

**UNIVERSIDADE FEDERAL DE ALAGOAS - UFAL
INSTITUTO DE COMPUTAÇÃO - IC
PROGRAMA DE PÓS-GRADUAÇÃO EM INFORMÁTICA - PPGI**

ALENILTON SANTOS SILVA

**VOLATILIDADE DE REQUISITOS EM AMBIENTES DINÂMICOS E COM
MÚLTIPLOS STAKEHOLDERS: ESTRATÉGIAS E BOAS PRÁTICAS PARA APOIAR
O PROCESSO ÁGIL DE DESENVOLVIMENTO DE APLICAÇÕES DE SOFTWARE
GOVERNAMENTAIS**

**MACEIÓ
2024**

Alenilton Santos Silva

Volatilidade de Requisitos em Ambientes Dinâmicos e com Múltiplos Stakeholders: Estratégias e Boas Práticas para Apoiar o Processo Ágil de Desenvolvimento de Aplicações de Software Governamentais

Dissertação apresentada como requisito parcial para obtenção do grau de Mestre em Informática no Programa de Pós-Graduação em Informática - PPGI da Universidade Federal de Alagoas - UFAL, Instituto de Computação - IC.

Orientador: Prof. Dr. André Magno C. de Araújo

Coorientador: Prof. Dr. Rafael Dias Araújo

Maceió

2024

Catálogo na fonte
Universidade Federal de Alagoas
Biblioteca Central
Divisão de Tratamento Técnico
Bibliotecária Myrtes Vieira do Nascimento CRB4/1680

S586v Silva, Alenilton Santos
 Volatilidade de requisitos em ambientes dinâmicos e com múltiplos stakeholders: estratégias e boas práticas para apoiar o processo ágil de desenvolvimento de aplicação de software governamentais. / Alenilton Santos Silva - 2024.
 123 f.; il.

Dissertação de Pós-graduação em Informática – Universidade Federal de Alagoas, Campus Ac. Simões. Maceió, 2024.

Orientação: Dr. André Magno C. de Araújo
Coorientação: Dr. Rafael Dias Araújo

Inclui bibliografia e anexo

1. Engenharia de software. 2. Gestão de stakeholders.
3. Metodologias ágeis. I. Título.

CDU: 004.41



MINISTÉRIO DA EDUCAÇÃO
UNIVERSIDADE FEDERAL DE ALAGOAS
INSTITUTO DE COMPUTAÇÃO
Av. Lourival Melo Mota, S/N, Tabuleiro do Martins, Maceió - AL, 57.072-970
PRÓ-REITORIA DE PESQUISA E PÓS-GRADUAÇÃO (PROPEP)
PROGRAMA DE PÓS-GRADUAÇÃO EM INFORMÁTICA

Folha de Aprovação

ALENILTON SANTOS SILVA

**VOLATILIDADE DE REQUISITOS EM AMBIENTES DINÂMICOS E COM
MÚLTIPLOS STAKEHOLDERS: ESTRATÉGIAS E BOAS PRÁTICAS PARA
APOIAR O PROCESSO ÁGIL DE DESENVOLVIMENTO DE APLICAÇÕES DE
SOFTWARE GOVERNAMENTAIS**

**REQUIREMENTS VOLATILITY IN DYNAMIC AND MULTI-STAKEHOLDER
ENVIRONMENTS: STRATEGIES AND BEST PRACTICES TO SUPPORT THE
AGILE DEVELOPMENT PROCESS OF GOVERNMENT SOFTWARE
APPLICATIONS**

Dissertação submetida ao corpo docente do
Programa de Pós-Graduação em Informática da
Universidade Federal de Alagoas e aprovada em
18 de dezembro de 2024.

Banca Examinadora:



Documento assinado digitalmente
ANDRÉ MAGNO COSTA DE ARAÚJO
Data: 18/12/2024 17:23:08-0300
Verifique em <https://validar.iti.gov.br>

Prof. Dr. ANDRÉ MAGNO COSTA DE ARAÚJO
UFAL – PPGI- Instituto de Computação
Orientador



Documento assinado digitalmente
PAULA TOLEDO PALOMINO
Data: 18/12/2024 17:27:53-0300
Verifique em <https://validar.iti.gov.br>

Prof. Dr. PAULA TOLEDO PALOMINO
Faculdade de Tecnologia de Matão - Luiz Marchesan
Examinadora Externa



Documento assinado digitalmente
DIEGO DERMEVAL MEDEIROS DA CUNHA MATOS
Data: 18/12/2024 18:21:17-0300
Verifique em <https://validar.iti.gov.br>

**Prof. Dr. DIEGO DERMEVAL MEDEIROS DA
CUNHA MATOS**
UFAL – FAMED
Examinador Interno



Documento assinado digitalmente
RAFAEL DIAS ARAÚJO
Data: 18/12/2024 18:36:55-0300
Verifique em <https://validar.iti.gov.br>

Prof. Dr. RAFAEL DIAS ARAÚJO
UFU – Universidade Federal de Uberlândia
Examinador Externo

*para todas as vozes em minha cabeça - dotadas
de corpo ou não.*

AGRADECIMENTOS

Antes de tudo, agradeço a cada corpo preto que se foi, mas deixou lampejos e sementes que garantiram o meu lugar nos espaços que ocupo e almejo. Em mesma medida, a cada corpo LGBTQIA+ que transformou dor em luta e construiu espaços possíveis para mim e para os nossos.

Eu acredito piamente, sem tirar nem pôr, que a vida é uma rede tecida a mão ou, ainda, uma parede de tijolos. Que nas duas possibilidades a base tem poder indescritível para dizer um terço do fim. No meu caso, Mãe e Pai, Elenita e Ailton, obrigado por tecerem ao modo que puderam, com as ferramentas que tinham, driblando a escassez e sobressaindo o medo do amanhã. Por deixarem viva a teimosia que me habita.

Aos que junto comigo se apropriaram das redes — irmãos e irmãs —, agradeço o apoio e carinho de sempre, e sim, talvez eu ainda tenha de enlouquecer de estudar e vocês verão. Essencialmente, à Lica e à Carla, as quem me estendo com orgulho e troca, que são presenças mesmo à distância e que vibram diretamente a cada passo dado — mesmo os miúdos.

Cativo lembranças estranhas dos dias primeiros, do início, quando meu nome saiu na lista de aprovados e me fez mudar de Penedo para capital. Lembrar dessa fase é motivo fácil para lembrar de Afonso Xavier e das noites insones e dias vários, de tantas angústias, mas de tanto olho atento. Do cuidado. Agradeço a paciência e o afeto, as conversas soltas e a ajuda na divisão dos pesos. Agradeço também ao Vitor Abreu pela troca e pelas doses de incentivo, por dividir angústias e me fazer perceber dos causos comuns. Me estendo um pouco mais e chego a Nathan Tavares e seu dom de me fazer rir das mesmas angústias que dividi com Vitor e Afonso. Você é desmedidamente importante para mim.

Ao meu orientador, André Magno, obrigado pelos quase dois anos de muito aprendizado. É muito importante conhecer pessoas que nos façam lembrar de nossas capacidades, pois, quando eu ainda era graduando tinha distanciado a Academia dos planos e você me acendeu essa vontade novamente. Agradeço também ao meu co-orientador, Rafael Dias, pelas contribuições e disposição, pelo crescimento proporcionado. Por aqui, ainda agradeço a Luciane e Emerson, por colaborarem, direta e indiretamente, com a condução deste estudo enquanto integrantes do mesmo grupo de pesquisa que eu.

Por último, mas também o primeiro, agradeço a você que escreveu — em terceira pessoa mesmo —, pelas pausas, pelo olhar para dentro, por ouvir as benditas vozes da cabeça, por não esquecer que és gente. A você que sou eu.

RESUMO

A crescente exigência por qualidade nos diversos aspectos de produtos de software ascende a complexidade nos processos que circundam o ciclo de vida do software. Algumas características agregam ainda mais complexidade ao processo de desenvolvimento, como a dinamicidade do ambiente e o alto número de stakeholders, que tornam os requisitos funcionais altamente voláteis. Nesse contexto, este estudo busca investigar, no estado da arte e da prática, como a qualidade de uma solução computacional e do seu processo de desenvolvimento, são impactados pela volatilidade de requisitos em ambientes dinâmicos e com múltiplos stakeholders. Para isso, foi executado um levantamento no estado da arte para compreender como a temática tem sido abordada e em que nível as investigações estão e, logo em seguida, foi considerada uma investigação prática de um ambiente real caracterizado como dinâmico e com múltiplos stakeholders. O objeto de estudo foi o PNLD Avaliação, software desenvolvido para apoiar a Avaliação Pedagógica do Programa Nacional do Livro e Material Didático. No primeiro estágio foi executada uma inspeção histórica e documental dos artefatos de software gerados no ciclo de desenvolvimento, que permitiu entender as nuances do objeto de pesquisa, bem como identificar as partes interessadas e seus papéis, e os impactos nas alterações periódicas no escopo do projeto. Com a aplicação do teste estatístico Qui-Quadrado confirmou-se que as diferenças nos números de histórias de usuário alteradas, descontinuadas e finalizadas entre os dois períodos estudados, foram estatisticamente significativas e provavelmente não são devidas à variação aleatória. Em seguida, no segundo estágio, por meio de entrevistas semiestruturadas com diferentes stakeholders do projeto, foram revelados alguns desafios amplos na gestão de ambientes dinâmicos e com múltiplos stakeholders, pois, embora se utilize de metodologias ágeis, sua eficácia foi limitada sem uma documentação clara, definição falha de processos, sobrecarga das equipes, bem como a falta de um mediador formal que dificultou o alinhamento e a resolução de problemas críticos. Por fim, no terceiro estágio, foi aplicado o método SUS integrado a um questionário aberto para capturar as percepções dos usuários finais sobre o sistema estudado em relação à usabilidade. Embora o resultado geral do SUS tenha apontado para uma pontuação positiva (73.2, cinco pontos acima do mínimo), cerca de 40% dos usuários apontaram insatisfações gerais com o sistema. Para maior aprofundamento, as respostas abertas do questionário foram submetidas a uma Análise Temática Dedutiva, que apontou problemas estruturais, como integração inadequada e redundâncias que prejudicaram a produtividade e a experiência do usuário; a falta de iteração e feedback no desenvolvimento resultou em uma interface complexa, exigindo re-aprendizagem constante e frustrando os usuários. De modo geral, ao longo dessa condução, foram destacados como os conflitos de interesses, a comunicação falha entre stakeholders, e a não centralização dos processos nas partes interessadas, adicionam complexidade no ciclo de vida de software que, por consequência, é fragilizado ainda mais frente a dinamicidade do ambiente. A partir disso, fundamenta-se a proposição de um conjunto de boas práticas para apoiar a gestão de projetos de desenvolvimento de software que operam sob a complexidade de ambientes dinâmicos e com múltiplos stakeholders. Além disso, foi definido um processo de desenvolvimento ágil, levando em conta as nuances das melhores práticas estabelecidas, para promover melhorias e robustez nos fluxos de trabalho em ambientes semelhantes aos já caracterizados.

Palavras-chave: Engenharia de Software, Metodologias Ágeis, Ambientes Dinâmicos, Volatilidade de Requisitos, Gestão de Stakeholders.

ABSTRACT

The increasing demand for quality across various aspects of software products heightens the complexity of processes surrounding the software lifecycle. Certain characteristics further amplify this complexity, such as the dynamic nature of the environment and the high number of stakeholders, which make functional requirements highly volatile. In this context, this study aims to investigate, both in the state of the art and practice, how the quality of a computational solution and its development process are impacted by requirement volatility in dynamic environments with multiple stakeholders. To achieve this, a state-of-the-art review was conducted to understand how the topic has been approached and the current level of investigations. Following this, a practical investigation was carried out in a real environment characterized as dynamic and involving multiple stakeholders. The case study focused on PNLD Avaliação, a software developed to support the Pedagogical Evaluation of the National Textbook and Teaching Material Program. In the first stage, a historical and documentary inspection of the software artifacts generated during the development lifecycle was conducted, allowing for a better understanding of the research object's nuances, the identification of stakeholders and their roles, and the impact of periodic scope changes. Applying the Chi-Square statistical test confirmed that differences in the number of user stories altered, discontinued, and completed across the two studied periods were statistically significant and likely not due to random variation. Subsequently, in the second stage, semi-structured interviews with various project stakeholders revealed several broad challenges in managing dynamic environments and multiple stakeholders. Despite the use of agile methodologies, their effectiveness was limited by unclear documentation, flawed process definitions, team overload, and the absence of a formal mediator, which hindered alignment and the resolution of critical issues. Finally, in the third stage, the SUS method was applied alongside an open questionnaire to capture end-user perceptions of the studied system's usability. Although the overall SUS score indicated a positive rating (73.2, five points above the minimum threshold), approximately 40% of users expressed general dissatisfaction with the system. For deeper insights, the open-ended questionnaire responses underwent Deductive Thematic Analysis, highlighting structural issues such as inadequate integration and redundancies that hampered productivity and user experience. The lack of iteration and feedback during development resulted in a complex interface requiring constant re-learning, frustrating users. Overall, this study emphasized how conflicting interests, poor communication among stakeholders, and the absence of process centralization around stakeholders add complexity to the software lifecycle, further weakening it in dynamic environments. Based on these findings, a set of best practices was proposed to support the management of software development projects operating in complex, dynamic environments with multiple stakeholders. Additionally, an agile development process was outlined, considering the nuances of established best practices to enhance workflows' robustness and efficiency in similar environments.

Keywords: Software Engineering, Agile Methodologies, Dynamic Environments, Requirement Volatility, Stakeholder Management.

LISTA DE FIGURAS

Figura 1 – Percurso metodológico	20
Figura 2 – Integração de IHC com Engenharia de Software	24
Figura 3 – Desafios e Lacunas Encontrados	33
Figura 4 – Estudos selecionados e abrangências temáticas	40
Figura 5 – Síntese dos estudos selecionados e tópicos comuns.	41
Figura 6 – Desafios e Lacunas Encontrados	42
Figura 7 – Etapas do PNLD	48
Figura 8 – Etapas do PNLD	50
Figura 9 – Fluxograma do PNLD Avaliação sob ótica do processo de avaliação pedagógica	53
Figura 10 – Mapa de stakeholders e correlações	56
Figura 11 – Representação do ciclo de vida do PNLD-Avaliação.	57
Figura 12 – Redesenho do fluxo de colaboração do PNLD Avaliação.	58
Figura 13 – Relação entre problemas abertos e fechados de Maio de 2023 a Maio de 2024 no GitLab (<i>front-end</i>).	64
Figura 14 – Relação entre problemas abertos e fechados de maio de 2023 a maio de 2024 no GitLab (<i>back-end</i>).	65
Figura 15 – Métodos adotados para condução do Estágio 02.	67
Figura 16 – Métodos adotados para condução do Estágio 03.	73
Figura 17 – Demonstração visual da Escala de Usabilidade de Sistema.	74
Figura 18 – População e amostra da pesquisa	77
Figura 19 – Distribuição por região, faixa etária e sexo da amostra	77
Figura 20 – Distribuição por título dos participantes	78
Figura 21 – Distribuição por título dos participantes	79
Figura 22 – Escala de Usabilidade do PNLD Avaliação frente ao SUS	80
Figura 23 – Pontuação individual das questões SUS.	82
Figura 24 – Correlação entre as questões do SUS com os componentes de qualidade. . .	83
Figura 25 – Avaliação por componente de qualidade.	83
Figura 26 – Facilidade de aprendizagem a partir das questões do SUS.	84
Figura 27 – Eficiência a partir das questões do SUS.	85
Figura 28 – Inconsistências e Facilidade de memorização.	86
Figura 29 – Satisfação a partir das questões do SUS.	87
Figura 30 – Recomendações práticas para lidar com a volatilidade de requisitos.	99
Figura 31 – Boas Práticas aplicadas no Processo de Desenvolvimento de Software . . .	105

LISTA DE TABELAS

Tabela 1	–	Frequência dos valores observados e esperados para os dois grupos	63
Tabela 2	–	Resultado estatístico dos grupos investigados	63
Tabela 3	–	Número de problemas no <i>front-end</i> no GitLab de maio de 2023 a maio de 2024.	64
Tabela 4	–	Número de problemas no <i>back-end</i> no GitLab de maio de 2023 a maio de 2024.	65
Tabela 5	–	Número de Chamados no GLPI por status.	65
Tabela 6	–	Dados Demográficos dos Participantes das Entrevistas	68
Tabela 7	–	Questões que constituem o SUS	74
Tabela 8	–	Pontuações médias consideradas em relação ao tipo de interface	75

LISTA DE ABREVIATURAS E SIGLAS

CAAE	Certificado de Apresentação de Apreciação Ética
CGMD	Coordenação Geral de Materiais Didáticos
ER	Engenharia de Requisitos
ES	Engenharia de Software
FNDE	Fundo Nacional de Desenvolvimento da Educação
HU	História de Usuário
IHC	Interação Humano-Computador
MEC	Ministério da Educação
PNLD	Programa Nacional do Livro Material Didático
RED's	Recursos Educacionais Digitais
SUS	System Usability Scale
SWEBOK	Software Engineering Body of Knowledge
TD	Transformação Digital
VR	Volatilidade de Requisitos

SUMÁRIO

1	INTRODUÇÃO	14
1.1	CONTEXTUALIZAÇÃO	14
1.2	MOTIVAÇÃO	16
1.3	OBJETIVOS	18
1.3.1	Objetivo geral	18
1.3.2	Objetivos específicos	18
1.4	QUESTÃO DE PESQUISA	19
1.5	METODOLOGIA	19
1.6	ORGANIZAÇÃO DO TRABALHO	21
2	REFERENCIAL TEÓRICO	23
2.1	IHC e Engenharia de Software no Ciclo de Vida do Software	23
2.2	Complexidades no desenvolvimento do software	25
2.2.1	Ambientes Dinâmicos e com Múltiplos Stakeholders	26
2.2.2	Volatilidade de Requisitos no Desenvolvimento de Software	28
2.3	Metodologias Ágeis e Respostas a Ambientes Dinâmicos	29
3	REVISÃO DA LITERATURA	32
3.1	PERCURSO METODOLÓGICO PARA SELEÇÃO DE ESTUDOS	32
3.2	AVALIAÇÃO DOS ESTUDOS ADERENTES	34
3.2.1	Análise do Estado da Arte	34
3.2.2	Resultados e Discussões	39
3.2.3	Desafios e Lacunas Identificados	42
3.3	CONSIDERAÇÕES FINAIS DO CAPÍTULO	44
4	INVESTIGANDO UM SOFTWARE GOVERNAMENTAL DESENVOLVIDO EM AMBIENTE ALTAMENTE DINÂMICO E COM MÚLTIPLOS STAKEHOLDERS	45
4.1	ASPECTOS ÉTICOS	45
4.2	PERCURSO METODOLÓGICO	46
4.3	ESTÁGIO 01: IDENTIFICAÇÃO DE CENÁRIO E INSPEÇÃO DE ARTIFATOS DE SOFTWARE	47
4.3.1	Instrumento de Coleta e Análise	47
4.3.2	O Programa Nacional do Livro e do Material Didático	49
4.3.2.1	PNLD Avaliação	51

4.3.2.2	Panorama da Transformação Digital do PNLD	54
4.3.2.3	Redefinição do fluxo de desenvolvimento da solução computacional de apoio a avaliação pedagógica do PNLD	58
4.3.3	Inspeção de Artefatos do Software do PNLD Avaliação	60
4.3.3.1	Análise das Histórias de Usuário	61
4.3.3.2	Análise de Chamados de Suporte	63
4.3.3.3	Resultados e Discussões	66
4.4	ESTÁGIO 02: CONDUÇÃO DE ENTREVISTAS COM MÚLTIPLOS STA- KEHOLDERS	67
4.4.1	Instrumento de Coleta e Análise e Dados Demográficos	67
4.4.2	Análise Temática de Abordagem Mista	68
4.4.3	Resultados e Discussões	71
4.5	ESTÁGIO 03: ANÁLISE DE USABILIDADE E CAPTURA DE PERCEP- ÇÕES DOS USUÁRIOS FINAIS	72
4.5.1	Instrumentos de Coleta e Análise	72
4.5.2	Descrição Demográfica da População e Amostra	75
4.5.3	Resultados Quantitativos da Escala SUS	80
4.5.4	Análise Temática Dedutiva das questões abertas do formulário	87
4.5.5	Resultados e Discussões	92
4.6	Consolidação dos Resultados	95
5	BOAS PRÁTICAS PARA O DESENVOLVIMENTO DE SOFTWARES QUE OPERAM EM AMBIENTES DINÂMICOS E COM MÚLTIPLOS STAKEHOLDERS	98
5.1	Boas Práticas para Lidar com a Volatilidade de Requisitos em Ambientes Dinâmicos e com Múltiplos Stakeholders	98
5.2	Processo de Desenvolvimento de Software	104
6	CONCLUSÃO	107
6.1	PRINCIPAIS CONTRIBUIÇÕES	107
6.2	LIMITAÇÕES E AMEAÇAS À VALIDADE	108
6.3	TRABALHOS FUTUROS	110
	REFERÊNCIAS	111
	ANEXO A – COMITÊ DE ÉTICA EM PESQUISA	119

1 INTRODUÇÃO

Este capítulo percorre uma contextualização e discussão acerca da crescente integração dos produtos de software às atividades que regem a sociedade e da importância de se considerar a qualidade no ciclo de vida do software, levando em conta aspectos não funcionais, como usabilidade e satisfação. Para isso, segue a seguinte estrutura: a seção 1.1 insere a discussão da temática proposta nessa dissertação sob uma perspectiva de atualidade, considerando a importância de abordagens que mesclam a Interação Humano-Computador e a Engenharia de Software. Na seção 1.2 apresentam-se as motivações que permeiam as temáticas aqui investigadas. Na seção 1.3 são discutidos os objetivos gerais e específicos. Em seguida, a Seção 1.4 detalha a metodologia utilizada na construção deste trabalho, e por fim, na seção 1.5, descreve-se como estão dispostos os demais capítulos deste documento de dissertação.

1.1 CONTEXTUALIZAÇÃO

Em se tratando de um dispositivo essencial ao funcionamento das sociedades, afirma-se que o mundo moderno não poderia existir sem o software [Sommerville 2011]. Justifica-se, principalmente, por sua aplicação presente em quase todos os processos de manutenção e ressignificação das atividades e comportamentos que permeiam a vida humana, como comunicar, colaborar, etc. [Kazman e Pasquale 2019, Boschma 2005]. Assim, a Transformação Digital (TD) se tornou essencial à sobrevivência para países, organizações e pessoas em todo o mundo e, em se tratando dos governos, tem-se um movimento significativo que compreende a adoção de estratégias aderentes a essa nova realidade [Arabiun et al. 2023].

No setor público, por exemplo, produtos de software têm sido aliados na promoção da transparência e promovem o fortalecimento da prestação de serviços, bem como o alcance e envolvimento do cidadão. Como resultado, a integração de soluções computacionais é essencial para promover o crescimento econômico e melhorar a qualidade de vida [Vial 2021]. No entanto, a crescente integração de sistemas digitais ao cotidiano das pessoas, do pessoal ao coletivo, evidencia que o aumento da exigência por qualidade nos diversos aspectos do produto de software tem ascendido na mesma escala dos termos de sua complexidade. Nesse sentido, entendem-se enquanto aspectos de qualidade do produto de software, desde a adequação funcional até as questões que permeiam o seu uso [Silva et al. 2022].

Considerando que os modelos de negócios atuais são fundamentalmente dinâmicos, caracterizados por requisitos voláteis e pelo envolvimento de várias partes interessadas [Witschel et al. 2019],

é essencial considerar os impactos destes aspectos na qualidade [Neumann et al. 2021]. Estes aspectos adicionam complexidade nos processos de desenvolvimento e representa desafios significativos para o ciclo de vida de aplicações de software, exigindo agilidade e adaptabilidade das equipes de desenvolvimento [Paranjpe e Pachlor 2024]. A frequente mudança nas condições de mercado, a dinâmica massiva dos avanços tecnológicos e as expectativas do cliente, exigem processos iterativos baseados em refinamento contínuos do produto de software [Moyon et al. 2018]. Além disso, a diversidade de interesses das partes interessadas geralmente exigem que times de desenvolvimento equilibrem demandas e prioridades conflitantes, complicando ainda mais o processo de desenvolvimento. Consequentemente, alcançar e manter aplicativos de software que reflitam alta qualidade, que sejam escaláveis e centrados nas partes interessadas, inclusive no usuário final, é um desafio persistente nos ambientes de negócios contemporâneos [Szabó e Hercegi 2023].

Esses desafios são ainda mais evidentes no setor governamental, onde soluções computacionais se apresentam como ferramentas vitais na execução de políticas públicas, atuando na promoção da inclusão social e fornecimento de serviços essenciais a populações carentes de modo escalonado [Filgueiras et al. 2019]. O desenvolvimento de soluções computacionais para uso governamental envolve navegar por requisitos regulatórios rigorosos, garantir a segurança e a privacidade dos dados e atender às diversas necessidades de uma ampla base de usuários [Bose et al. 2020]. Além disso, os projetos governamentais geralmente enfrentam restrições orçamentárias e pressões políticas, o que influi diretamente no escopo do projeto e impede o desenvolvimento [Ambreen 2019]. Apesar desses obstáculos, soluções digitais eficazes são essenciais para melhorar o acesso aos serviços públicos, aumentar a eficiência administrativa e promover oportunidades equitativas para todos os cidadãos [Eja e Ramegowda 2020, Popova et al. 2023].

De modo geral, observa-se um ciclo efeito-causa onde a crescente integração de tecnologias digitais no cotidiano das pessoas e das organizações, gera uma demanda por produtos de alta qualidade, que consequentemente apontam para o aumento da complexidade no ciclo de vida do software [Neumann et al. 2021]. Assim, a adoção de abordagens holísticas são necessárias, contemplando tanto funcionalidades quanto experiência e usabilidade desde as fases iniciais até a implementação [Barbosa et al. 2021].

Nesse cenário, uma questão recorrente é a integração entre os saberes e abrangências da Engenharia de Software (ES) [Pressman 2011, Sommerville 2011] e da Interação Humano-Computador (IHC) [Barbosa et al. 2021, Mayhew 1999, Hix e Rex 1993, Nielsen 1994], nos processos de desenvolvimento. É necessário explorar essa integração como forma de propor

processos que atinjam êxito não só na concepção, desenvolvimento e manutenção do software, em suas perspectivas de funcionalidade, mas também na forma como os usuários interagem e se relacionam com os sistemas [Ordones 2023, Sousa e Furtado 2003, Ferre 2003, Barbosa et al. 2021]. Essa interdisciplinaridade é vital para criar produtos que funcionem corretamente do ponto de vista funcional, mas também atendam de maneira eficiente e satisfatória às necessidades e expectativas dos usuários.

1.2 MOTIVAÇÃO

A presença de sistemas interativos permeando os diversos aspectos da vida em sociedade, faz emergir com criticidade a necessidade de se pensar com mais afinco a qualidade em todo o ciclo de desenvolvimento [Barbosa et al. 2021]. Como explicitado anteriormente, os processos de efeito-causa, que elevam a demanda por qualidade nos produtos de software e, conseqüentemente, adicionam novas camadas de complexidade ao processo de desenvolvimento, evidenciam algumas exigências [Neumann et al. 2021]. Uma delas é o fornecimento de uma experiência intuitiva e eficaz como uma prioridade absoluta para organizações que buscam garantir a aceitação e adesão de seus produtos de software no mercado [Preece et al. 1993, Barbosa et al. 2021]. No entanto, algumas características e peculiaridades de ambientes e regras de negócio, agregam significativamente mais complexidade, como o caso de software que atua em ambientes dinâmicos e com múltiplos *stakeholders* [Lewellen 2021, Andreescu e Mircea 2014].

Compreende-se dinâmico o ambiente caracterizado por mudanças frequentes, imprevistas ou não, muitas vezes abruptas, que influem nos critérios, regras de negócio, funcionalidades e expectativas do software ainda em desenvolvimento e/ou já em uso. Essas mudanças podem ser desencadeadas por diversos fatores, como a evolução natural do ciclo de vida, a partir de *feedbacks* dos usuários, novas tendências e avanços tecnológicos [Zhang et al. 2003, Stallinger e Grünbacher 2001], interpostos nos processos de manutenibilidade, mas também por algumas características específicas de alguns domínios que lidam com a dependência à legislação, por exemplo, enquanto requisitos regulatórios. No entanto, este cenário leva à volatilidade dos requisitos do software, exigindo processos incrementais e iterativos que respondam de forma eficiente às demandas emergentes [Larman e Basili 2003]. É nesse cenário é que se justifica o surgimento dos métodos ágeis, onde sua flexibilidade é evidenciada como resposta a ineficiência dos métodos tradicionais para lidar com a natureza dinâmica do software contemporâneo [Amajuoyi et al. 2024, Tetteh 2024].

Já um ambiente com múltiplos *stakeholders* se caracteriza pela presença de várias par-

tes interessadas, de diferentes níveis de interesses, necessidades, participação e expectativas em relação ao software em desenvolvimento. A multiplicidade de partes interessadas é uma característica comum em projeto de desenvolvimento, porém em alguns mais específicos, o desafio reside na criação e adoção de abordagens capazes de lidar com a diversidade destes interesses e necessidades quando a visão de cliente não pertence somente a uma parte interessada [Mishra et al. 2008]. O impacto desse cenário no ciclo de vida do software é significativo, pois as diferentes visões dos *stakeholders* podem ser conflitantes e contraditórias, que refletem na definição, priorização e gestão dos requisitos do software [Lewellen 2020]. Além disso, a comunicação entre os *stakeholders* pode ser complexa, uma vez que podem haver ruídos na comunicação e a dificuldade de colaboração varia desde a distribuição geográfica até uma compreensão abrangente do domínio do problema, devido o aumento no risco de mal-entendimentos [Constantino et al. 2019].

A complexidade é ainda maior quando os dois contextos, de ambientes dinâmicos e de ambientes com múltiplos *stakeholders*, estão associados a um único ciclo de vida de software. Pois, o desafio está em mapear, construir e/ou adaptar abordagens que considerem o impacto significativo dos múltiplos *stakeholders* a partir de seus interesses, bem como lidar com a dinamicidade do ambiente e como isso influi na estrutura de requisitos do software [Lewellen 2021]. Adentra-se, então, nas questões que permeiam a volatilidade de requisitos e a dificuldade em, nessas circunstâncias, garantir que todo o ciclo de vida de uma aplicação, da elicitação de requisitos à manutenção e evolução, permaneça alinhado com as metas dos stakeholders e do próprio domínio [Graessler et al. 2022].

Considerando todas essas nuances, as motivações desta investigação estão em diversos aspectos. O primeiro deles, leva em conta a necessidade de se aprofundar acerca da comunicação entre os múltiplos stakeholders em ambientes dinâmicos e, além disso, se aprofundar nas questões relacionadas a sub-representação das partes interessadas, principalmente do usuário final, no ciclo de vida do software [Nagel et al. 2023, Hoy e Xu 2023, Wang et al. 2024, Binder et al. 2023]. Outro fator importante é a ausência de práticas que repensem a gestão e ciclo de vida de requisitos voláteis para mitigar seus impactos na arquitetura de software e no planejamento de projetos [Jarzębowicz e Sitko 2020, Dasanayake et al. 2019, Hein et al. 2021]. Principalmente considerando que a gestão não ágil do ciclo de vida de requisitos voláteis pode levar a perda de utilidade da solução computacional desenvolvida. Por fim, considera-se como objetivo motivador, a ausência, no estado da prática, de técnicas, ferramentas e/ou abordagens que melhor direcionem o monitoramento de contextos dinâmicos [Binder et al. 2023, Limaylla-Lunarejo et al. 2023,

Hoy e Xu 2023, Wang et al. 2024].

Nesse contexto, justifica-se a necessidade de se investigar com criticidade como os diferentes domínios tratam os impactos de ambientes dinâmicos e com múltiplos stakeholders na volatilidade de requisitos. Considerando os insights concebidos no levantamento de Trabalhos Correlatos e percorridos no Capítulo 3, embora a literatura atual neste campo, demonstre um esforço significativo para compreender e mitigar os desafios associados às constantes mudanças nas demandas de software, ainda assim precisam de aprofundamento nos aspectos citados.

1.3 OBJETIVOS

1.3.1 Objetivo geral

Este trabalho investiga como a dinamicidade do ambiente e a presença de múltiplos stakeholders têm impactado a volatilidade de requisitos no processo de desenvolvimento de software governamental. A investigação analisa todo o ciclo de desenvolvimento, que inicia na fase de elicitação de requisitos e vai até a manutenção e evolução de soluções computacionais. A partir dessa condução, busca-se especificar um conjunto de estratégias e boas práticas que apoiem a gestão do ciclo de vida de desenvolvimento de aplicações que atuam em ambientes dinâmicos e com múltiplos stakeholders. Considera-se, ainda, a especificação de um processo ágil de desenvolvimento que envolva essas estratégias e melhor lide com a volatilidade de requisitos e a gestão de interesses sobre o produto.

1.3.2 Objetivos específicos

Para ser possível alcançar o objetivo geral deste estudo, os seguintes objetivos específicos serão adotados no percurso desta pesquisa:

- Mapear na literatura as abordagens acerca da volatilidade de requisitos de softwares que atuam em ambientes dinâmicos e com múltiplos stakeholders;
- Conduzir investigação aprofundada em um ambiente real, dinâmico e com múltiplos stakeholders, considerando todos os aspectos do seu ciclo de desenvolvimento;
- Realizar uma inspeção para análise de artefatos e demais documentações geradas no ciclo de desenvolvimento do software estudado;

- Conduzir entrevistas semi-estruturadas com múltiplos stakeholders que compõem o contexto prático estudado, para colher percepções e problemáticas sobre o ciclo de vida de desenvolvimento ao qual estão inseridos.
- Selecionar e aplicar método válido para avaliação de usabilidade do software investigado diretamente com usuários finais;
- Realizar uma análise temática através das respostas abertas do formulário, para consolidar insights sobre as percepções dos usuários finais;
- Definir um conjunto de boas práticas para adoção no ciclo de vida do software que atua em ambientes dinâmicos e com múltiplos stakeholders, além de definir um processo de desenvolvimento baseado nessas boas práticas.

1.4 QUESTÃO DE PESQUISA

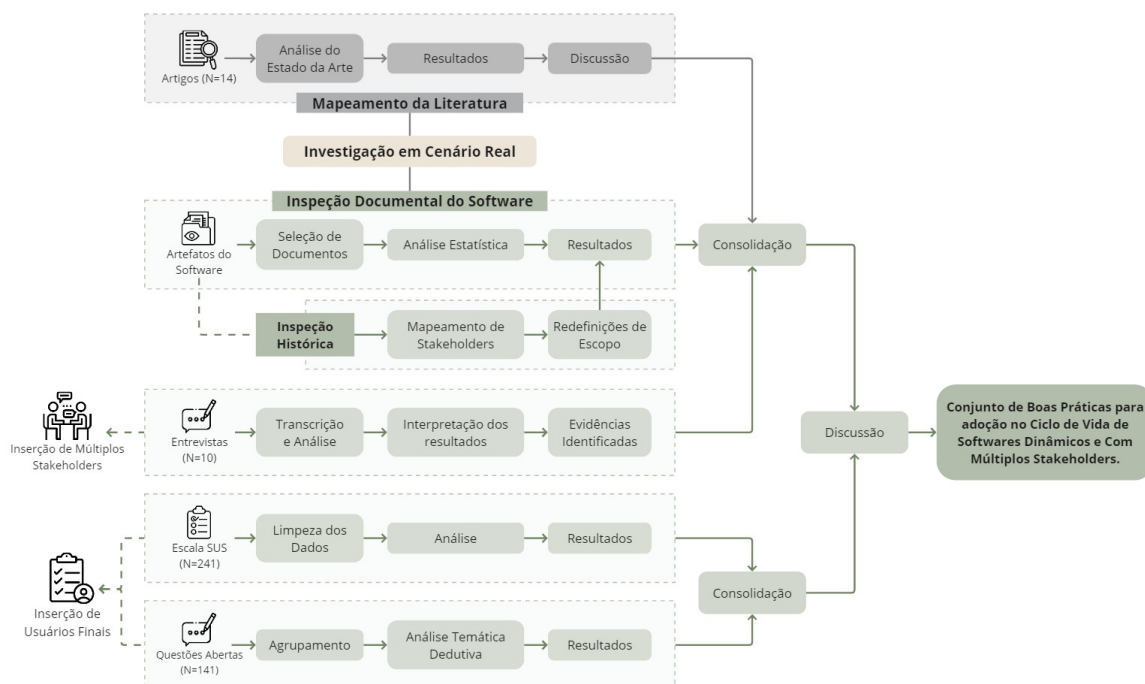
Assim, este trabalho propõe-se a responder a seguinte questão de pesquisa:

- Como a dinamicidade do ambiente e a presença de múltiplos stakeholders impactam a volatilidade de requisitos no ciclo de vida de software?

1.5 METODOLOGIA

Esta dissertação considerou quatro fases que contemplam a Revisão da literatura, uma Inspeção Documental do Software apoiada por uma inspeção histórica, entrevistas com profissionais de TI e demais stakeholders do projeto, e uma avaliação de usabilidade baseada em métodos mistos. Dessa forma, a Figura 1 apresenta a estrutura de execução.

Figura 1 – Percurso metodológico



Fonte: Autoria própria, 2024

No primeiro momento, foi realizada uma revisão da literatura para entender os principais conceitos e práticas relacionados a volatilidade de requisitos. Esta análise adaptou o método de mapeamento de [Budgen et al. 2008] e teve como objetivo mapear as dificuldades e limitações encontradas no estado da arte no tocante à volatilidade de requisitos em ambientes dinâmicos e com múltiplos stakeholders. Em sequência, os resultados dessa análise foram discutidos e sintetizados e apontaram para lacunas importantes, que justificam e consolidam a proposta desta pesquisa.

No segundo momento, foi executada uma inspeção documental de um software que atua em ambiente dinâmico e com múltiplos stakeholders. Essa é uma prática importante para garantir a qualidade, consistência e alinhamento dos documentos que descrevem o sistema em desenvolvimento [O'Regan e O'Regan 2018]. Nessa execução, foi considerada uma análise comparativa entre alguns períodos do projeto, para entender como as mudanças no escopo afetaram a execução dos processos que o circundam. Essa inspeção permitiu revisar e validar a exatidão das especificações, dos requisitos e de outros artefatos, como feedback do suporte, documentação técnica e histórias de usuário.

No terceiro momento, foi adotada entrevistas semiestruturadas [Lazar et al. 2017] para

consultar várias partes interessadas e, assim, obter insights sobre o progresso e os desafios dentro do ambiente investigado. Foram considerados gerente do governo e o pessoal de gerenciamento de projetos, como o coordenador, o gerente de projeto, o engenheiro de requisitos, o designer de UX e o desenvolvedor de software. Para enriquecimento desses dados, foram considerados também reuniões gerais do projeto, em que diversas partes interessadas de diferentes níveis estavam presentes. Este processo foi de suma importância para captar perspectivas variadas sobre as necessidades, expectativas e desafios relacionados ao sistema em desenvolvimento, bem como, identificar potenciais riscos e inconsistências.

No quarto momento, foi conduzida uma avaliação de usabilidade junto aos usuários finais de um software que atua em ambiente dinâmico e com múltiplos stakeholders, para isso foi considerando uma abordagem apoiada em métodos mistos. O método quantitativo selecionado foi o *System Usability Scale* (SUS) de [Brooke 2013]. A escolha se deu a partir de alguns requisitos que levaram em consideração as características do contexto e dos participantes. Em simultâneo, para dar mais robustez a análise quantitativa, aderiu-se a uma análise temática dedutiva, baseada na proposta de [Braun e Clarke 2006], utilizando categorias previamente definidas. Essa abordagem, permitiu captar padrões que compõem os resultados qualitativos da pesquisa.

1.6 ORGANIZAÇÃO DO TRABALHO

Esta dissertação está organizada em capítulos, dos quais este é o primeiro e apresenta as considerações iniciais sobre o trabalho, incluindo uma contextualização, as motivações, os objetivos e a metodologia. Os demais capítulos encontram-se organizados da seguinte maneira:

No Capítulo 2 é apresentada a fundamentação teórica, onde se discorre sobre a complexidade de desenvolvimento do software que atua em ambientes dinâmicos e com múltiplos stakeholders. Complementar a isto, aponta-se como a Interação Humano-Computador e a Engenharia de Software são integradas no ciclo de vida do software. Complementar a isso, traz-se o ciclo de vida ágil, suas nuances e capacidade de lidar com as complexidades inerentes a ambientes dinâmicos e com múltiplos stakeholders.

No Capítulo 3 é disposto uma análise da literatura visando investigar os principais conceitos e práticas relacionados a volatilidade de requisitos. São percorridas nesse capítulo as dificuldades e limitações encontradas no estado da arte no tocante à volatilidade de requisitos em ambientes dinâmicos e com múltiplos stakeholders.

No Capítulo 4, é apresentada a condução prática deste estudo, que investiga um software

governamental desenvolvimento em ambiente altamente dinâmico e com múltiplos stakeholders. Nesse capítulo são apresentados os aspectos éticos da pesquisa e a aplicação de métodos mistos, que executou uma inspeção aprofundada nos artefatos gerados no ciclo de vida do software. Foi possível aferir o nível de usabilidade do sistema investigado, e qualitativamente foi possível analisar as percepções dos usuários e demais partes interessadas que atuam em diferentes níveis do projeto.

No Capítulo 5, apresenta-se um conjunto de boas práticas para adoção no ciclo de vida do software que atua em ambientes dinâmicos e com múltiplos stakeholders. Essas boas práticas são se apoia em uma representação visual, que considera a projeção de um processo de desenvolvimento de software ágil para acomodar as recomendações.

Por fim, o capítulo 6 apresenta as conclusões do trabalho, destacando as principais contribuições, as ameaças à validade e as perspectivas para os próximos passos em pesquisas futuras.

2 REFERENCIAL TEÓRICO

Duas áreas importantes nos processos de desenvolvimento de softwares interativos são a Engenharia de Software (ES) e a Interação Humano-Computador (IHC). É notório que as duas áreas percorrem caminhos distintos e apontam para perspectivas diferentes quanto aos fatores importantes em sistemas interativos, sobre o que significa utilizá-los e sobre como desenvolvê-los [Barbosa e Silva 2010]. Mas, ainda assim, anuem, entre si, o objetivo de suprir as diversas necessidades dos usuários por meio de "sistemas de qualidade"[Pressman 2011].

No contexto de software, qualidade pode ser entendida como um conjunto de características a serem satisfeitas, de modo que o entregável atenda às necessidades explícitas e implícitas de seus usuários [Standardization 2023]. Em síntese, qualidade está diretamente ligada a conformidade de requisitos [Koscianski e Soares 2007], ideia essa que corrobora com a definição de [Pressman 2011], que enxerga qualidade como a conformidade com requisitos funcionais e não funcionais, levando em conta as características implícitas e desempenho explicitamente declarados.

Considerando tais aspectos, este capítulo será disposto em três seções voltadas à demonstração de teorias e conceitos que embasam e que são fundamentais para esta pesquisa. A primeira seção foca, de forma ampla, em como a Interação Humano-Computador e a Engenharia de Software são integradas no ciclo de vida do software. Em seguida, na segunda seção, o foco estará na complexidade de desenvolvimento do software que atua em ambientes dinâmicos e com múltiplos stakeholders. E, por último, será abordado o ciclo de vida ágil, suas nuances e capacidade de lidar com as complexidades inerentes a ambientes dinâmicos e com múltiplos stakeholders.

2.1 IHC E ENGENHARIA DE SOFTWARE NO CICLO DE VIDA DO SOFTWARE

Para a ES, um software é um conjunto de programas de computador e a documentação que lhe é associada [Sommerville 2011], de outro lado, compreendendo sistemas interativos, objeto comum em IHC, embora ainda se sirva desse conceito, dar-se ênfase ao fato de que os produtos de software são manipulados por usuários, o que configura a interação entre homem e máquina. Contudo, a qualidade dessa interação está diretamente relacionada com a modelagem dos componentes de um sistema interativo: trata-se da aplicação e da interface [Barbosa et al. 2021, Preece et al. 1993]. Assim, pode-se compreender que, enquanto a ES fornece os princípios e práticas para criar sistemas robustos, escaláveis e confiáveis, a IHC se concentra na concepção de

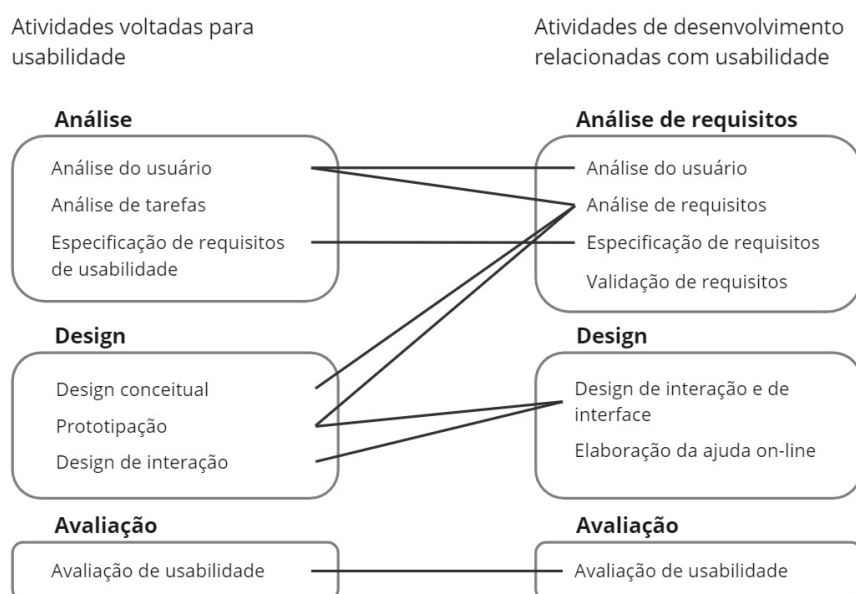
interfaces e interações que sejam intuitivas, acessíveis e satisfatórias para os usuários. Portanto, a união das duas áreas de conhecimento se faz crucial para a otimização dos produtos de software [Santos 2023].

Atualmente, há uma busca crescente por processos de desenvolvimento que integrem a IHC à ES [Neto et al. 2020]. Esse esforço compreende não somente o ponto de vista do sistema e suas abrangências funcionais, mas também como se dá o uso destes sistemas do ponto de vista dos seus usuários. Diversas iniciativas acadêmicas e profissionais têm se concentrado nessa integração, como ilustrado por [Barbosa e Silva 2010], que propõem três abordagens significativas:

definição de características de um processo de desenvolvimento que se preocupa com a qualidade de uso; definição de processos de IHC paralelos que devem ser incorporados aos processos propostos pela ES; indicação de pontos em processos propostos pela ES em que atividades e métodos de IHC podem ser inseridos.

Essas percepções de integração norteiam e explicam diversos processos de design de IHC pensados para a construção de sistemas interativos. Quando acentuadas e identificadas as características constituintes de um projeto de software, é importante considerar os paralelismos entre IHC e ES, como mostrado na Figura 2. Afunilando os interesses e tendo como foco adentrar processos específicos do desenvolvimento de um software, percebem-se ainda mais de perto algumas oportunidades para a desejada integração entre as áreas de conhecimento.

Figura 2 – Integração de IHC com Engenharia de Software



Fonte: [Barbosa e Silva 2010]

A integração é uma via para que as equipes de desenvolvimento possam considerar que os sistemas devem ser projetados e implementados levando em consideração as necessidades, habilidades e expectativas dos usuários [Silva e Baranauskas 2020]. Como resultado, tem-se mais assertividade com interfaces mais amigáveis, que reduzem a curva de aprendizado e aumentam a eficiência e a produtividade do usuário [Barbosa et al. 2021]. Além disso, uma abordagem centrada no usuário promove a aceitação e a adoção do sistema, contribuindo para o sucesso do projeto [Pagnan et al. 2019].

No geral, considerar a união de práticas e abordagens de IHC e ES é um passo importante para a continuidade de melhoria e otimização da qualidade do software [Neto et al. 2020]. Porém, é necessário explicitar que não se trata de uma mudança simples, pois há uma alteração na estrutura e nos processos convencionais, que instiga a uma nova cultura e dinâmica do ciclo de vida, conforme explicitado nas atividades e perspectivas da Figura 2. Mas, ainda assim, é válido ressaltar os ganhos dessa prática, considerando desde coleta de *feedbacks* dos usuários e a realização de testes de usabilidade ao longo do ciclo de vida do desenvolvimento, onde as equipes podem identificar e corrigir problemas de usabilidade antes que o sistema seja lançado e considerando quem de fato usa o sistema [Barbosa et al. 2021].

Dessa forma, assumir uma abordagem holística que procura a produção de soluções computacionais mais refinadas e satisfatórias para os usuários e que também otimize as atividades de manutenibilidade é importante considerando a busca crescente por qualidade. Neste sentido, é importante considerar o contexto, as interferências e características que influem na complexidade por trás de um software.

2.2 COMPLEXIDADES NO DESENVOLVIMENTO DO SOFTWARE

A complexidade no desenvolvimento de software está cada vez mais evidente, impulsionada pela necessidade de atender aos diversos contextos, o que eleva o desafio de gerenciar e adaptar requisitos ao longo do ciclo de vida do projeto. Alguns ambientes contam com uma rápida evolução tecnológica, o que impacta diretamente os requisitos, pois demandas e expectativas mudam constantemente, exigindo do software uma capacidade de adaptação frequente [Witschel et al. 2019, Paranjpe e Pachlor 2024]. Nesse sentido, essas são algumas complexidades adicionadas quando o software atua em ambientes dinâmicos e com múltiplos stakeholders. Assim, essa seção se dedica a evidenciar essas complexidades.

2.2.1 Ambientes Dinâmicos e com Múltiplos Stakeholders

O dinamismo de um ambiente está fortemente ligado à volatilidade dos requisitos de software. Diversos fatores externos e internos influenciam como o sistema é concebido e projetado ao longo de seu ciclo de vida, considerando a evolução constante dos recursos tecnológicos, que introduzem novas oportunidades e requerem atualizações contínuas para preservar a relevância, a utilidade e a eficácia do sistema [Moyon et al. 2018]. Esse cenário resulta em mudanças frequentes nos requisitos do sistema, demandando processos incrementais e iterativos que respondam de forma ágil e eficaz às demandas emergentes [Larman e Basili 2003].

Tradicionalmente, ambientes de desenvolvimento de software são naturalmente dinâmicos, o que motivou a adoção de metodologias ágeis projetadas para gerenciar mudanças de maneira eficiente [Alzeyani e Szabó 2023]. Entretanto, ambientes de alta complexidade e extrema dinamicidade apresentam características distintas e únicas, que desafiam essas metodologias. Em particular, os fatores imprevisíveis e o rápido ritmo de transformação geram requisitos voláteis e requerem adaptações rápidas e contínuas, muitas vezes superando a capacidade de resposta das práticas ágeis convencionais [Szabó e Hercegi 2023].

Nesses ambientes altamente dinâmicos, as mudanças de requisitos ocorrem de forma mais frequente e imprevisível em comparação com configurações tradicionais. A alta volatilidade é impulsionada por avanços tecnológicos acelerados, mudanças regulatórias e a evolução das necessidades dos stakeholders [Pohl 2010, Mohammad e Kollamana 2024]. Esses fatores, ao mesmo tempo que abrem novas possibilidades, também impõem desafios significativos para o gerenciamento de requisitos e para a manutenção da coesão entre as diferentes partes envolvidas [Lewellen 2021].

Além disso, a presença de múltiplos stakeholders agrava essa complexidade. Cada grupo de stakeholders — desde equipes internas e parceiros externos até órgãos reguladores e usuários finais — traz consigo objetivos e expectativas únicas, muitas vezes conflitantes, que aumentam a dificuldade na tomada de decisões [Zhang et al. 2008]. Essa diversidade de perspectivas contribui para a fragmentação do entendimento do produto e eleva o risco de desalinhamentos e conflitos de interesse, exigindo um gerenciamento contínuo e colaborativo dos requisitos para evitar desentendimentos e retrabalho [Lewellen 2020].

Outro fator desestabilizador são as mudanças frequentes na composição dos stakeholders. Em ambientes dinâmicos, a alta rotatividade ou o surgimento de novos participantes trazem novas prioridades e perspectivas, que podem desestabilizar o andamento de projetos em fase avançada e exigir um realinhamento ágil das metas e dos requisitos [Lewellen 2021, Ambreen 2019]. Essas

alterações na base de stakeholders impõem a necessidade de adaptação rápida e a habilidade de absorver e integrar novas demandas sem comprometer o progresso já realizado, o que exige flexibilidade e resiliência das equipes de desenvolvimento [Ambreen 2019].

Ambientes fortemente influenciados por fatores externos, como políticas públicas e interesses políticos, enfrentam pressões adicionais para adaptar-se constantemente a marcos legislativos e regulatórios [Ambreen 2019]. Essa necessidade de conformidade adiciona uma camada extra de dinamismo, visto que mudanças políticas e regulamentações podem surgir abruptamente, demandando ajustes imediatos. As pressões externas representam não apenas um desafio técnico, mas também uma complexidade estratégica, pois o sistema deve se moldar para atender exigências legais e ainda ser capaz de responder a futuras mudanças de forma ágil [Lewellen 2021].

Outro ponto relevante é o impacto das exigências de comunicação e coordenação em ambientes onde diversos stakeholders estão distribuídos geograficamente ou possuem níveis de conhecimento variados sobre o domínio do problema [Zhang et al. 2003]. Em cenários complexos e dinâmicos, a colaboração pode ser dificultada por barreiras culturais, fusos horários e diferentes níveis de entendimento sobre o projeto. A comunicação ruidosa ou incompleta é uma consequência comum, elevando a dificuldade para construir uma visão integrada e compartilhada do produto. Este desafio requer não apenas estratégias de comunicação eficazes, mas também uma infraestrutura robusta para apoiar a troca de informações em tempo real [Kaplan et al. 1997].

A combinação de interesses diversos e complexidade ambiental exige uma abordagem de tomada de decisão que vá além das práticas tradicionais. É necessário um processo de decisão adaptativo e orientado pela análise contínua das demandas e das mudanças contextuais, assegurando que o desenvolvimento do software permaneça alinhado com as metas dos stakeholders [Burgueño et al. 2024]. A capacidade de negociação e mediação é essencial para harmonizar expectativas divergentes, gerenciar conflitos de interesse e sustentar uma colaboração produtiva e focada nos objetivos comuns do projeto [Ningsih et al. 2022].

Por fim, a intersecção de todos esses fatores em um único ambiente exige que as equipes de desenvolvimento adotem abordagens adaptativas e que sejam aderentes ao contexto, logo, assumem-se iniciativas ágeis no escopo dos projetos, mas também, adere-se ao pensamento sistêmico e adaptativo [Anaba et al. 2024]. A flexibilidade organizacional e o aprendizado contínuo tornam-se essenciais para a adaptação rápida às novas demandas e à superação dos desafios impostos pelo dinamismo extremo. Assim, o sucesso em ambientes de alta complexidade depende de uma capacidade organizacional de resposta ágil, uma visão compartilhada de

objetivos e uma resiliência que permita a sustentação de ciclos de adaptação e inovação contínua [Szabó e Hercegi 2023].

2.2.2 Volatilidade de Requisitos no Desenvolvimento de Software

A volatilidade dos requisitos (VR) representa um desafio central no desenvolvimento de software, definida pela instabilidade, modificação ou exclusão de requisitos ao longo do ciclo de desenvolvimento [Siakas et al. 2022]. Essa volatilidade surge de fatores internos e externos que influenciam a concepção e o design do sistema, incluindo requisitos incompletos, avanços tecnológicos constantes, mudanças nas necessidades dos stakeholders, expansão do escopo, falhas de comunicação e alterações em normas ou regulamentações [Mohammad e Kollamana 2024]. Tais elementos compõem um ambiente de desenvolvimento que demanda flexibilidade e adaptabilidade para atender a ajustes contínuos necessários ao sucesso do software [Andreescu e Mircea 2014].

O dinamismo do ambiente está fortemente ligado à volatilidade dos requisitos de software. Os avanços tecnológicos introduzem novas oportunidades e exigem atualizações contínuas para manter a relevância, utilidade e eficácia do sistema, resultando em mudanças frequentes nos requisitos [Graessler et al. 2022]. Para responder a essas demandas emergentes de maneira ágil e eficiente, são necessários processos incrementais e iterativos [Mohammad e Kollamana 2024]. Em contextos de desenvolvimento mais tradicionais, essas mudanças geralmente podem ser antecipadas e gerenciadas por meio de metodologias ágeis, as quais acomodam requisitos dinâmicos por meio de ciclos iterativos e feedback dos stakeholders [Pohl 2010].

No entanto, ambientes com múltiplos stakeholders introduzem complexidades adicionais. Esses ambientes envolvem uma diversidade de participantes, cada um com perspectivas únicas sobre o produto de software, moldadas por seus interesses e expectativas individuais [Lewellen 2021]. A presença de stakeholders com interesses divergentes dificulta a tomada de decisões, pois exige o alinhamento de expectativas e a resolução de conflitos para manter a coesão do projeto [Mohammad e Kollamana 2024]. Além disso, desafios como ruídos na comunicação e dificuldades de colaboração — de distribuição geográfica a variações no entendimento do problema — complicam ainda mais o gerenciamento do projeto, tornando fundamental a implementação de estratégias de comunicação eficazes [Mohammad e Kollamana 2024].

Essa complexidade aumenta ainda mais quando contextos dinâmicos e com múltiplos stakeholders convergem em um único produto de software. Este ambiente de "contexto duplo" exige a reconciliação de diferentes necessidades e expectativas de stakeholders, enquanto requer uma compreensão profunda do domínio do problema em um cenário de transformação

constante [Andreescu e Mircea 2014, Zhang et al. 2008]. Diferentemente de ambientes de desenvolvimento convencionais, onde as mudanças são impulsionadas por avanços tecnológicos e feedback dos usuários, esses ambientes altamente dinâmicos e multifacetados são frequentemente impactados por mudanças na composição dos stakeholders e por interesses conflitantes, frequentemente motivados por pressões políticas ou organizacionais [Zhang et al. 2008].

Por fim, a presença de múltiplos stakeholders e a volatilidade dos requisitos demandam uma infraestrutura robusta e ferramentas colaborativas que facilitem o gerenciamento da complexidade e a comunicação eficaz entre as partes [Demissie et al. 2021]. Ferramentas de controle de versão, plataformas de integração contínua e mecanismos de registro transparente de decisões e contribuições são fundamentais para promover um ambiente de colaboração fluida, auxiliando no alinhamento contínuo entre stakeholders, independentemente de suas localizações ou níveis de entendimento sobre o projeto [Balasubramaniam et al. 2024].

2.3 METODOLOGIAS ÁGEIS E RESPOSTAS A AMBIENTES DINÂMICOS

As metodologias ágeis surgiram como uma resposta à necessidade de adaptabilidade e eficiência em ambientes de desenvolvimento de software, características comuns em ambientes caracterizados por dinamicidade e volatilidade [Hooda et al. 2023]. Diferente dos métodos tradicionais, como o modelo cascata, que segue uma sequência rígida de fases, as metodologias ágeis propõem uma abordagem iterativa e incremental, onde o desenvolvimento é realizado em ciclos curtos, conhecidos como *Sprints* [Nakash 2024]. Essa estrutura permite responder rapidamente a mudanças nos requisitos e nas condições externas, garantindo maior flexibilidade e alinhamento com as expectativas dos stakeholders.

A abordagem ágil organiza-se em torno de fases iterativas que, juntas, compreendem um ciclo contínuo de desenvolvimento. Cada uma dessas fases possui objetivos específicos e, em conjunto, são desenhadas para facilitar ajustes rápidos e frequentes consoante aos feedbacks dos stakeholders e as condições do ambiente [Eyieyien et al. 2024]. As principais fases incluem:

- **Planejamento:** Nesta fase, a equipe de desenvolvimento e os stakeholders identificam e priorizam os requisitos mais importantes para a próxima *Sprint*. Diferentemente dos métodos tradicionais, onde todo o escopo é definido no início do projeto, o planejamento ágil se concentra em uma visão geral do produto e um conjunto de funcionalidades específicas para o *Sprint*. Essa priorização é orientada pelos requisitos de negócio mais críticos ou aqueles que oferecem maior valor ao cliente [Eyieyien et al. 2024, Silva et al. 2023].

- **Desenvolvimento:** Com os requisitos do *Sprint* definidos, a equipe começa o processo de codificação. O desenvolvimento é realizado de maneira colaborativa, com uma abordagem incremental que facilita a adição de novas funcionalidades ao software em construção. A comunicação é constante, permitindo que membros da equipe discutam problemas e compartilhem soluções rapidamente, mantendo todos atualizados sobre o andamento do trabalho. A fase de desenvolvimento busca garantir que as funcionalidades desenvolvidas estejam alinhadas aos requisitos do *Sprint*, permitindo ajustes rápidos em caso de mudanças ou novos feedbacks [Rajakumari et al. 2024, Guerrero-Vásquez et al. 2023].
- **Testes e Qualidade:** Testes são realizados ao longo do processo de desenvolvimento, uma característica essencial das metodologias ágeis. O objetivo é identificar e corrigir falhas rapidamente, minimizando o retrabalho e garantindo que as funcionalidades atendam aos padrões de qualidade antes de serem entregues. Testes automatizados e manuais são utilizados para verificar a consistência e funcionalidade do sistema, e feedbacks imediatos dos testes garantem que as mudanças necessárias sejam feitas de maneira ágil e eficaz [Suganya et al. 2023, Bhanushali 2023, Kaur et al. 2023].
- **Revisão e Feedback:** Ao final de cada *Sprint*, ocorre uma revisão das funcionalidades desenvolvidas, permitindo que stakeholders revisem o progresso e forneçam feedback sobre o produto parcial. Esta fase é fundamental para garantir que o software atenda às expectativas e requisitos dos stakeholders, além de permitir ajustes no planejamento dos próximos *Sprints* com base nas necessidades emergentes. O feedback é incorporado ao planejamento, promovendo a melhoria contínua e a adaptação às novas demandas e mudanças nos requisitos [Ribeiro e Alves 2023, Begier e Wdowicki 2006].
- **Retrospectiva:** Após a revisão, a equipe realiza uma análise retrospectiva da *Sprint*, identificando o que funcionou bem e o que pode ser melhorado no próximo ciclo. Essa fase promove a autoavaliação e o aprendizado contínuo, buscando otimizar processos e eliminar ineficiências. A retrospectiva incentiva um ambiente de aprendizado onde a equipe pode discutir abertamente suas experiências e encontrar maneiras de aperfeiçoar o trabalho [Galeno et al. 2024, Torres e Cornide-Reyes 2022].

As metodologias ágeis são altamente eficazes em ambientes dinâmicos, pois permitem uma rápida adaptação a mudanças nos requisitos e nas condições externas [Amajuoyi et al. 2024]. Ao trabalhar em ciclos curtos, a equipe de desenvolvimento pode incorporar novas necessidades,

feedbacks e demandas regulatórias com facilidade [Rajakumari et al. 2024]. Esse modelo ágil de resposta minimiza o risco de desalinhamento entre o produto final e as expectativas dos stakeholders, garantindo que o software seja desenvolvido de maneira a atender às mudanças constantes e às exigências de um ambiente volátil.

Além disso, o feedback constante dos stakeholders ajuda a reduzir a incerteza, permitindo que o produto evolua de forma mais precisa e alinhada aos objetivos de negócio [Ribeiro e Alves 2023]. Em ambientes de múltiplos stakeholders, onde as perspectivas e prioridades podem ser conflitantes, as metodologias ágeis facilitam a integração de diferentes interesses ao longo do processo de desenvolvimento, contribuindo para um produto mais coeso e adaptado às demandas de um contexto dinâmico [Grubb e Spoletini 2023].

Assim busca-se entender como as metodologias ágeis, com suas fases bem-definidas e ciclo iterativo, podem oferecer uma estrutura robusta e flexível para responder à volatilidade e complexidade de ambientes de desenvolvimento de software, possibilitando que a equipe mantenha o foco nas prioridades de negócio e nos valores que precisam ser entregues ao cliente, enquanto se adapta às mudanças frequentes e complexidades inerentes ao cenário atual.

3 REVISÃO DA LITERATURA

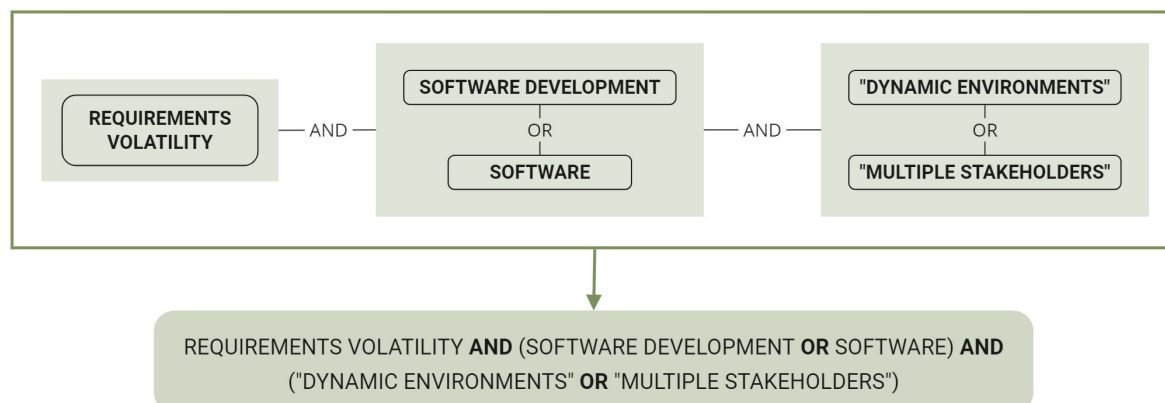
Este capítulo tem o objetivo de identificar e analisar como a volatilidade de requisitos tem sido investigada considerando ambientes de aplicação dinâmicos e dotados de múltiplos *stakeholders*, por meio de uma revisão da literatura. Assim, este capítulo apresenta a seleção de estudos relacionados, a partir das seguintes seções: A Seção 3.1 apresenta os objetivos e a estratégia de busca. A Seção 3.2 traz uma síntese dos estudos selecionados. Chegando à Seção 3.3, são apontados e discutidos os estudos selecionados. E por fim, na seção 3.4, são apresentadas as considerações finais deste capítulo.

3.1 PERCURSO METODOLÓGICO PARA SELEÇÃO DE ESTUDOS

Visando identificar e analisar as contribuições recentes acerca da temática delimitada, optou-se pela execução de uma revisão da literatura, para mapear e interpretar os estudos disponíveis e com relevância para uma determinada questão de pesquisa, área de tópico ou fenômeno de interesse. Para isso, foi usada uma adaptação do mapeamento sistemático de [Budgen et al. 2008]. Alguns motivos para se desempenhar um mapeamento no estado da arte, segundo o autor, destacam-se: **i.** resumir as evidências existentes sobre processos ou tecnologias, bem como analisar as evidências empíricas em termos de benefícios e limitações dos métodos; **ii.** Identificar lacunas na pesquisa atual, a fim de propor a ampliação das áreas já trabalhadas e/ou sugerir novas áreas para uma investigação mais aprofundada, podendo fornecer estrutura para posicionar adequadamente novas atividades de pesquisa.

O objetivo central dessa revisão está em identificar e analisar artigos atuais identificados no estado da arte que abordam vários aspectos da volatilidade de requisitos na engenharia de software, destacando suas contribuições e descobertas por todos os aspectos do gerenciamento de requisitos. Para isso, foi construída uma *String* de busca com os principais termos relacionados, conforme especificado na Figura 3.

Figura 3 – Desafios e Lacunas Encontrados



Fonte: Autoria própria (2023)

Essa string de busca foi construída para explorar a relação entre a volatilidade de requisitos e o desenvolvimento de software, considerando contextos desafiadores que envolvem ambientes dinâmicos e múltiplos stakeholders. Cada elemento da String tem uma justificativa específica:

- **(Requirement Volatility):** O termo central da pesquisa foca na volatilidade de requisitos, um fenômeno recorrente em projetos de software, onde mudanças frequentes e imprevisíveis nos requisitos podem impactar negativamente o desenvolvimento.
- **AND (Software Development OR Software):** A inclusão de "Software Development" ou simplesmente "Software" amplia o escopo para cobrir estudos que abordem tanto o processo de desenvolvimento quanto aspectos gerais relacionados ao software. O uso do operador OR garante a inclusão de documentos que utilizem diferentes terminologias.
- **AND ("Dynamic Environments" OR "Multiple Stakeholders"):** Esses termos complementam a pesquisa ao direcionar o foco para contextos específicos e complexos. "Dynamic Environments" refere-se a situações de rápida mudança e incerteza, enquanto "Multiple Stakeholders" enfatiza o impacto de múltiplos interessados no processo de tomada de decisão, frequentemente associados à volatilidade de requisitos.

Uma vez definida a String de busca, para realizar a seleção de estudos, pensando atingir o maior número de trabalhos relevantes dentro da temática aqui abordada, consideraram-se as principais bases de dados de Ciência da Computação e Engenharia de Software. Contudo, foram escolhidas a ACM Digital Library, IEEE Xplore, Springer Link e ScienceDirect. Além do

fator relevância desses repositórios, a escolha também se justifica pela possibilidade de acesso institucional via Portal de Periódicos da CAPES (Coordenação de Aperfeiçoamento de Pessoal de Nível Superior). Acrescenta-se ainda que os repositórios selecionados apontam para estudos presentes em Conferências, *workshops* e periódicos das áreas mencionadas.

Cada estudo retornado foi avaliado considerando quatro aspectos temáticos, podendo haver interseções ou não, os aspectos foram:

- Estudos que consideram a multiplicidade de stakeholders como fator que impacta a Volatilidade de Requisitos.
- Estudos que consideram práticas ágeis para lidar com a Volatilidade de Requisitos
- Estudos que, além de identificar desafios, propõe boas práticas ou ferramentas para lidar com a Volatilidade de Requisitos.
- Estudos que conduziram experimentos empíricos em contextos reais ou com dado reais dentro da temática delimitada.

É importante ressaltar que a primeira busca nos repositórios aconteceu nos mês de Agosto de 2024 e voltou a ser executada entre os meses de outubro e novembro de 2024. Considerando o baixo número de estudo aderentes nos últimos 5 anos, ampliou-se a busca para os últimos 15 anos, ou seja, de 2009 a 2024.

3.2 AVALIAÇÃO DOS ESTUDOS ADERENTES

3.2.1 Análise do Estado da Arte

Com a investigação crescente acerca da integração de IHC a ES e dos impactos que essa integração ou não podem causar num produto final de software, grandes discussões têm sido levantadas. As atividades da Engenharia de Requisitos têm se apresentado como um foco central nessa discussão, uma vez que estão nelas os primeiros passos no ciclo de vida de software. Pensar requisitos considerando a volatilidade e as complexidades impostas por ambiente dinâmicos com alta colaboração entre múltiplos stakeholders, com níveis de interesse e perspectivas diferentes sobre o mesmo produto, há tempos, se apresenta como uma preocupação na literatura, sob diferentes domínios e cenários. Assim, pôde-se perceber algumas abordagens registradas na literatura e descritas a seguir em ordem cronológica.

Na condução de [Aziz e Wong 2015] foi investigado como o conhecimento das relações entre requisitos impacta as mudanças nos requisitos e o sucesso de projetos de desenvolvimento

de software. A pesquisa utilizou um modelo de pesquisa com três constructos: Mudança de Requisitos, Conhecimento das Relações entre Requisitos e Sucesso. Utilizando o método PLS (*Partial Least Squares*) com dados de 173 analistas e stakeholders, concluíram que o entendimento dessas interconexões promove adaptações mais eficazes, resultando em maior sucesso do projeto, com impacto indireto mediado pelas mudanças nos requisitos.

Já [Jarzębowicz e Sitko 2020] examinaram práticas de priorização de requisitos no desenvolvimento ágil na indústria de TI Polonesa, respondendo a três perguntas de pesquisa: "Quando a priorização e re-priorização de requisitos acontecem? Quais critérios e técnicas são aplicados durante a priorização? Quem participa das tarefas de priorização?". Um *survey* com 69 profissionais revelou que a priorização ocorre antes de cada iteração, com foco no valor de negócio e uso de técnicas simples. A re-priorização é frequente durante mudanças, e há ampla participação de representantes do cliente e fornecedor. Que o valor de negócio associado a cada requisito é o principal critério de priorização, mas aspectos como interdependências, complexidade e estabilidade dos requisitos também são importantes, que técnicas simples de priorização, que não exigem competências avançadas, como escalas numéricas, são as mais utilizadas.

Considerando que a volatilidade de requisitos é um desafio significativo no desenvolvimento de software, com impacto direto na arquitetura do software, [Dasanayake et al. 2019] exploraram os desafios da volatilidade de requisitos no design de arquitetura de software e como as equipes de software gerenciam esses desafios. Em um estudo de caso com 15 especialistas e se utilizando de uma análise temática das respostas obtidas, eles identificaram impactos significativos mitigados por estratégias como comunicação eficaz, práticas ágeis, documentação e gerenciamento de dívida técnica, destacando a necessidade de adaptação contínua para manter a qualidade e desempenho.

Mesmo considerando que as nuances de processos ágeis de engenharia de requisitos de software permitem respostas rápidas para refletir as mudanças nos requisitos de software do cliente, [Hoy e Xu 2023] realizaram uma revisão sistemática da literatura para identificar desafios e soluções na engenharia de requisitos ágeis. Classificaram os desafios em contexto organizacional, gerenciamento de projetos e metodologia ágil, destacando um desequilíbrio nas soluções existentes. O estudo identificou, ainda, um desequilíbrio entre as soluções e os desafios, com uma concentração de soluções para metodologias ágeis, mas falta de soluções para abordar os desafios decorrentes do contexto organizacional e de negócios e do gerenciamento de projetos ágeis. Propuseram uma estrutura tridimensional que equilibra os três aspectos para melhor enfrentamento dos desafios.

Já o estudo de [Hein et al. 2021], considerou a falta de planejamento ao mudar os requisitos para refletir as expectativas das partes interessadas pode levar a mudanças propagadas que podem causar falhas no projeto. Assim, os autores analisaram técnicas de aprendizado de máquina para prever a volatilidade da mudança de requisitos (RCV) usando métricas de rede complexa. Com dados de uma ferramenta industrial, demonstraram que modelos de aprendizado multirrótulo conseguem prever a volatilidade de mudanças de requisitos de forma eficaz, destacando a importância de métricas de centralidade no desempenho preditivo. Apontou-se, ainda, que métricas de rede complexa podem ser exploradas e computadas para cada requisito, fornecendo *insights* sobre o comportamento dos requisitos em resposta à mudança. Por fim, o estudo identifica métricas de rede complexa úteis nos modelos de Machine Learning de maior desempenho, incluindo métricas de centralidade, que se mostraram importantes na previsão da volatilidade de requisitos.

O estudo de [Wang et al. 2024] propõe uma estrutura para elicitación dinâmica e previsão de requisitos para Sistemas de Produto-Serviço Inteligentes (smart PSS) com base na co-criação de valor entre usuários e fabricantes. Utilizando métodos como DTM (Modelagem de Tópicos Dinâmicos), BERT (*Bidirectional Encoder Representations from Transformers*) e matriz DIPA (*Driving power-Dependence power-Influence power-Autonomy power*), validaram a eficácia da abordagem. Pois, os resultados do estudo empírico demonstram a validade da estrutura proposta, destacando a importância da co-criação de valor na análise de requisitos para Smart PSS. Frisa-se também, como é importante observar que este resumo é baseado nas informações fornecidas nas fontes.

Já [Nagel et al. 2023], focam em como o sucesso de projetos de software depende do desenvolvimento de um sistema que atenda aos desejos e necessidades dos stakeholders, alinhado aos seus modelos mentais sobre o sistema pretendido. Dessa forma, os autores propuseram cinco conceitos para desenvolver um entendimento compartilhado entre stakeholders em contextos assíncronos de comunicação. Assim, o objetivo deste artigo, foi a proposição de cinco conceitos para ajudar os stakeholders a desenvolver um entendimento compartilhado em contextos de comunicação assíncrona. Avaliou-se a adaptabilidade de três ferramentas de software existentes a esses conceitos, adaptando essas ferramentas de acordo e desenvolvendo um protótipo específico que implementa os cinco conceitos. Em um experimento com 30 participantes, avaliaram-se as ferramentas de software e os autores executaram uma comparação com um grupo de controle que não teve suporte para desenvolver um entendimento compartilhado. Como resultados, foi demonstrado uma adequação positiva dos conceitos propostos, pois os participantes que os

utilizaram alcançaram um nível mais alto de entendimento compartilhado em comparação com o grupo de controle.

No artigo de [Ferreira et al. 2009] é apresentado um modelo de simulação de dinâmica de sistemas executável, desenvolvido para ajudar gerentes de projetos a compreender os impactos complexos relacionados à volatilidade de requisitos em projetos de desenvolvimento de software. O simulador expande pesquisas anteriores, incorporando resultados de uma pesquisa empírica, incluindo mais de 50 novos parâmetros derivados dos dados coletados, ao modelo base. O artigo discute resultados detalhados de dois casos que evidenciam impactos significativos nos custos, prazos e qualidade decorrentes da volatilidade de requisitos. O simulador pode ser utilizado como uma ferramenta eficaz para demonstrar o conjunto complexo de relações e efeitos associados à volatilidade de requisitos.

O estudo conduzido por [Binder et al. 2023], considera a avaliação de requisitos por meio do modelo de Kano como uma prática comum, essas avaliações são realizadas entrevistando stakeholders e perguntando sobre o nível de satisfação caso uma determinada funcionalidade seja bem implementada, e o nível de insatisfação caso uma funcionalidade não seja bem implementada. Considerando que avaliações realizadas por entrevistas são demoradas, caras e conseguem captar a opinião de apenas um conjunto limitado de stakeholders. Os autores investigaram a possibilidade de extrair os fatores do modelo de Kano (necessidades básicas, fatores de desempenho, encantadores, irrelevantes) a partir de um grande conjunto de feedbacks de usuários (isto é, avaliações de aplicativos). Foi implementado, treinado e testado, diversos classificadores em um conjunto de 2.592 avaliações. De modo geral, eles propõem alternativa leve e automatizada para identificar fatores do modelo de Kano a partir de um grande conjunto de feedbacks de usuários.

A incompletude em requisitos escritos em linguagem natural é um desafio significativo e uma abordagem comum para identificar lacunas em requisitos é compará-los com fontes externas. Com a ascensão de modelos de linguagem como o BERT, [Luitel et al. 2023] conduzem seu estudo a partir do seguinte questionamento: esses modelos podem ser úteis como fontes externas para identificar possíveis lacunas em requisitos?. Logo, os autores, adotaram como abordagem o modelo de linguagem mascarada (MLM) do BERT para gerar previsões contextualizadas que preencham essas lacunas. Para simular incompletude, foi ocultado parte do conteúdo dos requisitos e avaliou-se a capacidade do BERT de prever termos no conteúdo oculto, mas ausentes no conteúdo disponibilizado. No primeiro momento o BERT permitiu gerar múltiplas previsões para cada máscara. Como contribuição, identificou-se o número ideal de previsões por máscara,

equilibrando a descoberta de lacunas com a redução de ruído nas previsões. A segunda contribuição foi a criação de um filtro baseado em aprendizado de máquina que refina as previsões do BERT, reduzindo ainda mais o ruído. Os autores testaram a solução em 40 especificações de requisitos do conjunto de dados PURE. Os resultados mostram que o BERT é altamente eficaz para identificar termos ausentes em requisitos e o filtro proposto reduz significativamente o ruído, tornando o BERT uma ferramenta promissora para melhorar a completude de requisitos.

No estudo de [Limaylla-Lunarejo et al. 2023], investigou-se a eficácia de diferentes algoritmos de Machine Learning (ML) na classificação automática de requisitos funcionais e não funcionais (FR/NFR) em espanhol. O estudo se concentra em dois modelos de linguagem pré-treinados, fastText e BETO, comparando seu desempenho com algoritmos tradicionais de ML e Deep Learning (DL). O estudo é motivado pelo fato de que a classificação manual de requisitos é uma tarefa crucial, mas muitas vezes complexa e demorada em projetos de software. Logo, abordagens automatizadas usando ML e técnicas de Processamento de Linguagem Natural (PNL) oferecem uma solução promissora para este desafio. O estudo concluiu que o BETO é o modelo mais promissor para a classificação automática de requisitos em espanhol, devido à sua alta capacidade de generalização. No entanto, o fastText se mostra uma alternativa viável quando o tempo de processamento é um fator crítico. Os autores recomendam pesquisas futuras para expandir os datasets em espanhol e investigar a generalização dos modelos em diferentes variantes do idioma.

A Engenharia de Requisitos (RE) é uma fase crítica no desenvolvimento de software, englobando a elicitação, análise, especificação e validação de requisitos. Apesar de sua importância, a RE continua sendo um processo desafiador devido às complexidades da comunicação, às incertezas nas etapas iniciais e ao suporte insuficiente de automação. Considerando que nos últimos anos, grandes modelos de linguagem (LLMs) têm demonstrado um potencial significativo em diversos domínios, incluindo processamento de linguagem natural, geração de código e compreensão de programas, [Arora et al. 2023] exploram o potencial dos LLMs na condução de processos de RE, visando melhorar a eficiência e a precisão das tarefas relacionadas aos requisitos. Foram propostas direções-chave e uma análise SWOT para pesquisa e desenvolvimento no uso de LLMs na RE, com foco em sua aplicação na elicitação, análise, especificação e validação de requisitos. Os resultados dos experimentos sugerem que algoritmos Shallow ML são eficazes na classificação de requisitos em espanhol, especialmente para requisitos não funcionais; modelos pré-treinados como o BETO demonstram maior capacidade de generalização, classificando com precisão requisitos de diferentes tipos de projetos e domínios; e o fastText, apesar de apresentar

menor generalização que o BETO, destaca-se pela sua velocidade de processamento.

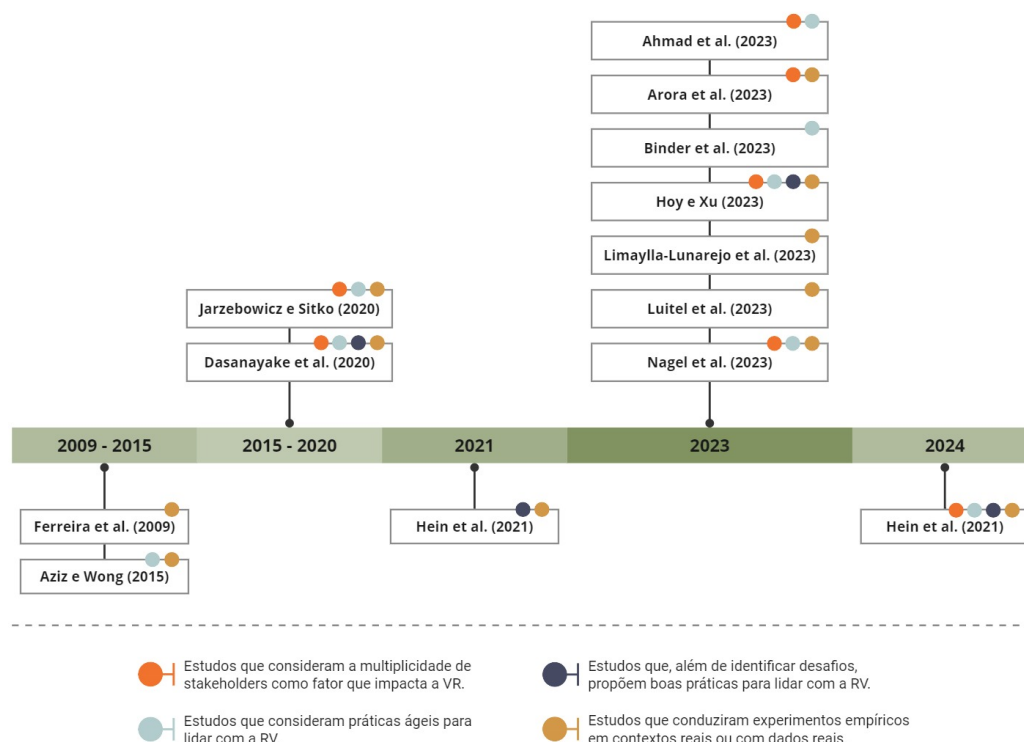
A Engenharia de Requisitos tradicional é amplamente consolidada no desenvolvimento de sistemas de software convencionais, porém, a criação de sistemas baseados em Inteligência Artificial (IA), onde o funcionamento interno é limitado ou opaco, apresenta desafios únicos para RE. Enquanto a literatura existente foca no uso da IA para gerenciar atividades de requisitos, há uma lacuna significativa na pesquisa voltada para Engenharia de Requisitos para IA (RE4AI). Assim, o estudo conduzido por [Ahmad et al. 2023] investigou abordagens atuais para especificar requisitos para sistemas de IA, analisando frameworks, metodologias, ferramentas e técnicas disponíveis, bem como desafios e limitações existentes. Por meio de um estudo de mapeamento sistemático, foram identificados 43 estudos primários que apontaram que as práticas de RE4AI atuais não são suficientemente adaptáveis para sistemas de IA, evidenciando a necessidade de novas ferramentas e métodos. Embora os estudos empíricos se concentrem majoritariamente em veículos autônomos e gestão de requisitos de dados, áreas como ética, confiança e explicabilidade ainda carecem de maior exploração.

3.2.2 Resultados e Discussões

Embora esse mapeamento sistemático tenha apontado para um número significativo de estudos, foram poucos os que apresentaram adesão à questão e objetivos da pesquisa. Ao todo, foram considerados 13 estudos, percorridos na seção anterior. Pode-se perceber diversidade nos métodos e abordagens apontados pelos autores para lidar com as questões da Volatilidade de Requisitos (VR) em ambientes de múltiplos stakeholders e de alta dinamicidade.

A primeira consideração a se fazer está nas interseções dos estudos encontrados em relação às temáticas abordadas. Algumas nuances importantes podem ser percebidas a partir da Figura 4. Embora os 13 estudos analisados tratem direta ou indiretamente sobre volatilidade de requisitos de software, somente 07 deles relacionam a multiplicidade de stakeholders a VR como um fator de alto impacto [Jarzębowicz e Sitko 2020, Dasanayake et al. 2019, Hein et al. 2021, Nagel et al. 2023, Hoy e Xu 2023, Arora et al. 2023, Ahmad et al. 2023]. Estes estudos, direta ou indiretamente, possibilitam inferir que diferentes stakeholders, com suas próprias necessidades e prioridades, podem contribuir para mudanças nos requisitos do projeto, impactando a elicitação, à priorização, bem como na manutenção.

Figura 4 – Estudos selecionados e abrangências temáticas



Fonte: Autoria própria (2023)

Ainda considerando a Figura 4, aponta-se que 8 estudos consideram práticas ágeis como abordagens diretas para mitigar os impactos gerados pela volatilidade de requisitos. Esse apontamento se deve à capacidade de adaptação a mudanças recorrentes, devido à definição de escopo e execução interativa dos processos. Dos estudos selecionados, 4 deles, apontam possíveis boas práticas ou meios para lidar com VR, seja em contextos específicos ou com a possibilidade de replicar em outros cenários, mas ainda sendo um desafio sua implementação em ambientes reais devido à complexidade técnica e à resistência organizacional.

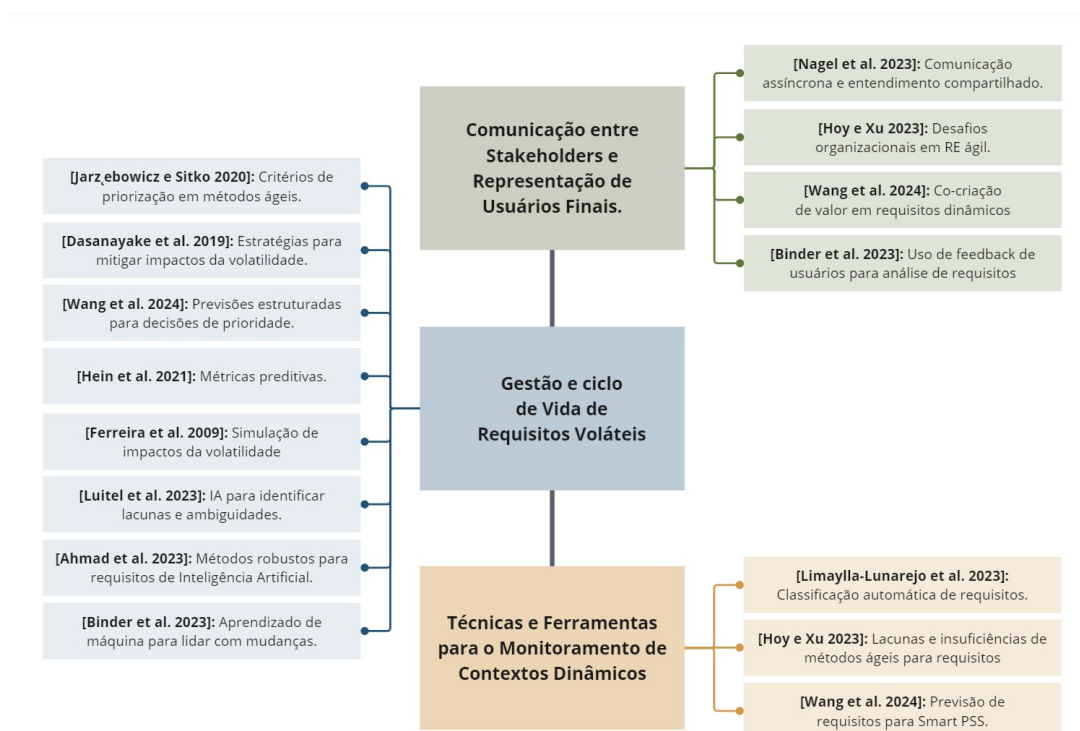
Dos 13 estudos analisados, somente 3 foram aderentes aos quatro aspectos analisados inicialmente. Logo, estes apontaram que a multiplicidade de stakeholders é um fator que impacta a VR, que consideram práticas ágeis para lidar com esses impactos e que, por fim, propõem boas práticas e/ou meios para mitigar os impactos. Os três estudos conduziram experimentos empíricos em contextos reais ou se utilizando de dados reais.

Considerando as áreas de foco de cada estudo, foi possível construir a Figura 5, que traz algumas relações entre os estudos, considerando os tópicos comuns, evidenciando algumas conexões e observações. Nessa análise, foram considerados os seguintes tópicos: **i.** Comunicação

entre Stakeholders e representação de Usuários Finais; **ii.** Gestão e ciclo de vida de Requisitos Voláteis; **iii.** Técnicas e Ferramentas para o Monitoramento de Contextos Dinâmicos.

A partir da análise da tabela, pode-se concluir que os desafios da Engenharia de Requisitos (ER) continuam sendo amplamente reconhecidos, mas as abordagens propostas nos estudos ainda apresentam lacunas específicas, especialmente no que diz respeito à integração de soluções tecnológicas com práticas humanas e colaborativas. Diversos tópicos, como gestão de requisitos voláteis, comunicação entre stakeholders e priorização de requisitos, têm recebido atenção significativa na literatura, evidenciando avanços na compreensão e mitigação de seus impactos. No entanto, há uma disparidade na distribuição das soluções: enquanto temas como ferramentas baseadas em *Machine Learning* (ML) e priorização ágil têm mostrado progresso, desafios como sub-representação de usuários finais, engajamento e monitoramento de contextos dinâmicos permanecem sub-explorados ou com soluções incipientes.

Figura 5 – Síntese dos estudos selecionados e tópicos comuns.



Fonte: Autoria própria (2023)

Outro ponto importante é a necessidade de maior equilíbrio entre soluções automatizadas e abordagens que promovam alinhamento humano, como a proposta de [Nagel et al. 2023] para comunicação assíncrona e os conceitos colaborativos de [Hoy e Xu 2023]. Esses estudos

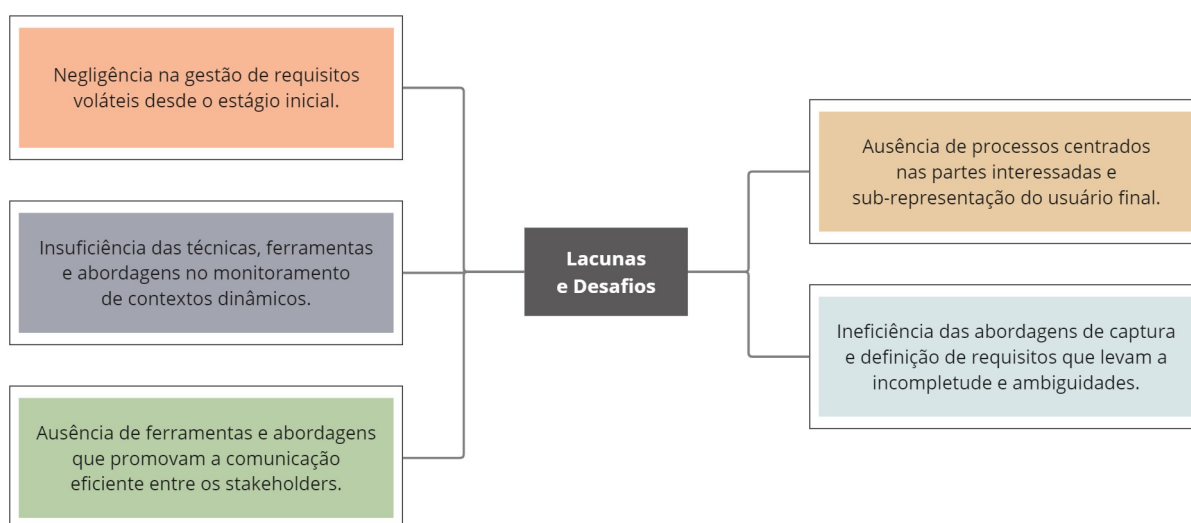
sugerem que, embora as ferramentas sejam fundamentais, o engajamento, a confiança e a representatividade dos stakeholders ainda precisam ser fortalecidos para alcançar um entendimento compartilhado e resultados efetivos. A tabela destaca uma lacuna significativa em soluções práticas que abordem esses aspectos humanos de forma integrada com tecnologias emergentes, apontando para a necessidade de novas pesquisas que combinem automação e interação colaborativa. Isso reflete a oportunidade de desenvolver abordagens mais equilibradas e holísticas para superar os desafios da ER.

Considerando as discussões e percepções apresentadas ao longo dessa seção, a seção subsequente apresenta as considerações finais e evidencia os principais pontos destacados nesta pesquisa, identificando lacunas e desafios relevantes.

3.2.3 Desafios e Lacunas Identificados

De modo geral, a literatura atual sobre volatilidade de requisitos (VR) em ambientes dinâmicos e com múltiplos stakeholders demonstra um esforço significativo para compreender e mitigar os desafios associados às constantes mudanças nas demandas de software. Esses estudos demonstram avanços em cada um dos tópicos mencionados, mas também deixam claro que alguns desafios persistem, como demonstrado na Figura 6. Áreas como volatilidade, ferramentas insuficientes e sub-representação de usuários ainda precisam de maior foco. Além disso, há uma tendência crescente de usar IA para abordar questões como priorização, previsão de mudanças e monitoramento de contextos dinâmicos.

Figura 6 – Desafios e Lacunas Encontrados



Fonte: Autoria própria (2023)

Essas lacunas são apontadas considerando a importância de integrar metodologias ágeis e frameworks híbridos que combinem estruturas tradicionais com a flexibilidade das práticas ágeis. Embora haja incidências alinhadas a essa prática, ainda é necessário abordagens que busquem equilibrar a necessidade de estrutura com a capacidade de adaptação às mudanças, especialmente em contextos em que os requisitos estão em constante evolução devido a fatores externos, como mudanças organizacionais, alterações nas legislações, e novas necessidades dos clientes. Além disso, embora haja uma essencialidade em práticas como o desenvolvimento iterativo, feedback contínuo e priorização flexível, ainda não há uma aplicação ampla delas e dos conceitos já pré-estabelecidos na Engenharia de Software. Logo, não há mitigação dos impactos da dinamicidade do domínio que levariam à responsividade das equipes diante da complexidade dos projetos e da multiplicidade de interesses dos stakeholders.

Outro aspecto relevante discutido nos estudos, mas não aprofundado no campo da prática, são os impactos da VR em elementos críticos, como a arquitetura de software e o planejamento de projetos. A ausência de ferramentas, técnicas e abordagens de monitoramento de contextos dinâmicos é um desafio, pois ainda algumas iniciativas ainda se encontram em estágio inicial. Nesse sentido, é necessário elucidar soluções práticas, como frameworks para avaliar e gerenciar riscos arquiteturais e simuladores de processos que permitem análises preditivas, contribuindo para a antecipação de impactos causados por mudanças nos requisitos. Do outro lado, é necessário também, uma combinação de estratégias adaptativas e colaboração contínua com as partes interessadas desde as etapas iniciais do desenvolvimento, pois embora haja bastantes iniciativas, ainda são necessários processos centrados nas partes interessadas, sem sub-representação de quaisquer partes, inclusive do usuário final.

Outro fator importante é que, embora haja uma diversidade de técnicas e abordagens de captura e definição de requisitos, ainda há, seja por parte do escopo das próprias técnicas que não foram pensadas para ambientes dinâmicos, ou seja, por sua condução incorreta, ineficiências que levam a incompletude e ambiguidades de requisitos, o que é um acréscimo de complexidade além da própria volatilidade. Nesse cenário, embora literatura enfatize a necessidade de abordagens holísticas que integrem documentação robusta, canais de comunicação claros e revisões regulares, ainda não há uma adoção satisfatória, mesmo que a participação efetiva dos stakeholders e a elicitação precisa de requisitos é um caminho plausível na redução de incertezas e melhora da estabilidade dos projetos.

3.3 CONSIDERAÇÕES FINAIS DO CAPÍTULO

Este capítulo buscou analisar como a volatilidade de requisitos tem sido investigada considerando ambientes de aplicação dinâmicos e dotados de múltiplos stakeholders. De modo geral, os estudos encontrados têm apontamentos importantes acerca dos impactos e desafios, e percebe-se que há diversidade nos métodos e abordagens apontados pelos autores para lidar com essas questões, porém há algumas preocupações específicas quanto a gestão e ciclo de vida de requisitos voláteis, considerando que há ausência significativa de técnicas e Ferramentas para o monitoramento de contextos dinâmicos. Além disso, a comunicação entre Stakeholders e representação de usuários finais no ciclo de vida de aplicação que atuam em ambientes caracterizados como dinâmicos e com múltiplos stakeholders ainda é falha.

Contudo, os *insights* obtidos com esse levantamento no estado da arte apontam para a urgência em pensar o ciclo de vida do software dinâmico para melhor lidar com a volatilidade de requisitos, levando em conta as dificuldades de gerir seus impactos, bem como de mitigá-los. Além disso, a necessidade de aderir a abordagens centradas nas partes interessadas, sem sub-representação inclusive dos usuários finais, é um fator essencial para que a comunicação e engajamento sejam tratados como cerne nos projetos de desenvolvimento. Com isso, pretende-se realizar um estudo de caso de um software desenvolvido e que atue em ambiente dinâmico e com múltiplos stakeholders. O estudo vai investigar todo o ciclo de vida para compreender as complexidades por trás da solução computacional, considerando as diversas perspectivas das partes interessadas, para propor abordagens e boas práticas replicáveis para condução em outros cenários parecidos.

4 INVESTIGANDO UM SOFTWARE GOVERNAMENTAL DESENVOLVIDO EM AMBIENTE ALTAMENTE DINÂMICO E COM MÚLTIPLOS STAKEHOLDERS

Esta seção dedica-se à avaliação aprofundada de um sistema que opera em ambiente real e que possui características de um ambiente dinâmico e que sofre a incidência de muitos stakeholders, o que aumenta o grau de complexidade devido à diversidade de interesses e contribuições das partes envolvidas. Permeando a volatilidade de requisitos nesses ambientes, o estudo permeará por abordagens centradas nas partes interessadas. Para isso, o PNLD Avaliação se torna objeto de estudo deste capítulo, e é um sistema de apoio ao processo de avaliação pedagógica do Programa Nacional do Livro Didático (PNLD).

Portanto, esta seção seguirá o seguinte roteiro: na subseção 4.1, são apresentados os aspectos éticos, na 4.2 está percorrido o percurso metodológico utilizado no estado da prática para a avaliação do sistema; do 4.3 ao 4.5, dedica-se aos três estágios da condução prática, apresentando os instrumentos de coleta e análise utilizados em cada estágio, os resultados obtidos, *insights* e discussões inferidas; por fim, no 4.6, há a consolidação dos resultados obtidos em cada estágio da condução prática e as considerações finais do capítulo.

4.1 ASPECTOS ÉTICOS

Na condução desta investigação, em que há o envolvimento direto de seres humanos, como professores-avaliadores, profissionais de TI e demais partes interessadas, é imperativo abordar e respeitar princípios éticos em pesquisa. Isso se deve, para garantir a confiabilidade dos objetos levantados e também garantir a integridade e o bem-estar dos participantes. Assim, o projeto desta pesquisa foi submetido pela primeira vez, em 12 de Abril de 2023, ao Comitê de ética em Pesquisa (CEP) da Universidade Federal de Alagoas, que emitiu parecer solicitando alterações e adição de informação. Em 24 de Maio de 2023, foi enviada a nova versão com as alterações solicitadas e somente em 13 de junho de 2023, o Comitê de Ética emitiu parecer positivo, liberando a execução de pesquisa sob o **CAAE: 69059623.1.0000.5013**, conforme Anexo A.

Todos os contatos com os participantes se deram com a utilização do Termo de Consentimento Livre e Esclarecido (TCLE), com uma condução centrada na compreensão de todos os participantes, dos objetivos da pesquisa, os procedimentos envolvidos e quaisquer potenciais impactos. Foram apresentados os métodos de tratamento e a confidencialidade dos dados coletados, como meio de proteger a identidade dos participantes e resguardar informações sensíveis que pudessem emergir durante as entrevistas ou questionários.

4.2 PERCURSO METODOLÓGICO

O percurso metodológico desta condução prática compreende três estágios distintos. O Primeiro Estágio se preocupa com a identificação do cenário estudado, considerando sua aderência à temática aqui delimitada. Essa busca inicial acontece em artigos, páginas e portais governamentais e demais registros históricos disponíveis, justamente para compreender a natureza do software investigado, seus objetivos, campos de atuação e alcance. Ainda nesse primeiro momento, há a execução de uma inspeção nos artefatos [O'Regan e O'Regan 2018] de software gerados no ciclo de vida de desenvolvimento da solução computacional. O objetivo central dessa inspeção está em entender como o envolvimento de múltiplos stakeholders e o ambiente altamente dinâmico das políticas públicas influenciam a volatilidade dos requisitos de software e a eficiência do processo de desenvolvimento de software. Para isso, são consideradas desde histórias de usuários, documentação de requisitos funcionais e não funcionais e feedbacks registrados na rede de suporte. Com esses documentos selecionados e criteriosamente analisados, foi aplicada uma análise estatística comparativa em momentos diferentes do projeto, para compreender o comportamento geral do software em relação ao usuário e suas documentações nas duas fases identificadas.

No Segundo Estágio, acontece um processo de escuta para com os profissionais de TI e demais stakeholders, a partir de entrevistas semiestruturadas [Lazar et al. 2017] e escuta de reuniões gerais de avaliação e relatoria do projeto, nesse último momento, são considerados não só os profissionais de TI, mas Instituições públicas envolvidas, usuários finais e outras instituições externas envolvidas no projeto. Para analisar os dados capturados, após o processo de coleta e transcrição, foi aplicada uma análise temática semântica, com foco nos relatos e suposições que sustentam suas experiências e, por fim, informando o conteúdo semântico dos dados obtidos.

Por fim, no terceiro estágio, utilizou-se de uma análise de usabilidade e captura de percepções dos usuários finais da plataforma computacional investigada. Para isso, foram utilizados dois métodos condicionados em um formulário distribuído digitalmente. O primeiro foi a aplicação da Escala SUS, com foco em avaliar três aspectos da interface do software estudado: a efetividade, que está para o sucesso no uso do sistema, a eficiência, que diz do esforço empregado para o uso do sistema e a satisfação que parte da experiência do usuário ao usar o sistema [Brooke 2013]. Além da própria escala, os usuários finais foram convidados a responder a algumas questões abertas, para complementar e ampliar seus relatos, enquanto usuários finais da plataforma go-

vernamental. Os dados obtidos foram tratados e lidos a partir de uma análise temática dedutiva, que considerou os aspectos de qualidade que fundamentam a compressão de usabilidade. Neste caso, foram investigados a aprendizagem, a memorabilidade, eficiência do sistema, erros e inconsistências, satisfação de uso e acessibilidade [Nielsen 1994, Rubin e Chisnell 2008].

De modo geral, em cada estágio que se segue, será detalhado cuidadosamente cada passo da metodologia adotada. Como foi utilizada uma abordagem metodológica mista, é essencial definir e esclarecer cada etapa de sua execução, respeitando os momentos específicos de cada aplicação, a fim de garantir uma melhor compreensão.

4.3 **ESTÁGIO 01: IDENTIFICAÇÃO DE CENÁRIO E INSPEÇÃO DE ARTEFATOS DE SOFTWARE**

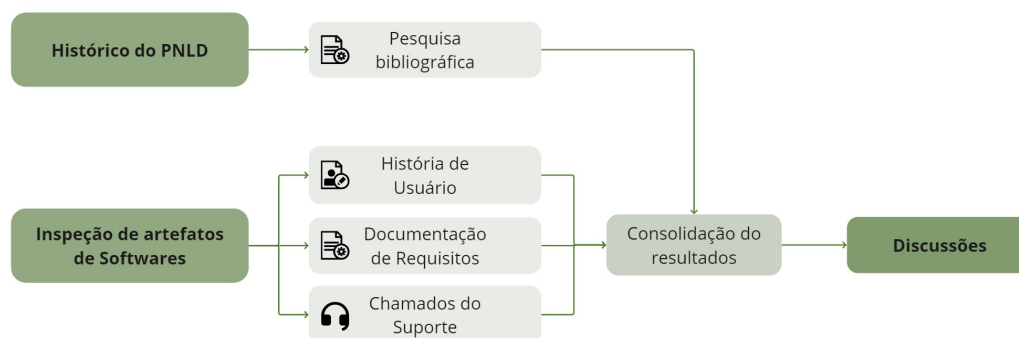
Esta subseção será dividida em duas partes, a primeira trará um breve histórico sobre o que é o PNLD e se fará uma descrição do ambiente acerca da estrutura à qual está inserido, apresentando o fluxo de funcionamento e os atores que estão envolvidos no processo. Na segunda, será mapeado e descrito o minimundo a ser investigado, considerando o contexto de uso e a análise detalhada dos artefatos envolvidos no desenvolvimento.

4.3.1 Instrumento de Coleta e Análise

Foi adotada uma abordagem de estudo de caso para investigar a volatilidade de requisitos e o impacto da participação de stakeholders no processo de desenvolvimento de software para o Programa Nacional do Livro Didático (PNLD). Estudos de caso são adequados para explorar fenômenos complexos em contextos do mundo real, proporcionando uma compreensão aprofundada das questões envolvidas [Larman e Basili 2003]. Esse método mostrou-se apropriado para a pesquisa, pois permite analisar a dinâmica da volatilidade de requisitos e a influência de stakeholders em um ambiente específico e prático.

A Figura 7 ilustra a metodologia utilizada para investigar o ciclo de desenvolvimento de um aplicativo governamental com requisitos frequentemente mutáveis e com múltiplos stakeholders. Na etapa inicial, foi considerado um levantamento bibliográfico e documental interno para compreender o histórico do PNLD. Dessa forma, foram utilizados artigos, páginas governamentais oficiais, projetos de leis e legislações vigentes. Em seguida, artefatos de software foram coletados e analisados para avaliar como a mudança desses requisitos afetaram o gerenciamento do projeto.

Figura 7 – Etapas do PNLD



A coleta de dados envolveu a obtenção de diversos artefatos do projeto e abrangeu dois períodos distintos: a fase inicial, com quatro stakeholders e uma fase posterior, com um modelo simplificado após a redefinição de papéis. Os artefatos coletados foram:

- **Histórias de usuário:** Coletamos histórias de usuário para avaliar a natureza e a frequência das mudanças nos requisitos. As histórias de usuário fornecem insights sobre as funcionalidades específicas solicitadas pelos stakeholders e como essas solicitações evoluíram.
- **Documentação de requisitos:** Inclui registros detalhados de todos os requisitos especificados ao longo do projeto. A análise desses documentos nos ajuda a entender os requisitos iniciais e as modificações subsequentes.
- **Chamados de suporte:** Os chamados de suporte foram analisados para identificar problemas relatados pelos usuários, tempos de resolução e quaisquer problemas recorrentes. Esses dados ajudam a avaliar a estabilidade do sistema e sua capacidade de responder às necessidades dos usuários.

Para analisar esses dados, optou-se por uma análise comparativa abrangendo os dois períodos do projeto. Dessa forma, consideraram-se os seguintes aspectos:

- **Análise de volatilidade de requisitos:** Calculou-se a taxa de mudanças nos requisitos comparando o número de modificações, adições e exclusões de requisitos durante ambos os períodos. Essa métrica ajuda a quantificar a volatilidade e identificar padrões ou gatilhos para as mudanças.
- **Análise do impacto dos stakeholders:** Examinaram-se os papéis e as interações dos stakeholders durante as duas fases. Essa análise envolveu compreender como a diversidade

e a rotatividade de stakeholders impactaram os processos de tomada de decisão e a estabilidade dos requisitos.

- **Análise de eficiência:** Foram mensuradas as melhorias na eficiência comparando o número de histórias de usuário concluídas, chamados de suporte resolvidos e o tempo médio para implementar mudanças. Essa comparação ajuda a determinar se a simplificação dos stakeholders resultou em um processo de desenvolvimento mais ágil e responsivo.

Como percebido, utilizaram-se métodos mistos, uma abordagem que combina dados quantitativos e qualitativos para analisar de forma abrangente a dinâmica do projeto. Os dados quantitativos fornecem evidências mensuráveis sobre a volatilidade de requisitos e eficiência, enquanto os dados qualitativos oferecem compreensão contextual e insights sobre as interações entre stakeholders e práticas de gestão do projeto [Creswell e Clark 2017].

4.3.2 O Programa Nacional do Livro e do Material Didático

O PNLD Avaliação é uma instância dentro do Programa Nacional do Livro e do Material Didático (PNLD), que é uma política instituída em 1937¹, cujo propósito é beneficiar estudantes e educadores das escolas públicas de ensino fundamental e médio, bem como organizações sem fins lucrativos, por meio da distribuição de diversos recursos didáticos e literários de apoio educacional [Batista 2001], submetidos a um criterioso processo de avaliação para assegurar a adequação aos padrões de qualidade e às exigências pedagógicas do currículo nacional de educação. Com o passar de 85 anos, o Programa Nacional do Livro Didático (PNLD) se estabeleceu como uma política pública duradoura no Brasil e uma das maiores do mundo [Pinheiro et al. 2021]. Seus impactos na educação brasileira são evidentes, como demonstrado pelos impressionantes números de aquisição de livros. Apenas em 2021, foram adquiridos aproximadamente 172 milhões de livros, beneficiando 30 milhões de estudantes em 140 mil unidades de ensino espalhadas por todo o país [Educação 2022].

O PNLD tem como objetivo a análise e distribuição regular e gratuita de materiais didáticos, pedagógicos, literários e outros recursos de apoio ao ensino para as escolas públicas de educação básica. Isso inclui tanto as escolas das redes federal, estaduais, municipais e distrital, quanto as instituições comunitárias, confessionais ou filantrópicas sem fins lucrativos que possuam convênio com o governo [MEC 2023]. Em linhas gerais, O PNLD possui como objetivos específicos o aprimoramento do processo de ensino e aprendizagem nas escolas

¹ O nome PNLD foi criado somente em 1985 por meio do Decreto n.º 91.542, de 19 de Agosto de 1985.

públicas de educação básica com a consequente melhoria da qualidade da educação; a garantia do padrão de qualidade do material de apoio à prática educativa utilizado nas escolas públicas de educação básica; a democratização do acesso às fontes de informação e cultura; o fomento à leitura e o estímulo à atitude investigativa dos estudantes; o apoio a atualização, autonomia e o desenvolvimento profissional do professor; e a implementação da Base Nacional Comum Curricular.

Regularmente, o MEC em cooperação com o FNDE, publica editais referentes aos processos de aquisição de materiais didáticos para atendimento das etapas de educação básica. Tais editais são publicados de forma alternada para atender, em ciclos diferentes, as etapas e os segmentos de ensino de educação infantil, anos iniciais do ensino fundamental, anos finais do ensino fundamental e ensino médio. Para cada edital do PNLD, percorrem-se as seguintes etapas [AbreLivros 2020], demonstradas na Figura 8 e percorridas em seguida:

Figura 8 – Etapas do PNLD



Fonte: Elaborada pelo Autor (2024)

1. **Publicação do Edital:** Ação centralizada no MEC e FNDE, que compreende a publicação do edital com definição do objeto, prazos e especificações técnicas e pedagógicas dos livros e materiais didáticos e literários.
2. **Inscrição das obras:** Ação descentralizada entre autores e editoras, que envolve o processo de escrita e produção das obras que atendem à legislação educacional, às orientações do edital e às necessidades das redes de ensino.
3. **Avaliação Pedagógica:** Ação centralizada no MEC que nomeia uma comissão técnica formada por professores e especialistas que buscam a garantia da qualidade do material e a conformidade com os objetivos da legislação brasileira de educação, respeitando

o pluralismo de ideias e concepções pedagógicas, as diversidades sociais, culturais e regionais.

4. **Seleção das obras:** Ação descentralizada que envolve diretores, coordenadores e professores, que têm acesso às obras habilitadas por meio do Guia do PNLD, e entre eles optam por aquelas que melhor atendam ao projeto pedagógico da escola.
5. **Negociação:** Ação centralizada no FNDE que compreende a definição de quantidade de obras a ser adquirida, é feita a compilação das informações decorrentes da escolha, o processamento dos dados e estabelecidos os parâmetros dos preços mínimos e máximos de cada obra.
6. **Produção gráfica:** Ação descentralizada e que envolve as editoras, sendo uma etapa que compreende a produção em massa de todas as obras adquiridas, assegurando a qualidade gráfica e a adequação às especificações técnicas do edital.
7. **Distribuição:** Ação centralizada no FNDE e Correios, onde há a distribuição de todas as obras para todas as escolas que aderiram ao PNLD antes do início do ano letivo.
8. **Uso do material:** Ação descentralizada que envolve alunos e professores, no uso inclusivo e democrático por toda comunidade escolar, dos materiais didáticos e literários distribuídos.

Porém, o software a ser analisado está restrito à etapa 3, que trata da Avaliação Pedagógica e conta com o intermédio do sistema de software PNLD Avaliação, para a condução dos diversos processos que constituem a avaliação pedagógica dos livros e materiais que apoiam a educação brasileira. Assim, na próxima subseção, o sistema será apresentado com mais detalhes.

4.3.2.1 PNLD Avaliação

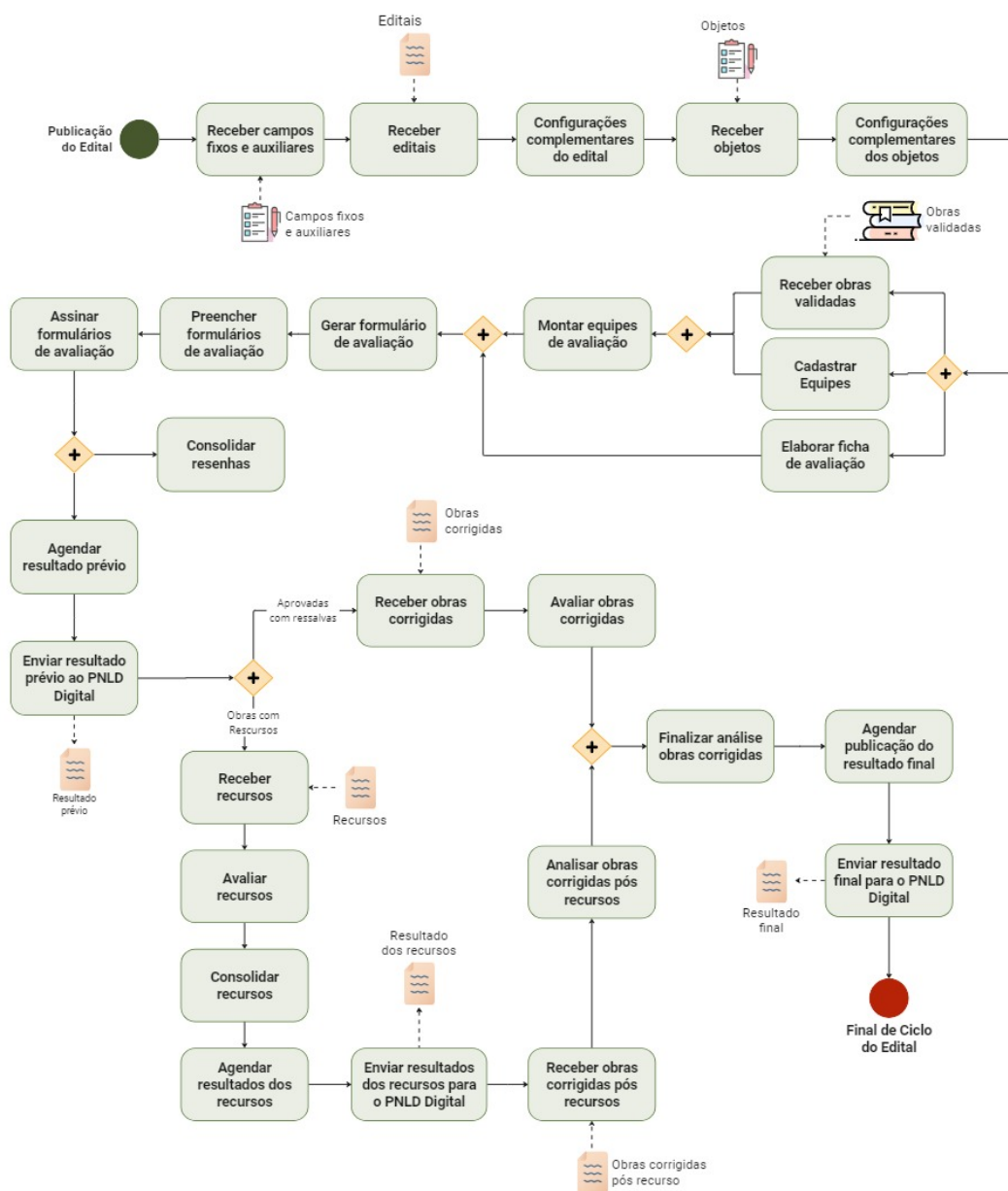
O PNLD Avaliação é parte de um processo contínuo que tem buscado a sistematização dos processos que circundam projetos e políticas públicas, tais quais o PNLD, alinhado a estratégias ordenadas para a Transformação Digital (TD). No contexto do PNLD, a estratégia em andamento para integralização de todos os seus processos sob a ótica da sistematização, se dá pela busca do melhor gerenciamento e viabilização do processo de escolha e aquisição de materiais didáticos e literários que aprimora a educação em todo território brasileiro [RNP 2021].

O PNLD Avaliação compreende no fluxo de avaliação pedagógica diversas equipes que desempenham papéis distintos e que são fundamentais, cada uma com um nível de colaboração.

Estas equipes podem ser vistas como usuários finais. Esses usuários podem ser qualquer integrante de algum dos seguintes grupos: Comissão Técnica, Coordenação Pedagógica, Assessores Pedagógicos, Coordenadores Adjuntos e, principalmente, as Pessoas Avaliadoras. São as pessoas avaliadoras responsáveis diretas por analisar os materiais didáticos submetidos pelas editoras e avaliar sua qualidade, adequação pedagógica e relevância para o ensino. As pessoas avaliadoras aplicam os critérios estabelecidos pela Comissão Técnica, que incluem aspectos como conteúdo, metodologia, linguagem, recursos complementares e alinhamento com as diretrizes curriculares. Esses profissionais são integrantes de um Banco de Avaliadores, um cadastro nacional que reúne especialistas habilitados, com experiência e expertise, que possibilitam uma avaliação criteriosa e fundamentada, contribuindo para a seleção dos melhores materiais didáticos. O Banco de Avaliadores conta com representantes de todos os estados do país, considerando a representatividade das diferentes regiões geográficas do Brasil e suas pluralidades [Silva et al. 2024, Educação 2022].

Todas essas partes interessadas atuam em conjunto nas diversas partes de um processo extremamente longo e complexo, que envolve o cadastro das equipes avaliadoras, a criação da ficha de avaliação e o recebimento das obras inscritas e validadas para avaliação. Em seguida, os recursos educacionais são distribuídos às pessoas avaliadoras selecionadas, as quais avaliam cada recurso educacional a partir dos critérios estabelecidos na ficha. Após a conclusão das análises pelos avaliadores, os resultados são registrados no PNLD Avaliação e, logo depois, encaminhados de volta ao PNLD Digital, juntamente com o parecer para as editoras. A Figura 9 traz um fluxograma com aspectos gerais do processo de avaliação pedagógica.

Figura 9 – Fluxograma do PNLD Avaliação sob ótica do processo de avaliação pedagógica



Fonte: Elaborada pelo Autor (2024)

Analisando o fluxograma apresentado na Figura 9, é possível então compreender onde se encontra um dos principais motivadores na compreensão do PNLD Avaliação enquanto sistema dinâmico. O início do fluxo se dá a partir da publicação de um edital, e, como já descrito, compreende um documento que especifica regras que ditarão algumas funcionalidades do sistema, campos que já não devem compor as interfaces do software e novos campos. Essas mudanças se dão a partir principalmente da legislação educacional que está em vigência no país, e que é um componente volátil, uma demanda que acompanha os interesses públicos e

políticos educacionais do Brasil. Essas demandas são expressas no edital pela Coordenação Geral de Materiais Didáticos (CGMD), ligada ao Ministério da Educação (MEC). Logo, as primeiras cinco atividades do fluxo, "Receber campos fixos e auxiliares", "Receber editais", "Configurações complementares do edital", "Receber objetos" e "Configuração complementares do objeto", definem campos que serão adicionados ou retirados das páginas que trazem a ficha e formulários de avaliação dos livros e materiais didáticos. Ou seja, parte dos campos do sistema são alterados a cada edital lançado e suas respectivas demandas [Silva et al. 2024].

O desenvolvimento desse sistema perpassa diversas mudanças no escopo do projeto, seja pelas mudanças frequentes dos agentes públicos que fazem a política pública, seja pelos novos parceiros que integram o PNLD Avaliação. Por tanto, é extremamente importante entender algumas nuances do processo de desenvolvimento, como as diversas partes interessadas interferem nesse processo, quais seus papéis e os níveis de relação. Dessa forma, a seção a seguir traz um panorama da transformação digital que alicerça o PNLD Avaliação atual.

4.3.2.2 Panorama da Transformação Digital do PNLD

Um dos passos essenciais para compreender o objeto de estudo desta dissertação é mapear os stakeholders e identificar os diferentes atores envolvidos no processo de desenvolvimento. Compreender seus papéis e influências diretas no ciclo de vida do software é crucial para uma análise abrangente dos aspectos que envolvem a satisfação do usuário e o atendimento às necessidades dos usuários finais. Adicionalmente, é necessário considerar como a transformação digital afeta a estrutura do software, trazendo novas preocupações relacionadas à sua aplicação e interação com os usuários. Nesse quesito, são investigados os processos que permeiam a transformação digital da avaliação pedagógica do PNLD até a atual solução computacional, o PNLD Avaliação.

Um aspecto fundamental do processo de avaliação pedagógica do PNLD é a natureza dinâmica dos editais públicos para a aquisição de livros didáticos em cada ciclo do programa. Esses editais são essenciais para garantir transparência e equidade na seleção e aquisição de materiais educacionais, assegurando que os materiais distribuídos atendam aos critérios pedagógicos estabelecidos e contribuam para a qualidade da educação nas escolas públicas. A elaboração desses editais inclui definição de prazos, etapas de avaliação dos materiais, critérios de seleção, requisitos técnicos e pedagógicos, e procedimentos administrativos para a participação das editoras interessadas na submissão de seus livros. No entanto, a relevância desses editais também acarreta consequências específicas, com atualizações e ajustes ao longo do tempo,

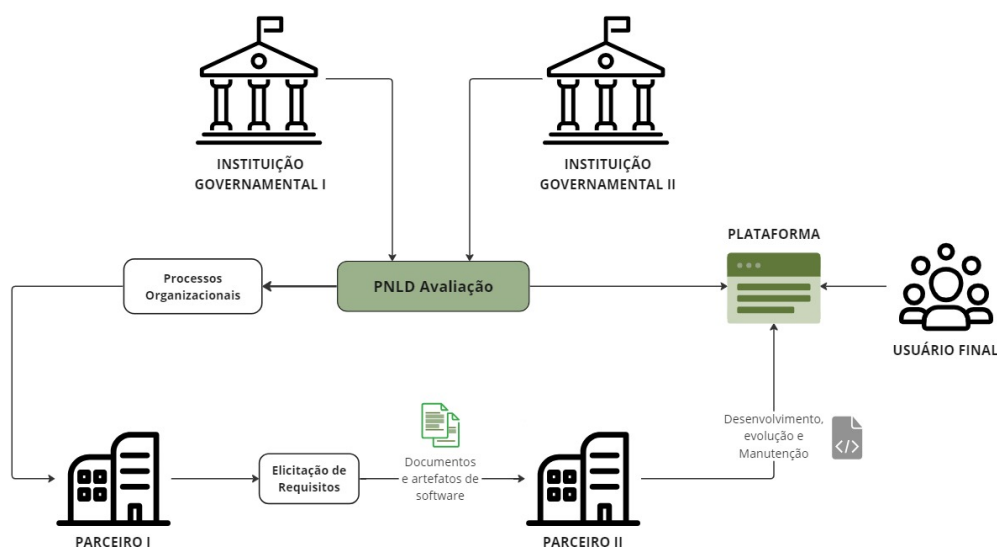
conforme necessidades e melhorias identificadas pelas instituições governamentais e pela equipe responsável pelo PNLD. Assim, cada edital pode apresentar características e composições diferentes, exigindo que o software que gerencia esse processo de avaliação seja capaz de se adaptar a tais mudanças.

Além disso, a interação com aplicativos externos e o envolvimento de diversos stakeholders tornam o processo de avaliação dos materiais submetidos aos editais públicos mais complexo. Cada processo de avaliação exige demandas específicas e configurações particulares, conforme especificado no edital. Assim, o desenvolvimento do aplicativo de software chamado PNLD Avaliação, do ponto de vista da engenharia de software, envolve:

- Uma arquitetura complexa.
- Centenas de requisitos.
- Comunicação com múltiplas partes interessadas.
- Gestão das expectativas dos stakeholders.
- Atendimento às suas diversas necessidades.

Assim, o PNLD Avaliação ocorre em uma estrutura complexa e por meio de uma execução prolongada que envolve diferentes tipos de agentes em diversas instâncias governamentais, colaboração de instituições de ensino superior, outros parceiros e usuários finais. Essa amplitude no escopo do projeto envolve um processo colaborativo que vai desde a definição das regras de negócio até o uso da plataforma por diferentes stakeholders, conforme ilustrado na Figura 10.

Figura 10 – Mapa de stakeholders e correlações



Fonte: Elaborada pelo Autor (2024)

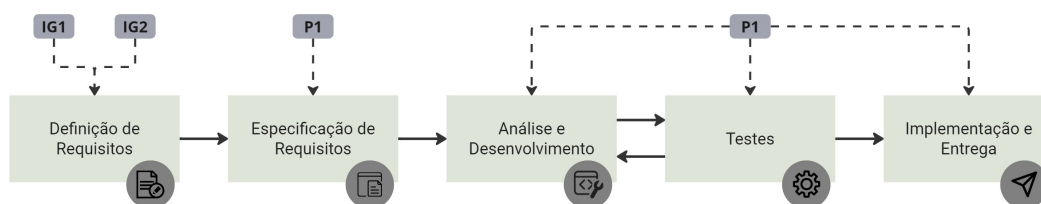
Conforme a mesma figura, o processo de desenvolvimento começa com as instituições (MEC e FNDE) formulando e implementando políticas públicas. A IG1 (FNDE), em colaboração com a IG2 (MEC), responsável pela etapa de avaliação do programa, trabalha para definir e documentar os processos organizacionais e as regras de negócio. Esses documentos são compartilhados com o P1, onde analistas de requisitos se familiarizam com o domínio do problema e utilizam esse conhecimento para escrever histórias de usuário, documentá-las e validá-las com especialistas da IG2.

Quando o P2 recebe a documentação, artefatos de software são criados, como modelos arquiteturais, definições de requisitos funcionais e protótipos de interfaces de usuário. Nesse caso, o P2 é responsável pelo desenvolvimento, evolução e manutenção da plataforma. Uma vez pronta, a plataforma é implantada em ambiente de produção, onde o usuário final, representando as pessoas responsáveis pela avaliação dos materiais submetidos e outros stakeholders, realiza ações para conduzir as atividades de avaliação pedagógica de livros e materiais educacionais.

De modo geral, esse processo de desenvolvimento se assemelha ao modelo em cascata *Waterfall*, onde as atividades são executadas sequencialmente e só avançam para a próxima fase após a conclusão da etapa anterior. Nesse modelo, o processo de desenvolvimento ocorre de forma linear, sem inclusão de iterações, iniciando com a comunicação para levantamento de requisitos, seguida pela análise (incluindo atividades como planejamento e modelagem),

construção ou desenvolvimento desses requisitos, validação dos artefatos entregues e, finalmente, entrega do software. Portanto, o usuário final só pode dar feedback quando o desenvolvimento das funcionalidades for considerado completo e o software já estiver entregue. A Figura 11 apresenta um resumo diagramático desse processo.

Figura 11 – Representação do ciclo de vida do PNLD-Avaliação.



Fonte: Elaborada pelo Autor (2024)

Observando as duas figuras, 10 e 11, um aspecto interessante está entre as fases de elicitação e validação de requisitos, onde se evidencia a falta de interação com os usuários finais, uma ação que ocorreu exclusivamente entre P1 e IG2. Embora os especialistas da IG2 compreendam todo o processo, a experiência subjetiva e as preocupações específicas de outros atores, especialmente os usuários finais, não foram consideradas desde o início. A complexidade e a amplitude do projeto não foram levadas em conta, e o projeto, em sua operação, lida com perfis variados de stakeholders, que desempenham papéis distintos, utilizam diferentes funcionalidades no sistema de software e possuem diferentes formações e expectativas em relação ao uso da tecnologia.

De forma semelhante, a lacuna entre o P2, responsável pelo desenvolvimento, e o P1, responsável pela elicitação de requisitos, é notável. Nessa etapa da concepção da plataforma, é necessária a presença de um representante da equipe de desenvolvimento ou dos usuários finais. Essa distância entre as partes resultou em uma equipe técnica menos envolvida na compreensão prévia do domínio, afetando a eficiência com que a documentação pôde ser interpretada e compreendida. Consequentemente, essa situação gerou cenários que podem impactar a percepção dos usuários finais em relação ao sistema de software.

Sobre a natureza flexível do processo de avaliação, os processos relacionados ao programa podem sofrer alterações significativas com base nas definições do edital. Esses editais são frequentemente publicados e, a partir deles, todas as especificações precisam ser constantemente revisadas e adaptadas a novas necessidades emergentes. Todas essas mudanças exigem comunica-

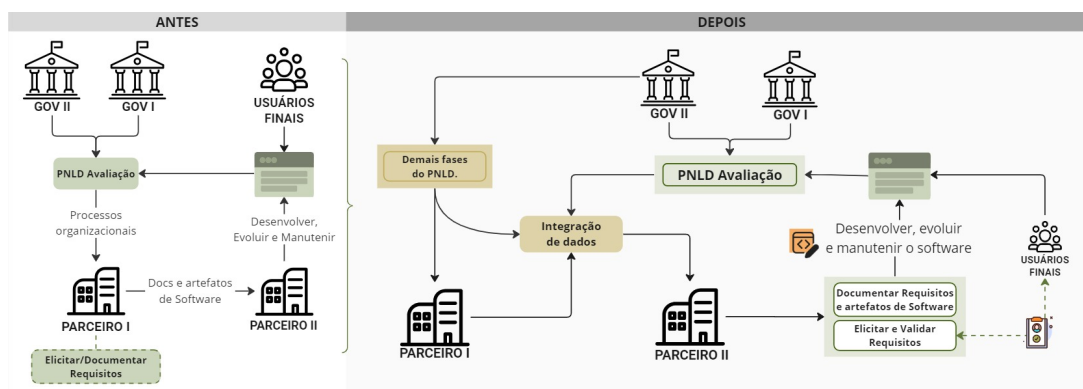
ção contínua entre stakeholders e outras equipes envolvidas no PNLD Avaliação. No entanto, ao usar o sistema de software, na prática, foi possível perceber discrepâncias entre o entendimento das necessidades dos usuários finais e as especificações e compreensões das equipes de requisitos e desenvolvimento.

4.3.2.3 Redefinição do fluxo de desenvolvimento da solução computacional de apoio a avaliação pedagógica do PNLD

A transformação digital no ciclo de vida do PNLD, enquanto ação contínua que sofre readequações frequentes, aponta muitos desafios e demandas recorrentes em seu fluxo. A partir dos aspectos já delineados sobre a formatação do projeto de desenvolvimento do sistema de apoio à avaliação pedagógica, algumas ações foram implementadas para sua melhoria.

A primeira ação compreende a otimização do processo de desenvolvimento, evolução e manutenção da solução computacional, a partir da redefinição do fluxo colaborativo. Essa redefinição do fluxo, conforme Figura 12, considerou a complexidade do contexto sob a qual a avaliação pedagógica do PNLD funciona e ainda, se apropriou da amplitude do corpo do projeto, em seu cerne colaborativo, que vai desde a compreensão da política pública, percorre o ciclo de vida de desenvolvimento de software, até o uso da plataforma pelos diversos interessados. Como já explicitado, inicialmente o fluxo compreendia a colaboração e a integração entre dois entes governamentais caracterizados como GOV I e GOV II, e dois parceiros externos, o Parceiro I e o Parceiro II.

Figura 12 – Redesenho do fluxo de colaboração do PNLD Avaliação.



Fonte: Elaborada pelo Autor (2024)

Na primeira versão, considerando o lado esquerdo da Figura 12, os fluxos colaborativos de desenvolvimento do software cursava do Parceiro I ao Parceiro II, o primeiro era responsável

pela elicitação e documentação dos requisitos, e o segundo, a partir do recebimento dos artefatos de software iniciais, era responsável pelo desenvolvimento, evolução e manutenção da solução computacional. Esse fluxo não atendia a dinâmica do domínio da aplicação e era bastante impactado por falhas na comunicação, atrasos e retrabalho, devido à distância entre o elicitor e o desenvolvedor. Nesse sentido, o fluxo passou a centralizar o ciclo de vida de software de apoio à avaliação pedagógica no Parceiro II.

Outro apontamento importante é que o usuário final era tido somente como reacionário à plataforma durante a execução de cada edital/objeto, ou seja, ele não era inserido adequadamente no ciclo de vida do software, restrito apenas ao *feedback* ao nível de suporte. Considerando esses impactos, o usuário passou a ser considerado no ciclo de desenvolvimento com foco nas atividades de elicitação e de validação dos requisitos do software, além do uso e avaliação da plataforma. Essas alterações surgem com o decorrer do tempo e com a execução de novos ciclos de editais/objetos, pois o funcionamento do fluxo anterior já não se mostrava eficaz e não conseguia acompanhar a dinamicidade da política pública. Esse fato desencadeou uma série de problemas, justamente por não suprir a demanda acerca da volatilidade das regras de negócio e dos requisitos de software.

De modo geral, o redesenho do fluxo, plotado à direita da Figura 12, percorreu quatro pilares:

- A adoção de uma **abordagem ágil de desenvolvimento** devido a sua capacidade de se adaptar rapidamente a mudanças e fornecer respostas ágeis e flexíveis à volatilidade legislativa, garantindo que o software permaneça relevante e eficaz ao longo do tempo;
- Uso e adoção de **padronização de formatos e protocolos**, considerando padrões abertos, para integrar os diferentes sistemas de software e as respectivas bases de dados que compõem o ecossistema do PNLD;
- Adoção de fluxos e processos **centrados nas partes interessadas**, incluindo o usuário final e suas perspectivas como parte importante na tomada de decisão, quando considerado o design de soluções computacionais para o setor público. Nesse cenário, o usuário sai do lugar somente de uso e é levado em consideração no ciclo de vida do software;
- Fluxo que apoie a **comunicação entre as partes interessadas** e distribua as atividades pensando a otimização do processo e seu impacto no produto final, considerando a geração de valor ao ecossistema organizacional.

- Ações de **capacitação de usuários finais e integração contínua das partes interessadas**, visando consolidar os conhecimentos-base e necessários para utilizar com efetividade as ferramentas e os sistemas digitais implementados.

Essa redefinição do fluxo impactou diretamente a relação da rede de colaboração não só para com o produto de software desenvolvido, mas também entre as partes interessadas. As ações executadas que levaram à mudança do fluxo foram pensadas e aplicadas a partir de estratégias centradas nas partes interessadas, bem como a colaboração entre as partes em evidência. A solução computacional desenvolvida e mantida no âmbito governamental, conta com mais de quinze mil avaliadores, contempla funcionalidades de integração de dados, incluindo, o recebimento e o envio de dados para outras aplicações de software do governo federal e dispõe de recursos de análise de dados para apoiar os gestores públicos no processo de tomada de decisão.

4.3.3 Inspeção de Artefatos do Software do PNLD Avaliação

É sabido que o ciclo de vida de um software não termina após sua implantação. Na verdade, a partir da entrega do sistema de software para uso, se inicia um novo processo dentro do ciclo de vida: a manutenção e a evolução. A manutenção de software diz do processo de melhorias e correções de um software em desenvolvimento ou já desenvolvido para que o produto permaneça em funcionamento adequado e atendendo as necessidades que o ambiente real de uso demanda, e isso independe do domínio de aplicação, seu tamanho e complexidade [Pressman Roger S; MAXIM].

Alguns aspectos a serem considerados no processo de manutenção: um deles é que esse processo não é uma atividade isolada, mas envolve uma interação entre a organização cliente, o responsável pela manutenção e o usuário [Polo et al. 1999]. Além disso, as atividades de manutenção são classificadas de acordo com sua natureza [Lientz e Shwanson 1980] [Pressman 2009], podendo ser:

1. **Corretiva:** serve para eliminar as falhas encontradas em produção, buscando a correção de defeitos de funcionalidades;
2. **Adaptativa:** são alterações que buscam a adaptação do software a uma nova realidade ou novo ambiente externo;
3. **Perfectiva:** envolve o acréscimo de novos recursos de funcionalidade e melhorias para usuário, bem como integração com outros sistemas; e ainda.

4. **Preventiva:** buscam correções antecipadas de possíveis fontes de problemas encontradas previamente no software.

A natureza do PNLD Avaliação, conforme já apontado no decorrer deste trabalho, exercita uma dinâmica intensa que eleva a complexidade da manutenção. Além disso, existem alguns fatores durante o desenvolvimento que impactam a capacidade de manutenção do sistema. Logo, esta seção apresenta os resultados da inspeção de artefatos de software e da análise temática de entrevistas realizadas com os principais stakeholders. A análise quantitativa dos artefatos de software focou na volatilidade das Histórias de Usuário (US) durante as diferentes fases do projeto e nos Chamados de Suporte, com ênfase na volatilidade e na eficiência de resolução dos problemas relatados.

4.3.3.1 Análise das Histórias de Usuário

Inicialmente, com quatro stakeholders envolvidos, entre o segundo semestre de 2020 e meados de 2022, o projeto apresentou uma volatilidade significativa nos requisitos. Das 190 HU, 30 foram alteradas, 32 foram descontinuadas e 128 foram finalizadas. As alterações realizadas nas 30 HU referem-se a mudanças nas regras de negócio e no escopo, exigindo, consequentemente, alterações nas interfaces gráficas de usuário e no código-fonte. Ao analisar as 128 HU finalizadas, observou-se uma alta taxa de manutenção na equipe de desenvolvimento (P2). Nesse aspecto, os relatórios investigados apontam baixa completude das HU e dificuldades de comunicação entre o responsável pela elicitação dos requisitos e o desenvolvedor (P2).

Destaca-se que as 32 HU descontinuadas indicam que o dinamismo das políticas públicas tornou obsoletos requisitos funcionais previamente modelados, mas que, ao serem revisitados, não faziam mais sentido ser incorporados à plataforma computacional. Nesse caso, modificar as HU não era viável, e novos artefatos de software foram criados para atender à nova realidade das instituições governamentais. Notavelmente, todas as HU passaram por, no máximo, uma atualização, evidenciando alta rotatividade e frequentes mudanças impulsionadas pelos interesses diversos e conflitantes dos stakeholders.

Durante a análise das histórias de usuário, observou-se que a documentação dos artefatos de software não suportava controle de versão e gestão colaborativa por múltiplos engenheiros de software. Isso gerou dificuldades em tarefas como busca, rastreamento de alterações e documentação da evolução da aplicação durante o processo de desenvolvimento.

No segundo semestre de 2022, ocorreu uma mudança no projeto. A Instituição Governamental I (GI1) e o Parceiro II (P2) assumiram a gestão da avaliação pedagógica do PNLD,

enquanto a Instituição Governamental II (GI2) e o Parceiro I (P1) passaram a ser responsáveis por outras fases da política pública. A partir dessa transição até maio de 2024, o P1 criou 55 novas HU, validadas pelo GI1. Das 55 HU analisadas, 5 exigiram alterações ao longo do tempo, mas nenhuma foi descontinuada. As mudanças nas 5 HU referem-se à evolução da política pública para atender às especificidades do processo de avaliação pedagógica de recursos educacionais digitais. Nossa análise revelou dificuldades persistentes no controle de versão dos artefatos de software; no entanto, houve maior completude dos documentos, registros de validação de requisitos e melhor comunicação entre os stakeholders envolvidos.

Dois aspectos centrais relacionados à volatilidade dos requisitos foram identificados. O primeiro diz respeito ao dinamismo inerente às políticas públicas. O segundo refere-se às mudanças solicitadas pelos stakeholders governamentais em relação às regras de negócio especificadas. Nesse segundo ponto, constatou-se que as principais causas dessas mudanças são a rotatividade de pessoas no setor público, a padronização de processos e o conflito de entendimento entre os stakeholders.

Ao investigar as mudanças no escopo do projeto relacionadas às responsabilidades do P1 e P2, verificou-se que a decisão foi tomada pelas instituições governamentais para melhorar as atividades do ciclo de vida do desenvolvimento de software e, consequentemente, a eficiência da solução computacional no suporte à política pública. Para a fase de avaliação pedagógica do PNLD, decidiu-se que o P2 seria responsável por todo o ciclo de desenvolvimento de software.

Para entender o impacto da mudança feita pelas instituições governamentais, aplicamos um teste estatístico para verificar a significância da centralização em dois stakeholders no processo de desenvolvimento de software. Assim, o **Grupo 1** representa a formação inicial com quatro stakeholders, e o **Grupo 2** representa a nova formação com dois stakeholders responsáveis pelo ciclo de vida do desenvolvimento de software. Para essa atividade, foram realizados os seguintes testes:

- **Estatística qui-quadrado:** indica a diferença geral entre os dados observados e os dados esperados caso não houvesse relação entre os grupos.
- **Valor-p:** indica a probabilidade de que as diferenças observadas sejam devido ao acaso. Um valor-p inferior a 0,05 é geralmente considerado estatisticamente significativo.
- **Graus de liberdade:** refere-se ao número de valores independentes ou quantidades que podem ser atribuídos a uma distribuição estatística.

- **Frequências esperadas:** são as frequências esperadas caso os grupos não apresentassem relação.

Tabela 1 – Frequência dos valores observados e esperados para os dois grupos

Grupos	Alteradas	Descontinuadas	Finalizadas
Quatro Stakeholders (Observado)	30	32	128
Quatro Stakeholders (Esperado)	26,6	24,32	139,08
Dois Stakeholders (Observado)	5	0	55
Dois Stakeholders (Esperado)	8,4	7,68	43,92

A Tabela 1 apresenta os valores observados e esperados para as métricas das histórias de usuário (alteradas, descontinuadas e finalizadas) entre os grupos 1 e 2. A Tabela 2 mostra os resultados estatísticos calculados, cuja análise detalhada é apresentada a seguir.

Tabela 2 – Resultado estatístico dos grupos investigados

Estatística	Valor
Qui-quadrado	15,594
Valor-p	0,000411
Graus de Liberdade	2

Os graus de liberdade para o teste qui-quadrado são calculados com base no número de categorias menos um para cada dimensão. Neste caso, temos três categorias (alteradas, descontinuadas, finalizadas) e dois grupos. Assim, os graus de liberdade são $(3-1) \times (2-1) = 2$. A estatística qui-quadrado (15,594) e o valor-p muito pequeno ($0,000411 < 0,05$) indicam uma diferença estatisticamente significativa entre os dois grupos (quatro stakeholders e dois stakeholders) em relação ao número de histórias de usuário alteradas, descontinuadas e finalizadas.

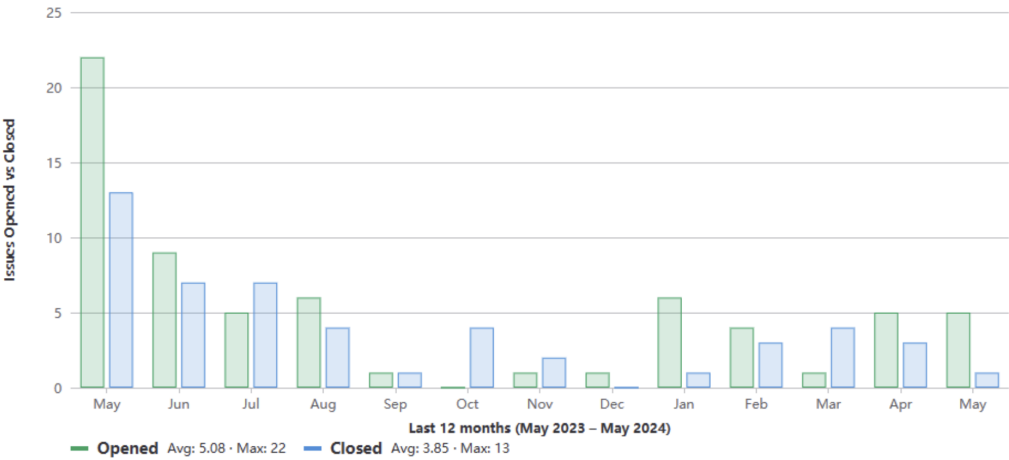
Os resultados do teste qui-quadrado sugerem que as diferenças nos números de histórias de usuário alteradas, descontinuadas e finalizadas entre os dois grupos são estatisticamente significativas e provavelmente não são devidas à variação aleatória. A redução no número de participantes parece ter levado à redução no número de histórias alteradas e descontinuadas e ao aumento no número de histórias finalizadas, em comparação com o esperado caso a mudança nos participantes não tivesse efeito. Isso apoia a hipótese de que reduzir o número de participantes melhora a eficiência e a estabilidade do projeto.

4.3.3.2 Análise de Chamados de Suporte

Os resultados da análise de chamados de suporte, são apresentados com foco na volatilidade e eficiência de resolução dos problemas relatados na plataforma PNLD Avaliação. Esta

análise baseia-se em dados extraídos do rastreador de problemas do GitLab e do sistema de chamados GLPI. Considerando os dados de *front-end* no GitLab e analisando os problemas registrados, conforme Figura 13 e baseada nos dados da Tabela 3, indica que uma proporção significativa de problemas (64,5%) foi resolvida, com um percentual menor de *bugs* relatados (57,1%) sendo fechados. Isso reflete um nível moderado de eficiência na resolução de problemas e *bugs* no *front-end*.

Figura 13 – Relação entre problemas abertos e fechados de Maio de 2023 a Maio de 2024 no GitLab (*front-end*).



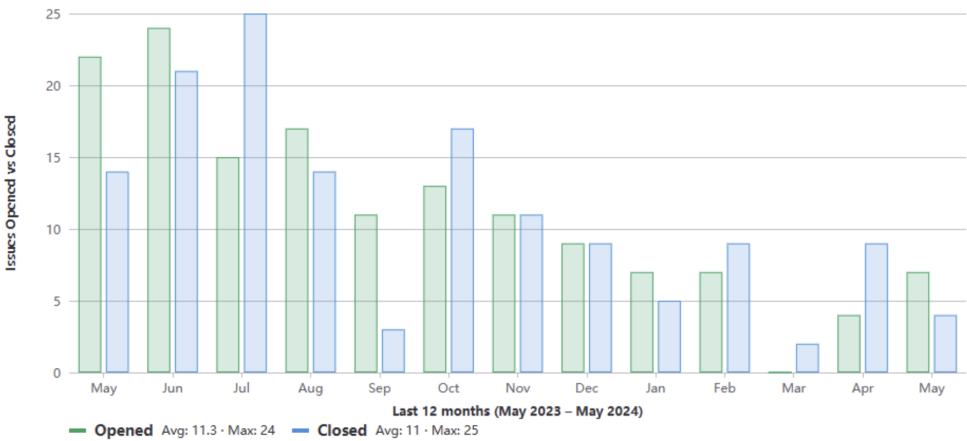
Fonte: Elaborada pelo Autor (2024)

Tabela 3 – Número de problemas no *front-end* no GitLab de maio de 2023 a maio de 2024.

Status	Número de Problemas no GitLab, relacionados ao frontend	Total
Problemas Abertos	113	318
Problemas Fechados	205	
Bugs Fechados	12	21
Bugs Abertos	9	

Quando analisados os dados de *back-end*, revela-se uma taxa de resolução ainda mais alta, conforme mostrado na Figura 14 e baseando-se nos dados da Tabela 4, com 72,3% do total de problemas resolvidos e uma impressionante taxa de fechamento de bugs de 84,6%. Isso sugere maior eficiência no tratamento de problemas no *back-end* em comparação ao *front-end*.

Figura 14 – Relação entre problemas abertos e fechados de maio de 2023 a maio de 2024 no GitLab (*back-end*).



Fonte: Elaborada pelo Autor (2024)

Tabela 4 – Número de problemas no *back-end* no GitLab de maio de 2023 a maio de 2024.

Status	Número de Problemas no GitLab, relacionados ao backend	Total
Problemas Abertos	361	1304
Problemas Fechados	943	
Bugs Fechados	44	52
Bugs Abertos	8	

Os dados do sistema GLPI indicam uma alta taxa de fechamento de 97,3% dos chamados relacionados à plataforma PNLD Avaliação, destacando um desempenho sólido na resolução de problemas relatados pelos usuários, conforme mostrado na Tabela 5.

Tabela 5 – Número de Chamados no GLPI por status.

Status	Número de Chamados no GLPI
Abertos	42
Fechados	571
Resolvidos	26
Total de Chamados	639

A análise revela vários insights principais: i) Os dados do GitLab demonstram muitos problemas relatados e resolvidos, indicando um processo de manutenção ativo e responsivo. No entanto, o *front-end* apresenta uma taxa de resolução de *bugs* menor que o *back-end*, sugerindo áreas potenciais para melhoria nos processos de desenvolvimento e testes do *front-end*; ii) Comparando as taxas de resolução antes e depois da centralização dos stakeholders para GI1 e

P2, houve uma notável melhoria na eficiência; iii) As altas taxas de fechamento no sistema GLPI após a centralização sugerem que a redução do número de stakeholders impactou positivamente o processo, otimizando a resolução de problemas e reduzindo redundâncias.

A equipe de *back-end* mostrou maior eficiência na resolução de problemas e *bugs* em comparação ao *front-end*, ou a gestão do projeto priorizou essas resoluções no *back-end*. As altas taxas de fechamento no sistema GLPI enfatizam ainda mais a eficácia da estrutura de suporte atual em abordar e resolver prontamente os problemas relatados.

4.3.3.3 Resultados e Discussões

Os resultados apresentados oferecem insights valiosos sobre os desafios e avanços no gerenciamento de requisitos e suporte técnico na plataforma PNLD Avaliação, destacando tanto os sucessos quanto as áreas que necessitam de atenção.

A análise das histórias de usuário evidencia que a volatilidade dos requisitos foi um fator predominante, especialmente no período inicial do projeto. A presença de quatro stakeholders distintos, com interesses e dinâmicas diversas, contribuiu para uma alta taxa de alterações e descontinuações de requisitos. No entanto, quando houve a centralização em dois stakeholders (GI1 e P2) gerou uma notável redução na volatilidade e uma maior eficiência na finalização de histórias de usuário, sugerindo maior estabilidade e alinhamento no processo de engenharia de requisitos. Este resultado é corroborado pelos testes estatísticos, que indicam uma diferença significativa entre as duas configurações de gestão. Ainda assim, os desafios relacionados à documentação, controle de versão e comunicação inicial entre as equipes refletem a necessidade de práticas mais robustas de engenharia de requisitos.

De forma semelhante, a análise dos *tickets* de suporte oferece uma visão complementar sobre os desafios operacionais do sistema e a eficiência na resolução de problemas. Os dados apontam um bom desempenho geral, com altas taxas de resolução tanto no GitLab quanto no GLPI. Entretanto, observa-se uma disparidade entre a eficiência na resolução de problemas do *back-end* e do *front-end*. Essa diferença pode indicar uma priorização gerencial dos problemas de *back-end*, o que está alinhado com a maior eficiência geral observada neste processo, mas também, pode-se inferir que haja possíveis limitações na capacidade técnica e de teste do *front-end*. Além disso, a alta taxa de resolução no GLPI sugere que, após a centralização dos stakeholders, o processo de suporte tornou-se mais ágil e eficiente, reduzindo redundâncias e potencializando a resposta às demandas dos usuários.

Ao comparar ambos os conjuntos de dados, fica evidente que a centralização dos stakehol-

ders não apenas melhorou a estabilidade dos requisitos, como também aumentou a eficiência na resolução de *tickets* de suporte. Essa melhoria é refletida nas altas taxas de resolução e na redução significativa de problemas e *tickets* abertos ao longo do tempo. Os resultados destacam os benefícios de uma abordagem simplificada na engenharia de requisitos e na resolução de problemas, com menos stakeholders, porém mais alinhados, contribuindo para uma melhor manutenção do sistema e maior satisfação dos usuários.

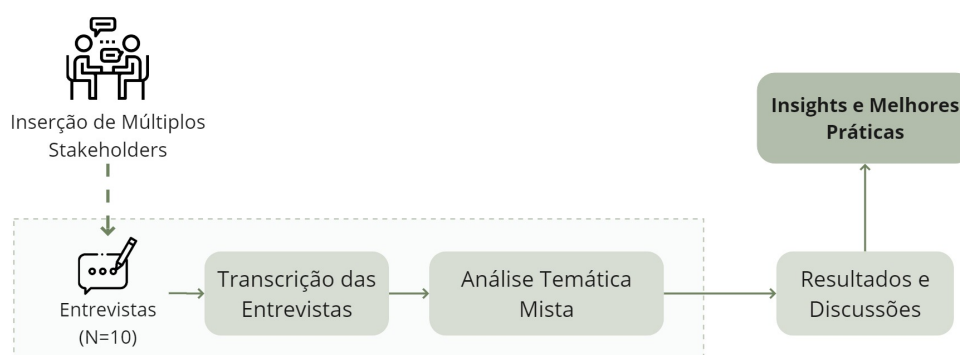
4.4 ESTÁGIO 02: CONDUÇÃO DE ENTREVISTAS COM MÚLTIPLOS STAKEHOLDERS

Nesta seção, serão discorridas algumas questões metodológicas que permeiam a execução das entrevistas com múltiplos stakeholders. Assim, serão discutidos os instrumentos de coleta e análise, a descrição demográfica dos participantes e os resultados e discussões.

4.4.1 Instrumento de Coleta e Análise e Dados Demográficos

Para coletar os dados necessários, optou-se por uma entrevista semiestruturada com diferentes stakeholders envolvidos na Avaliação Pedagógica do PNLD. Entrevistas semiestruturadas permitem flexibilidade no questionamento ao mesmo tempo, em que garantem que os tópicos principais sejam abordados [Lazar et al. 2017], nessa condução foi essencial para reunir insights profundos sobre as necessidades e percepções das partes interessadas. Todas as entrevistas ocorreram por meio de videochamadas, considerando que a equipe do projeto é descentralizada e atua em diversos estados brasileiros. A execução desta fase da pesquisa seguiu o percurso metodológico especificado na Figura 15.

Figura 15 – Métodos adotados para condução do Estágio 02.



Após entrevistas com 10 stakeholders principais, identificados como participantes da entrevista (IP) na Tabela 6, a análise foi realizada utilizando uma abordagem mista de análise temática, combinando métodos indutivos e dedutivos. A análise temática é uma abordagem qualitativa que identifica, analisa e relata padrões (temas) nos dados. No nosso caso, a análise indutiva permitiu que novos temas emergissem diretamente dos dados, enquanto a análise dedutiva foi usada para explorar temas pré-identificados, como a volatilidade de requisitos e a influência de múltiplos stakeholders. De acordo com [Braun e Clarke 2006], a análise temática envolve várias fases, incluindo familiarização com os dados, geração de códigos iniciais, busca de temas, revisão de temas, definição e nomeação de temas e produção do relatório final. Esse processo sistemático e rigoroso garantiu que os temas identificados representassem com precisão os dados coletados.

Tabela 6 – Dados Demográficos dos Participantes das Entrevistas

Código	Grupo	Função	Duração	Contagem de Palavras
IP 1	P2	Coordenador de Projeto	15 min	493
IP 2	GI2	Coordenador Geral	10 min	438
IP 3	P2	Designer UX	30 min	1901
IP 4	P2	Arquiteto de Software	30 min	994
IP 5	P2	Gerente de Projetos	30 min	1040
IP 6	GI2	Coordenador Local	10 min	392
IP 7	GI2/P2	Engenheiro de Requisitos	10 min	613
IP 8	P2	Engenheiro Líder de Requisitos	20 min	797
IP 9	P2	Engenheiro de Requisitos	30 min	377
IP 10	P2	Gerente de Projetos	30 min	208

4.4.2 Análise Temática de Abordagem Mista

- **Comunicação e coordenação**

Um tema recorrente nas entrevistas foi a importância crítica da comunicação e coordenação entre os diversos stakeholders. O Participante 1 destacou como a falta de comunicação eficaz entre a equipe de levantamento de requisitos e a equipe de desenvolvimento resultou em funcionalidades mal especificadas e uma alta taxa de manutenção. Ele mencionou: *"Isso impactou diretamente as funcionalidades desenvolvidas, pois não atendiam às necessidades da (Instituição Governamental I) e geraram uma alta taxa de manutenção."* A pandemia agravou essas questões, complicando ainda mais a validação e a implementação de requisitos: *"Não tivemos a contraprova para validar com eles e esclarecer dúvidas de requisitos"*, relatou o Participante 1. O Participante 2 também mencionou a complexidade da comunicação entre

diferentes agências e a necessidade de coordenação para evitar problemas ao implementar novos editais de livros: *"As regras de negócio mudam a cada edital... precisamos refazer toda a configuração e regras cada vez."* Já o Participante 6 notou uma melhoria na comunicação da equipe e na empatia após a mudança nas atribuições do parceiro, embora, em sua opinião, a equipe P1 estivesse significativamente sobrecarregada devido às suas funções distribuídas em outros projetos: *"Agora temos muito mais aderência e comunicação... No entanto, a equipe P1 continua sobrecarregada e precisa de mais tempo dedicado a tópicos de integração."* O Participante 7 mencionou dificuldades iniciais na transição do sistema antigo para o PNLD Avaliação devido a especificações e cronogramas desalinhados. O Participante 9 enfatizou a necessidade de reuniões mais objetivas e produtivas para melhorar a eficiência, mencionando: *"O principal desafio foi a falta de treinamento inicial adequado... isso atrasou meu progresso e gerou inseguranças."*

- **Volatilidade dos requisitos**

Vários entrevistados mencionaram a volatilidade dos requisitos como um grande desafio. O Participante 3 destacou que as mudanças frequentes nas regras dos editais impactam diretamente o desenvolvimento, dificultando a estabilização do sistema: *"As mudanças frequentes nas regras dos editais impactam diretamente o desenvolvimento, dificultando a estabilização do sistema."* Ele também mencionou a alta rotatividade de pessoal na (Instituição Governamental I), o que adiciona mais complexidade à gestão de requisitos: *"A alta rotatividade de pessoal na (Instituição Governamental I) adiciona mais complexidade à gestão de requisitos."* O Participante 4 corroborou essa visão, apontando que mudanças frequentes nos requisitos durante a fase inicial causaram confusões e atrasos: *"Houve uma série de confusões de comunicação, e o processo se arrastou bastante, muitas mudanças ao longo do caminho."* Os Participantes 7 e 8 destacaram os desafios da transição do sistema antigo para o PNLD Avaliação. Muitos requisitos foram copiados sem inovação, levando à instabilidade inicial. Eles afirmaram que: *"Inicialmente, não houve inovação, apenas um copiar e colar do sistema antigo... Isso levou a muitos erros e muito retrabalho."* No entanto, o Participante 8 mencionou que mudanças na gestão levaram a um novo treinamento e re-documentação de processos, causando atrasos e ineficiências: *"Cada mudança de gestão traz novas visões sobre o fluxo de avaliação, levando a re-treinamentos e à re-documentação de grandes módulos."*

- **Processo de desenvolvimento**

O processo de desenvolvimento foi um tema central, destacando as dificuldades enfrentadas quando os requisitos são coletados por uma equipe e desenvolvidos por outra. O Participante 5 descreveu como a falta de participação adequada da equipe de desenvolvimento na fase de levantamento de requisitos resultou em muitas dúvidas e ajustes durante o desenvolvimento: *"Não participamos adequadamente desse processo de requisitos. Herdamos tudo da (Parceira 1) e tivemos que descobrir como continuar."* Ele mencionou que a mudança para um modelo em que a equipe de desenvolvimento também assume o levantamento de requisitos melhorou a comunicação e a eficiência: *"Essa mudança reduziu a necessidade de ajustes e retrabalho, permitindo um desenvolvimento mais ágil e eficiente."* O Participante 7 discutiu as dificuldades de integrar novas funcionalidades e a falta de fases adequadas de testes, o que resultou em *bugs*: *"As ferramentas foram construídas à medida que a avaliação progredia, sem espaço para testes... isso resultou em muitos bugs."*

• Metodologias ágeis e documentação

Metodologias ágeis e documentação foram temas recorrentes. Os Participantes 3 e 4 destacaram a dificuldade de equilibrar a necessidade de documentação com os princípios das metodologias ágeis. O Participante 3 sugeriu criar documentos mais acessíveis e informais para facilitar o entendimento dos desenvolvedores: *"Sinto falta de algo onde possamos dizer: olha, isso vai mudar aqui."* O Participante 4 enfatizou a necessidade de um papel específico para atuar como intermediário entre o negócio e a equipe de desenvolvimento, sugerindo um líder técnico focado em requisitos: *"Eu poderia fazer isso se fosse minha função. Mas não posso fazer tudo."* O Participante 8 apontou o desafio de transitar o estilo de documentação, mantendo o formato existente para garantir consistência. Ele afirmou: *"Mantive o layout existente das histórias de usuário para garantir consistência, embora isso exigisse ajustes significativos."*

• Integração e melhoria contínua

A integração entre sistemas e a melhoria contínua foram destacadas pelos Participantes 4 e 5. O Participante 4 mencionou que a integração inicial foi difícil devido à falta de comunicação, mas melhorou com o tempo, à medida que as equipes se familiarizaram: *"Em um certo ponto, houve uma virada, e conseguimos avançar com isso e entregar aquela parte da integração."* O Participante 5 ressaltou a importância de revisar constantemente os requisitos e a documentação, mencionando que a introdução de testes automatizados poderia beneficiar significativamente o projeto: *"Uma das ações que trabalharemos em breve são os testes automatizados. Acredito*

que isso trará muito para o projeto." O Participante 6 destacou a necessidade de uma melhor integração, sugerindo que a etapa de integração seja internalizada: *"Embora a comunicação tenha melhorado, mais tempo e atenção são necessários para a integração."* O Participante 10 apontou a necessidade contínua de mais mão de obra qualificada e os potenciais benefícios de transferir algumas tarefas do P1 para o P2, afirmando: *"Precisamos de mais mão de obra qualificada e mais reuniões presenciais para abordar as nuances das regras de negócio."*

- **Múltiplos stakeholders**

O envolvimento de múltiplos stakeholders foi um fator significativo na complexidade do projeto. O Participante 1 mencionou: *"Não é fácil ter duas instituições, uma levantando e outra desenvolvendo. Para isso funcionar, é necessário um grande ajuste de comunicação, o que não ocorreu."* O Participante 3 observou que: *"O PNLD é um processo complexo... (Instituição Governamental I) e (Instituição Governamental II) têm entendimentos e prioridades diferentes."* O Participante 5 discutiu a mudança na abordagem de requisitos: *"Com o tempo, percebemos que essa ação de criar os requisitos e nos passar, todos documentados, não era mais necessária."* O Participante 6 destacou a sobrecarga da equipe P1, afirmando: *"A equipe P1 está muito sobrecarregada e precisa de mais tempo dedicado às tarefas de integração."* Os Participantes 7 e 8 discutiram problemas decorrentes das múltiplas responsabilidades de validação dentro da GI2. O Participante 7 reconheceu os benefícios de ter uma única equipe gerenciando tanto a especificação quanto o desenvolvimento, mencionando que: *"A transição trouxe melhorias significativas, pois é mais fluido ter a especificação e o desenvolvimento, concentrados em uma equipe."*

4.4.3 Resultados e Discussões

A análise temática das entrevistas forneceu insights qualitativos sobre as complexidades de gerenciar um ambiente dinâmico com múltiplos stakeholders. Os principais temas identificados incluem comunicação e coordenação, volatilidade de requisitos, processo de desenvolvimento, metodologias ágeis e documentação, integração e melhoria contínua, além dos desafios impostos pela multiplicidade de stakeholders. Os participantes destacaram questões como a necessidade de uma comunicação mais eficaz entre engenheiros de requisitos e desenvolvedores. A lacuna de comunicação entre as equipes de levantamento de requisitos e desenvolvimento, exacerbada pela falta de processos colaborativos robustos e pela complexidade

inerente ao projeto, não apenas resultou em mal-entendidos e retrabalho, mas também impactou negativamente a qualidade e a estabilidade do sistema entregue.

A volatilidade dos requisitos, somada à alta rotatividade de pessoal nas instituições envolvidas, aponta para uma necessidade urgente de estratégias mais eficazes de gestão de mudanças e documentação contínua. O uso de metodologias ágeis, embora tenha potencial para flexibilizar processos, mostrou-se limitado na ausência de um equilíbrio entre a produção de documentação clara e a implementação de fluxos de trabalho dinâmicos. A sobrecarga das equipes, especialmente da P2, também ressalta a importância de uma alocação de recursos mais eficiente e de um planejamento estratégico que considere a carga de trabalho realista para as equipes envolvidas.

Adicionalmente, o envolvimento de múltiplos stakeholders com entendimentos e prioridades divergentes amplificou os desafios de alinhamento e integração. A ausência de um papel mediador formalizado, como o de um líder técnico focado em requisitos, contribuiu para a falta de clareza e para o atraso na resolução de problemas críticos. Apesar das dificuldades, alguns avanços foram observados, como a melhoria da comunicação após mudanças organizacionais e a proposta de implementação de testes automatizados, que podem trazer benefícios substanciais para a estabilidade e qualidade do sistema.

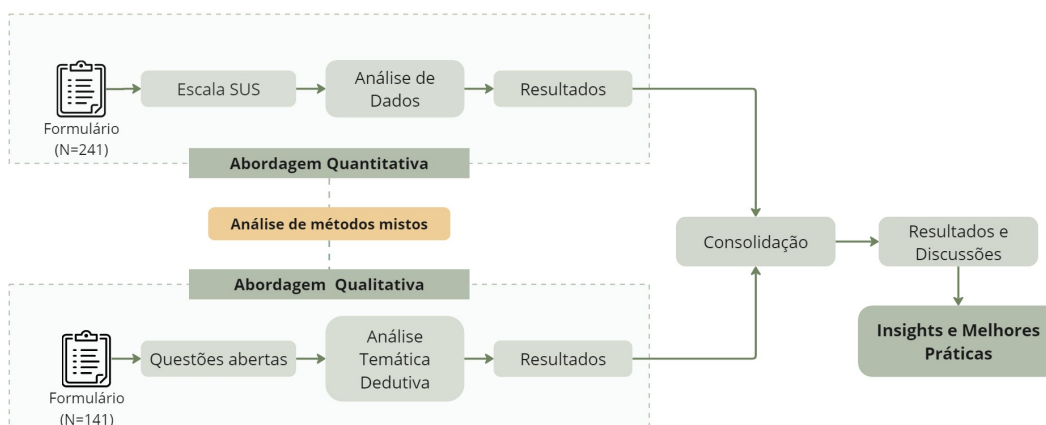
4.5 ESTÁGIO 03: ANÁLISE DE USABILIDADE E CAPTURA DE PERCEPÇÕES DOS USUÁRIOS FINAIS

Nesta seção serão discorridas algumas questões metodológicas que permeiam a avaliação de usabilidade do sistema de apoio à execução da avaliação pedagógica do PNLD. Dessa forma, serão discutidos os instrumentos de coleta utilizados e as justificativas por sua escolha, os métodos de análise, o nível de confiança da pesquisa com base no quantitativo de participantes, e, por fim, os resultados e discussões.

4.5.1 Instrumentos de Coleta e Análise

Para coletar os dados necessários, inclusive aplicar o questionário de avaliação, foram utilizadas algumas ferramentas. As atividades foram centralizadas em um criador de questionário digital e com acesso online, para distribuição via link e pelo e-mail, o canal de comunicação entre os participantes. A metodologia aplicada nesse estágio está representada na Figura 16.

Figura 16 – Métodos adotados para condução do Estágio 03.



O método escolhido para avaliação de usabilidade foi a Escala de Usabilidade do Sistema (SUS, do inglês System Usability Scale) que foi desenvolvido por John Brooke na década de 80 [Brooke 2013] e é utilizado para avaliar três aspectos de qualquer tipo de interface: a efetividade, que está para o sucesso no uso do sistema, a eficiência, que diz do esforço empregado para o uso do sistema e a satisfação que parte da experiência do usuário ao usar o sistema. Segundo próprio criador, a construção do SUS se dá por vários critérios considerados, sendo que no método tinha que haver fortes intercorrelações entre todos os itens/questões do formulário; que o número total de itens/questões teria de ser limitado devido às limitações de tempo para gestão do questionário à época da pesquisa e ainda considerar que, embora o intuito fosse obter resultados com rapidez, os usuários precisavam pensar sobre o que estavam respondendo.

O SUS é composto por dez afirmações, conforme Tabela 7, cada uma com uma escala de cinco pontos que varia de 1, que equivale a "Discordo totalmente", até 5, que equivale a "Concordo plenamente. Há cinco afirmações positivas e cinco afirmações negativas, que se alternam, obrigando o usuário a ler sempre a questão seguinte, empregando um esforço para compreensão dos itens [Bangor et al. 2009].

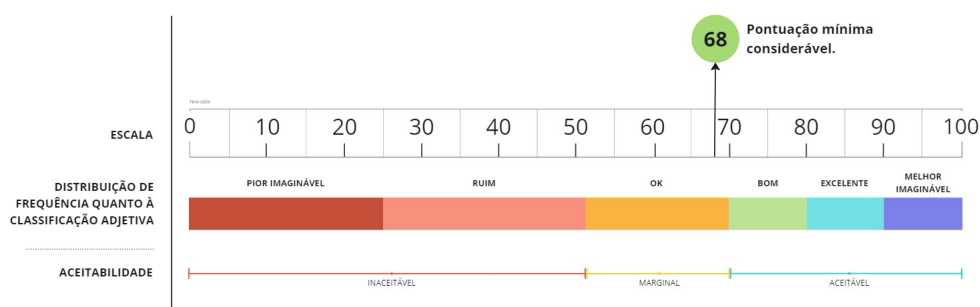
Tabela 7 – Questões que constituem o SUS

ORDEM	ITENS
1º	Eu acho que gostaria de usar esse sistema com frequência.
2º	Eu acho o sistema desnecessariamente complexo.
3º	Eu achei o sistema fácil de usar.
4º	Eu acho que precisaria de ajuda de uma pessoa com conhecimentos técnicos para usar o sistema.
5º	Eu acho que as várias funções do sistema estão muito bem integrado.
6º	Acho que o sistema apresenta muita inconsistência.
7º	Eu imagino que as pessoas aprenderão como usar esse sistema rapidamente.
8º	Eu encontrei o sistema bloqueado de uso.
9º	Eu me sinto confiante ao usar o sistema.
10º	Eu preciso aprender várias coisas novas antes de conseguir usar o sistema.

Fonte: Adaptado de [Brooke 2013]

Após a aplicação do questionário, deve-se seguir alguns passos, bem especificados por [Braum 2019]: quando um usuário responde ao questionário, sua resposta terá como pontuação o valor correspondente a posição que sua resposta ocupa na escala e, então, para as perguntas ímpares (1, 3, 5, 7, 9) se subtrai 1 da pontuação que o usuário respondeu, ou seja, se o usuário marcou a posição 3, logo será calculado $3 - 1 = 2$, sendo o 2 o novo valor correspondente ao item em questão. Para as perguntas pares (2, 4, 6, 8, 10) subtraia 5, ou seja, $5 - X$, onde X é o valor correspondente à posição que a resposta do usuário ocupa na escala. Em seguida, somam-se todos os valores encontrados e multiplica-se por 2,5, por fim, os resultados do questionário vão variar entre 0 a 100. A localização que a pontuação obtida ocupa na escala atribui ao sistema um grau nominal de aceitabilidade e uma classificação também nominal Adjetiva; ambas, o grau e a classificação, podem ser munidas de uma representação da distribuição de frequência quanto a elas. A representação da escala SUS pode ser observada na Figura 17.

Figura 17 – Demonstração visual da Escala de Usabilidade de Sistema.



Fonte: Adaptado de [Brooke 2013, Bangor et al. 2009, Braum 2019]

A pontuação média apontada parte da defesa de [Sauro 2011], que diz que para ser considerado satisfatório em termos de usabilidade, deve pontuar no mínimo 68,2 pontos. Abaixo disso, já se deve rever o sistema e sua construção. Essa pontuação média é endossada por outros estudos que chegaram a algumas médias para serem consideradas a partir da interface que será avaliada, conforme Tabela 8. Os diversos estudos apontam para a robustez do SUS como ferramenta para análise de usabilidade percebida, e tem sido usada para avaliar uma ampla gama de interfaces que incluem plataformas web, smartphones/celulares, dentre outras [Bangor et al. 2009]. Essa pontuação pode ser usada como uma das formas de interpretar os resultados obtidos pelo SUS, porém, como mostrado na Figura 17, há outras possíveis interpretações. Essa pontuação média é endossada por outros estudos que chegaram a algumas médias para serem consideradas a partir da interface que será avaliada, conforme Tabela 8.

Tabela 8 – Pontuações médias consideradas em relação ao tipo de interface

Tipo de interface	Pontuação média
Web	68,2
Smartphones	65,9
IVR	72,4
GUI	76,2
Hardware	71,8
TV	67,8
Total	69,5

Fonte: Adaptado de [Bangor et al. 2009]

Alguns fatores específicos do ambiente de uso e da realidade dos participantes da pesquisa motivaram a utilização do SUS. Os usuários finais, professores avaliadores do PNLD, lidam com um sistema bastante complexo e com uma lógica informacional também muito complexa. E uma das principais características está na ficha de avaliação dos materiais didáticos, que se trata de um formulário extenso que conta com mais de 280 campos para preenchimento. Pensando nisso, procurou-se um método que tivesse eficácia e condissesse com a investigação proposta, mas que não apresentasse muitas questões e campos de preenchimento. Portanto, o SUS foi mais aderente a esses fatores.

4.5.2 Descrição Demográfica da População e Amostra

A aplicação do questionário SUS foi direcionada a professores e especialistas de diversas áreas do conhecimento que estiveram em contato com a plataforma PNLD Avaliação. Para

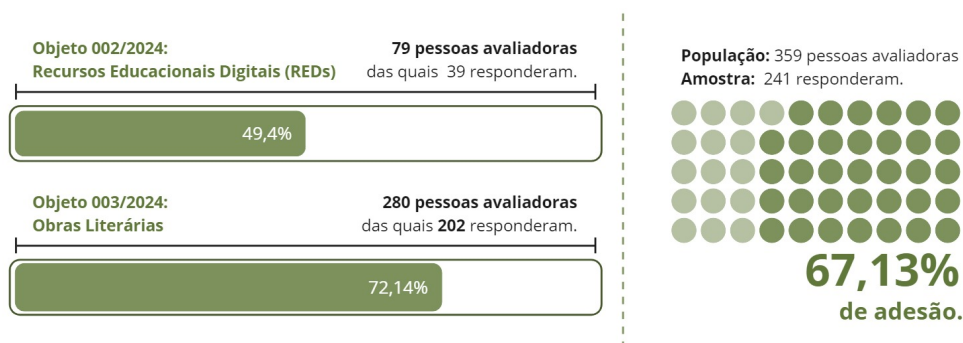
isso, foi utilizada uma plataforma digital de produção de formulário e disparo via e-mail para todos os possíveis participantes. Parte-se do pressuposto de que todos os professores avaliadores possuem hábitos tecnológicos, ocasionados pela contemporaneidade e aptidão confirmada para ocuparem as funções as quais executam. No todo, os participantes deveriam se enquadrar em dois principais:

1. **Requisito 1:** Estar ativo no Banco de Avaliadores do PNLD;
2. **Requisito 2:** Estar participando de um dos editais que estivesse sendo executado à época;

O primeiro requisito é justificado pelo fato de que, à época em que foram executadas as primeiras atividades dessa pesquisa, o Banco de Avaliadores do PNLD contava com 5682 (cinco mil, seiscentos e oitenta e dois) professores e especialistas cadastrados na plataforma. Desses, só poderiam ser considerados os que estavam ativos e com cadastros funcionais e atualizados. O segundo requisito diz respeito ao método escolhido para avaliação de usabilidade, pois se defende que, para melhor percepção e julgamento correto da plataforma pelos avaliadores, os mesmos precisam ter tido contato recente com a plataforma e/ou estar utilizando com recorrência a plataforma.

O disparo via e-mail do convite para avaliação iniciou quando a população foi definida, e uma vez que foram aplicados os requisitos citados, a população estudada compreendeu dois Editais Objetos que estavam sendo executados no ano de 2023. O Objeto 002/2024 avaliou os Recursos Educacionais Digitais (REDs) que apoiam e reforçam a aprendizagem e contou com a participação de 79 (setenta e nove) professores avaliadores. E o Objeto 003/2024, que teve foco em Obras Literárias e contou com a participação de 280 (duzentos e oitenta) professores avaliadores. O formulário foi disparado em remessas por e-mail no período entre 03/08/2023 e 27/11/2023, conforme evolução das pessoas avaliadoras na plataforma enquanto avaliava as obras e materiais didáticos. Ao todo, foram 116 (cento e dezesseis) dias dedicados ao recebimento de avaliações. Considerando os requisitos e somando os participantes dos dois Editais Objetos, a população considerada é de 359 professores avaliadores, conforme plotado na Figura 18.

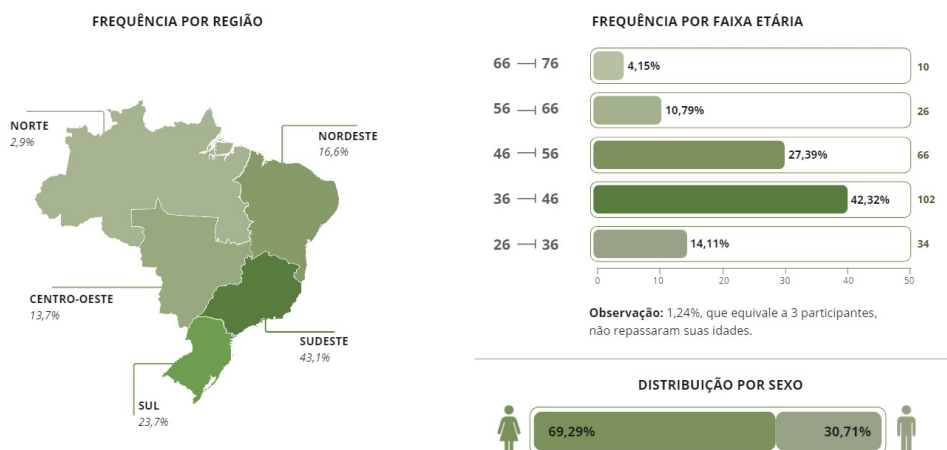
Figura 18 – População e amostra da pesquisa



Fonte: Elaborada pelo Autor a partir dos dados da pesquisa (2024)

Considerando os números acima, a partir de agora serão dispostas as características demográficas da amostra obtida, conforme demonstrado na Figura 19. A primeira observação a ser pontuada está na distribuição de participantes nas regiões. Embora haja representação em todas as cinco regiões, há predominâncias bastante visíveis. Quase metade da amostra, 43,15% dos participantes, estão alocados na região Sudeste, enquanto um pouco menos de 1/4 da amostra, 23,65%, estão na região Sul. As duas regiões juntas concentram 161 participantes, o que equivale a 66,80% da amostra estudada. O Nordeste compreende uma frequência de 16,60% dos participantes, seguido do Centro-Oeste com frequência de 13,69% dos participantes. Por último, aparece a Região Norte com a menor concentração de participantes considerando a amostra da pesquisa, com somente 2,90%, que equivale a somente 7 participantes.

Figura 19 – Distribuição por região, faixa etária e sexo da amostra

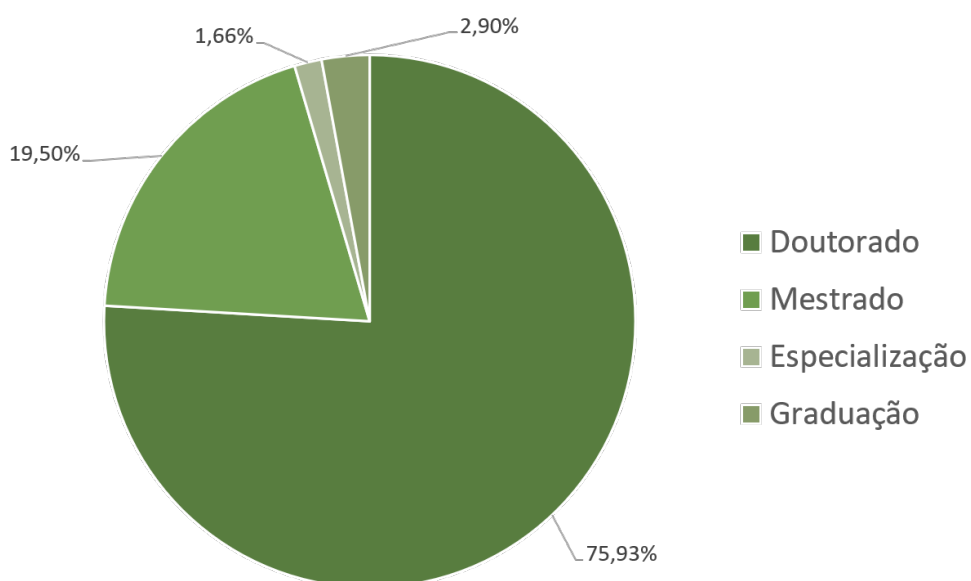


Fonte: Elaborada pelo Autor a partir dos dados da pesquisa (2024)

Ainda na Figura 19, podem-se mapear algumas características dos participantes, como Faixa Etária e Sexo. Quanto à distribuição por faixa etária, tem-se a maior frequência de participantes com idade de 36 até 46 anos, uma representação de 42,32% da amostra. Com idades entre 46 e 56 anos, têm-se 27,39% dos participantes. Com a terceira maior frequência, tem-se a faixa de 26 a 36 anos, com 14,11%, seguido 56 a 66, com 10,79% e, por último, com idades entre 66 e 76, 4,15% dos participantes desta pesquisa. Considerando o sexo dos participantes, pontua-se a maioria do sexo feminino, com uma representação de 69,29%, ou seja, 167 dos participantes, enquanto 30,71% dos participantes são do sexo masculino, o que contabiliza 30,71% da amostra.

É possível também analisar a distribuição de participantes em relação à formação acadêmica, conforme Figura 20. Reforça-se que todos os participantes da pesquisa têm pelo menos uma graduação, licenciatura ou bacharelado, em qualquer área do conhecimento. Assim, percebe-se que pouco mais de 3/4 (75,93%) da amostra estudada apresenta o título de Doutorado. Mestres e mestras representam pouco menos de 1/5 da amostra (19,50%); Especialistas são 1,66% da amostra, enquanto os 2,90% dos participantes são graduados ou graduadas. Como se trata de um processo de avaliação de obras e materiais didáticos, o interesse por participar dos processos de avaliação é em sua maioria de doutores/as ou mestres/as, consequentemente a expressividade entre participantes com esses títulos prevalece.

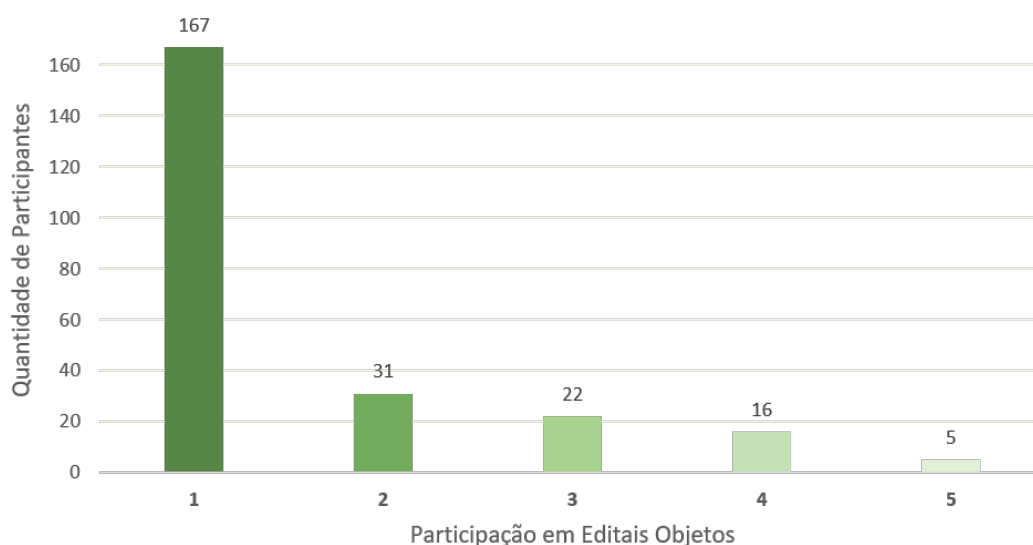
Figura 20 – Distribuição por título dos participantes



Fonte: Elaborada pelo Autor a partir dos dados da pesquisa (2024)

O tempo/experiência de uso que esses avaliadores têm com o software PNLD Avaliação também foi investigado. Assim, foi mapeada também a quantidade de Editais Objetos que cada avaliador participou conforme Figura 21. Pode-se, ainda, dizer do fato de que 69,29% dos participantes estavam tendo sua primeira experiência com a plataforma enquanto esta pesquisa estava acontecendo, o que equivale a 167 dos 241 que participaram. Da amostra, 31 participantes (12,86%) estavam em seu segundo Edital Objeto; 22 participantes (9,13%) estavam em seu terceiro Edital Objeto; 16 participantes (6,64%) estavam em seu quarto Edital Objeto, enquanto somente 5 participantes desta pesquisa, (2,07%) estavam em seu quinto Edital Objeto. Esta análise acerca da experiência é fundamental para compreender como o tempo em contato com o software reflete no julgamento dos participantes em termos de usabilidade do sistema.

Figura 21 – Distribuição por título dos participantes



Fonte: Elaborada pelo Autor a partir dos dados da pesquisa (2024)

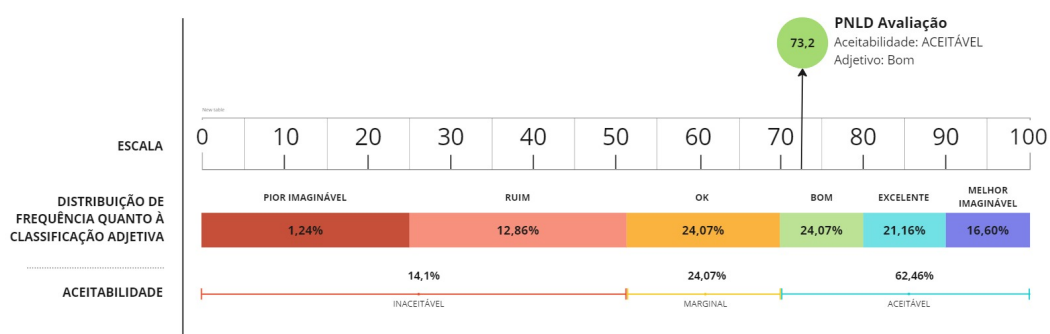
Uma vez percorridas as características da população e amostra, é importante destacar que o nível de confiança desta pesquisa, a partir do número de respostas obtidas frente a população, é de 99%, que indica que a probabilidade de que a verdadeira média ou proporção na população esteja dentro da margem de erro estimada em 4,8%. Sabendo disto, a próxima subseção é dedicada a discorrer acerca dos resultados quantitativos obtidos com a aplicação do SUS.

4.5.3 Resultados Quantitativos da Escala SUS

Uma vez aplicado os questionários junto aos professores-avaliadores, num período de 2 meses, puderam-se colher diversas informações e *insights* importantes acerca da usabilidade percebida. Como já especificado, a avaliação contou com a participação de 240 professores-avaliadores de uma população de 359, definida apta justamente por estar em atividade no Banco de Avaliadores do PNLD Avaliação.

Num primeiro momento, analisando o comportamento da plataforma diante do SUS, como plotado na Figura 22, obteve-se uma pontuação média de 73,2, o que pode ser lido como uma aplicação de software com classificação adjetiva “Bom” e que a coloca num nível de aceitabilidade satisfatório denominado “Aceitável”. De forma literal e partindo de uma visão imediatista, pressupõe-se que a plataforma não tenha enfrentado problemas significativos em termos de usabilidade percebida. Isso acontece, pois quando analisado quantitativamente e considerando que a pontuação mínima para que o sistema seja “aprovado” perante o SUS é um *benchmark* de 68,2 pontos, o sistema de PNLD Avaliação apresenta-se como aceitável em termos de usabilidade e aderência. Porém, ainda sob essa perspectiva visual da pontuação média quando localizada na escala de usabilidade, percebe-se que o sistema só não é classificado como marginal em termos de aceitabilidade por uma diferença de 3,2 pontos.

Figura 22 – Escala de Usabilidade do PNLD Avaliação frente ao SUS



Fonte: Elaborada pelo Autor a partir dos dados da pesquisa (2024)

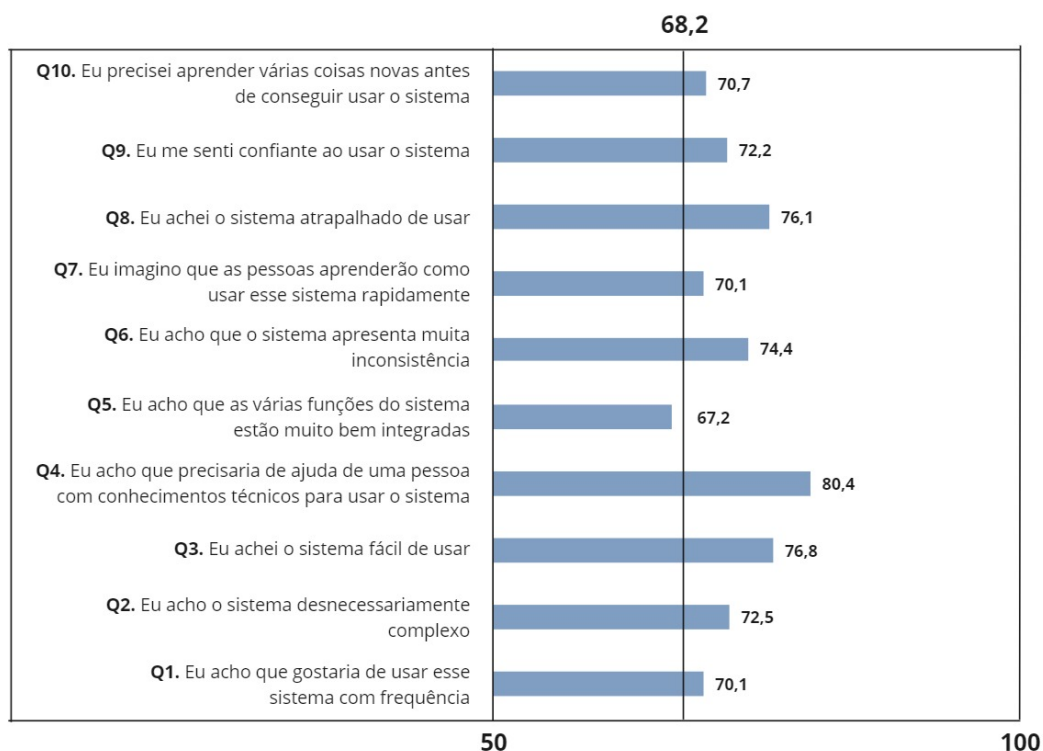
Ainda considerando a Figura 22, atentando-se especificamente à área de distribuição de frequência quanto à classificação adjetiva, somam-se 14,1% de avaliadores que consideraram o sistema “Ruim” ou “Pior imaginável”. Essa porcentagem diz das pontuações abaixo de 51 pontos considerando a escala SUS, que sinaliza a necessidade de priorização de investimentos dentro do

plano de negócio em design e usabilidade, para evitar maiores problemas futuramente. Outros 24,07%, apontam a marginalidade do sistema, classificando-o como “Ok”, que diz da necessidade de aderir a processos que insira e aproxime o usuário final nas atividades de desenvolvimento, além de se atentar ao contexto de uso ao qual o sistema é proposto. Em contraponto, 62,46% é a soma de professores-avaliadores que consideram o sistema “Bom”, “Excelente” ou “Melhor imaginável”, que classifica o sistema como aceitável em termos de usabilidade.

Considerando o nível de aceitabilidade, e entendendo que a marginalidade aponta para problemas estruturais que afetam a usabilidade e que requerem atenção, opta-se aqui, por considerar que 38,17%, a soma entre as avaliações que consideraram o sistema “inaceitável” e “marginal”, ou seja, mais de 1/3 da amostra utilizada, considera que o sistema apresenta problemas com usabilidade, e embora o SUS não apresente onde estão os problemas especificamente, tem-se uma sinalização em termos de usabilidade percebida por quem usa de fato o sistema.

Numa análise individual das questões do formulário SUS, conforme Figura 23, percebe-se como se comportam as percepções dos usuários. Quando considerado a pontuação média de 68 pontos, podemos observar que apenas a Questão 5, "Eu acho que as várias funções do sistema estão muito bem integradas", não atingiu a pontuação mínima considerável e está relacionada diretamente com a eficiência do sistema. A questão 1, que avalia se o usuário gostaria de usar o sistema com frequência, questão 2 que trata se as pessoas aprenderão como usar o sistema rapidamente e a questão 10, que aborda a necessidade de se aprender diversas coisas antes de conseguir o usar o sistema, embora atinjam a média positiva esperada, continuam muito próximas de uma pontuação marginal em termos de usabilidade.

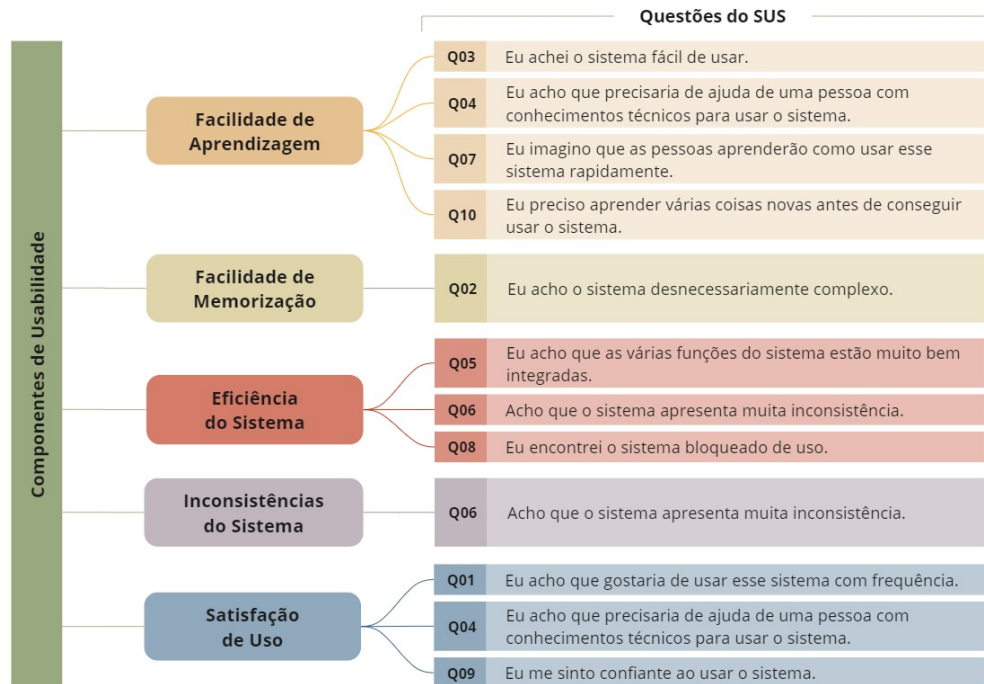
Figura 23 – Pontuação individual das questões SUS.



Fonte: Elaborada pelo Autor a partir dos dados da pesquisa (2024)

Por outro lado, buscando mais objetividade e clareza para a análise, pode-se correlacionar as questões do SUS com alguns dos componentes de qualidade indicados por Nielsen, discorridos conforme a Figura 24. A partir dessa correlação, a Figura 25 mostra a pontuação média de cada componente de qualidade frente à média do SUS, onde cada um deles se mostrou com uma pontuação positiva. Para uma análise mais detalhada, foram elaboradas análises de frequência por questão relacionada a cada componente de qualidade e são discorridas separadamente a seguir.

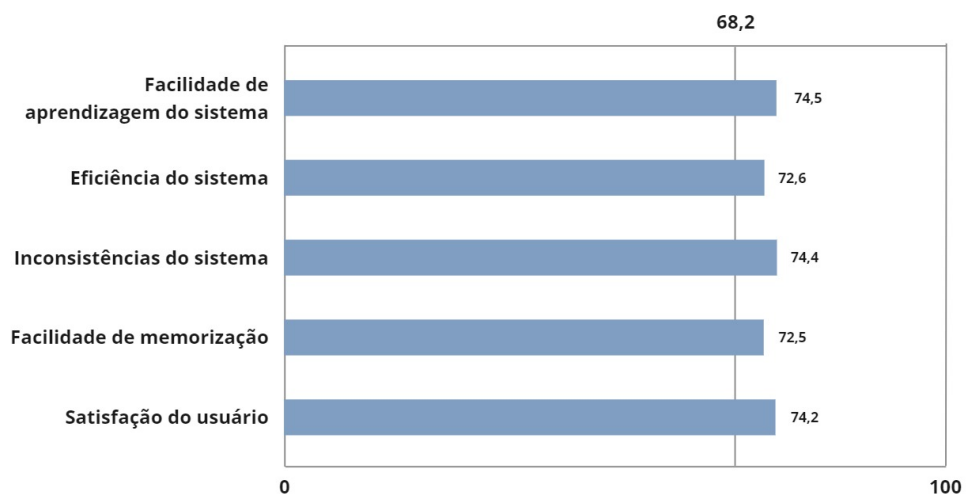
Figura 24 – Correlação entre as questões do SUS com os componentes de qualidade.



Fonte:

Fonte: Adaptado de [Tenório et al. 2010]

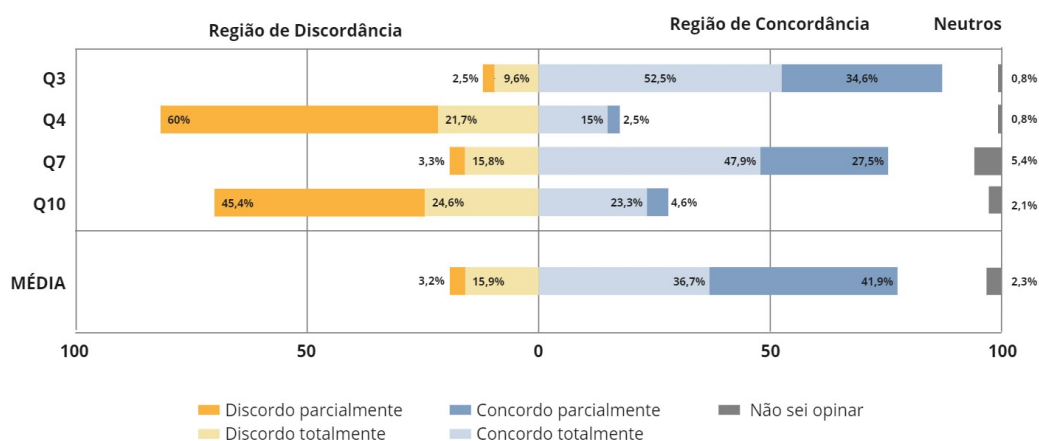
Figura 25 – Avaliação por componente de qualidade.



Fonte: Elaborada pelo Autor (2024)

Considerando a Facilidade de aprendizagem, a percepção dos professores-avaliadores está impressa na Figura 26. Quando perguntado se o usuário achou o sistema fácil de usar (Q3), percebe-se que mais 87% das respostas foram concordantes, e que somente 12,1% discordaram da facilidade de uso. Em relação à necessidade de ajuda de uma pessoa com conhecimentos técnicos para usar o sistema (Q4), 81,7% dos avaliadores discordaram da necessidade de ajuda técnica e, menos de 1/5 das respostas, concordaram com a necessidade de uma ajuda técnica para conseguir usar o sistema de forma proveitosa. Em termos de rapidez na aprendizagem (Q7), 75,4% das respostas foram consoantes a premissa de que as pessoas aprenderão como usar esse sistema rapidamente, enquanto quase 1/5 dos avaliadores discordam dessa afirmação, acreditando que novos usuários podem apresentar dificuldades para aprender usar o sistema. Por fim, quando questionado se era necessário aprender várias coisas novas antes de conseguir usar o sistema (Q10), 70% dos professores-avaliadores discordaram, enquanto 27,9% concordaram com a necessidade de aprender diversos processos para então conseguir usar o sistema. De modo geral, os números de professores neutros em relação às questões discorridas foram baixos e variaram entre 0,8% e 5,4%.

Figura 26 – Facilidade de aprendizagem a partir das questões do SUS.



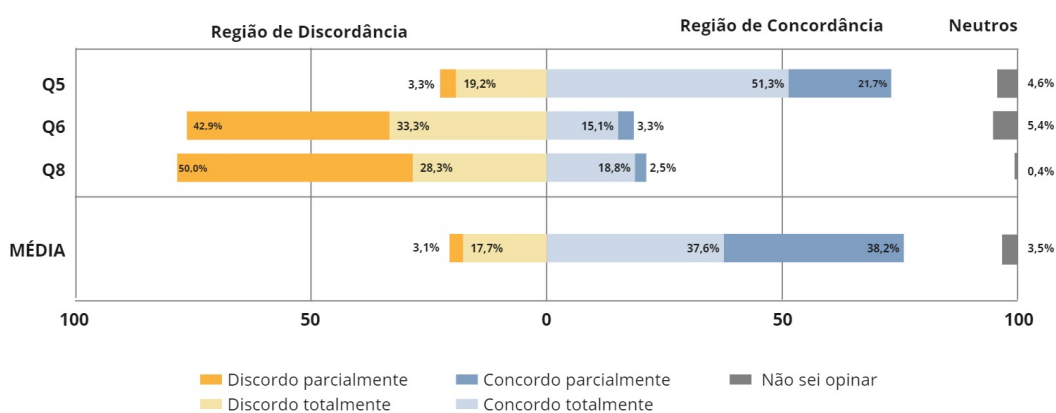
Fonte: Elaborada pelo Autor (2024)

Ainda considerando a Figura 26, na região em que estão impressos a distribuição percentual em relação à média geral, percebe-se que 78,6% dos professores-avaliadores consideram que o sistema apresenta um ótimo índice de facilidade de aprendizagem. Em contraponto, em média, 19,1%, cerca de 46 professores-avaliadores consideram que o sistema apresenta alguma

deficiência em termos de facilidade de aprendizagem. E por fim, somente 2,3% dos avaliadores não souberam opinar sobre essa questão especificamente.

Observando a Figura 27, percebem-se algumas ocorrências em termos de eficiência do sistema sob perspectiva dos professores avaliadores. Em termos de integração das várias funções do sistema(Q5), 73% dos professores-avaliadores, acreditam que sim, as funções estão muito bem integradas e, em contraponto, 19,2% não concorda completa ou parcialmente com a afirmação de que a solução compreende um sistema em que suas várias funções estão muito bem integradas. Quando questionados se o sistema apresenta muitas inconsistências (Q6), 76,2% dos professores-avaliadores que responderam ao questionário discordaram completa ou parcialmente, e 15,1% concordam que o sistema apresenta algum tipo de inconsistências. Considerando a afirmação “Eu achei o sistema atrapalhado de usar” (Q8), 78,3% dos avaliadores discordaram, enquanto 18,8% acredita que em algum momento o sistema é atrapalhado de usar. Em termos de neutralidade, os números não são tão expressivos e variaram entre 0,4% e 5,4%.

Figura 27 – Eficiência a partir das questões do SUS.



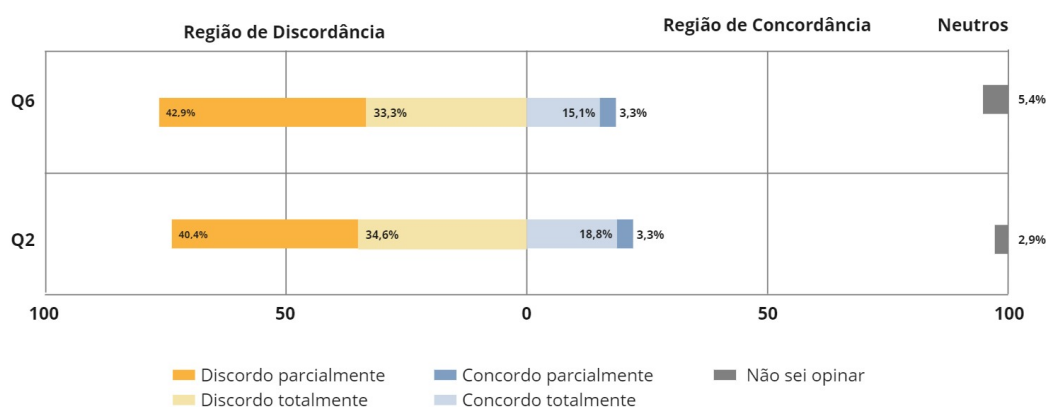
Fonte: Elaborada pelo Autor (2024)

Ainda analisando a região inferior da Figura 27, na região em que estão impressos a distribuição percentual e a média geral, aponta-se que em termos de eficiência 75,8% dos professores avaliadores considera de modo geral o sistema eficiente, enquanto 2,8%, cerca de 51 professores-avaliadores, compreendem que o sistema apresenta algum deficit em relação à eficiência. Em média, 3,5% dos professores avaliadores não souberam opinar sobre a questão.

Conforme Figura 28, são apresentados os resultados acerca da percepção dos professores avaliadores sobre as inconsistências do sistema e facilidade de memorização. Quando avaliaram se o sistema apresenta muita inconsistência (Q6), 76,2% se mostraram discordantes parcial ou

totalmente, enquanto 18,4% concordaram que o sistema apresenta algum nível de inconsistência sem sua estrutura visual e/ou funcional, e somente 5,4% não souberam opinar. Quando analisada a questão Q2 que questiona se o sistema é desnecessariamente complexo, 75% dos avaliadores discordaram, enquanto 22,1% acredita que o sistema lida com algum grau de complexidade que afeta a fluidez de uso e, por fim, somente 2,9% não soube opinar acerca da complexidade do sistema.

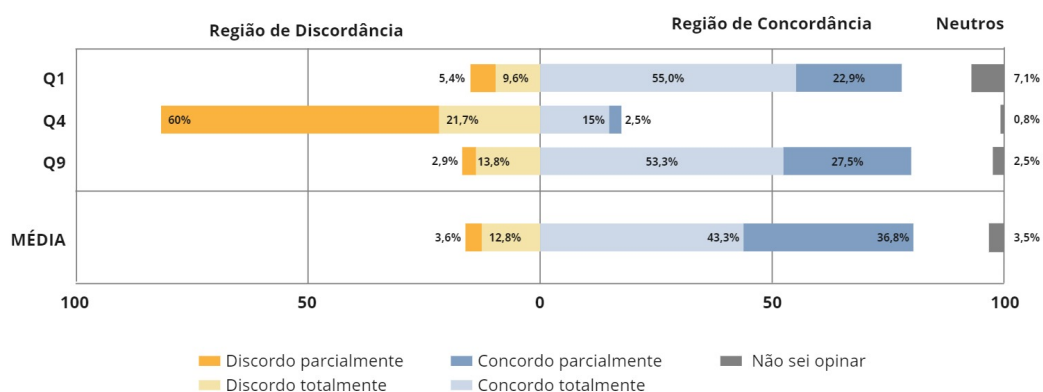
Figura 28 – Inconsistências e Facilidade de memorização.



Fonte: Elaborada pelo Autor (2024)

Na Figura 29, são apresentados os resultados acerca da percepção dos professores avaliadores sobre a satisfação de uso. Quando avaliaram se gostariam de usar o sistema com frequência (Q1), 77,9% se mostraram concordantes parcial ou totalmente, enquanto somente 9,6% dos professores-avaliadores discordaram da afirmação. Considerando a necessidade de ajuda de uma pessoa com conhecimentos técnicos para usar o sistema (Q4), 81,7% dos avaliadores discordaram dessa necessidade e somente 15% concordaram com a ideia de que precisaria de ajuda de uma pessoa com conhecimentos técnicos para usar o sistema de forma proveitosa. Analisando a confiança ao usar o sistema (Q9), 80,8% dos professores-avaliadores afirmam parcial ou completamente que se sentem confiantes enquanto usam, enquanto 13,8% se mostraram parcial ou completamente não confiantes ao usar o sistema. Dos professores avaliadores, registrou-se que 0,8% a 7,1%, se mostraram neutros e não souberam opinar sobre o nível de confiança ao usar.

Figura 29 – Satisfação a partir das questões do SUS.



Fonte: Elaborada pelo Autor (2024)

A partir da Figura 29, ainda é possível perceber a partir da média geral, aponta-se que em termos de satisfação de uso 80,1% dos professores-avaliadores consideraram que o sistema é satisfatório de usar, enquanto somente 16,4%, cerca de 40 professores-avaliadores, compreendem que o sistema apresenta algum ponto negativo e que acarreta a má satisfação de uso. Em média, 3,5% dos professores avaliadores não souberam opinar sobre a questão.

4.5.4 Análise Temática Dedutiva das questões abertas do formulário

A análise temática aplicada neste estudo apoiou-se na literatura, considerando os aspectos que permeiam a qualidade do software, considerando um conjunto de atributos relacionados ao esforço necessário para uso da plataforma computacional que apoia a Avaliação Pedagógica do PNLD. A fim de entender os processos de efeito-causa acerca dos problemas de usabilidade percebidos na plataforma, os relatos de usuários finais foram analisados dedutivamente a partir dos cinco atributos que formatam a usabilidade enquanto critério de qualidade. Em seguida os resultados preliminares foram analisados criticamente, para entender o impacto que a complexidade do sistema e as nuances no ciclo de desenvolvimento, como envolvimento do usuário desde a concepção e a comunicação entre as partes interessadas, têm sobre a usabilidade do sistema e consequentemente sobre sua eficácia.

No total, embora a pesquisa tenha envolvido 241 participantes para a aplicação do SUS, todos foram voluntariamente convidados a compartilhar suas experiências e percepções como usuários finais da plataforma de avaliação pedagógica do PNLD. Recebemos relatos de 141 participantes de forma voluntária. Na análise temática, todos os relatos foram considerados e exa-

minados com base nos cinco componentes de qualidade: facilidade de aprendizado, memorização, eficiência do sistema, erros e inconsistências, e satisfação do usuário. Para exemplificar como os agrupamentos temáticos foram abordados, 30 relatos foram selecionados para exemplificar como eles apresentaram conformidade com os componentes de qualidade do estudo. A seguir, apresentamos as percepções desses usuários finais sobre a plataforma em questão.

- **Facilidade de aprendizagem:**

Um tema bastante recorrente nos relatos dos usuários finais abrange as nuances acerca da aprendizagem que, embora apontadas sob percepções mistas, ainda evidenciam as dificuldades de se conseguir compreender o sistema a ponto de usá-lo de forma fluida desde o primeiro contato. Muitos relatam que o sistema é amigável e intuitivo desde o início, como evidenciado por P82: "A plataforma é bastante amigável, não enfrentei nenhuma dificuldade ao usá-la", enquanto outros usuários indicam que o sistema só se torna fácil de usar após orientação, como mencionado pelo P9: "Não é intuitivo, porém depois de alguém indicar os caminhos é tranquilo de usar", mas mesmo após treinamento ainda sim se tem algumas dificuldades de executar suas atividades no sistema, o que corrobora com a necessidade de um treinamento mais robusto, principalmente quando se é novo usuário, como apontado pelo P28: "para os iniciantes, é preciso da ajuda dos tutoriais, disponibilizados pela equipe responsável, para conseguir utilizar os recursos do sistema." e P67: "Como sou avaliadora iniciante do Processo de Avaliação... Que fosse assegurado ao avaliador, antes do acesso ao Manual do Avaliador, Livros do Estudante e do Professor, digital ou não, esse treinamento."

A complexidade na compreensão das funcionalidades e a necessidade de orientação inicial indicam que a facilidade de aprendizado não é completamente atendida, especialmente nas primeiras interações, apontando que a falta de intuitividade nas primeiras interações foram barreiras significativas. Por exemplo, P165: relatou dificuldades em entender a divisão por obras e avaliadores ("Demorei para compreender a divisão por obras e avaliadores"), enquanto P170 mencionou as múltiplas pastas e subpastas que não eram intuitivas, o que o fez ter "bastante dificuldades em realizar minha primeira avaliação". Esses relatos sugerem que o sistema não é imediatamente fácil de aprender e que uma melhor introdução ou tutoriais poderiam facilitar o processo de aprendizagem. Esse feedback destaca a importância de um suporte inicial adequado, que não apenas explique as funcionalidades do sistema, mas também permita aos novos usuários praticarem antes de se envolverem em tarefas reais. Implementar sessões de treinamento mais práticas e detalhadas pode reduzir a curva de aprendizagem e aumentar a confiança dos novos

usuários.

- **Memorabilidade:**

A facilidade e/ou dificuldade para lembrar os componentes do sistema também foi bastante citado pelos usuários, indicando dificuldades na memorização e na recuperação de informações necessárias, destacando a necessidade de relembrar constantemente os procedimentos para acessar certas funcionalidades. Essas percepções correspondem a relatos como o do P12: "Como avaliadora não há acesso fácil ao módulo de falhas pontuais discriminadas... é preciso fazer download para conferência." e P60: "A página não é intuitiva. É necessário ficar clicando para saber onde estão as informações e corre-se o risco de não encontrá-las". A necessidade de baixar informações para conferência e a complexidade em encontrar dados específicos dentro do sistema sugerem que a interface não é intuitiva o suficiente para facilitar a retenção de procedimentos. Essas dificuldades são amenizadas conforme a experiência aumenta, como apontado pelo P118 que diz que "O Sistema parece complexo no início. Porém, passada essa primeira impressão, é possível desenvolver o trabalho." e reforçado por relatos como o do P235: "Há alguns problemas de usabilidade, que vamos 'dominando' com mais precisão ao longo do uso cotidiano" e P83: "Como em todo início de processo avaliativo, faz-se necessário um tempo para se familiarizar com o Sistema. Depois, torna-se tranquilo seu uso."

Para a memorabilidade, um fato crucial é entender qual o tempo dedicado para usar a plataforma e o nível de familiaridade com plataformas digitais, o que distingue usuários assíduos dos esporádicos e como isso afeta suas experiências ao usar o sistema. O relato do P134 é importante nesse sentido, pois ela aponta que "O sistema é pouco amigável para quem pouco lida com sistemas do tipo", indicando que, enquanto o sistema é memorável para muitos, uma interface mais intuitiva e suporte contínuo podem beneficiar usuários menos experientes e que não o utilizam constantemente. A melhoria na organização das informações e a simplificação das etapas necessárias para acessar funcionalidades críticas poderiam aliviar esse problema. Uma interface mais limpa e organizada, possivelmente com a inclusão de uma função de busca eficiente, pode melhorar significativamente a facilidade de memorização.

- **Eficiência do sistema:**

A eficiência do sistema é uma das áreas mais críticas quando analisados os relatos dos usuários, embora haja boas avaliações acerca desse aspecto. O P91, por exemplo, destacou a produtividade alcançável: "Quando se entende o que se deve fazer até que se vai mais rápido na

avaliação. Ah, o sistema também não trava. Isso é muito bom”, reforçado pelo relato do P110 que afirma que "Nosso tempo de trabalho nele é produtivo". O P127 evidencia a interface do sistema e o considera “esteticamente bem acabado” e que isso influi no fato de que “o sistema em si é muito bom, fácil de usar, intuitivo e eficiente”. Os P91, P110 e P127, destacam ainda a ausência de travamentos, refletida pelo bom design e infraestrutura estável, essenciais para a eficiência operacional.

Por outro lado, a perda de dados durante a execução de atividades, relatada por diversos participantes como o P47 que, embora enfatize os avanços, diz que problemas sérios ocorrem, como “...quando estamos lançando uma resposta e apaga ou não salva, ou mesmo após ter salvo as ocorrências, elas somem inexplicavelmente”. Isto evidencia um problema crítico que afeta a confiança do usuário e consequentemente a eficiência da plataforma, corroborando com comentários como o do P4 que aponta que é “...necessário um retrabalho de copiar e colar a resposta” para então ela ficar salva. O usuário P217 critica a ausência de artefatos que apoiem a visibilidade do status do sistema quando diz que “a plataforma seria mais amigável se alertasse para os procedimentos, antes do preenchimento.” Esses problemas têm indicado que o sistema não é otimizado para permitir um alto nível de produtividade.

Outros aspectos que têm afetado a eficiência do sistema, são os processos redundantes e a necessidade de muitos cliques para realizar tarefas simples. O P189 exemplifica: “Um ponto que acho que poderia melhorar é o trânsito entre os itens. . . se estou respondendo ao item 1.2 e quero ir ao 1.10, tenho de subir a barra de rolagem, clicar na flecha lateral do menu de itens até chegar no local onde está o item 1.10. São muitos movimentos”. A falta de funcionalidade direta e o tempo gasto em tarefas repetitivas têm afetado negativamente a produtividade dos usuários, e eles mesmos apontam para possíveis soluções, como a proposta do P256: “Precisávamos clicar em muitos botões até chegar a uma resposta. Talvez um sumário na lateral das seções pudesse ajudar”. O P189 relata, também, que para “sair de um volume ao outro, é preciso reiniciar o bloco”, isso causa certa lentidão no uso da ferramenta e reduz a taxa de produtividade do usuário.

Estes pontos sugerem que o sistema, embora funcional em muitas áreas, ainda tem processos que não são otimizados. Algumas ações são reivindicações claras dos próprios usuários a partir de seus relatos, considerando suas experiências, elas são: é urgente repensar a navegabilidade do sistema a partir da redução no número de cliques necessários para completar tarefas; considerar o sistema de salvamento automático é um dos fatores mais críticos diante dos relatos e aparece repetidas vezes como interferência no uso da plataforma; e por fim, garantir a estabilidade do sistema como requisito crucial para efetividade da plataforma. Estas são medidas

apontadas pelos usuários e que poderiam aumentar substancialmente a eficiência operacional dos fluxos que compõem a avaliação pedagógica do PNLD.

- **Erros e inconsistências:**

Os usuários relataram diversos problemas de inconsistência, como falhas no salvamento de dados, que tem sido um fator de bastante impacto na eficiência do sistema, como mencionado no tópico anterior, e outras instabilidades que resultam na perda de informações. Participantes como P17 relata que a instabilidade da plataforma não lhes dá confiança suficiente para usá-la diretamente para executar suas atividades e “geralmente faço em editor de texto para depois inserir na plataforma” o que é reforçado no relato do P40 que diante da instabilidade é “preciso fazer login várias vezes, a cada queda do sistema perde-se dados já inseridos.” O usuário P163 compartilhou uma experiência frustrante de perda de dados: “Também, por duas vezes, tive um bloco inteiro de respostas perdidas.” Esses fatores acabam obrigando-os a adotar métodos alternativos, como usar editores de texto, o que afeta a aderência ao sistema.

Diante desses relatos, percebe-se que a baixa taxa de erros esperada não está sendo atendida, indicando a necessidade de melhorias na estabilidade e consistência do sistema, pois essas problemáticas são recorrentes, têm minado a confiança dos usuários e implicam diretamente em atrasos no trabalho. Para abordar esses problemas, é fundamental realizar uma revisão técnica completa e implementar testes rigorosos de qualidade, para identificar e corrigir inconsistências antes que elas impactem os usuários finais. Ressalta-se, mais uma vez, que é crucial implementar mecanismos robustos de salvamento e recuperação de dados, para proporcionar uma experiência mais tranquila, reduzindo erros e aumentando a satisfação dos usuários.

- **Satisfação de uso:**

A satisfação de uso variou entre os participantes. Enquanto alguns, como P11, elogiaram a modernidade e a redução de problemas em comparação com sistemas anteriores, afirmando que “O sistema é bastante intuitivo e seu design moderno”, outros, como P26 e o P243 se mostram não satisfeitos e enfrentam inconveniente, seja processo de verificação para entrar no sistema ou pelo “formato de baixar os volumes não ficou bom e houve muita instabilidade no sistema”, respectivamente. Muitos dos usuários apontam para uma experiência positiva e acharam o sistema fácil de usar e adequado para os objetivos de avaliação, porém essa impressão se dá em primeira instância, comparando o atual sistema às plataformas anteriores, como evidencia o P110 “Nem consigo compará-lo com o anterior (Simec) em questão de inconsistência”, o que pode render

interpretações, pois o questionamento a ser feito é: o sistema é aderente e supre as necessidades dos usuários ou ele é só melhor que o anterior?

Pois, diante disso, alguns usuários mencionam aspectos que afetam suas experiências. O usuário P191, por exemplo, aponta que “ao terminar de escrever uma justificativa, o avaliador precisa digitar espaço antes de acionar o botão de salvar... isso atrasa alguns momentos da navegação”. Já o P163 relata a necessidade de ajustes para tornar o sistema mais eficiente e diz como a perda de dados afetou sua experiência: “por duas vezes, tive um bloco inteiro de respostas perdidas. Preenchi e salvei todas as respostas, aparecia a mensagem de "salvo", no entanto, de repente sumiu tudo”. É importante considerar que a satisfação de uso é produto de uma coleção de outros fatores já mencionados que influenciam diretamente, como a facilidade de uso, a eficiência e a confiabilidade, e muitos dos participantes mostram seus anseios e inseguranças em relação ao sistema justamente a partir da percepção agregada desses aspectos. No geral, essa variação de percepção indica que, embora haja melhorias perceptíveis, há uma necessidade contínua de ajustes para alcançar uma satisfação mais uniforme. Ouvir os usuários, mediante feedback contínuo e processos de melhoria iterativa, pode ajudar a alinhar melhor o sistema com as expectativas e necessidades dos seus usuários.

4.5.5 Resultados e Discussões

Garantir que um produto de software atenda às necessidades e expectativas dos usuários de forma eficaz e satisfatória é um dos maiores desafios da atualidade e a avaliação de usabilidade de um software interativo é um dos pilares fundamentais. Sabe-se que, ao analisar a usabilidade, os especialistas conseguem identificar pontos de fricção na interação, fluxos de trabalho complexos e/ou funcionalidades confusas e não condizentes com o ambiente de uso, permitindo otimização contínua e melhorias antes mesmo do lançamento. A inserção do usuário final no ciclo de vida do software é outro pilar bastante importante no percurso de desenvolvimento, uma vez que há apropriação de mais um recurso para garantia de qualidade e a relevância do produto final. A partir disso, este estudo se propôs a investigar como é a usabilidade de um sistema real, dinâmico e com múltiplos stakeholders.

Pensando essas características e estudando o PNLD Avaliação como uma instância que compreende esses aspectos, percebeu-se que a aplicação de software desde lançado não tinha passado por uma avaliação de usabilidade considerando o usuário final e, por tanto, foi considerado a própria avaliação paralela a uma inspeção em artefatos de software gerados no ciclo de vida, como primordial para compreender o cenário. Contudo, esta seção se restringiu

a resultados iniciais e quantitativos obtidos com a avaliação de usabilidade junto aos usuários finais do PNLD Avaliação.

Embora os resultados expressem uma facilidade e satisfação inicialmente experimentadas pelos usuários, é necessário compreender que a pontuação que o sistema alcançou frente ao SUS se encontra numa região ainda muito próxima da não aceitabilidade em termos de usabilidade. E que nesse mesmo aspecto o sistema se apresenta como “Inaceitável” ou “Marginal” para quase 40% dos participantes da pesquisa e que para eles, o software pode ser lido como “pior imaginável”, “ruim” ou “ok”, sendo que este último não deve ser interpretado como positivo, uma vez que ele é um limiar entre uma avaliação boa e ruim, como se um “meio-termo”.

Conforme o levantamento do referencial teórico, o mapeamento da literatura e os resultados do teste de usabilidade aplicado, pode-se ter alguns apontamentos, uma vez que o método utilizado ele não é estritamente conclusivo e seu objetivo não é apresentar as raízes do problema, mas sim apontar para alguns possíveis pontos que requerem atenção. Esses pontos incluem:

- Há sinalizações para a necessidade de priorização de investimentos dentro do plano de negócio em design e usabilidade, para evitar maiores problemas futuramente e a necessidade de aderir a processos que insira e aproxime o usuário final nas atividades de desenvolvimento, além de se atentar ao contexto de uso ao qual o sistema é proposto.
- Numa análise minuciosa, percebe-se que há apontamentos para o não funcionamento da integração das várias funções do sistema, o que implica negativamente na sua eficiência. Assim, faz-se necessário maior aprofundamento para entender, mapear e minimizar redundâncias e inconsistências, para facilitar a comunicação entre os componentes do sistema, resultando em uma experiência mais coesa para o usuário final.
- A necessidade de atenção recai também sobre o usuário gostar de usar e querer usar o sistema com frequência, pois esses aspectos dizem do sucesso e grau de adoção do PNLD Avaliação. Esses fatores afetam diretamente a produtividade e consequentemente a adesão ou não desse sistema pelos usuários para execução das atividades de forma contínua, consistente e eficiente.
- Outro fator é a fluidez para aprendizagem, que precisa considerar a baixa ou inexistente necessidade de se aprender diversas coisas antes de conseguir usar o sistema de forma plena. Aprofundar essa problemática é importante para reduzir a frustração do usuário, para assim aumentar a satisfação geral e promover uma relação positiva entre o usuário e o sistema, o que, perante a avaliação, o PNLD Avaliação apresentou alguns *déficits*.

Embora o resultado geral obtido através do método SUS esteja 5 pontos acima da média necessária para considerá-lo aceitável, as frequências, quando distribuídas, apontam para algumas questões importantes que podem ser lidas, como a existência de problemas que impedem uma maior e melhor utilização pelos usuários. Todos esses apontamentos recaem nos componentes de qualidade, sendo eles a Facilidade de aprendizagem, eficiência do sistema, inconsistência do sistema, facilidade de memorização e a Satisfação de uso. Essas problemáticas podem ser revertidas a partir de algumas ações estratégicas no ciclo de vida, principalmente na etapa de manutenção, considerando que a plataforma já está em uso.

Considerando todos os autos e limitações conclusivas do método e que esses resultados imprimem os extratos de um único estágio, foi, então, utilizado, o método de análise temática dedutiva para aprofundamento nas questões apontadas e complementar os resultados, para melhor entendê-los.

Um dos aspectos a serem considerados é que a complexidade do contexto e domínio de aplicação, pede que haja um ciclo de vida caracterizado pela sua capacidade de adaptação. A flexibilidade para lidar com as mudanças contínuas no contexto estudado, pede que haja um ciclo dinâmico que promova comunicação adequada e fluida entre as partes interessadas e que seja suscetível à interação. A partir disso, percebe-se que as falhas apontadas pelos usuários ocorrem justamente pela ausência desses fatores na tomada de decisão, apontando para a impressão de que as fases de design, desenvolvimento e teste têm sido realizadas de forma isolada, e sem representação adequada/satisfatória das partes interessadas.

Umas das percepções mais frequentes permeia o envolvimento do usuário final no ciclo de desenvolvimento, retirando-o do papel de reacionário e considerando suas expectativas desde a concepção. Isso não significa que tudo o que o usuário quer deva ser considerado, mas garantir que o que ele precisa para executar suas atividades de forma satisfatória dentro do sistema, seja pensado em conjunto e, então, implementado. No contexto estudado, pode-se significar que o feedback dos usuários sobre dificuldades de aprendizagem pode ter chegado tarde demais, ou não ter sido considerado adequadamente durante o desenvolvimento. Esses fatores têm levado a um sistema que não foi projetado com a usabilidade em mente, apresentando uma interface complexa ou pouco intuitiva que os usuários acham difícil de entender e navegar desde o início.

A ausência de iteração ou sua adoção de maneira não suficiente, também pode ter sido prejudicial, pois implicou na ausência de comunicação contínua entre as partes interessadas. Pois, sem ciclos curtos de feedback e ajustes, a equipe de desenvolvimento pode não ter percebido ou ter tido canais suficientes para tratar com a equipe de design e teste, e então, as dificuldades

dos usuários em utilizar o sistema em seu dia a dia ou após períodos de inatividade, não foram de fato consideradas. Esses aspectos, apontam para uma interface que exige re-aprendizagem frequente, com menus ou funcionalidades mal organizados, aumentando a frustração dos usuários e reduzindo a eficiência.

Nesse sentido, alguns dos relatos levam ao fato de que foi entregue uma plataforma computacional que, embora funcional, não está otimizada para a produtividade, pois não promove a potencialização dos fluxos existentes e, pelo contrário, se depara com redundâncias críticas e frequentes. Os problemas relatados, como as redundâncias, cliques desnecessários e as dificuldades com a navegação, podem ser problemas que surgem quando não há um ciclo de desenvolvimento que permita ajustes rápidos e incrementais baseados no uso real do sistema pelos usuários. Além disso, problemas de performance podem não ser detectados e corrigidos em tempo hábil, principalmente quando se trata de um contexto altamente dinâmico e com alta volatilidade de requisitos e de pessoal, o que compromete a experiência do usuário, justamente porque a detecção e correção de erros podem ser muito mais lentas. Nesse sentido, é necessário considerar a iteração e automação de testes, para reduzir ocorrências de erros e inconsistências que impactam negativamente a confiança dos usuários.

Um fato relevante e que, como já dito anteriormente, faz toda diferença quando se faz uma avaliação de usabilidade é compreender o ponto de partida de quem está avaliando. Muitos dos usuários mencionam em seus relatos como a plataforma atual é melhor que a anterior, e a plataforma anterior é inválida como métrica considerada. Embora seja percebido que as evoluções estão ocorrendo e são percebidas pelos usuários, ainda é frequente as reivindicações para se manter um ciclo constante de melhorias, o que significa que o sistema pode não estar atendendo às necessidades evolutivas dos usuários e do próprio ciclo de atividades. Ou seja, a falta de uma cultura de feedback dos usuários, pode estar resultando em um sistema que, apesar de ser funcional, não está proporcionando uma experiência agradável, uma vez que há tantos relatos que apontam para a baixa satisfação e frustrações coletivas.

4.6 CONSOLIDAÇÃO DOS RESULTADOS

A investigação conduzida ao longo de três estágios distintos revelou insights valiosos sobre os desafios e as oportunidades relacionados à gestão de requisitos, usabilidade e gestão de ambientes dinâmicos com múltiplos stakeholders em um sistema computacional. A centralização de stakeholders, a análise de usabilidade e a integração de equipes destacam os impactos de diferentes práticas no desenvolvimento e na operação de sistemas interativos, ressaltando a

importância de abordagens robustas e iterativas na engenharia de software.

No primeiro estágio, a análise das histórias de usuário evidenciou que a volatilidade dos requisitos foi um dos principais entraves no início do projeto. A presença de quatro stakeholders distintos, com interesses variados, resultou em uma alta taxa de alterações e descontinuações. A centralização em dois stakeholders (GI1 e P2) representou um marco positivo, reduzindo a volatilidade e trazendo maior eficiência na finalização de histórias de usuário. Essa mudança gerencial não apenas estabilizou o processo de engenharia de requisitos, mas também facilitou a manutenção do sistema, contribuindo para a satisfação dos usuários. Testes estatísticos corroboraram esses resultados, indicando uma diferença significativa entre as configurações antes e depois da centralização.

O segundo estágio concentrou-se na avaliação de usabilidade de um sistema dinâmico com múltiplos stakeholders, destacando a necessidade de um planejamento mais centrado no usuário final. Embora o sistema tenha alcançado uma pontuação SUS aceitável, quase 40% dos participantes expressaram insatisfação com aspectos fundamentais, como facilidade de aprendizado e eficiência. Problemas estruturais, como integração inadequada entre funções e redundâncias no fluxo de trabalho, foram apontados como barreiras à produtividade e à adoção do sistema. As análises destacaram que a falta de iteração contínua e de feedback durante o desenvolvimento resultou em uma interface complexa, que exige re-aprendizagem frequente e gera frustração entre os usuários.

No terceiro estágio, a análise temática das entrevistas revelou desafios mais amplos relacionados à gestão de um ambiente com múltiplos stakeholders. A lacuna de comunicação entre equipes e a volatilidade dos requisitos foram agravadas pela alta rotatividade de pessoal, expondo a necessidade de estratégias mais eficazes de gestão de mudanças e documentação contínua. Metodologias ágeis, embora benéficas em termos de flexibilidade, demonstraram limitações em contextos de alta complexidade quando não acompanhadas por documentação clara e fluxos de trabalho bem definidos. A sobrecarga das equipes e a ausência de um mediador formal para alinhar requisitos e prioridades comprometeram a clareza e a agilidade na resolução de problemas.

Ao consolidar os resultados, torna-se evidente que as práticas de gestão centralizada e a implementação de mudanças organizacionais contribuíram para melhorias significativas na comunicação e na estabilidade do sistema. A alta taxa de resolução de *tickets*, observada após a centralização, ilustra os benefícios de um modelo de gestão mais focado e alinhado. Além disso, a introdução de testes automatizados emergiu como uma estratégia promissora para aumentar

a qualidade e a eficiência do sistema, mitigando problemas relacionados a inconsistências e atrasos.

Apesar dos avanços, a investigação destaca lacunas críticas que exigem atenção contínua. A falta de integração entre as fases de design, desenvolvimento e teste, bem como a ausência de uma cultura de feedback iterativo, impactaram negativamente a experiência do usuário. Isso reforça a importância de um ciclo de vida de software adaptativo, que promova comunicação fluida e envolva todas as partes interessadas desde a concepção até a manutenção.

Os relatos dos usuários sobre dificuldades de aprendizado e insatisfação com a interface sugerem que a interação isolada entre equipes pode ter prejudicado a qualidade do produto final. A necessidade de priorizar investimentos em design e usabilidade é um ponto central para evitar maiores problemas no futuro e garantir um sistema que atenda às necessidades reais dos usuários. Além disso, uma estratégia proativa para envolver usuários no desenvolvimento pode potencializar a adoção e melhorar a percepção geral do sistema.

Em suma, os três estágios da investigação convergem para uma conclusão crítica: sistemas interativos projetados para ambientes dinâmicos e com múltiplos stakeholders requerem abordagens integradas e iterativas. A centralização de stakeholders e a priorização de práticas de usabilidade são elementos-chave para alcançar um equilíbrio entre funcionalidade e satisfação do usuário. No entanto, a evolução contínua do sistema e a adoção de uma cultura de aprendizado e feedback iterativo permanecem como pilares indispensáveis para o sucesso a longo prazo. Considerando estes aspectos e insights valiosos, o próximo capítulo propõe um conjunto de boas práticas para o desenvolvimento de software que operam em ambientes caracterizados como dinâmicos e com múltiplos stakeholders, além de uma aplicação direta dessas boas práticas em um processo de desenvolvimento.

5 BOAS PRÁTICAS PARA O DESENVOLVIMENTO DE SOFTWARES QUE OPERAM EM AMBIENTES DINÂMICOS E COM MÚLTIPLOS STAKEHOLDERS

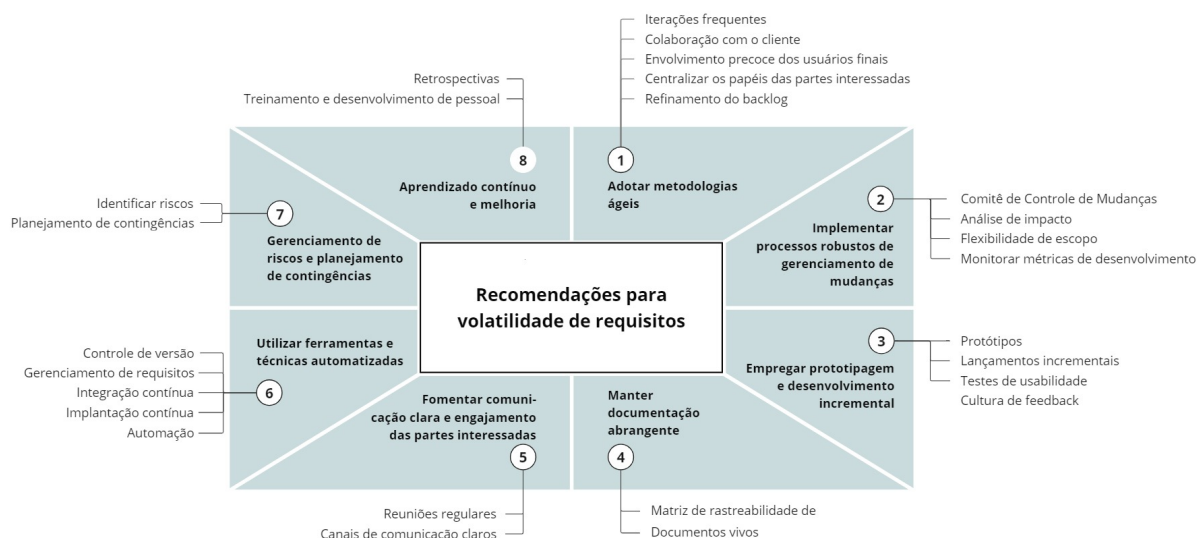
Este capítulo apresenta as contribuições desta dissertação, considerando os resultados das análises anteriores. A primeira contribuição envolve a proposição de um conjunto de boas práticas para apoiar a gestão de projetos de desenvolvimento de software que operam sob a complexidade de ambientes dinâmicos com múltiplos stakeholders. Além disso, foi definido um processo de desenvolvimento ágil, levando em conta as nuances das melhores práticas estabelecidas, para promover melhorias e robustez nos fluxos de trabalho em ambientes semelhantes aos já caracterizados.

As diretrizes apresentadas não propõem um novo método de desenvolvimento de software; em vez disso, fornecem estratégias valiosas para aplicar, de forma eficaz, práticas consolidadas dentro do contexto específico de ambientes dinâmicos e com múltiplos stakeholders.

5.1 BOAS PRÁTICAS PARA LIDAR COM A VOLATILIDADE DE REQUISITOS EM AMBIENTES DINÂMICOS E COM MÚLTIPLOS STAKEHOLDERS

Considerando todos os resultados obtidos com o percurso metodológico aplicado em um cenário real de software que atua em ambiente dinâmico e com múltiplos stakeholders, foram definidas algumas boas práticas baseadas nas recomendações do SWEBOK [Bourque e Fairley 2014]. Essas boas práticas foram projetadas para auxiliar na gestão de projetos de desenvolvimento de software em ambientes complexos e dinâmicos com múltiplos stakeholders. Essas recomendações são fundamentadas em dados quantitativos e qualitativos. Ao usar o SWEBOK como base para essa proposta, garante-se que a abordagem está alinhada com padrões internacionalmente reconhecidos e com um conhecimento abrangente. Isso aumenta a eficácia e a confiabilidade na gestão de projetos de desenvolvimento de software em ambientes dinâmicos e complexos. A Figura 30 apresenta a proposta de oito melhores práticas e atividades para lidar com os desafios de requisitos voláteis e ambientes com múltiplos stakeholders. A definição de cada boa prática é fornecida abaixo.

Figura 30 – Recomendações práticas para lidar com a volatilidade de requisitos.



Fonte: Elaborada pelo Autor (2024)

• BP1: Adotar Metodologias Ágeis

Como uma boa prática, a adoção de metodologias ágeis considera aspectos fundamentais para lidar com as nuances de ambientes dinâmicos com múltiplos stakeholders, onde há alta volatilidade de requisitos e a necessidade de comunicação estruturada e clara. Um dos insights principais do nosso estudo é o impacto positivo da redução do número de stakeholders envolvidos no processo de engenharia de requisitos. **Mapear e centralizar as funções dos stakeholders**, combinados com responsabilidades claras, reduz significativamente atrasos na comunicação, melhora o entendimento e promove melhor alinhamento com os objetivos do projeto.

Com a centralização das funções, a **colaboração com o cliente torna-se mais eficiente** devido à menor distância, eliminando a necessidade de um agente intermediário entre o cliente e as equipes de requisitos, desenvolvimento e manutenção do software. Ter um *stakeholder* principal responsável pela elicitação e desenvolvimento de requisitos torna a comunicação mais direta, minimizando a probabilidade de requisitos conflitantes. Recomendamos, portanto, designar uma única entidade ou uma pequena equipe bem coordenada para gerenciar a elicitação e o desenvolvimento dos requisitos, com autoridade e responsabilidade claras para tomar decisões e garantir o alinhamento com os objetivos do projeto.

Outros aspectos importantes incluem a **frequência de iterações** e o **gerenciamento do backlog**. Ambas as atividades permitem rápida adaptação às mudanças ao apoiar o acompa-

mento do progresso, ajustes de prioridades e a adição de novos requisitos. O *backlog* deve refletir as necessidades atuais e prioritárias dos stakeholders, pois o refinamento contínuo é essencial para identificar rapidamente as mudanças necessárias para manter o alinhamento com o plano de desenvolvimento.

A análise temática indicou uma lacuna no **envolvimento dos usuários finais nas etapas iniciais do projeto**. A contribuição desses usuários é crucial para garantir que as funcionalidades desenvolvidas atendam às suas necessidades e melhorem a satisfação geral. Assim, sugerimos envolver os usuários finais desde o início do projeto e engajá-los em ciclos regulares de feedback, por meio de sessões de teste, pesquisas e validação direta de requisitos.

- **BP2: Implementar Processos Robustos de Gerenciamento de Mudanças**

Dado o caráter dinâmico do contexto e sua influência no desenvolvimento de soluções computacionais, o escopo do projeto deve permanecer adaptável para acomodar continuamente o dinamismo, garantindo revisões e adaptações constantes em resposta a novas informações, feedback de stakeholders e prioridades em mudança.

Em relação a essa **flexibilidade do escopo**, é necessária uma interface centralizada para facilitar o acompanhamento e a gestão das mudanças ocorridas ou que possam ocorrer no projeto. Essa interface deve oferecer uma visão holística do status das mudanças propostas, incluindo aquelas aprovadas, implementadas e em andamento. Um **painel de controle de mudanças** promove comunicação clara e atualizada, garantindo que todos estejam cientes das mudanças e de suas implicações.

Consequentemente, para possibilitar um escopo flexível e um painel que apoie o monitoramento das mudanças resultantes, a **análise de impacto** é fundamental para a eficácia e eficiência dos processos. Compreender completamente as implicações de uma mudança aumenta a confiança da equipe ao avaliar sua viabilidade, considerando os riscos, dependências e esforços necessários para a implementação.

Nesse contexto, o **monitoramento de métricas de desenvolvimento** pode ser essencial, pois envolve coletar, analisar e interpretar dados relacionados ao processo de desenvolvimento. Esse aspecto influencia diretamente a robustez do sistema ao fornecer insights que ajudam a tomar decisões informadas sobre planejamento, gestão de recursos, priorização de tarefas e identificação de gargalos no processo de desenvolvimento, por meio da medição do progresso.

- **BP3: Empregar Prototipagem e Desenvolvimento Incremental**

A confiança em projeções preliminares guiadas por uma cultura de validação é fundamental para esclarecer e validar rapidamente os requisitos. A **prototipagem**, com seu aspecto inerente de explorar ideias e funcionalidades antes da implementação, é um artefato de software que facilita a visualização de mudanças e melhorias propostas. Ela ajuda a identificar problemas de design e usabilidade precocemente, mitiga riscos e evita desalinhamentos de perspectivas.

Em alinhamento com esse aspecto, os testes de **usabilidade** e a promoção de uma **cultura de feedback** são essenciais para garantir que as mudanças propostas atendam às necessidades de usuários finais e outros stakeholders. Envolver esses atores é crucial para ajustar e refinar os designs, compreender os requisitos e funcionalidades do software e garantir que as mudanças propostas sejam relevantes para a realidade. O feedback contínuo permite adaptações rápidas e promove uma cultura de interação que influencia a evolução do produto de software consoante as expectativas e necessidades dos stakeholders.

A entrega fragmentada ao longo do ciclo de vida do desenvolvimento de software é outro fator importante em um ambiente com alta volatilidade de requisitos. As **entregas incrementais** são uma maneira eficaz de garantir que o software esteja consistentemente alinhado com as necessidades atuais dos stakeholders. Elas reduzem o risco de falhas significativas ao facilitar a identificação e resolução de problemas em uma escala menor e mais gerenciável.

- **BP4: Manter Documentação Abrangente**

Embora as metodologias ágeis sejam benéficas para gerenciar requisitos dinâmicos, nossos achados sugerem a necessidade de equilibrar agilidade com documentação. Os princípios ágeis devem ser complementados com documentação suficiente para garantir clareza e continuidade, especialmente em projetos complexos. Nesse sentido, a disponibilidade de uma **matriz de rastreabilidade de requisitos** é crucial. Essa matriz documenta e rastreia o relacionamento entre vários requisitos e outros artefatos do ciclo de vida do software, garantindo que todos os artefatos estejam cobertos e verificáveis ao longo do desenvolvimento.

A matriz de rastreabilidade é complementada por uma **documentação viva**, continuamente atualizada e mantida ao longo do ciclo de vida do software por meio de um histórico de versões. Esse histórico não apenas reflete o estado atual do projeto, mas também preserva a documentação prévia sobre requisitos, regras de negócios, design, arquitetura e outras informações críticas.

Recomendamos, portanto, a adoção de metodologias ágeis, como Scrum ou Kanban, enquanto documentamos decisões, requisitos e mudanças em um formato acessível. Essa docu-

mentação deve ser atualizada regularmente e disponibilizada para todos os membros da equipe.

- **BP5: Fomentar Comunicação Clara e Engajamento dos Stakeholders**

Comunicação eficaz e coordenação são essenciais para gerenciar ambientes dinâmicos. **Reuniões regulares e programadas ao longo do ciclo de vida do projeto** são cruciais para atualizações de status, discussão de impedimentos, planejamento de tarefas futuras, coleta de feedback e promoção de melhorias contínuas. Manter essa consistência facilita a comunicação contínua, ajuda na identificação rápida e resolução de problemas e melhora a colaboração.

Um tema recorrente nas entrevistas foi a falta de canais de comunicação adequados, o que resultou em funcionalidades mal especificadas e altas taxas de manutenção. Recomendamos, portanto, o estabelecimento de canais e protocolos de comunicação regulares entre todos os stakeholders, incluindo reuniões programadas, relatórios de progresso e ferramentas colaborativas, para garantir que todas as partes estejam informadas e possam fornecer contribuições conforme necessário.

Definir canais de comunicação claros é um fator-chave para facilitar o acesso à informação por meio da centralização, reduzindo conflitos e retrabalho, minimizando o risco de perda de informações e aumentando a eficiência e a transparência.

- **BP6: Utilizar Ferramentas e Técnicas Automatizadas**

Como mencionado anteriormente, um problema significativo destacado nas entrevistas foi a ausência de um histórico de mudanças nos artefatos gerados durante o ciclo de vida do software. Os **sistemas de controle de versão** são aliados valiosos nesse aspecto, pois facilitam o gerenciamento e registro de alterações no código-fonte ao longo do projeto. Esses sistemas promovem uma cultura de colaboração eficiente, permitindo que vários desenvolvedores trabalhem simultaneamente no mesmo projeto, além de possibilitarem o rastreamento de mudanças e a reversão para versões anteriores.

As **ferramentas de gerenciamento de requisitos** também são pilares fundamentais em projetos com alta volatilidade de requisitos. Essas ferramentas capturam, documentam, rastreiam e gerenciam requisitos durante todo o ciclo de vida do software, facilitando o gerenciamento de mudanças, rastreamento e priorização com base nas necessidades dos stakeholders e no impacto do projeto.

Nas entrevistas, ficou evidente que a integração contínua e a melhoria contínua foram destacadas como fatores cruciais para o sucesso da plataforma PNLD Avaliação. Recomendamos,

portanto, a implementação de práticas de **Integração Contínua (CI)** e **Implantação Contínua (CD)**. A CI é fundamental para detectar erros precocemente, já que cada integração é verificada por um *build* automatizado, incluindo testes para identificar problemas rapidamente. Isso impacta positivamente a qualidade, promovendo uma base de código estável e funcional.

A CD permite a liberação automatizada de software, possibilitando que códigos atualizados sejam entregues de maneira rápida e eficiente em ambientes de produção. Isso reduz o tempo necessário para implementar e liberar novas funcionalidades e correções de bugs, acelerando a entrega de valor. Outros fatores considerados incluem a confiabilidade proporcionada pela automação e a facilidade de adaptação às mudanças de requisitos, pois novas versões podem ser necessárias devido à natureza dinâmica do ambiente.

Tanto CI quanto CD apontam para a **automação de testes**, que, juntamente com revisões regulares, pode ajudar a manter a estabilidade do sistema e identificar problemas rapidamente. O teste automatizado detecta problemas precocemente e garante que o software permaneça robusto, mesmo com mudanças frequentes. A automação pode ser uma aliada na redução de custos, aumento da eficiência e garantia de que funcionalidades críticas sejam consistentemente testadas em cada nova versão.

- **BP7: Gestão de Riscos e Planejamento de Contingência**

Vários pontos-chave precisam ser considerados para gerenciar com sucesso o dinamismo de projetos de software e as nuances de múltiplos stakeholders envolvidos. **Identificar riscos antecipadamente** é essencial para mitigar impactos, promovendo uma cultura de proatividade na equipe em vez de reatividade, abordando problemas antes que se tornem críticos. Além disso, é um excelente recurso para apoiar a tomada de decisões, fornecendo informações valiosas que ajudam a priorizar recursos e ações para enfrentar os riscos mais críticos.

Nesse contexto, ter um **plano de contingência** é um recurso estratégico essencial para garantir a continuidade do projeto, mesmo quando erros críticos ocorrem, minimizando perdas significativas por meio de ações como planos de backup de dados e a alocação de recursos adicionais, cruciais para manter a agilidade.

- **BP8: Aprendizado e Melhoria Contínuos**

O aprendizado contínuo e a melhoria contínua formam um pilar crucial no ciclo de vida de um projeto. Realizar **reuniões retrospectivas** em intervalos consistentes e regulares é essencial para refletir sobre o trabalho já realizado, identificar o que funcionou bem e o que

não funcionou, além de determinar o que pode ser melhorado para futuras iterações. Esses aspectos promovem melhorias contínuas nos processos por meio de comunicação aberta entre os stakeholders, impactando positivamente o engajamento da equipe.

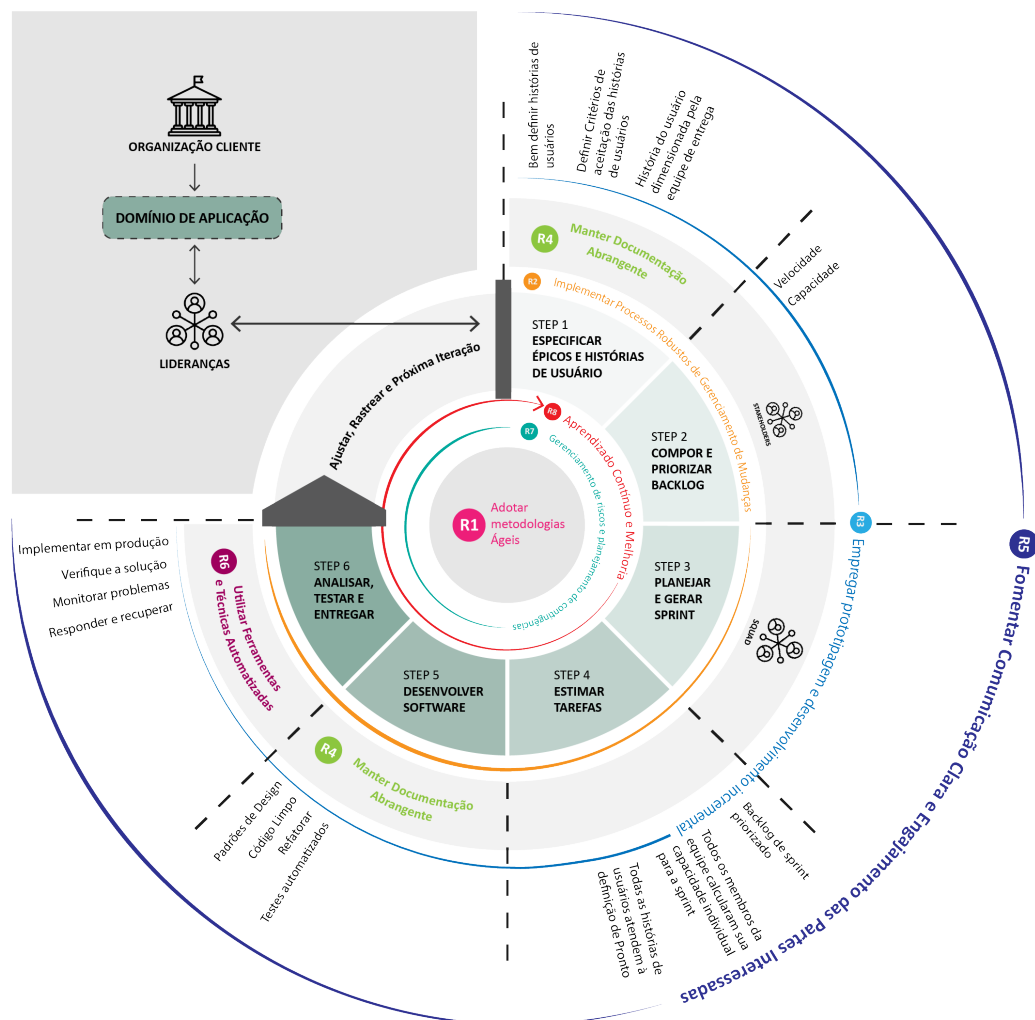
Outro aspecto identificado nas entrevistas, importante para pessoas que trabalham em ambientes dinâmicos com múltiplos stakeholders, é a importância de **treinamento e desenvolvimento**. Garantir que toda a equipe esteja alinhada e atualizada sobre as nuances do projeto, tendências, tecnologias e metodologias é crucial. Isso assegura que práticas modernas e eficazes possam ser aplicadas no desenvolvimento de software.

Com essas boas práticas, o gerenciamento de projetos em ambientes complexos e dinâmicos pode se tornar mais eficiente, proporcionando maior estabilidade e satisfação entre stakeholders e usuários finais.

5.2 PROCESSO DE DESENVOLVIMENTO DE SOFTWARE

Ao longo deste estudo, percebeu-se a necessidade de adaptar e projetar um processo ágil de desenvolvimento de software para acomodar as recomendações descritas acima. O processo é baseado em seis etapas e permite que uma equipe realize suas atividades em um processo iterativo que incorpora feedback contínuo, promove a colaboração entre todas as partes interessadas e responde rapidamente a mudanças nos requisitos. A Figura 31 ilustra o processo de desenvolvimento proposto, e cada estágio é explicado abaixo.

Figura 31 – Boas Práticas aplicadas no Processo de Desenvolvimento de Software



Fonte: Elaborada pelo Autor (2024)

Nesse processo ágil de desenvolvimento de software, o início se dá com a especificação de épicos e histórias de usuário em colaboração com as partes interessadas, garantindo que estejam bem definidos e adequadamente dimensionados. Em seguida, o *backlog* do produto é composto e priorizado com base na velocidade e capacidade da equipe. Durante o planejamento da *sprint*, a equipe define metas claras e seleciona as histórias de usuário para a próxima *sprint*. As histórias de usuário são então divididas em tarefas, com o esforço estimado e as capacidades individuais da equipe calculadas. A fase de desenvolvimento envolve a implementação das histórias de

usuário, com foco em padrões de design, código limpo, refatoração e testes automatizados. Após a conclusão do desenvolvimento, as funcionalidades são analisadas e validadas por meio de testes rigorosos. As funcionalidades que passaram nos testes são implantadas em produção. O monitoramento contínuo do ambiente de produção segue, com respostas rápidas a quaisquer problemas que surgirem para garantir o bom funcionamento das operações. Esse processo estruturado facilita a entrega contínua de software de alta qualidade.

O processo ágil de desenvolvimento proposto aborda questões identificadas na análise dos artefatos de software e nos relatórios das principais partes interessadas entrevistadas. Os principais pontos identificados são a integração contínua, comunicação eficaz e documentação abrangente, que possibilitam que quaisquer mudanças ou necessidades emergentes sejam rapidamente identificadas e incorporadas, conforme ilustrado na Figura 31.

A investigação se concentrou em um domínio específico que enfrenta constantemente novos desafios. O processo de desenvolvimento proposto no estudo poderia, potencialmente, ser aplicado a outras situações e domínios com características semelhantes. Esse processo e as recomendações sugeridas podem beneficiar o desenvolvimento de software para implementação de políticas públicas baseadas em chamadas públicas e ciclos de avaliação com datas e fases bem definidas, sujeitas à evolução e mudanças resultantes das escolhas dos formuladores de políticas em relação a novas prioridades políticas e outras situações que envolvem partes interessadas dinâmicas e diversas.

6 CONCLUSÃO

Neste capítulo são abordadas as principais contribuições da pesquisa, discutindo os resultados alcançados e seu impacto no campo de estudo investigado. Além disso, são apontadas as limitações do estudo, considerando desde os possíveis vieses dos dados autodeclarados até as restrições iniciais considerando a especificidade do domínio. Por fim, são destrinchados os principais campos para ampliação dos resultados e validação das propostas, discorrendo acerca dos trabalhos futuros.

6.1 PRINCIPAIS CONTRIBUIÇÕES

A partir de uma investigação aprofundada em um contexto real de desenvolvimento de software governamental, este estudo analisou como a qualidade de uma solução computacional e do seu processo de desenvolvimento, são impactados pela volatilidade de requisitos em ambientes dinâmicos e com múltiplos stakeholders. Ao longo dessa condução, foram destacados como os conflitos de interesses, a comunicação falha entre stakeholders, e a não centralização dos processos nas partes interessadas, adicionam complexidade no ciclo de vida de software que, por consequência, é fragilizado ainda mais frente a dinamicidade do ambiente.

Com a análise de trabalhos correlatos encontrados na literatura, foi possível identificar preocupações específicas relacionadas à gestão e ao ciclo de vida de requisitos voláteis. Essa lacuna decorre, principalmente, da ausência de técnicas e abordagens voltadas ao monitoramento de contextos dinâmicos. Além disso, estudos destacam preocupações recorrentes, como a comunicação entre stakeholders e a sub-representação de usuários finais no ciclo de vida do software. Assim, torna-se urgente repensar o ciclo de vida de softwares dinâmicos para lidar de forma mais eficaz com a volatilidade dos requisitos, enfrentando os desafios de gestão e mitigação de seus impactos. Ademais, é fundamental adotar abordagens amplas e centradas nas partes interessadas, tratando a comunicação e o engajamento como elementos centrais nos projetos de desenvolvimento.

A partir dessas percepções, a identificação de uma ambiente real que se enquadrasse nas características investigadas, colaborou para um estudo acerca dos aspectos centrais desta dissertação. Para isso, foram executados três estágios práticos junto ao PNLD Avaliação, software de apoio a Avaliação Pedagógica do Programa Nacional do Livro e do Material Didático. Os desafios e oportunidades na gestão de requisitos, usabilidade e ambientes dinâmicos com múltiplos stakeholders, foram questões centrais na investigação conduzida. E de modo geral, os

resultados destacam como a centralização de stakeholders, a análise de usabilidade e a integração de equipes afetam o desenvolvimento e operação de sistemas, reforçando a importância de abordagens robustas e iterativas na engenharia de software.

No Primeiro Estágio, a análise das histórias de usuário revelou que a volatilidade dos requisitos era um grande desafio inicial. A presença de quatro stakeholders com interesses distintos gerou mudanças frequentes e interrupções e, a centralização em dois stakeholders reduziu essa volatilidade, aumentando a eficiência e facilitando a manutenção do sistema. No Segundo Estágio, as entrevistas revelaram desafios amplos na gestão de ambientes com múltiplos stakeholders. Problemas de comunicação entre equipes, volatilidade dos requisitos e alta rotatividade de pessoal exigem estratégias mais eficazes para gestão de mudanças e documentação. Embora metodologias ágeis ofereçam flexibilidade, sua eficácia foi limitada sem documentação clara e processos bem definidos. A sobrecarga das equipes e a falta de um mediador formal dificultaram o alinhamento e a resolução de problemas críticos.

Já no Terceiro Estágio, a avaliação de usabilidade apontou a necessidade de um planejamento mais centrado no usuário. Apesar de alcançar uma pontuação SUS aceitável, muitos participantes relataram insatisfação com aspectos como facilidade de aprendizado e eficiência. Problemas estruturais, como integração inadequada e redundâncias, prejudicaram a produtividade e a experiência do usuário. A falta de iteração e feedback no desenvolvimento resultou em uma interface complexa, exigindo re-aprendizagem constante e frustrando os usuários.

Em síntese, para lidar com esses desafios, a proposição de um conjunto de boas práticas se dá baseada em metodologias ágeis, gestão de mudanças e desenvolvimento incremental. A adoção dessas práticas é justificada pela necessidade que os projetos têm de enfrentar a volatilidade de requisitos e promover uma comunicação clara entre stakeholders, a partir de processo centrados nas partes interessadas. As boas práticas se baseiam em abordagens focadas em controle de versão e integração contínua para garantir rastreabilidade e agilidade, além de considerar a documentação abrangente para equilibrar a flexibilidade ágil com a necessidade de continuidade e clareza em projetos complexos. De modo geral, essas boas práticas pretendem apoiar toda a gestão do processo de desenvolvimento do software, como meio de garantir qualidade da ideação até a manutenção.

6.2 LIMITAÇÕES E AMEAÇAS À VALIDADE

As conclusões apontadas por esta investigação apresentam importantes contribuições acerca dos impactos de ambientes dinâmicos e com múltiplos stakeholders na qualidade do ciclo

de vida do software. Mesmo assim, é de suma importância compreender os aspectos que podem influenciar a validade desses resultados. Dessa forma, embora as metodologias empregadas nesta investigação sejam robustas e tenham sido extremamente detalhadas, são apresentadas algumas limitações e ameaças à validade que devem ser consideradas.

- No **Primeiro Estágio**, a análise estatística comparativa e a inspeção de artefatos de software dependem da qualidade e consistência da documentação existente. Logo, a condução do pesquisador responsável considerou registros de terceiros, tendo, assim, que considerar exatamente o que lhe disponibilizado. Esta é uma preocupação necessária, pois, se os registros analisados não forem completos ou atualizados, os resultados podem não refletir com precisão a evolução do projeto. Além disso, a dependência de dados históricos pode introduzir vieses, já que a documentação pode ser influenciada por diferentes contextos e percepções ao longo do tempo.
- No **Segundo Estágio**, as entrevistas e a escuta de reuniões, embora sejam fundamentais para capturar percepções e experiências, estão sujeitas a vieses de resposta e interpretação. Os participantes podem omitir informações relevantes ou apresentar vieses inconscientes, especialmente em um ambiente altamente dinâmico e politicamente sensível, como o de políticas públicas. A análise temática dedutiva, embora poderosa para interpretar dados qualitativos, também pode ser influenciada pelas interpretações do pesquisador responsável, introduzindo um risco de subjetividade.
- O **Terceiro Estágio**, focado na usabilidade por meio da Escala SUS e questões abertas, embora não apresente limitações relacionadas à amostragem, uma vez que considerando as características e exigências do próprio método, houve-se um grau de adesão significativa, ainda assim, ela representa um estado e tempo específicos do Banco de Avaliadores. Em termos de generalização de resultados, frisa-se que, a aplicação de questionários e escalas autoavaliativas pode ser influenciada pelo contexto de uso, pela familiaridade dos usuários com o sistema ou por suas expectativas prévias, enviesando potencialmente os resultados. A análise temática dedutiva, apesar de complementar a Escala SUS, está sujeita a interpretações subjetivas que podem variar conforme os critérios definidos pelos pesquisadores.

É válido mencionar como ameaça à validade externa, o fato de ter analisado uma única plataforma com os aspectos de interesse. Isso reduz as possibilidades de generalização dos resultados encontrados para outros contextos. Dessa forma, embora o estudo ofereça contribuições

relevantes para a área de desenvolvimento de software que atuam em ambientes dinâmicos e com múltiplos stakeholders, é fundamental reconhecer essas limitações e interpretá-lo à luz dessas considerações.

6.3 TRABALHOS FUTUROS

Com base nas limitações e nos insights obtidos em toda investigação, a pesquisa abre várias possibilidades para trabalhos futuros, que podem aprofundar e expandir o entendimento sobre a gestão de sistemas interativos em ambientes dinâmicos com múltiplos stakeholders. Algumas propostas incluem:

- Implementar e avaliar as boas práticas recomendadas neste trabalho em um ambiente de desenvolvimento de software real, a fim de avaliar se e como elas respondem às necessidades específicas. Nesse sentido, é válido considerar aspectos amplos, como diferentes contextos e tipos de organizações, percorrendo o público e o privado.
- Conduzir estudos comparativos entre diferentes organizações ou países para verificar se os resultados encontrados são generalizáveis ou se dependem de contextos específicos. Isso poderia ajudar a entender como fatores culturais, políticos e organizacionais influenciam a gestão de requisitos e a usabilidade em sistemas computacionais.
- Projetar e implementar soluções específicas para facilitar a comunicação, o gerenciamento de mudanças e a documentação em projetos com múltiplos stakeholders. Estes foram tópicos sensíveis na condução deste estudo, logo, avaliar a eficácia das ferramentas já existentes e até novas propostas, em diferentes cenários e suas contribuições para melhorar a eficiência e a qualidade dos sistemas.

Considerando que os trabalhos futuros são essenciais para aprofundar o conhecimento sobre a gestão de sistemas em ambientes dinâmicos, os que foram apontados acima, não apenas consolidariam os achados desta investigação, mas também contribuiriam para um corpo mais robusto de conhecimento aplicado à Engenharia de Software em contextos complexos.

REFERÊNCIAS

- ABRELIVROS, A. B. de Livros e C. E. **Como Funciona o PNLD: Guia Prático**. 2020. Acessado em: Janeiro de 2024. Disponível em: <https://abrelivros.org.br/site/wp-content/uploads/2020/10/2020_Como_funciona_o_PNLD.pdf>.
- AHMAD, K.; ABDELRAZEK, M.; ARORA, C.; BANO, M.; GRUNDY, J. Requirements engineering for artificial intelligence systems: A systematic mapping study. **Information and Software Technology**, Elsevier, v. 158, p. 107176, 2023.
- ALZEYANI, E. M. M.; SZABÓ, C. A study on the effectiveness of agile methodology using a dataset. **Acta Electrotechnica et Informatica**, v. 23, n. 1, p. 3–10, 2023.
- AMAJUOYI, P.; BENJAMIN, L. B.; ADEUS, K. B. Agile methodologies: Adapting product management to rapidly changing market conditions. **GSC Advanced Research and Reviews**, GSC Advanced Research and Reviews, v. 19, n. 2, p. 249–267, 2024.
- AMBREEN, T. Handling socio-technical barriers involved in crowd-based requirements elicitation. In: **2019 IEEE 27th International Requirements Engineering Conference (RE)**. [S.l.]: IEEE, 2019. p. 476–481.
- ANABA, D.; KESS-MOMOH, A.; AYODEJI, S. Strategic negotiation and contract management: Best practices for high-stakes projects. **International Journal of Applied Research in Social Sciences**, v. 6, n. 7, p. 1310–1320, 2024.
- ANDREESCU, A.; MIRCEA, M. issues and challenges of business rules modeling in software systems for business management. **Informatica Economica**, INFOREC Association, v. 18, n. 1, p. 162, 2014.
- ARABIUN, A.; HOSSEINI, E.; MAHMOODI, K.; MOHSEN, B. An investigation of the socio-cultural impacts of the digital transformation in developed, developing, and middle: Africa and asia pacific. In: **Exploring Business Ecosystems and Innovation Capacity Building in Global Economics**. [S.l.]: IGI Global, 2023. p. 112–130.
- ARORA, C.; JOHN, G.; MOHAMED, A. Advancing requirements engineering through generative ai: assessing the role of llms. **arXiv preprint**, arXiv:2310.13976, 2023.
- AZIZ, R. A.; WONG, B. The interplay between requirements relationships knowledge and requirements change towards software project success: an assessment using partial least square (pls). **Procedia Computer Science**, Elsevier, v. 46, p. 732–741, 2015.
- BALASUBRAMANIAM, V. S.; SALUNKHE, V.; AGRAWAL, S.; GOEL, P. P.; GUPTA, V.; GUPTA, A. Optimizing cross functional team collaboration in it project management. **Deleted Journal**, v. 12, n. 1, p. 140–179, 2024.
- BANGOR, A.; KORTUM, P.; MILLER, J. Determining what individual sus scores mean: Adding an adjective rating scale. **Journal of usability studies**, Usability Professionals' Association Bloomingdale, IL, v. 4, n. 3, p. 114–123, 2009.
- BARBOSA, S.; SILVA, B. **Interação humano-computador**. [S.l.]: Elsevier Brasil, 2010.
- BARBOSA, S. D. J.; SILVA, B. d.; SILVEIRA, M. S.; GASPARINI, I.; DARIN, T.; BARBOSA, G. D. J. Interação humano-computador e experiência do usuario. **Auto publicação**, 2021.

BATISTA, A. A. G. Recomendações para uma política pública de livros didáticos. **Ministério da Educação, Secretaria de Educação Fundamental**, 2001.

BEGIER, B.; WADOWICKI, J. Feedback from users on a software product to improve its quality in engineering applications. In: **Software Engineering Techniques: Design for Quality**. [S.l.]: Springer, 2006. p. 167–178.

BHANUSHALI, A. Challenges and solutions in implementing continuous integration and continuous testing for agile quality assurance. **International Journal of Science and Research (Raipur, India)**, v. 12, n. 10, p. 1626–1644, 2023.

BINDER, M.; VOGT, A.; BAJRAKTARI, A.; VOGELSANG, A. Automatically classifying kano model factors in app reviews. In: **International Working Conference on Requirements Engineering: Foundation for Software Quality**. [S.l.]: Springer, 2023. p. 245–261.

BOSCHMA, R. Proximity and innovation: A critical assessment. **Regional Studies**, Routledge, v. 39, n. 1, p. 61–74, 2005. Disponível em: <<https://doi.org/10.1080/0034340052000320887>>.

BOSE, R. J. C.; SINGI, K.; KAULGUD, V.; PODDER, S.; BURDEN, A. P. Software engineering in a governed world: Opportunities and challenges. In: **Proceedings of the IEEE/ACM 42nd International Conference on Software Engineering Workshops**. [S.l.: s.n.], 2020. p. 309–310.

BOURQUE, P.; FAIRLEY, R. E. Swebok v3. 0: Guide to the software engineering body of knowledge. **IEEE Computer Society**, p. 1–335, 2014.

BRAUM, M. **Guia: Como Medir a Usabilidade de Produtos com System Usability Scale (SUS)**. 2019. Acessado em: Fevereiro de 2024. Disponível em: <<https://brasil.uxdesign.cc/guia-como-medir-a-usabilidade-de-produtos-com-system-usability-scale-sus-e08f4361d9db>>.

BRAUN, V.; CLARKE, V. Using thematic analysis in psychology. **Qualitative research in psychology**, Taylor & Francis, v. 3, n. 2, p. 77–101, 2006.

BROOKE, J. Sus: a retrospective. **Journal of usability studies**, Usability Professionals' Association Bloomingdale, IL, v. 8, n. 2, p. 29–40, 2013.

BUDGEN, D.; TURNER, M.; BRERETON, P.; KITCHENHAM, B. A. Using mapping studies in software engineering. In: **Ppig**. [S.l.: s.n.], 2008. v. 8, p. 195–204.

BURGUEÑO, L.; FOURES, D.; COMBEMALE, B.; KIENZLE, J.; MUSSBACHER, G. Global decision making support for complex system development. In: **2024 IEEE 32nd International Requirements Engineering Conference (RE)**. [S.l.: s.n.], 2024. p. 252–263.

CONSTANTINO, H. R. A.; JUNIOR, E. R. de O.; BORTOLI, L. Â. D. Redes sociais e engenharia de requisitos: é possível conciliar? In: SBC. **Anais da III Escola Regional de Engenharia de Software**. [S.l.], 2019. p. 17–24.

CRESWELL, J. W.; CLARK, V. L. P. **Designing and conducting mixed methods research**. [S.l.]: Sage publications, 2017.

DASANAYAKE, S.; AARAMAA, S.; MARKKULA, J.; OIVO, M. Impact of requirements volatility on software architecture: How do software teams keep up with ever-changing requirements? **Journal of software: evolution and process**, Wiley Online Library, v. 31, n. 6, p. e2160, 2019.

DEMISSIE, S.; KEENAN, F.; MCCAFFERY, F. The sync-up process to assist multiple stakeholder communication of requirement analysis in embedded medical software development. In: **2021 4th International Conference on Information and Computer Technologies (ICICT)**. [S.l.: s.n.], 2021. p. 131–136.

EDUCAÇÃO, F. N. de Desenvolvimento da. **Encontro do PNLD aborda a importância do programa para o aprendizado dos estudantes**. 2022. Acessado em: Janeiro de 2024. Disponível em: <<https://www.gov.br/fnde/pt-br/assuntos/noticias/encontro-do-pnld-aborda-a-importancia-do-programa-para-o-aprendizado-dos-estudantes>>.

EJA, K. M.; RAMEGOWDA, M. Government project failure in developing countries: a review with particular reference to nigeria. **Global Journal of Social Sciences**, v. 19, p. 35–47, 2020.

EYIEYIEN, O. G.; IDEMUDIA, C.; PAUL, P. O.; IJOMAH, T. I. Advancements in project management methodologies: Integrating agile and waterfall approaches for optimal outcomes. **Engineering Science & Technology Journal**, v. 5, n. 7, p. 2216–2231, 2024.

FERRE, X. Integration of usability techniques into the software development process. In: **ICSE Workshop on SE-HCI**. [S.l.: s.n.], 2003. p. 28–35.

FERREIRA, S.; COLLOFELLO, J.; SHUNK, D.; MACKULAK, G. Understanding the effects of requirements volatility in software engineering by using analytical modeling and software process simulation. **Journal of Systems and Software**, Elsevier, v. 82, n. 10, p. 1568–1577, 2009.

FILGUEIRAS, F.; FLÁVIO, C.; PALOTTI, P. Digital transformation and public service delivery in brazil. **Latin American Policy**, Wiley Online Library, v. 10, n. 2, p. 195–219, 2019.

GALENO, L.; MOTA, W.; MELLO, R. Investigating the participation of female software developers in retrospective meetings. In: **Anais do IX Workshop sobre Aspectos Sociais, Humanos e Econômicos de Software**. Porto Alegre, RS, Brasil: SBC, 2024. p. 46–57. ISSN 2763-874X. Disponível em: <<https://sol.sbc.org.br/index.php/washes/article/view/29441>>.

GRAESSLER, C.; OLEFF, C.; PREUß, D. Proactive management of requirement changes in the development of complex technical systems. **Applied Sciences**, MDPI, v. 12, n. 4, p. 1874, 2022.

GRUBB, A. M.; SPOLETINI, P. Bringing stakeholders along for the ride: Towards supporting intentional decisions in software evolution. In: SPRINGER. **International Working Conference on Requirements Engineering: Foundation for Software Quality**. [S.l.], 2023. p. 56–64.

GUERRERO-VÁSQUEZ, L. F.; BUSTAMANTE-CACAO, K. C.; VÁSQUEZ-VÁSQUEZ, C. E.; SERPA-ANDRADE, L. J.; ORDOÑEZ-ORDOÑEZ, J. O.; CHASI-PESANTEZ, P. A.; PLAZA-VEGA, V. M. Proposal of an agile approach based on scrum for managing continuing education programs. In: SPRINGER. **International Conference on Science, Technology and Innovation for Society**. [S.l.], 2023. p. 318–328.

HEIN, P. H.; KAMES, E.; CHEN, C.; MORKOS, B. Employing machine learning techniques to assess requirement change volatility. **Research in engineering design**, Springer, v. 32, p. 245–269, 2021.

HIX, D.; REX, H. H. Developing user interfaces: Ensuring usability through product process. **Wiley Sons Inc.**, Taylor & Francis, p. 381, 1993.

HOODA, S.; SOOD, V. M.; SINGH, Y.; DALAL, S.; SOOD, M. Agile software development: Trends, challenges and applications. John Wiley & Sons, 2023.

HOY, Z.; XU, M. **Agile software requirements engineering challenges-solutions—a conceptual framework from systematic literature review**. *Information (Basel)* **14** (6), 322 (2023). 2023.

JARZĘBOWICZ, A.; SITKO, N. Agile requirements prioritization in practice: Results of an industrial survey. *Procedia Computer Science*, Elsevier, v. 176, p. 3446–3455, 2020.

KAPLAN, F.; STEELS, L.; MCINTYRE, A. **An architecture for evolving robust shared communication systems in noisy environments**. 1997.

KAUR, G.; KAUR, I.; HARNAL, S.; MALIK, S. Factors and techniques for software quality assurance in agile software development. **Agile Software Development: Trends, Challenges and Applications**, Wiley Online Library, p. 257–272, 2023.

KAZMAN, R.; PASQUALE, L. Software engineering in society. *IEEE Software*, IEEE, v. 37, n. 1, p. 7–9, 2019.

KOSCIANSKI, A.; SOARES, M. dos S. **Qualidade de Software-2ª Edição: Aprenda as metodologias e técnicas mais modernas para o desenvolvimento de software**. [S.l.]: Novatec Editora, 2007.

LARMAN, C.; BASILI, V. R. Iterative and incremental developments. a brief history. *Computer*, IEEE, v. 36, n. 6, p. 47–56, 2003.

LAZAR, J.; FENG, J. H.; HOCHHEISER, H. Chapter 8 - interviews and focus groups. In: LAZAR, J.; FENG, J. H.; HOCHHEISER, H. (Ed.). **Research Methods in Human Computer Interaction (Second Edition)**. Second edition. Boston: Morgan Kaufmann, 2017. p. 187–228. ISBN 978-0-12-805390-4. Disponível em: <<https://www.sciencedirect.com/science/article/pii/B978012805390400008X>>.

LEWELLEN, S. Identifying key stakeholders as part of requirements elicitation in software ecosystems. In: **Proceedings of the 24th ACM International Systems and Software Product Line Conference-Volume B**. [S.l.: s.n.], 2020. p. 88–95.

LEWELLEN, S. A comprehensive approach to identifying key stakeholders in complicated software ecosystems. In: **2021 IEEE 29th International Requirements Engineering Conference (RE)**. [S.l.: s.n.], 2021. p. 492–497.

LEWELLEN, S. A comprehensive approach to identifying key stakeholders in complicated software ecosystems. In: **2021 IEEE 29th International Requirements Engineering Conference (RE)**. [S.l.: s.n.], 2021. p. 492–497.

LIENTZ, B. P.; SHWANSON, E. B. **Software maintenance management**. [S.l.]: Addison-Wesley Longman Publishing Co., Inc., 1980.

LIMAYLLA-LUNAREJO, M.-I.; CONDORI-FERNANDEZ, N.; LUACES, M. R. Requirements classification using fasttext and beto in spanish documents. In: **International Working Conference on Requirements Engineering: Foundation for Software Quality**. [S.l.]: Springer, 2023. p. 159–176.

LUITEL, D.; HASSANI, S.; SABETZADEH, M. Using language models for enhancing the completeness of natural-language requirements. In: **International Working Conference on Requirements Engineering: Foundation for Software Quality**. [S.l.]: Springer, 2023. p. 87–104.

MAYHEW, D. J. **The Usability Engineering Lifecycle: A Practitioner's Handbook for User Interface Design**. 1st. ed. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 1999. ISBN 1558605614.

MEC, M. d. E. Pnld. **Governo Brasileiro**, 2023. Disponível em: <<http://portal.mec.gov.br/component/content/article?id=12391:pnld>>.

MISHRA, D.; MISHRA, A.; YAZICI, A. Successful requirement elicitation by combining requirement engineering techniques. In: IEEE. **2008 First International Conference on the Applications of Digital Information and Web Technologies (ICADIWT)**. [S.l.], 2008. p. 258–263.

MOHAMMAD, A.; KOLLAMANA, J. M. Causes and mitigation practices of requirement volatility in agile software development. **Informatics**, MDPI, v. 11, n. 1, p. 12, 2024.

MOYON, F.; BECKERS, K.; KLEPPER, S.; LACHBERGER, P.; BRUEGGE, B. Towards continuous security compliance in agile software development at scale. In: **Proceedings of the 4th International Workshop on Rapid Continuous Software Engineering**. [S.l.: s.n.], 2018. p. 31–34.

NAGEL, L.; KARRAS, O.; AMIRI, S. M.; SCHNEIDER, K. Supporting shared understanding in asynchronous communication contexts. In: **International Working Conference on Requirements Engineering: Foundation for Software Quality**. [S.l.]: Springer, 2023. p. 39–55.

NAKASH, M. Agile software development: The experience of working in sprints [research in progress]. In: **InSITE 2024: Informing Science+ IT Education Conferences**. [S.l.: s.n.], 2024. p. 002.

NETO, M.; ABICH, D.; CORREA, C.; PARIZI, R. Abordagem metodológica de integração das disciplinas de engenharia de requisitos e interação humano-computador: Um estudo de caso. In: SBC. **Anais do VII Encontro Nacional de Computação dos Institutos Federais**. [S.l.], 2020. p. 85–92.

NEUMANN, W. P.; WINKELHAUS, S.; GROSSE, E. H.; GLOCK, C. H. Industry 4.0 and the human factor—a systems framework and analysis methodology for successful development. **International journal of production economics**, Elsevier, v. 233, p. 107992, 2021.

NIELSEN, J. **Usability Engineering**. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 1994. ISBN 9780080520292.

NIELSEN, J. **Usability engineering**. [S.l.]: Morgan Kaufmann, 1994.

NINGSIH, A. D.; ARIANI, D.; SAGALA, S.; HARAHAH, D. Project team bulding, conflict and negotiation. **Devotion: Journal of Research and Community Service**, v. 3, n. 14, p. 2519–2533, 2022.

ORDONES, S. A. D. Diretrizes para o desenvolvimento de plataformas digitais: contribuições dos estudos sobre usabilidade, experiência do usuário e engenharia semiótica. Universidade Estadual Paulista (Unesp), 2023.

O'REGAN, G.; O'REGAN, G. Software inspection methodology. **The Innovation in Computing Companion: A Compendium of Select, Pivotal Inventions**, Springer, p. 235–238, 2018.

PAGNAN, A. S.; SIMPLÍCIO, G. C.; SANTOS, V. C. Design centrado no usuário e seus princípios éticos norteadores no ensino do design. **Estudos em Design**, v. 27, n. 1, 2019.

PARANJPE, S. S.; PACHLOR, R. Devsecops: Improving software development life cycle. **International Journal of Advanced Research in Science, Communication and Technology**, Shivkrupa Publication's, p. 316–325, 2024.

PINHEIRO, R. M. d. S.; ECHALAR, A. D. L. F.; QUEIROZ, J. R. d. O. As políticas públicas de livro didático no brasil: editais do pnld de biologia em questão. **Educar em Revista**, SciELO Brasil, v. 37, 2021.

POHL, K. **Requirements engineering: fundamentals, principles, and techniques**. [S.l.]: Springer Publishing Company, Incorporated, 2010.

POLO, M.; PIANTTINI, M.; RUIZ, F.; CALERO, C. Roles in the maintenance process. **ACM SIGSOFT Software Engineering Notes**, ACM New York, NY, USA, v. 24, n. 4, p. 84–86, 1999.

POPOVA, L.; SENIV, B.; KOROL, V.; GALUSHKO, O.; BIRIUKOV, I. The role of digital technologies in the public administration sphere. *Cuestiones Políticas.*, 2023.

PREECE, J.; BENYON, D.; UNIVERSITY, O. **A Guide to Usability: Human Factors in Computing**. 1st. ed. USA: Addison-Wesley Longman Publishing Co., Inc., 1993. ISBN 020162768X.

PRESSMAN, R. **Engenharia de Software - 7.ed.** [S.l.]: McGraw Hill Brasil, 2009. ISBN 9788580550443.

PRESSMAN, R. S. **Engenharia de Software uma abordagem profissional**. 7ª ed.. ed. [S.l.]: AMGH Editora Ltda, 2011.

PRESSMAN ROGER S; MAXIM, B. R. Engenharia de software, uma abordagem profissional–8ª ed–amgh editora ltda. **Porto Alegre–RS-2016**.

RAJAKUMARI, K.; HEMALATHA, D. et al. To survive in a fast-changing business landscape in the age of digital transformation organizations must be flexible and adaptive. In: **Practical Approaches to Agile Project Management**. [S.l.]: IGI Global, 2024. p. 289–304.

RIBEIRO, A. B. C.; ALVES, C. F. Feedback in virtual software development teams. In: **IEEE. 2023 18th Iberian Conference on Information Systems and Technologies (CISTI)**. [S.l.], 2023. p. 1–6.

RNP, R. N. de Ensino e P. **PNLD: solução tecnológica leva para ambiente digital Programa que distribui livros a 30 milhões de estudantes**. 2021. Acessado em: Janeiro de 2024. Disponível em: <<https://www.rnp.br/noticias/pnld-digital>>.

RUBIN, J.; CHISNELL, D. **Handbook of usability testing: How to plan, design, and conduct effective tests**. [S.l.]: John Wiley & Sons, 2008.

SANTOS, A. A. A. d. Um estudo exploratório sobre as contribuições do uso de técnicas de ihc no ensino de engenharia de requisitos. 2023.

SAURO, J. Measuring usability with the system usability scale (sus). 2011.

SIAKAS, E.; RAHANU, H.; GEORGIADOU, E.; SIAKAS, K. Requirements volatility in multicultural situational contexts. In: SPRINGER. **European Conference on Software Process Improvement**. [S.l.], 2022. p. 633–655.

SILVA, D.; LENCASTRE, M.; PIMENTEL, J.; CASTRO, J.; LIRA, L. Applying gamification to prioritize requirements in agile projects. In: **Proceedings of the 38th ACM/SIGAPP Symposium on Applied Computing**. [S.l.: s.n.], 2023. p. 1498–1507.

SILVA, J. V. d.; BARANAUSKAS, M. C. C. Interaction spaces and socioenactive dimensions: Exploring perturbations of ioht. In: **Proceedings of the 19th Brazilian Symposium on Human Factors in Computing Systems**. [S.l.: s.n.], 2020. p. 1–10.

SILVA, L.; SARAIVA, J.; AGUIAR, Y. Functional suitability assessment: A preliminary study on perceptions and trends observed in software industry. In: **Proceedings of the XXXVI Brazilian Symposium on Software Engineering**. [S.l.: s.n.], 2022. p. 208–213.

SILVA, L. de F.; SILVA, A.; ARAUJO, R.; ARAUJO, A.; DERMEVAL, D.; CANTARINO, R. Digital transformation in a brazilian public policy: A case study of pedagogical evaluation within the national textbook program. In: **Proceedings of the 25th Annual International Conference on Digital Government Research**. [S.l.: s.n.], 2024. p. 515–524.

SOMMERVILLE, I. **Engenharia de software-9**. [S.l.]: Pearson Education do Brasil, 2011.

SOUSA, K. S.; FURTADO, E. Rupi-a unified process that integrates human-computer interaction and software engineering. In: **ICSE Workshop on SE-HCI**. [S.l.: s.n.], 2003. p. 41–48.

STALLINGER, F.; GRÜNBACHER, P. System dynamics modelling and simulation of collaborative requirements engineering. **Journal of Systems and Software**, Elsevier, v. 59, n. 3, p. 311–321, 2001.

STANDARDIZATION, I. O. for. **NBR ISO/IEC 9126-1:2003 - Engenharia de software - Qualidade de Produto Parte 1: Modelo de Qualidade**. Rio de Janeiro, 2023.

SUGANYA, R.; BANERJEE, D.; MISHRA, A.; SUBBULAKSHMI, T.; SUBRAMANIAN, G. Enhancing agile development in tech companies: Backlog management, tool integration, and stakeholder collaboration. In: IEEE. **2023 6th International Conference on Recent Trends in Advance Computing (ICRTAC)**. [S.l.], 2023. p. 718–724.

SZABÓ, B.; HERCEGFI, K. User-centered approaches in software development processes: Qualitative research into the practice of hungarian companies. **Journal of Software: Evolution and Process**, Wiley Online Library, v. 35, n. 2, p. e2501, 2023.

TENÓRIO, J. M.; COHRS, F. M.; SDEPANIAN, V. L.; PISA, I. T.; MARIN, H. de F. Desenvolvimento e avaliação de um protocolo eletrônico para atendimento e monitoramento do paciente com doença celíaca. **Revista de Informática teórica e aplicada**, v. 17, n. 2, p. 210–220, 2010.

TETTEH, S. G. Empirical study of agile software development methodologies: A comparative analysis. **Asian Journal of Research in Computer Science**, v. 17, n. 5, p. 30–42, 2024.

TORRES, P. L.; CORNIDE-REYES, H. Team performance assessment and improvement in an agile methodology framework. In: IEEE. **2022 XLVIII Latin American Computer Conference (CLEI)**. [S.l.], 2022. p. 1–10.

VIAL, G. Understanding digital transformation: A review and a research agenda. **Managing digital transformation**, Routledge, p. 13–66, 2021.

WANG, J.; SUN, K.; LIU, P.; ZHANG, K.; FENG, L.; WU, X.; ZHANG, Z. Dynamic elicitation and forecasting innovation requirement of smart product-service system via user-manufacturer value co-creation perspective using multi-source data. **Computers & Industrial Engineering**, Elsevier, v. 197, p. 110511, 2024.

WITSCHEL, D.; DÖHLA, A.; KAISER, M.; VOIGT, K.-I.; PFLETSCHINGER, T. Riding on the wave of digitization: Insights how and under what settings dynamic capabilities facilitate digital-driven business model change. **Journal of Business Economics**, Springer, v. 89, n. 8, p. 1023–1095, 2019.

ZHANG, J.; CHANG, C. K.; CHUNG, J.-Y. Mockup-driven fast-prototyping methodology for web requirements engineering. In: IEEE. **Proceedings 27th Annual International Computer Software and Applications Conference. COMPAC 2003**. [S.l.], 2003. p. 263–268.

ZHANG, Y.; FINKELSTEIN, A.; HARMAN, M. Search based requirements optimisation: Existing work and challenges. In: **Requirements Engineering: Foundation for Software Quality: 14th International Working Conference, REFSQ 2008 Montpellier, France, June 16-17, 2008 Proceedings 14**. [S.l.]: Springer, 2008. p. 88–94.

ANEXO A – Comitê de Ética em Pesquisa

UNIVERSIDADE FEDERAL DE
ALAGOAS



PARECER CONSUBSTANCIADO DO CEP

DADOS DO PROJETO DE PESQUISA

Título da Pesquisa: Análise da usabilidade da plataforma computacional da avaliação pedagógica do PNLD e redesenho do instrumento de avaliação sob a ótica da Transformação Digital

Pesquisador: ALENILTON SANTOS SILVA

Área Temática:

Versão: 2

CAAE: 69059623.1.0000.5013

Instituição Proponente: Programa de Pós-Graduação em Informática

Patrocinador Principal: Financiamento Próprio

DADOS DO PARECER

Número do Parecer: 6.114.614

Apresentação do Projeto:

O estudo a ser desenvolvido utilizará uma abordagem exploratória para análise diagnóstica quantitativa e qualitativa, buscando uma familiarização com a problemática a ser estudada, utilizando de um levantamento bibliográfico no estado arte e documental através da análise de documentação interna do PNLD Avaliação, numa interação direta com os avaliadores pedagógicos. A pesquisa será realizada apenas com professores avaliadores e demais partes interessadas que estão habilitados no Banco do Programa Nacional do Livro e do Material Didático (PNLD), onde o tamanho da amostra no Brasil é 5318 (cinco mil, trezentos e dezoito). Estima-se a participação de pelo menos 15 (quinze) avaliadores neste estudo, 3 (três) por região. O estudo percorre uma análise da usabilidade da plataforma computacional da avaliação pedagógica do PNLD, utilizando dos pilares da Interação Humano-Computador (IHC) e de instrumentos da Inteligência Artificial (IA), a fim de aferir a aderência e a relação entre o sistema de avaliação pedagógica e seus usuários. Para isso, será aplicado formulário de identificação do perfil, formulário de escala de usabilidade de sistemas computacionais interativos e entrevista com grupo focal, a ser realizada de forma presencial, híbrida e/ou remota a depender da disponibilidade das partes envolvidas. Uma vez levantados e analisados os dados coletados nas etapas da pesquisa, serão aplicados os conhecimentos obtidos no processo de redesenho do instrumento de avaliação pedagógica e em seguida serão

Endereço: Av. Longitudinal UFAL 1, nº 1444, térreo do prédio do Centro de Interesse Comunitário (CIC) entre o SINTUFAL
Bairro: Cidade Universitária **CEP:** 57.072-900
UF: AL **Município:** MACEIO
Telefone: (82)3214-1041 **E-mail:** cep@ufal.br

**UNIVERSIDADE FEDERAL DE
ALAGOAS**

Continuação do Parecer: 6.114.614

reaplicados os testes de escala de usabilidade de sistemas computacionais interativos, para analisar como as melhorias implantadas afetaram a experiência do usuário. A pesquisa terá duração estimada de 2 (dois) anos, compreendendo todas as etapas descritas.

Objetivo da Pesquisa:**Objetivo Primário:**

Este projeto de pesquisa tem como objetivo aferir a aderência e usabilidade, bem como a percepção do usuário final, quanto a ferramenta de apoio usada pelos professores-avaliadores e demais partes interessadas, no processo de avaliação dos livros didáticos pelo PNLD. A proposta aqui descrita, busca avançar no estado da arte e da prática com o melhoramento de uma solução computacional observando aspectos da Interação Humano-Computador com foco em usabilidade, como meio de aprimorar e agilizar o processo de avaliação dos materiais submetidos. Além desses aspectos mencionados, pretende-se nesse projeto de pesquisa entender como o uso de ferramentas computacionais inteligentes podem trazer uma melhor experiência e efetividade na atuação dos professores-avaliadores. Para isso, busca-se primariamente, identificar o perfil dos avaliadores envolvidos e especificar um framework para avaliar a plataforma computacional atualmente usada em quesito de aderência (aqui deverão ser apontadas as percepções dos usuários frente ao uso de sistema, seus sentimentos, dificuldades e motivações de uso e/ou abandono). A partir dos resultados obtidos na avaliação, identificar os gargalos e as dores dos usuários, como forma de embasar possíveis modificações, num processo de redesign da solução computacional pautado nas heurísticas de usabilidade. Tendo mapeado as melhorias, pretende-se numa segunda etapa, avançar numa proposta de evolução de algumas etapas da avaliação por meio de uso de algoritmos inteligentes para recomendações personalizadas, com intuito de facilitar a análise dos avaliadores, levando eficiência no processo como um todo. Uma vez redesenhado e aplicadas as alterações, o sistema passa pelo mesmo processo de avaliação junto a usuários finais.

Objetivo Secundário:

Considerando as possibilidades e caminhos diante das problemáticas as quais motivam este estudo, os objetivos específicos são:

1. Realizar a análise do estado da arte e da prática.
2. Definir framework de avaliação para pontuar a partir das heurísticas de usabilidade, consequente eficiência e aplicabilidade ao meio de uso, da atual plataforma usada no processo de

Endereço: Av. Longitudinal UFAL 1, n°1444, térreo do prédio do Centro de Interesse Comunitário (CIC) entre o SINTUFAL
Bairro: Cidade Universitária **CEP:** 57.072-900
UF: AL **Município:** MACEIO
Telefone: (82)3214-1041 **E-mail:** cep@ufal.br

UNIVERSIDADE FEDERAL DE ALAGOAS



Continuação do Parecer: 6.114.614

avaliação do PNLD.

3. Identificar e mapear o perfil dos professores avaliadores e demais partes interessadas.
4. Analisar resultados primários obtidos na avaliação, com enfoque nas deficiências e não correspondências às heurísticas de usabilidade.
5. Desenvolver protótipo utilizável/aplicar o redesign com base nos apontamentos obtidos, para apresentar nova solução a nível de testagem.
6. Avaliar a eficiência da solução computacional incrementada proposta por meio de experimentos de observação e/ou formulário.
7. Avaliar a relevância das análises e avaliações propostas pelos algoritmos inteligentes e o seu impacto na experiência dos avaliadores.
8. Escrever relatórios técnicos e artigos para publicação em conferências e periódicos com qualis restrito em Ciência da Computação.

Avaliação dos Riscos e Benefícios:

Riscos:

Considera-se que a pesquisa tem risco mínimo. Pontua-se possível cansaço e/ou aborrecimento ao responder questionários, bem como constrangimento devido a possíveis dificuldades em realizar tarefas solicitadas e até medo/vergonha de não saber responder algum questionamento durante o processo.

Considera-se que a questão ténue, em se tratando dos riscos, e que pode gerar desconforto e até insegurança nos participantes, é a quebra de sigilo e anonimato, porém, para mitigar esses riscos, garante-se que os dados coletados serão consolidados partindo da premissa de que todos os indivíduos terão suas identidades preservadas através de um código de identificação, excluindo a necessidade da coleta de nomes. Durante a execução das atividades de pesquisa, quando em contato direto com os avaliadores, deixaremos claro que o objetivo do estudo não está para julgar as habilidades dos indivíduos e sim, para avaliar o sistema em quesito de usabilidade e satisfação de uso. Sempre será esclarecido que o participante poderá interromper qualquer uma das etapas sem nenhum prejuízo e com a garantia de que sua identidade será preservada. Os resultados da pesquisa também serão publicados assegurando a confidencialidade e a privacidade, a proteção da imagem e a não estigmatização, garantindo a não utilização das informações em prejuízo dos entrevistados. Os dados da pesquisa serão mantidos em disco rígido local, sob a guarda dos pesquisadores responsáveis, por um período mínimo de 5 (cinco) anos após o término da pesquisa, além disso os pesquisadores se comprometem a divulgar os resultados da pesquisa, em

Endereço: Av. Longitudinal UFAL 1, n°1444, térreo do prédio do Centro de Interesse Comunitário (CIC) entre o SINTUFAL
Bairro: Cidade Universitária **CEP:** 57.072-900
UF: AL **Município:** MACEIO
Telefone: (82)3214-1041 **E-mail:** cep@ufal.br

UNIVERSIDADE FEDERAL DE ALAGOAS



Continuação do Parecer: 6.114.614

formato acessível ao grupo ou população que foi pesquisada (Resolução CNS nº 510 de 2016, Artigo 3º, Inciso IV).

Benefícios:

Dos benefícios advindos desse estudo, apontam-se o fato de que, uma vez aplicados os questionários e executadas as entrevistas, pode-se construir uma base de conhecimento, pautada na experiência e perspectiva do usuário final, capaz de fornecer apontamentos para possíveis problemas de usabilidade que podem estar atrelado a dificuldades na navegação, layout confuso, problemas de legibilidade e acessibilidade, etc., e que uma vez mapeados e corrigidos, geram aumento da satisfação do usuário, maior eficácia e eficiência, diminuição de erros e mais rapidez durante uso da ferramenta e, conseqüentemente maior confiança durante o uso do sistema.

Comentários e Considerações sobre a Pesquisa:

Trata-se de uma pesquisa de mestrado em que se pretende aferir a usabilidade da plataforma computacional da avaliação pedagógica do Programa Nacional do Livro e do material Didático (PNLD), a partir de aplicação de questionários e entrevistas com os usuários.

Considerações sobre os Termos de apresentação obrigatória:

Documentos apresentados e apreciados:

PB_INFORMAÇÕES_BÁSICAS_DO_PROJETO_2122173.pdf

projeto_detalhado_.pdf

termo_publicizacao.pdf

TCLE_Corrigido.pdf

cronograma_doc.pdf

orcamento_doc.pdf

Folha_de_Rosto.pdf

autorizacao_de_infraestruturaassinado.pdf

Recomendações:

Não há.

Conclusões ou Pendências e Lista de Inadequações:

APROVADO.

O presente estudo se encontra de acordo com a Resolução 510/16, tendo sido as pendências anteriormente listadas respondidas, conforme carta resposta e documentos reanexados.

Endereço: Av. Longitudinal UFAL 1, nº1444, térreo do prédio do Centro de Interesse Comunitário (CIC) entre o SINTUFAL
Bairro: Cidade Universitária **CEP:** 57.072-900
UF: AL **Município:** MACEIO
Telefone: (82)3214-1041 **E-mail:** cep@ufal.br

UNIVERSIDADE FEDERAL DE ALAGOAS



Continuação do Parecer: 6.114.614

Considerações Finais a critério do CEP:

Protocolo Aprovado

Prezado (a) Pesquisador (a), lembre-se que, segundo a Res. CNS 466/12 e sua complementar 510/2016:

O participante da pesquisa tem a liberdade de recusar-se a participar ou de retirar seu consentimento em qualquer fase da pesquisa, sem penalização alguma e sem prejuízo ao seu cuidado e deve receber cópia do TCLE, na íntegra, assinado e rubricado pelo (a) pesquisador (a) e pelo (a) participante, a não ser em estudo com autorização de declínio;

V.S^a. deve desenvolver a pesquisa conforme delineada no protocolo aprovado e descontinuar o estudo somente após análise das razões da descontinuidade por este CEP, exceto quando perceber risco ou dano não previsto ao sujeito participante ou quando constatar a superioridade de regime oferecido a um dos grupos da pesquisa que requeiram ação imediata;

O CEP deve ser imediatamente informado de todos os fatos relevantes que alterem o curso normal do estudo. É responsabilidade do pesquisador assegurar medidas imediatas adequadas a evento adverso ocorrido e enviar notificação a este CEP e, em casos pertinentes, à ANVISA;

Eventuais modificações ou emendas ao protocolo devem ser apresentadas ao CEP de forma clara e sucinta, identificando a parte do protocolo a ser modificada e suas justificativas. Em caso de projetos do Grupo I ou II apresentados anteriormente à ANVISA, o pesquisador ou patrocinador deve enviá-las também à mesma, junto com o parecer aprovatório do CEP, para serem juntadas ao protocolo inicial;

Seus relatórios parciais e final devem ser apresentados a este CEP, inicialmente após o prazo determinado no seu cronograma e ao término do estudo. A falta de envio de, pelo menos, o relatório final da pesquisa implicará em não recebimento de um próximo protocolo de pesquisa de vossa autoria.

O cronograma previsto para a pesquisa será executado caso o projeto seja APROVADO pelo Sistema CEP/CONEP, conforme Carta Circular nº. 061/2012/CONEP/CNS/GB/MS (Brasília-DF, 04 de maio de 2012).

Este parecer foi elaborado baseado nos documentos abaixo relacionados:

Tipo Documento	Arquivo	Postagem	Autor	Situação
Informações	PB_INFORMAÇÕES_BÁSICAS_DO_P	24/05/2023		Aceito

Endereço: Av. Longitudinal UFAL 1, n°1444, térreo do prédio do Centro de Interesse Comunitário (CIC) entre o SINTUFAL
Bairro: Cidade Universitária **CEP:** 57.072-900
UF: AL **Município:** MACEIO
Telefone: (82)3214-1041 **E-mail:** cep@ufal.br

UNIVERSIDADE FEDERAL DE ALAGOAS



Continuação do Parecer: 6.114.614

Básicas do Projeto	ETO_2122173.pdf	00:42:18		Aceito
Projeto Detalhado / Brochura Investigador	projeto_detalhado_.pdf	24/05/2023 00:41:55	ALENILTON SANTOS SILVA	Aceito
Outros	carta_resposta.pdf	24/05/2023 00:40:40	ALENILTON SANTOS SILVA	Aceito
Outros	termo_publicizacao.pdf	24/05/2023 00:38:57	ALENILTON SANTOS SILVA	Aceito
TCLE / Termos de Assentimento / Justificativa de Ausência	TCLE_Corrigido.pdf	24/05/2023 00:38:11	ALENILTON SANTOS SILVA	Aceito
Cronograma	cronograma_doc.pdf	21/04/2023 13:47:23	ALENILTON SANTOS SILVA	Aceito
Orçamento	orcamento_doc.pdf	21/04/2023 13:47:02	ALENILTON SANTOS SILVA	Aceito
Folha de Rosto	Folha_de_Rosto.pdf	21/04/2023 13:46:32	ALENILTON SANTOS SILVA	Aceito
Declaração de Instituição e Infraestrutura	autorizacao_de_infraestrutura_assinado. pdf	12/04/2023 16:59:56	ALENILTON SANTOS SILVA	Aceito

Situação do Parecer:

Aprovado

Necessita Apreciação da CONEP:

Não

MACEIO, 13 de Junho de 2023

Assinado por:

Carlos Arthur Cardoso Almeida
(Coordenador(a))

Endereço: Av. Longitudinal UFAL 1, n°1444, térreo do prédio do Centro de Interesse Comunitário (CIC) entre o SINTUFAL
Bairro: Cidade Universitária **CEP:** 57.072-900
UF: AL **Município:** MACEIO
Telefone: (82)3214-1041 **E-mail:** cep@ufal.br