

**INSTITUTO
FEDERAL**
Brasília

INSTITUTO FEDERAL DE BRASÍLIA
Campus Brasília
Curso Superior de Tecnologia em Sistemas para Internet

ÂNGELO DE OLIVEIRA ARAÚJO
HAAYRON MATHEUS TAVARES QUEIROZ

IFB + ESTÁGIOS: SOFTWARE DE VAGAS DE ESTÁGIO

Brasília
2025

ÂNGELO DE OLIVEIRA ARAÚJO
HAAYRON MATHEUS TAVARES QUEIROZ

IFB + ESTÁGIOS: SOFTWARE DE VAGAS DE ESTÁGIO

Trabalho apresentado ao Curso Superior de Tecnologia em Sistemas para Internet do Instituto Federal de Brasília como requisito parcial para obtenção do grau de Tecnólogo em Sistemas para Internet.

Orientador: Me. Fernando Wagner Brito Hortencio Filho

Brasília
2025

ÂNGELO DE OLIVEIRA ARAÚJO
HAAYRON MATHEUS TAVARES QUEIROZ

IFB + ESTÁGIOS: SOFTWARE DE VAGAS DE ESTÁGIO

Trabalho de conclusão de curso de graduação apresentado à Coordenação do Curso Superior de Tecnologia em Sistemas para Internet do Instituto Federal de Brasília Campus Brasília, como requisito parcial para obtenção do título de Tecnólogo em Sistemas para Internet.

Aprovado em: _____ de _____ de _____.

BANCA EXAMINADORA

Prof. Me. Fernando Wagner Brito Hortencio Filho
Computação/IFB

Prof. Dr. Dauster Souza Pereira
Computação/IFB

Prof. Dra. Paula Felipe Schlemper de Oliveira
Computação/IFB

Brasília
2025

Q3 Queiroz, Haayron Matheus Tavares.
IFB + estágios: software de vagas de estágio / Haayron Matheus Tavares, Ângelo de Oliveira Araújo. — Brasília, 2025.
98 f. : il. color.

Orientador: Fernando Wagner Brito Hortêncio Filho.
Trabalho de conclusão de curso (Graduação) — Instituto Federal de Educação, Ciência e Tecnologia de Brasília, Tecnologia em Sistemas para Internet, 2025.

1. Estágio. 2. Desenvolvimento de software. 3. Sistema de gerenciamento. I. Araújo, Ângelo de Oliveira. II. Hortêncio Filho, Fernando Wagner Brito (orient.). III. Título.

CDU: 004

Ficha catalográfica elaborada com os dados fornecidos pelos autores.

AGRADECIMENTOS

Haayron:

Primeiramente, desejo destacar que a realização deste trabalho foi possível graças ao apoio, incentivo e dedicação de pessoas especiais que estiveram ao meu lado em diferentes etapas dessa jornada acadêmica. Cada um de vocês, de diferentes maneiras, contribuiu para o sucesso deste trabalho. Agradeço à minha família, em especial à Luisa Maria, e pelo suporte e compreensão nos momentos de maior dificuldade e pela confiança depositada em minha trajetória.

Ao meu querido amigo, Leonardo Santos, cuja amizade sincera e apoio foram fundamentais em minha vida. Obrigado por compartilhar momentos que tornaram este percurso mais leve e significativo. À Yara Helena, pelo carinho, compreensão e companheirismo.

À equipe da Ciplan, expresse minha profunda gratidão pelo suporte e encorajamento durante minha formação.

Ângelo:

Primeiramente, um agradecimento especial aos familiares, que sempre acreditaram em mim e me deram forças para continuar: Fernanda Rosa, Airton Ribeiro, Pablo Aleksander e Heito Sá, que estiveram ao meu lado em todos os momentos, me dando amor, suporte e inspiração para alcançar este objetivo.

Aos colegas de trabalho da Global Red, que me apoiaram e contribuíram diretamente para minha formação, me motivando a superar desafios e a crescer tanto pessoal quanto profissionalmente. A cada um de vocês, minha mais profunda gratidão.

Haayron e Ângelo:

Ao professor Fernando Wagner Brito Hortencio Filho, nosso orientador, por toda a paciência, incentivo e orientação ao longo deste percurso. Seu apoio contribuiu muito para o desenvolvimento deste trabalho e para nossa formação profissional e acadêmica.

E, por fim, aos amigos que conquistamos durante o curso: Rafaela Barros, Bárbara Silva, Anna Carol e Micaelly Matos, que tornaram essa jornada muito mais leve, compartilhando desafios, aprendizados e momentos inesquecíveis. A amizade de vocês foi o que permitiu que seguíssemos em frente, e levaremos isso para sempre durante os próximos anos de nossas vidas.

RESUMO

O estágio visa à preparação dos estudantes para o mercado de trabalho, promovendo o aprendizado prático alinhado à programação curricular de cada curso. Apesar disso, no Instituto Federal de Brasília (IFB), campus Brasília, o processo atual de divulgação e gestão de vagas apresenta desafios significativos. A divulgação ocorre, predominantemente, por meio de e-mails, grupos de *WhatsApp* e documentos em formato de PDFs enviados semanalmente no site da instituição, o que dificulta a centralização das informações, a filtragem eficiente de vagas compatíveis com o perfil dos alunos e o acompanhamento pela coordenação de estágios. Essa abordagem dispersa pode levar à perda de oportunidades, à baixa participação dos estudantes em estágios e à dificuldade em monitorar e organizar as vagas de maneira eficaz. Diante desse cenário, este trabalho apresenta o desenvolvimento e a proposta de um software chamado "IFB + Estágios", projetado para superar as limitações do modelo atual de divulgação de vagas de estágio. O sistema visa centralizar as informações das vagas de estágio em uma única plataforma, permitindo a filtragem personalizada dos requisitos indicados pelas empresas, além de oferecer notificações sobre novas oportunidades com base nas configurações de perfil dos usuários. Para a coordenação de estágios, o software simplifica a gestão e distribuição de vagas, otimizando os processos administrativos e facilitando a organização das ofertas e acompanhamentos. Espera-se que o "IFB + Estágios" amplie o acesso às vagas, melhore a experiência dos estudantes ao buscar estágios e aumente sua participação nessas atividades. Dessa forma, o sistema busca promover um fluxo mais organizado e eficiente de oportunidades de estágio para os alunos. Com isso, o software tem potencial para contribuir significativamente para o desenvolvimento profissional dos alunos e para o fortalecimento do papel do IFB na formação de futuros profissionais qualificados.

Palavras-chave: estágio; sistema de gerenciamento de estágio; desenvolvimento de software.

ABSTRACT

Internships aim to prepare students for the job market by promoting practical learning aligned with each courses curriculum. However, at the Instituto Federal de Brasília (IFB), Brasília campus, the current process of internship vacancy dissemination and management presents significant challenges. Job postings are primarily shared through emails, WhatsApp groups, and weekly PDF documents uploaded to the institution's website. This fragmented approach makes it difficult to centralize information, efficiently filter vacancies that match students profiles, and facilitate monitoring by the internship coordination team. As a result, students may miss opportunities, engage less in internships, and struggle with tracking and organizing available positions effectively. Given this scenario, this study presents the development and proposal of a software solution called "IFB + Estágios," designed to overcome the limitations of the current internship vacancy dissemination model. The system aims to centralize internship information into a single platform, allowing personalized filtering based on the requirements set by companies and providing notifications about new opportunities according to users profile settings. For the internship coordination team, the software simplifies the management and distribution of vacancies, optimizes administrative processes, and facilitates the organization of offers and follow-ups. The "IFB + Estágios" system is expected to expand access to internship opportunities, enhance students experience in searching for placements, and increase their participation in such activities. Ultimately, the system seeks to establish a more organized and efficient internship opportunity flow for students. By doing so, the software has the potential to significantly contribute to students professional development and strengthen IFB role in training future qualified professionals.

Keywords: internship; internship management system; software development.

LISTA DE FIGURAS

1	Ciclo de Vida Scrum	32
2	<i>Business Process Modeling</i> (BPM) AS-IS	42
3	BPM TO BE	43
4	Protótipo da Página inicial - Usuário.	45
5	Protótipo da Página de Feed de Vagas.	46
6	Arquitetura do projeto IFB + Estágios	49
7	Diagrama de Casos de Uso Macro - Usuário	57
8	Diagrama de Casos de Uso Macro - Administrador	58
9	Diagrama de Classes	68
10	Diagrama de Entidade e Relacionamento	69
11	Diagrama Relacional	70
12	Código responsável pela consulta de vagas na <i>Application Programming Interface</i> (API)	72
13	Trecho do código responsável por inserir as vagas de estágio.	74
14	Pergunta 1 - O aplicativo informa claramente o que está acontecendo (ex.: carregamento, progresso de uma tarefa).	88
15	Pergunta 2 - Eu acho que gostaria de usar este sistema com frequência.	88
16	Pergunta 3 - Eu achei o sistema desnecessariamente complexo.	88
17	Pergunta 4 - Eu achei o sistema fácil de usar.	89
18	Pergunta 5 - Eu acho que precisaria da ajuda de uma pessoa com conhecimentos técnicos para conseguir usar este sistema.	89
19	Pergunta 6 - Eu achei as várias funções deste sistema bem integradas.	89
20	Pergunta 7 - Eu achei que havia muita inconsistência neste sistema.	90
21	Pergunta 8 - Eu imagino que a maioria das pessoas aprenderia a usar este sistema rapidamente.	90
22	Pergunta 9 - Eu achei o sistema muito confuso.	90
23	Pergunta 10 - Eu me senti muito confiante usando o sistema.	91
24	Pergunta 11 - Eu precisei aprender várias coisas antes de conseguir usar o sistema.	91
25	Pergunta 12 - As palavras, ícones e instruções usadas no aplicativo são fáceis de entender e correspondem ao seu conhecimento do mundo real.	91
26	Pergunta 13 - Você consegue desfazer ou refazer ações facilmente no aplicativo (ex.: botões "voltar" ou "desfazer").	92
27	Pergunta 14 - Os botões, menus e elementos visuais do aplicativo possuem um padrão consistente em todas as telas.	92

28	Pergunta 15 - O aplicativo ajuda a evitar erros, fornecendo avisos antes de ações importantes ou perigosas (ex.: confirmação antes de excluir algo).	92
29	Pergunta 16 - Os ícones, rótulos e funcionalidades no aplicativo são facilmente reconhecíveis sem a necessidade de você memorizar ações anteriores.	93
30	Pergunta 17 - Você consegue realizar suas tarefas rapidamente e com facilidade no aplicativo, mesmo se for a primeira vez que o utiliza.	93
31	Pergunta 18 - O design do aplicativo é simples, sem informações desnecessárias que possam causar confusão.	93
32	Pergunta 19 - Se você cometeu um erro no aplicativo, ele forneceu uma mensagem clara para ajudá-lo a corrigir o problema.	94
33	Pergunta 20 - O aplicativo oferece instruções ou uma seção de ajuda clara caso você precise de orientação para realizar suas tarefas.	94

LISTA DE TABELAS

1	Comparação por Funcionalidade de Plataformas de Estágio e Emprego	24
2	Análise <i>Strengths, Weaknesses, Opportunities, and Threats (SWOT)</i> do sistema "IFB + Estágios"	39
3	Matriz de rastreabilidade entre regras de negócio, requisitos funcionais e não funcionais	55
4	Resultados da Escala de Usabilidade do Sistema (SUS)	78
5	Resultados das Heurísticas de Usabilidade de Nielsen	79

LISTA DE ABREVIATURAS E SIGLAS

API	<i>Application Programming Interface</i>	71
BPM	<i>Business Process Modeling</i>	41
CI/CD	<i>Continuous Integration/Continuous Deployment</i>	28
CSS	<i>Cascading Style Sheets</i>	25
IFB	<i>Instituto federal de Brasília</i>	11
HTML	<i>HyperText Markup Language</i>	39
IFB	Instituto Federal de Brasília	11
SQL	<i>Structured Query Language</i>	48
SWOT	<i>Strengths, Weaknesses, Opportunities, and Threats</i>	32
REST	<i>Representational State Transfer</i>	74
UML	<i>Unified Modeling Language</i>	56
HTTP	<i>Hypertext Transfer Protocol</i>	74
JSON	<i>JavaScript Object Notation</i>	74
SUS	<i>System Usability Scale</i>	34

SUMÁRIO

1	INTRODUÇÃO	11
1.1	Justificativa	12
1.2	Tema	12
1.3	Problema	12
1.4	Objetivo Geral	13
1.5	Objetivos específicos	13
1.6	Estrutura do TCC	13
1.7	Classificação do Trabalho	15
2	CONCEITOS GERAIS E REVISÃO DE LITERATURA	17
2.1	Conceitos acerca da atividade de estágio	17
2.1.1	<i>Regulamento de Estágio do IFB</i>	18
2.1.2	<i>Penalidades para Organizações em Desconformidade com a Legislação de Estágio</i>	19
2.1.2.1	<i>Multas Administrativas e Sanções Trabalhistas</i>	20
2.1.2.2	<i>Penalidades Relativas à Duração do Estágio</i>	20
2.1.3	<i>Importância do Estágio para a Formação Profissional dos Discentes</i>	20
2.2	Análise Comparativa de Trabalhos Correlatos	22
2.3	Tecnologias Utilizadas	25
2.3.1	<i>HTML, CSS e JavaScript</i>	25
2.3.2	<i>React</i>	25
2.3.3	<i>Figma</i>	26
2.3.4	<i>SQL Server</i>	27
2.3.5	<i>Mulesoft</i>	27
2.3.6	<i>Azure DevOps</i>	28
2.3.7	<i>Git e GitHub</i>	29
2.3.8	<i>Overleaf</i>	29
3	METODOLOGIA	31
3.1	Pesquisa Bibliográfica	33
3.2	Metodologia Baseada no Design Thinking	34
3.3	Levantamento de Requisitos e diálogo com <i>Stakeholders</i>	35
3.4	Questionários para Avaliação do Sistema	35
3.4.1	<i>Questionário 1: Avaliação da Usabilidade com o SUS</i>	35
3.4.2	<i>Questionário 2: Avaliação Baseada nas Heurísticas de Nielsen</i>	36
3.5	<i>Metodologia Análise SWOT</i>	37
3.6	<i>Metodologia Scrum</i>	40

3.6.1	<i>Cerimônias do Scrum</i>	40
3.7	Análise de Processos com BPM	41
3.7.1	<i>BPM As-Is</i>	41
3.7.2	<i>BPM To-Be</i>	42
3.7.3	<i>Integração com Outras Metodologias</i>	43
3.8	Desenvolvimento de Protótipos	43
3.8.1	<i>Descrição das Principais Telas do Sistema</i>	45
4	DESENVOLVIMENTO	47
4.1	Arquitetura do sistema	47
4.2	Análise de Requisitos do Sistema	50
4.2.1	<i>Regras de Negócio</i>	50
4.2.2	<i>Requisitos Funcionais</i>	52
4.2.3	<i>Requisitos Não Funcionais</i>	53
4.3	Matriz de Rastreabilidade	54
4.4	Diagramas	55
4.4.1	<i>Diagrama de Casos de Uso Macro</i>	56
4.4.2	<i>Especificação de Casos de Uso</i>	59
4.4.3	Diagrama de Classe	68
4.4.4	<i>Diagrama Entidade-Relacionamento</i>	69
4.4.5	<i>Diagrama Relacional</i>	69
4.5	Implementação do código e testes	70
4.5.1	<i>Principais Componentes React</i>	72
5	RESULTADOS	77
5.1	Validação dos Resultados e Atendimento aos Objetivos	77
5.2	Análise dos Resultados	79
5.3	Ponderações	80
5.4	Limitações e Melhorias Futuras	81
6	CONSIDERAÇÕES FINAIS	83
	Referências	84
	Apêndice	87

1

INTRODUÇÃO

O estágio é uma atividade fundamental no percurso acadêmico dos estudantes. Ele permite que os estudantes apliquem conhecimentos adquiridos ao longo do percurso acadêmico em situações reais de trabalho, contribuindo para o seu desenvolvimento profissional e pessoal.

Franks e Oliver (2012) afirmam que incentivar os alunos através de experiências de estágio pode apoiar a aprendizagem desde a graduação, ao unir ensino teórico e prático. Nesse sentido, para os alunos, o estágio é uma ótima oportunidade de ganhar experiência prática, desenvolver habilidades profissionais e estar inserido no ambiente profissional no qual proporciona conexões no mercado de trabalho. A prática do estágio, do ponto de vista das instituições de ensino, é uma forma de enriquecer o currículo, que por sua vez fortalece a relação da instituição com o setor produtivo.

No entanto, encontrar oportunidades de estágio que se alinhem com os objetivos educacionais e as aspirações dos alunos pode ser um desafio. Muitas vezes, os estudantes enfrentam dificuldades como a falta de informações claras sobre vagas disponíveis. Atualmente, o Instituto Federal de Educação, Ciência e Tecnologia de Brasília Instituto Federal de Brasília (IFB), campus Brasília, utiliza grupos de *WhatsApp*, documentos em formato de PDF no site do IFB, além do envio de e-mails para a divulgação de vagas para os alunos, fazendo com que os estudantes recebam diversas vagas que não correspondem ao seu perfil, o que dificulta a filtragem de vagas e busca de informações. De acordo com a Lei nº 11.788/2008, o estágio visa a preparação dos estudantes para o mercado de trabalho, através de incentivo ao aprendizado e de práticas compatíveis com a programação curricular de cada curso (BRASIL, 2008). Neste contexto, o presente trabalho visa estabelecer a relevância do desenvolvimento do sistema de estágios "IFB + Estágios", no contexto do IFB, Campus Brasília, como ferramenta para enfrentar os desafios na busca por oportunidades de estágio, e possibilitar que os alunos adquiram experiência.

A implementação de um sistema eficiente de gerenciamento de estágios no IFB é muito importante, não apenas para a administração eficaz desses programas, mas também para potencializar o impacto positivo dos estágios na empregabilidade dos estudantes.

1.1 Justificativa

Conforme destacado por Gault, Leach e Duey (2010), estudos indicam que estudantes com experiência de estágio têm significativamente mais oportunidades de emprego após a graduação e tendem a receber ofertas de salários iniciais mais altas, evidenciando a importância do acesso às boas oportunidades de estágio. Este cenário ressalta a necessidade do sistema "IFB + Estágios", que visa otimizar a gestão e a eficácia dos estágios, beneficiando tanto os estudantes quanto os empregadores. No contexto atual, onde o mercado de trabalho enfrenta transformações rápidas e a necessidade de qualificação profissional é cada vez mais evidente, o papel de iniciativas como o "IFB + Estágios" torna-se ainda mais crucial. Como apontado em um estudo recente:

A expansão e diversificação do ensino superior têm possibilitado o acesso a esse nível de ensino a pessoas que dificilmente o teriam antes da expansão, contribuindo para qualificar a inserção destas no mercado de trabalho e permitir o acesso a vagas que normalmente não alcançariam sem o ensino superior, aumentando assim sua empregabilidade. Martins *et al.*, 2017

Isso enfatiza a importância de um sistema que conecte estudantes às oportunidades de estágio, assim, contribuindo para a sua preparação para um mercado de trabalho em constante evolução.

1.2 Tema

O tema central deste trabalho de conclusão de curso é o desenvolvimento de um software para facilitar a interação dos alunos com oportunidades de estágio no IFB, campus Brasília. Esse tema explora a criação de uma solução tecnológica proposta, o software "IFB + Estágios", que tem como objetivo aprimorar o acesso dos alunos às informações sobre vagas de estágio no IFB, campus Brasília, e, assim, contribuir para sua inclusão no mercado de trabalho e desenvolvimento profissional. A proposta também contempla a análise das funcionalidades do software, seus possíveis benefícios para os alunos e a instituição, além dos desafios associados à sua implementação.

1.3 Problema

Como desenvolver uma solução tecnológica para substituir os métodos atuais de divulgação de oportunidades de estágio no IFB campus Brasília, que atualmente dependem de grupos de aplicativos de mensagem instantânea, envio de e-mails e documentos semanais em formato de PDF, lançado no site do IFB, por um sistema mais eficiente que centralize as informações, ofereça filtragem personalizada de vagas, atendendo às necessidades específicas de cada curso?

1.4 Objetivo Geral

Elaborar uma solução tecnológica para facilitar o gerenciamento de vagas de estágio no IFB, Campus Brasília, visando unificar as informações sobre oferta de vagas, personalizadas de acordo com o perfil dos alunos

1.5 Objetivos específicos

Como objetivos específicos deste trabalho, citam-se:

1. Identificar os desafios enfrentados por alunos e coordenação de estágios no IFB, campus Brasília.
2. Desenvolver o software "IFB + Estágios", integrando requisitos como compatibilidade móvel e usabilidade eficiente.
3. Testar e validar a usabilidade do sistema com alunos e coordenadores, visando melhorias.
4. Avaliar o impacto do sistema na otimização da gestão de estágios e na experiência dos alunos.
5. Propor um plano de expansão do projeto, identificando áreas para futuros estudos e melhorias.

1.6 Estrutura do TCC

A estrutura deste Trabalho de Conclusão de Curso (TCC) segue uma organização lógica que proporcionará aos leitores uma compreensão completa e clara do projeto "IFB + Estágios". A seguir, apresenta-se a estrutura detalhada do TCC:

Capítulo 1: Introdução

No primeiro capítulo, é abordado o contexto do projeto, que trata da divulgação de vagas de estágio no IFB, campus Brasília. O problema central identificado é a ausência de uma centralização e personalização eficiente da forma como as informações chegam para os alunos, já que as oportunidades são atualmente divulgadas semanalmente por meio de documentos em formato de PDF disponibilizados no site do IFB, além de grupos de aplicativos de mensagem instantânea, e através do envio de e-mails aos alunos. Este capítulo também estabelece o objetivo geral do trabalho e os objetivos específicos que orientaram o desenvolvimento da solução proposta, além de apresentar a estrutura geral do TCC.

Capítulo 2: Conceitos Gerais e Revisão da Literatura

No segundo capítulo, são abordados os conceitos fundamentais e a revisão bibliográfica utilizada para fundamentar este trabalho. Primeiramente, discute-se os conceitos acerca

das atividades de estágio e a importância dos estágios no contexto da formação acadêmica e profissional, destacando sua relevância para o mercado de trabalho. Em seguida, são analisados os sistemas de gerenciamento de estágios e emprego existentes, assim como as metodologias de desenvolvimento de software e tecnologias que serão utilizadas. Esses conceitos teóricos embasam a proposta de desenvolvimento do sistema "IFB + Estágios".

Capítulo 3: Metodologia

O terceiro capítulo detalha a metodologia adotada para o desenvolvimento da proposta do sistema "IFB + Estágios". A metodologia envolve um estudo de caso específico para o IFB campus Brasília, com foco na análise do processo atual de divulgação de vagas. Serão descritos os métodos de coleta de dados, como entrevistas e questionários aplicados aos alunos e à coordenação de estágios, para levantar as necessidades e expectativas. Este capítulo também aborda o processo de definição dos requisitos do sistema, a escolha das ferramentas tecnológicas e o desenvolvimento do protótipo, bem como a abordagem para testes e validação da proposta.

Capítulo 4: Desenvolvimento

O quarto capítulo apresenta o desenvolvimento técnico da solução proposta. Primeiramente, são definidos os requisitos funcionais e não funcionais do sistema, são apresentados diagramas do projeto, como o Diagrama de Entidade-Relacionamento (DER), que define a estrutura do banco de dados, Diagrama relacional, e os Diagramas de Casos de Uso, que ilustram as interações do usuário com o sistema. Em seguida, são detalhados o design da interface e a arquitetura do sistema "IFB + Estágios", com explicações sobre os módulos principais e suas funcionalidades.

Capítulo 5: Resultados Alcançados e Discussão

No quinto capítulo, são discutidos os resultados alcançados com a implementação do sistema "IFB + Estágios". O foco está nos benefícios que o sistema trará, como a melhoria no acesso às vagas de estágio, maior personalização das oportunidades. Também é apresentada uma comparação entre o modelo atual de divulgação de vagas e o sistema proposto, destacando os ganhos de eficiência e organização. Além disso, este capítulo aborda as possíveis dificuldades na implementação do sistema, como a adaptação da coordenação e dos alunos ao novo formato.

Capítulo 6: Considerações Finais

O sexto capítulo apresenta as considerações finais do trabalho. Nele, é feito um resumo dos principais pontos discutidos ao longo do TCC, ressaltando a relevância de um sistema de gerenciamento de estágios para o IFB - Campus Brasília. São feitas sugestões para trabalhos futuros, como a possível expansão do sistema para outros *campi* do IFB e a implementação de novas funcionalidades. Também é discutida a importância de continuar monitorando e aperfeiçoando o sistema, caso ele seja implementado, para garantir sua eficácia a longo prazo.

1.7 Classificação do Trabalho

A pesquisa deste trabalho é classificada como exploratória e aplicada, com foco no desenvolvimento de um sistema de software voltado para o gerenciamento de estágios no IFB, campus Brasília. O estudo busca investigar os desafios enfrentados pelos alunos na busca por oportunidades de estágio e pela coordenação de estágios no gerenciamento e divulgação eficiente dessas vagas. Além disso, visa explorar as possibilidades de otimização dos processos relacionados à experiência de estágio, destacando a importância dessa etapa para a formação profissional dos estudantes e para a integração entre a instituição de ensino e o mercado de trabalho.

Como parte da fundamentação teórica, foram utilizadas fontes de referência como artigos científicos, livros e publicações acadêmicas, priorizando materiais publicados nos últimos dez anos (a partir de 2014) em bases de dados reconhecidas, como *Google Scholar* e *Oasis.br*. Os principais termos de busca empregados incluíram: "a importância do estágio para a formação dos estudantes", "sistema de gerenciamento de estágios", "*internship management system*", "plataformas de vagas" e "sistema de vagas de estágio". Adicionalmente, estudos mais antigos foram incluídos para oferecer suporte teórico consolidado e estabelecer conexões históricas relevantes com o tema.

A seleção das bases de dados utilizadas na pesquisa foi baseada nos seguintes critérios:

- **Acesso amplo e diversificado:** O *Google Scholar* fornece uma ampla gama de artigos acadêmicos, teses e conferências, garantindo variedade e abrangência.
- **Foco em pesquisas aplicadas:** O *Oasis.br* é uma base relevante para repositórios acadêmicos no Brasil, permitindo encontrar estudos alinhados ao contexto educacional e tecnológico nacional.
- **Disponibilidade e facilidade de acesso:** Ambas as bases são de **acesso aberto**, facilitando a obtenção de materiais sem a necessidade de assinaturas institucionais.

Embora bases como **SciELO**, **Scopus** e **IEEE** sejam amplamente reconhecidas, algumas razões justificam sua não priorização nesta pesquisa:

- **SciELO:** Embora relevante, possui maior foco em *ciências sociais e biomédicas*, o que pode resultar em menor disponibilidade de estudos sobre desenvolvimento de software e usabilidade.
- **Scopus:** É uma base robusta, porém, de acesso pago, o que pode limitar a obtenção de artigos para pesquisadores sem vínculo institucional.
- **IEEE e ACM Digital Library:** São bases altamente técnicas e com publicações de ponta em computação, mas muitas exigem assinaturas ou licenças para acesso completo.

Portanto, a escolha priorizou fontes **de acesso livre, ampla cobertura e alinhadas ao tema da pesquisa**, garantindo uma investigação abrangente e acessível.

Além disso, é importante ressaltar, que este trabalho configura-se como um estudo de caso, uma vez que o sistema "IFB + Estágios" foi desenvolvido especificamente para atender às necessidades do IFB, campus Brasília. O projeto contempla a criação de uma solução de software que centralize as informações de vagas de estágio, permita a filtragem personalizada com base nos perfis dos usuários e automatize notificações sobre novas oportunidades. O objetivo geral é avaliar como o sistema pode impactar positivamente a obtenção de oportunidades de estágio pelos alunos e facilitar a gestão de vagas pela coordenação de estágios.

A metodologia de coleta de dados incluiu a aplicação de questionários e entrevistas com os usuários finais, abrangendo tanto alunos quanto membros da coordenação de estágios. Esses instrumentos foram projetados para investigar a experiência atual dos alunos com os métodos tradicionais de busca de estágios e para entender as expectativas em relação ao sistema proposto. A abordagem adotada é mista, combinando métodos qualitativos e quantitativos. Os dados qualitativos exploraram as percepções e opiniões dos usuários sobre a usabilidade e eficácia do sistema, enquanto os dados quantitativos permitiram medir o impacto do "IFB + Estágios" na melhoria do acesso às vagas e na satisfação geral dos usuários.

Essa combinação metodológica permite uma análise abrangente do impacto do sistema, proporcionando *insights* valiosos sobre os benefícios e limitações da solução proposta. Com isso, o trabalho busca contribuir para a melhoria dos processos de gestão de estágios, fortalecendo o papel do IFB como facilitador do desenvolvimento profissional de seus estudantes.

2

CONCEITOS GERAIS E REVISÃO DE LITERATURA

2.1 Conceitos acerca da atividade de estágio

De acordo com a Lei nº 11.788/08, o estágio é definido como um ato educativo com objetivos claros de integração entre o ambiente escolar e o mundo do trabalho. O texto da legislação afirma:

Art. 1º Estágio é ato educativo escolar supervisionado, desenvolvido no ambiente de trabalho, que visa à preparação para o trabalho produtivo de educandos que estejam frequentando o ensino regular em instituições de educação superior, de educação profissional, de ensino médio, da educação especial e dos anos finais do ensino fundamental, na modalidade profissional da educação de jovens e adultos. (BRASIL, 2008)

Essa definição legal embasa o papel das instituições responsáveis envolvidas na atividade de estágio, que possuem responsabilidades específicas. A prática do estágio envolve a interação entre três principais partes responsáveis por garantir a sua realização: a entidade concedente, a entidade integradora (quando aplicável) e a instituição de ensino. Cada uma dessas partes desempenha um papel fundamental no processo, com responsabilidades distintas, mas interdependentes, que asseguram o cumprimento dos objetivos educativos e legais do estágio.

A **entidade concedente** refere-se à organização, pública ou privada, que oferece a oportunidade de estágio para os estudantes. Essa entidade é responsável por proporcionar um ambiente de trabalho adequado e compatível com os objetivos formativos do estágio. Além disso, deve designar um supervisor interno para acompanhar as atividades do estagiário, garantindo que as tarefas sejam compatíveis com o plano de estágio e promovam o aprendizado prático do aluno. A entidade concedente também tem a obrigação de fornecer infraestrutura apropriada, bem como contratar seguro contra acidentes pessoais para o estagiário.

A **entidade integradora**, por sua vez, atua como um intermediário entre as instituições de ensino e as entidades concedentes, facilitando o processo de organização, oferta e acompanhamento das vagas de estágio. Essa entidade não é obrigatória em todos os estágios, mas sua presença pode ser especialmente útil para simplificar a busca e a gestão das oportunidades. Exemplos comuns incluem empresas especializadas em recrutamento ou plataformas digitais que conectam estagiários às organizações. Além disso, as entidades integradoras ajudam a garantir a conformidade com as exigências legais e especificam suporte operacional para a formalização dos estágios.

Por fim, a **instituição de ensino** é a responsável por supervisionar e orientar o estágio no contexto acadêmico. Cabe à instituição avaliar a adequação das atividades propostas ao currículo do curso, bem como indicar um orientador acadêmico que acompanhará o desenvolvimento do estagiário. A instituição também deve celebrar o termo de compromisso com o estudante e a entidade concedente, avaliar as instalações da empresa para verificar sua capacidade de atender às demandas do estágio e exigir relatórios periódicos para monitorar o aprendizado do aluno. Além disso, a instituição de ensino tem o papel fundamental de garantir que o estágio atenda aos objetivos pedagógicos previstos no curso, promovendo a articulação entre o aprendizado teórico e prático.

Essas três partes formam a base estrutural do estágio, atuando de maneira colaborativa para promover uma experiência educativa supervisionada que contribua para o desenvolvimento acadêmico e profissional dos estudantes.

Além disso, os estagiários podem receber bolsa ou outra forma de contraprestação, como auxílio-transporte e seguro contra acidentes pessoais. A legislação estabelece que a duração máxima do estágio em uma mesma organização é de 2 anos, exceto para pessoas com deficiência, garantindo maior acessibilidade a esse grupo. Também é garantido o direito a um recesso remunerado de 30 dias para estágios com duração igual ou superior a 1 ano, reforçando o compromisso de preservar o bem-estar do estagiário ao longo da sua experiência.

2.1.1 *Regulamento de Estágio do IFB*

O Regulamento de Estágio do IFB (Instituto Federal de Brasília, 2016), reforça as diretrizes da Lei nº 11.788/2008, destacando as especificidades aplicáveis aos estudantes do IFB. O regulamento organiza os estágios em duas modalidades principais:

- **Estágio Obrigatório:** Componente curricular indispensável para a formação do estudante, configurado como requisito obrigatório para a obtenção do diploma.
- **Estágio Não Obrigatório:** Atividade opcional que, embora não componha a carga horária obrigatória do curso, enriquece o percurso formativo do estudante.

O Art. 20 do regulamento estabelece limites para a jornada dos estagiários, de modo a preservar o equilíbrio entre atividades acadêmicas e práticas. Os limites são:

- **Educação Especial e Ensino Fundamental na Modalidade EJA:** Até 4 horas diárias e 20 horas semanais.
- **Ensino Médio e Educação Profissional de Nível Médio:** Até 6 horas diárias e 30 horas semanais.
- **Ensino Superior e Cursos com Alternância Teórico-Prática:** Até 8 horas diárias e 40 horas semanais, conforme previsto no Projeto Pedagógico do Curso (PPC).

Além disso, durante os períodos de avaliação acadêmica, a jornada deve ser reduzida em pelo menos 50%, conforme especificado no Termo de Compromisso.

O regulamento também especifica as responsabilidades das partes envolvidas no estágio:

- **Instituição de Ensino:** Cabe ao IFB zelar pela conformidade dos estágios, avaliando as condições oferecidas pelas partes concedentes, celebrando Termos de Compromisso e designando orientadores para acompanhar as atividades dos estagiários.
- **Parte Concedente:** Deve fornecer instalações adequadas, garantir a supervisão das atividades e cumprir as exigências de segurança, como a contratação de seguro contra acidentes pessoais.
- **Estagiários:** Devem atender aos requisitos de elegibilidade, cumprir a jornada estipulada e realizar as atividades designadas, reportando regularmente as atividades desempenhadas durante o estágio ao orientador.

O regulamento reafirma o papel do estágio como um elemento central na formação integral dos estudantes, promovendo o aprendizado prático em contextos reais de trabalho. Ao enfatizar a supervisão acadêmica e as condições adequadas para a realização do estágio, o IFB assegura que essa experiência seja enriquecedora tanto para o desenvolvimento profissional dos alunos quanto para o fortalecimento das parcerias institucionais.

Além disso, as diretrizes do IFB incentivam a reflexão crítica sobre as atividades desempenhadas durante o estágio, estimulando os alunos a articularem suas vivências práticas com os conhecimentos teóricos adquiridos no curso.

2.1.2 Penalidades para Organizações em Desconformidade com a Legislação de Estágio

A legislação também prevê penalidades para organizações que mantêm estagiários em desconformidade com seus requisitos. Quando organizações desrespeitam essas diretrizes, estão sujeitas a diversas penalidades que podem comprometer tanto sua estabilidade jurídica quanto sua reputação.

Uma das consequências mais graves do descumprimento da legislação é a possibilidade de descaracterização do estágio. Caso não sejam cumpridos os requisitos legais, como a celebração do Termo de Compromisso de Estágio ou a supervisão adequada das atividades, a relação entre estagiário e organização poderá ser interpretada como vínculo empregatício. Nesse caso, a organização será obrigada a formalizar o vínculo por meio da assinatura da carteira de trabalho do estagiário. Isso implica o pagamento retroativo de todos os direitos trabalhistas previstos pela Consolidação das Leis do Trabalho (CLT), incluindo FGTS, INSS, 13º salário, férias proporcionais e aviso prévio. Essa penalidade pode gerar um ônus financeiro significativo para a organização, especialmente em casos de descumprimentos reiterados ou em situações envolvendo múltiplos estagiários.

2.1.2.1 *Multas Administrativas e Sanções Trabalhistas*

A falta de cumprimento de obrigações básicas, como a contratação de seguro contra acidentes pessoais ou a designação de supervisores para acompanhamento do estágio, também pode acarretar multas administrativas impostas pelos órgãos competentes, como o Ministério do Trabalho e Emprego.

2.1.2.2 *Penalidades Relativas à Duração do Estágio*

A Lei n.º 11.788/2008 estabelece um limite de dois anos para a duração do estágio na mesma organização, exceto para pessoas com deficiência. Caso esse limite seja excedido, a relação pode ser interpretada como vínculo empregatício. Além das implicações financeiras, o descumprimento pode prejudicar o planejamento interno da organização, comprometendo sua relação com instituições de ensino.

2.1.3 *Importância do Estágio para a Formação Profissional dos Discentes*

O estágio é uma atividade fundamental no processo de formação profissional dos estudantes, pois proporciona uma oportunidade única para a aplicação prática dos conhecimentos adquiridos em sala de aula. De acordo com Machineski *et al.* (2011), "o estágio, seja ele profissional ou de iniciação científica, é uma das principais oportunidades que o estudante tem para poder se preparar para o mercado de trabalho".

Além disso, as instituições de ensino têm um papel essencial no incentivo à realização de estágios. Conforme salientado pelos autores, "o estágio extracurricular é incentivado pelas instituições de ensino e, geralmente, faz a diferença na capacitação do futuro profissional". Essa experiência permite ao estudante desenvolver habilidades que serão exigidas no ambiente de trabalho, além de facilitar sua inserção no mercado.

Os benefícios do estágio vão além do estudante, beneficiando também as empresas, que podem usar essa experiência como uma estratégia eficiente para recrutar e selecionar talentos. Segundo Machineski *et al.* (2011), "o estágio constitui-se num eficaz sistema de recrutamento e seleção de futuros profissionais, facilitando a descoberta de novos talentos e assegurando a formação de recursos humanos qualificados".

Conforme estabelece a Lei de Diretrizes e Bases da Educação Nacional (LDB), a formação profissional deve preparar os estudantes para o exercício pleno de suas profissões e para a vida produtiva e social, articulando os saberes teóricos e práticos (Brasil, 1996). Nesse contexto, o estágio emerge como uma ponte que conecta a teoria acadêmica à prática no ambiente profissional, permitindo que os alunos desenvolvam habilidades essenciais para sua atuação futura. A LDB destaca, ainda, a importância do acompanhamento pedagógico no estágio, garantindo que ele esteja alinhado aos objetivos formativos e às diretrizes curriculares, consolidando sua relevância no processo educativo.

Perspectivas teóricas como a de David Kolb (2014) em sua teoria do aprendizado experiencial, reforçam a relevância do estágio na formação acadêmica e profissional. Segundo Kolb, o aprendizado se dá por meio de um ciclo contínuo de experiência concreta, reflexão, generalização e aplicação. Nesse contexto, o estágio surge como um ambiente privilegiado para o aprendizado ativo, em que os estudantes podem aplicar teorias aprendidas no ambiente acadêmico em situações reais, promovendo o aprendizado significativo e consolidando competências práticas.

Complementando essa visão, John Dewey (2012) em sua obra clássica *Democracy and Education*, argumenta que a educação deve estar profundamente conectada às experiências práticas e sociais do indivíduo. Assim, o estágio não é apenas uma ferramenta para adquirir habilidades técnicas, mas também um meio de integrar os estudantes ao tecido social e profissional, promovendo uma formação holística e alinhada às demandas contemporâneas.

Além de proporcionar a aplicação prática de conhecimentos teóricos, o estágio desempenha um papel central no desenvolvimento da adaptabilidade de carreira dos estudantes universitários. Segundo Bento (2013), a adaptabilidade de carreira é um constructo psicossocial que reflete a prontidão e os recursos individuais para enfrentar as mudanças e desafios no ambiente de trabalho. Esse conceito ganha especial relevância no cenário atual, onde as transformações tecnológicas e sociais têm tornado o mercado de trabalho cada vez mais dinâmico e competitivo.

Os resultados apresentados por Bento destacam que o estágio vai além de ser um instrumento de aprendizado técnico. Ele também se configura como um importante mecanismo para o desenvolvimento de habilidades psicossociais e interpessoais, tais como resiliência, adaptabilidade e confiança, contribuindo para a capacidade de lidar com incertezas no contexto profissional. De acordo com os autores, essa exposição prática ajuda os estudantes a reconhecerem suas aptidões e limitações, promovendo um autoconhecimento que é essencial para a tomada de decisões de carreira. Além disso, os estagiários têm a chance de se familiarizar com as exigências do mercado, ajustar suas expectativas profissionais e adquirir competências técnicas e comportamentais que aumentam sua empregabilidade.

Ademais, o estágio contribui para a construção de redes de relacionamento profissional, conhecidas como *networking*. Esse tipo de interação não apenas facilita a inserção no mercado de trabalho, mas também promove a troca de conhecimentos e a criação de oportunidades futuras. Dessa forma, o estágio pode ser visto como um catalisador que conecta os objetivos educacionais dos estudantes às demandas do mercado.

O impacto positivo do estágio se estende também à saúde mental e emocional dos estudantes. A experiência prática contribui para reduzir a ansiedade e as incertezas relacionadas ao ingresso no mercado de trabalho, ao mesmo tempo que reforça a autoconfiança e a motivação para atingir metas profissionais.

Além disso, a vivência em ambientes colaborativos, como os grupos de trabalho discutidos por Cohen *et al.* (1993), oferece uma plataforma para que os estudantes desenvolvam

habilidades interpessoais, como trabalho em equipe, comunicação e resolução de problemas. Ainda de acordo com os autores, equipes autogerenciadas têm um impacto significativo na integração de novos membros, promovendo aprendizado coletivo, inovação organizacional e uma cultura de responsabilidade compartilhada. Quando estagiários são inseridos em grupos de trabalho bem-estruturados, essa dinâmica não só contribui para o desenvolvimento individual dos estudantes, mas também reforça competências sociais e interpessoais, como comunicação, cooperação e resolução de problemas.

A integração de estagiários em grupos de trabalho bem-estruturados não só contribui para seu desenvolvimento individual, mas também amplia a capacidade organizacional de inovação e solução de problemas. Essa dinâmica ressalta a importância de as empresas e instituições de ensino prepararem os estudantes para trabalharem em equipe, promovendo um alinhamento mais próximo entre as expectativas do mercado de trabalho e as competências adquiridas durante o estágio.

Dessa forma, o estágio vai além da simples aplicação prática de conhecimentos, funcionando como uma ponte entre a teoria acadêmica e a realidade corporativa, ao mesmo tempo que fortalece habilidades sociais e interpessoais essenciais para o sucesso no ambiente profissional.

2.2 Análise Comparativa de Trabalhos Correlatos

Com o objetivo de traçar diferenciais e nortear o desenvolvimento do protótipo do sistema "IFB + Estágios", foi realizada uma análise comparativa entre diferentes plataformas de gerenciamento de estágios e empregos, tanto de uso acadêmico quanto generalista. Essa análise buscou identificar as funcionalidades mais relevantes, pontos fortes e limitações de cada sistema, fornecendo uma base sólida para a concepção e aprimoramento do protótipo.

A partir desse levantamento, foi possível compreender as lacunas existentes nas soluções disponíveis e propor funcionalidades que atendam de forma mais eficaz às necessidades dos estudantes e da coordenação de estágios do IFB, Campus Brasília. Na análise, foram incluídas as seguintes plataformas e trabalhos correlatos:

- **UFRGS** (Universidade Federal do Rio Grande do Sul): Portal de Vagas da UFRGS, que centraliza a oferta de bolsas e estágios para seus alunos. (Flesch, 2018).
- **UTFPR** (Universidade Tecnológica Federal do Paraná): Plataforma da UTFPR, focada no gerenciamento de estágios acadêmicos. (Cól, 2016).
- **Univ. Porto** (Universidade do Porto): Plataforma de Estágios da Universidade do Porto, em Portugal, que facilita a comunicação entre empresas e instituições acadêmicas. (Rolino, 2013).
- **UFPI** (Universidade Federal do Piauí): Sistema de Gerenciamento de Estágios Universitários (SisGEU) da UFPI, voltado para o curso de Sistemas de Informação. (Oliveira, 2015).

- **Sistema de Inscrição de Maricá:** Sistema utilizado pela Prefeitura de Maricá para monitorar e gerenciar estágios remunerados. (Tavares; Barroso, 2022).

As fontes para as plataformas de mercado (*LinkedIn*¹ e *Indeed*²) foram obtidas diretamente de seus respectivos sites.

Para entender melhor o contexto do sistema "**IFB + Estágios**", segue uma comparação com outras plataformas acadêmicas e de emprego, destacando as principais funcionalidades e especificidades, a tabela 1 sintetiza essas informações, permitindo uma visualização rápida e clara das principais características de cada sistema.

¹<https://www.linkedin.com>

²<https://www.indeed.com>

Tabela 1 Comparação por Funcionalidade de Plataformas de Estágio e Emprego

Funcionalidade	"IFB + Estágios"	UFRGS	UTFPR	Univ. Porto	UFPI	Maricá	LinkedIn	Indeed
Login Integrado	X	X	X	X	X	X		
Envio de Currículos		X		X	X	X	X	X
Pesquisar Vagas	X	X	X	X	X		X	X
Salvar Vagas	X	X					X	X
Recomendações Personalizadas	X	X					X	X
Estágios Não Remunerados	X	X		X				
Estágios Remunerados	X	X		X		X		
Interação Aluno/Professor		X	X	X	X	X		
Gerenciamento de Vagas	X	X	X	X	X	X		
Feedback de Usuários	X	X		X	X	X		
Plataforma de Chat				X			X	
Responsividade (mobile)	X	X				X	X	X
Notificações Automáticas	X	X				X	X	X
Integração com Redes Sociais				X			X	
Análise de Competências		X		X		X	X	
Satisfação dos Usuários	X	X		X	X		X	X

Fonte: Autoria própria.

Conforme apresentado na tabela 1, é possível observar as diferenças entre esses sistemas e suas propostas para cada tipo de público-alvo e regra de negócio específica. Enquanto plataformas como *LinkedIn* e *Indeed* têm um foco mais generalista e abrangem diferentes perfis profissionais em várias áreas, o "IFB + Estágios" foi projetado para atender exclusivamente às necessidades dos alunos do campus IFB Brasília, oferecendo uma experiência mais personalizada e direcionada ao contexto acadêmico

Apesar de as plataformas analisadas apresentarem funcionalidades como "envio de currículos", "análise de competências", "interação aluno/professor" e "integração com redes sociais", o sistema "IFB + Estágios" foi concebido com foco nas necessidades mais críticas dos usuários do IFB, campus Brasília. As escolhas de funcionalidades priorizaram o objetivo de centralizar informações sobre vagas de estágio em uma plataforma única e acessível, atendendo a um dos principais desafios enfrentados pelos alunos e pela coordenação de estágios.

Essa abordagem permite que o sistema cumpra sua proposta inicial de resolver o problema da dispersão de informações, considerado um gargalo significativo no modelo atual de gestão de vagas. Funcionalidades adicionais, como envio de currículos e integração com redes sociais, apesar de valiosas, exigiriam um desenvolvimento mais robusto e uma maior dedicação de tempo e recursos, o que ultrapassaria as limitações definidas para este Trabalho de Conclusão de Curso. Contudo, essas funcionalidades estão previstas como potenciais melhorias futuras, ampliando o impacto do sistema para atender a novas demandas e enriquecer a experiência do usuário.

Além disso, o foco atual em centralizar as informações de vagas permite uma validação mais prática e direcionada da plataforma, assegurando que as funcionalidades implementadas sejam otimizadas e bem avaliadas pelos usuários antes de se expandir para áreas mais amplas de atuação. Essa estratégia garante um desenvolvimento gradual e sustentável, alinhado às necessidades reais da comunidade do IFB.

2.3 Tecnologias Utilizadas

A seleção das tecnologias apropriadas desempenha um papel fundamental no desenvolvimento do presente Trabalho de Conclusão de Curso. Nesta seção, são apresentadas e discutidas as ferramentas, linguagens de programação, *frameworks* e demais recursos adotados para a implementação da solução *web* "IFB + Estágios".

Ao longo desta seção, são detalhadas as tecnologias utilizadas, justificando suas escolhas, descrevendo suas características principais e demonstrando como cada uma contribui para o alcance dos objetivos propostos para o projeto.

2.3.1 HTML, CSS e JavaScript

A tríade composta por *HTML*, (*Cascading Style Sheets (CSS)*) e *JavaScript* é amplamente reconhecida como a base do desenvolvimento *web* moderno, desempenhando papéis complementares e indispensáveis na criação de sistemas interativos e funcionais (Flanagan, 2013).

O *HTML (Hypertext Markup Language)* é a linguagem padrão para estruturar o conteúdo na *web*, definindo os elementos de uma página, como cabeçalhos, parágrafos, *links*, imagens e formulários, por meio de *tags* organizadas hierarquicamente (MDN Web Docs, 2023). Sua simplicidade e flexibilidade tornam o *HTML* a escolha ideal para estruturar o conteúdo do sistema "IFB + Estágios", garantindo uma base clara e organizada para a integração com tecnologias adicionais.

O *CSS (Cascading Style Sheets)*, por sua vez, é responsável por estilizar e controlar a aparência visual de documentos *HTML*, permitindo a personalização de cores, fontes, *layout* e animações (MDN Web Docs, 2022a). Essa linguagem proporciona uma interface visualmente atraente e responsiva, assegurando que o design do sistema seja acessível em diferentes dispositivos e intuitivo para os usuários.

Por fim, o *JavaScript (JS)* adiciona dinamismo e interatividade às páginas *web*, manipulando conteúdos, eventos e o comportamento do navegador (MDN Web Docs, 2022b).

2.3.2 React

React é uma biblioteca de *JavaScript* mantida pela empresa *Meta* e utilizada na criação de produtos digitais como a rede social *Facebook*. Lançada em 2013, a biblioteca tornou-se amplamente adotada devido à sua abordagem baseada em componentes reutilizáveis, que permitem a construção de interfaces complexas de forma eficiente e modular (Meta, 2024).

A escolha do *React* como tecnologia principal para o desenvolvimento do sistema "IFB + Estágios" baseou-se em sua robustez, flexibilidade e ampla comunidade de suporte. Entre as principais vantagens do *React*, destacam-se:

- **Componentização:** A capacidade de dividir a interface em componentes reutilizáveis

facilita a manutenção e a escalabilidade do sistema. Cada componente encapsula lógica, estilo e estrutura, o que promove a coesão do código.

- **Virtual DOM:** O uso do *Virtual DOM* para atualização eficiente da interface melhora significativamente o desempenho do sistema, tornando-o responsivo mesmo em cenários de grande complexidade.
- **Biblioteca Extensível:** O *React* oferece suporte para integração com uma ampla gama de bibliotecas e ferramentas adicionais, como *React Router* para navegação e *Redux* para gerenciamento de estado.
- **Suporte à Reatividade:** O *React* permite que a interface se atualize dinamicamente em resposta às mudanças no estado ou nos dados do aplicativo, o que é crucial para sistemas como "IFB + Estágios", onde a exibição de vagas e informações deve ser dinâmica e interativa.

Ao longo do desenvolvimento do sistema, o *React* foi utilizado para estruturar e gerenciar toda a interface de usuário, desde a exibição de vagas de estágio até a implementação de funcionalidades dinâmicas, como filtros personalizados e notificações em tempo real.

Estudos anteriores reforçam a relevância e o impacto do *React* no desenvolvimento web moderno. Flanagan (2013) destaca a importância de bibliotecas como o *React* na evolução do ecossistema JavaScript, enquanto trabalhos mais recentes, como o de Hoque (2020), evidenciam a ampla adoção do *React* em soluções de larga escala, graças à sua capacidade de lidar com interfaces complexas e de alto desempenho.

Além disso, a escolha do *React* está alinhada às práticas recomendadas para desenvolvimento ágil, permitindo iterações rápidas e flexíveis durante o ciclo de desenvolvimento do "IFB + Estágios". Essa abordagem foi crucial para o cumprimento dos objetivos do projeto, ao proporcionar uma interface intuitiva e eficiente tanto para os alunos quanto para a coordenação de estágios.

2.3.3 Figma

O *Figma* é uma ferramenta de design colaborativo baseada na web, amplamente utilizada para o desenvolvimento de protótipos e interfaces de usuário (UI). Uma de suas principais vantagens é a possibilidade de colaboração em tempo real, permitindo que equipes multidisciplinares trabalhem juntas no design de interfaces, independentemente de sua localização. Além disso, o *Figma* oferece integração com outras ferramentas de design e desenvolvimento, facilitando a transição do protótipo para o código final.

No desenvolvimento do software "IFB + Estágios", o *Figma* desempenhou um papel central na concepção das interfaces e protótipos do sistema. Conforme discutido por Staiano (2022), o uso de ferramentas como o *Figma* não só otimiza o processo de design e prototipagem,

mas também capacita equipes a iterar rapidamente com base no *feedback* dos usuários e a alinhar o design às expectativas de diferentes partes interessadas. A plataforma permitiu validar as ideias iniciais do design com os usuários e coordenadores de estágio, garantindo que a interface fosse intuitiva e eficiente para o público-alvo.

Além disso, conforme Staiano (2022) o *Figma* pode ser usado para aplicar princípios de design centrados no usuário, conectando a prototipagem a fundamentos essenciais de usabilidade. A escolha pelo *Figma* neste projeto se fundamenta também por sua adoção ampla no mercado tecnológico e por sua capacidade de promover consistência visual e acessibilidade. Conforme ressalta o autor, a prototipagem interativa com ferramentas como o *Figma* não apenas economiza tempo de desenvolvimento, mas também reduz o risco de retrabalho ao alinhar as expectativas de design desde os estágios iniciais.

2.3.4 *SQL Server*

O *Microsoft SQL Server* é um Sistema de Gerenciamento de Banco de Dados Relacional (SGBDR) desenvolvido pela *Microsoft*, amplamente utilizado em ambientes corporativos devido à sua robustez, escalabilidade e alto desempenho. Este sistema oferece um conjunto abrangente de funcionalidades que incluem armazenamento, recuperação e manipulação eficiente de grandes volumes de dados, além de ferramentas avançadas de análise e segurança (Microsoft, 2023).

A escolha do *Microsoft SQL Server* para o sistema "IFB + Estágios" se fundamenta em sua confiabilidade e capacidade de atender às demandas de um sistema acadêmico em constante evolução. Sua arquitetura robusta permite o armazenamento seguro de informações essenciais, como dados de estágios, alunos e empresas, garantindo a integridade e acessibilidade dos dados. Além disso, o *SQL Server* é altamente integrado a outras ferramentas e tecnologias, o que facilita o desenvolvimento e a expansão do sistema no futuro, caso novas funcionalidades sejam incorporadas.

Outro diferencial é o suporte nativo a consultas complexas e recursos avançados de segurança, como criptografia de dados e controle de acesso baseado em funções, o que assegura a proteção das informações confidenciais dos usuários. Essa combinação de características faz do *SQL Server* uma escolha estratégica para garantir o sucesso do sistema, alinhando-se aos objetivos de oferecer uma solução confiável e eficiente para o gerenciamento de estágios.

2.3.5 *Mulesoft*

O *Mulesoft* é uma plataforma de integração amplamente utilizada para conectar sistemas, dados e dispositivos, permitindo que as organizações criem redes de aplicação mais ágeis e eficientes. Desenvolvido com base em uma arquitetura orientada a APIs (*Application Programming Interfaces*), o *Mulesoft* possibilita a integração de diferentes sistemas e serviços em ambientes complexos, tanto locais quanto na nuvem (Mulesoft, 2024).

No contexto do projeto "IFB + Estágios", o *Mulesoft* foi considerado uma ferramenta

estratégica para garantir a integração entre diferentes sistemas e bases de dados que possam ser incorporadas ao sistema no futuro. Embora atualmente o foco do projeto seja a centralização das informações de vagas e funcionalidades diretamente relacionadas aos estudantes, a escolha do *Mulesoft* visa facilitar a escalabilidade e a interoperabilidade do sistema em longo prazo.

Entre as principais vantagens do uso do *Mulesoft*, destacam-se:

- **Modularidade e Reutilização:** A arquitetura orientada a APIs permite que componentes sejam desenvolvidos e reutilizados, otimizando o tempo e os recursos necessários para futuras expansões do sistema.
- **Flexibilidade:** O *Mulesoft* é capaz de integrar dados e serviços de diferentes fontes, independentemente de suas plataformas ou tecnologias subjacentes.
- **Segurança:** A plataforma oferece ferramentas avançadas para gerenciar acessos, autenticações e monitorar as interações entre sistemas, garantindo a proteção das informações e conformidade com as políticas de privacidade e segurança.
- **Monitoramento em Tempo Real:** Com painéis detalhados, o *Mulesoft* facilita o rastreamento das integrações e o diagnóstico de falhas, otimizando a gestão operacional do sistema.

A implementação do *Mulesoft* no "IFB + Estágios" reforça o compromisso do projeto com práticas modernas de engenharia de software e integração de sistemas. A escolha dessa ferramenta reflete a visão de longo prazo de criar uma solução robusta, que não apenas atende às necessidades atuais, mas também se adapta às futuras demandas do IFB.

2.3.6 Azure DevOps

O *Azure DevOps* é uma plataforma abrangente de desenvolvimento e colaboração criada pela *Microsoft*, projetada para suportar equipes no planejamento, desenvolvimento, entrega e manutenção de software. A ferramenta oferece uma série de serviços integrados, incluindo gerenciamento de *backlog*, controle de versão, integração e entrega contínuas (*Continuous Integration/Continuous Deployment (CI/CD)*) e monitoramento do progresso das tarefas (*Microsoft*, 2024).

No desenvolvimento do sistema "IFB + Estágios", o *Azure DevOps* desempenhou um papel essencial na aplicação da metodologia *Scrum*. A plataforma foi utilizada para estruturar *sprints*, que dividiram o processo de desenvolvimento em ciclos curtos e iterativos, focados na entrega de funcionalidades prioritárias. As atividades foram organizadas em itens de *backlog*, permitindo a definição clara de responsabilidades, prazos e critérios de conclusão.

Os painéis de controle visuais e relatórios automatizados do *Azure DevOps* facilitaram o monitoramento contínuo do progresso das tarefas, promovendo maior transparência e alinhamento entre os membros da equipe. Além disso, a integração com a metodologia *Scrum*

permitiu revisões regulares e ajustes no planejamento durante os *sprints*, garantindo entregas mais alinhadas aos objetivos do projeto.

A escolha do *Azure DevOps* reflete a busca por uma solução que não apenas organiza o fluxo de trabalho, mas também promove uma comunicação eficiente e adaptabilidade no desenvolvimento ágil.

2.3.7 *Git e GitHub*

O *Git* é um sistema de controle de versão distribuído amplamente utilizado em projetos de desenvolvimento de software. Ele permite rastrear alterações no código, gerenciar diferentes versões e colaborar de forma eficiente com outros desenvolvedores. Desenvolvido por Linus Torvalds em 2005, o *Git* é essencial para projetos de software de todos os tamanhos, fornecendo ferramentas para acompanhar a evolução do código e resolver conflitos que podem surgir no trabalho colaborativo Torvalds (2005).

O *GitHub*, por sua vez, é uma plataforma de hospedagem baseada na nuvem que utiliza o *Git* como tecnologia central para controle de versão. Além de servir como um repositório para armazenar e compartilhar código, o *GitHub* oferece funcionalidades adicionais, como ferramentas de colaboração, integração contínua (*CI/CD*), rastreamento de problemas (*issues*) e documentação. Sua interface amigável e as capacidades avançadas tornaram o *GitHub* uma das plataformas mais populares entre desenvolvedores e equipes de software (GitHub Docs, 2023).

O uso de ferramentas como *Git* e *GitHub* é crucial em ambientes acadêmicos e científicos, pois promove o desenvolvimento de projetos de maneira colaborativa, reproduzível e transparente (Blischak; Davenport; Wilson, (2016). No contexto deste TCC, o *Git* foi utilizado para gerenciar as diferentes versões do código-fonte, evitando problemas de versões conflitantes e garantindo a rastreabilidade de todas as modificações realizadas durante o desenvolvimento do sistema. Isso assegurou um histórico claro e organizado, possibilitando revisões e aprimoramentos contínuos.

As principais vantagens no uso dessas ferramentas incluem:

- **Versionamento eficiente:** O *Git* permitiu acompanhar o histórico completo das alterações feitas no código, ajudando na identificação e correção de erros.
- **Colaboração simplificada:** O *GitHub* possibilitou que as partes envolvidas no desenvolvimento compartilhassem código e informações de maneira estruturada e segura.
- **Armazenamento em nuvem:** Todo o projeto foi mantido em repositórios remotos, eliminando riscos de perda de dados e aumentando a acessibilidade.

2.3.8 *Overleaf*

O *Overleaf* é uma plataforma baseada na web que utiliza o *LaTeX*, uma linguagem de marcação amplamente empregada para a produção de documentos acadêmicos, técnicos e

científicos. O *LaTeX* destaca-se por oferecer um controle rigoroso sobre a formatação, permitindo a criação de documentos altamente estruturados e esteticamente consistentes, especialmente em contextos onde há exigências rigorosas de normas e padrões acadêmicos.

A principal motivação para a adoção do *Overleaf* neste projeto foi sua funcionalidade de colaboração em tempo real, que possibilitou uma comunicação mais eficiente entre os membros da equipe durante a escrita do TCC. Além disso, a segurança de salvar automaticamente os documentos na nuvem e a acessibilidade a partir de qualquer dispositivo com conexão à internet garantiram que o trabalho pudesse ser conduzido de forma consistente e segura.

Outro aspecto relevante é a facilidade oferecida pelo *Overleaf* em relação ao uso do *LaTeX*. A interface amigável da plataforma, aliada à ampla base de modelos prontos e recursos de suporte, permite que mesmo usuários com pouca experiência em *LaTeX* consigam produzir documentos de alta qualidade.

O uso do *Overleaf* reflete uma tendência crescente no meio acadêmico para ferramentas que integram tecnologia de nuvem e colaboração em tempo real. Estudos, como o de Eaton (2019) destacam como essas plataformas têm transformado o processo de escrita acadêmica, promovendo eficiência, organização e qualidade no resultado final.

3

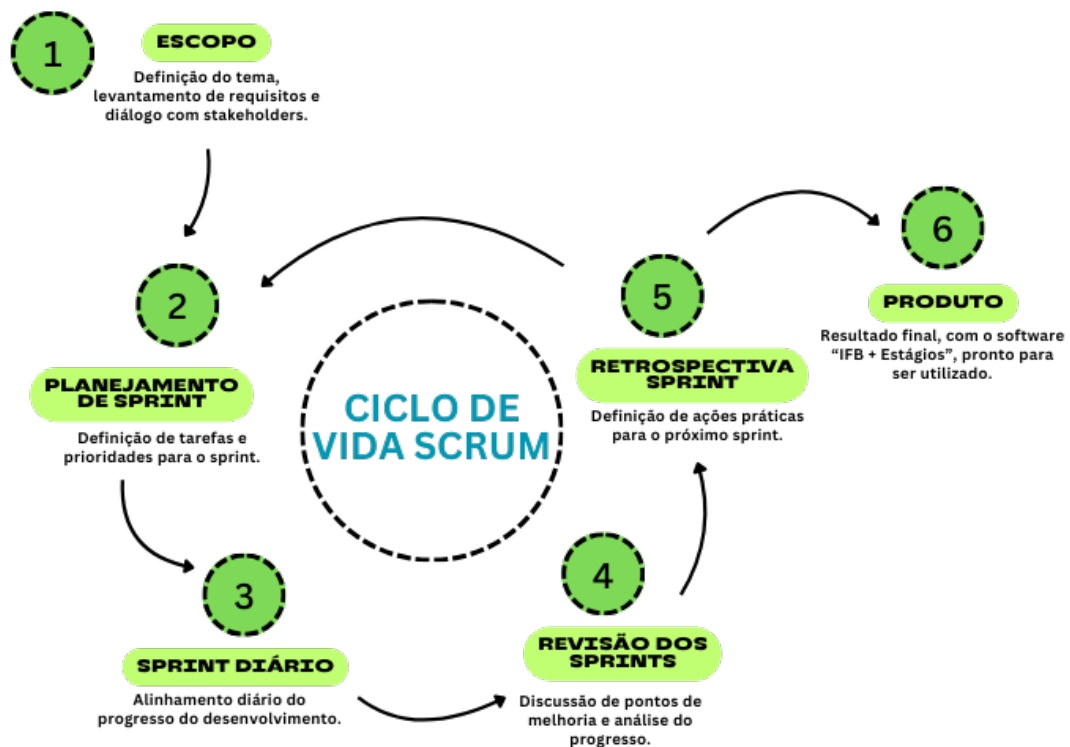
METODOLOGIA

A metodologia adotada para o desenvolvimento do software "IFB + Estágios" seguiu uma abordagem estruturada e multidisciplinar, combinando ferramentas e práticas consagradas para garantir o alinhamento às necessidades dos principais *stakeholders*, como os coordenadores de estágio e os estudantes. Essa combinação de metodologias foi essencial para abordar o projeto sob diferentes perspectivas, desde a análise estratégica até a execução prática e o acompanhamento contínuo.

Inicialmente, foi utilizada a metodologia de *Design Thinking* para compreender profundamente as necessidades dos usuários, promovendo uma abordagem empática e centrada nas pessoas. Por meio dessa metodologia, foi possível mapear problemas, gerar ideias e protótipos iniciais que guiaram o desenvolvimento do sistema.

Durante a execução do projeto, a metodologia ágil *Scrum* foi adotada para o gerenciamento das atividades. Essa abordagem iterativa e incremental permitiu o desenvolvimento do sistema em *sprints*, facilitando o acompanhamento do progresso e possibilitando ajustes contínuos com base no *feedback* dos *stakeholders*. Na Figura 1, é exemplificado a maneira como foi implementado a metodologia *Scrum* no projeto.

Figura 1 Ciclo de Vida Scrum



Fonte: Autoria própria

Paralelamente, foi realizada uma análise *Strengths, Weaknesses, Opportunities, and Threats (SWOT)* (*Strengths, Weaknesses, Opportunities, and Threats*), que auxiliou na avaliação estratégica do sistema "IFB + Estágios". Essa análise permitiu identificar pontos fortes e fracos do projeto, bem como explorar oportunidades e mitigar possíveis ameaças ao longo de seu desenvolvimento.

Outra abordagem importante foi a aplicação de técnicas de **Modelagem de Processos de Negócio (BPM)**, que incluiu a criação dos modelos **As-Is** e **To-Be**. O modelo *As-Is* permitiu mapear o estado atual dos processos de gerenciamento de estágios, identificando gargalos, ineficiências e áreas que poderiam ser otimizadas. Com base nesse levantamento, foi desenvolvido o modelo *To-Be*, que descreveu a visão futura dos processos, incorporando melhorias como a centralização de informações, a personalização de filtros de busca e notificações automáticas.

Essa combinação de metodologias garantiu um processo dinâmico e adaptável, no qual a pesquisa, o planejamento e a execução foram constantemente ajustados para atender às expectativas dos usuários e às demandas institucionais. As subseções a seguir detalham cada uma dessas abordagens, descrevendo como foram aplicadas para alcançar os objetivos do projeto e entregar um sistema eficiente e inovador.

3.1 Pesquisa Bibliográfica

A pesquisa bibliográfica realizada neste trabalho desempenhou um papel essencial na fundamentação teórica do sistema "IFB + Estágios". Para conduzir essa pesquisa, foram utilizadas bases de dados acadêmicas amplamente reconhecidas, como *Google Scholar* e *Oasisbr*, que proporcionaram acesso a uma vasta quantidade de publicações científicas relacionadas a temas como gestão de estágios, sistemas de gerenciamento de vagas e sua aplicação no contexto educacional. O foco principal recaiu sobre estudos e publicações dos últimos dez anos (2014-2024), garantindo a atualidade das informações e o alinhamento às mais recentes inovações tecnológicas. Obras anteriores a esse período também foram consideradas devido à sua relevância teórica consolidada, especialmente em relação às bases legais e diretrizes pedagógicas do estágio.

Os termos de busca utilizados incluíram combinações estratégicas de palavras-chave, como: "sistema de gerenciamento de estágios", importância dos estágios, "*internship management system*", "plataformas de vagas de estágio" e "sistemas de vagas de estágio". Após a coleta inicial, os artigos e publicações foram avaliados e selecionados com base na sua relevância prática e teórica para o projeto, com foco no contexto do IFB, campus Brasília. As publicações escolhidas exploraram temas fundamentais, como:

- A importância do estágio na formação acadêmica e sua conexão com o mercado de trabalho;
- Desenvolvimento de plataformas digitais para a gestão de estágios em instituições de ensino superior;
- Boas práticas e desafios na implementação de sistemas educacionais informatizados;
- Aspectos legais e regulatórios relacionados ao estágio, com foco na Lei nº 11.788/2008, que regulamenta a prática do estágio no Brasil.

A análise das publicações permitiu identificar *insights* valiosos sobre as melhores práticas e ferramentas aplicáveis ao desenvolvimento de sistemas de gerenciamento de estágios. Essa investigação contribuiu diretamente para a definição dos requisitos funcionais e não funcionais do sistema "IFB + Estágios" e orientou a escolha das tecnologias mais adequadas ao projeto.

Os resultados dessa pesquisa bibliográfica foram aplicados diretamente ao desenvolvimento do sistema, servindo como base para a criação de funcionalidades, como personalização de filtros de vagas, notificações automáticas e a centralização de informações sobre estágios em uma única plataforma, focada no IFB, campus Brasília. Estudos sobre usabilidade e design de interfaces foram particularmente úteis na construção de um sistema intuitivo e acessível para alunos e equipe administrativa.

Ao longo do processo de desenvolvimento, a pesquisa bibliográfica foi constantemente revisitada para garantir que as decisões de design e implementação permanecessem alinhadas às

recomendações teóricas e às necessidades identificadas. Essa abordagem assegurou que o sistema final fosse não apenas funcional, mas também estivesse em conformidade com as melhores práticas do campo de gestão de estágios. Dessa forma, a pesquisa bibliográfica fundamentou de forma adequada às demandas práticas da coordenação de estágios e dos estudantes do IFB, campus Brasília.

3.2 Metodologia Baseada no Design Thinking

O desenvolvimento do sistema IFB + Estágios foi orientado pela abordagem do *Design Thinking*, conforme proposta por Brown (2020) em *Design Thinking: Integrating Innovation, Customer Experience, and Brand Value*. Essa abordagem coloca o usuário no centro do processo de criação, focando na resolução de problemas reais e na geração de soluções inovadoras por meio de um ciclo iterativo.

De acordo com Brown (2020), o *Design Thinking* é estruturado em cinco etapas principais:

- **Empatia:** Nesta etapa, realizou-se um levantamento das principais dificuldades enfrentadas pelos alunos e pela coordenação de estágios. Foram aplicadas entrevistas para entender as *dores dos usuários*.
- **Definição:** Os *insights* obtidos durante a fase de empatia serviram para **definir os problemas centrais** que o sistema deveria resolver.
- **Ideação:** Nesta etapa, foram geradas ideias e soluções por meio de sessões de *brainstorming*. Por exemplo, a ideia de permitir que os alunos **salvem vagas de interesse** surgiu como uma solução prática para facilitar a gestão das oportunidades.
- **Prototipagem:** Foram criados protótipos iniciais em alta fidelidade validados com um grupo piloto de alunos. O *feedback* obtido ajudou a refinar a interface e a garantir que as funcionalidades fossem **intuitivas e acessíveis**.
- **Teste:** Após o desenvolvimento do sistema, aplicaram-se testes de usabilidade (como *System Usability Scale* (SUS)) e análises heurísticas (Nielsen) para medir a **eficácia e a satisfação dos usuários** com a solução proposta.

Como resultado, o *Design Thinking* forneceu uma base sólida para a construção do sistema, alinhando-se à proposta de resolver problemas específicos de forma centrada no usuário. Brown (2020) ressalta que o *Design Thinking* é essencial para integrar inovação, experiência do usuário e valor agregado em projetos de software.

3.3 Levantamento de Requisitos e diálogo com *Stakeholders*

Na fase de Análise de Requisitos do sistema "IFB + Estágios", foi adotado uma abordagem interativa e colaborativa. Inicialmente, foram realizadas entrevistas com os principais envolvidos do projeto, como a coordenação de Estágios e o professor orientador, incluindo estudantes, para entender suas necessidades e expectativas em relação ao sistema. Além disso, observações diretas das práticas atuais de gestão de estágios no IFB forneceram informações valiosas. A análise dessas informações possibilitou a identificação de requisitos funcionais e não funcionais para o desenvolvimento do sistema, assegurando que o "IFB + Estágios" atendesse efetivamente às necessidades de seus usuários.

Este processo de coleta de requisitos é muito importante, considerando que a eficiência na gestão de informações de estágio é um desafio comum enfrentado pelas instituições educacionais. De acordo com estudo de Alvarenga (2013), realizado para verificar a disponibilidade de informações sobre vagas de estágio nos portais das Universidades Federais revelou que a utilização de recursos digitais para a gestão de estágios ainda é moderada pelas instituições. O estudo sugere a ampliação da utilização dessas ferramentas como uma oportunidade de consolidar a política institucional de estágio, indicando um caminho para a adoção de boas práticas na divulgação e gestão de estágios. Assim, o "IFB + Estágios" surge como uma resposta a essa necessidade, oferecendo uma solução para melhorar o acesso e a gestão das informações de estágio no IFB, campus Brasília.

3.4 Questionários para Avaliação do Sistema

Para garantir a avaliação da usabilidade e eficácia do sistema "IFB + Estágios", foram elaborados dois questionários fundamentados em metodologias reconhecidas na literatura científica: o *System Usability Scale (SUS)* desenvolvido por Brooke (1996), e as **10 heurísticas de usabilidade de Nielsen** (1994). Essas abordagens fornecem *insights* valiosos sobre a experiência do usuário, além de identificar possíveis pontos de melhoria no sistema proposto.

As respostas coletadas foram analisadas com base em métricas quantitativas e qualitativas, possibilitando a criação de gráficos e relatórios para facilitar a interpretação dos dados. A coleta foi realizada junto aos usuários finais, como alunos e coordenadores de estágio, por meio da plataforma *Google Forms*.

3.4.1 *Questionário 1: Avaliação da Usabilidade com o SUS*

O SUS é amplamente utilizado para avaliar a usabilidade de sistemas de software. Ele consiste em 10 perguntas padronizadas, que abrangem diferentes dimensões da experiência do usuário. As perguntas utilizadas incluem:

1. Eu gostaria de utilizar este sistema com frequência?

2. Considero o sistema desnecessariamente complexo?
3. Acredito que o sistema é fácil de usar?
4. Acredito que precisaria da ajuda de uma pessoa com conhecimentos técnicos para conseguir usar este sistema?
5. Percebo que as diferentes funcionalidades do sistema estão bem integradas?
6. Acredito que o sistema apresenta inconsistências?
7. Imagino que a maioria das pessoas aprenderia a usar este sistema rapidamente?
8. Acredito que o sistema é confuso ou difícil de navegar?
9. Eu me senti muito confiante usando o sistema?
10. Precisei aprender várias coisas novas antes de conseguir usar o sistema?

Cada uma dessas perguntas é respondida em uma escala Likert de 1 a 5, onde 1 significa *Discordo totalmente* e 5 significa *Concordo totalmente*. A pontuação final do SUS é calculada para obter um valor que pode ser comparado com *benchmarks* de usabilidade amplamente aceitos.

A aplicação deste questionário permite avaliar o sistema de maneira simples e objetiva, gerando uma métrica única que reflete a usabilidade percebida pelos usuários.

3.4.2 Questionário 2: Avaliação Baseada nas Heurísticas de Nielsen

As 10 heurísticas de Nielsen são um conjunto de princípios reconhecidos para a avaliação de interfaces de usuário. Para este trabalho, as perguntas foram elaboradas para refletir cada uma das heurísticas, sendo elas:

1. **Visibilidade do status do sistema:** O sistema informa claramente o status das ações em andamento?
2. **Correspondência entre o sistema e o mundo real:** A linguagem utilizada no sistema é familiar ao usuário e reflete seus objetivos?
3. **Controle e liberdade do usuário:** O sistema permite que o usuário desfça ou refaça ações de maneira fácil?
4. **Consistência e padrões:** O sistema segue padrões consistentes em toda a interface?
5. **Prevenção de erros:** O sistema ajuda a prevenir erros antes que eles ocorram?
6. **Reconhecimento em vez de memorização:** As opções e ações disponíveis são visíveis e fáceis de acessar, reduzindo a carga de memória do usuário?

7. **Flexibilidade e eficiência de uso:** O sistema atende tanto usuários iniciantes quanto avançados?
8. **Estética e design minimalista:** A interface apresenta apenas informações relevantes e evita elementos desnecessários?
9. **Ajuda aos usuários a reconhecer, diagnosticar e corrigir erros:** O sistema fornece mensagens claras para resolver problemas?
10. **Ajuda e documentação:** O sistema oferece suporte ou documentação acessível quando necessário?

A escolha dessas metodologias de questionários são baseadas na sua ampla aceitação na área de usabilidade. O SUS oferece uma métrica quantitativa padronizada, permitindo comparações com benchmarks de mercado (BROOKE, 1996). Já as heurísticas de Nielsen permitem uma análise detalhada e qualitativa, identificando problemas específicos na interface e funcionalidades do sistema.

Essa abordagem combinada possibilita identificar pontos de fricção na experiência do usuário, garantindo que o sistema "IFB + Estágios" atenda às necessidades dos usuários finais de forma intuitiva, eficiente e satisfatória.

Os questionários foram estruturados com perguntas baseadas na escala Likert, além de algumas questões abertas para comentários adicionais. As respostas coletadas serviram de base para identificar pontos fortes e oportunidades de melhoria no sistema.

A análise detalhada dos dados obtidos será apresentada na seção de Resultados, onde os dados quantitativos serão exibidos em gráficos e tabelas, enquanto os dados qualitativos serão interpretados para fornecer *insights* sobre a experiência dos usuários. Essa abordagem garante que as conclusões sejam fundamentadas em evidências sólidas, alinhando os resultados às metas estabelecidas para o projeto

3.5 Metodologia Análise SWOT

A análise *SWOT*, (Strengths, Weaknesses, Opportunities, and Threats) desenvolvida por Albert Humphrey nos anos 1960, é amplamente reconhecida como uma ferramenta estratégica poderosa, capaz de integrar ambientes internos e externos para destacar pontos fortes e fracos, bem como oportunidades e ameaças no desenvolvimento de projetos. Em seu artigo, Leite e Gasparotto (2018), destacam como essa matriz ajuda as organizações a obter um controle mais amplo de seus processos produtivos, proporcionando *insights* valiosos para o planejamento estratégico e a tomada de decisões.

A análise *SWOT* é uma etapa no processo de planejamento, que ajuda a identificar vantagens competitivas, bem como desafios que podem ser enfrentados durante o ciclo de vida do projeto.

A fim de aprofundar a compreensão sobre os fatores que podem influenciar o sucesso do software "IFB + Estágios", foi realizada uma análise *SWOT*, na qual analisa as forças, fraquezas, oportunidades e ameaças associadas ao desenvolvimento e à implementação do sistema.

Com base nos requisitos levantados junto aos *stakeholders* e nas características do projeto, a análise *SWOT* foi estruturada para mapear os principais fatores que afetam o desenvolvimento do software.

A análise *SWOT* do sistema "IFB + Estágios" destaca seu potencial para otimizar a gestão de estágios no IFB, centralizando informações e facilitando o acesso dos alunos às oportunidades. Entre suas principais forças estão a personalização das vagas e a adaptação ao regulamento institucional, enquanto suas fraquezas incluem a dependência da qualidade dos dados inseridos e desafios na manutenção do sistema. As oportunidades indicam um possível crescimento para outros institutos federais e a adoção de tecnologias como inteligência artificial para melhorar a filtragem de vagas. No entanto, ameaças como a resistência dos administradores e a concorrência com outras plataformas podem dificultar sua adoção e evolução.

Dessa forma, o sucesso do sistema depende de sua aceitação por alunos e coordenação de estágios, além de um compromisso contínuo com atualizações e melhorias. Se bem implementado e mantido, pode se tornar um modelo eficiente para a gestão de estágios em instituições públicas, fortalecendo a relação entre ensino e mercado de trabalho. A seguir, apresenta-se a análise completa, na Tabela 2.

Tabela 2 Análise SWOT do sistema "IFB + Estágios"

Forças	Fraquezas
<ul style="list-style-type: none"> ■ O software unifica as informações de estágios, facilitando o acesso para os estudantes. ■ Filtros e notificações personalizados de acordo com o perfil dos usuários melhoram a eficiência na busca de estágios adequados. ■ O design centrado no usuário e os protótipos simples tornam o sistema intuitivo para estudantes e administradores. ■ O sistema é especificamente projetado para o IFB, o que garante uma adaptação completa às suas necessidades e regulamentações, como o regulamento de estágios. 	<ul style="list-style-type: none"> ■ O uso de tecnologias como <i>HyperText Markup Language (HTML)</i>, <i>CSS</i>, <i>JavaScript</i> e <i>SQL Server</i> pode criar uma barreira para futuros desenvolvedores que não tenham familiaridade com essas ferramentas. ■ Poucas pessoas atuando no projeto pode exigir um maior esforço e mais demora para manutenção e aprimoramentos. ■ O sucesso do sistema depende diretamente da qualidade de vagas preenchidas pelo IFB, com o máximo de informações nos campos de cada vaga.
Oportunidades	Ameaças
<ul style="list-style-type: none"> ■ A plataforma pode ser expandida para atender Institutos Federais de outros campi e estados, aumentando seu impacto. ■ Adoção de novas tecnologias: A implementação de ferramentas como inteligência artificial ou <i>machine learning</i> pode melhorar ainda mais a filtragem de vagas e a personalização de perfis. ■ O sistema pode aumentar a participação dos alunos em estágios, contribuindo para o sucesso no mercado de trabalho. 	<ul style="list-style-type: none"> ■ Surgimento de plataformas semelhantes que ofereçam mais funcionalidades pode ameaçar a adoção do software. ■ Alguns estudantes ou administradores podem resistir à adoção da nova tecnologia, preferindo os métodos tradicionais de comunicação e divulgação de vagas.

Fonte: Autoria própria

3.6 Metodologia Scrum

A metodologia *Scrum* é um framework ágil amplamente utilizado no desenvolvimento de projetos, especialmente em software, devido à sua abordagem iterativa e incremental. Criado por Jeff Sutherland e Ken Schwaber nos anos 1990, o *Scrum* se destaca por permitir flexibilidade e adaptação contínua ao longo do ciclo de vida do projeto, priorizando a entrega de valor ao cliente (Stopa; Rachid, 2019).

O *Scrum* é uma metodologia que divide o trabalho em ciclos curtos e repetitivos, chamados de *sprints*, geralmente com duração de 2 a 4 semanas. Cada *sprint* tem como objetivo entregar incrementos funcionais do produto, permitindo ajustes contínuos com base no *feedback* dos *stakeholders*. O *Scrum* é baseado em valores fundamentais, como foco, coragem, compromisso, respeito e transparência, que promovem a colaboração e a eficiência no desenvolvimento de projetos (Stopa; Rachid, 2019).

Essa abordagem é particularmente útil em contextos de incerteza, como o desenvolvimento do sistema "IFB + Estágios", pois permite adaptar-se rapidamente às mudanças nos requisitos e priorizar funcionalidades de maior impacto para os usuários.

3.6.1 Cerimônias do Scrum

As cerimônias do *Scrum* são práticas essenciais para a gestão eficiente do trabalho. No projeto "IFB + Estágios", as seguintes cerimônias foram realizadas:

- *Sprint Planning*: Planejamento colaborativo das atividades que serão realizadas no *sprint*. As tarefas foram organizadas com base nas prioridades e nas funcionalidades mais críticas.
- *Daily Scrum*: Reuniões diárias para alinhar o progresso das atividades, identificar bloqueios e ajustar prioridades quando necessário.
- *Sprint Review*: Apresentação do incremento desenvolvido ao final de cada *sprint*, permitindo o recebimento de *feedback* dos *stakeholders*.
- *Sprint Retrospective*: Análise do desempenho da equipe durante o *sprint*, destacando pontos fortes e oportunidades de melhoria.

A implementação do *Scrum* no projeto foi facilitada pelo uso do *Azure DevOps*, uma plataforma que fornece ferramentas integradas para gerenciar projetos de desenvolvimento de software. Com o *Azure DevOps*, foi possível organizar os itens de *backlog*, acompanhar o progresso das tarefas por meio de quadros *Kanban*, e gerar relatórios detalhados para avaliar o desempenho do time.

As tarefas foram distribuídas em *user stories*, permitindo que cada membro da equipe tivesse clareza sobre suas responsabilidades e os prazos estabelecidos. Além disso, o *Azure*

DevOps ofereceu suporte para a integração de *pipelines* automatizados, possibilitando a entrega contínua de incrementos funcionais ao longo dos *sprints*. Benefícios do Scrum no Projeto

A aplicação do *Scrum* no desenvolvimento do sistema "IFB + Estágios" proporcionou diversos benefícios, como:

- Melhoria na comunicação e colaboração: As cerimônias promoveram o alinhamento contínuo entre os membros da equipe e os *stakeholders*.
- Entrega contínua de valor: A estrutura de *sprints* permitiu a entrega incremental de funcionalidades, facilitando a validação contínua do sistema.
- Flexibilidade: A possibilidade de priorizar novas funcionalidades ou ajustar requisitos ao longo do projeto garantiu maior aderência às necessidades dos usuários.

A utilização do Scrum no projeto também reflete sua adaptabilidade a diferentes contextos, como discutido por Sutherland (2014), reforçando que a metodologia é uma escolha eficaz para o desenvolvimento de soluções em ambientes educacionais. Ao adotar o Scrum, o projeto conseguiu alinhar as expectativas dos *stakeholders* com os objetivos pedagógicos e funcionais do sistema, garantindo entregas de alta qualidade em prazos estipulados.

3.7 Análise de Processos com BPM

Para o desenvolvimento do sistema "IFB + Estágios", foi utilizada a abordagem de **Modelagem de Processos de Negócio (Business Process Modeling - Business Process Modeling (BPM))**, que permitiu uma análise detalhada dos fluxos de trabalho atuais relacionados à gestão de estágios (**BPM As-Is**) e a proposição de melhorias (**BPM To-Be**). Essa técnica foi essencial para identificar gargalos, mapear oportunidades de otimização e alinhar o desenvolvimento do sistema às necessidades dos usuários.

3.7.1 BPM As-Is

O modelo **As-Is** é uma representação detalhada do estado atual de um processo. Ele mapeia como as atividades e fluxos são realizados no momento, sem incluir melhorias ou mudanças propostas. O objetivo do *As-Is* é identificar os pontos fortes e fracos do processo existente, oferecendo uma base sólida para a análise e o redesenho.

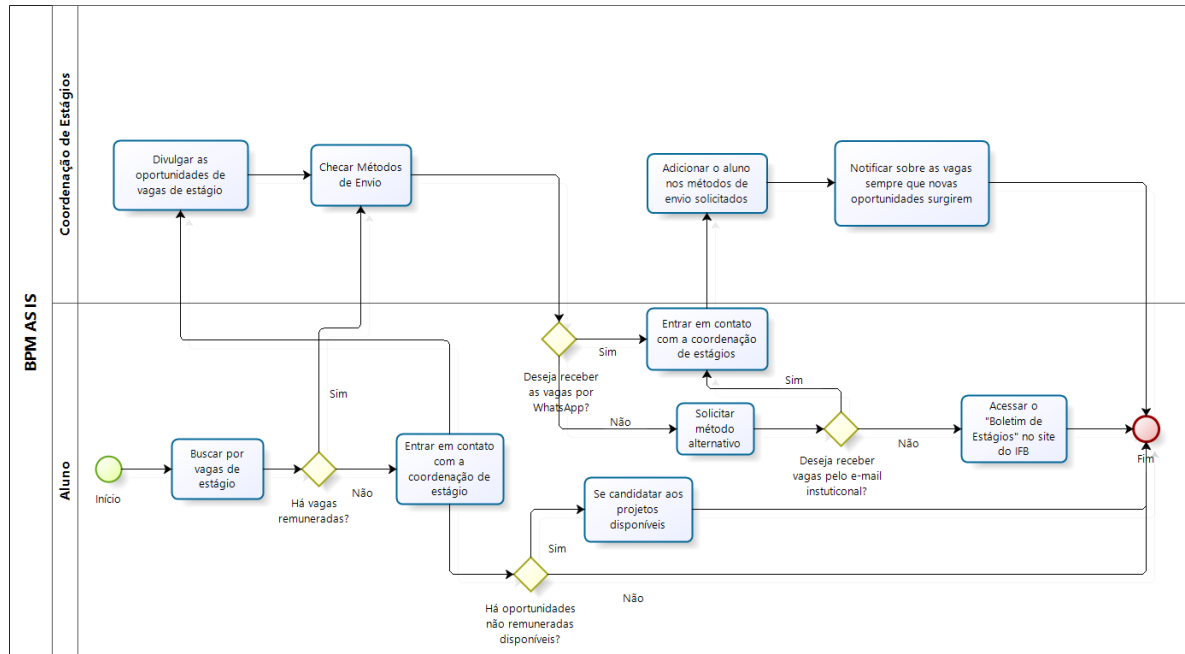
No contexto deste projeto, o modelo *As-Is* foi elaborado a partir de entrevistas e discussões com *stakeholders*, como a coordenação de estágios e os alunos. Ele descreve o fluxo atual de atividades, que inclui a divulgação de vagas por e-mail, grupos de *WhatsApp*, e documentos em formato PDF enviados semanalmente. Essa abordagem evidenciou problemas como:

- A dispersão das informações;
- A dificuldade de filtragem de vagas relevantes;

- A falta de centralização das oportunidades.

Na Figura 2, é possível visualizar o modelo de *BPM AS-IS*:

Figura 2 BPM AS-IS



Fonte: Autoria própria

3.7.2 *BPM To-Be*

O modelo **To-Be**, por sua vez, representa a visão futura do processo, incorporando as melhorias necessárias para atender aos objetivos do projeto. Ele descreve como os fluxos de trabalho serão realizados após a implementação das mudanças, sendo um guia para o desenvolvimento de soluções mais eficientes e otimizadas.

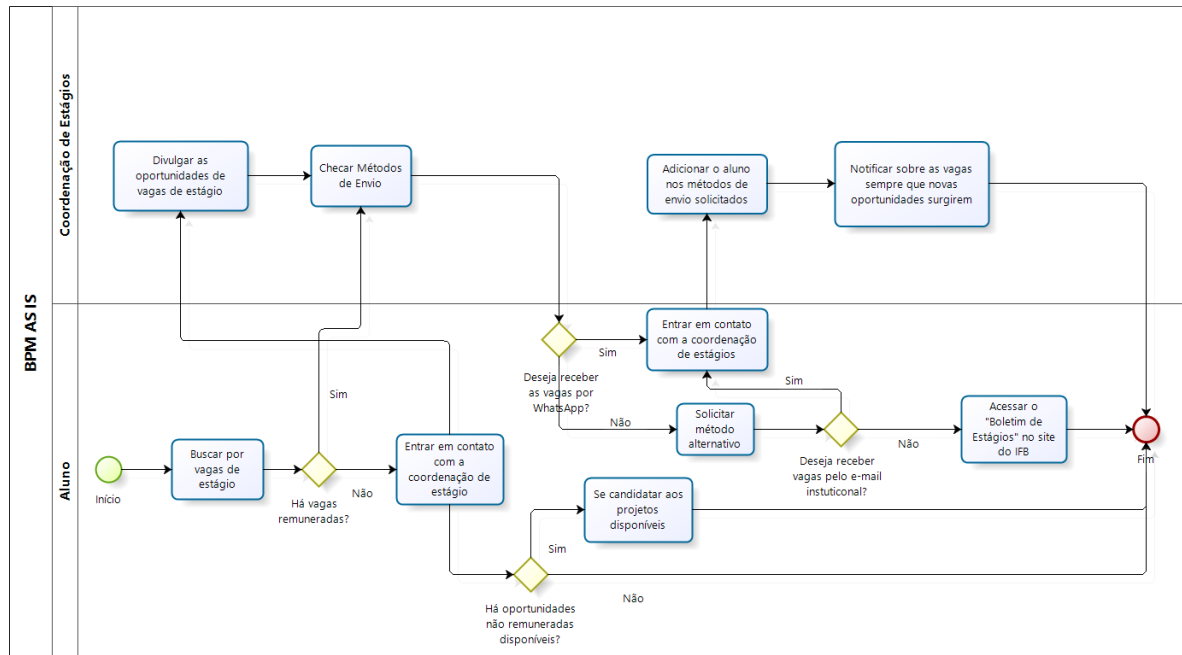
Para o sistema "IFB + Estágios", o modelo *To-Be* propõe:

- A centralização de informações sobre vagas em uma única plataforma;
- A inclusão de funcionalidades como filtragem personalizada e notificações automáticas;
- A gestão simplificada de vagas pela coordenação de estágios.

Na Figura 3, é possível visualizar o modelo de *BPM TO-BE*:

Esse modelo foi desenvolvido com base nos problemas identificados no *As-Is* e serviu de base para estruturar os requisitos e funcionalidades do sistema.

Figura 3 BPM TO BE



Fonte: Autoria própria

3.7.3 Integração com Outras Metodologias

A aplicação do BPM foi complementada pelo uso de outras metodologias, como o **Design Thinking**, que possibilitou um entendimento profundo das necessidades dos usuários durante as etapas de empatia e definição. Além disso, a metodologia *Scrum* foi utilizada para gerenciar o desenvolvimento iterativo, permitindo que o modelo *To-Be* fosse ajustado continuamente com base no *feedback* dos *stakeholders*. A análise *As-Is* e *To-Be* guiou o levantamento de requisitos e regras de negócio, garantindo que o sistema "IFB + Estágios" fosse desenvolvido com foco na solução de problemas reais. Ao mapear o estado atual e propor o estado desejado, essa abordagem assegurou que as funcionalidades implementadas estivessem alinhadas às expectativas dos usuários, aumentando as chances de sucesso do projeto.

3.8 Desenvolvimento de Protótipos

O desenvolvimento de protótipos foi uma etapa essencial no projeto do sistema "IFB + Estágios", permitindo a visualização e validação das funcionalidades e da interface antes da implementação final. Essa abordagem não apenas reduziu retrabalhos, mas também garantiu que o produto final estivesse alinhado às necessidades dos *stakeholders*, como estudantes e a coordenação de estágios.

O uso da ferramenta Figma foi fundamental para esta etapa. O Figma possibilitou o design iterativo da interface do sistema, promovendo discussões e ajustes em tempo real. Sua

funcionalidade baseada na nuvem também facilitou o trabalho colaborativo, garantindo que todos os membros da equipe pudessem contribuir de forma integrada, independentemente de sua localização.

Para guiar o desenvolvimento dos protótipos, as metodologias discutidas anteriormente desempenharam um papel crucial. A aplicação do **design thinking** permitiu explorar as necessidades dos usuários de forma criativa e empática, fornecendo insights valiosos para o design da interface. Já o **Scrum** estruturou o processo em *sprints*, possibilitando o planejamento e execução de protótipos em ciclos curtos, com entregas incrementais e feedback contínuo.

Além disso, a **análise SWOT** contribuiu para identificar pontos fortes e oportunidades, orientando as decisões de design para maximizar o impacto positivo do sistema. Dessa forma, os protótipos foram desenvolvidos não apenas como um reflexo estético das funcionalidades, mas também como uma ferramenta estratégica que uniu as dimensões técnica, pedagógica e operacional do projeto.

Essa abordagem holística assegurou que os protótipos fossem mais do que representações visuais do sistema, funcionando como um elo entre as necessidades dos usuários, as diretrizes metodológicas e os objetivos do projeto. Nas subseções a seguir, são detalhados os protótipos desenvolvidos e as funcionalidades validadas durante esta etapa.

Abaixo, na seção de "Descrição de Telas do Sistema", apresentamos a descrição detalhada de todas as telas do sistema "IFB + Estágios", abordando suas funcionalidades, elementos principais e o propósito de cada interface no contexto do projeto. Para facilitar a compreensão e visualização, as imagens correspondentes a cada tela estão disponíveis na seção de "Apêndice".

3.8.1 Descrição das Principais Telas do Sistema

Página Inicial (Usuário e Administrador): A Figura ?? ilustra o protótipo da página inicial do sistema, onde os usuários e administradores são recebidos e orientados a iniciar a navegação. Essa página serve como um ponto de partida intuitivo, oferecendo uma visão geral das funcionalidades disponíveis e acesso direto às principais ações que o usuário pode realizar.

Figura 4 Protótipo da Página inicial - Usuário.

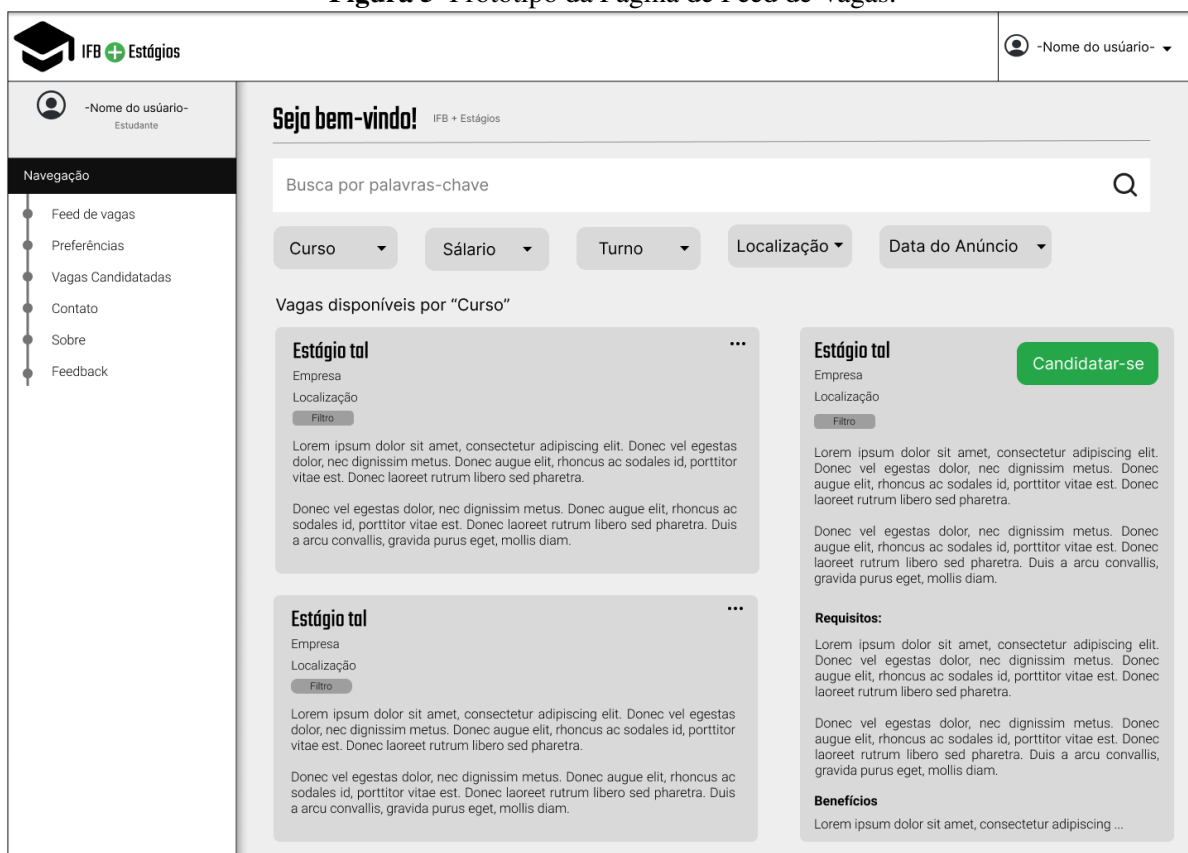


Fonte: Autoria própria

Página de Acesso Inicial: O protótipo apresentado na Figura 4 demonstra a interface de acesso, que foi projetada para ser um processo simples e direto, sendo obrigatório apenas um e-mail institucional e uma senha para acesso.

Página do *Feed* de Vagas: Na Figura 5. Esta página é o ponto central do software, onde os usuários visualizam as vagas disponíveis. Esta tela foi desenhada para que os estudantes visualizem as vagas de estágio de forma organizada e possam filtrar as oportunidades com base em critérios específicos, além de possuir um menu de navegação na lateral para acessar as páginas do software.

Figura 5 Protótipo da Página de Feed de Vagas.



Fonte: Autoria própria

4

DESENVOLVIMENTO

Esta seção apresenta o processo técnico do desenvolvimento do sistema "IFB + Estágios", detalhando os principais componentes desde a definição de regras de negócio até a implementação do código. O objetivo é descrever as principais etapas que levaram à construção da solução, passando pela análise dos requisitos, arquitetura do sistema e os testes que garantiram sua funcionalidade e usabilidade.

Primeiramente, são descritas as Regras de Negócio que orientam o comportamento do sistema e garantem que ele esteja alinhado às necessidades dos usuários. Em seguida, abordaremos a Análise de Requisitos, que inclui a definição de requisitos funcionais e não funcionais, fundamentais para assegurar que o sistema atenda às expectativas.

Na sequência, são apresentados os diagramas de Casos de Uso, que ilustram as interações entre os usuários e o sistema, e a Arquitetura do Sistema, onde são detalhados os componentes técnicos e as interações entre eles. Além disso, esta seção inclui os testes e validação aplicados para garantir a qualidade da solução desenvolvida.

Por fim, será discutida a Modelagem de Dados, utilizando o DER (Diagrama Entidade-Relacionamento) e o MER (Modelo Entidade-Relacionamento) para descrever a estrutura do banco de dados, bem como a implementação do código-fonte, destacando as principais decisões técnicas e tecnologias empregadas.

4.1 Arquitetura do sistema

A arquitetura do sistema "IFB + Estágios" foi desenvolvida com base nos princípios do *React*, priorizando a componentização e a modularidade. De acordo com Accomazo *et al.* (2017), o *React* permite construir interfaces de usuário organizadas em componentes reutilizáveis e independentes, o que facilita a manutenção e a escalabilidade do projeto. Cada componente é responsável por gerenciar sua própria lógica e renderização, promovendo um design mais limpo e organizado. O *React* utiliza um modelo baseado em componentes e um Virtual DOM eficiente para garantir atualizações rápidas e responsivas na interface do usuário. Isso é particularmente útil para aplicações que requerem alta interatividade, como o sistema desenvolvido. Ainda de acordo com o livro, a abordagem de criar componentes como blocos construtivos reduz significativamente a complexidade do desenvolvimento, uma vez que os mesmos podem ser

facilmente testados, mantidos e integrados a outras partes do sistema.

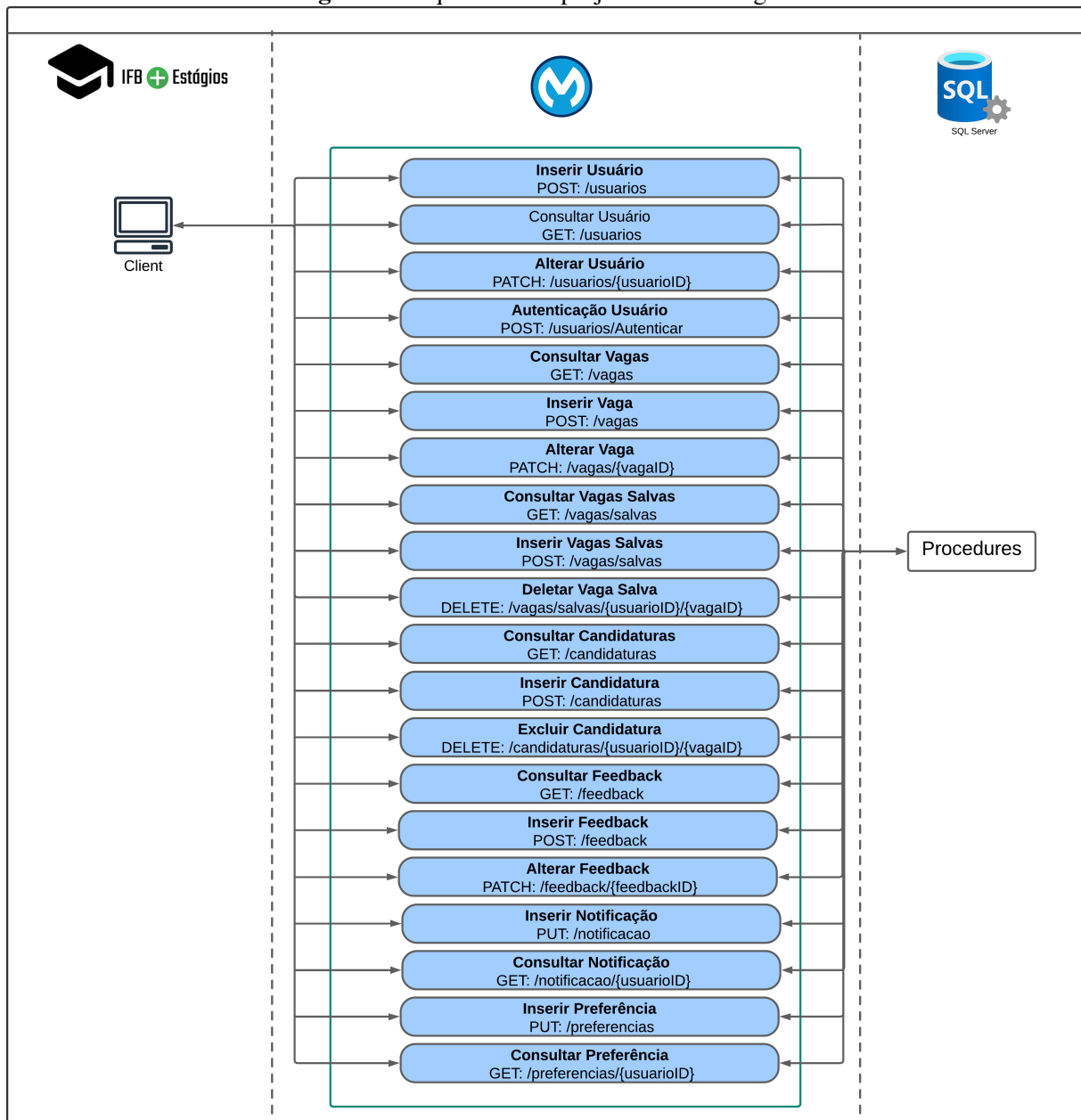
No projeto "IFB + Estágios", a arquitetura do *React* foi usada para criar componentes que representam as funcionalidades principais, como visualização de vagas, login e recuperação de senha. Cada funcionalidade foi encapsulada em componentes separados, promovendo um fluxo de trabalho modular. Essa abordagem baseada em componentes facilita a manutenção e expansão do sistema, garantindo que novas funcionalidades possam ser adicionadas sem impactar a estrutura existente. O modelo de Virtual DOM do *React* também assegura uma performance superior, especialmente em aplicações que demandam interatividade e atualizações dinâmicas. A escolha do *React* para a arquitetura do sistema foi estratégica, considerando sua popularidade e suporte contínuo pela comunidade, conforme destacado por Accomazo *et al.*, (2017). Isso assegura que a equipe de desenvolvimento tenha acesso a uma vasta quantidade de recursos e boas práticas, bem como suporte a longo prazo para atualizações e melhorias na aplicação.

Além disso, para gerenciar a conexão com o banco de dados e as *APIs* do sistema, o *Mulesoft* foi escolhido como uma solução robusta e eficiente. Conforme destacado na documentação oficial do *Mulesoft*, a plataforma simplifica a criação, publicação e gestão de *APIs*, além de oferecer segurança e escalabilidade para operações de *backend*. Essa escolha reduziu significativamente a complexidade do código no lado cliente e possibilitou a integração de futuros serviços de forma mais ágil (Mulesoft, 2024).

O *Mulesoft* desempenhou um importante papel na abstração da lógica de conexão ao banco de dados, permitindo que os desenvolvedores focassem na implementação de funcionalidades orientadas ao usuário.

A Figura 6 abaixo ilustra de forma objetiva a arquitetura do sistema IFB + Estágios, demonstrando a interação entre o cliente, as *APIs* gerenciadas pela *Mulesoft* e o banco de dados *Structured Query Language (SQL) Server*:

Figura 6 Arquitetura do projeto IFB + Estágios



Fonte: Autoria própria

Durante o desenvolvimento do projeto, o *Git* foi utilizado para controle de versionamento do código, enquanto o *GitHub* serviu como plataforma de colaboração e armazenamento em nuvem. Essa combinação garantiu um histórico detalhado de todas as alterações realizadas, facilitando o rastreamento e a reversão de mudanças quando necessário. Ademais, o *GitHub* foi usado para gerenciar o repositório de código de forma segura e acessível, permitindo a colaboração em tempo real entre os membros da equipe. O uso de ferramentas de versionamento também assegurou maior controle e organização do projeto.

4.2 Análise de Requisitos do Sistema

A análise de requisitos é uma etapa fundamental no desenvolvimento de qualquer sistema de software, pois define as funcionalidades e restrições que devem ser atendidas para garantir que o produto final satisfaça as necessidades dos usuários e dos demais *stakeholders*. No caso do sistema "IFB + Estágios", esse processo foi realizado com base em entrevistas, e discussões com os principais *stakeholders*, incluindo coordenadores de estágio e alunos. A aplicação da metodologia de *Design Thinking* foi essencial para estruturar e enriquecer essa etapa, promovendo uma abordagem centrada no usuário.

O levantamento dos requisitos e das regras de negócio foi conduzido de maneira estruturada, permitindo a identificação dos principais objetivos do sistema, assim como os desafios e oportunidades a serem abordados. Esses elementos foram posteriormente documentados em uma matriz de rastreabilidade, que assegura o alinhamento entre as regras de negócio, os requisitos funcionais e os requisitos não funcionais, facilitando o monitoramento e a validação do projeto durante todas as etapas de desenvolvimento.

Nas subseções a seguir, são detalhados os principais elementos da análise de requisitos, incluindo a descrição das regras de negócio, os requisitos funcionais e não funcionais, bem como a matriz de rastreabilidade.

4.2.1 Regras de Negócio

As regras de negócio são declarações que definem ou restringem aspectos de um sistema, alinhando-o às políticas organizacionais e operacionais. , Elas garantem que o software seja projetado para atender às condições e restrições específicas da organização (Pressman; Maxim, 2021). Nesta seção, detalhamos as regras de negócio do sistema "IFB + Estágios", que servem como base para a definição de seus requisitos e funcionalidades. Abaixo, seguem as regras de negócio levantadas para o desenvolvimento do sistema:

RN001 - Diferentes níveis de acesso devem ser atribuídos aos usuários (por exemplo, administradores e alunos) para garantir que apenas usuários autorizados possam realizar determinadas ações no sistema.

RN002 - Para editar o conteúdo do sistema o usuário deve fazer parte da equipe da coordenação de estágio da instituição e estar cadastrado como usuário administrador do sistema.

RN003 - Os usuários podem se candidatar a diversas vagas conforme disponibilidade no sistema.

RN004 - Os usuários devem ser alunos do IFB, campus Brasília, e estar atualmente matriculados em cursos oferecidos pela instituição.

RN005 - Os usuários devem fornecer informações precisas e verificáveis ao criar seus perfis.

RN006 - Cada usuário deve ter um perfil único associado a um endereço de e-mail da instituição válido.

RN007 - O administrador pode publicar, editar e excluir vagas de emprego no sistema.

RN008 - As vagas e informações publicadas devem obedecer a políticas de não discriminação, não podendo conter conteúdo ofensivo, discriminatório ou ilegal.

RN009 - As vagas devem estar em conformidade com as regulamentações de estágio.

RN010 - Todas as vagas publicadas devem incluir informações detalhadas sobre curso, contato e local de candidatura.

RN011 - Os usuários podem filtrar vagas com base em critérios como curso, período, salário, etc.

RN012 - O sistema deve fornecer informações com base nos critérios colocados pelo usuário.

RN013 - Os usuários poderão compartilhar informações de vagas através do link da vaga.

RN014 - As informações pessoais dos usuários, como nome, endereço de e-mail e número de telefone, devem ser protegidas e não podem ser compartilhadas com terceiros sem consentimento.

RN015 - O sistema deve estar em conformidade com regulamento da Lei Geral de Proteção de Dados (LGPD).

RN016 - Os usuários devem ter a opção de recuperar suas senhas por meio de um processo seguro de redefinição de senha.

RN017 - O sistema deve realizar backups regulares dos dados para evitar perda de informações importantes e ter um plano de recuperação de desastres em vigor.

RN018 - Os usuários devem ser notificados sobre atualizações importantes e novas funcionalidades do sistema.

RN019 - Os usuários podem optar por receber notificações por e-mail de novas vagas com base em suas preferências.

RN020 - O sistema deve ser regularmente atualizado para melhorar a experiência do usuário e adicionar novos recursos se necessário.

RN021 - O sistema deve receber *feedbacks* de usuários, quanto a performance e sugestões de melhorias.

RN022 - O sistema deve ser desenvolvido considerando padrões de acessibilidade para garantir que pessoas com deficiência possam utilizar todas as funcionalidades de forma equitativa.

4.2.2 *Requisitos Funcionais*

Os requisitos funcionais descrevem as funcionalidades ou serviços que o sistema deve fornecer para atender às necessidades dos usuários e alcançar os objetivos do sistema. Esses requisitos detalham as interações entre o usuário e o sistema, especificando o que o sistema deve fazer em termos de entradas, comportamentos e saídas esperadas (Pressman; Maxim, 2021). Exemplos incluem autenticação de usuários, envio de notificações e gerenciamento de banco de dados. Segue, abaixo, o conjunto de requisitos funcionais de identificados para nortear o desenvolvimento do sistema:

RF001 - O sistema deve possuir meios para o cadastro de usuários que utilizem o e-mail institucional do IFB.

RF002 - O sistema deve permitir o cadastro de usuários administradores (coordenação de estágio).

RF003 - O sistema deve manter as informações dos usuários cadastrados.

RF004 - O sistema deve permitir que os usuários recuperem a sua senha.

RF005 - O sistema deve dispor de meios para o cadastro de vagas de estágio pelo administrador.

RF006 - O sistema deve manter as informações das vagas de estágio cadastradas pelos usuários administradores.

RF007 - O sistema deve permitir a filtragem de vagas de acordo com as tags selecionadas pelo usuário, onde será possível filtrar por:

- Curso;
- Modalidade;
- Salario;
- PCD;
- Busca de palavra-chave;

RF008 - O sistema deve sugerir vagas de acordo com os interesses selecionados pelo usuário.

RF009 - O sistema deve notificar os usuários automaticamente de novas vagas de acordo com seus interesses.

RF010 - O sistema deve permitir que os usuários modifiquem as configurações de notificações.

RF011 - O sistema deve permitir que os usuários compartilhem as vagas.

RF012 - O sistema deve permitir o envio de sugestões feitas pelos usuários.

4.2.3 *Requisitos Não Funcionais*

Os requisitos não funcionais especificam os critérios de qualidade e as restrições do sistema, mas não estão diretamente relacionados a funcionalidades específicas (Pressman; Maxim, 2021). Esses requisitos abrangem aspectos como desempenho, segurança, confiabilidade, escalabilidade e usabilidade. Eles definem como o sistema deve operar e quais restrições ele deve obedecer, impactando a arquitetura e o design técnico do sistema. Por exemplo, o tempo de resposta para uma ação deve ser inferior a dois segundos, e o sistema deve estar disponível 99,9% do tempo.

Ambos os tipos de requisitos são fundamentais para o sucesso do projeto, pois garantem que o sistema atenda tanto às expectativas dos usuários quanto às especificações técnicas e de qualidade. A seguir, são apresentados os requisitos não funcionais definidos para garantir a qualidade, desempenho e usabilidade do sistema durante seu desenvolvimento e utilização.

RNF001 - O sistema deve ser responsivo, permitindo o acesso em diferentes dispositivos, como celulares, computadores e tablets.

RNF002 - O sistema será compatível com os principais navegadores:

- Google Chrome
- Edge
- Opera
- Firefox

RNF003 - O sistema deve estar disponível no idioma Português.

RNF004 - O sistema deve estar disponível 99,7% do ano, minimizando períodos de inatividade para manutenção ou falhas técnicas.

RNF005 - O sistema deve ser capaz de processar solicitações de busca de vagas e candidaturas em menos de 3 segundos, mesmo durante horários de pico.

RNF006 - O sistema deve cumprir as regulamentações de proteção de dados pessoais, como a Lei Geral de Proteção de Dados (LGPD).

RNF007 - O sistema deve ser totalmente funcional em dispositivos de acesso aos protocolos HTTPS.

RNF008 - Apenas os usuários administradores do sistema previamente definidos podem fazer alterações no conteúdo do sistema.

RNF009 - O código-fonte do sistema deve ser bem documentado, com comentários e padrões de codificação definidos para facilitar futuras atualizações e manutenções.

RNF010 - O sistema será desenvolvido com as seguintes tecnologias: HTML5, CSS3, JavaScript e React para a construção de seus componentes, estrutura, estilo e funcionalidades, além de APIs desenvolvidas na plataforma Mulesoft para intermediar a comunicação entre o cliente e o banco de dados. O banco de dados utilizado será o Microsoft SQL Server.

RNF011 - O sistema deve possuir mecanismos de backup para assegurar a integridade e a segurança dos dados armazenados.

4.3 Matriz de Rastreabilidade

A matriz de rastreabilidade é uma ferramenta que estabelece vínculos entre diferentes componentes do projeto, como regras de negócio, requisitos funcionais e não funcionais. A rastreabilidade é um mecanismo fundamental para garantir que todos os requisitos sejam atendidos e validados durante o desenvolvimento (Pressman; Maxim, 2021). Nesta seção, apresentamos a matriz de rastreabilidade do projeto, demonstrando como cada regra de negócio está associada aos requisitos que sustentam o sistema.

Tabela 3 Matriz de rastreabilidade entre regras de negócio, requisitos funcionais e não funcionais

Regra de negócio	Requisito funcional	Requisito não-funcional
RN001	RF001, RF002	RNF008
RN002	RF001, RF002, RF004, RF005	RNF008
RN003	RF006, RF007	RNF005, RNF007
RN004	RF001	RNF006
RN005	RF001, RF002, RF003	RNF006
RN006	RF001, RF002, RF003	RNF006
RN007	RF005	RNF008
RN008	RF005, RF006	RNF003
RN009	RF005, RF006	RNF010
RN010	RF005, RF006, RF007	RNF008
RN011	RF007	RNF005
RN012	RF008	RNF005
RN013	RF011	RNF001, RNF002, RNF007
RN014	RF003	RNF001, RNF006
RN015	-	RNF006
RN016	RF004	RNF006, RNF007
RN016	-	RNF004, RNF006, RNF007, RNF009, RNF011
RN017	RF010	RNF003, RNF007
RN019	RF009, RF010	RNF003, RNF007
RN020	-	RNF009, RNF010
RN021	RF012	RNF007
RN022	RF012	RNF001, RNF002, RNF003 , RNF009

Fonte: Autoria própria

4.4 Diagramas

Os diagramas são ferramentas fundamentais na documentação e no planejamento de sistemas de software, pois oferecem uma visão estruturada e detalhada de diferentes aspectos do sistema. Eles permitem a comunicação clara entre os desenvolvedores, *stakeholders* e outros envolvidos no projeto, promovendo uma compreensão compartilhada das funcionalidades e da estrutura do sistema.

No desenvolvimento de software, os diagramas atuam como ferramentas visuais fun-

damentais para modelar, especificar, construir e documentar a arquitetura e o comportamento de sistemas de software. A Linguagem de Modelagem Unificada (UML), conforme descrita por Glasson (1999), fornece um padrão consolidado para representar tanto aspectos estruturais quanto comportamentais dos sistemas. Essa abordagem é essencial para gerenciar a complexidade dos projetos de software modernos, permitindo que as partes interessadas compreendam e comuniquem de forma eficaz os componentes e as interações do sistema.

Os diagramas capturam requisitos, visualizam relacionamentos e facilitam o refinamento iterativo dos designs ao longo do ciclo de vida do desenvolvimento. Diagramas da UML, como os de casos de uso e classes, sequência e atividades, abordam diferentes visões arquitetônicas, garantindo uma representação abrangente do sistema. Por exemplo, os diagramas de classe modelam relacionamentos estáticos, enquanto os diagramas de atividade enfatizam fluxos de trabalho e processos.

No contexto do desenvolvimento do sistema "IFB + Estágios", os diagramas são cruciais para ilustrar a funcionalidade e a estrutura do sistema, oferecendo uma visão clara das interações dos usuários, organização dos dados e processos internos. Ao utilizar os princípios apresentados na *Unified Modeling Language (UML)* e fundamentados nesta obra, garantimos um design robusto e escalável, alinhado às necessidades dos usuários

Nesta seção, serão apresentados os diagramas desenvolvidos para o sistema "IFB + Estágios", cobrindo aspectos funcionais, estruturais e relacionais. Os diagramas incluídos são:

- **Diagrama de Casos de Uso Macro:** Apresenta uma visão geral das interações entre os atores e os principais casos de uso do sistema.
- **Especificação de Casos de Uso:** Detalha os fluxos de interação entre os atores e o sistema, especificando os cenários de uso.
- **Diagrama de Classe:** Representa a estrutura orientada a objetos do sistema, descrevendo classes, atributos, métodos e os relacionamentos entre elas.
- **Diagrama Entidade-Relacionamento (DER):** Modela a estrutura lógica do banco de dados, destacando entidades, atributos e relacionamentos.
- **Diagrama Relacional:** Expande o diagrama DER, incorporando detalhes técnicos adicionais, como cardinalidade e restrições.

Cada diagrama desempenha um papel específico no desenvolvimento do sistema, contribuindo para uma modelagem mais precisa e completa. Eles serão detalhados ao longo desta seção, destacando suas características, propósitos e as informações que fornecem para o projeto.

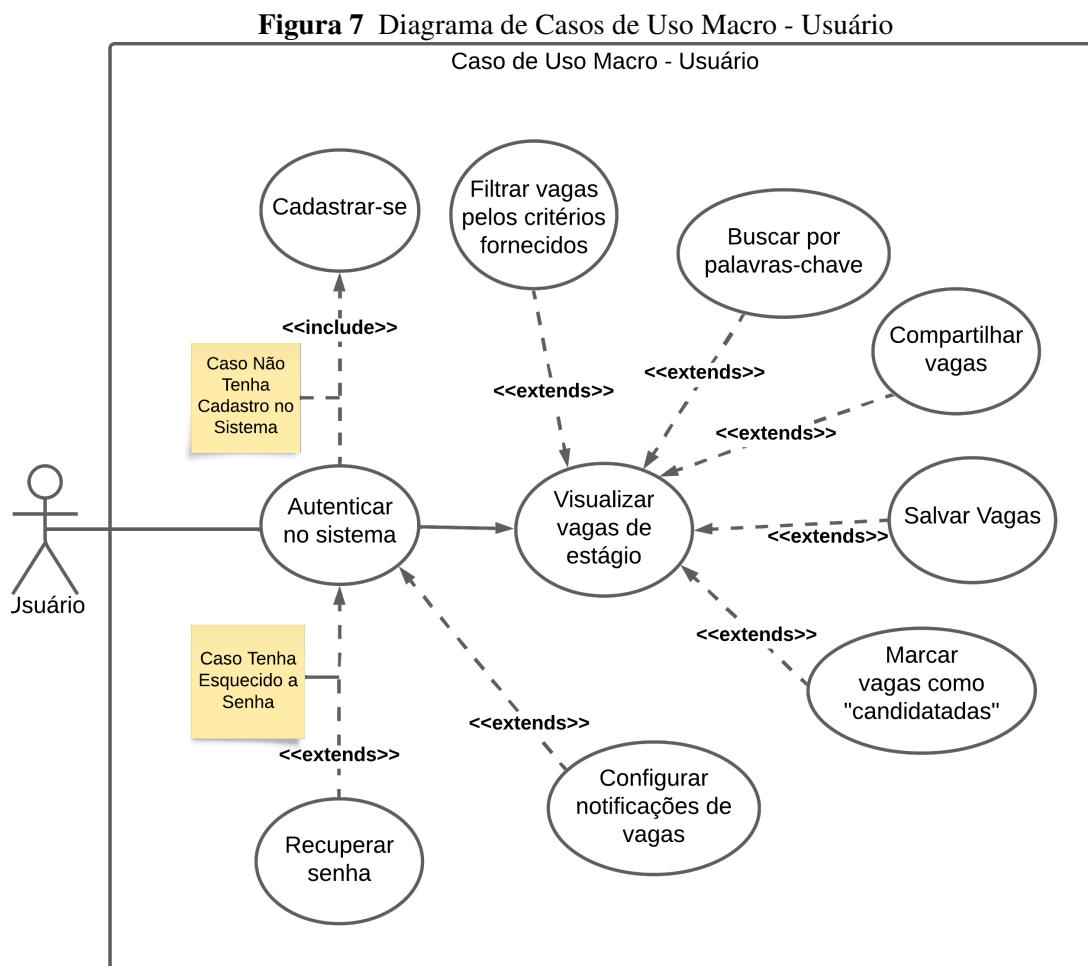
4.4.1 *Diagrama de Casos de Uso Macro*

O diagrama de casos de uso macro fornece uma visão geral das principais interações entre os atores e as funcionalidades do sistema. Ele permite identificar os principais casos de uso

e os atores envolvidos, servindo como um ponto de partida para a análise detalhada do sistema. Os diagramas de casos de uso são essenciais para capturar os requisitos funcionais de forma visual e compreensível (Pressman; Maxim, 2021).

No contexto do desenvolvimento do sistema "IFB + Estágios", optou-se por criar dois diagramas de casos de uso macro distintos: um para os usuários (estudantes) e outro para os administradores (coordenação de estágios). Essa abordagem foi adotada para melhorar a compreensão das funcionalidades do sistema, destacando claramente as interações específicas de cada grupo de atores com as principais funcionalidades oferecidas.

Abaixo segue o diagrama de casos de uso macro para o usuário (Figura 7). Ele ilustra as principais interações que os usuários, representados pelos estudantes, têm com o sistema "IFB + Estágios", detalhando funcionalidades como autenticação, visualização de vagas, configuração de notificações e outras ações disponíveis para esse ator.

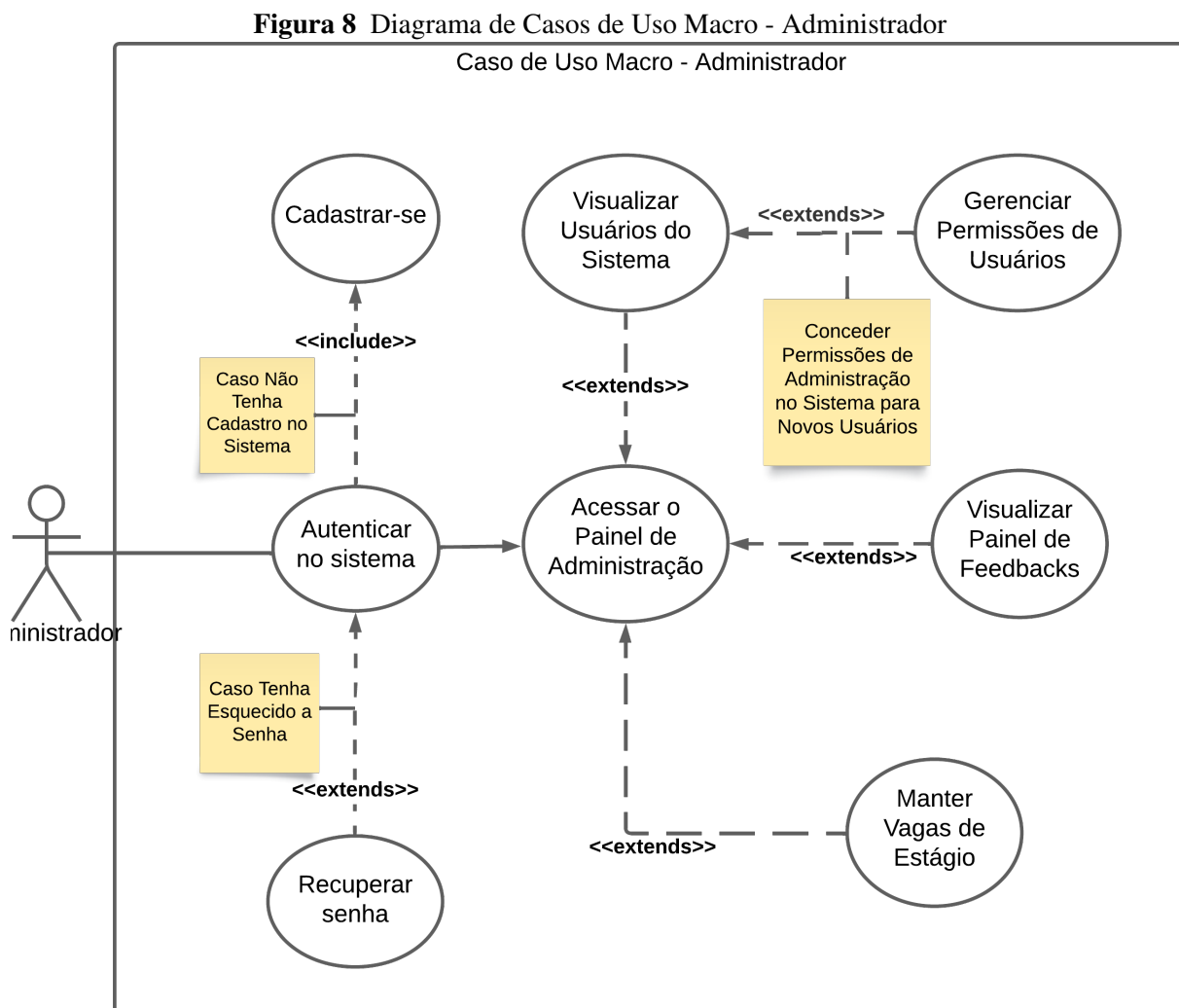


Fonte: Autoria própria

Os casos de uso destacados incluem ações como autenticação no sistema, visualização de vagas, aplicação de filtros de busca, configuração de notificações e marcação de vagas como candidatas. Além disso, estão presentes funcionalidades importantes, como recuperação de senha e compartilhamento de vagas, que ampliam a usabilidade e acessibilidade do sistema para

os estudantes. Essa organização clara das interações do usuário proporciona uma visão geral da jornada do estudante dentro do sistema, destacando os principais pontos de contato com as funcionalidades disponíveis.

Em seguida, é apresentado o diagrama de casos de uso macro para o administrador (Figura 8). Este diagrama destaca as interações específicas da coordenação de estágios, incluindo o gerenciamento de vagas, visualização de feedbacks, controle de permissões de usuários e demais responsabilidades atribuídas a este ator.



Fonte: Autoria própria

Entre os casos de uso destacados estão ações como acesso ao painel administrativo, visualização de usuários do sistema, manutenção de vagas de estágio, e gerenciamento de permissões de usuários. Essas funcionalidades permitem que os administradores acompanhem de forma centralizada as atividades no sistema, otimizando a administração e garantindo a conformidade com os processos institucionais.

Esse diagrama reflete o papel estratégico da coordenação de estágios no gerenciamento do sistema, destacando como cada funcionalidade foi projetada para atender às necessidades institucionais e melhorar a eficiência das operações administrativas. Ao isolar as interações do ad-

ministrador em um diagrama específico, reforça-se a clareza e organização das responsabilidades atribuídas a esse ator no sistema.

4.4.2 Especificação de Casos de Uso

A especificação de casos de uso é uma ferramenta fundamental para representar as interações entre os usuários (atores) e o sistema em desenvolvimento. Conforme apresentado por Glasson (1999) no *The Unified Modeling Language User Guide*, esse tipo de diagrama desempenha um papel crucial na modelagem inicial de sistemas, sendo utilizado para capturar requisitos funcionais e detalhar os cenários que refletem o comportamento do sistema sob diferentes condições.

Complementando o diagrama macro, o diagrama de especificação de casos de uso detalha os fluxos de interação entre os atores e o sistema. Ele descreve os passos necessários para a realização de cada caso de uso, permitindo uma análise mais precisa dos cenários e das condições de interação.

Abaixo é possível visualizar com mais detalhes a especificação de casos de uso, que norteou a composição do projeto:

CDU01 - Autenticar no Sistema
Resumo: Permitir que o usuário faça login no sistema inserindo suas credenciais.
Atores: Usuário, Administrador
Pré-condição: O usuário deve ter uma conta cadastrada no sistema.
Fluxo principal: <ol style="list-style-type: none">1. O usuário acessa a página de login.2. O usuário insere o e-mail e a senha cadastrados.3. O sistema autentica o usuário e permite o acesso ao sistema.
Fluxo alternativo: <ol style="list-style-type: none">1. Se as credenciais forem inválidas, o sistema exibe uma mensagem de erro e solicita nova tentativa.2. Se o usuário não possui conta, o sistema exibe a opção de registro.

CDU02 - Recuperar Senha
Resumo: Permitir que o usuário recupere a senha caso tenha esquecido.
Atores: Usuário
Pré-condição: O usuário deve ter um e-mail cadastrado no sistema.
Fluxo principal: <ol style="list-style-type: none">1. O usuário acessa a página de login e seleciona a opção de recuperação de senha.2. O sistema solicita que o usuário insira o e-mail cadastrado.3. O usuário fornece o e-mail associado à sua conta.4. O sistema envia um código de recuperação ou link para o e-mail do usuário.5. O usuário utiliza o código ou link para redefinir a senha.
Fluxo alternativo: <ol style="list-style-type: none">1. Se o e-mail não estiver cadastrado, o sistema exibe uma mensagem de erro e solicita que o usuário verifique as informações inseridas.

CDU03 - Cadastrar no Sistema
Resumo: Permitir que novos usuários se cadastrem no sistema utilizando o e-mail institucional.
Atores: Usuário
Pré-condição: O usuário deve ter um e-mail institucional válido.
Fluxo principal: <ol style="list-style-type: none">1. O usuário acessa a página de cadastro.2. O usuário preenche os campos necessários (e-mail, nome, etc.).3. O sistema valida os dados e cria a conta do usuário.

CDU04 - Visualizar Vagas de Estágio
Resumo: Permitir que o administrador e o usuário visualizem as vagas de estágio disponíveis.
Atores: Usuário, Administrador
Pré-condição: O ator deve estar logado no sistema.
Fluxo principal: <ol style="list-style-type: none">1. O ator (usuário ou administrador) acessa a seção de vagas.2. O sistema exibe todas as vagas de estágio disponíveis, apresentando detalhes como título, empresa, localização e requisitos.
Fluxo alternativo: <ol style="list-style-type: none">1. Se não houver vagas disponíveis, o sistema exibe uma mensagem informando que não há vagas no momento.

CDU05 - Filtrar Vagas
Resumo: Permitir que o usuário aplique filtros para refinar a busca por vagas de estágio.
Atores: Usuário
Pré-condição: O usuário deve estar logado no sistema e ter acesso à seção de vagas.
Fluxo principal: <ol style="list-style-type: none">1. O usuário acessa a seção de vagas.2. O usuário seleciona filtros específicos (como curso, período, localização e modalidade).3. O sistema exibe as vagas que atendem aos critérios selecionados.
Fluxo alternativo: <ol style="list-style-type: none">1. Se nenhum resultado for encontrado com os critérios aplicados, o sistema exibe uma mensagem sugerindo a remoção ou ajuste dos filtros.

CDU06 - Buscar por Palavras-chave
Resumo: Permitir que o usuário faça uma busca por vagas de estágio utilizando palavras-chave.
Atores: Usuário
Pré-condição: O usuário deve estar logado no sistema.
Fluxo principal: <ol style="list-style-type: none">1. O usuário acessa a barra de pesquisa na seção de vagas.2. O usuário insere palavras-chave relacionadas ao tipo de estágio desejado.3. O sistema exibe as vagas correspondentes às palavras-chave.
Fluxo alternativo: <ol style="list-style-type: none">1. Se nenhum resultado for encontrado com as palavras-chave usadas na busca, o sistema exibe uma mensagem sugerindo que não há vagas disponíveis, com os critérios de busca utilizados.

CDU07 - Salvar Vagas
Resumo: Permitir que o usuário salve vagas de interesse para visualização posterior.
Atores: Usuário
Pré-condição: O usuário deve estar logado no sistema.
Fluxo principal: <ol style="list-style-type: none">1. O usuário acessa a seção de vagas.2. O usuário visualiza uma vaga de interesse.3. O usuário seleciona a opção de salvar a vaga.4. O sistema armazena a vaga na lista de favoritos do usuário.
Fluxo alternativo: <ol style="list-style-type: none">1. Se a vaga já estiver salva, o sistema exibe uma mensagem informando que ela já está nos favoritos.

CDU08 - Compartilhar Vagas
Resumo: Permitir que o usuário compartilhe vagas de interesse com outras pessoas.
Atores: Usuário
Pré-condição: O usuário deve estar logado no sistema.
Fluxo principal: <ol style="list-style-type: none">1. O usuário acessa a seção de vagas.2. O usuário visualiza uma vaga de interesse.3. O usuário seleciona a opção de compartilhar a vaga.4. O sistema gera um link para compartilhamento.

CDU09 - Configurar Notificações
Resumo: Permitir que o usuário configure notificações para receber atualizações sobre novas vagas.
Atores: Usuário
Pré-condição: O usuário deve estar logado no sistema.
Fluxo principal: <ol style="list-style-type: none">1. O usuário acessa a seção de configurações de notificações.2. O usuário escolhe os critérios para as notificações.3. O sistema salva as preferências de notificação do usuário.

CDU10 - Marcar Vagas como Candidatadas
Resumo: Permitir que o usuário marque as vagas às quais já se candidatou.
Atores: Usuário
Pré-condição: O usuário deve estar logado no sistema.
Fluxo principal: <ol style="list-style-type: none">1. O usuário acessa uma vaga.2. O usuário seleciona a opção "Candidatar-se".3. O sistema salva a vaga em "Vagas Candidatadas" para o usuário.

CDU11 - Fornecer Feedback
Resumo: Permitir que o usuário forneça feedback sobre uma vaga ou sobre o sistema.
Atores: Usuário
Pré-condição: O usuário deve estar logado no sistema.
Fluxo principal: <ol style="list-style-type: none">1. O usuário acessa a seção de feedback.2. O usuário insere seu feedback sobre a vaga ou sobre o sistema.3. O sistema registra o feedback fornecido.

CDU12 - Cadastrar Usuário e Administrador
Resumo: Permitir que o administrador cadastre novos usuários e administradores no sistema.
Atores: Administrador
Pré-condição: O administrador deve estar logado no sistema.
Fluxo principal: <ol style="list-style-type: none">1. O administrador acessa a seção de gerenciamento de usuários.2. O administrador insere as informações do novo usuário ou administrador.3. O sistema salva o novo cadastro.

CDU13 - Cadastrar Vagas
Resumo: Permitir que o administrador cadastre novas vagas de estágio no sistema.
Atores: Administrador
Pré-condição: O administrador deve estar logado no sistema.
Fluxo principal: <ol style="list-style-type: none">1. O administrador acessa a seção de vagas.2. O administrador seleciona a opção de cadastrar uma nova vaga.3. O administrador insere os dados da vaga, como título, descrição, requisitos e benefícios.4. O sistema salva as informações e exibe uma mensagem de confirmação.
Fluxo alternativo: <ol style="list-style-type: none">1. Se os dados obrigatórios não forem preenchidos, o sistema exibe uma mensagem de erro e solicita a correção.

CDU14 - Editar Vagas
Resumo: Permitir que o administrador edite informações de vagas de estágio previamente cadastradas.
Atores: Administrador
Pré-condição: <ul style="list-style-type: none">■ O administrador deve estar logado no sistema.■ Deve existir pelo menos uma vaga cadastrada.
Fluxo principal: <ol style="list-style-type: none">1. O administrador acessa a seção de vagas.2. O administrador seleciona uma vaga previamente cadastrada.3. O administrador edita os dados da vaga, como título, descrição, requisitos e benefícios.4. O sistema salva as alterações e exibe uma mensagem de confirmação.
Fluxo alternativo: <ol style="list-style-type: none">1. Se os dados obrigatórios não forem preenchidos ou editados corretamente, o sistema exibe uma mensagem de erro e solicita a correção.

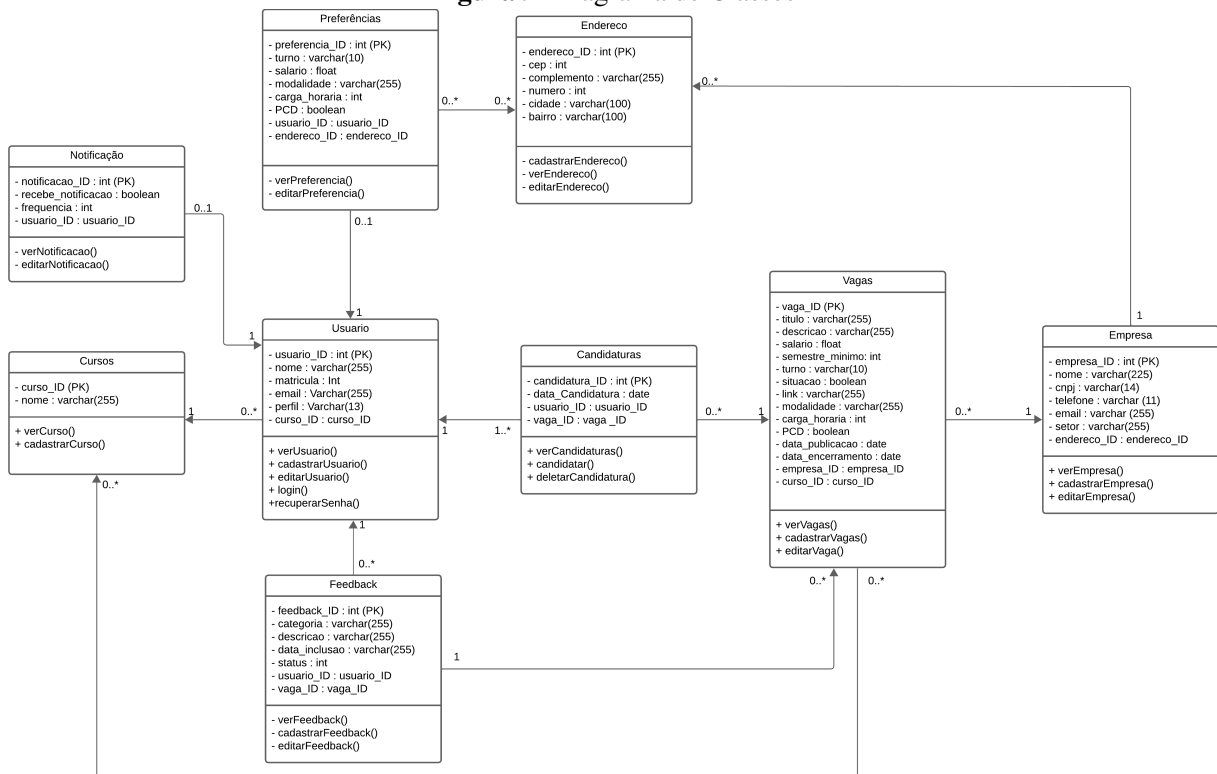
CDU15 - Visualizar Feedback
Resumo: Permitir que o administrador visualize feedbacks fornecidos pelos usuários.
Atores: Administrador
Pré-condição: O administrador deve estar logado no sistema.
Fluxo principal: <ol style="list-style-type: none">1. O administrador acessa a seção de feedbacks.2. O sistema exibe uma lista de feedbacks fornecidos pelos usuários.3. O administrador pode selecionar um feedback para visualizar detalhes adicionais.

CDU16 - Visualizar Usuários
Resumo: Permitir que o administrador visualize informações sobre os usuários cadastrados no sistema.
Atores: Administrador
Pré-condição: O administrador deve estar logado no sistema.
Fluxo principal: <ol style="list-style-type: none">1. O administrador acessa a seção de gerenciamento de usuários.2. O sistema exibe a lista de usuários cadastrados, com informações detalhadas, como nome, e-mail e status de ativação.
Fluxo alternativo: <ol style="list-style-type: none">1. Se não houver usuários cadastrados, o sistema exibe uma mensagem informando que não há registros disponíveis.

4.4.3 Diagrama de Classe

O Diagrama de Classe define a estrutura de um sistema orientado a objetos e descreve os tipos de objetos, seus atributos, métodos e relacionamentos entre classes. Esse diagrama é amplamente utilizado na modelagem de sistemas para organizar os dados e comportamentos de forma modular, permitindo que este modelo sirva como referência para o desenvolvimento técnico do sistema.

Figura 9 Diagrama de Classes



Fonte: Autoria própria

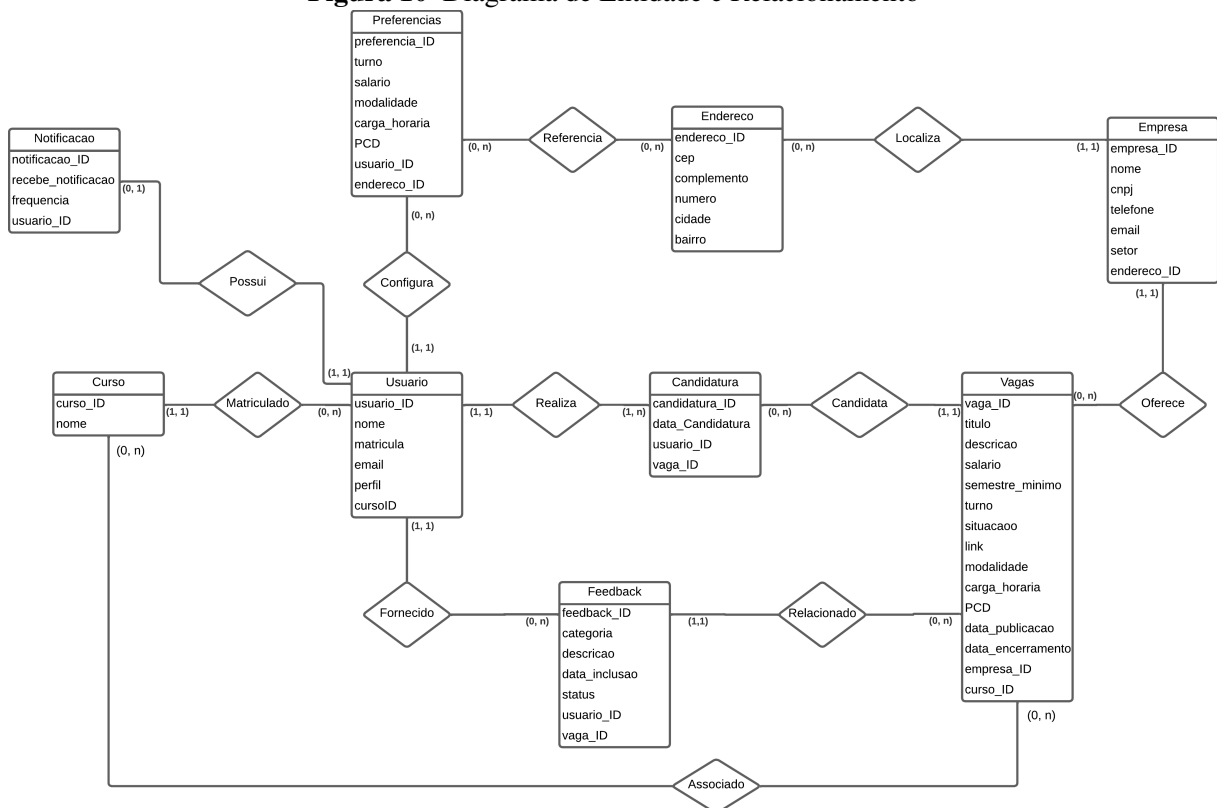
É possível visualizar as classes centrais, como Usuário, Vagas, Candidaturas, Feedback, e Cursos, estão sendo representadas com seus atributos e métodos necessários para suportar as funcionalidades do sistema. A modularidade e encapsulamento estão evidentes, permitindo fácil manutenção e expansão futura. Os relacionamentos entre as classes requer uma análise da interação entre as tabelas, assegurando que as interações entre os diferentes componentes do sistema sejam compreensíveis e gerenciáveis.

4.4.4 Diagrama Entidade-Relacionamento

O Diagrama Entidade-Relacionamento (DER) tem a função de representar a estrutura conceitual do banco de dados do sistema "IFB + Estágios". Ele ilustra as entidades envolvidas, seus respectivos atributos e os relacionamentos entre elas, proporcionando uma visão clara de como os dados serão armazenados e manipulados dentro do sistema.

No contexto do "IFB + Estágios", o DER foi elaborado para refletir as interações essenciais entre os principais componentes do sistema, tais como usuários, vagas de estágio, candidaturas e *feedbacks*. Essa modelagem permite estruturar um banco de dados coerente e alinhado aos requisitos do sistema, assegurando integridade e eficiência na manipulação das informações. Abaixo, apresenta-se o DER desenvolvido (Figura 10), que será detalhado na sequência.

Figura 10 Diagrama de Entidade e Relacionamento

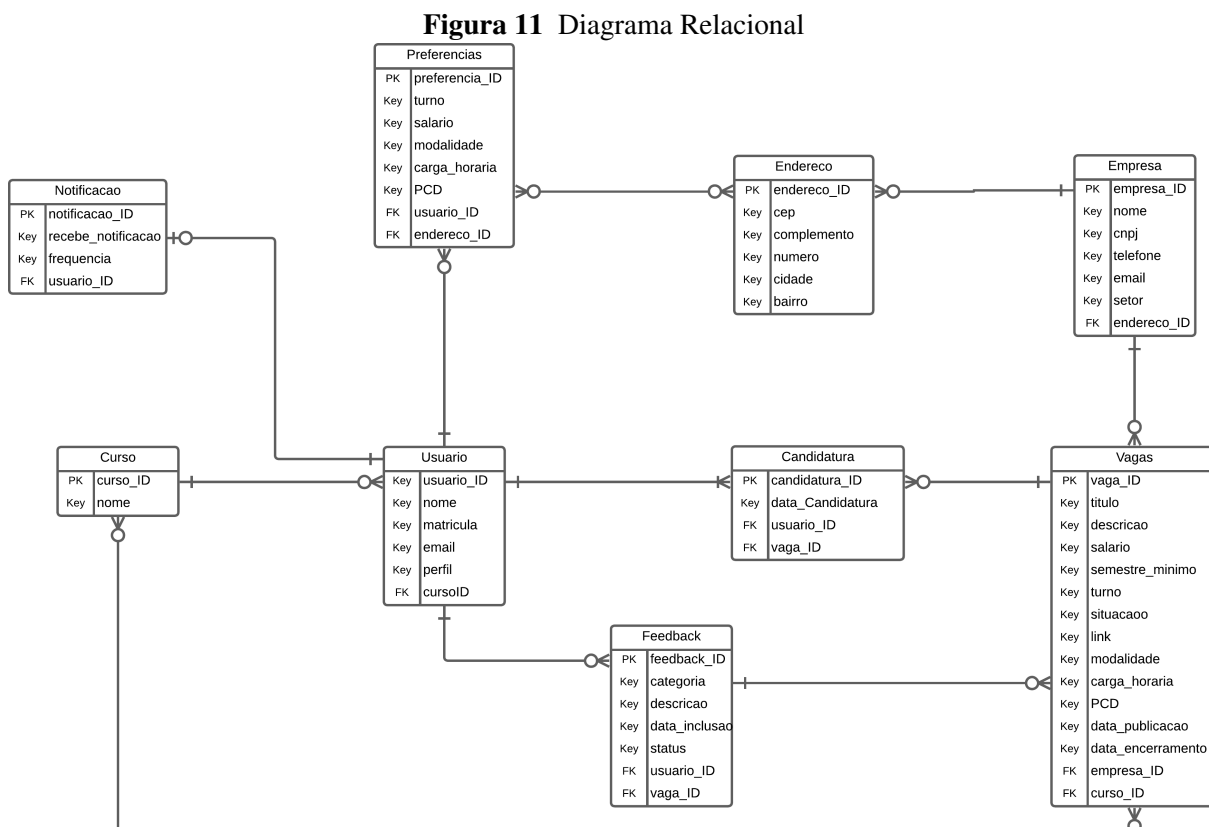


Fonte: Autoria própria

4.4.5 Diagrama Relacional

O Diagrama Relacional complementa o DER ao detalhar a implementação lógica do banco de dados, especificando tabelas, chaves primárias e estrangeiras, além da estrutura dos relacionamentos entre os dados. Essa representação é essencial para garantir que o modelo conceitual seja corretamente traduzido para a prática, otimizando a organização e o acesso às informações.

Para o sistema "IFB + Estágios", o Diagrama Relacional foi desenvolvido com o objetivo de definir claramente a estrutura das tabelas e suas conexões, assegurando integridade referencial e eficiência no armazenamento e recuperação dos dados. A Figura 11 apresenta essa modelagem, ilustrando as relações entre as entidades e proporcionando uma visão técnica detalhada do banco de dados.



Fonte: Autoria própria

4.5 Implementação do código e testes

O projeto foi implementado utilizando o *framework React*, o que proporcionou uma arquitetura baseada em componentes, modular e reutilizável. Essa abordagem permitiu o desenvolvimento de funcionalidades de forma isolada e escalável, resultando em uma aplicação mais organizada e de fácil manutenção.

A estrutura do sistema foi construída com base em componentes principais como *Navbar*, *Sidebar*, *JobCards* e *GerenciarVagas*. Cada um desses componentes foi projetado para desempenhar uma função específica dentro da interface do usuário. Por exemplo, o componente *JobCards* foi responsável pela exibição dinâmica das vagas, oferecendo recursos como visualização de detalhes, salvamento e interação com as vagas. A modularidade dos componentes reduziu a redundância de código e facilitou a integração com outras partes do sistema.

Para o gerenciamento de estados e variáveis globais, foi implementado o **Context API** do *React*. Este recurso possibilitou o armazenamento centralizado de informações essenciais,

como *nome*, *perfil* e *usuarioID*, eliminando a necessidade de passar dados manualmente entre os componentes. A implementação do contexto foi especialmente útil na integração entre as telas de autenticação e os componentes que exibem as informações do usuário, como *Navbar* e *Sidebar*. Essa abordagem assegurou consistência no estado global, mesmo durante a navegação entre diferentes páginas da aplicação.

Além disso, foram realizadas diversas otimizações na exibição de informações e no design dos componentes, como a implementação de um *layout em grid* para os *JobCards* e a adição de efeitos de *hover* para melhorar a experiência do usuário. A integração completa do sistema com o *backend* e a modularidade proporcionada pelo *React* garantiram que o *IFB + Estágios* fosse um sistema funcional, escalável e alinhado aos requisitos iniciais do projeto.

A comunicação com o *backend* foi realizada por meio de requisições HTTP para *APIs* gerenciadas pela *Mulesoft*. A *Mulesoft* foi utilizada para criar e gerenciar toda a estrutura de *Application Programming Interface (API)s*, permitindo a integração eficiente entre o *frontend* desenvolvido em *React* e o banco de dados *SQL Server*. O uso da *Mulesoft* facilitou o tratamento de dados, garantindo que os *endpoints* fossem robustos e seguros, além de assegurar a padronização das respostas.

No *backend*, o banco de dados *SQL Server* foi utilizado para armazenar informações dos usuários, e suas preferências e dados das vagas. Durante o desenvolvimento, foi necessário a criação de *procedures*, que são blocos de comandos que validam as regras de consulta, inserção, alteração e exclusão dos dados armazenados, assim, evitando duplicação de dados e conflitos entre as tabelas.

4.5.1 Principais Componentes React

O código apresentado na Figura 12 representa um componente *React* funcional, denominado *JobCards*. Este componente é responsável por buscar e exibir as vagas de estágio disponíveis, utilizando uma *API* externa para obter os dados. Ele exemplifica o uso de **hooks do React** e gerência de estado para manipulação dinâmica das vagas.

Figura 12 Código responsável pela consulta de vagas na API

```
import React, { useState, useEffect, useRef, useContext } from "react";
import { useNavigate } from "react-router-dom";
import "../styles/JobCards.css";
import { SavedJobsContext } from "../contexts/SavedJobsContext";
import { useCandidatedJobs } from "../contexts/CandidatedJobsContext";

const JobCards = ({ jobsProp = [], onAction, actionLabel = "Salvar", role }) => {
  const [jobs, setJobs] = useState(jobsProp.length > 0 ? jobsProp : []);
  const [selectedJob, setSelectedJob] = useState(null);
  const [openMenuId, setOpenMenuId] = useState(null);
  const menuRef = useRef(null);
  const navigate = useNavigate();
  const { addJobToSaved } = useContext(SavedJobsContext);
  const { addJobToCandidated } = useCandidatedJobs();

  //=====
  // API que faz a consulta das vagas a serem mostradas pelo feed
  useEffect(() => {
    if (jobsProp.length === 0 && jobs.length === 0) {
      const fetchJobs = async () => {
        try {
          const response = await fetch("http://192.168.15.5:8081/vagas");
          const data = await response.json();
          setJobs(data);
        } catch (error) {
          console.error("Erro ao buscar os jobs:", error);
        }
      };
      fetchJobs();
    }
  }, [jobsProp, jobs]);
};
```

Fonte: Autoria própria

O componente *JobCards* serve como base para exibir as vagas retornadas da *API* e integrar funcionalidades adicionais, como filtros dinâmicos e interação com o banco de dados através da *Mulesoft*.

- **Linhas 6 a 8: Declaração de Estados com *useState*** Inicialmente, são definidos

estados locais utilizando o hook *useState*. O estado *jobs* armazena a lista de vagas retornadas pela API, enquanto *selectedJob* representa a vaga atualmente selecionada pelo usuário. O estado *openMenuId* gerencia o controle de um menu suspenso associado a cada vaga.

- **Linha 14: Configuração de um *useEffect* para consulta à API** O hook *useEffect* é utilizado para executar a função de busca das vagas (*fetchJobs*) assim que o componente é montado. Este comportamento só é ativado caso as propriedades *jobsProp* (recebidas como parâmetro) e *jobs* estejam vazias. Essa verificação evita consultas desnecessárias à API.
- **Linhas 15 a 23: Função assíncrona *fetchJobs*** A função realiza uma chamada *HTTP GET* para o endpoint da API (*http://192.168.15.5:8081/vagas*). Utilizando o método *fetch*, a resposta é convertida para JSON e armazenada no estado *jobs* através do *setJobs*. Em caso de erro, a função exibe no console o problema ocorrido, facilitando a depuração.
- **Integração com Context API:** Para melhorar a modularidade, o código utiliza *useContext* para consumir os contextos *SavedJobsContext* e *CandidatedJobsContext*. Isso permite salvar vagas ou marcá-las como candidatas, acessando estados globais compartilhados no projeto.
- **Conceito de Hooks Reativos:** O uso de hooks como *useEffect*, *useState* e *useRef* demonstra a aplicação de conceitos reativos do React. Eles garantem que a interface do usuário seja atualizada dinamicamente à medida que os dados ou estados do componente são alterados.

Além disso, o componente *Gerenciar Vagas* é responsável por adicionar ou editar vagas de estágio no banco de dados. Ele utiliza uma abordagem modular para a manipulação dos dados, fazendo uso de funções assíncronas para se comunicar com a *API de backend*. Um exemplo desse processo é a função *vagaSubmit*, que é detalhada na sequência, conforme Figura 13

Figura 13 Trecho do código responsável por inserir as vagas de estágio.

```
const vagaSubmit = async (normalizedData) => {
  try {
    const response = await fetch("http://192.168.15.5:8081/vagas", {
      method: "POST",
      headers: {
        "Content-Type": "application/json",
      },
      body: JSON.stringify(normalizedData),
    });

    if (!response.ok) {
      throw new Error(`Erro: ${response.status} - ${response.statusText}`);
    }

    const data = await response.json();
    console.log("Resposta da API:", data);
  } catch (error) {
    console.log(normalizedData);
    throw error;
  }
};
```

Fonte: Autoria própria

A função *vagaSubmit* é projetada para enviar dados de vagas para a *API Representational State Transfer (REST)* do sistema. A implementação desta função segue os princípios do *design de APIs* e utiliza a biblioteca *fetch*, nativa do *JavaScript*, para realizar operações *Hypertext Transfer Protocol (HTTP) POST*. A seguir, é apresentada uma análise detalhada do funcionamento desta seção do código:

- **Envio de Dados Normalizados** A função recebe como argumento um objeto denominado *normalizedData*, que contém as informações da vaga previamente validadas e transformadas no formato esperado pela *API*. Este processo de normalização é realizado no componente principal antes da chamada da função, garantindo a conformidade dos dados enviados.
- **Requisição POST (Linhas 3 a 8)** A *API* é acessada por meio do método *fetch*, onde a URL do *endpoint* está configurada como *http://192.168.15.5:8081/vagas*. A opção *method: "POST"* especifica que os dados serão enviados para criar ou atualizar um registro no *backend*. As informações do cabeçalho (*headers*) incluem o tipo de conteúdo ("*Content-Type*": "*application/json*"), indicando que o *payload* será enviado no formato *JavaScript Object Notation (JSON)*.

- **Serialização dos Dados (Linha 8)** O corpo da requisição é gerado pela função *JSON.stringify*, que converte o objeto *normalizedData* para uma *string* no formato *JSON*. Este passo garante a compatibilidade com a *[API! ([API!)]*, uma vez que a maioria dos *endpoints REST* processam dados nesse formato.
- **Tratamento de Respostas da API (Linhas 10 a 13)** Após a requisição, a função verifica o status da resposta utilizando a propriedade *response.ok*. Caso a *API* retorne um código HTTP fora da faixa 200-299, é lançada uma exceção com informações detalhadas do erro, permitindo que ele seja tratado de forma centralizada em outros níveis do componente.
- **Log e Depuração (Linhas 15 a 18)** Em cenários de sucesso, o método *response.json()* é chamado para processar o corpo da resposta da *API*, e os dados retornados são exibidos no console para depuração. Caso um erro seja detectado, o bloco *catch* registra o erro e exibe o objeto *normalizedData* para facilitar a análise.

A função *vagaSubmit* é integrada ao fluxo principal do componente, sendo acionada quando o formulário de criação ou edição de vagas é submetido. Este fluxo é composto pelas seguintes etapas:

1. Validação dos dados inseridos no formulário.
2. Normalização dos valores, garantindo que campos numéricos, como *salário* e *cargaHorária*, sejam convertidos corretamente.
3. Chamada da função *vagaSubmit* com os dados normalizados.
4. Resposta da *API* é processada e exibida para o usuário, indicando sucesso ou possíveis erros.

O processo de desenvolvimento do *IFB + Estágios* foi acompanhado por uma estratégia de testes iterativos, com o objetivo de validar a funcionalidade dos componentes, a integridade das requisições às *APIs* e a exibição correta dos dados na interface.

Durante o desenvolvimento, os *console logs* desempenharam um papel crucial no diagnóstico de problemas e na validação dos dados em trânsito entre o *frontend* e o *backend*. Para cada requisição feita para a *API*, os dados enviados e as respostas recebidas eram registrados no console. Este procedimento facilitava a identificação de erros de lógica ou inconsistências nos dados, permitindo ajustes rápidos no código.

As requisições enviadas à *API* foram validadas para garantir que os dados estavam no formato esperado. Essa validação incluiu:

- Verificação de inserção dos dados no banco de dados.
- Verificação dos *payloads*.

Após garantir que as requisições e respostas estavam funcionando conforme o esperado, foram realizados testes na interface do usuário para verificar a exibição correta dos dados. Além dos testes automatizados e de depuração com *console logs*, foram realizados testes manuais para simular cenários reais de uso. Por exemplo:

1. Testes de login para verificar se as credenciais corretas resultavam no acesso às funcionalidades de administrador ou estudante.
2. Submissão de vagas para o banco de dados e exibição dessas vagas no componente *JobCards*.
3. Alteração de preferências de vagas, incluindo a escolha de modalidade, faixa salarial e inclusão de requisitos para garantir que as mudanças eram refletidas na *API*.

5

RESULTADOS

O presente tópico tem como objetivo validar o sistema “*IFB + Estágios*” em comparação com o modelo atual de gerenciamento de estágios, destacando os benefícios esperados em termos de usabilidade, eficiência e organização das oportunidades de estágio. Esta seção aborda os resultados esperados, as possíveis limitações do projeto e as contribuições previstas para a área acadêmica e tecnológica.

5.1 Validação dos Resultados e Atendimento aos Objetivos

A validação do sistema será realizada utilizando métodos consolidados, como os questionários SUS (*System Usability Scale*) e as heurísticas de usabilidade de Nielsen. Esses instrumentos permitem mensurar a facilidade de uso, eficiência e satisfação geral dos usuários. Espera-se que o sistema alcance uma pontuação média no questionário SUS superior a 68, o que indica uma usabilidade considerada acima da média segundo Brooke (1996). Além disso, a análise heurística deve identificar menos de três inconsistências críticas após os ajustes finais na interface.

Outro aspecto da validação será a medição da eficiência em relação ao modelo atual. Projeta-se que o tempo necessário para os alunos encontrarem vagas adequadas ao seu perfil será reduzido em até 50%, em comparação com o método atual de envio de listas de vagas por e-mail e *WhatsApp*. A centralização e personalização oferecidas pelo “*IFB + Estágios*” também devem melhorar a relevância das vagas apresentadas, atendendo às expectativas tanto dos alunos quanto da coordenação.

Os resultados obtidos são apresentados na Tabela 4, que exhibe a média das respostas, representada como "MR", para algumas questões-chave e a pontuação SUS média do sistema, representada como "PSM".

Tabela 4 Resultados da Escala de Usabilidade do Sistema (SUS)

Pergunta (SUS)	MR	PSM
Eu gostaria de utilizar este sistema com frequência	4.47	70.0
Considero o sistema desnecessariamente complexo	2.27	70.0
Eu achei o sistema fácil de usar	4.73	70.0
Acredito que precisaria da ajuda de uma pessoa com conhecimentos técnicos para conseguir usar este sistema	3.40	70.0
Percebo que as diferentes funcionalidades do sistema estão bem integradas	4.53	70.0
Acredito que o sistema apresenta inconsistências	3.80	70.0
Imagino que a maioria das pessoas aprenderia a usar este sistema rapidamente	4.80	70.0
Achei o sistema muito confuso ou difícil de navegar	2.00	70.0
Eu me senti muito confiante usando o sistema	4.87	70.0
Eu precisei aprender várias coisas antes de conseguir usar o sistema	3.13	70.0

Fonte: Autoria própria

Com base nas respostas coletadas, foi possível calcular a **média geral de satisfação** dos usuários em relação às 10 heurísticas de Nielsen. A pontuação média foi de **4.51**, indicando um alto nível de conformidade do aplicativo com os princípios de usabilidade.

Aspectos positivos: As heurísticas com maiores pontuações foram:

- **Consistência e Padrões (4.67):** Os usuários destacaram que os botões, menus e elementos visuais possuem um padrão consistente, facilitando a navegação e a usabilidade.
- **Correspondência com o mundo real (4.67):** As palavras e ícones foram considerados claros e intuitivos, demonstrando uma boa correspondência com o conhecimento prévio dos usuários.

Aspectos a melhorar:

Apesar da alta média, algumas heurísticas apresentaram pontuações ligeiramente inferiores, como:

- **Flexibilidade e eficiência de uso (4.33):** Houve sugestões para aprimorar opções de desfazer/refazer ações, como incluir mais atalhos e recursos contextuais.
- **Recuperação de erros (4.40):** Embora considerada funcional, mensagens de erro poderiam ser ainda mais detalhadas e específicas.

Os resultados mostram que o aplicativo foi bem avaliado em todas as heurísticas de Nielsen, demonstrando boa usabilidade e design centrado no usuário. Os pequenos ajustes apontados, como melhorias no suporte a erros e funcionalidades de recuperação, podem elevar

ainda mais a satisfação e eficácia do sistema. A pontuação consistente reflete que o sistema *IFB + Estágios* atende às expectativas de simplicidade, clareza e eficiência para seus usuários.

A Tabela 5 apresenta a média das respostas, representada como "MR", obtidas para cada um dos aspectos avaliados. Esses resultados são fundamentais para direcionar melhorias na interface, garantindo maior clareza, consistência e eficiência na experiência do usuário.

Tabela 5 Resultados das Heurísticas de Usabilidade de Nielsen

Pergunta (Heurísticas de Nielsen)	MR
O aplicativo informa claramente o que está acontecendo (ex.: carregamento, progresso de uma tarefa)	4.60
As palavras, ícones e instruções usadas no aplicativo são fáceis de entender e correspondem ao seu conhecimento do mundo real	4.67
Você consegue desfazer ou refazer ações facilmente no aplicativo (ex.: botões 'voltar' ou 'desfazer')	4.33
Os botões, menus e elementos visuais do aplicativo possuem um padrão consistente em todas as telas	4.67
O aplicativo ajuda a evitar erros, fornecendo avisos antes de ações importantes ou perigosas (ex.: confirmação antes de excluir algo)	4.53
Os ícones, rótulos e funcionalidades no aplicativo são facilmente reconhecíveis sem a necessidade de você memorizar ações anteriores	4.47
Você consegue realizar suas tarefas rapidamente e com facilidade no aplicativo, mesmo se for a primeira vez que o utiliza	4.40
O design do aplicativo é simples, sem informações desnecessárias que possam causar confusão	4.53
Se você cometeu um erro no aplicativo, ele forneceu uma mensagem clara para ajudá-lo a corrigir o problema	4.40
O aplicativo oferece instruções ou uma seção de ajuda clara caso você precise de orientação para realizar suas tarefas	4.53

Fonte: Autoria própria

Com base nas respostas coletadas, foi possível calcular a **média geral de satisfação** dos usuários em relação às 10 heurísticas de Nielsen. A pontuação média foi de **4.51**, indicando um alto nível de conformidade do aplicativo com os princípios de usabilidade.

5.2 Análise dos Resultados

- **Aspectos positivos:** As heurísticas com maiores pontuações foram:
 - **Consistência e Padrões (4.67):** Os usuários destacaram que os botões, menus e elementos visuais possuem um padrão consistente, facilitando a navegação e a usabilidade.

- **Correspondência com o mundo real (4.67):** As palavras e ícones foram considerados claros e intuitivos, demonstrando uma boa correspondência com o conhecimento prévio dos usuários.
- **Aspectos a melhorar:** Apesar da alta média, algumas heurísticas apresentaram pontuações ligeiramente inferiores, como:
 - **Flexibilidade e eficiência de uso (4.33):** Houve sugestões para aprimorar opções de desfazer/refazer ações, como incluir mais atalhos e recursos contextuais.
 - **Recuperação de erros (4.40):** Embora considerada funcional, mensagens de erro poderiam ser ainda mais detalhadas e específicas.

5.3 Ponderações

Os resultados mostram que o aplicativo foi bem avaliado em todas as heurísticas de Nielsen, demonstrando boa usabilidade e design centrado no usuário. Os pequenos ajustes apontados, como melhorias no suporte a erros e funcionalidades de recuperação, podem elevar ainda mais a satisfação e eficácia do sistema. A pontuação consistente reflete que o sistema *IFB + Estágios* atende às expectativas de simplicidade, clareza e eficiência para seus usuários.

A pontuação SUS média obtida foi de 70, o que indica que o sistema possui uma usabilidade aceitável, ficando acima do valor de referência de 68. Entretanto, observamos que algumas questões indicam oportunidades de melhoria. Por exemplo, a necessidade de ajuda técnica teve uma média de 3.40, sugerindo que alguns usuários podem encontrar dificuldades em utilizar o sistema sem suporte adicional.

O formulário de usabilidade foi utilizado para coletar feedback dos usuários sobre diversos aspectos do sistema. As perguntas abordaram desde a clareza das informações até a facilidade de uso e a necessidade de assistência técnica. A seguir, são apresentados os resultados das perguntas do formulário, com as respectivas referências aos gráficos:

- **Pergunta 14:** O aplicativo informa claramente o que está acontecendo (ex.: carregamento, progresso de uma tarefa).
- **Pergunta 15:** Eu acredito que gostaria de usar este sistema com frequência.
- **Pergunta 16:** Eu acredito que o sistema é desnecessariamente complexo.
- **Pergunta 17:** Eu acredito que o sistema é fácil de usar.
- **Pergunta 18:** Eu acredito que precisaria da ajuda de uma pessoa com conhecimentos técnicos para conseguir usar este sistema.
- **Pergunta 19:** Eu acredito que as várias funções deste sistema são bem integradas.

- **Pergunta 20:** Eu acredito que havia muita inconsistência neste sistema.
- **Pergunta 21:** Eu imagino que a maioria das pessoas aprenderia a usar este sistema rapidamente.
- **Pergunta 22:** Eu acredito que o sistema é muito confuso.
- **Pergunta 23:** Eu me senti muito confiante usando o sistema.
- **Pergunta 24:** Eu precisei aprender várias coisas antes de conseguir usar o sistema.
- **Pergunta 25:** As palavras, ícones e instruções usadas no aplicativo são fáceis de entender e correspondem ao seu conhecimento do mundo real.
- **Pergunta 26:** Você consegue desfazer ou refazer ações facilmente no aplicativo (ex.: botões "voltar" ou "desfazer").
- **Pergunta 27:** Os botões, menus e elementos visuais do aplicativo possuem um padrão consistente em todas as telas.
- **Pergunta 28:** O aplicativo ajuda a evitar erros, fornecendo avisos antes de ações importantes ou perigosas (ex.: confirmação antes de excluir algo).
- **Pergunta 29:** Os ícones, rótulos e funcionalidades no aplicativo são facilmente reconhecíveis sem a necessidade de você memorizar ações anteriores.
- **Pergunta 30:** Você consegue realizar suas tarefas rapidamente e com facilidade no aplicativo, mesmo se for a primeira vez que o utiliza.
- **Pergunta 31:** O design do aplicativo é simples, sem informações desnecessárias que possam causar confusão.
- **Pergunta 32:** Se você cometeu um erro no aplicativo, ele forneceu uma mensagem clara para ajudá-lo a corrigir o problema.
- **Pergunta 33:** O aplicativo oferece instruções ou uma seção de ajuda clara caso você precise de orientação para realizar suas tarefas.

5.4 Limitações e Melhorias Futuras

Embora o projeto tenha sido desenvolvido com foco na escalabilidade e na acessibilidade, algumas limitações são reconhecidas:

- **Dependência da Coordenação:** O sucesso do sistema está diretamente relacionado à quantidade e qualidade das vagas cadastradas pela coordenação de estágios, que irá realizar o cadastro das vagas no sistema. Uma base de dados inicial limitada pode impactar a eficácia das recomendações personalizadas.

- **Testes Restritos:** Os testes foram aplicados a alunos e coordenadores do IFB campus Brasília, o que pode limitar a generalização dos resultados para outras instituições ou contextos.
- **Acessibilidade:** Embora o sistema tenha sido projetado para ser responsivo, ainda não foram realizados testes extensivos em dispositivos móveis ou com usuários com deficiência.
- **Infraestrutura Tecnológica:** Algumas funcionalidades avançadas, como notificações em tempo real, não foram implementadas devido a limitações de infraestrutura e tempo de desenvolvimento.

Para trabalhos futuros, sugere-se as seguintes melhorias:

- Expansão do sistema para todos os campi do IFB, validando sua escalabilidade e adaptabilidade a diferentes cursos dos abordados no projeto.
- Integração de tecnologias como *Web Scrapping* para preencher as vagas de estágio automaticamente e evitar a dependência com a coordenação de estágios
- Realização de testes com maior diversidade de dispositivos e grupos de usuários, garantindo acessibilidade universal e compatibilidade tecnológica.
- Desenvolvimento de funcionalidades adicionais, como integração com redes sociais profissionais e métricas analíticas para empresas parceiras.

6

CONSIDERAÇÕES FINAIS

O projeto "IFB + Estágios" foi desenvolvido com o objetivo de otimizar a gestão de estágios e ampliar o acesso dos estudantes do IFB, campus Brasília, a oportunidades de vagas de estágio. O sistema foi concebido para atender à necessidade de uma solução mais eficiente para conectar alunos a empresas, oferecendo um ambiente que facilite o acesso a experiências práticas e complementares à formação teórica.

O desenvolvimento do sistema envolveu diversos desafios, especialmente na etapa inicial de levantamento de requisitos, que exigiu um trabalho detalhado de análise das necessidades dos usuários e das especificidades do ambiente institucional.

Apesar do avanço representado pelo projeto, algumas limitações foram identificadas. A implantação completa do sistema depende de recursos institucionais e do apoio contínuo para sua manutenção e expansão. Além disso, a integração com o banco de dados e a capacitação dos usuários para operar o sistema são pontos que demandam planejamento adicional. A escalabilidade do sistema também representa um desafio, especialmente no caso de sua adaptação para outros campi ou instituições de ensino que possuam contextos específicos.

O impacto do "IFB + Estágios" se estende além da solução imediata para a gestão de estágios. O sistema tem o potencial de beneficiar empresas parceiras, facilitando o recrutamento de estudantes qualificados e fortalecendo a conexão entre a instituição de ensino e o mercado de trabalho. Ademais, o sistema pode servir como modelo para outras unidades do IFB ou mesmo para outras instituições, mediante adaptações adequadas, promovendo uma melhoria no acesso às informações das vagas de estágios para alunos em diferentes cenários.

Em síntese, o "IFB + Estágios" representa uma contribuição significativa para a integração entre o ensino e a prática profissional. O projeto reflete um esforço em alinhar tecnologia, inovação e as necessidades educacionais, oferecendo uma ferramenta que pode ser ampliada e replicada para atender a diferentes contextos. Com o apoio necessário, o sistema pode atingir seu pleno potencial, tornando-se um componente essencial na gestão de estágios e na promoção do acesso à informação em um ambiente educacional cada vez mais competitivo.

Referências

- ACCOMAZZO, A. et al. **Fullstack React: the complete guide to reactjs and friends**. [S.l.]: Fullstack. io, 2017.
- ALVARENGA, B. O Acesso à Informação nos Portais das Universidades Federais Brasileiras. , [S.l.], p.5, 2013.
- BENTO, P. d. F. C. **Adaptabilidade de carreira empregabilidade percebida e auto-eficácia na transição para o trabalho: estudo comparativo entre estudantes universitários com e sem experiência de trabalho**. 2013. Dissertação (Mestrado em Ciência da Computação) — Universidade do Algarve (Portugal).
- BLISCHAK, J. D.; DAVENPORT, E. R.; WILSON, G. A quick introduction to version control with Git and GitHub. **PLoS computational biology**, [S.l.], v.12, n.1, p.e1004668, 2016.
- Brasil. **Lei de Diretrizes e Bases da Educação Nacional, Lei nº9.394, de 20 de dezembro de 1996**. Acesso em: 20 dez. 2024, <http://www.planalto.gov.br/ccivil03/leis/19394.htm>.
- BRASIL. **Lei nº11.788, de 25 de setembro de 2008**. Acesso em: 10 de Nov. de 2023., Diário Oficial da União.
- BROOKE, J. **System Usability Scale (SUS): a quick and dirty usability scale**. Disponível em: <https://digital.ahrq.gov/sites/default/files/docs/survey/systemusabilityscale>
- BROWN, T. **Design Thinking – Edição Comemorativa 10 anos**. Rio de Janeiro: Editora Alta Books, 2020. E-book, p.243.
- CÓL, L. B. de. **Sistema para Acompanhamento de Realização de Estágios Curriculares**. 2016. Dissertação (Mestrado em Ciência da Computação) — Universidade Tecnológica Federal do Paraná, Pato Branco. Trabalho de Conclusão de Curso - Tecnologia em Análise e Desenvolvimento de Sistemas.
- DEWEY, J. **Democracy and education: an introduction to the philosophy of education**. [S.l.]: Courier Corporation, 2012.
- EATON, E. A lightweight approach to academic research group management using online tools: spend more time on research and less on management. In: AAAI CONFERENCE ON ARTIFICIAL INTELLIGENCE. **Proceedings...** [S.l.: s.n.], 2019. v.33, n.01, p.9644–9647.
- FLANAGAN, D. **JavaScript: o guia definitivo**. E-book.ed. [S.l.]: Grupo A, 2013. Disponível em: <https://integrada.minhabiblioteca.com.br//books/9788565837484/>. Acesso em: 28 nov. 2023.
- FLESCH, J. A. **Portal de Vagas - implementação de uma ferramenta para divulgação de bolsas e estágios a alunos**. 2018. Dissertação (Mestrado em Ciência da Computação) — Universidade Federal do Rio Grande do Sul, Porto Alegre. Monografia de Bacharelado em Ciência da Computação.
- FRANKS, P. C.; OLIVER, G. C. Experiential learning and international collaboration opportunities: virtual internships. **Library Review**, [S.l.], v.61, n.4, p.272–285, 2012.

- GAULT, J.; LEACH, E.; DUEY, M. Effects of business internships on job marketability: the employers' perspective. **Education + Training**, [S.l.], v.52, n.1, p.76–88, 2010. Acesso em: 10 de Nov. de 2023.
- GLASSON, G. The Unified Modeling Language user guide. **Linux Journal**, [S.l.], v.1999, n.65es, p.30–es, 1999.
- HOQUE, S. **Full-Stack React Projects: learn mern stack development by building modern web apps using mongodb, express, react, and node. js**. [S.l.]: Packt Publishing Ltd, 2020.
- INC., G. **GitHub Documentation**. [S.l.: s.n.], 2023.
- Instituto Federal de Brasília. **Regulamento de Estágio Supervisionado dos Cursos Técnicos de Nível Médio Profissional e de Graduação do Instituto Federal de Brasília**. 2016.
- KOLB, D. A. **Experiential learning: experience as the source of learning and development**. [S.l.]: FT press, 2014.
- LEITE, M. S. R.; GASPAROTTO, A. M. S. ANÁLISE SWOT E SUAS FUNCIONALIDADES: o autoconhecimento da empresa e sua importância. **Revista interface tecnológica**, [S.l.], v.15, n.2, p.184–195, 2018.
- MACHINESKI, R. d. S.; MACHADO, A. C. T. A.; SILVA, R. T. M. d. A importância do estágio e do programa de iniciação científica na formação profissional e científica. **Enciclopédia Biosfera**, [S.l.], v.7, n.13, 2011.
- MARTINS, B. V. et al. Qualificação Profissional, Mercado de Trabalho e Mobilidade Social: cursos superiores de tecnologia. **Sociedade, Contabilidade e Gestão**, [S.l.], v.12, n.2, p.45, 2017.
- MDN Web Docs. **CSS**. Acesso em: 26 de Nov. de 2023.
- MDN Web Docs. **JavaScript**. Acesso em: 26 de Nov. de 2023.
- MDN Web Docs. **HTML básico**. Acesso em: 26 de Nov. de 2023.
- META. **React Documentation - A JavaScript Library for Building User Interfaces**. 2024.
- Microsoft. **O que é o SQL Server?** Acesso em: 26 de Nov. de 2023.
- Microsoft. **O que é o Azure DevOps?** Acessado em: 24 dez. 2024.
- MULESOFT. **API-led Connectivity Overview**. Accessed on: 20 Dec. 2024.
- NIELSEN, J. **10 Usability Heuristics for User Interface Design**. Disponível em: <https://www.nngroup.com/articles/ten-usability-heuristics/>.
- OLIVEIRA, F. W. B. de. **Sistema para Gerenciamento de Estágios Universitários do curso de Sistemas de Informação da UFPI (SisGEU)**. 2015. Dissertação (Mestrado em Ciência da Computação) — Universidade Federal do Piauí, Picos. Monografia de Bacharelado em Sistemas de Informação.

- PRESSMAN, R. S.; MAXIM, B. R. **Engenharia de Software**. 9.ed. Porto Alegre: AMGH, 2021. E-book. p.i. ISBN 9786558040118. Disponível em: <https://integrada.minhabiblioteca.com.br/reader/books/9786558040118/>. Acesso em: 29 dez. 2024.
- ROLINO, A. P. M. **Plataforma de gestão e disseminação de estágios profissionais**. 2013. Dissertação (Mestrado em Ciência da Computação) — Faculdade de Engenharia da Universidade do Porto, Porto. Dissertação de Mestrado Integrado em Engenharia Eletrotécnica e de Computadores.
- STAIANO, F. **Designing and Prototyping Interfaces with Figma: learn essential ux/ui design principles by creating interactive prototypes for mobile, tablet, and desktop**. [S.l.]: Packt Publishing Ltd, 2022.
- STOPA, G. R.; RACHID, C. L. SCRUM: metodologia ágil como ferramenta de gerenciamento de projetos. **CES Revista**, [S.l.], v.33, n.1, 2019.
- SUTHERLAND, J. **SCRUM: a arte de fazer o dobro de trabalho na metade do tempo**. [S.l.]: Leya, 2014.
- TAVARES, J. M. A.; BARROSO, M. V. E. **Um Sistema de Inscrição e Monitoramento para o Programa de Estágio Remunerado da Prefeitura de Maricá**. 2022. Dissertação (Mestrado em Ciência da Computação) — Universidade Federal Fluminense, Niterói. Trabalho de Conclusão de Curso - Ciência da Computação.
- TORVALDS, L. **Git Documentation**. [S.l.: s.n.], 2005.

Apêndice

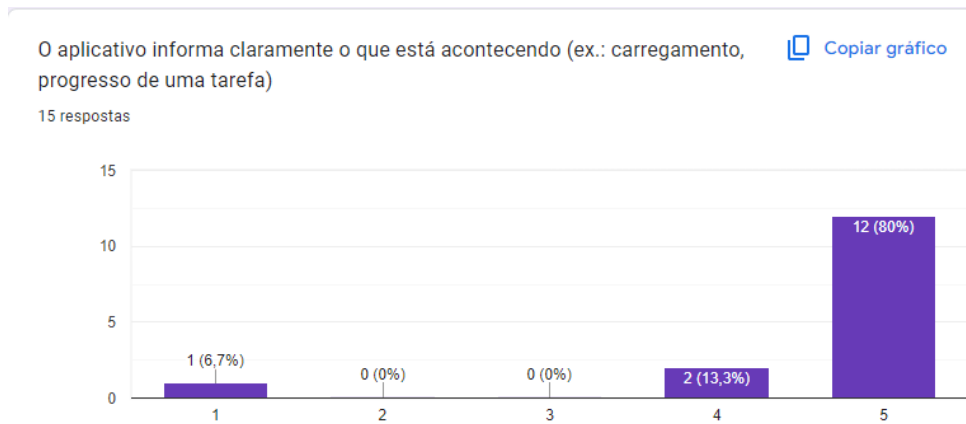


Figura 14 Pergunta 1 - O aplicativo informa claramente o que está acontecendo (ex.: carregamento, progresso de uma tarefa).

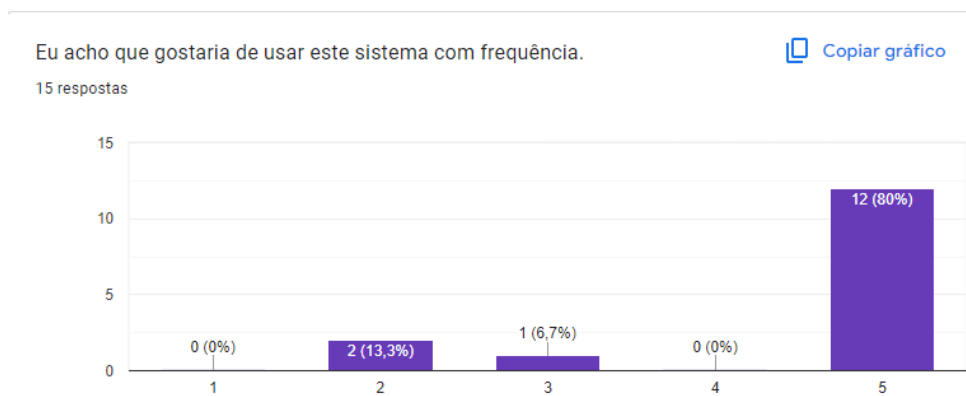


Figura 15 Pergunta 2 - Eu acho que gostaria de usar este sistema com frequência.

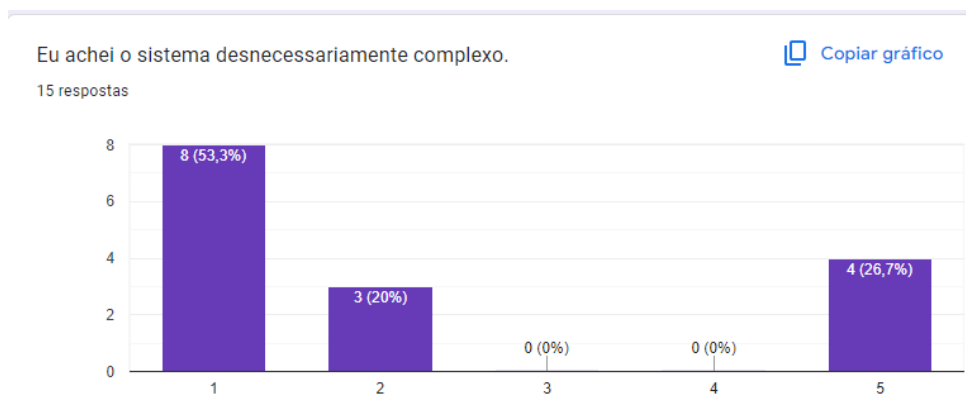


Figura 16 Pergunta 3 - Eu achei o sistema desnecessariamente complexo.

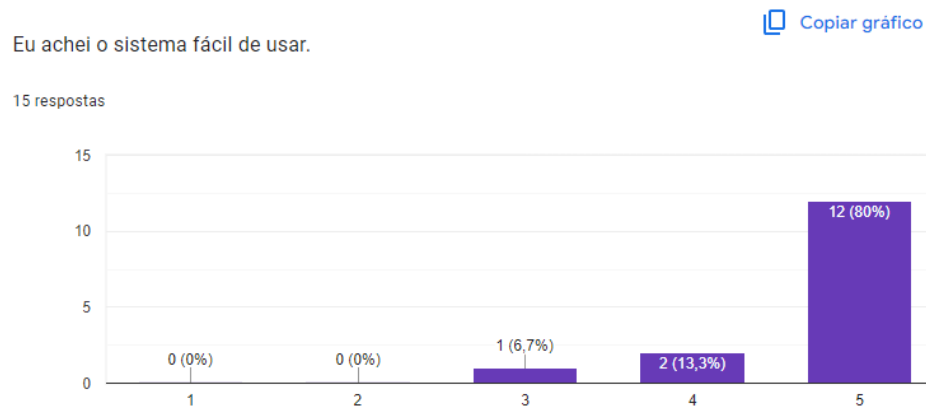


Figura 17 Pergunta 4 - Eu achei o sistema fácil de usar.

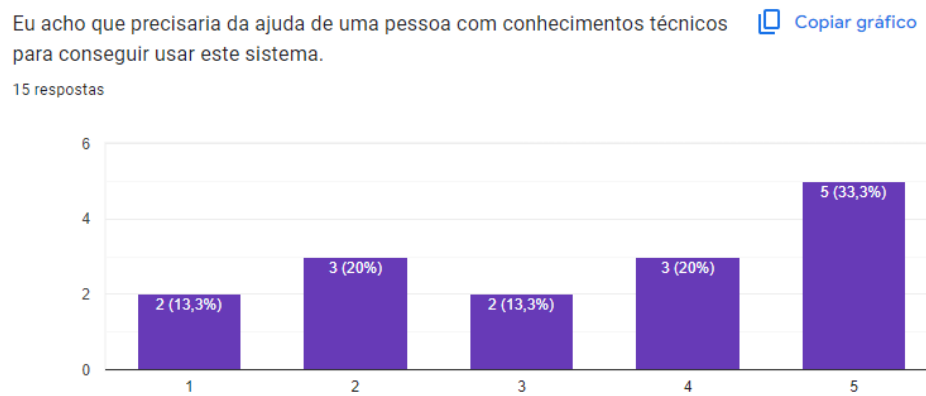


Figura 18 Pergunta 5 - Eu acho que precisaria da ajuda de uma pessoa com conhecimentos técnicos para conseguir usar este sistema.

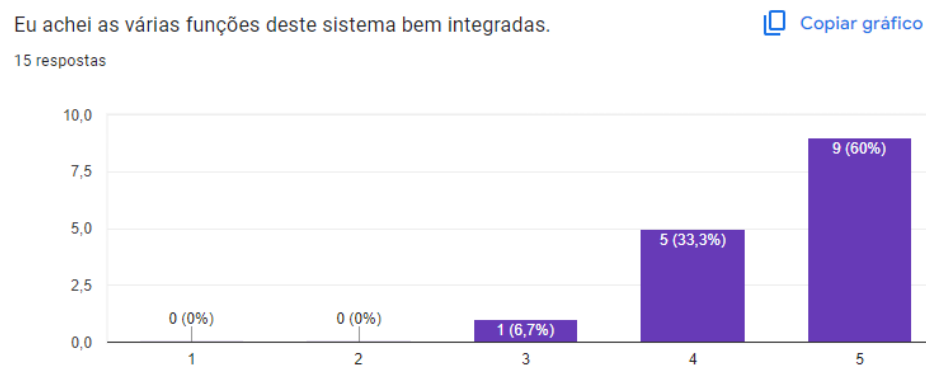


Figura 19 Pergunta 6 - Eu achei as várias funções deste sistema bem integradas.

Eu achei que havia muita inconsistência neste sistema.

[Copiar gráfico](#)

15 respostas

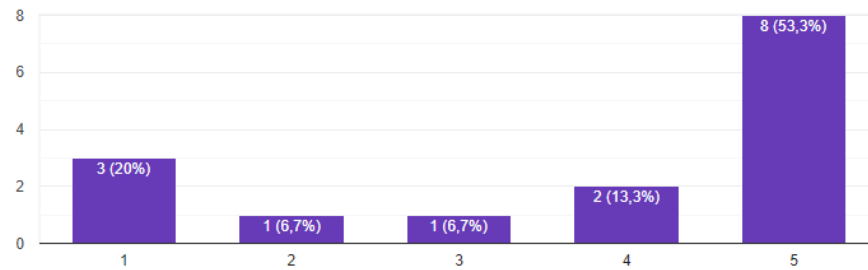


Figura 20 Pergunta 7 - Eu achei que havia muita inconsistência neste sistema.

Eu imagino que a maioria das pessoas aprenderia a usar este sistema rapidamente.

[Copiar gráfico](#)

15 respostas

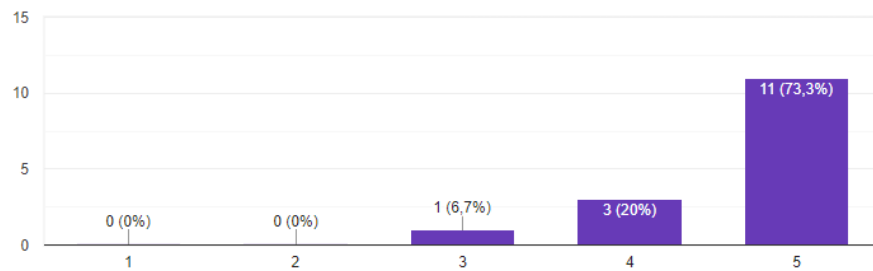


Figura 21 Pergunta 8 - Eu imagino que a maioria das pessoas aprenderia a usar este sistema rapidamente.

Eu achei o sistema muito confuso.

[Copiar gráfico](#)

15 respostas

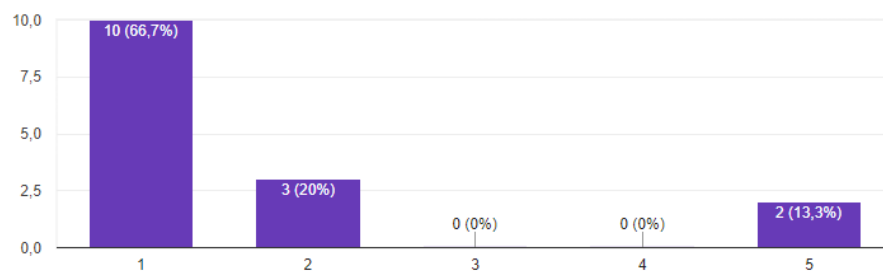


Figura 22 Pergunta 9 - Eu achei o sistema muito confuso.



Figura 23 Pergunta 10 - Eu me senti muito confiante usando o sistema.

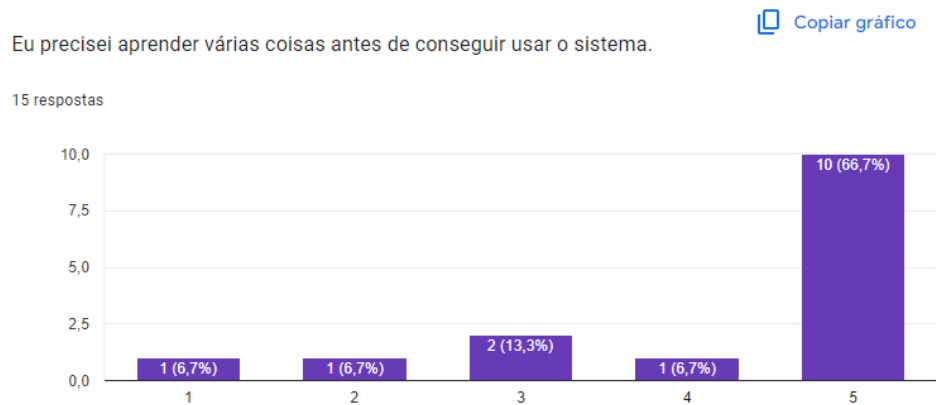


Figura 24 Pergunta 11 - Eu precisei aprender várias coisas antes de conseguir usar o sistema.



Figura 25 Pergunta 12 - As palavras, ícones e instruções usadas no aplicativo são fáceis de entender e correspondem ao seu conhecimento do mundo real.

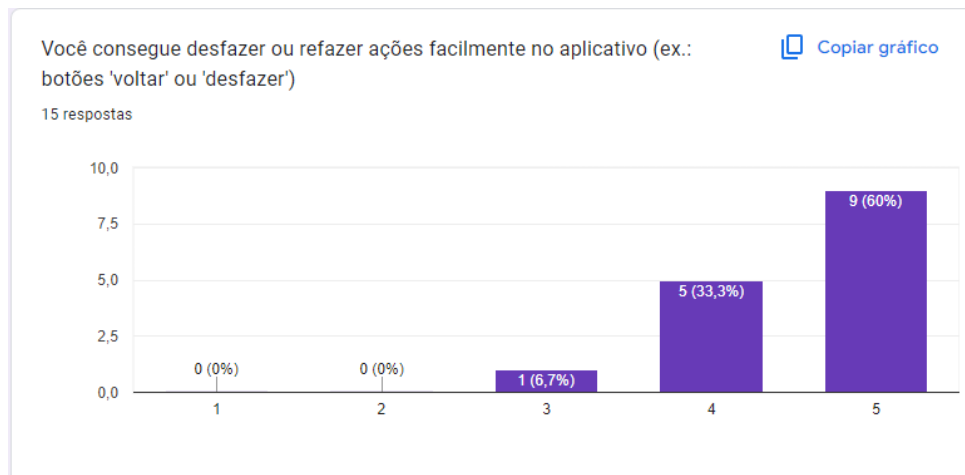


Figura 26 Pergunta 13 - Você consegue desfazer ou refazer ações facilmente no aplicativo (ex.: botões "voltar" ou "desfazer").

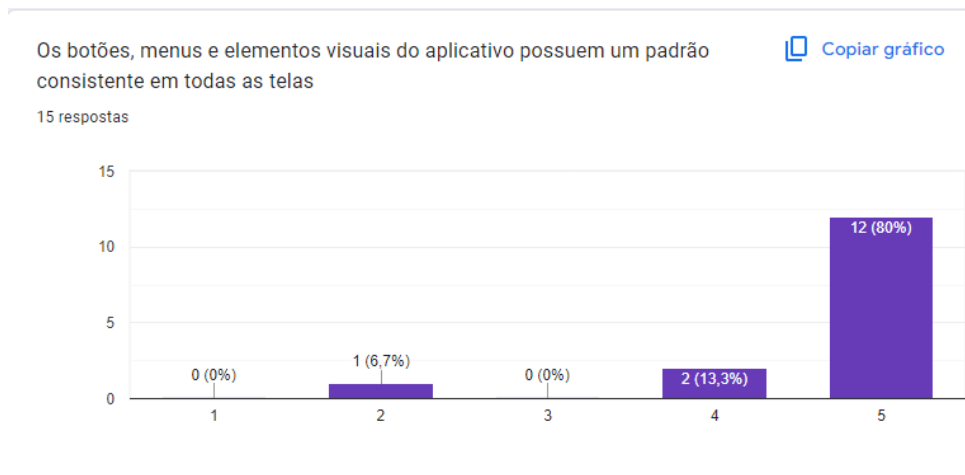


Figura 27 Pergunta 14 - Os botões, menus e elementos visuais do aplicativo possuem um padrão consistente em todas as telas.

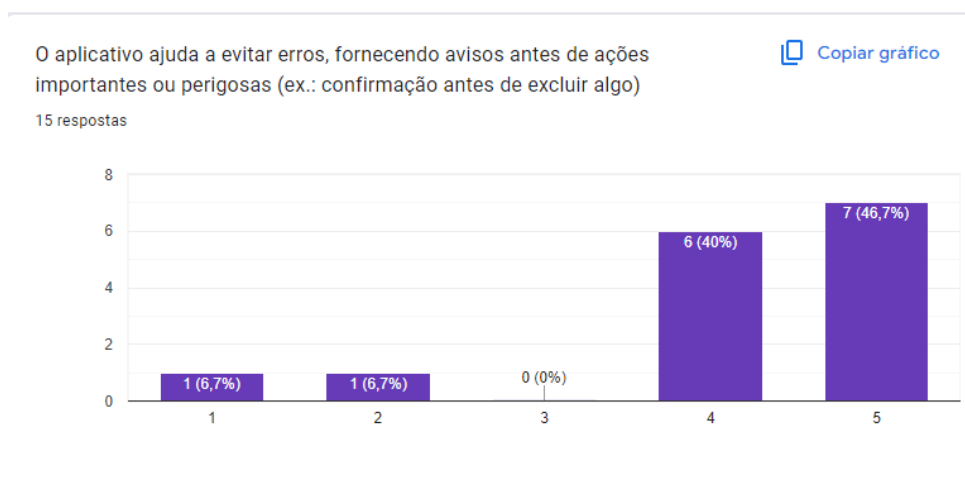


Figura 28 Pergunta 15 - O aplicativo ajuda a evitar erros, fornecendo avisos antes de ações importantes ou perigosas (ex.: confirmação antes de excluir algo).

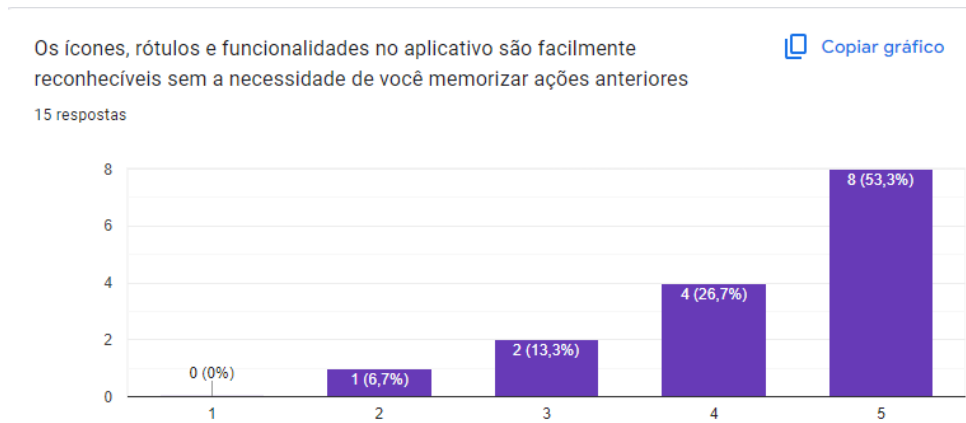


Figura 29 Pergunta 16 - Os ícones, rótulos e funcionalidades no aplicativo são facilmente reconhecíveis sem a necessidade de você memorizar ações anteriores.

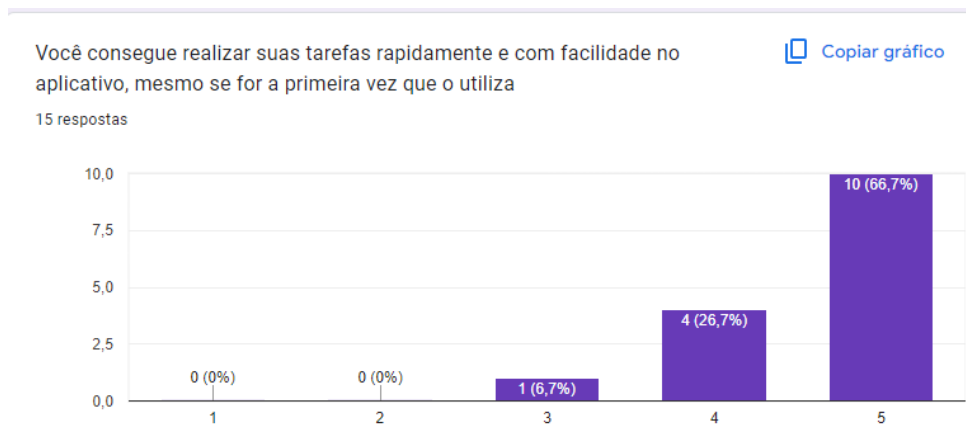


Figura 30 Pergunta 17 - Você consegue realizar suas tarefas rapidamente e com facilidade no aplicativo, mesmo se for a primeira vez que o utiliza.

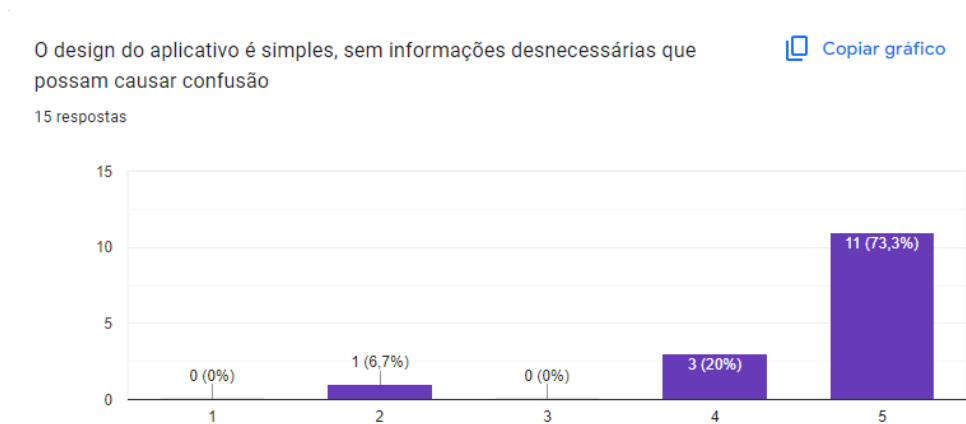


Figura 31 Pergunta 18 - O design do aplicativo é simples, sem informações desnecessárias que possam causar confusão.

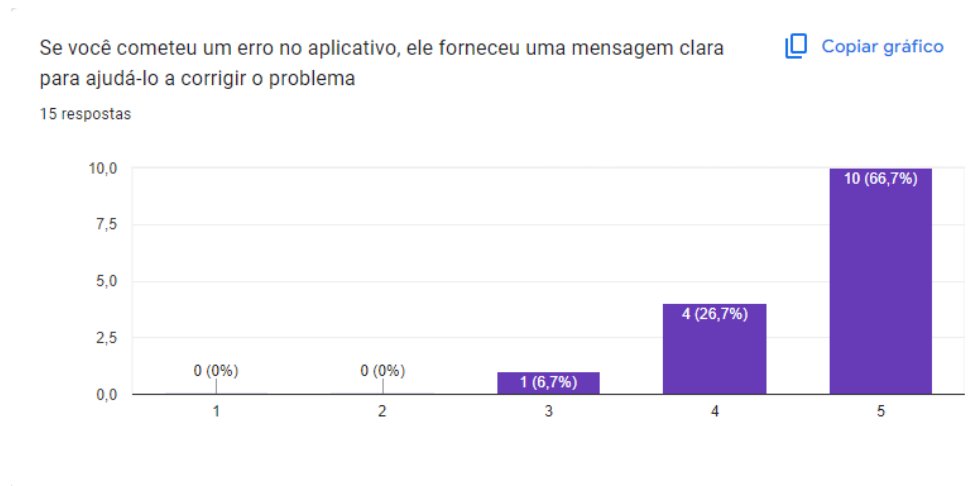


Figura 32 Pergunta 19 - Se você cometeu um erro no aplicativo, ele forneceu uma mensagem clara para ajudá-lo a corrigir o problema.

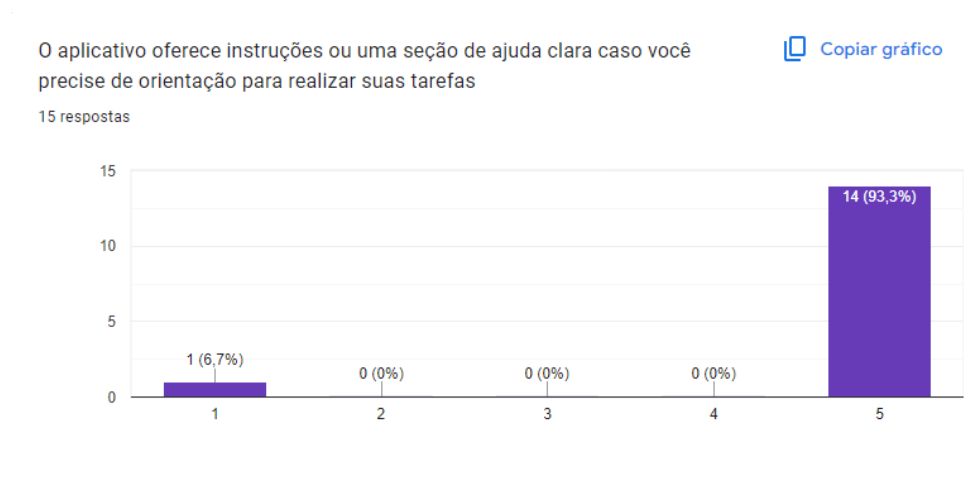


Figura 33 Pergunta 20 - O aplicativo oferece instruções ou uma seção de ajuda clara caso você precise de orientação para realizar suas tarefas.