas as técnicas do *Scrumban* e do UCD escolhidas para compor o novo processo, levando em consideração as limitações da empresa.
- **Oportunidades de Inserção:** Foi explicado como as técnicas selecionadas poderiam se encaixar conforme cada vulnerabilidade apresentada.
- **Proposta do novo Processo:** Por fim, foi apresentado o processo final, resultante da junção *Scrumban* da literatura, com o UCD da literatura, e o atual processo de desenvolvimento de software, ocasionando um novo quadro *Kanban* apresentado na Figura 10, e do novo *Workflow Scrumban-UCD* da Figura 11

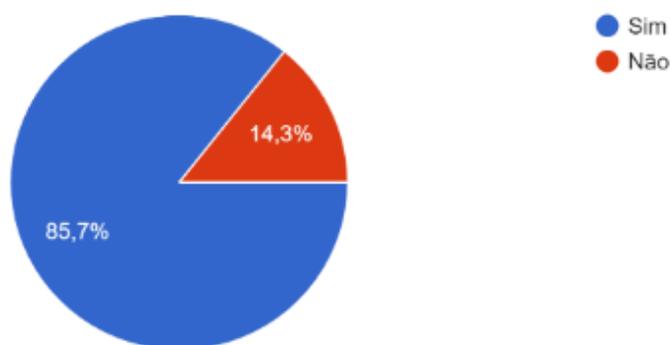
Por fim, por conta dos benefícios de estudos iniciais dos usuários, o time demonstrou entusiasmo ao novo processo de desenvolvimento de software e sua implementação.

5.4.2 Resultados

Os resultados foram analisados por respostas submetidas pelos participantes ao Formulário disponibilizado no Apêndice C e geraram os resultados encontrados nas Figuras 16 a 28. O modelo de avaliação envolve questões de escala linear, onde os usuários realizaram uma escala de prioridade enumerando a importância de uma técnica ou diretriz de 1 a 5, onde 1 é a mais baixa prioridade e 5 a mais alta:

Figura 16 – Resposta do Questionário

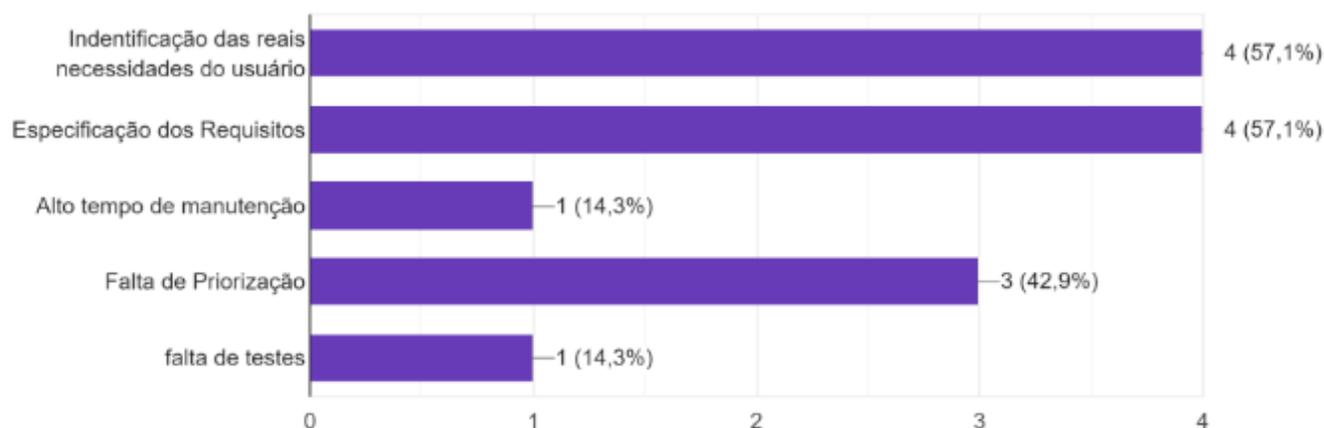
A organização está sensível a necessidade de atividades de reconhecimento da experiência do usuário?
7 respostas



Fonte: Os autores

Figura 17 – Resposta do Questionário

Quais as principais vulnerabilidades dentro do atual processo?
7 respostas

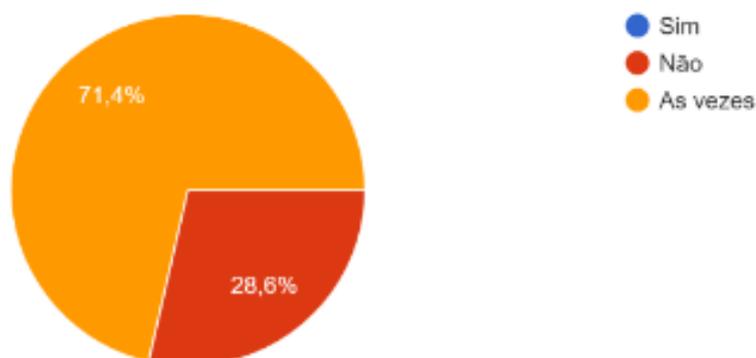


Fonte: Os autores

Figura 18 – Resposta do Questionário

Ao surgir uma nova demanda, a empresa tem estratégias de identificação e adaptação às necessidades do usuário?

7 respostas

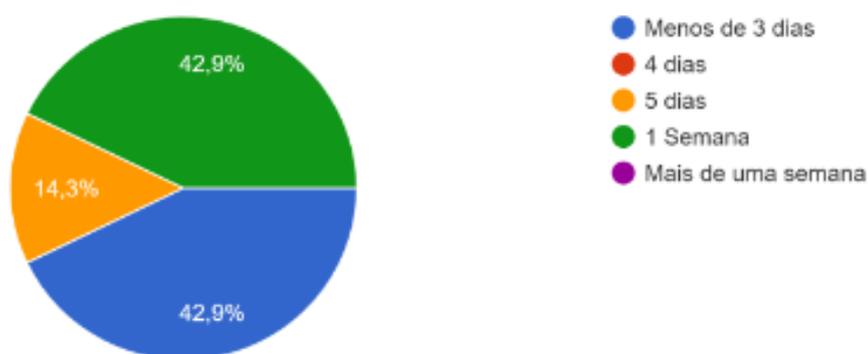


Fonte: Os autores

Figura 19 – Resposta do Questionário

Qual o tempo ideal para se dedicar a atividade de identificação das necessidades do usuário?

7 respostas

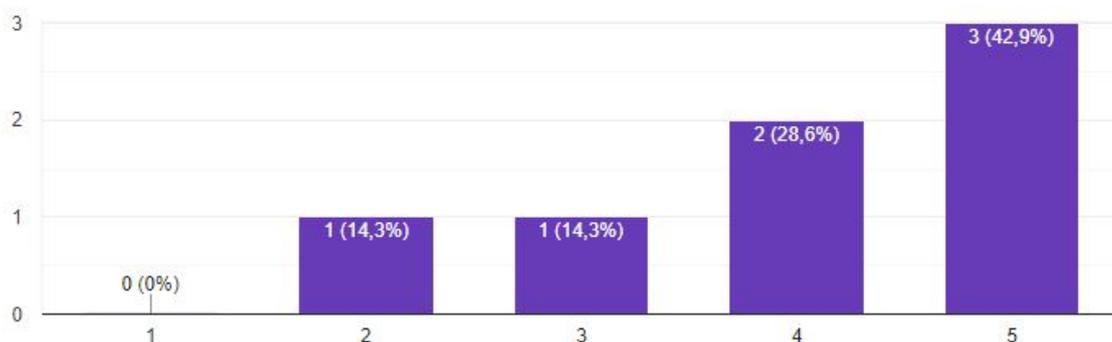


Fonte: Os autores

Figura 20 – Resposta do Questionário

Considerando as diretrizes de UCD da literatura e os métodos e as técnicas selecionadas. Enumere de 1 a 5 como você avalia a aplicação da Técnica de Grupo Focal para estudo dos cenários dos usuários?

7 respostas

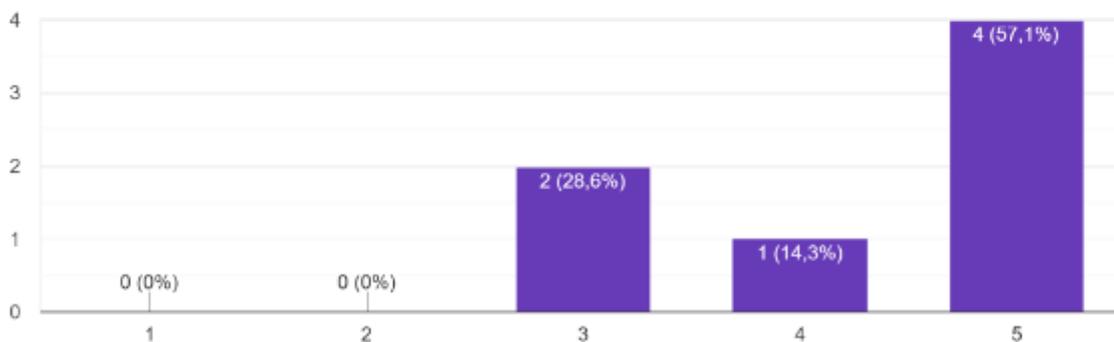


Fonte: Os autores

Figura 21 – Resposta do Questionário

Enumere de 1 a 5 como você avalia a aplicação da Técnica de Carding Sorting para elicitación de requisitos?

7 respostas

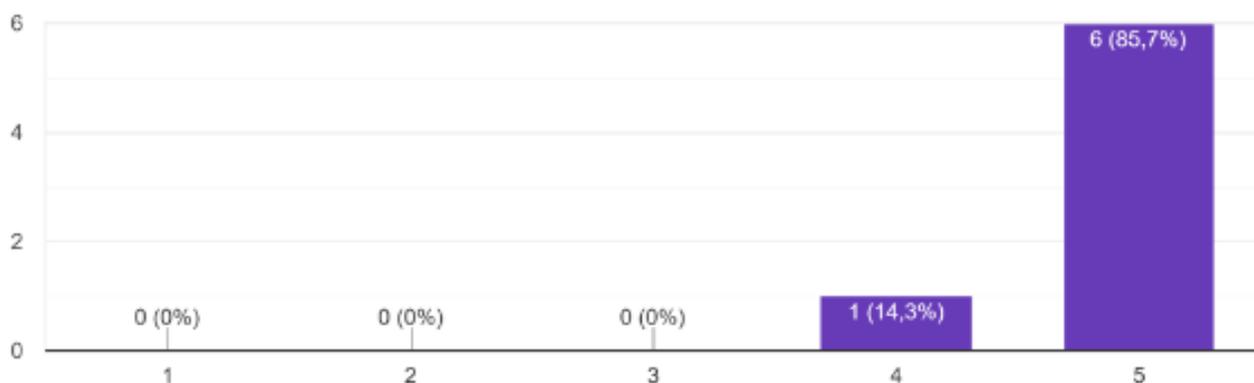


Fonte: Os autores

Figura 22 – Resposta do Questionário

Enumere de 1 a 5 como você avalia a aplicação da Técnica de Design Participativo para elicitacão de prototipacão?

7 respostas

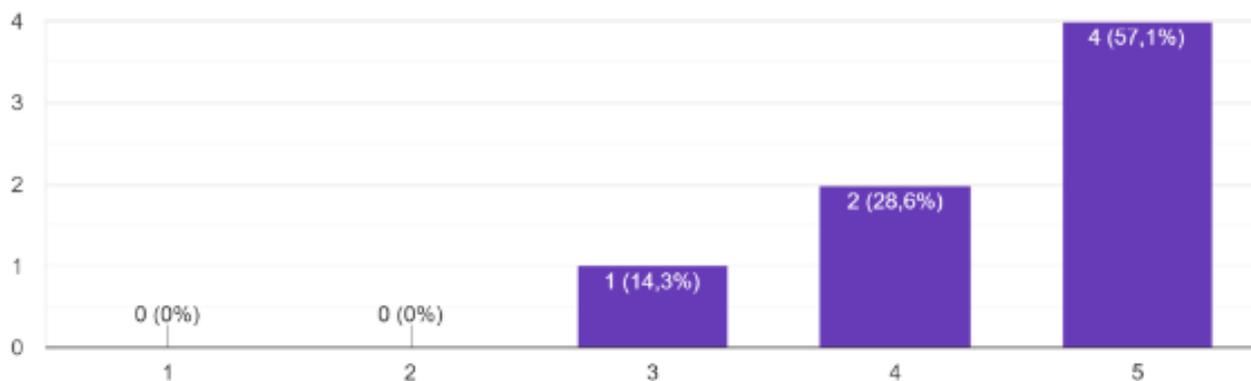


Fonte: Os autores

Figura 23 – Resposta do Questionário

Enumere de 1 a 5 como você avalia a aplicacão da Técnica de Thinking Aloud para validacão de protótipos?

7 respostas

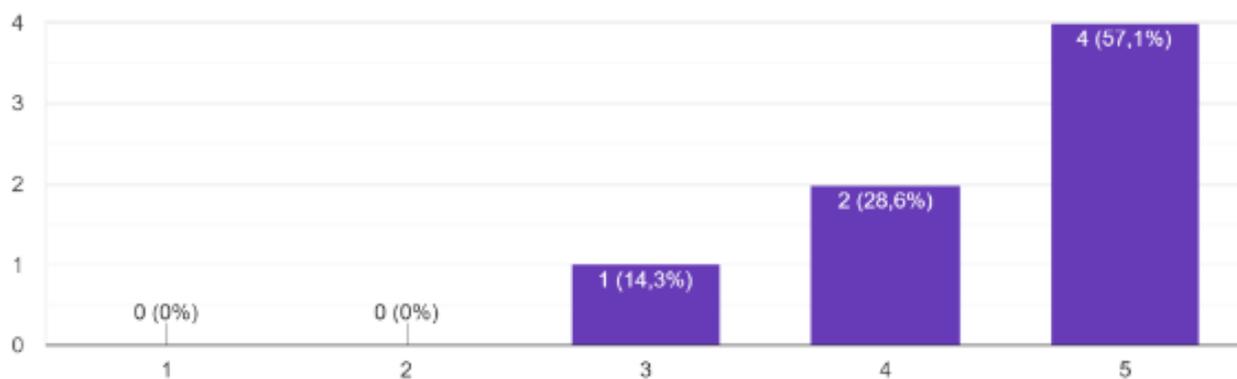


Fonte: Os autores

Figura 24 – Resposta do Questionário

Enumere de 1 a 5 como você avalia a aplicação da Técnica Ready Queue para priorização de tarefas?

7 respostas

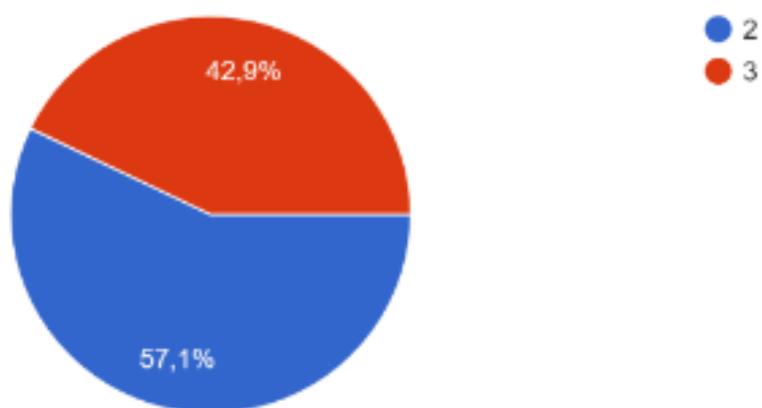


Fonte: Os autores

Figura 25 – Resposta do Questionário

Considerando a equipe atual, qual seria o Limite de WIP ideal para o processo?

7 respostas

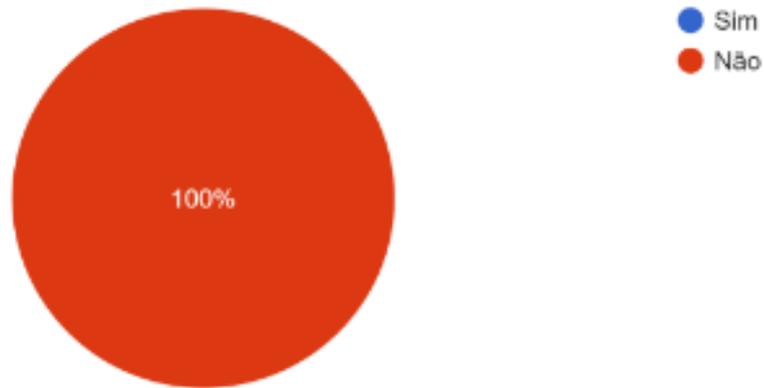


Fonte: Os autores

Figura 26 – Resposta do Questionário

A empresa tem Políticas de Trabalho explícitas?

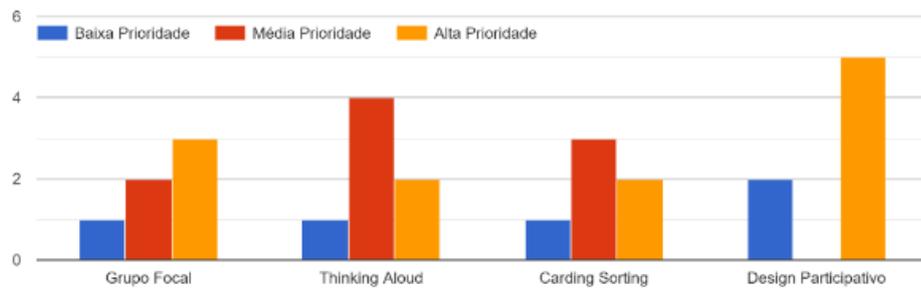
7 respostas



Fonte: Os autores

Figura 27 – Resposta do Questionário

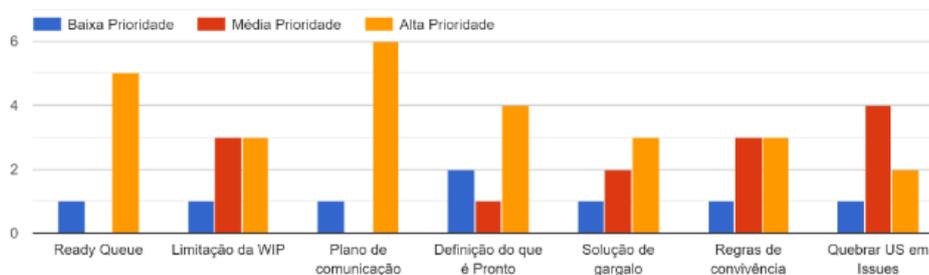
Entre as técnicas do UCD apresentadas, numa escala de prioridade, defina as técnicas mais prioritárias, considerando custo/benefício:



Fonte: Os autores

Figura 28 – Resposta do Questionário

Entre as técnicas do Scrumban apresentadas, numa escala de prioridade, defina as técnicas mais prioritárias, considerando custo/benefício:



Fonte: Os autores

Por meio dos resultados obtidos, é possível concluir que a empresa está aberta à mudança para reconhecer as necessidades dos usuários conforme a Figura 16, e tem como vulnerabilidade principal a falta desta prática como mostram as Figura 17 e 18. E a indecisão quanto ao tempo para reconhecer essas necessidades elucidadas na Figura 19 deixam claro esta dificuldade do time.

As Figuras 20 a 24, demonstram que a equipe reconhece as técnicas selecionadas como práticas importantes ao desenvolvimento.

Porém, algumas técnicas foram colocadas como mais prioritárias do que outras, as Figura 27 e 28 apresentam o *Card Sorting*, *Thinking Aloud* e Quebrar US (*User Story*) em *Issues*, como técnicas importantes, mas de média prioridade enquanto todas as outras foram consideradas de alta.

A limitação de WIP e as Regras de convivência ficaram com a mesma quantidade de votos para média e alta prioridade.

Os votos em alta prioridade para as técnicas de Políticas de Trabalho Explícitas podem ser explicadas pela falta da mesma representada em unanimidade na Figura 26.

5.5 Considerações Finais do Capítulo

Neste capítulo foram abordadas as diretrizes selecionadas do *Design Centrado no Usuário* - UCD e do Modelo Kanban da literatura. A partir desse estudo, a proposta de incorporação ao atual processo de desenvolvimento do *software* do Projeto Groovoo foi gerada. Por fim, foram demonstrados dados gerados da entrevista semi-estruturada entre os autores e os colaboradores da empresa, no qual os resultados demonstraram sensibilidade para a inserção de diretrizes de UCD, em vista as fragilidades em relação a estudos iniciais sobre o contexto dos usuários no atual processo de desenvolvimento.

6 Conclusão

Este trabalho possibilitou entender as fragilidades do Processo de Desenvolvimento de Software Ágil do Projeto Groovoo, enquanto as necessidades de estudos iniciais sobre a experiência do usuário. Com isso, foram estabelecidas proposta de inserção diretrizes de UCD para o processo ágil Scrumban, com o intuito de aprimorar a experiência do usuário e promover a criação de um produto de qualidade que atenda às necessidades e expectativas dos usuários.

Para compreensão desse cenário, definiram-se três objetivos específicos. O primeiro, de analisar e selecionar diretrizes empregadas pelo *framework Scrum-UCD* em relação ao *Scrumban* e ao UCD, demandou um estudo das técnicas e metodologias propostas pela literatura, em relação as suas aplicabilidades técnicas no processo de desenvolvimento, tendo em vista o cenário e as limitações da empresa.

Após restringir a amostra de técnicas e metodologias propostas pela literatura, foi identificado no Processo de Desenvolvimento de Software Groovoo oportunidades de inserção. Com isso, o processo *Scrumban-UCD*, adaptado do processo antigo a inserção de diretrizes do UCD, foi desenvolvido e proposto. O procedimento atendeu ao segundo objetivo específico.

Desse modo, foram realizadas análises aprofundadas do contexto do projeto, da *Startup Atena Solutions*, e do processo de desenvolvimento de software adotado pela equipe. Por meio de reuniões com *stakeholders* e *shareholders*, foram identificadas lacunas e oportunidades para a aplicação de princípios, metodologias e técnicas de UCD, visando aprimorar a usabilidade e a experiência do usuário. A validação da proposta do novo processo com o Time de produto atendeu ao terceiro objetivo específico: avaliar, validar e consolidar possibilidades de inserção de diretrizes UCD no Processo de desenvolvimento de Software do Groovoo.

A aplicação das diretrizes de UCD propostas neste trabalho para o processo de desenvolvimento de software, representa uma valiosa contribuição para a melhoria da usabilidade e da experiência do usuário. A adaptação do modelo de processo Scrumban, integrando práticas ágeis com elementos de UCD, permitiu propor um processo que fosse ágil enquanto se preocupa com a experiência do usuário.

Em pesquisas futuras, pode-se implementar o processo proposto para a coleta de dados em relação à melhoria da aceitação do usuário em relação às funcionalidades implementadas. Esse estudo poderia envolver a aplicação de métricas e métodos de avaliação, como questionários de satisfação do usuário, análise de métricas de uso e feedback contínuo dos usuários. Essa abordagem permitiria a obtenção de dados quantitativos e qualitativos

sobre a aceitação e a usabilidade das funcionalidades implementadas, fornecendo informações valiosas para orientar melhorias iterativas no produto e atender ainda mais às necessidades e expectativas dos usuários.

Referências

- ALGER, L. *The Upcoming Ticket-Gating App Issuing NFTs as Event Tickets*. 2022. Disponível em: <<https://nftlately.com/the-upcoming-ticket-gating-app-issuing-nfts-as-event-tickets/>>. Citado na página 15.
- ALMUGHARAM, O.; ALYAHYA, S. Coordination support for integrating user centered design in distributed agile projects. In: *2017 IEEE 15th International Conference on Software Engineering Research, Management and Applications (SERA)*. [S.l.: s.n.], 2017. p. 229–238. Citado 4 vezes nas páginas 14, 28, 34 e 35.
- ALQUDAH, M.; RAZALI, R. An Empirical Study of Scrumban Formation based on the Selection of Scrum and Kanban Practices. *International Journal on Advanced Science, Engineering and Information Technology*, v. 8, n. 6, p. 2315–2322, dez. 2018. ISSN 2460-6952. Number: 6. Disponível em: <<http://www.insightsociety.org/ojaseit/index.php/ijaseit/article/view/6566>>. Citado na página 14.
- ALYAHYA, S.; IVINS, W.; GRAY, W. A holistic approach to developing a progress tracking system for distributed agile teams. *Proceedings - 2012 IEEE/ACIS 11th International Conference on Computer and Information Science, ICIS 2012*, 05 2012. Citado na página 34.
- ARGUMANIS, D.; MOQUILLAZA, A.; PAZ, F. A Framework Based on UCD and Scrum for the Software Development Process. In: SOARES, M. M.; ROSENZWEIG, E.; MARCUS, A. (Ed.). *Design, User Experience, and Usability: UX Research and Design*. Cham: Springer International Publishing, 2021. (Lecture Notes in Computer Science), p. 15–33. ISBN 978-3-030-78221-4. Citado 16 vezes nas páginas 14, 26, 31, 35, 36, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47 e 63.
- BAXEVANIS, A. *User-centred design: 6 popular UCD methods*. 2006. Section: Blog. Disponível em: <<https://inviqa.com/blog/user-centred-design-6-popular-ucd-methods>>. Citado 6 vezes nas páginas 29, 30, 31, 39, 40 e 41.
- BEYER, H.; HOLTZBLATT, K.; BAKER, L. *An Agile Customer-Centered Method: Rapid Contextual Design*. Springer, Berlin, Heidelberg, p. 50–59, 2004. Citado na página 34.
- BLUESTONE, D. *Overcoming the Agile/UCD Divide | UX Booth*. 2015. Disponível em: <<https://www.uxbooth.com/articles/overcoming-the-agile-ux-divide/>>. Citado na página 34.
- BRERETON, P. et al. Using a protocol template for case study planning. In: *Proceedings of the 12th international conference on Evaluation and Assessment in Software Engineering*. Swindon, GBR: BCS Learning & Development Ltd., 2008. (EASE'08), p. 41–48. Citado 3 vezes nas páginas 17, 48 e 49.

- BROWNE, C. *What Are User Flows? A Beginner's Guide to UX Design*. 2023. <<https://careerfoundry.com/en/blog/ux-design/what-are-user-flows/>>. Acessado em 3 de julho de 2023. Citado na página 100.
- CAGLEY, T. *What Makes Scrumban, Scrumban?* 2013. Disponível em: <<https://tcagley.wordpress.com/2013/09/24/what-makes-scrumban-scrumban-daily-process-thoughts/>>. Citado na página 20.
- CECIL, R. F. WebSite, *Clash of the Titans: Agile and UCD*. 2006. Disponível em: <<https://www.uxmatters.com/mt/archives/2006/12/clash-of-the-titans-agile-and-ucd.php>>. Citado na página 14.
- CLINE, A. *Agile Development in the Real World*. [S.l.: s.n.], 2015. ISBN 978-1-4842-1678-1. Citado na página 35.
- COOPER, A.; REIMAN, R. About face 2.0: The essentials of interaction design. *Wiley Publishing*, 2003. Citado na página 27.
- FUIOR, F. Key elements for the success of the most popular Agile methods. *Revista Română de Informatică și Automatică*, v. 29, p. 7–16, dez. 2019. Citado na página 13.
- FUNDATION, T. I. D. *What is User Centered Design?* 2019. Disponível em: <<https://www.interaction-design.org/literature/topics/user-centered-design>>. Citado na página 32.
- FUNDATION, T. I. D. *What is Design Thinking?* 2023. Disponível em: <<https://www.interaction-design.org/literature/topics/design-thinking>>. Citado na página 36.
- GOEL, G.; TANWAR, P.; SHARMA, S. UI-UX Design Using User Centred Design (UCD) Method. *12th International Conference on Computer Communication and Informatics, ICCCI 2022*, 2022. Disponível em: <<https://doi.org/10.1109/ICCCI54379.2022.9740997>>. Citado 2 vezes nas páginas 33 e 34.
- GUIMARÃES, F. M. *O que é Design Centrado no Usuário?* 2019. Disponível em: <<https://medium.com/aela/o-que-%C3%A9-design-centrado-no-usu%C3%A1rio-11a9c13c3a2f>>. Citado 2 vezes nas páginas 27 e 29.
- GULLIKSEN, J. et al. Key principles for user-centred systems design. *Behaviour & Information Technology*, v. 22, n. 6, p. 397–409, nov. 2003. ISSN 0144-929X. Publisher: Taylor & Francis _eprint: <https://doi.org/10.1080/01449290310001624329>. Disponível em: <<https://doi.org/10.1080/01449290310001624329>>. Citado na página 14.
- HASSENZAHN, M. User Experience-Towards a unified view. *UX WS NordiCHI*, jan. 2006. Disponível em: <https://www.academia.edu/2880260/User_Experience_Towards_a_unified_view>. Citado na página 32.
- HOGAN, A. et al. *The Six Steps For Justifying Better UX*. 2016. Disponível em: <<https://www.forrester.com/report/the-six-steps-for-justifying-better-ux/RES117708>>. Citado na página 37.
- HUSSEIN, I.; MAHMUD, M.; TAP, A. O. User Experience Design (UXD). 2019. Citado na página 32.

- ISO 13407. *Human-centred design processes for interactive systems*. Geneva, CH, 1999. Disponível em: <<https://www.sis.se/api/document/preview/615069/>>. Citado na página 26.
- JUSTINMIND. *UCD vs UX: What's the difference?* 2018. Disponível em: <<https://uxplanet.org/ucd-vs-ux-whats-the-difference-255443efa5f>>. Citado na página 14.
- KHAN, Z. A. *Scrumban - Adaptive Agile Development Process*. 2014. Citado 6 vezes nas páginas 19, 20, 21, 22, 23 e 24.
- KNIBERG, H. *Kanban vs. scrum: How to make the most of both*. 2009. Citado na página 20.
- LADAS, C. *crumban: Essays On Kanban Systems For Lean Software Development*. [S.l.: s.n.], 2008. Citado 3 vezes nas páginas 21, 23 e 25.
- LEONHARDT, M. *About 12 percent of people buying concert tickets get scammed*. 2018. Disponível em: <<https://www.cnn.com/2018/09/13/about-12-percent-of-people-buying-concert-tickets-get-scammed-.html>>. Citado na página 15.
- LITTMAN, W. *Marrying User-Centered Design with the Agile Software Development Process: 7 Tips for Success*. 2014. Section: UX Guidelines. Disponível em: <<https://usabilitygeek.com/marrying-user-centered-design-agile-software-development-process-tips-success/>>. Citado na página 64.
- LOWDERMILK, T. *User-Centered Design*. 2013. Citado na página 33.
- MARCUS, A.; GUNTHER, R.; SIEFFERT, R. Validating a standardized usability/user-experience maturity model: A progress report. In: . [S.l.: s.n.], 2009. p. 104–109. ISBN 978-3-642-02805-2. Citado na página 32.
- MIRCEA, E. Project Management using Agile Frameworks. *ECONOMY INFORMATICS JOURNAL*, v. 19, p. 34–43, set. 2019. Citado na página 13.
- NIELSEN, J.; MOLICH, R. Heuristic evaluation of user interfaces. In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. New York, NY, USA: Association for Computing Machinery, 1990. (CHI '90), p. 249–256. ISBN 978-0-201-50932-8. Disponível em: <<https://doi.org/10.1145/97243.97281>>. Citado na página 30.
- NIELSEN, L. Engaging personas and narrative scenarios. *Vol 17, PhD Series. Copenhagen: Samfundslitteratur*, 2004. Citado na página 27.
- NIKITINA, N.; KAJKO-MATTSSON, M.; STRALE, M. From Scrum to Scrumban: a case study of a process transition. *2012 International Conference on Software and System Process, ICSSP 2012 - Proceedings*, jun. 2012. Citado na página 13.
- NILSEN, J. Thinking Aloud: The #1 Usability Tool. In: . [s.n.], 2012. Disponível em: <<https://www.nngroup.com/articles/thinking-aloud-the-1-usability-tool/>>. Citado 2 vezes nas páginas 30 e 31.

- NORMAN, D. A. *The design of everyday things*. [S.l.: s.n.], 2013. Citado 8 vezes nas páginas 26, 28, 31, 32, 33, 38, 61 e 63.
- NORMAN, D. A.; DRAPER, S. W. *User centered system design : new perspectives on human-computer interaction*. [S.l.: s.n.], 1986. Citado na página 25.
- NOVOSELTSEVA, E. *User-Centered Design: An Introduction*. 2017. Section: Terminology. Disponível em: <<https://usabilitygeek.com/user-centered-design-introduction/>>. Citado na página 26.
- NURMULIANI, N.; ZOWGHI, D.; WILLIAMS, S. Using card sorting technique to classify requirements change. *Proceedings of the IEEE International Conference on Requirements Engineering*, p. 240– 248, 10 2004. Citado 2 vezes nas páginas 27 e 28.
- OGLE, J. Design Spiking. In: . [s.n.], 2014. Disponível em: <<https://thoughtbot.com/blog/design-spiking>>. Citado na página 35.
- PAHUJA, S. *What is Scrumban?* 2012. Disponível em: <<http://www.solutionsiq.com/resources/agileiq-blog/bid/87799/What-is-Scrumban>>. Citado na página 20.
- PATTERSON, M. et al. Towards an Agile User Experience Virtual Assistant and Management Platform. p. 1, jul. 2017. Publisher: BCS Learning & Development. Disponível em: <<https://www.scienceopen.com/hosted-document?doi=10.14236/ewic/HCI2017.77>>. Citado 2 vezes nas páginas 13 e 14.
- RADICS, S. *Scrum, Kanban, Scrumban - a fast overview and rough categorization when to use what method*. 2013. Disponível em: <<https://www.ontheagilepath.net/2013/09/scrum-kanban-scrumban-fast-overview-and.html>>. Citado na página 20.
- ROSENBAUM, M.; OTÁLORA, M.; RAMIREZ, G. How to create a realistic customer journey map. *Business Horizons*, out. 2016. Citado na página 28.
- RUBIN, J.; WILEY, J.; SONS. Handbook of usability testing: How to plan, design and conduct effective tests. 1984. Citado na página 26.
- SCHWABER, K. SCRUM Development Process. In: SUTHERLAND, J. et al. (Ed.). *Business Object Design and Implementation*. London: Springer London, 1997. p. 117–134. ISBN 978-3-540-76096-2 978-1-4471-0947-1. Disponível em: <http://link.springer.com/10.1007/978-1-4471-0947-1_11>. Citado na página 13.
- SENSUSE, D. et al. (PDF) Integrating UCD into Scrumban for better and faster usability design. *Conference: 2017 International Conference on Information Technology Systems and Innovation (ICITSI)*, out. 2017. Disponível em: <https://www.researchgate.net/publication/322706963_Integrating_UCD_into_Scrumban_for_better_and_faster_usability_design>. Citado 2 vezes nas páginas 13 e 14.
- SHARMA, S.; SARKAR, D.; GUPTA, D. Agile processes and methodologies: A conceptual study. *International Journal on Computer Science and Engineering*, v. 4, 05 2012. Citado na página 34.
- SILVA, T. D. et al. User experience design and agile development: From theory to practice. *Journal of Software Engineering and Applications*, v. 05, p. 743–751, 01 2012. Citado na página 34.

SOARES, M. D. S. Metodologias Ágeis Extreme Programming e Scrum para o Desenvolvimento de Software. *Revista Eletrônica de Sistemas de Informação*, v. 3, n. 1, p. 1, jun. 2004. ISSN 1677-3071. Disponível em: <<http://www.periodicosibepes.org.br/index.php/reinfo/article/view/146>>. Citado na página 13.

SY, D. *Adapting Usability Investigations for Agile User-Centered Design* - *JUX*. 2007. Disponível em: <<https://uxpajournal.org/adapting-usability-investigations-for-agile-user-centered-design/>>. Citado na página 35.

SY, D.; MILLER, L. Optimizing agile user-centred design. In: . [S.l.: s.n.], 2008. p. 3897–3900. Citado na página 35.

VENTURI, G.; TROOST, J.; JOKELA, T. People, organizations, and processes: An inquiry into the adoption of user-centered design in industry. *International Journal of Human-Computer Interaction*, v. 21, p. 219, 11 2006. Citado na página 32.

YERET, Y. *So what IS Scrumban?* - *Yeret Agility*. 2012. Disponível em: <<https://yuvalyeret.com/so-what-is-scrumban/>>. Citado na página 21.

Apêndices

APÊNDICE A – Carta de Solicitação

Figura 29 – Carta de solicitação de acesso e citação



UNIVERSIDADE DE BRASÍLIA - UNB
FACULDADE DO GAMA - FGA
CURSO DE ENGENHARIA DE SOFTWARE

A Atena Soluções Informática Ltda. pessoa jurídica devidamente inscrita no CNPJ nº 08.189.785/0001-02, com endereço Rua da Imprensa 560, bairro Ipiranga, na cidade de São Paulo, Estado de São Paulo, telefone para contato (11)97070-1925; neste ato representada por seu responsável legal Frederico José Pelúcio de Castro., profissão administrador, portador da R.G nº 63137154-0 e inscrito no CPF/MF 057648118-13, por intermédio da presente autoriza a realização, em suas dependências e fora delas, do Projeto de Pesquisa do trabalho de conclusão de curso intitulado: Diretrizes de desenvolvimento centradas no usuário no processo Scrumban: um estudo de caso.....que tem por objetivo analisar a inserção de procedimentos de *UX Design* no modelo de processo de software da empresa. Autoriza expressamente a divulgação da pesquisa, do nome da empresa, dos modelos de processo, dados e do resultado. Declara que tem conhecimento e que concorda plenamente que a participação da empresa que representa se dá à título gratuito, não recebendo, portanto nenhum honorário ou gratificação referente ao projeto de pesquisa. Concorda com a possibilidade de as informações relacionadas ao estudo serem inspecionadas pelo orientador da pesquisa.

São Paulo, dia 27 de outubro de 2022

(Assinatura)

Empresa: Atena Soluções Informática Ltda.
Nome completo do responsável legal: Frederico José Pelúcio de Castro
Telefone p/ contato: (11)97070-1925
CPF/MF: 057648118-13

APÊNDICE B – Questionário Grupo Focal

Figura 30 – Formulário do Grupo Focal

Formulário de Grupo Focal - Groovoo

Dada a conversa sobre a funcionalidade apresentada. Solicitamos um parecer sobre sua possibilidade de inserção no produto Groovoo.



Qual é o seu nível de habilidade técnica em relação ao uso de produtos de software?

Iniciante

Intermediário

Avançado

Você já utiliza o Groovoo como plataforma para compra ou venda de ingressos?

Sim

Não

Quão satisfeito você está com as funcionalidades e desempenho do produto de software atualmente?

Muito satisfeito

Satisfeito

Insatisfeito

Muito insatisfeito

Você considera que a sua cultura ou origem social influencia como você utiliza produtos de software?

Sim

Não

Quão confortável você se sente ao usar novas tecnologias ou recursos em produtos de software?

- Muito desconfortável
- Moderadamente confortável
- Muito confortável

Quão familiarizado você está com a funcionalidade apresentada?

- Nada familiarizado
- Um pouco familiarizado
- Muito familiarizado

Você considera que a localização geográfica influencia suas necessidades e expectativas em relação à funcionalidade apresentada?

- Sim
- Não

Você considera que documentação irregular pode influenciar o uso da funcionalidade apresentada?

- Sim
- Não

Você teria dificuldade técnica com uso da funcionalidade apresentada?

- Sim
- Não
- Talvez

Você teria dificuldade social com uso da funcionalidade apresentada?

- Sim
- Não
- Talvez

Você teria dificuldade cognitiva com uso da funcionalidade apresentada?

- Sim
- Não
- Talvez

Você acredita que a inserção da nova funcionalidade pode melhorar a sua produtividade ou rotina diária no uso do produto?

- Sim
- Não

Como você avalia a importância de receber suporte técnico e assistência em relação à funcionalidade apresentada?

- Muito importante
- Moderadamente importante
- Pouco importante

Fonte: Os autores

APÊNDICE C – Questionário da Entrevista Groovoo

Figura 31 – Formulário Entrevista

Dada a apresentação do cenário da pesquisa de Diretrizes Centradas no Usuário a serem inseridas no Processo de Desenvolvimento da Groovoo. ●

Solicitamos um parecer estratégico da empresa.

A organização está sensível a necessidade de atividades de reconhecimento da experiência do usuário?

Sim

Não

Quais as principais vulnerabilidades dentro do atual processo?

Indentificação das reais necessidades do usuário

Especificação dos Requisitos

Alto tempo de manutenção

Falta de Priorização

Outros...

Ao surgir uma nova demanda, a empresa tem estratégias de identificação e adaptação às necessidades do usuário?

Sim

Não

As vezes

Outros...

Qual o tempo ideal para se dedicar a atividade de identificação das necessidades do usuário?

Menos de 3 dias
 4 dias
 5 dias
 1 Semana
 Mais de uma semana
 Outros...

Considerando as diretrizes de UCD da literatura e os métodos e as técnicas selecionadas. Enumere de 1 a 5 como você avalia a aplicação da Técnica de Grupo Focal para estudo dos cenários dos usuários?

Ruim 1 2 3 4 5 Ótima

Enumere de 1 a 5 como você avalia a aplicação da Técnica de Carding Sorting para elicitación de requisitos?

Não faz sentido 1 2 3 4 5 Essencial

Enumere de 1 a 5 como você avalia a aplicação da Técnica de Design Participativo para elicitación de prototipação?

Não faz sentido 1 2 3 4 5 Essencial

Enumere de 1 a 5 como você avalia a aplicação da Técnica de Thinking Aloud para validação de protótipos?

Não faz sentido 1 2 3 4 5 Essencial

Enumere de 1 a 5 como você avalia a aplicação da Técnica Ready Queue para priorização de tarefas?

Não faz sentido 1 2 3 4 5 Essencial

Considerando a equipe atual, qual seria o Limite de WIP ideal para o processo?

- 2
- 3
- Outros...

A empresa tem Políticas de Trabalho explícitas?

- Sim
- Não
- Outros...

Entre as técnicas do UCD apresentadas, numa escala de prioridade, defina as técnicas mais prioritárias, considerando custo/benefício:

	Baixa Prioridade	Média Prioridade	Alta Prioridade
Grupo Focal	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Thinking Aloud	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Carding Sorting	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Design Participativo	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

...

Entre as técnicas do Scrumban apresentadas, numa escala de prioridade, defina as técnicas mais prioritárias, considerando custo/benefício:

	Baixa Prioridade	Média Prioridade	Alta Prioridade
Ready Queue	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Limitação da WIP	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Plano de comunicação	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Definição do que é Pronto	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Solução de gargalo	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Regras de convivência	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Quebrar US em Issues	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Fonte: Os autores

APÊNDICE D – Questionário e *User Flow* *Thinking Aloud*

Figura 32 – Questionário *Thinking Aloud*

Questionário Thinking Aloud

Este formulário tem o intuito de coletar dados de uma tarefa do protótipo de alta fidelidade através da técnica Thinking Aloud.

Breve descrição sobre o perfil do usuário

Texto de resposta longa
.....

O usuário conseguiu concluir a tarefa?

Sim

Não

O usuário cometeu algum erro na realização da tarefa? Se sim, quais?

Texto de resposta longa
.....

O usuário levou quanto tempo para realizar a tarefa?

Texto de resposta curta
.....

O usuário expressou alguma reação negativa? Se sim, qual?

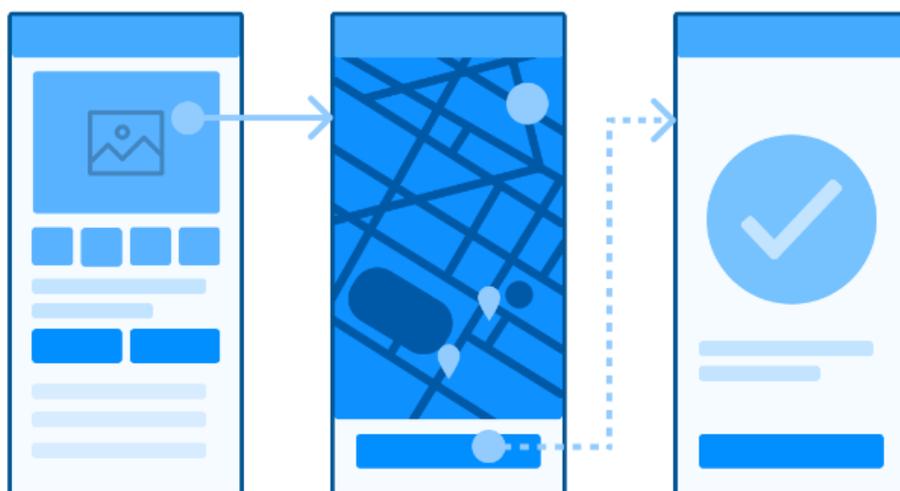
Texto de resposta longa
.....

Após terminar a tarefa, o usuário tem algum comentário a fazer?

Texto de resposta longa
.....

Fonte: Os autores

Figura 33 – Exemplo de *User Flow*



Fonte: (BROWNE, 2023)