

UNIVERSIDADE FEDERAL DO RIO DE JANEIRO  
INSTITUTO DE COMPUTAÇÃO  
CURSO DE BACHARELADO EM CIÊNCIA DA COMPUTAÇÃO

GABRIELE JANDRES CAVALCANTI  
SABRINA ALENCAR DAS NEVES

Redefinindo o *Handoff*: Mais do que uma passagem de bastão, uma  
colaboração contínua

RIO DE JANEIRO  
2024

GABRIELE JANDRES CAVALCANTI  
SABRINA ALENCAR DAS NEVES

Redefinindo o *Handoff*: Mais do que uma passagem de bastão, uma  
colaboração contínua

Trabalho de conclusão de curso de graduação  
apresentado ao Instituto de Computação da  
Universidade Federal do Rio de Janeiro como  
parte dos requisitos para obtenção do grau de  
Bacharel em Ciência da Computação.

Orientador: Profa. Adriana Santarosa Vivacqua  
Co-orientador: Profa. Juliana Baptista dos Santos França

RIO DE JANEIRO

2024

C377r

Cavalcanti, Gabriele Jandres

Redefinindo o Handoff: mais do que uma passagem de bastão, uma colaboração contínua / Gabriele Jandres Cavalcanti e Sabrina Alencar das Neves. – 2024.

61 f.

Orientadora: Adriana Santarosa Vivacqua.

Coorientadora: Juliana Baptista dos Santos França.

Trabalho de Conclusão de Curso (Bacharelado em Ciência da Computação)-  
Universidade Federal do Rio de Janeiro, Instituto de Computação, Bacharel em  
Ciência da Computação, 2024.

1. Colaboração. 2. Comunicação. 3. Handoff. 4. Design. 5. Scrum. I. Neves,  
Sabrina Alencar das. II. Vivacqua, Adriana Santarosa (Orient.). III. França, Juliana  
Baptista dos Santos. IV. Universidade Federal do Rio de Janeiro, Instituto de  
Computação. V. Título.

GABRIELE JANDRES CAVALCANTI  
SABRINA ALENCAR DAS NEVES

Redefinindo o *Handoff*: Mais do que uma passagem de bastão, uma  
colaboração contínua

Trabalho de conclusão de curso de graduação  
apresentado ao Instituto de Computação da  
Universidade Federal do Rio de Janeiro como  
parte dos requisitos para obtenção do grau de  
Bacharel em Ciência da Computação.

Aprovado em 27 de março de 2024  
BANCA EXAMINADORA:

Documento assinado digitalmente  
 ADRIANA SANTAROSA VIVACQUA  
Data: 02/04/2024 16:37:17-0300  
Verifique em <https://validar.iti.gov.br>

---

Adriana Santarosa Vivacqua  
Doutora em Informática (Instituto de  
Computação - UFRJ)

Documento assinado digitalmente  
 GISELI RABELLO LOPES  
Data: 02/04/2024 21:29:48-0300  
Verifique em <https://validar.iti.gov.br>

---

Giseli Rabello Lopes  
Doutora em Computação (Instituto de  
Computação - UFRJ)

Documento assinado digitalmente  
 JULIANA BAPTISTA DOS SANTOS FRANCA  
Data: 02/04/2024 15:47:41-0300  
Verifique em <https://validar.iti.gov.br>

---

Juliana Baptista dos Santos França  
Doutora em Informática (Instituto de  
Computação - UFRJ)

Documento assinado digitalmente  
 MARIA LUIZA MACHADO CAMPOS  
Data: 03/04/2024 20:20:56-0300  
Verifique em <https://validar.iti.gov.br>

---

Maria Luiza Machado Campos  
Doutora em Sistemas de Informação  
(Instituto de Computação - UFRJ)

Documento assinado digitalmente  
 SILVANA ROSSETTO  
Data: 02/04/2024 20:59:50-0300  
Verifique em <https://validar.iti.gov.br>

---

Silvana Rossetto  
Doutora em Informática (Instituto de  
Computação - UFRJ)

Primeiramente, dedico esse trabalho a Deus, pelo amor incondicional pela minha vida e por ter me dado suporte nos momentos mais difíceis.

À minha amiga Sabrina, por toda parceria durante a faculdade, e também por tudo o que passamos juntas para terminar esse trabalho, hoje vemos que tudo valeu a pena. O processo foi mais prazeroso e satisfatório porque estávamos juntas. Tenho plena certeza da sua capacidade, profissionalismo e carinho em tudo que faz, e sei que você vai longe.

Aos meus pais, Denize e Joilson, pelo apoio durante não só a minha jornada acadêmica, mas durante toda a minha vida. Obrigada pelos inúmeros sacrifícios feitos para me ensinar o caminho que eu deveria trilhar e por toda a dedicação para me proporcionar uma educação de qualidade, obrigada por serem meus maiores incentivadores.

À minha irmã, Letícia, por todo apoio, incentivo e paciência na conciliação da acadêmica, profissional e pessoal. Obrigada por ser minha parceira, sempre vou estar aqui torcendo pelo seu sucesso e admirando quem você está se tornando.

Ao meu noivo, Victor, pela parceria de vida, mas em especial na jornada acadêmica. Obrigada por estar comigo nos momentos de alegrias, tristezas, surtos e vitórias. Obrigada por me apoiar e incentivar em todos os momentos que duvidei de mim mesma e achei que não conseguiria. Você torna tudo mais leve, é um privilégio dividir a vida com você.

À minha sogra, Maria Helena, por todo o apoio na jornada da graduação, desde as caronas até o acolhimento nos momentos em que precisei.

À minha família, em especial meus avós e tia Glória, que me incentivaram desde pequena nos estudos.

Por fim, aos meus amigos do Clube das *Winx*, por dividirem a jornada acadêmica comigo e tornarem os dias na UFRJ mais leves e divertidos.

*Gabriele Jandres Cavalcanti*

À Deus e Nossa Senhora de Fátima por ter me dado saúde e motivação todos os dias para que eu conseguisse me dedicar da melhor forma à esse trabalho e pudesse colocar um pouquinho de mim nele.

À minha dupla e amiga, Gabi, que tornou esse processo muito mais prazeroso e especial para mim. Todas as reuniões, surtos, ideias e mudanças valeram a pena. Espero que você brilhe e seja muito feliz em sua carreira e vida.

Aos meus pais, Valquíria e Edson, pelo apoio com relação a meus estudos durante toda minha vida, por estarem e serem tão presentes. *E por cada carona para faculdade, pai. Elas tornaram tudo um pouco mais fácil.*

À meu irmão, Gui, por sempre me escutar, principalmente as reclamações e empolgações sobre o TCC. Você sempre faz diferença.

Ao meu bem, João, pelo apoio emocional e por ter dividido mais essa jornada comigo. Toda caminhada com você faz mais sentido e me faz mais feliz.

À minha psicóloga Tainã, por me ajudar a me entender melhor e tornar minha jornada com os estudos muito mais leve e satisfatória.

E à todos que participam da minha vida, me incentivando a chegar até aqui. Professores, amigos, família, vocês me fazem muito sortuda.

*Sabrina Alencar das Neves*

## AGRADECIMENTOS

Primeiramente, gostaríamos de expressar nossos sinceros agradecimentos às nossas orientadoras, Adriana Vivacqua e Juliana França, por desempenharem com tanto carinho e dedicação o papel de orientação. Seus incentivos e contribuições foram fundamentais para o sucesso deste trabalho. Vocês nos inspiram e nos motivam a entregar o melhor de nós.

Além disso, não poderíamos deixar de agradecer à professora Maria Luiza pela orientação acadêmica e pelo apoio durante todo o percurso acadêmico. Você foi a maior responsável por nos guiar com sabedoria e possibilitar vivenciarmos períodos mais saudáveis na faculdade. Seu estímulo para aproveitarmos as diversas atividades acadêmicas ampliou nossa visão sobre a importância da faculdade além das salas de aula.

Também expressamos nossa gratidão ao professor Cláudio Miceli, que, apesar de não estar diretamente envolvido na orientação deste projeto, aceitou nos ajudar na primeira ideia de tema que tivemos. Suas dicas e reuniões contribuíram muito para nosso aprendizado de escrita acadêmica e planejamento do trabalho.

Não poderíamos de deixar de agradecer também aos profissionais, designers e desenvolvedores, que participaram de nossas entrevistas e contribuíram significativamente para os resultados desse estudo.

E finalmente, agradecemos aos estudantes membros das empresas juniores EJCM e Fluxo, cuja participação em nossos testes foi fundamental para validar nossas propostas. Agradecemos sinceramente pelo tempo e esforço dedicados.

*“Designers craft the look and feel of a product, while developers code it. But they still have to work in tandem to create a well-designed final product. That’s why the better collaboration between designers and developers is established, the better outcome from the design process will be.”*

**Coren Feldman**

## RESUMO

O aumento da importância e visibilidade do UX tem levado a uma maior interdisciplinaridade nas equipes de desenvolvimento de *software*, que cada vez mais abrangem uma gama diversificada de profissionais. Mas, essa crescente integração também tem gerado desafios consideráveis, porque a colaboração entre profissionais com perspectivas e interesses diversos sobre um mesmo produto pode ser complexa, resultando em dificuldades de comunicação e alinhamento. Um dos pontos de colaboração mais conhecidos no ciclo de desenvolvimento é o *handoff*. Esse é um momento em que a equipe de *design* passa o trabalho à equipe de desenvolvimento, sendo crucial no ciclo de vida de um projeto de *software*. No entanto, é também um ponto onde surgem desafios significativos entre as duas equipes, que podem levar a mal-entendidos, retrabalho e resultados insatisfatórios. Apesar desses problemas serem evidenciados muitas vezes no *handoff*, a raiz deles não reside apenas nesse momento, mas, principalmente, na falta de comunicação e colaboração entre as equipes durante todo o processo. Nesse cenário, o objetivo desse estudo é investigar os desafios inerentes ao processo de transição atualmente, focando nas possíveis causas que permeiam todo o ciclo de desenvolvimento. Foram realizadas pesquisas quantitativas e qualitativas com o objetivo de compreender as interações diárias das equipes, identificar conflitos e desafios enfrentados, bem como as práticas que funcionam ou não. Após a identificação e categorização dos desafios, foi proposta uma adaptação do Scrum e do ciclo de vida do desenvolvimento de *software*, com o objetivo de aprimorar a colaboração. Tal proposta de extensão visa introduzir três pontos de contato adicionais entre as equipes em momentos estratégicos: *Design Alignment*, *Design Review* e *Design Validation*. Para avaliar a eficácia das adições e adaptações implementadas no ciclo de desenvolvimento e no quadro do Scrum, foram realizados testes com equipes de projetos de duas empresas juniores, que em geral demonstraram um bom resultado das práticas sugeridas.

**Palavras-chave:** colaboração; comunicação; *handoff*; design; desenvolvimento; scrum; ciclo de vida do desenvolvimento de *software*

## ABSTRACT

The increased importance and visibility of UX has led to greater interdisciplinarity in software development teams, which increasingly encompass a diverse range of professionals. But this growing integration has also created considerable challenges, because collaboration between professionals with different perspectives and interests on the same product can be complex, resulting in difficulties in communication and alignment. One of the best-known collaboration points in the development cycle is the handoff. This is a moment when the design team hands over work to the development team, and is crucial in the life cycle of a software project. However, it is also a point where significant challenges arise between the two teams, which can lead to misunderstandings, rework and unsatisfactory results. Although these problems are often evident at the handoff, the root of them lies not only at this point, but mainly in the lack of communication and collaboration between the teams during all the process. In this context, the aim of this study is to investigate the challenges inherent in the transition process today, focusing on the possible causes that permeate the entire development cycle. Quantitative and qualitative research was carried out with the aim of understanding the teams' daily interactions, identifying conflicts and challenges faced, as well as practices that work or not. After identifying and categorizing the challenges, an adaptation of Scrum and the software development lifecycle was proposed, with the aim of improving collaboration. This proposed extension aims to introduce three additional points of contact between the teams at strategic moments: Design Alignment, Design Review and Design Validation. In order to evaluate the effectiveness of the additions and adaptations implemented in the development cycle and within the Scrum framework, tests were carried out with project teams from two junior enterprises, which in general showed a good result from the suggested practices.

**Keywords:** collaboration; communication; handoff; design; development; scrum; software development lifecycle

## LISTA DE ILUSTRAÇÕES

Figura 1 – Cerimônias e artefatos do Scrum . . . . .	21
Figura 2 – Ciclo de vida do desenvolvimento de software . . . . .	21
Figura 3 – Scrum board . . . . .	22
Figura 4 – Método de pesquisa utilizado . . . . .	27
Figura 5 – Matriz de Certezas, Suposições e Dúvidas . . . . .	28
Figura 6 – Histórias dos usuários . . . . .	41
Figura 7 – SDLC estendido . . . . .	43
Figura 8 – Scrum Board estendido . . . . .	45

## LISTA DE TABELAS

Tabela 1 – Descrição de participantes da pesquisa quantitativa . . . . .	29
Tabela 2 – Descrição de participantes das entrevistas . . . . .	31
Tabela 3 – Desafios percebidos no <i>handoff</i> e suas fases de origem . . . . .	38
Tabela 4 – Desafios percebidos no <i>handoff</i> e propostas de soluções . . . . .	44
Tabela 5 – Uso de cerimônias propostas pelos entrevistados . . . . .	45

## LISTA DE ABREVIATURAS E SIGLAS

UX	<i>User Experience</i> (Experiência do Usuário)
UI	<i>User Interface</i> (Interface do Usuário)
CSD	<i>Certainties, Suppositions and Doubts</i> (Certezas, Suposições e Dúvidas)
SDLC	<i>Software Development Lifecycle</i> (Ciclo de vida do desenvolvimento de software)
PO	<i>Product Owner</i>
PM	<i>Product Manager</i>
QA	<i>Quality Assurance</i>
EJCM	Empresa Júnior de Consultoria em Microinformática
UFRJ	Universidade Federal do Rio de Janeiro
MEJ	Movimento Empresa Júnior
MER	Modelo Entidade-Relacionamento

## SUMÁRIO

<b>1</b>	<b>INTRODUÇÃO . . . . .</b>	<b>15</b>
1.1	MOTIVAÇÃO . . . . .	15
1.2	PROBLEMA . . . . .	15
1.3	OBJETIVO . . . . .	16
1.4	METODOLOGIA . . . . .	16
1.5	PROPOSTA DE SOLUÇÃO . . . . .	16
1.6	ESTRUTURA . . . . .	17
<b>2</b>	<b>REFERENCIAL TEÓRICO . . . . .</b>	<b>18</b>
2.1	CONCEITOS . . . . .	18
2.1.1	Experiência do usuário . . . . .	18
2.1.2	Interface do usuário . . . . .	18
2.1.3	<i>Handoff</i> ou passagem de bastão . . . . .	19
2.1.3.1	Definições autorais . . . . .	19
2.1.4	Metodologia ágil . . . . .	20
2.1.4.1	Scrum . . . . .	20
2.2	TRABALHOS RELACIONADOS . . . . .	21
2.2.1	Importância do <i>handoff</i> . . . . .	22
2.2.2	Desafios do <i>handoff</i> . . . . .	23
2.2.2.1	Ferramentas comerciais . . . . .	24
2.2.3	Adaptações do Scrum . . . . .	25
2.2.3.1	Proposta . . . . .	25
<b>3</b>	<b>METODOLOGIA . . . . .</b>	<b>27</b>
3.1	ESTUDO UM: PESQUISA QUANTITATIVA . . . . .	28
3.1.1	Recrutamento . . . . .	28
3.1.2	Participantes . . . . .	29
3.1.3	Procedimento . . . . .	29
3.1.4	Coleta de dados . . . . .	30
3.1.5	Análise . . . . .	30
3.2	ESTUDO DOIS: PESQUISA QUALITATIVA . . . . .	30
3.2.1	Recrutamento . . . . .	31
3.2.2	Participantes . . . . .	31
3.2.3	Procedimento . . . . .	31
3.2.4	Coleta de dados . . . . .	32
3.2.5	Análise . . . . .	32

4	<b>RESULTADOS</b>	33
4.1	ESTUDO UM: PESQUISA QUANTITATIVA	33
4.1.1	Contexto profissional	33
4.1.2	Interação entre <i>designers</i> e desenvolvedores e seus meios de comunicação	34
4.1.3	Comunicação restrita ao momento do <i>handoff</i>	34
4.1.4	Desenvolvedores priorizam <i>handoff</i> sobre contato	35
4.1.5	Efeitos da inexistência do <i>handoff</i>	36
4.1.6	Conclusão a partir dos dados quantitativos	36
4.2	ESTUDO DOIS: PESQUISA QUALITATIVA	37
4.2.1	<i>Pré-handoff</i>	37
4.2.1.1	Falta de alinhamento sobre a entrega ideal	38
4.2.1.2	Ausência de fluxos alternativos	38
4.2.1.3	Inviabilidade técnica do protótipo	39
4.2.2	<i>Handoff</i>	39
4.2.2.1	Ausência de reunião de apresentação	39
4.2.2.2	Pouca familiaridade com a ferramenta de design	39
4.2.3	<i>Pós-handoff</i>	40
4.2.3.1	Ausência de validação entre protótipo e <i>software</i> final	40
4.2.4	Histórias dos usuários	40
4.3	EXTENSÃO DO SDLC	40
4.3.1	Perspectiva do <i>Handoff: Hands Off?</i>	42
4.3.2	Adaptação do ciclo	42
4.3.2.1	<i>Design Alignment</i>	43
4.3.2.2	<i>Design Review</i>	43
4.3.2.3	<i>Design Validation</i>	44
4.3.3	Adaptação do <i>board</i> Scrum	44
4.3.4	Análise da aplicabilidade da solução nos contextos iniciais	44
5	<b>TESTES</b>	46
5.1	ENTREVISTAS	46
5.1.1	EJCM Consultoria	46
5.1.2	Fluxo Consultoria	47
5.1.3	Conclusão das entrevistas	48
5.2	APRESENTAÇÃO DA PROPOSTA	48
5.3	ANÁLISE DOS RESULTADOS	49
5.3.1	Pontos positivos	49
5.3.2	Pontos de limitação	49
5.3.3	Pontos de melhoria	50

<b>6</b>	<b>CONCLUSÃO</b> . . . . .	<b>51</b>
<b>6.1</b>	RESUMO . . . . .	<b>51</b>
<b>6.2</b>	LIMITAÇÕES . . . . .	<b>51</b>
<b>6.3</b>	TRABALHOS FUTUROS . . . . .	<b>52</b>
	<b>REFERÊNCIAS</b> . . . . .	<b>54</b>
	<b>GLOSSÁRIO</b> . . . . .	<b>56</b>
	<b>ANEXO A – ROTEIRO QUANTITATIVA</b> . . . . .	<b>57</b>
<b>A.1</b>	GERAIS . . . . .	<b>57</b>
<b>A.2</b>	DESIGNER . . . . .	<b>57</b>
<b>A.3</b>	DESENVOLVEDOR . . . . .	<b>57</b>
<b>A.4</b>	DEMAIS PROFISSIONAIS . . . . .	<b>58</b>
	<b>ANEXO B – ROTEIRO QUALITATIVA</b> . . . . .	<b>59</b>
<b>B.1</b>	GERAIS . . . . .	<b>59</b>
<b>B.2</b>	DESIGNER . . . . .	<b>59</b>
<b>B.3</b>	DESENVOLVEDOR . . . . .	<b>59</b>
	<b>ANEXO C – ROTEIRO TESTE</b> . . . . .	<b>61</b>

# 1 INTRODUÇÃO

Não existe um dia em que não tenhamos contato com algum tipo de tecnologia, ela tomou conta do ambiente ao nosso redor e para onde quer que se olhe há algo tecnológico - mesmo que seja uma televisão antiga que logo será trocada por uma *Smart TV* com Netflix. E falando sobre *streamings*, eles só existem devido a um processo de desenvolvimento de *software* por trás, que tem passado por várias evoluções e diferentes metodologias ao longo do tempo.

Com o avanço da Internet, a usabilidade e a experiência do usuário (UX) têm ganhado cada vez mais destaque, tornando-se prioridades para as empresas de *software* que buscam garantir a satisfação e que desejam fidelizar seus usuários.

## 1.1 MOTIVAÇÃO

O aumento da importância e visibilidade do UX tem levado a uma maior interdisciplinaridade nas equipes de desenvolvimento de *software*, que cada vez mais abrangem uma gama diversificada de profissionais, com formações e conhecimentos diferentes, como desenvolvedores, *designers* de experiência (UX, UI (*User Interface*), *Research*, *Writing*), analistas de qualidade, *Product Managers*, *Product Owners* e várias outras profissões que surgem a cada dia.

No entanto, apesar de benéfica, essa crescente integração também tem gerado desafios consideráveis. A colaboração entre profissionais com perspectivas e interesses diversos sobre um mesmo produto pode ser complexa, resultando em dificuldades de comunicação e alinhamento (MAUDET et al., 2017; FERATI, 2022). Por isso, garantir que uma equipe esteja alinhada e se comunique de maneira clara não é uma tarefa simples, e frequentemente, surgem conflitos entre dois protagonistas deste cenário: o desenvolvedor e o *designer*.

## 1.2 PROBLEMA

O *handoff* é um momento crucial no ciclo de vida de um projeto de *software* em uma empresa de desenvolvimento, especialmente no contexto ágil. No entanto, é também um ponto onde frequentemente surgem desafios significativos de comunicação e colaboração entre as duas equipes, que podem levar a mal-entendidos, retrabalho e resultados insatisfatórios. Por isso, identificar os desafios e as práticas colaborativas que facilitam essa transição de conhecimento e garantem uma execução eficaz é essencial para aprimorar esse processo (FERATI, 2022; FENG; ZHANG, 2022).

Apesar desses problemas serem evidenciados muitas vezes no momento do *handoff* - foco inicial dessa pesquisa - ao longo do processo, ficou evidente que a raiz deles não reside apenas nesse momento, mas, principalmente, na falta de comunicação e colaboração entre as equipes de *design* e desenvolvimento durante todo o processo. Os impactos dessa ausência de alinhamento são diversos e impactam diretamente a qualidade e eficiência do produto final.

### 1.3 OBJETIVO

Esse estudo endereça os desafios do *handoff* através da lente da metodologia ágil, que busca dar maior rapidez aos processos, e do ciclo do desenvolvimento de *software*. Durante a investigação, foi descoberto que muitos envolvidos já estão familiarizados com o conceito de *handoff*. Porém, neste estudo, optou-se por abordar o assunto sob outra lente ainda não considerada em trabalhos relacionados: as raízes dos desafios e possíveis soluções foram buscadas não só nesse momento do *handoff*, mas também no *pré-handoff* e *pós-handoff*, ou seja em todo ciclo de vida do desenvolvimento, desde a fase de descoberta até a entrega final.

Além de analisar os problemas já previamente identificados na literatura, o objetivo desse estudo é investigar os desafios inerentes ao processo de *handoff* em times ágeis atualmente, focando nas possíveis causas que permeiam todo o ciclo de desenvolvimento e posteriormente, com os desafios categorizados em mãos, propor uma solução que incentive a constante colaboração entre os desenvolvedores e *designers*. Com essa abordagem, foi possível identificar pontos de dor e problemas ao longo de todo o processo, bem como identificar momentos estratégicos para possíveis soluções.

### 1.4 METODOLOGIA

Foram realizadas pesquisas quantitativas e qualitativas com o objetivo de aprofundar o conhecimento sobre a realidade desses profissionais. Por meio delas, investigou-se suas interações diárias, conflitos, desafios enfrentados e também quais práticas de *handoff* funcionam ou não em seus times.

### 1.5 PROPOSTA DE SOLUÇÃO

É perceptível que a colaboração entre esses dois protagonistas é essencial para a qualidade e consistência do entregável final. E se o *designer* é responsável por criar a solução da experiência que desejamos proporcionar ao cliente, e o desenvolvedor por tornar isso realidade, por que não priorizamos uma transição eficiente e passamos a considerar o desenvolvedor como o usuário final das entregas do *designer*? (GREEN; LABRECQUE, 2023).

Nesse contexto, esse estudo propõe uma adaptação do Scrum, que é um *framework* prático para que equipes consigam desenvolver produtos conforme a metodologia ágil, e do ciclo de vida do desenvolvimento de *software*, com o objetivo de aprimorar a colaboração entre as equipes de desenvolvimento e *design*. Essa adaptação incluirá pontos de contato adicionais, visando verificar viabilidades técnicas para evitar retrabalhos e discutir todos os fluxos alternativos no momento de *pré-handoff*, facilitar a transição no momento do *handoff* e validar a implementação em comparação com o protótipo inicial, a fim de reduzir inconsistências e problemas de UI em produção, no momento de *pós-handoff*. É importante destacar que o foco desse estudo estará na colaboração entre as equipes e o aumento dos pontos de contato, mas isso não substitui a necessidade de uma boa documentação do processo, que servirá como base de entendimento para futuros colaboradores.

## 1.6 ESTRUTURA

Esse estudo inicia com a introdução, apresentando o contexto e os objetivos da pesquisa. No referencial teórico, serão abordados os conceitos fundamentais discutidos ao longo do texto, além de trabalhos relacionados que foram base para a investigação sobre o processo de *handoff* e as metodologias ágeis. Em seguida, serão detalhados os procedimentos adotados para realizar as pesquisas quantitativa e qualitativa na seção de metodologia, bem como as estratégias utilizadas para análise dos dados obtidos. No capítulo de resultados, serão destacadas as principais conclusões alcançadas a partir das pesquisas e será apresentada a proposta de solução, discutindo as adaptações do ciclo de desenvolvimento e como elas resolvem os desafios identificados. Já na seção dos testes, serão apresentadas informações sobre os testes realizados e suas implicações, e finalmente, no capítulo de conclusões, os principais achados da pesquisa serão sintetizados, as contribuições serão discutidas e possíveis direções para pesquisas futuras serão delineadas.

## 2 REFERENCIAL TEÓRICO

Nesse capítulo, serão apresentados os conceitos principais que serão abordados nesse estudo, bem como alguns dos trabalhos encontrados na literatura que também se propuseram a entender os desafios do *handoff*.

### 2.1 CONCEITOS

Primeiramente, é essencial que os conceitos que estarão presentes durante o desenvolvimento desse texto sejam apresentados.

#### 2.1.1 Experiência do usuário

De acordo com (DESMET; HEKKERT, 2007), a experiência do usuário é

"Todo o conjunto de efeitos que é provocado pela interação entre um usuário e um produto incluindo o grau em que todos os nossos sentidos são satisfeitos (experiência estética), os significados que atribuímos ao produto (experiência de significado) e os sentimentos e emoções que são suscitados (experiência emocional)."

De uma forma um pouco mais prática, a experiência do usuário não precisa ser explicada somente através de produtos digitais, ela pode ser percebida desde os objetos mais simples até os mais complexos do cotidiano. Um trecho do livro "*Design do dia a dia*" (NORMAN, 2018) evidencia esse conceito:

"Considerem a porta. Não há muito que se possa fazer com uma porta: você pode abri-la ou fechá-la. Suponhamos que você esteja num prédio de escritórios, caminhando por um corredor. Você chega a uma porta. Em que direção ela abre? Você deve puxar ou empurrar, do lado direito ou esquerdo? Talvez a porta seja de correr, se for, em que direção? Já vi portas que deslizam para cima, para dentro do teto. Uma porta formula apenas duas perguntas essenciais: Para que direção ela se move? De qual dos lados deveríamos manejá-la? As respostas deveriam ser dadas pelo *design*, sem necessidade de palavras ou símbolos, e certamente sem nenhuma necessidade de tentativa e erro."

#### 2.1.2 Interface do usuário

De acordo com (JACOB, 2003), uma interface de usuário (UI) é a parte de um sistema ou produto - físico ou digital - que se comunica com o usuário, isto é, o *design* da UI inclui qualquer aspecto do sistema que seja visível. À medida que grupos mais amplos de pessoas usam computadores, ela se torna mais importante no mundo digital, mas sempre esteve muito presente no cotidiano.

Na prática, ela pode aparecer sob a forma de botões de fogão, o *feed* do Instagram, a tela de busca do Google ou as maçanetas de portas a serem abertas. Outro trecho do

livro "*Design do dia a dia*" (NORMAN, 2018) explica que quanto mais simples e intuitiva essa interface for, melhor para quem está utilizando-a:

"A regra prática que a experiência me ensinou reza: quando é necessário que as instruções sejam coladas em alguma coisa (aperte ou empurre aqui, insira desta maneira, desligue antes de fazer isso), o *design* é ruim."

### 2.1.3 *Handoff* ou passagem de bastão

Outro conceito principal nesse estudo é o *handoff*. No processo de desenvolvimento de um produto, é fácil imaginar as responsabilidades de cada profissional como sendo caixinhas diferentes dentro do fluxo. Nesse sentido, há um momento em que as responsabilidades do UX/UI *design* diminuem e dão lugar para o trabalho dos desenvolvedores.

De acordo com (BABICH; WHITE, 2017) as equipes de desenvolvimento gastam 50% do seu tempo realizando trabalhos evitáveis, devido à falta de comunicação adequada. Por esse motivo, esse processo de transição de conhecimento, conhecido como *handoff* ou passagem de bastão, desempenha um papel crucial no ciclo de desenvolvimento de produtos e projetos (IMRAN, 2018), envolvendo a transferência de ideias dos *designers* para os desenvolvedores (FENG; ZHANG, 2023).

#### 2.1.3.1 Definições autorais

Durante o texto também serão utilizados os conceitos de *pré-handoff* e *pós-handoff*, que foram definidos devido às necessidades dessa pesquisa. O *pré-handoff* é o momento que antecede a entrega do protótipo, quando o *designer* ainda está trabalhando na solução, realizando o processo de UX. É nesta etapa que é aconselhável discutir questões sobre viabilidade técnica e a inclusão de cenários alternativos - como telas de erro e outras interações fora do que comumente é chamado de caminho feliz, evitando retrabalhos.

No *handoff* propriamente dito, ocorre a transferência do projeto do *designer* para o desenvolvedor, seja em formato de protótipo, *design* de interface (somente as telas) ou fluxograma. Nessa fase, detalhes como a organização do arquivo na ferramenta - como o Figma (FIGMA, 2024) - são importantes, garantindo que o desenvolvedor possa facilmente encontrar as informações relevantes e discernir entre o que é parte do *design* final e o que é experimental.

Já no estágio do *pós-handoff*, o desenvolvedor assume a responsabilidade de traduzir o *design* em código, trabalhando no desenvolvimento do projeto. É nesse momento que o *designer* deve acompanhar o projeto e esclarecer eventuais dúvidas do desenvolvedor sobre a solução. É também nessa fase que há a validação no final do desenvolvimento, para verificar se o que foi desenvolvido está de acordo com o protótipo.

### 2.1.4 Metodologia ágil

A metodologia ágil é um conjunto de estratégias e práticas para o desenvolvimento e gestão de projetos, que busca dar maior velocidade à conclusão de processos e tarefas. Isso porque a coordenação entre clientes e equipe de engenharia de *software* leva a um alto nível de adaptação com quaisquer requisitos adicionais e incentiva a resposta rápida ao cliente. Uma característica importante do ágil é o seu suporte à cultura de mudança com o ideal de que a mudança terá um impacto positivo no desenvolvimento de produtos (ZAYAT; SENVAR, 2020).

Após as entrevistas qualitativas - que serão detalhadas na metodologia - ficou claro que a agilidade desempenha um papel fundamental na realidade dos times de desenvolvimento. Os relatos dos entrevistados destacaram sua presença, especialmente através do método Scrum. Dessa forma, a agilidade tornou-se um aspecto importante a ser compreendido e analisado neste estudo.

#### 2.1.4.1 Scrum

Segundo (HEMA et al., 2020), o Scrum é uma abordagem ágil incremental e de natureza iterativa usada para gerenciar o trabalho. A aplicação do scrum é feita por meio de cerimônias e o *software* é desenvolvido por partes nas chamadas *sprints*, que são um espaço de tempo de duas a quatro semanas com objetivos bem definidos.

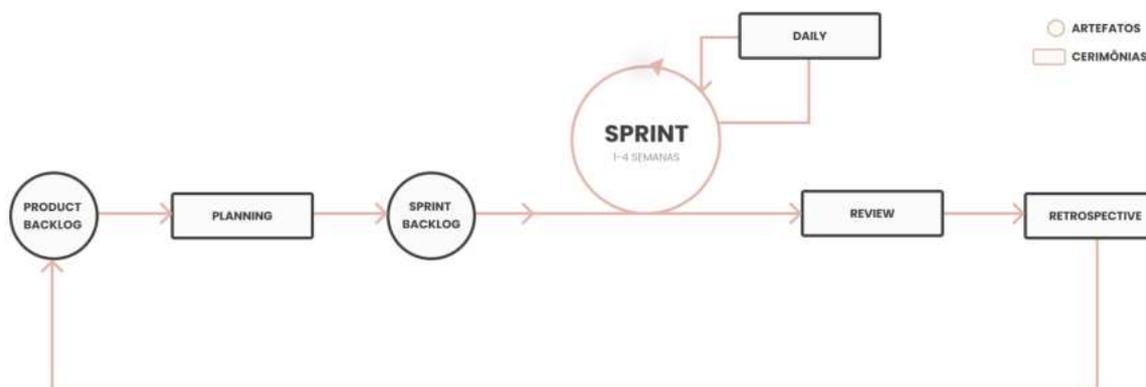
As principais cerimônias do Scrum são: *Sprint Planning*, que é a reunião de definição do que será feito na *sprint*; *Daily*, que é a reunião com objetivo de manter a coordenação e alinhamento entre a equipe em relação ao que cada um está fazendo; *Sprint Review*, que é a reunião que deve incluir toda a equipe e *stakeholders* para revisão do que foi feito na *sprint* e *Sprint Retrospective*, que é uma oportunidade de examinar os obstáculos que foram enfrentados e o que poderia ser melhor na próxima *sprint*. Essas cerimônias, bem como os artefatos, podem ser encontrados na Figura 1.

Conforme descrito por (LEAU et al., 2012), no contexto do desenvolvimento ágil, as fases do ciclo de vida de desenvolvimento são revisitadas várias vezes ao longo das múltiplas iterações, como é o caso das *sprints* no framework Scrum. Em contraste com o SDLC convencional, no qual um único modelo de processo é implementado, o ciclo de vida do desenvolvimento ágil é dividido em incrementos ou iterações menores. Cada um desses incrementos aborda todas as fases convencionais do desenvolvimento de *software*.

A Figura 2 ilustra as fases do ciclo de vida do desenvolvimento de *software* (AWS, 2021). Nesse contexto, dentro de cada *sprint*, esse ciclo ocorre, mas em vez de desenvolver todo o *software* de uma só vez, partes dele são desenvolvidas em cada iteração. Essa abordagem incremental permite entregas frequentes e iterativas, garantindo que o produto seja refinado continuamente com base no *feedback* e nas necessidades do cliente.

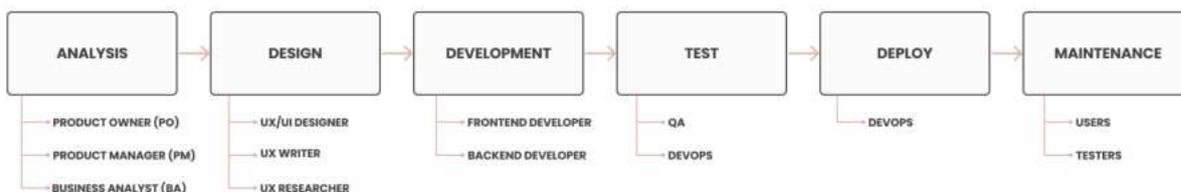
O Scrum materializa o ciclo de vida do desenvolvimento de software e torna suas fases

Figura 1 – Cerimônias e artefatos do Scrum



Fonte: Adaptada de Maria Delos Santos, Jose (Agile Scrum Ceremonies & Meetings, 2024)

Figura 2 – Ciclo de vida do desenvolvimento de software



Fonte: Adaptada de Yağmur Derviş, İlayda (A Deep Dive into the Software Development Life Cycle (SDLC), 2023)

tangíveis por meio de um quadro de tarefas usado para organizar o trabalho durante a *sprint*, como ilustrado na Figura 3. Dentro de cada *sprint*, é estabelecido o *backlog* dela, onde as histórias são concentradas na coluna "*To Do*". À medida que o time seleciona tarefas para execução, elas são movidas para as colunas "*In Progress*" e "*Done*". Para representar mais explicitamente as fases do SDLC, algumas extensões do quadro Scrum incluem colunas adicionais, como "*Testing*", "*Homologation*" e "*Code Review*". Essas colunas permitem um acompanhamento mais detalhado do progresso das histórias ao longo das diversas etapas do ciclo de vida do desenvolvimento de *software* durante a *sprint*.

## 2.2 TRABALHOS RELACIONADOS

Nesta seção, será apresentada uma visão geral dos trabalhos que abordam os desafios do *handoff*, destacando as principais descobertas, metodologias e contribuições desses

Figura 3 – Scrum board



Fonte: Adaptada de Prokopets, Marie (The Top 10 Scrum Boards You Should Be Using, 2024)

estudos. Ao examinar o que foi investigado e proposto anteriormente, será possível identificar lacunas ou oportunidades para contribuir com novas soluções.

### 2.2.1 Importância do *handoff*

Em primeiro lugar, é pertinente ressaltar que o processo de *handoff*, ou transferência de conhecimento, é um tema de grande relevância em diversas áreas. Por exemplo, na área médica, o *handoff* é justamente o momento da passagem de caso de um responsável para outro durante a troca de plantão. Várias publicações, como (JR et al., 2016; DESMEDT et al., 2021), se dedicaram a investigar os desafios e soluções adotadas, porém mesmo em campos consolidados como a medicina, esse processo pode apresentar desafios significativos. No âmbito do desenvolvimento de *software*, onde novas profissões e necessidades emergem constantemente, o desafio é ainda mais acentuado, devido à relativa imaturidade geral da profissão e da área de tecnologia.

Nesse contexto, a presença de vários estudos dedicados a investigar os desafios enfrentados durante o *handoff* reflete a importância desse tema na prática. Vários trabalhos têm se dedicado a explorar esse processo, como descrito por (LEINONEN; ROTO, 2023), onde evidencia-se que a transferência de informações entre *designers* e desenvolvedores desempenha uma função importante nos projetos de criação de serviços digitais.

### 2.2.2 Desafios do *handoff*

Tendo em vista a criticidade e importância do momento do *handoff*, é esperado que esse processo apresente uma série de desafios. Eles têm sido amplamente estudados e soluções têm sido desenvolvidas e refinadas ao longo do tempo para facilitar o ciclo de desenvolvimento. Tais soluções emergem tanto no âmbito acadêmico quanto no comercial. Nesta subseção, discutiremos os desafios identificados e as soluções propostas até o momento.

Um desafio recorrente em diversos estudos é a insuficiente colaboração entre *designers* e desenvolvedores. Muitos autores de fato reconhecem essa necessidade de cooperação e identificam sua ausência como uma questão problemática. De acordo com (BABICH; WHITE, 2017), nem os *designers* nem os desenvolvedores sozinhos são capazes de compreender plenamente o panorama por si só, destacando assim a importância da colaboração mútua para a criação de um *software* de excelência.

Além disso, no artigo (FENG; ZHANG, 2022), os autores sustentam que o processo de concepção tem evoluído para uma abordagem mais colaborativa, porém, ressaltam que a colaboração pode ser muito mais do que convidar outros *stakeholders* para um arquivo compartilhado. A proposta central do artigo é estabelecer um ambiente de co-criação no qual ambos os profissionais possam trabalhar simultaneamente e na mesma plataforma. No entanto, um ponto de limitação é que os profissionais podem perceber esse ambiente como excessivamente aberto e transparente, onde o processo de trabalho de um (como a geração de ideias de *design*) podem impactar o outro. Portanto, embora a colaboração seja incentivada através dessa co-criação, ela enfrenta desafios relacionados a conflitos de interesses e preferências de trabalho.

Já em (IMRAN, 2018), o autor se dedica a aprimorar o processo de *handoff* através da implementação de uma lista de verificação de ações destinada a facilitar essa transição. Embora a aplicação da lista tenha sido bem-sucedida no caso específico discutido no artigo, sua eficácia não é garantida em outros contextos. O estudo foi conduzido em parceria com uma empresa de *software* e a lista desenvolvida em conjunto com os desenvolvedores envolvidos. Dessa forma, as constatações e os resultados podem ser enviesados por essa empresa e esse projeto em particular, o que pode dificultar a generalização das conclusões e o uso dessa lista em outros contextos de desenvolvimento de *software*.

Explorando outros desafios, (MAUDET et al., 2017) apresentam um estudo sobre as dificuldades enfrentadas por *designers* e desenvolvedores ao colaborarem em projetos de sistemas interativos. Os pesquisadores conduziram três estudos para analisar as causas e consequências das falhas de *design*. Eles identificaram três principais tipos de falhas: falta de informações, cenários extremos e restrições técnicas. Constatou-se que o atual fluxo de trabalho resulta em retrabalho desnecessário, com os *designers* criando documentos de *design* redundantes e os desenvolvedores precisando recriá-los em suas próprias ferramen-

tas, muitas vezes resultando em inconsistências de UI com o protótipo original. Buscando resolver o problema, o artigo fornece quatro sugestões para o *design* das ferramentas de apoio à colaboração entre *designers* e desenvolvedores.

Mais recentemente, em 2023, (LIN; KNEAREM; GIESING, 2023) também abordaram os problemas de transição de informação e trouxeram a problemática de que algumas decisões são perdidas quando os desenvolvedores se esforçam para interpretar e implementar a intenção original do *designer*. Para resolver esse problema, eles criaram uma ferramenta chamada *Relay* que utiliza modelos de interface do usuário para capturar a intenção do *design* e gerar um modelo de UI diretamente em código. Essa ferramenta ultrapassou o âmbito acadêmico e atualmente conta com um *plugin* no Figma que é utilizado por aproximadamente 28 mil usuários. Diferentemente das soluções anteriores, que foram sugestões de *design* de ferramentas e de processos, a solução que deram aos problemas de comunicação no *handoff* foi a criação de uma ferramenta que permite a transferência instantânea de componentes da UI do Android entre *designers* e programadores.

#### 2.2.2.1 Ferramentas comerciais

Como visto anteriormente, alguns estudos citavam *plugins* e soluções emergentes que surgem constantemente na tentativa de simplificar o processo. Porém, devido ao rápido avanço da tecnologia, é difícil se manter atualizado sobre as ferramentas utilizadas pelo mercado, principalmente o Figma, que ocupa uma posição de liderança no mercado das ferramentas escolhidas pelos times de desenvolvimento de *software* atualmente (VILLAIN; SILVEIRA, 2023).

Com 1/3 de desenvolvedores como usuários ativos semanais do *Figma*, a ferramenta otimizou o processo de trabalho técnico implementando o *Dev Mode* (BLOG, 2024). Com a proposta de "apoiar uma variedade de equipes, da ideia ao código", a nova funcionalidade permite os desenvolvedores inspecionarem os *designs* de forma facilitada, copiar linhas de código relevantes e até trazer o *design* para mais perto do código através de uma extensão do Figma para o Visual Studio Code.

Além disso, a atualização apresenta novas funcionalidades voltadas especificamente ao aprimoramento do processo de *handoff*: indicadores de telas prontas para o desenvolvimento definidas pelo *designer*, comparações de versões de telas que foram modificadas além de componentes de espaçamentos e comentários mais explícitos para o *designer* se comunicar de forma mais fácil e o desenvolvedor ter por onde se guiar.

No entanto, embora tenha conquistado rapidamente a popularidade nas redes sociais e despertado o interesse dos times de desenvolvimento, surgiram incertezas sobre a natureza paga ou gratuita da funcionalidade, o que gerou desânimo em relação à possível necessidade de enfrentar uma curva de aprendizado para compreender o *Dev Mode* para não tê-lo acessível em um futuro próximo.

### 2.2.3 Adaptações do Scrum

Além de analisarmos as contribuições e lacunas dos trabalhos citados, é interessante também olharmos para aqueles que se dedicaram a adaptar ou estender o framework ágil Scrum. Especificamente, é relevante analisar as principais razões por trás de cada adaptação e determinar se alguma delas é pertinente dentro do contexto do processo de *handoff*.

O uso de métodos ágeis se tornou uma prática comum para times de desenvolvimento de *software*. Como um dos mais famosos, o Scrum tem sido foco de numerosas adaptações e modificações. No estudo (HRON; OBWEGESER, 2018), o Scrum foi escolhido como uma "janela" para o universo ágil, devido à sua ampla difusão e aceitação prática. Os pesquisadores conduziram uma análise de 31 artigos relevantes que descreviam adaptações do Scrum para atender a diferentes circunstâncias. Eles afirmaram que "Frameworks de desenvolvimento ágil e métodos são tipicamente não implementáveis sem adaptações para as circunstâncias únicas de cada ambiente de desenvolvimento".

A conclusão extraída do estudo anterior sugere que a metodologia de desenvolvimento e as técnicas devem ser ajustadas às necessidades específicas de cada contexto. No entanto, na prática, observa-se uma ausência de adaptações ou extensões direcionadas para solucionar as dificuldades encontradas durante o processo de *handoff*. Além disso, as adaptações identificadas não se concentram na perspectiva de promover práticas colaborativas entre as equipes.

#### 2.2.3.1 Proposta

Embora algumas iniciativas tenham sido tomadas para mitigar os problemas de *handoff*, como a criação de *checklists* para ajudar os *designers* a preparar um bom arquivo de *handoff* ou o desenvolvimento de ferramentas que geram a interface diretamente a partir dos arquivos do Figma, essas soluções muitas vezes carecem de personalização e foco nas particularidades de cada equipe e projeto.

Além disso, apesar de identificar que os desafios muitas vezes são consequências de problemas de comunicação, o foco das soluções em si não é na melhora dos pontos de contato e comunicação entre os dois agentes, mas sim soluções mais pontuais para o momento do *handoff*.

No Capítulo *Developer Handoff* do Livro "*A Guide to UX Design and Development*" (GREEN; LABRECQUE, 2023), embora seu tema seja o *handoff*, o autor defende que esse processo não acontece no momento que o protótipo recebe aprovação e é transferido aos desenvolvedores, mas na realidade quando o relacionamento entre *designers* e desenvolvedores é formado, o que geralmente acontece antes do processo de *design*. E apesar de reconhecer que não existe uma "receita" ou um fluxo de trabalho padronizado para o *handoff*, o autor traz algumas etapas a serem consideradas nesse momento de transferência

como: documentar o *design*, organizar o protótipo, nomear todos os elementos, fornecer especificações do projeto e exportar todos os ativos necessários.

Todos os pontos acima são importantes e realmente etapas a se considerar em um momento de *handoff*, e com a prática, muitos deles se tornam naturalmente hábitos. Como o próprio autor cita "À medida que o *design* começa a ganhar vida, o envolvimento dos desenvolvedores pode render benefícios significativos", acreditamos que o cerne da questão não reside simplesmente em seguir dicas ou práticas específicas, seja do *designer* ou do desenvolvedor, mas sim, na criação da relação sólida e colaboração entre ambos.

Tendo em vista essas lacunas dos trabalhos atuais, o objetivo desse estudo é desenvolver uma solução que seja facilmente incorporada no dia a dia de trabalho dos *designers* e desenvolvedores, estimulando a colaboração frequente entre ambos.

### 3 METODOLOGIA

Inicialmente, o foco deste estudo estava na análise do processo e das práticas realizadas durante o *handoff*, incluindo os desafios associados a ele. Entretanto, ao empregar uma abordagem metodológica adaptável, conforme representado na Figura 4, observou-se que as questões enfrentadas nessa fase têm suas raízes em problemas mais amplos de comunicação e alinhamento, que permeiam todo o ciclo de desenvolvimento.

Figura 4 – Método de pesquisa utilizado



Fonte: Elaboração própria

Após revisão da literatura, desenvolvemos uma matriz CSD, com objetivo de reunir todas as certezas, com o intuito de identificar os principais pontos de certezas, suposições e dúvidas, além das áreas passíveis de melhoria no processo, como ilustrado na Figura 5.

Dentre as hipóteses identificadas na Figura 5, destacam-se as seguintes:

- A comunicação entre o *designer* e o desenvolvedor se restringe ao momento do *handoff*;
- Os desenvolvedores valorizam um *handoff* bem estruturado para reduzir a necessidade de comunicação direta com o *designer* durante o processo de desenvolvimento;
- A ausência de um processo estruturado de *handoff* resulta em retrabalho e interpretações equivocadas da solução.

Com base nas suposições e hipóteses levantadas, optou-se por conduzir dois estudos distintos: o primeiro com uma abordagem mais quantitativa, e o segundo com uma abordagem mais qualitativa, sob forma de entrevista semiestruturada.

Na pesquisa quantitativa, o foco estava em validar estatisticamente as hipóteses, sem necessariamente entender as motivações por trás das respostas. A partir da análise dos

Figura 5 – Matriz de Certezas, Suposições e Dúvidas

Certezas			Suposições			Dúvidas		
Não existe um processo padronizado de handoff	A comunicação dev/designer muitas vezes só acontece no momento do handoff em si	Existem ferramentas e funcionalidade que facilitam o handoff (figma devmode, relay)	Alguns desenvolvedores podem ignorar o handoff e não ter utilidade suficiente	Ele é visto como perda de tempo/mais trabalho para alguns designers	Os designers fazem o processo de handoff ao final do processo de UX e assim pode deixado de lado	Empresas maiores possuem um processo de handoff mais estruturado?	Quais problemas podem surgir em um projeto ao não fazer handoff?	Como otimizar o processo de handoff e descobrir o que funciona melhor para sua equipe?
O figma for devs veio para facilitar muito a questão relacionada a espaçamentos e tamanhos então o handoff manual pode ser mais focado na questão de interação do design e especificações de negócio.	Ele pode ser composto por vários elementos ao mesmo tempo como componente no Figma + conversas no chat + reuniões de alinhamento	O handoff é útil como forma de documentação e não necessitar da presença do design e dev criadores para entendimento do projeto	Os desenvolvedores preferem um handoff ao contato síncrono com o designer	Existem funcionalidades e ferramentas que os devs/designers desconhecem ou não sabem utilizar	Os resultados de pesquisas acadêmicas podem ser relevantes mas às vezes não são acessíveis à "comunidade do mercado"	Como manter um processo de handoff em um projeto com muita variação de fluxos?	Existem padrões de handoff dentro de um time ou cada designer o faz como acha melhor?	Como os desenvolvedores enxergam sua participação no processo de criação de páginas e fluxos?
Não ter um handoff estruturado traz retrabalho e maiores esforços de comunicação síncrona	Em muitas empresas o layout final tem que ser validado pelo designer após o desenvolvimento	Desenvolvedores deveriam ser envolvidos mais cedo no processo de criação	Mudanças de interface no meio do projeto atrapalham o desenvolvimento	Testes unitários de interface no front tendem a ter menos adesão e menos priorização	Algumas inconsistências do que foi desenvolvido e o que foi prototipado poderiam ser evitadas com testes de interface	Quais os pontos de dor do designer no processo do handoff?	Quais os pontos de dor do desenvolvedor no processo do handoff?	Quais problemas são evitados por um bom handoff?
						Existem problemas de comunicação e alinhamento no handoff?	Como adaptar o handoff à sua realidade?	Como as pessoas mais fazem handoffs no dia a dia? Qual seu handoff ideal e qual seu real?

Fonte: Elaboração própria

dados desse formulário quantitativo, por sua vez, o roteiro da entrevista qualitativa foi estruturado, focando em explorar informações subjetivas e entender o contexto dos desafios e práticas relacionadas ao processo de *handoff* em maior profundidade.

### 3.1 ESTUDO UM: PESQUISA QUANTITATIVA

Como mencionado anteriormente, o objetivo da pesquisa quantitativa residia na validação estatística das hipóteses, sem necessariamente compreender as motivações por trás das respostas. Neste momento, o foco era mapear quantitativamente o comportamento dos *designers* e desenvolvedores no dia a dia das equipes.

A prioridade foi conhecer seus processos de trabalho, explorar as ferramentas utilizadas, cerimônias estabelecidas e características dos times. Foram investigados os padrões de comunicação e interação entre os *designers* e desenvolvedores, a frequência com que ocorrem os *handoffs*, além da regularidade da validação das entregas e outros pontos de contato entre esses dois atores.

#### 3.1.1 Recrutamento

Para atrair participantes para a pesquisa quantitativa, o *link* do formulário do Google foi compartilhado via *e-mail* para estudantes universitários e divulgado em canais dentro

de empresas de tecnologia. O público alvo eram *designers*, desenvolvedores e outros profissionais que desempenham um papel importante no processo de *handoff*.

### 3.1.2 Participantes

Foram obtidas 95 respostas no formulário, distribuídas conforme a Tabela 1.

Tabela 1 – Descrição de participantes da pesquisa quantitativa

Cargo	Quantidade
UX/UI Designer	31
Full Stack Developer	30
Frontend Developer	4
Backend Developer	13
Outros (QAs, PM, PO, SAP)	17

Fonte: Elaboração própria

### 3.1.3 Procedimento

O Google *Forms* foi utilizado como ferramenta principal, organizando as perguntas em seções distintas. Existia também a opção de responder em português ou inglês, visando à inclusão de profissionais de empresas internacionais. O roteiro completo das perguntas pode ser encontrado no anexo A.

A primeira seção, intitulada "Sobre você", foi projetada para todos os profissionais participantes e englobou aspectos como tempo de atuação na profissão, nível de senioridade e composição do time (mistos com desenvolvedores e *designers* juntos, ou times separados).

A segunda seção era condicionada a partir da resposta da profissão do participante e continha perguntas de acordo com sua função. Na seção "Ponto de vista do *designer*", foram explorados aspectos como o grau de envolvimento e frequência de interações com os desenvolvedores, detalhando quando e como essas comunicações ocorriam.

Além disso, também foi investigada a frequência de participação no *handoff*, validação do projeto através do protótipo, tipos de entregáveis fornecidos aos desenvolvedores (*design* de interface, protótipo interativo e documentação), ferramentas utilizadas e principais desafios enfrentados nesse processo. Na seção "Ponto de vista do desenvolvedor", o foco estava em aspectos muito semelhantes aos citados anteriormente, mas adaptados ao contexto dos desenvolvedores.

Para os demais profissionais participantes, foi oferecida uma seção de perguntas abertas, incentivando-os a compartilhar suas percepções sobre os desafios enfrentados no relacionamento entre *designers* e desenvolvedores.

Além disso, uma seção destinada a sugestões e comentários foi incluída, bem como uma pergunta direcionada àqueles interessados em participar das entrevistas qualitativas posteriores.

### 3.1.4 Coleta de dados

Para a coleta de dados da pesquisa quantitativa, a ferramenta do Google Forms foi utilizada, pois ela já conta com sua própria área de respostas que só os editores do formulário têm acesso. Esses dados podem ser visualizados por meio da interface da própria ferramenta ou através de uma planilha do Google Sheets.

### 3.1.5 Análise

Antes do momento da análise os dados, o formulário foi encerrado para prevenir a entrada de novas respostas e assegurar a consistência estatística dos dados. Como a maioria das perguntas da entrevista quantitativa eram múltipla escolha, elas foram o foco inicial.

A análise das porcentagens das opções de respostas foi feita diretamente da interface do *Forms*, porém, para facilitar a compreensão e comparação dos dados, os gráficos de setores foram convertidos em gráficos de barras, organizando as porcentagens de forma decrescente.

Para as questões que permitiam múltiplas seleções - *checkbox* - e as perguntas abertas, uma análise detalhada foi feita individualmente, visando uma interpretação mais aprofundada das intenções e opiniões dos respondentes. Para centralizar gráficos, dados e *insights* mais importantes, a ferramenta Notion, que é um recurso de anotações e produtividade (NOTION, 2024), foi utilizada.

## 3.2 ESTUDO DOIS: PESQUISA QUALITATIVA

A partir da análise dos dados obtidos na pesquisa quantitativa, foi possível observar que um dos principais desafios no processo de *handoff* é a carência de colaboração e comunicação eficaz entre as equipes de *design* e desenvolvimento. Essa percepção foi um momento crucial que levou a uma reorientação do foco da pesquisa. Enquanto anteriormente a fase de transição de conhecimento entre *designers* e desenvolvedores era o foco principal, agora a investigação foi ampliada para abranger todo o ciclo de desenvolvimento de *software*.

Por meio das entrevistas semiestruturadas, foram explorados em maior profundidade os desafios identificados na etapa anterior, com objetivo de compreender mais detalhadamente os processos e dificuldades envolvidos no desenvolvimento de *software* de forma abrangente. Além disso, foi investigado como cada equipe lida com os desafios inerentes ao processo, visando uma compreensão mais ampla da dinâmica colaborativa entre equipes.

### 3.2.1 Recrutamento

Para a seleção dos participantes para a fase de entrevistas, foi disponibilizada, ao término da pesquisa quantitativa, uma seção para que os interessados pudessem voluntariamente fornecer seus endereços de *e-mail*. Essa estratégia foi adotada com o intuito de viabilizar a participação na etapa subsequente do estudo, porque os participantes foram recrutados com base nessa lista.

### 3.2.2 Participantes

Foram recrutados oito participantes para a fase qualitativa da pesquisa, representando uma variedade de experiências relevantes para o escopo do estudo. A distribuição desses participantes está detalhada na Tabela 2.

Tabela 2 – Descrição de participantes das entrevistas

Pessoa	Cargo	Senioridade
P1	UX/UI Designer	Junior
P2	UX/UI Designer	Pleno
P3	UX/UI Designer	Pleno
P4	UX/UI Designer	Senior
P5	Full Stack Developer	Pleno
P6	Full Stack Developer	Senior
P7	Frontend Developer	Pleno
P8	Frontend Developer	Senior

Fonte: Elaboração própria

Um dado interessante sobre os participantes é que um deles trabalha em uma empresa em Londres, portanto alguns dos aprendizados e experiências relatados nessa pesquisa não se limitam somente ao Brasil.

### 3.2.3 Procedimento

Inicialmente, foram enviados convites por *e-mail* para pessoas interessadas no tema da pesquisa, que estavam dispostas a contribuir e também a receber os resultados do estudo. Em seguida, um roteiro de perguntas foi elaborado com base na literatura e nos dados da pesquisa quantitativa. O roteiro completo das perguntas pode ser encontrado no anexo B.

Esse roteiro serviu como base para as entrevistas, abordando diferentes aspectos do ciclo de desenvolvimento de software, como *pré-handoff*, *handoff* e *pós-handoff*. A escolha da abordagem semiestruturada para as entrevistas permitiu adaptar as perguntas conforme a necessidade e explorar mais a fundo as respostas dos participantes. Além disso, essa abordagem permitiu que as perguntas relativas a metodologias ágeis, como o Scrum,

surgissem de forma natural durante as conversas, mesmo que esse não fosse um ponto focal da entrevista.

### 3.2.4 Coleta de dados

Durante a fase das entrevistas, elas foram conduzidas de forma síncrona utilizando a plataforma Google *Meet*. Cada entrevista contava com a presença de dois facilitadores: um condutor, encarregado de conduzir a entrevista, e um observador, cuja responsabilidade era registrar os pontos principais das respostas fornecidas a cada pergunta e destacar os aspectos-chave da discussão.

Optou-se por não gravar esses momentos das entrevistas para os participantes se sentirem à vontade ao compartilharem suas vivências e exemplos de documentos de *handoff*.

### 3.2.5 Análise

Após a condução das oito entrevistas, foi realizada a análise das informações coletadas. Utilizou-se a metodologia conhecida como *Grounded Theory*, ou teoria fundamentada, conforme discutido por (THORNBERG; PERHAMUS; CHARMAZ, 2014), como abordagem principal para categorizar e compreender os dados.

A teoria fundamentada é um método que estrutura o processo de pesquisa e, ao mesmo tempo libera o pesquisador para explorar os dados e desenvolver novas análises teóricas a partir deles. Esse método permitiu a identificação de padrões e tendências nos relatos dos participantes, facilitando a elaboração de propostas para adaptações no ciclo de vida do desenvolvimento de *software*.

## 4 RESULTADOS

Ambos os estudos desempenharam papéis fundamentais na identificação e categorização dos desafios inerentes ao processo de *handoff* entre equipes de *design* e desenvolvimento.

No primeiro estudo, evidenciou-se que muitos dos obstáculos enfrentados derivam da ausência de colaboração efetiva e comunicação profunda entre as equipes envolvidas, particularmente no que concerne ao alinhamento de expectativas e viabilidades técnicas.

Por sua vez, o segundo estudo complementou e reforçou as conclusões obtidas na pesquisa quantitativa. Além disso, buscou-se aprofundar a compreensão sobre como os entrevistados lidavam com os desafios identificados. Especificamente, investigou-se as práticas adotadas por suas equipes para solucionar ou mitigar os problemas evidenciados durante esse processo de transição de trabalho entre equipes.

### 4.1 ESTUDO UM: PESQUISA QUANTITATIVA

Através da pesquisa quantitativa, algumas das hipóteses levantadas foram validadas e novas foram identificadas para serem investigadas posteriormente na entrevista qualitativa.

#### 4.1.1 Contexto profissional

Ao todo, foram contabilizadas a participação de 47 desenvolvedores, 31 *designers* e 17 participantes de outras áreas profissionais como QAs, *Product Managers* e outros. Dado o caráter voluntário da participação no formulário, a capacidade de controlar o equilíbrio entre as profissões dos respondentes foi limitada. No entanto, a disparidade não foi significativa, o que possibilitou explorar algumas distinções entre esses dois protagonistas.

A primeira seção do questionário visava compreender melhor a realidade profissional dos participantes. Em relação ao nível de senioridade, a seguinte distribuição foi observada: 8,7% são estagiários, 28,3% júniores, 21,7% plenos, 31,5% sêniores, 5,4% *Leads* e 4,3% *Staffs*. A predominância de profissionais sêniores pode ser atribuída ao seu maior interesse em processos de trabalho, em comparação com aqueles no início de suas carreiras, que podem ter menos experiência para compartilhar.

Além disso, 59% (equivalente a 54 pessoas) trabalham em times mistos, com desenvolvedores e *designers* juntos. Em contraste, 35% (32 pessoas) trabalham em equipes separadas, enquanto os outros 6% (9 pessoas) relataram a ausência de *designers* em seus times. Esses resultados sustentam a premissa de que a colaboração entre *designers* e desenvolvedores é de extrema importância no cotidiano por, normalmente, pertencerem ao mesmo time.

#### 4.1.2 Interação entre *designers* e desenvolvedores e seus meios de comunicação

A segunda parte da pesquisa era destinada a conhecer as perspectivas tanto do designer quanto do desenvolvedor, apresentando perguntas semelhantes em ambas seções, porém adaptadas à realidade de cada profissão. Antes de explorar mais especificamente sobre o *handoff*, optou-se por investigar sobre a interação e comunicação entre *designers* e desenvolvedores.

Dos *designers* entrevistados, 86,7% afirmaram manter contato diário ou semanal com os desenvolvedores dos projetos em que estão envolvidos. Em contrapartida, apenas 44,7% dos desenvolvedores relataram ter contato diário ou semanal com designers. Embora inicialmente pareça que os *designers* estão em maior interação com os desenvolvedores do que o contrário, esse dado pode ser justificado devido à natureza do trabalho de cada profissional: a maioria das demandas dos *designers* é encaminhada para algum desenvolvedor implementar posteriormente, enquanto nem toda demanda do desenvolver teve necessariamente algum protótipo ou interface pensada anteriormente por um *designer*. Além disso, notavelmente, 31,9% dos desenvolvedores declararam não ter nenhum contato com *designers*. Essa diferença tornou-se compreensível ao analisar individualmente cada resposta, revelando que a maioria dos que relataram não ter contato com designers são especializados em *backend*, que é o *software* que está por trás das aplicações desenvolvidas.

Quanto aos meios de comunicação utilizados, a maioria dos *designers* respondeu que a comunicação normalmente acontece de forma assíncrona via Slack, Microsoft Teams ou outras ferramentas (80%), além de reuniões (70%). Surpreendentemente, 26,7% dos *designers* mencionaram que o contato ocorre na verdade de forma indireta, sendo intermediado pelo *Product Owner* ou Gerente de Projeto, que transmite as considerações aos desenvolvedores. Esta porcentagem é ainda maior entre os desenvolvedores, com 40,5% indicando que o contato com os *designers* ocorre de maneira indireta. A existência dessa mediação não apenas compromete a transferência de informações essenciais do projeto, mas também distancia a possibilidade de se criar um contato direto e constante entre ambos.

#### 4.1.3 Comunicação restrita ao momento do *handoff*

Para validação da primeira hipótese de que a comunicação entre o designer e o desenvolvedor se restringe ao momento do *handoff*, não só a frequência e ferramentas utilizadas para o contato foram exploradas, mas também os momentos em que ocorre. Descobriu-se que a interação entre os designers e os desenvolvedores acontece em diversos momentos, tais como: reuniões de alinhamento para verificar viabilidades técnicas (80%), *daily*s e outros ritos do time (76,7%), e ao entregar o protótipo final do projeto (63,3%). Entre-

tanto, é importante ressaltar que, dado o formato de múltipla seleção dessa pergunta no formulário, é possível que os participantes tenham marcado todas as opções, mesmo que algumas delas não ocorram com tanta frequência.

Dos desenvolvedores participantes, somente 48,7% afirmaram ter interação com o *designer* ao receber o protótipo final do projeto. A expectativa era de uma porcentagem maior nessa pergunta, considerando que a apresentação do protótipo pelo *designer* ao desenvolvedor é um momento importante para explicar decisões de *design* e interações fundamentais para a experiência do usuário na aplicação, além de alinhar expectativas.

Por isso, a hipótese de que o contato entre *designers* e desenvolvedores ocorre apenas durante o momento do *handoff* foi adaptada pelo reconhecimento de que, na verdade, há interações em diversas fases do projeto, cada uma com sua importância. Assim, alguns momentos passaram a receber mais atenção, como a verificação de viabilidades técnicas - que estão presentes no *pré-handoff* - rituais como *daily*s, *plannings* e retrospectivas, juntamente com a apresentação do protótipo final - o *handoff* - e momentos de validação da experiência do usuário entre o que foi desenvolvido e o que foi prototipado - que ocorrem no *pós-handoff*. Foi decidido que esses momentos distintos seriam explorados durante a entrevista qualitativa.

#### 4.1.4 Desenvolvedores priorizam *handoff* sobre contato

Para validar a segunda hipótese sobre o desenvolvedor valorizar um *handoff* bem estruturado para reduzir a necessidade de comunicação direta com o *designer* durante o processo de desenvolvimento, buscamos entender os desejos e desafios enfrentados por eles.

O dado de que 46,8% dos desenvolvedores relataram receber apenas telas de *design* de interface ao receberem um projeto, sem um protótipo interativo, suscitou a curiosidade para explorar esse aspecto mais a fundo durante a fase qualitativa da pesquisa. Assim, buscou-se entender como os desenvolvedores conectam essas telas isoladas, como estabelecem suas relações e se esse processo representa um ponto de dificuldade para eles.

Quando perguntados sobre o que mais facilitaria seu trabalho na entrega de protótipos para o desenvolvimento, 66% dos desenvolvedores destacaram a importância de um arquivo de protótipo bem organizado como o aspecto mais importante. Em seguida, 63,8% mencionaram a relevância de manter um contato frequente com o *designer* para esclarecer dúvidas, seguido por documentação de *design* (46,8%), guia de estilo bem definido (38,3%) e Design Tokens (17%).

É relevante ressaltar a alta porcentagem de desenvolvedores que priorizam a disponibilidade e o contato constante com o *designer* para esclarecimentos, pois isso desmistifica a ideia de que todos os desenvolvedores preferem trabalhar de forma independente e com pouco envolvimento com a equipe. Portanto, é possível concluir que de fato os desenvolvedores valorizam um *handoff* bem estruturado, mas este fato não exclui a vantagem de

manter um contato mais próximo com o *designer* no dia a dia para esclarecer dúvidas de forma ágil e eficiente.

#### 4.1.5 Efeitos da inexistência do *handoff*

Nas últimas perguntas do formulário, foi possível entender um pouco mais a relação dos *designers* e desenvolvedores com momentos de passagem de bastão, de validação de *design*, e também, mapear os desafios encontrados pelo caminho. O objetivo era validar a hipótese de que a ausência de um processo estruturado de *handoff* resulta em retrabalho e interpretações equivocadas da solução.

Quando perguntados sobre os principais desafios encontrados durante o *handoff*, problemas como a dificuldade de comunicação, falta de conhecimento técnico de código para saber o que seria viável ou não por parte dos *designers*, falta de conhecimento de *design* por parte dos desenvolvedores e falta de documentação adequadas lideraram a pesquisa.

Em relação a frequência com que validam se o que foi desenvolvido está de acordo com o protótipo entregue, 63% dos *designers* disseram validar todas as vezes ou na maioria das vezes, enquanto 23% afirmaram validar algumas vezes e 10% dos *designers* nunca validam. Na pesquisa qualitativa foi viável explorar mais profundamente se existem ritos ou processos definidos para esses momentos de validação e como eles normalmente acontecem.

Na seção de perguntas abertas e sugestões surgiram comentários interessantes sobre essas dificuldades vividas no dia a dia como: "Na minha experiência, quanto mais cedo e mais frequentemente envolver os desenvolvedores no processo de design, menos dor de cabeça lá na frente." Ou ainda: "O mais importante sempre é a proximidade entre os times de design e de desenvolvimento. Quanto mais próximos estiverem, quanto mais trocas tiverem, entendendo que estão projetando juntos, menos atritos terão."

E apesar das sugestões serem de pessoas diferentes e redigidas de maneira distinta, a mensagem transmitida é a mesma. Por fim, uma última observação sobre o *handoff* ressalta: "Não acho que deva existir um momento único de *handoff*. Protótipos não são perfeitos e na minha experiência o que funciona é uma interação mais fluida entre a equipe de dev e design."

Portanto, fica claro que, na verdade, não só a ausência de um processo de *handoff* resulta em retrabalho e interpretações equivocadas da solução, mas principalmente, a falta de alinhamento e comunicação constante entre *designers* e desenvolvedores como um todo.

#### 4.1.6 Conclusão a partir dos dados quantitativos

Com base nos resultados, foi possível validar as hipóteses propostas neste estudo. Ao recapitular os novos *insights* obtidos e considerados para a realização da entrevista

qualitativa, destacam-se:

- **A comunicação entre designer e desenvolvedor não se restringe ao momento do *handoff*:** Os dados mostram que a interação entre *designers* e desenvolvedores ocorre em várias fases do projeto, não se limitando apenas ao momento do *handoff*. Isso sugere que a colaboração é contínua ao longo do processo de desenvolvimento.
- **Os desenvolvedores valorizam um *handoff* bem estruturado, mas este não substitui a necessidade de proximidade com o *designer*:** A pesquisa revela que os desenvolvedores destacam a importância de um *handoff* bem estruturado, mas também valorizam o contato frequente com os *designers* para esclarecer dúvidas. Isso indica que, embora valorizem um processo definido e um arquivo organizado de entregáveis, reconhecem a necessidade de comunicação contínua.
- **A ausência de um processo estruturado de *handoff* pode resultar em retrabalho, mas principalmente, a falta de alinhamento e comunicação entre *designer* e desenvolvedor:** A pesquisa deixou claro que são vários os desafios encontrados, mas a dificuldade de comunicação entre o time corrobora diretamente para retrabalhos e interpretações equivocadas da solução, ressaltando a importância de um alinhamento eficaz entre *designers* e desenvolvedores durante todo o processo de desenvolvimento.

## 4.2 ESTUDO DOIS: PESQUISA QUALITATIVA

Conforme evidenciado nos resultados obtidos da pesquisa quantitativa, constatou-se que os desafios observados durante o momento do *handoff* são, na realidade, consequências de questões que permeiam todo o ciclo de desenvolvimento de *software*.

Durante as entrevistas semiestruturadas, as perguntas foram organizadas de acordo com as fases identificadas previamente - *pré-handoff*, *handoff* e *pós-handoff*. Essa abordagem permitiu uma categorização eficiente, encontrada na Tabela 3, dos desafios de acordo com a etapa em que ocorrem.

### 4.2.1 *Pré-handoff*

Inicialmente, serão abordados os desafios que surgem devido a etapas anteriores ao momento do *handoff*, no chamado *pré-handoff*. Estes desafios estão relacionados a atividades que ocorrem durante a concepção do projeto, antes mesmo da transição formal entre *designers* e desenvolvedores.

Tabela 3 – Desafios percebidos no *handoff* e suas fases de origem

Fase	Desafio
<i>Pré-handoff</i>	Ausência de alinhamento sobre a entrega ideal
<i>Pré-handoff</i>	Ausência de fluxos alternativos
<i>Pré-handoff</i>	Inviabilidade técnica do protótipo
<i>Handoff</i>	Ausência de reunião de apresentação
<i>Handoff</i>	Pouca familiaridade com a ferramenta de design
<i>Pós-handoff</i>	Ausência de validação entre protótipo e <i>software</i> final

Fonte: Elaboração própria

#### 4.2.1.1 Falta de alinhamento sobre a entrega ideal

A quebra de expectativas do desenvolvedor é um desafio evidente decorrente da ausência de comunicação direta entre desenvolvedores e *designers*. Alinhar sobre a entrega ideal que funciona melhor para cada equipe pode evitar a sobrecarga para o *designer* e também facilitar o trabalho dos desenvolvedores.

Essa falta de alinhamento torna-se aparente ao analisar alguns relatos das entrevistadas. Por exemplo, o designer P3 mencionou que utiliza setas e comentários para indicar interações no protótipo, ao invés de fornecer um protótipo interativo, recurso disponível no próprio Figma. De maneira similar, outro designer P1 acredita que os desenvolvedores não utilizam protótipos interativos e, portanto, não os fornecem mais.

Em contrapartida, P8 expressou preferência por receber protótipos interativos em vez de arquivos de *handoff* repletos de especificações detalhadas que podem ser encontradas facilmente na própria ferramenta de *design*. Esse desenvolvedor destacou que é mais benéfico examinar o fluxo das interações e com isso facilitar o processo de desenvolvimento do fluxo do usuário no código. Além disso, o desenvolvedor P6 relatou receber protótipos sem interações e ter que inferir as conexões entre as páginas, fazendo com que, eventualmente, tenha que adicionar comentários no Figma para sanar dúvidas.

Em alguns casos, essa falta de alinhamento é agravada devido à mediação de terceiros no momento do *handoff*. P4 mencionou que a mediação por terceiros dificulta o entendimento das motivações e pode levar ao esquecimento de detalhes importantes para o fluxo do usuário durante o desenvolvimento. Nesse sentido, ela expressou o desejo de realizar reuniões com os desenvolvedores, especialmente para explicar o caminho até a chegada no protótipo.

#### 4.2.1.2 Ausência de fluxos alternativos

Um desafio recorrentemente mencionado pelos desenvolvedores é a ausência de consideração dos fluxos alternativos nos protótipos. Observou-se que, com frequência, os *designs* se concentram apenas no caminho feliz do usuário, deixando de fora situações menos comuns, como cenários de erros. Isso resulta em interrupções e atrasos no processo

de desenvolvimento, já que os desenvolvedores precisam solicitar ao *designer* que inclua esses casos especiais no protótipo. Os desenvolvedores P6, P7 e P8 relataram já terem recebido protótipos que não cobriam adequadamente os fluxos alternativos do projeto.

#### 4.2.1.3 Inviabilidade técnica do protótipo

Além do desafio anterior, alguns desenvolvedores também mencionaram a inviabilidade técnica dos protótipos que recebem, que muitas vezes só é percebida durante o *handoff*, quando o *design*, teoricamente, já está finalizado. O desenvolvedor P7 destacou ainda que, em algumas ocasiões, a inviabilidade técnica é agravada pelo cronograma apertado do projeto. Embora uma solução possa ser tecnicamente viável, sua implementação dentro do prazo estipulado pode ser inviável. Portanto, é muito importante estabelecer uma cultura de consulta ao desenvolvedor durante a fase de concepção do *design*.

### 4.2.2 *Handoff*

Nesta seção, serão discutidos os desafios que têm suas origens no momento específico de transição entre as equipes de *design* e desenvolvimento.

#### 4.2.2.1 Ausência de reunião de apresentação

O principal desafio identificado no momento do *handoff* é a ausência de uma reunião de apresentação do entregável de *design*, o que pode resultar em lacunas de entendimento entre o *designer* e o desenvolvedor, além de não incentivar o processo colaborativo entre esses atores. Muitas vezes, as histórias de desenvolvimento são criadas no *backlog* pelo *Product Owner* ou *Manager*, e o *link* do protótipo é adicionado como anexo, sem necessariamente ocorrer uma reunião de transição.

O designer P4 relatou que, em seu caso, o momento do *handoff* é mediado por terceiros, dificultando muito o entendimento dos desenvolvedores sobre as decisões de *design* tomadas. Ele mencionou que ter uma reunião de apresentação seria ideal, mas isso é difícil devido ao intervalo de tempo entre o *design* e o desenvolvimento, que pode chegar a dois meses. Nesses casos em que a reunião síncrona não é viável, os *designers* poderiam gravar um vídeo explicando os principais detalhes e decisões e adicioná-lo como anexo também. Apesar de não ser o ideal, pode ser funcional.

#### 4.2.2.2 Pouca familiaridade com a ferramenta de design

Outro desafio que surge no momento do *handoff* é a pouca familiaridade do desenvolvedor com a ferramenta de *design*. A maioria dos entrevistados (7 de 8) utilizavam o Figma, e mesmo com sua popularidade, assumir que todos os desenvolvedores o dominam e conseguem utilizar seus recursos da melhor forma é otimismo.

Por exemplo, o desenvolvedor P6 relatou não ter familiaridade com ele e por isso ter dificuldades para encontrar algumas especificações quando recebe o protótipo dos *designers*. Já o desenvolvedor P8 se sente confortável com a ferramenta, mas admite que isso é uma facilidade dele e reconhece que outros no time enfrentam dificuldades, tendo que frequentemente solicitar assistência aos *designers*.

### 4.2.3 Pós-handoff

Por último, serão abordados os desafios que surgem após o momento de *handoff*, denominado *pós-handoff*.

#### 4.2.3.1 Ausência de validação entre protótipo e *software* final

O desafio mais presente na fase do *pós-handoff* é a ausência de um processo de validação entre protótipo e *software* final, gerando inconsistências encontradas entre o que foi desenvolvido em comparação com o protótipo. Tais inconsistências abrangem não apenas aspectos de interface, mas também de interação e fluxo do usuário no *software* resultante.

O designer P4 disse que no seu time não há um processo de validação consolidado, mas expressou o desejo de implementá-lo, pois, nas ocasiões em que validou o produto final, identificou problemas na interface, como desalinhamentos e uso de fontes inadequadas.

Por sua vez, o designer P2 relatou a recente adoção de verificações de UX em seu time, que embora não seja uma etapa formalizada, tornou-se uma prática recorrente de forma natural. Essa rotina de validação tem solucionado não só problemas de interface, mas também de interação. Isso contribui para a criação da experiência desejada para o cliente final e evita significativamente erros em produção.

### 4.2.4 Histórias dos usuários

A partir dos desafios identificados, foram criadas seis histórias de usuários utilizando a ferramenta *User Stories*. Essas histórias têm como objetivo destacar as necessidades específicas de cada ator envolvido no processo de *handoff*: *designers* e desenvolvedores. A finalidade dessas narrativas é guiar o desenvolvimento de soluções que abordem de forma eficaz os desafios identificados durante a transição de conhecimento entre as equipes. As histórias podem ser visualizadas na Figura 6.

## 4.3 EXTENSÃO DO SDLC

A análise feita anteriormente revelou que todos os problemas identificados são resultantes diretos ou indiretos da falta de colaboração e comunicação entre as equipes envolvidas no *handoff*.

Figura 6 – Histórias dos usuários



Fonte: Elaboração própria

Um exemplo desse cenário é a situação relatada pelo designer P3, que geralmente não participa das reuniões diárias com a equipe de desenvolvimento devido à sua natureza técnica. No entanto, ao se distanciar desses encontros, ele percebe uma desconexão com os desenvolvedores. Como resultado, pode ocorrer falta de clareza nas instruções fornecidas durante o repasse do trabalho, resultando em dúvidas que só são recebidas meses depois. Além disso, P3 menciona que, quando os desenvolvedores têm questionamentos, eles tendem a não abordá-lo diretamente, optando pela comunicação por meio de seus gestores, o que gera atrasos no ciclo de desenvolvimento e ainda pode ocasionar um afastamento maior entre *designers* e desenvolvedores, criando atritos desgastantes no cotidiano.

Por outro lado, o desenvolvedor P8 destaca uma experiência positiva em um momento de *handoff* que não se limitou à transferência de arquivos, mas sim contou com uma reunião de alinhamento com o *designer* do projeto. Nesse encontro, as partes colaboraram para encontrar uma solução em conjunto. Para P8, esse tipo de interação é mais valioso

do que documentos elaborados de handoff. P8 defendeu que "Se o time estiver integrado desde o começo, o momento do handoff é mais facilitado porque o desenvolvedor já entende um pouco do processo".

#### 4.3.1 Perspectiva do *Handoff*: *Hands Off*?

Antes de prosseguir, é pertinente destacar que durante esse estudo foi perceptível que, embora amplamente utilizado na indústria de desenvolvimento de *software*, o termo *handoff* pode ser considerado inadequado, pois sugere uma transferência de responsabilidade ou uma separação entre o *designer* e o desenvolvedor, o que não reflete precisamente a natureza colaborativa e contínua do trabalho entre esses profissionais.

Na prática, o processo de *handoff* envolve uma série de interações e colaborações ao longo de todo o ciclo de desenvolvimento, desde a concepção da ideia até a implementação final do produto. Portanto, o termo pode gerar uma interpretação equivocada de que o *designer* se desvincula completamente do projeto após a entrega das especificações, quando na realidade, sua contribuição e envolvimento muitas vezes persistem durante todo o processo.

Por esses motivos, a partir de agora o termo *handoff* não será o foco principal, mas sim, o ciclo de desenvolvimento, entendendo que a comunicação e contato frequente são essenciais para garantir uma entrega de um *software* de qualidade e consistente.

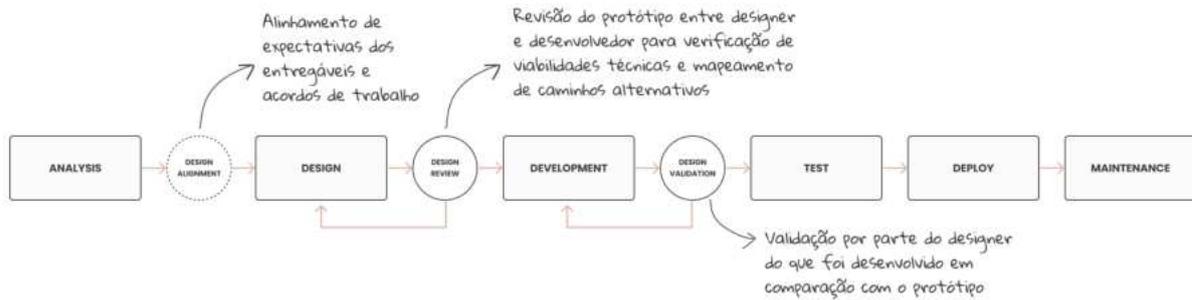
#### 4.3.2 Adaptação do ciclo

A partir dos dados categorizados e da compreensão dos desafios presentes em cada etapa, procedeu-se à avaliação do fluxo de trabalho atual do Scrum e do ciclo de vida do desenvolvimento de *software*, especialmente no contexto das *sprints*, conforme vemos na Figura 2.

Com base na natureza colaborativa inerente ao processo de desenvolvimento observada nas pesquisas, a proposta de extensão do ciclo de vida de desenvolvimento de software (SDLC) desse estudo visa introduzir pontos de contato adicionais entre as equipes em momentos estratégicos do desenvolvimento. Através dos pontos de dor identificados nas histórias dos usuários da Figura 6, são sugeridas três cerimônias, que podem ser reuniões ou contatos assíncronos, para serem integradas ao ciclo de desenvolvimento. Essas cerimônias podem ser observadas na Figura 7.

Conforme é possível observar na figura, as três cerimônias propostas são: *Design Alignment*, *Design Review* e *Design Validation*. É relevante ressaltar que, embora algumas das práticas recomendadas em cada etapa já sejam adotadas por alguns times no cotidiano, o fato de não estarem formalmente definidas no ciclo de desenvolvimento e não serem rotineiramente praticadas pode resultar em seu esquecimento.

Figura 7 – SDLC estendido



Fonte: Elaboração própria

#### 4.3.2.1 *Design Alignment*

Primeiramente, a cerimônia de *Design Alignment* tem como propósito alinhar as expectativas em relação ao entregável na fase final do processo de *design*, visando determinar o material a ser fornecido no momento do *handoff*. Isso inclui considerações sobre a necessidade de arquivos detalhados contendo informações como espaçamentos, margens, entre outros, ou se o acesso a esses detalhes na própria ferramenta de *design* é viável. Além disso, avalia-se o nível de familiaridade do desenvolvedor com a referida ferramenta.

É compreensível que nem sempre é viável realizar esta etapa com o desenvolvedor específico encarregado do projeto, pois, frequentemente, essa designação não está previamente estabelecida. No entanto, é comum que a equipe responsável já esteja definida desde o início, o que permite o alinhamento com o time como um todo.

Além disso, cabe ressaltar que esta etapa está representada de forma pontilhada na Figura 7, pois ela não é essencial em todas as iterações (*sprints*), dado que as equipes geralmente permanecem as mesmas e, conseqüentemente, os acordos de trabalho também. Entretanto, é crucial estar ciente dessa variável sempre que houver uma alteração na composição da equipe de trabalho, como a chegada de novos membros, por exemplo.

Com a adição dessa etapa, o problema da falta de alinhamento sobre a entrega ideal será resolvido, uma vez que os acordos estarão claros desde o início.

#### 4.3.2.2 *Design Review*

A cerimônia de *Design Review* é a fase destinada a ser uma sessão de apresentação do progresso do trabalho do *designer* aos desenvolvedores envolvidos no projeto.

Durante a entrevista, alguns *designers* relataram ter esse momento de apresentação para o time durante o rito da *Sprint Review*. Porém, isso é de fato somente uma revisão do trabalho final do *designer*, ocorrendo no fim da *sprint*.

Já a *Design Review* tem o intuito de ser realizada de forma regular, idealmente cerca de duas vezes por *sprint*, sendo adaptado de acordo com a necessidade do projeto. Durante

essa cerimônia, tanto os desenvolvedores quanto os *designers* têm a oportunidade de identificar em conjunto os possíveis cenários alternativos, além de avaliar a viabilidade técnica das ideias concebidas pelo *designer*.

#### 4.3.2.3 *Design Validation*

Por fim, a etapa de *Design Validation* tem como propósito a validação de UX do *software* desenvolvido. Essa etapa visa evitar inconsistências de UI e interações em produção, bem como reduzir a sobrecarga dos membros da equipe responsáveis pela garantia de qualidade.

Em resumo, na Tabela 4 é apresentado um mapeamento dos problemas levantados com as soluções que se propõem a resolvê-los:

Tabela 4 – Desafios percebidos no *handoff* e propostas de soluções

Desafio	Possível solução
Ausência de alinhamento sobre a entrega ideal	<i>Design Alignment</i>
Ausência de fluxos alternativos	<i>Design Review</i>
Inviabilidade técnica do protótipo	<i>Design Review</i>
Ausência de reunião de apresentação	<i>Design Review</i>
Pouca familiaridade com a ferramenta de design	<i>Design Review</i>
Ausência de validação entre protótipo e <i>software</i> final	<i>Design Validation</i>

Fonte: Elaboração própria

#### 4.3.3 Adaptação do *board* Scrum

Além da introdução de cerimônias no Ciclo de Vida de Desenvolvimento de Software, esse estudo sugere uma pequena modificação no quadro do Scrum utilizado durante as *sprints*. Propõe-se a inclusão de uma nova coluna no quadro, denominada *Design Validation*, ou *UX Validation*, destinada a destacar especificamente o processo de validação de *design* no cotidiano das equipes. Essa adaptação visa integrar de forma mais explícita os momentos de interação entre *designers* e desenvolvedores no fluxo de trabalho regular. A representação desta proposta pode ser visualizada na Figura 8.

Optou-se por não incluir colunas para os momentos de *Design Alignment* e *Design Review*, devido ao entendimento de que esses processos devem acontecer durante o andamento das histórias. Assim, não é necessário representá-los explicitamente no quadro do Scrum.

#### 4.3.4 Análise da aplicabilidade da solução nos contextos iniciais

Após a categorização dos desafios e a proposta da extensão do ciclo de desenvolvimento, os dados das entrevistas qualitativas foram revisados para identificar equipes que

Figura 8 – Scrum Board estendido



Fonte: Elaboração própria

já adotavam cerimônias semelhantes. O objetivo era verificar se as propostas estavam alinhadas com os desafios identificados. O resultado desse mapeamento está disponível na Tabela 5.

Tabela 5 – Uso de cerimônias propostas pelos entrevistados

Pessoa	<i>Design Alignment</i>	<i>Design Review</i>	<i>Design Validation</i>
P1	Não	Sim*	Não
P2	Não	Não	Sim
P3	Não	Não	Não
P4	Não	Não	Não
P5	Sim	Sim*	Sim
P6	Não	Sim*	Sim*
P7	Não	Não	Sim
P8	Não	Sim*	Sim*

Fonte: Elaboração própria

Aqueles que empregam uma cerimônia semelhante à *Design Review* geralmente o fazem de forma não obrigatória, sendo a realização dessas reuniões de progresso entre os *designers* e desenvolvedores condicionada à vontade do *designer* do projeto. Em alguns casos (os marcados com asteriscos), também não são abordados os fluxos alternativos, priorizando-se apenas as viabilidades técnicas.

Quanto à *Design Validation*, os casos marcados com asteriscos indicam situações em que a validação é conduzida pelo próprio analista de qualidade do time. Nessas circunstâncias, foi relatado que esses profissionais tendem a ficar ligeiramente sobrecarregados por terem que realizar essa comparação com o protótipo, além dos testes de funcionalidades habituais.

## 5 TESTES

Para avaliar a eficácia das adições e adaptações implementadas no ciclo de desenvolvimento e no quadro do Scrum, foram realizados testes com duas equipes de projetos, compostas por gerentes de *design* e desenvolvimento, pertencentes a duas empresas juniores da Universidade Federal do Rio de Janeiro: EJCM Consultoria e Fluxo Consultoria.

Ao investigar os desafios enfrentados no contexto dessas empresas, constatou-se que muitos deles eram similares aos que foram observados nas entrevistas com as pessoas que trabalham no mercado. Por isso, o propósito desses testes era validar a relevância das modificações propostas, por meio da coleta de *feedbacks*, identificando possíveis pontos de falha ou áreas de melhoria na solução apresentada.

### 5.1 ENTREVISTAS

Na fase inicial do experimento, foram conduzidas entrevistas com os gerentes e coordenadores de projetos envolvidos no contexto do *handoff*. Durante esse primeiro contato, as conversas seguiram o roteiro do anexo B, com o objetivo de replicar o estudo realizado com empresas do mercado dentro do contexto do Movimento Empresa Júnior (MEJ), a fim de identificar os pontos de dor e os desafios enfrentados, bem como as práticas adotadas durante o *handoff*.

É relevante destacar que nos ambientes das duas empresas em questão os times de desenvolvimento e *design* operam de forma separada, diferentemente da maioria dos casos avaliados no capítulo 4, em que os times eram mistos com *designers* e desenvolvedores. Porém, este fato não interfere em nenhum aspecto da solução, já que ela foca na colaboração entre eles, independentemente dos profissionais pertencerem à mesma equipe ou não.

Além disso, em termos de metodologia ágil, ambas as empresas adotam o Scrum, utilizando *sprints* com duração de uma a duas semanas, variando conforme as necessidades específicas de cada projeto.

#### 5.1.1 EJCM Consultoria

Foram conduzidas entrevistas com o diretor de projetos, dois gerentes de *design* e dois gerentes de desenvolvimento, ambos envolvidos em um processo de *handoff* de um projeto no momento da pesquisa. Eles informaram que não possuem um processo formalizado, no entanto, normalmente a equipe de *design* finaliza o protótipo e o gerente de *design* realiza uma reunião de *handoff* com os desenvolvedores encarregados do projeto. O objetivo dessa reunião é explicar o protótipo e esclarecer quaisquer dúvidas que possam surgir.

É relevante destacar que esse é o único momento em que se concentram as dúvidas e críticas ao protótipo, uma vez que não há nenhum ponto de contato prévio. Isso significa que questões como a avaliação de viabilidades técnicas e o mapeamento de fluxos alternativos dependem da iniciativa do próprio *designer*, como observado em entrevistas anteriores. Além disso, não há etapa de validação pelo *designer* do *software* desenvolvido, apesar de existirem relatos de inconsistências.

A ausência de um ponto de contato estruturado e bem definido tem se mostrado problemática. Por exemplo, os entrevistados relataram situações em que perceberam a ausência de telas e tiveram que retornar para solicitar aos *designers*, resultando em atrasos no fluxo de trabalho. Esse cenário é agravado pela ausência dos *designers* nas reuniões de *handoff*, que normalmente envolvem apenas os gerentes responsáveis. Essa falta de participação pode levar à omissão de detalhes específicos que não foram explicados pelo *designer* que criou o protótipo, além de dificultar a construção de uma relação sólida entre os profissionais.

### 5.1.2 Fluxo Consultoria

Foram conduzidas entrevistas com o coordenador de projetos, três gerentes de *design* e um gerente de desenvolvimento.

O processo na Fluxo se assemelha ao observado na EJCM Consultoria. Quando um projeto engloba tanto a parte de *design* quanto a de desenvolvimento, há uma fase inicial de concepção, na qual os *designers* criam o protótipo, que consiste no *design* de interface com interação, e o fluxograma (Sitemap), que é um arquivo que lista e organiza todas as páginas de uma aplicação. Anteriormente, essa etapa não envolvia os desenvolvedores, mas atualmente os desenvolvedores *backend* também participam da concepção, focando na construção do Modelo Entidade-Relacionamento (MER) e na análise de requisitos. Geralmente, os desenvolvedores *frontend*, responsáveis pela parte visual das aplicações, não são envolvidos no início do processo.

Os entrevistados observaram a existência de uma lacuna na comunicação entre desenvolvedores e *designers* durante o desenvolvimento dos projetos. Eles citaram um exemplo em que os *designers* iniciaram a concepção e o fluxograma sem alinhamento com os desenvolvedores. Os programadores *backend* só entraram no projeto posteriormente, mas nesse momento conseguiram dar *insights* interessantes sobre inconsistências, inviabilidades técnicas e sobre a ausência de algumas funcionalidades.

Eles relataram também um outro desafio que percebiam: a concepção ficava muito grande em comparação com o que seria efetivamente desenvolvido, especialmente quando os desenvolvedores não estavam envolvidos na etapa inicial do processo.

Diante dessas dificuldades, eles decidiram incluir mais pontos de contato no projeto atualmente em desenvolvimento. Uma medida adotada foi a criação de um grupo em um aplicativo de mensagens que reúne todos os desenvolvedores e *designers*. Esse canal tem

contribuído significativamente para melhorar a comunicação, permitindo que os *designers* esclareçam dúvidas sobre viabilidades com os desenvolvedores, sem precisar esperar pelas reuniões.

Outra medida adotada foi a implementação de uma reunião semanal na qual os *designers* apresentam seus protótipos e fluxogramas aos desenvolvedores, visando manter todos atualizados sobre o progresso do trabalho. Durante essas sessões, os desenvolvedores fornecem *feedback* sobre o *design* e identificam possíveis funcionalidades faltantes. Quando discutem com a equipe de desenvolvimento *frontend*, costumam abordar questões relacionadas a inviabilidades técnicas.

Essa reunião semanal de alinhamento e o contato assíncrono, que envolve toda a equipe, reflete a ideia da *Design Review* proposta neste estudo. Esse caso particular corrobora a eficiência dessa prática e como a adição de pontos de contato entre os profissionais pode melhorar o entregável final. A equipe teve a oportunidade de testar e implementar adaptações análogas às sugeridas para a *Design Review*, permitindo uma avaliação prática do impacto das mudanças propostas.

Por fim, eles disseram que não existe etapa de validação do *software* final pelo *designer*, mas compartilharam situações em que ocorreram inconsistências entre o que foi prototipado e o que foi desenvolvido, que só foram percebidas em produção.

### 5.1.3 Conclusão das entrevistas

Essa abordagem de realizar uma entrevista como primeiro ponto de contato permitiu uma comparação direta com os resultados obtidos anteriormente, ajudando a determinar a viabilidade e relevância das propostas dentro do contexto específico das empresas.

Tendo em vista que os desafios encontrados são muito similares àqueles da Tabela 3, foi dado prosseguimento à segunda etapa da validação: a apresentação da proposta de solução.

## 5.2 APRESENTAÇÃO DA PROPOSTA

Inicialmente, enviamos uma apresentação da proposta para os dez participantes do teste, de ambas as empresas juniores, com o objetivo de obterem o material que seria apresentado em reunião posterior para coleta de *feedbacks*. Na reunião, o ciclo estendido proposto foi apresentado em detalhes, bem como as motivações para adição de cada cerimônia. Além disso, também foram explicadas as modificações no quadro do Scrum.

Ao fim da apresentação, houve uma seção de dúvidas para os participantes perguntarem sobre a proposta. Além disso, esse foi um momento em que eles puderam fornecer suas opiniões sobre a eficácia da solução proposta em seu contexto, além de identificar possíveis lacunas e sugerir melhorias. Isso foi essencial para o entendimento do quanto a solução está clara e o que poderia ser melhorado.

Em seguida, um formulário contendo perguntas específicas foi disponibilizado, conforme descrito no anexo C, para que os entrevistados pudessem compartilhar suas percepções de forma assíncrona, caso preferissem.

## 5.3 ANÁLISE DOS RESULTADOS

### 5.3.1 Pontos positivos

Durante a interação com as empresas juniores, houve relatos mais detalhados sobre os problemas enfrentados em projetos que abrangem *design* e desenvolvimento. Todas essas experiências compartilhadas coincidem com desafios previamente identificados na pesquisa, tanto quantitativa quanto qualitativa, conforme delineado na Tabela 3. Esse alinhamento fortalece a motivação inicial, evidenciando o fato de que a identificação das principais lacunas de comunicação e colaboração nos momentos de *pré-handoff*, *handoff* e *pós-handoff*, foi um trabalho bem sucedido.

Além disso, foi possível observar que uma das cerimônias propostas, a *Design Review*, já está sendo implementada por uma das empresas, embora de maneira não-estruturada. Os relatos destacam melhorias significativas no ciclo de desenvolvimento de *software* desde que a interação entre *designers* e desenvolvedores se tornou mais próxima e frequente, através de reuniões e grupos de mensagens assíncronas. Essa implementação prática de parte da proposta em um ambiente de trabalho real e as consequentes melhorias contribuem para validar sua eficiência.

### 5.3.2 Pontos de limitação

Ambas as empresas trouxeram questões de validação com o cliente final e como isso se integraria ao ciclo estendido proposto. Essa questão é particularmente relevante no contexto das empresas juniores, uma vez que nesse contexto são os mesmos profissionais que estão envolvidos em quase todos os processos do fluxo, desde a análise de negócios e requisitos até o *deploy* e manutenção do *software*.

Porém, analisando a Figura 2, que traz o ciclo de vida do desenvolvimento de software, percebe-se que no mercado são os profissionais de produto, normalmente PO ou PM, que assumem a responsabilidade por esses alinhamentos com clientes. Portanto, o momento de concepção e desenvolvimento não são diretamente dependentes dos clientes, que fazem parte desse escopo distinto previamente definido.

Devido a esses comentários, foi perceptível que deveríamos ter delineado de maneira mais clara as limitações do escopo desse estudo na apresentação para as empresas, uma vez que a ênfase dele está em aprimorar a comunicação e colaboração entre *designers* e desenvolvedores, por isso na Figura 7 é possível observar que as três cerimônias propostas estão cercando os momentos de *Design* e *Development* do ciclo atual.

### 5.3.3 Pontos de melhoria

O primeiro ponto de melhoria que merece menção se relaciona à presença da palavra "Design" nas nomenclaturas das três etapas: *Design Alignment*, *Design Review* e *Design Validation*. Ela pode erroneamente sugerir uma carga adicional de trabalho para os *designers* ao longo do ciclo de desenvolvimento, quando na verdade, todas as fases são interdependentes entre os profissionais envolvidos e visam evitar retrabalhos de ambas as partes.

Além disso, foi possível perceber que a semelhança entre os nomes pode causar confusão, levando-nos a refletir que trabalhar melhor o *Writing* dessas cerimônias ajudaria a trazer maior reconhecimento ao invés de lembrança, para o significado de cada etapa, tal como uma das Heurísticas de Nielsen (NIELSEN, 2024). No entanto, é importante relembrar que o foco principal desse estudo é levantar os desafios, as práticas de solução e os momentos adequados para aplicá-las, portanto a melhoria dos nomes poderá ser feita à medida que *feedbacks* forem recebidos.

Ainda com relação ao ponto de coleta de *feedbacks*, seria muito benéfico realizar uma avaliação das cerimônias propostas em um ambiente de mercado durante um período significativamente longo, a fim de observar as mudanças ocorridas no relacionamento entre *designers* e desenvolvedores, nos entregáveis de cada *sprint* e na qualidade do produto final. No entanto, tal teste demandaria vários meses de implementação, o que seria inviável nesse estudo. Mas, essa aplicação prática representa um objetivo e uma necessidade para pesquisas futuras.

## 6 CONCLUSÃO

Por fim, as contribuições e um resumo desse estudo serão pontuados. Além disso, as limitações e também algumas perspectivas de trabalhos futuros serão mapeadas.

### 6.1 RESUMO

Esse estudo interpreta o momento de *handoff* como mais que uma simples passagem de bastão e transferência de responsabilidades e o enxerga como um processo contínuo de colaboração. O objetivo era não apenas analisar os problemas identificados na literatura, mas também explorar novos desafios e possíveis causas que permeiam esse momento de transferência. Ao longo dessa pesquisa, foram identificados problemas de comunicação e falta de alinhamento não só durante o momento de *handoff*, mas durante todo o ciclo de desenvolvimento de *software*. Foram observadas dificuldades no relacionamento entre os *designers* e desenvolvedores envolvidos no projeto que compromete o seu sucesso.

Com isso, tornou-se evidente que, mais do que uma solução genérica de como fazer o *handoff* perfeito, o foco deveria estar em como criar uma relação sólida entre *designers* e desenvolvedores, para a partir disso, chegar ao *handoff* ideal no contexto do time. Com esse objetivo, foi proposta a implementação de três cerimônias no ciclo de vida do desenvolvimento de *software*: *Design Alignment*, *Design Review* e *Design Validation*. O propósito é que elas sejam momentos para uma interação contínua durante as *sprints*, voltadas para a resolução de problemas de alinhamento de expectativas, viabilidade técnica, mapeamento de fluxos alternativos e para enfrentar o desafio da validação final da experiência do usuário do que foi desenvolvido em comparação com o que foi inicialmente prototipado.

### 6.2 LIMITAÇÕES

Ao nos concentrarmos na comunicação e colaboração entre *design* e desenvolvimento, é importante reconhecer que a solução proposta não aborda outros desafios que podem surgir durante o processo, como a documentação efetiva de arquivos de *design* ou a implementação de um *Design System* e outros processos estruturados para facilitar a transição do *design* para o código.

Além disso, é crucial considerar que o contexto dos times ágeis varia consideravelmente, dependendo da empresa, do ambiente, das pessoas envolvidas e das dinâmicas internas. Essa diversidade apresenta desafios adicionais para a implementação das sugestões em todos os contextos. O propósito não era propor uma solução universal e perfeita, mas sim fornecer orientações para equipes que enfrentam dificuldades na entrega de valor devido

à comunicação entre *designers* e desenvolvedores. Por meio dessas sugestões é que essas equipes conseguiriam chegar a uma solução ideal no seu contexto.

Embora a implementação completa das práticas de *Design Alignment*, *Design Review* e *Design Validation* possa não ser viável para todas as equipes, algumas já incorporam naturalmente partes dessa solução em sua rotina diária, promovendo um ambiente de trabalho mais colaborativo e harmonioso. No entanto, para um panorama geral, é primordial que a implementação prática do ciclo estendido seja feita ao longo de vários meses em um ambiente empresarial, conforme observado durante os testes. Por meio dessa aplicação é que será possível obter informações significativas sobre o impacto no ambiente de trabalho e na qualidade do produto final.

Por fim, convém mencionar que a escassez de tempo no desenvolvimento de *software* é uma questão que a metodologia ágil busca resolver, priorizando entregas frequentes e enfatizando a interação entre as pessoas em detrimento da documentação excessiva. Por conta disso, a proposta desse estudo pode ser inicialmente criticada por parecer demandar mais tempo de trabalho das equipes. No entanto, é importante ressaltar que ela serve como um ponto de partida para promover uma relação mais estreita entre *designers* e desenvolvedores, permitindo que eles descubram o que funciona melhor para a equipe e para o ambiente de trabalho. Mesmo que a equipe não adote formalmente as práticas sugeridas, se o contato entre esses profissionais ocorrer de maneira mais frequente e eficaz e o trabalho se tornar mais colaborativo, o objetivo da proposta desse estudo foi alcançado.

### 6.3 TRABALHOS FUTUROS

Concluimos este estudo com novas perspectivas sobre a colaboração entre *design* e desenvolvimento, reconhecendo que pesquisas posteriores devem aprimorar as descobertas, afinal, o processo de desenvolvimento e a área da tecnologia evoluem a cada dia e o que funciona hoje para um time, pode já não funcionar daqui a alguns meses.

Dentre as possibilidades de aprofundamento futuro do tema, destacam-se:

- **Análises de longo prazo da implementação das práticas propostas:** Conduzir estudos para avaliar a eficácia e a sustentabilidade da implementação das práticas sugeridas ao longo do tempo seria essencial. Isso ajudaria a entender melhor como essas práticas impactam a colaboração entre *designers* e desenvolvedores em diferentes contextos e ao longo do ciclo de vida de diferentes tipos de projetos de *software*.
- **Exploração de outras metodologias:** Além do Scrum, seria interessante explorar como as práticas propostas podem ser adaptadas e implementadas com outras metodologias, como o Kanban ou Lean Software Development. Cada metodologia tem suas próprias características e desafios, e entender como as práticas de cola-

colaboração e comunicação entre *design* e desenvolvimento podem ser integradas em diferentes contextos ampliaria o impacto e a aplicabilidade das propostas.

- **Avaliação do impacto na experiência do usuário final:** Uma outra vertente a ser investigada seria a relação da melhoria da colaboração entre *designers* e desenvolvedores com a experiência do usuário final. Isso poderia envolver estudos de usabilidade, *feedback* dos usuários e análises de métricas de desempenho para entender melhor como as práticas propostas influenciam a qualidade e a aceitação dos produtos.

Em resumo, as perspectivas futuras dessa pesquisa oferecem uma oportunidade ampla para continuar aprimorando a colaboração entre *design* e desenvolvimento e, consequentemente, melhorar a qualidade dos projetos de *software*.

## REFERÊNCIAS

- AWS, H. **What is SDLC (Software Development Lifecycle)?** 2021. <https://aws.amazon.com/what-is/sdlc>. Artigo online.
- BABICH, N.; WHITE, A. Building better ui designs with layout grids. **Smashing Magazine**, v. 20, 2017.
- BLOG, F. **Everything you need to know about Dev Mode**. 2024. <https://www.figma.com/blog/everything-you-need-to-know-about-dev-mode/>. [Online; accessed 02-march-2024].
- DESMEDT, M. et al. Clinical handover and handoff in healthcare: a systematic review of systematic reviews. **International Journal for Quality in Health Care**, Oxford University Press UK, v. 33, n. 1, p. mzaa170, 2021.
- DESMET, P.; HEKKERT, P. Framework of product experience. **International journal of design**, v. 1, n. 1, p. 57–66, 2007.
- FENG, K. J. K.; ZHANG, A. X. From Handoffs to Co-Creation: Deepening Collaboration between Designers, Developers, and Data Science Workers in UX Design. **CHI**, p. 7, 2022.
- FENG, T. W. L. K. J. K.; ZHANG, A. X. Understanding Collaborative Practices and Tools of Professional UX Practitioners in Software Organizations. **CHI**, p. 20, 2023.
- FERATI, N. Automating design handoff between UX/UI design and development team. p. 80, 2022.
- FIGMA. 2024. <https://www.figma.com/ui-design-tool/>. [Online; accessed 02-march-2024].
- GREEN, T.; LABRECQUE, J. Developer handoff. In: **A Guide to UX Design and Development: Developer’s Journey Through the UX Process**. [S.l.]: Springer, 2023. p. 203–224.
- HEMA, V. et al. Scrum: An effective software development agile tool. In: IOP PUBLISHING. **IOP Conference Series: Materials Science and Engineering**. [S.l.], 2020. v. 981, n. 2, p. 022060.
- HRON, M.; OBWEGESER, N. Scrum in practice: an overview of scrum adaptations. 2018.
- IMRAN, M. A design handoff guide for software development. Laurea-ammattikorkeakoulu, 2018.
- JACOB, R. J. User interface. In: **Encyclopedia of Computer Science**. [S.l.: s.n.], 2003. p. 1821–1826.
- JR, H. G. H. et al. Handoff practices in emergency medicine: are we making progress? **Academic Emergency Medicine**, Wiley Online Library, v. 23, n. 2, p. 197–201, 2016.

- LEAU, Y. B. et al. Software development life cycle agile vs traditional approaches. In: **International Conference on Information and Network Technology**. [S.l.: s.n.], 2012. v. 37, n. 1, p. 162–167.
- LEINONEN, A.; ROTO, V. Service design handover to user experience design—a systematic literature review. **Information and Software Technology**, Elsevier, v. 154, p. 107087, 2023.
- LIN, J.; KNEAREM, T.; GIESING, K. Relay: A collaborative ui model for design handoff. In: **Adjunct Proceedings of the 36th Annual ACM Symposium on User Interface Software and Technology**. [S.l.: s.n.], 2023. p. 1–3.
- MAUDET, N. et al. Design breakdowns: designer-developer gaps in representing and interpreting interactive systems. In: **Proceedings of the 2017 ACM Conference on Computer Supported Cooperative Work and Social Computing**. [S.l.: s.n.], 2017. p. 630–641.
- NIELSEN, J. **10 Usability Heuristics for User Interface Design**. 2024. <https://www.nngroup.com/articles/ten-usability-heuristics/>. [Online; accessed 15-march-2024].
- NORMAN, D. **Design do Dia a Dia**. [S.l.]: Anfiteatro, 2018. 328 p.
- NOTION, S. 2024. <https://www.notion.so>. [Online; accessed 30-march-2024].
- THORNBERG, R.; PERHAMUS, L.; CHARMAZ, K. Grounded theory. **Handbook of research methods in early childhood education: Research methodologies**, Information Age Publishing, Inc, v. 1, p. 405–439, 2014.
- VILLAIN, M.; SILVEIRA, M. I. **Figma: o que é a ferramenta, Design e uso**. 2023. <https://www.alura.com.br/artigos/figma>. Artigo online.
- ZAYAT, W.; SENVAR, O. Framework study for agile software development via scrum and kanban. **International journal of innovation and technology management**, World Scientific, v. 17, n. 04, p. 2030002, 2020.

## GLOSSÁRIO

**Android** Sistema operacional baseado no núcleo Linux, projetado principalmente para dispositivos eletrônicos móveis.

**Design System** Ecossistema de bibliotecas instaláveis, com componentes programados e padrões semânticos de design, que reúne padrões de comportamentos.

**Design Tokens** Módulos de informação reutilizáveis que compõem a linguagem visual de um sistema.

**Figma** Plataforma colaborativa para construção de design de interfaces e protótipos.

**Google Forms** Aplicativo de gerenciamento de pesquisas.

**Google Sheets** Aplicativo de planilhas.

**Kanban** Metodologia ágil que permite fazer o acompanhamento de tarefas e identificar gargalos no fluxo de trabalho.

**Lean Software Development** Metodologia ágil baseada na otimização do tempo e dos recursos de desenvolvimento, eliminando desperdícios e, em última análise, entregando apenas o que o produto precisa.

**Microsoft Teams** Plataforma unificada de comunicação e colaboração que combina bate-papo, videoconferências, armazenamento de arquivos e integração de aplicativos no local de trabalho.

**Notion** Aplicativo de produtividade focado na organização de tarefas e no trabalho colaborativo em projetos.

**Sitemap** Arquivo responsável por esclarecer aos mecanismos de busca a estrutura e o conteúdo disponibilizado por um site.

**Slack** Plataforma de comunicação de equipe baseada em nuvem.

**Visual Studio Code** Editor de código redefinido e otimizado para criar e depurar aplicativos modernos da Web e da nuvem.

## ANEXO A – ROTEIRO QUANTITATIVA

Importante destacar que essas perguntas eram de múltipla escolha, não eram de resposta livre.

### A.1 GERAIS

1. Há quanto tempo você atua na sua profissão?
2. Qual seu nível de senioridade?
3. No seu time há desenvolvedores e *designers* ou são times separados?
4. Qual a sua função principal no processo de desenvolvimento de *software*?

### A.2 DESIGNER

1. Com que frequência você tem contato com os desenvolvedores dos projetos em que trabalha?

Caso tenha contato, como ele costuma ser?

Em quais momentos geralmente ocorrem interações com os desenvolvedores?

2. Com que frequência você está envolvido no processo de *handoff*?
3. Com qual frequência você valida se o que foi desenvolvido está de acordo com o protótipo? (Interações e *design* de interface coerentes)
4. Como você costuma entregar o projeto para os desenvolvedores?
5. Quais ferramentas você costuma utilizar durante o *handoff*?
6. Para você, quais são os principais desafios encontrados durante o *handoff*?
7. Em uma escala de importância, o quanto você considera que o *handoff* / documentação do *design* é valorizado/incentivado por sua empresa?

Você faz alguma checklist ou segue algum padrão que seja adaptável e que facilite o momento do *handoff* em diferentes projetos? Nos conte um pouco.

### A.3 DESENVOLVEDOR

1. Com que frequência você tem contato com os *designers* dos projetos em que trabalha?

Caso tenha contato, como ele costuma ser?

Em quais momentos geralmente ocorrem interações com os *designers*?

2. Como você costuma receber os protótipos do time de *design*?
3. O que você considera que mais facilita seu trabalho em uma entrega de protótipo para desenvolvimento?
4. Para você, quais são os principais desafios encontrados durante o *handoff*?

#### A.4 DEMAIS PROFISSIONAIS

1. Você costuma acompanhar o momento da transição de trabalho do time de *design* para o time de desenvolvimento?
2. Você percebe algum problema no fluxo de transição de trabalho entre essas duas equipes?

## ANEXO B – ROTEIRO QUALITATIVA

### B.1 GERAIS

1. Conte um pouco sobre seu trabalho, em qual empresa e quanto tempo trabalha nela?
2. Como é seu dia a dia de trabalho? O que você faz normalmente e quais são suas demandas?

### B.2 DESIGNER

1. Como costuma ser seu processo de trabalho?
2. Quem são as pessoas envolvidas nesse processo?
3. Quando você costuma apresentar seu progresso para elas?
4. Normalmente suas demandas englobam os caminhos alternativos como mensagens de erro, alertas ou mais o caminho feliz?
5. Quando você finaliza o protótipo e o entrega aos desenvolvedores, você pratica alguma ação relacionada ao *handoff*? Como ela acontece normalmente?
6. Pode nos contar sobre alguma dificuldade que você geralmente enfrenta durante o processo de *handoff*?
7. Durante o desenvolvimento do código, você costuma acompanhar o que está sendo desenvolvido? Os desenvolvedores te mostram durante o processo?
8. Ao final do desenvolvimento, você normalmente valida o que foi desenvolvido e verifica se está de acordo com o protótipo? Como isso acontece?
9. Você costuma encontrar problemas de interface quando revisa o que foi desenvolvido em comparação com o que foi prototipado?

Quais problemas você normalmente encontra?

### B.3 DESENVOLVEDOR

1. Como funciona a interação com o time de *design*?  
Essa interação é somente ao receber o protótipo final?
2. No seu time, acontecem reuniões para verificar viabilidades técnicas?

3. Quando você recebe o protótipo da equipe de *design*, as especificações visuais (ex: espaçamento, cor, fontes, tamanho de margem) dos projetos geralmente estão claras e fáceis de serem encontradas?

Já teve que solicitar isso diretamente ao *designer* em algum momento?

Você se sente confortável para encontrar essas especificações no Figma?

4. Você normalmente recebe protótipos interativos ou somente o *design* de interface?
5. Quando você recebe somente o *design* de interface (sem interação), como você faz a conexão das telas e a criação da interação no código?
6. Durante o desenvolvimento, você normalmente dá falta de algum fluxo alternativo?
7. Você costuma realizar testes de interface?
8. Você costuma validar o que está desenvolvendo com o *designer* durante o desenvolvimento?
9. Ao final do desenvolvimento, normalmente o *designer* valida o que você desenvolveu?

Existe algum rito específico para isso? É uma etapa obrigatória?

**ANEXO C – ROTEIRO TESTE**

1. Quais pontos da solução proposta você considera mais eficazes?
2. Quais problemas você enxerga no ciclo proposto?
3. Você tem alguma sugestão adicional ou comentário sobre a solução que gostaria de compartilhar?