

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

JOÃO PEDRO SILVEIRA GONÇALVES

CONSTRUINDO UM QUADRO DE VAGAS PARA O INSTITUTO DE  
COMPUTAÇÃO

RIO DE JANEIRO  
2025

JOÃO PEDRO SILVEIRA GONÇALVES

CONSTRUINDO UM QUADRO DE VAGAS PARA O INSTITUTO DE  
COMPUTAÇÃO

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

Orientadora: Profa. Silvana Rossetto

RIO DE JANEIRO

2025

## CIP - Catalogação na Publicação

G635c Gonçalves, João Pedro Silveira  
Construindo um quadro de vagas para o Instituto  
de Computação / João Pedro Silveira Gonçalves. -- Rio  
de Janeiro, 2025.  
59 f.

Orientador: Silvana Rossetto.  
Trabalho de conclusão de curso (graduação) -  
Universidade Federal do Rio de Janeiro, Instituto  
de Computação, Bacharel em Ciência da Computação,  
2025.

1. Recrutamento. 2. Engenharia de Software. 3.  
Metodologias Ágeis. 4. Computação em Nuvem. I.  
Rossetto, Silvana, orient. II. Título.

JOÃO PEDRO SILVEIRA GONÇALVES

CONSTRUINDO UM QUADRO DE VAGAS PARA O INSTITUTO DE  
COMPUTAÇÃO

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

Aprovado em 20 de Agosto de 2025

BANCA EXAMINADORA:

Documento assinado digitalmente  
 SILVANA ROSSETTO  
Data: 27/08/2025 08:55:42-0300  
Verifique em <https://validar.iti.gov.br>

---

Silvana Rossetto  
D.Sc. (IC/UFRJ)  
Documento assinado digitalmente  
 ADRIANA SANTAROSA VIVACQUA  
Data: 27/08/2025 09:03:36-0300  
Verifique em <https://validar.iti.gov.br>

---

Adriana Vivacqua  
D.Sc. (IC/UFRJ)  
Documento assinado digitalmente  
 MARIA LUIZA MACHADO CAMPOS  
Data: 27/08/2025 14:50:46-0300  
Verifique em <https://validar.iti.gov.br>

---

Maria Luiza Campos  
Ph.D. (IC/UFRJ)

À Christiani, Alexandre, Jaumir, Terezinha, Miguel e Doki.

## **AGRADECIMENTOS**

Agradeço à minha orientadora, a Profa. Silvana Rossetto por todo o suporte que me deu, carinho e paciência com que me tratou ao longo da execução deste trabalho e 5 anos de graduação.

Agradeço à Amazon Web Services (AWS) por oferecer uma faixa de uso gratuita que possibilitou o desenvolvimento deste trabalho e hospedagem da solução criada.

*“Make something people want.”*

**Paul Graham**  
Cofundador, YCombinator

## RESUMO

A participação em atividades extracurriculares e de extensão é obrigatória para a graduação no Instituto de Computação da Universidade Federal do Rio de Janeiro, porém tanto estudantes quanto equipes ofertantes enfrentam dificuldades para divulgar e descobrir oportunidades como iniciações científicas e ações de extensão. Plataformas comerciais como LinkedIn e Indeed apresentam custos elevados e não apresentam funcionalidades adequadas ao público acadêmico. Já os métodos internos, como e-mails, grupos de mensagens e murais, resultam em informações dispersas e baixa taxa de leitura. Visando suprir essas lacunas, desenvolveu-se um portal de vagas para otimizar a divulgação e a descoberta de vagas em atividades extracurriculares por estudantes do Instituto. O projeto foi conduzido utilizando metodologias ágeis inspiradas no Scrum, enquanto a implementação foi orientada pelos requisitos da solução e pelo contexto universitário em que será utilizada, o que impôs a necessidade de baixos custos de hospedagem e manutenção. Em função disso, foram adotadas técnicas e tecnologias contemporâneas de desenvolvimento web, como o modelo de *Single Page Applications (SPA)* e o uso de serviços *serverless* de computação em nuvem, especificamente por meio da Amazon Web Services. Ao final, potenciais usuários avaliaram que a solução resolveu suas principais dores e sugeriram que ela poderia ser utilizada até mesmo por outras unidades acadêmicas dentro da própria Universidade.

**Palavras-chave:** Recrutamento; Engenharia de Software; Metodologias Ágeis; Computação em Nuvem; Single Page Applications; Arquitetura Publish Subscriber;

## ABSTRACT

Participation in extracurricular and extension activities is mandatory for graduation at the Institute of Computing of the Federal University of Rio de Janeiro. However, both students and offering teams face challenges in publicizing and discovering opportunities such as research internships and extension projects. Commercial platforms like LinkedIn and Indeed are costly and do not provide features tailored to the academic audience. Internal methods—such as emails, messaging groups, and bulletin boards—result in dispersed information and low readership rates. To address these gaps, a job portal was developed to optimize the dissemination and discovery of extracurricular opportunities for students at the Institute. The project was carried out using agile methodologies inspired by Scrum, while the implementation was guided by the solution's requirements and the university context in which it would be used, which required low hosting and maintenance costs. As a result, contemporary web development techniques and technologies were adopted, such as the Single Page Application (SPA) model and the use of serverless cloud computing services, specifically through Amazon Web Services. In the end, potential users evaluated that the solution addressed their main pain points and suggested that it could even be used by other institutions within the university itself.

**Keywords:** Recruitment; Software Engineering; Agile Methodologies; Cloud Computing; Single Page Applications (SPA); Publish-Subscribe Architecture;

## **LISTA DE ABREVIATURAS E SIGLAS**

AWS	Amazon Web Services
URL	Uniform Resource Locator
SQS	Amazon Simple Notification Service
SES	Amazon Simple Email Service
S3	Amazon Simple Storage Service
IC	Instituto de Computação
UFRJ	Universidade Federal do Rio de Janeiro
MEC	Ministério da Educação
API	Application Programming Interface
CSS	Cascading Style Sheets

## SUMÁRIO

<b>1</b>	<b>INTRODUÇÃO . . . . .</b>	<b>11</b>
1.1	OBJETIVOS DO TRABALHO . . . . .	12
1.2	METODOLOGIA DE PESQUISA . . . . .	12
1.3	PRÓXIMOS CAPÍTULOS . . . . .	13
1.4	USO DE INTELIGÊNCIA ARTIFICIAL COMO APOIO . . . . .	13
<b>2</b>	<b>TRABALHOS RELACIONADOS . . . . .</b>	<b>15</b>
2.1	SOLUÇÕES COMERCIAIS . . . . .	15
2.1.1	LinkedIn . . . . .	15
2.1.2	Indeed . . . . .	16
2.1.3	Conclusões . . . . .	16
2.2	SOLUÇÕES NÃO-COMERCIAIS INTERNAS . . . . .	16
2.2.1	E-mail Institucional . . . . .	16
2.2.2	Grupos e Canais de Mensagens Instantâneas (WhatsApp, Telegram) . . . . .	17
2.2.3	Cartazes Físicos e Murais . . . . .	17
2.2.4	Sites ou Páginas Não-Padronizadas . . . . .	17
2.2.5	Conclusões . . . . .	18
2.3	DIFERENCIAL DA SOLUÇÃO PROPOSTA . . . . .	18
<b>3</b>	<b>ESPECIFICAÇÃO E PLANEJAMENTO . . . . .</b>	<b>20</b>
3.1	PESQUISA COM USUÁRIOS . . . . .	20
3.1.1	Questionário A: Candidatos a Vagas . . . . .	20
3.1.2	Questionário B: Anunciantes de Vagas . . . . .	21
3.2	ANÁLISE DOS RESULTADOS . . . . .	21
3.3	METODOLOGIA DE DESENVOLVIMENTO SCRUM . . . . .	23
3.4	APLICAÇÃO DA METODOLOGIA SCRUM NO PROJETO . . . . .	24
3.4.1	Construção de <i>Product Backlog</i> . . . . .	25
3.4.2	Backlog da primeira <i>sprint</i> e escopo do projeto . . . . .	25
<b>4</b>	<b>DESENVOLVIMENTO . . . . .</b>	<b>27</b>
4.1	ARQUITETURA LÓGICA DA SOLUÇÃO . . . . .	27
4.2	ARQUITETURA FÍSICA DA SOLUÇÃO . . . . .	28
4.2.1	Cliente: camada de apresentação . . . . .	28
4.2.2	Servidor: camadas de aplicação e dados . . . . .	29
4.2.3	Comunicação assíncrona e eventos . . . . .	29

4.2.4	Resumo das tecnologias utilizadas no sistema . . . . .	32
4.3	IMPLEMENTAÇÃO DAS FUNCIONALIDADES . . . . .	34
4.3.1	Visualização de Vagas Disponíveis . . . . .	34
4.3.2	Filtragem de Vagas . . . . .	36
4.3.3	Cadastro de Vagas . . . . .	39
5	<b>AVALIAÇÕES . . . . .</b>	43
5.1	METODOLOGIA DE AVALIAÇÃO . . . . .	43
5.2	PERFIL DOS PARTICIPANTES . . . . .	44
5.3	RESULTADOS QUALITATIVOS . . . . .	44
5.3.1	Pontos Fortes Identificados . . . . .	44
5.3.2	Dificuldades e Pontos de Melhoria . . . . .	45
5.3.2.1	Estudantes . . . . .	45
5.3.2.2	Anunciantes . . . . .	45
5.3.3	Satisfação Geral . . . . .	45
5.4	CONSIDERAÇÕES FINAIS . . . . .	46
6	<b>CONCLUSÃO . . . . .</b>	47
	<b>REFERÊNCIAS . . . . .</b>	49
	<b>APÊNDICE A – QUESTIONÁRIOS APLICADOS . . . . .</b>	51
	<b>APÊNDICE B – CAPTURAS DE TELA DA SOLUÇÃO DESEN- VOLVIDA . . . . .</b>	56
B.1	VISUALIZAÇÃO E FILTRAGEM DE VAGAS . . . . .	56
B.2	PAGINAÇÃO DE RESULTADOS . . . . .	58
B.3	CADASTRO DE NOVAS VAGAS . . . . .	59

## 1 INTRODUÇÃO

Por determinação do Ministério da Educação, os currículos de cursos de graduação no Brasil devem apresentar como componentes curriculares: disciplinas obrigatórias, atividades de extensão, atividades complementares, disciplinas optativas, disciplinas de escolha restrita, disciplinas de livre escolha, estágio (quando previsto), trabalho de conclusão de curso ou equivalente (quando previsto). Dessa forma, todo aluno precisa realizar uma atividade curricular de extensão ou complementar de escolha livre ou condicionada em algum momento da sua vida universitária. Parte dessas exigências está respaldada pela Resolução CNE/CES nº 7 de 18 de dezembro de 2018 (Conselho Nacional de Educação, 2018), que estabelece as Diretrizes para a Extensão na Educação Superior Brasileira, determinando que, nos cursos de graduação, pelo menos 10% da carga horária total deve ser dedicada a atividades de extensão universitária integradas ao currículo.

Um exemplo é o curso de graduação associado a este trabalho de conclusão de curso, o Bacharelado em Ciência da Computação no Instituto de Computação (IC) da Universidade Federal do Rio de Janeiro. Sua estrutura curricular estipula que os alunos devem cumprir no mínimo: 90 horas de trabalho em atividades extracurriculares, como iniciações científicas e estágios; e 320 horas em extensão<sup>1</sup>.

No contexto do IC, a participação nesse tipo de atividade envolve, por parte das equipes responsáveis, a divulgação das vagas; e, por parte dos estudantes, a busca ativa pelas oportunidades oferecidas. No entanto, como a vivência do Autor deste texto mostra e sua pesquisa evidencia, os processos de divulgação e descoberta de vagas são marcados por dificuldades e insatisfações.

Essas dificuldades manifestam-se tanto do ponto de vista de quem anuncia quanto de quem procura vagas. Do lado dos anunciantes, reclamações comuns são que os meios mais utilizados internamente para divulgação — como cartazes físicos, postagens em listas de transmissão no sistema de e-mail institucional ou em grupos de mensagens instantâneas — apresentam alcance restrito, baixa taxa de leitura e contribuem para a dispersão da informação. Dessa forma, são insuficientes para que a informação seja difundida efetivamente pelo corpo discente. Ademais, soluções comerciais como LinkedIn ou Indeed cobram altos e proibitivos valores por anúncios, além de não oferecerem funcionalidades que facilitem ou permitam a divulgação de atividades acadêmicas extracurriculares.

Do lado dos estudantes, as dificuldades assumem outra forma: uma reclamação comum é que, para descobrir vagas, precisam protagonizar uma busca ativa, isto é, perguntar a professores e colegas de curso se conhecem alguma oportunidade do tipo que buscam. Por

<sup>1</sup> Universidade Federal do Rio de Janeiro (s.d.) define atividades de extensão como ações que promovem a interação entre a instituição e a sociedade, como projetos e grupos de pesquisa com aplicação social, cursos abertos à comunidade, eventos culturais e científicos, e serviços oferecidos à população.

vezes, o autor, que foi membro da Comissão de Orientação e Acompanhamento Acadêmico do Instituto, vivenciou cenas de estudantes que precisavam encontrar vagas em atividades acadêmicas que oferecessem bolsas para ajudá-los a se manter na Universidade, mas não sabiam como ou não conseguiam encontrar essa informação.

Em conjunto, essas dificuldades podem afetar negativamente toda a comunidade universitária, acarretando:

- Para estudantes: o atraso ou até a impossibilidade na conclusão do curso, pois tais participações são obrigatórias para a graduação;
- Para os grupos de atividade extracurricular: a impossibilidade ou dificuldade de continuidade de suas atividades, dado o número reduzido de participantes;
- Para a universidade: perda de produtividade, aumento do tempo médio de graduação e até elevação da evasão escolar.

Dessa forma, considerando que é obrigatório que estudantes participem dessas ações extracurriculares para cumprirem todos os requisitos de formação e que é um objetivo comum evitar a evasão e a retenção universitária, torna-se clara a importância de se atuar na resolução dos problemas que afetem a divulgação e descobertas dessas ações.

## 1.1 OBJETIVOS DO TRABALHO

Dessa forma, este trabalho visa desenvolver uma aplicação Web para facilitar o compartilhamento e a descoberta de vagas em atividades destinadas a estudantes universitários no contexto do Instituto de Computação da Universidade Federal do Rio de Janeiro. São exemplos dessas atividades cujas vagas serão divulgadas na plataforma: iniciações científicas, estágios, voluntariado, projetos de extensão. Objetiva-se que, quando em funcionamento, a solução resolva os problemas listados anteriormente, contribuindo para a redução da evasão e retenção universitária e promovendo o crescimento acadêmico-profissional desses alunos.

## 1.2 METODOLOGIA DE PESQUISA

Este trabalho é de natureza aplicada, pois envolve o desenvolvimento de um sistema para solucionar um problema específico identificado no contexto do Instituto de Computação. Além disso, documenta de forma sistemática o processo de concepção, implementação e avaliação de uma prova de conceito da solução proposta, analisando os desafios enfrentados e as decisões tomadas ao longo do percurso. A abordagem adotada é qualitativa, por basear-se na análise detalhada dos processos e nas percepções coletadas junto aos usuários envolvidos.

O desenvolvimento do sistema foi conduzido de forma iterativa e incremental, possibilitando a incorporação contínua de melhorias. Para a organização das atividades, aplicaram-se práticas de desenvolvimento ágil, como o planejamento das tarefas em ciclos curtos (*sprints*) e a revisão periódica das prioridades de implementação, permitindo maior adaptação às necessidades identificadas no decorrer do projeto.

O fluxo de trabalho seguiu as seguintes etapas:

1. **Levantamento de requisitos e definição do escopo:** nesta etapa, foram aplicados formulários a estudantes e anunciantes para compreender suas dificuldades e identificar oportunidades de otimização nos processos de descoberta e divulgação de vagas. Como resultado, elaborou-se uma lista de funcionalidades a serem desenvolvidas em *sprints*, ordenadas de acordo com as prioridades apontadas pelos participantes.
2. **Desenvolvimento iterativo:** implementação progressiva das funcionalidades priorizadas.
3. **Testes com usuários:** execução de testes de uso e planejamento de melhorias com base nos *feedbacks* obtidos.

O detalhamento de cada uma dessas etapas será apresentado nos capítulos seguintes, de modo a evidenciar como a metodologia adotada contribuiu para o alcance dos objetivos estabelecidos.

### 1.3 PRÓXIMOS CAPÍTULOS

O restante do texto está organizado para mostrar como a solução foi construída. O capítulo 2 apresenta o estudo sobre trabalhos e projetos relacionados. O capítulo 3 descreve o processo de planejamento e especificação, e o capítulo 4 detalha a implementação. Por fim, o capítulo 5 traz o resultado da avaliação feita por estudantes e anunciantes, que mostrou que a solução atingiu o objetivo de facilitar a divulgação e a descoberta de vagas para atividades extracurriculares e de extensão no Instituto de Computação. Nesse capítulo final também são apresentadas sugestões de novas funcionalidades e melhorias nas que já foram implementadas.

### 1.4 USO DE INTELIGÊNCIA ARTIFICIAL COMO APOIO

Durante o desenvolvimento deste trabalho, foram utilizadas ferramentas de inteligência artificial generativa (particularmente o ChatGPT (OpenAI, 2025), da OpenAI) de forma pontual, exclusivamente para esclarecer dúvidas de programação, apoiar na configuração de ambientes de hospedagem em nuvem pública e auxiliar na formatação de texto e gráficos em L<sup>A</sup>T<sub>E</sub>X incluindo, por exemplo, a construção de diagramas de sequência apresentados

no Capítulo 4. Ressalta-se que todas as decisões de projeto, bem como o código e os textos produzidos, são de responsabilidade integral do autor.

## 2 TRABALHOS RELACIONADOS

Neste capítulo são apresentados trabalhos relacionados ao tema deste projeto, destacando plataformas comerciais e não-comerciais existentes que facilitam a divulgação e descoberta de vagas. Contudo, como será discutido, nenhuma das soluções analisadas atende plenamente às necessidades específicas do contexto acadêmico universitário abordado neste trabalho, particularmente no Instituto de Computação da Universidade Federal do Rio de Janeiro.

### 2.1 SOLUÇÕES COMERCIAIS

Nesta seção, são apresentadas as principais plataformas profissionais utilizadas para recrutamento e selecionadas por oferecerem quadros de vagas, assim como se pretende oferecer. Destaca-se seu funcionamento e, principalmente, suas limitações no contexto acadêmico brasileiro.

#### 2.1.1 LinkedIn

O LinkedIn é uma das plataformas profissionais mais conhecidas mundialmente, amplamente utilizada para recrutamento na indústria (LINKEDIN, 2025). Seu foco está em conectar profissionais e empresas, oferecendo ferramentas avançadas para recrutamento corporativo, como anúncios pagos, segmentação detalhada e análise sobre o desempenho das vagas divulgadas. Apesar de suas funcionalidades, a plataforma apresenta algumas limitações, entre as quais se destacam:

- **Custo elevado e baixa acessibilidade para o público e instituições acadêmicas:** O modelo de negócio da plataforma é baseado em anúncios pagos com preços dinâmicos, o que resulta em custos elevados para a divulgação eficiente de vagas. Isso torna a plataforma pouco acessível para instituições acadêmicas, laboratórios ou projetos de pesquisa que geralmente não possuem orçamento para investir nesses recursos.
- **Público-alvo desalinhado:** A plataforma é orientada principalmente para o mercado de trabalho formal e para profissionais já inseridos no setor, não contemplando de forma adequada oportunidades acadêmicas como iniciação científica, estágios não-obrigatórios ou projetos de extensão universitária.
- **Ausência de funcionalidades específicas:** Não há recursos próprios para a divulgação de oportunidades acadêmicas que envolvem requisitos diferenciados, como participação em projetos de pesquisa, perfil estudantil ou vínculo institucional.

### 2.1.2 Indeed

A Indeed é uma plataforma global dedicada ao recrutamento em massa, sendo bastante utilizada por empresas para divulgação de vagas e captação de candidatos em grande escala (INDEED, 2025). Oferece ferramentas como anúncios pagos, filtros detalhados e acompanhamento do desempenho das vagas. No entanto, apesar dessas funcionalidades, a plataforma apresenta as mesmas limitações no contexto acadêmico brasileiro que o LinkedIn e que foram citadas acima.

### 2.1.3 Conclusões

Ainda que as plataformas comerciais como LinkedIn e Indeed ofereçam recursos sofisticados para recrutamento no mercado profissional, apresentam sérias limitações quanto ao atendimento das necessidades específicas do contexto universitário brasileiro, tais como:

- Barreiras financeiras para publicação de vagas de baixo orçamento.
- Falta de funcionalidades adaptadas para oportunidades acadêmicas.
- Dificuldade de alcançar estudantes universitários ainda em formação.
- Ausência de integração com fluxos, requisitos e práticas do ambiente acadêmico.

## 2.2 SOLUÇÕES NÃO-COMERCIAIS INTERNAS

A seguir, são detalhados os principais métodos atualmente empregados no Instituto de Computação da UFRJ para divulgação de oportunidades acadêmicas. Embora tais estratégias sejam largamente utilizadas, elas apresentam limitações importantes em termos de alcance, organização, acessibilidade e eficiência.

### 2.2.1 E-mail Institucional

O e-mail institucional é um dos meios mais tradicionais e ainda amplamente utilizados para comunicação acadêmica. Professores, coordenações e setores administrativos enviam comunicados sobre bolsas, estágios, iniciações científicas e eventos diretamente para listas de e-mails de alunos. Entre as limitações desse meio de divulgação, destacam-se:

- **Baixa organização e arquivamento:** As oportunidades se perdem rapidamente entre diversos outros e-mails, dificultando a busca por vagas antigas ou ainda vigentes.
- **Ruído de informações e baixa taxa de leitura:** Muitos estudantes deixam de acompanhar suas caixas institucionais devido ao alto volume de mensagens não relacionadas diretamente a oportunidades, levando à baixa taxa de leitura.

### 2.2.2 Grupos e Canais de Mensagens Instantâneas (WhatsApp, Telegram)

O uso de grupos de mensagens instantâneas, especialmente serviços comerciais como WhatsApp e Telegram, tornou-se popular para divulgação ágil de vagas, projetos, eventos e outras oportunidades. Tais grupos podem ser criados por professores, coordenação ou pelos próprios alunos. Entre as limitações desse meio de divulgação, destacam-se:

- **Excesso de mensagens, dispersão e baixa taxa de leitura:** O alto volume de conversas, memes, avisos diversos e dúvidas cotidianas faz com que anúncios importantes rapidamente “subam” no histórico, tornando difícil para novos membros localizarem informações relevantes.
- **Alcance restrito:** Nem todos os alunos participam dos grupos corretos, seja por desconhecimento, lotação do grupo, ou porque não desejam misturar assuntos pessoais com acadêmicos nos seus aplicativos de mensagens.
- **Falta de padronização:** Não existe uma estrutura uniforme para apresentação das oportunidades, prejudicando a clareza e a comparação entre vagas.
- **Privacidade e notificações:** Muitos alunos silenciaram notificações dos grupos, reduzindo ainda mais o alcance das mensagens importantes.

### 2.2.3 Cartazes Físicos e Murais

A afixação de cartazes em murais espalhados pelo campus, corredores do Instituto ou laboratórios também é uma prática comum para divulgação de oportunidades. No entanto, observa-se limitações como:

- **Alcance restrito:** A informação só é acessível para quem circula física e regularmente pelos locais em que os cartazes estão afixados.
- **Atualização lenta:** Cartazes frequentemente ficam desatualizados ou são retirados sem reposição, dificultando o acesso a informações recentes.
- **Sustentabilidade e acessibilidade:** Além de demandar papel e impressão, cartazes não são acessíveis a pessoas com deficiência visual.

### 2.2.4 Sites ou Páginas Não-Padronizadas

Alguns laboratórios, departamentos ou docentes mantêm páginas próprias, geralmente em sites simples ou blogs, onde divulgam oportunidades específicas. Entre as limitações dessa forma de divulgação, destacam-se:

- **Fragmentação da informação:** Não há um repositório centralizado, obrigando o aluno a consultar múltiplas páginas para se manter informado.

- **Dificuldade de manutenção:** Muitas páginas não são atualizadas com frequência, resultando em informações desatualizadas ou vagas expiradas.
- **Falta de visibilidade:** A maioria desses sites não é amplamente divulgada, limitando o alcance das oportunidades publicadas.

### 2.2.5 Conclusões

A despeito de sua importância histórica e do esforço dos docentes e setores administrativos, as soluções existentes apresentam sérias limitações, que afetam negativamente suas eficiências na divulgação de oportunidades extracurriculares obrigatórias para a graduação. São limitações comuns:

- Dificuldade de acesso igualitário a todos os alunos, pois nem todos os alunos têm conhecimento ou acesso a todos os pontos de divulgação.
- Baixa taxa de leitura devido a perda ou esquecimento de oportunidades por falhas de arquivamento ou excesso de informações.
- Pouca transparência nas divulgações.

## 2.3 DIFERENCIAL DA SOLUÇÃO PROPOSTA

A aplicação desenvolvida neste projeto visa preencher lacunas identificadas na divulgação e descoberta de vagas no ambiente acadêmico e é especialmente adaptada às restrições financeiras e estruturais comuns no Instituto de Computação e em outras instituições universitárias. Ao contrário das plataformas comerciais como LinkedIn e Indeed, esta solução é desenhada para ser mantida pelas próprias instituições ou organizações a um baixo custo de manutenção, tanto humano quanto financeiro. Isso possibilita que a divulgação e consumo de informações nela sejam gratuitas e acessíveis a todos os membros da comunidade. Ademais, é projetada para ser fácil de utilizar e oferecer funcionalidades adaptadas ao contexto universitário, como: categorização clara das atividades (pesquisa, extensão, estágio, voluntariado) e possibilidade de buscas e filtragem de critérios específicos (como tipo de vaga ou oferta de bolsas); arquivamento e centralização de divulgações de múltiplos grupos em um só lugar.

Além disso, identifica-se um trabalho correlato, mas com escopo distinto: o *Extra\_curriculum* (BORBON; MARTINS, 2024), desenvolvido por Borbon e Martins (2024), que propõe uma solução digital voltada à descoberta e gerenciamento de oportunidades extracurriculares na UFRJ, como grupos de extensão, eventos e monitorias. Apesar de ambos os trabalhos compartilharem o objetivo de centralizar e facilitar o acesso a informações acadêmicas, o foco do *Extra\_curriculum* está nas atividades extracurriculares em

geral, enquanto o presente projeto concentra-se especificamente na divulgação de vagas. Assim, trata-se de iniciativas complementares no contexto universitário.

Assim, destaca-se a originalidade e relevância da solução proposta ao atender diretamente ao público-alvo universitário, suprindo necessidades não contempladas adequadamente pelas soluções comerciais e não comerciais existentes.

### 3 ESPECIFICAÇÃO E PLANEJAMENTO

Neste capítulo, será apresentada a especificação e o planejamento do projeto, com ênfase no levantamento das necessidades dos usuários potenciais por meio de pesquisa direta. Inicialmente, detalha-se a metodologia adotada para coleta de informações — que envolveu a aplicação de questionários distintos para os principais perfis de usuário da solução proposta: candidatos a vagas e anunciantes de oportunidades. Em seguida, serão descritos em detalhes os objetivos, públicos-alvo, ferramentas utilizadas e estratégias de divulgação de cada questionário. Por fim, analisa-se os resultados obtidos e enumera-se as principais prioridades para o desenvolvimento da plataforma.

#### 3.1 PESQUISA COM USUÁRIOS

Para compreender as dificuldades enfrentadas pelas partes interessadas e identificar oportunidades de otimização dos processos de divulgação e descoberta de vagas, foram aplicados dois questionários distintos, direcionados aos diferentes perfis de usuários da futura plataforma:

- **Questionário A:** direcionado a candidatos a vagas (estudantes universitários);
- **Questionário B:** direcionado a anunciantes de vagas (professores, técnicos, pesquisadores, empresas, etc.).

A seguir, detalha-se cada questionário aplicado.

##### 3.1.1 Questionário A: Candidatos a Vagas

###### Objetivos:

- Descobrir o grau de interesse dos usuários no processo que se deseja otimizar;
- Identificar quais tipos de vagas são mais procuradas pelos estudantes;
- Verificar as soluções atualmente utilizadas para descoberta de vagas;
- Avaliar o nível de satisfação dos usuários em relação às soluções existentes;
- Levantar os principais problemas encontrados nas soluções atuais;
- Mapear expectativas em relação a novas soluções para o problema identificado.

**Público-alvo:** estudantes universitários interessados em oportunidades extracurriculares.

**Ferramentas e Divulgação:** o questionário foi elaborado e disponibilizado na plataforma Google Formulários, sendo divulgado por e-mail diretamente a estudantes do Instituto de Computação e de outros centros da Universidade Federal do Rio de Janeiro (UFRJ) entre 23 de outubro e 5 de novembro de 2023.

**Perguntas:** As perguntas utilizadas encontram-se no Apêndice 6.

### 3.1.2 Questionário B: Anunciantes de Vagas

#### Objetivos:

- Descobrir o grau de interesse dos usuários no processo de divulgação de vagas;
- Identificar quais tipos de vagas são mais divulgadas pelos anunciantes;
- Verificar as soluções atualmente utilizadas para divulgação de oportunidades;
- Avaliar o nível de satisfação dos anunciantes em relação às soluções existentes;
- Levantar os principais problemas encontrados nas soluções atuais;
- Mapear expectativas em relação a novas soluções para o problema identificado.

**Público-alvo:** anunciantes de vagas, como professores, técnicos, pesquisadores, líderes de atividades, empresas, etc.

**Ferramentas e Divulgação:** o questionário foi criado na plataforma Google Formulários, sendo divulgado por e-mail a docentes, técnicos e estudantes do Instituto de Computação, bem como de outros centros da UFRJ entre 23 de outubro e 5 de novembro de 2023. Apesar do interesse em alcançar também empresas externas, a divulgação ficou restrita ao ambiente universitário devido à dificuldade de encontrar representantes das mesmas.

**Perguntas:** As perguntas utilizadas encontram-se no Apêndice 6.

## 3.2 ANÁLISE DOS RESULTADOS

Após duas semanas de coleta, 92 candidatos e 27 anunciantes responderam aos questionários. A análise dos dados coletados foi realizada de forma quantitativa (para perguntas fechadas) e qualitativa (para perguntas abertas), permitindo identificar padrões e prioridades comuns em ambos os perfis de usuário.

### Principais achados – Candidatos a vagas

A grande maioria dos estudantes demonstrou alto interesse na busca por oportunidades extracurriculares, indicando a relevância de uma solução voltada para centralização dessas vagas. Os meios mais utilizados atualmente para descobrir vagas são mensagens

de *broadcast*, como e-mails institucionais e grupos de redes sociais, além do contato por terceiros.

Apesar do interesse elevado, predomina a insatisfação em relação às ferramentas atualmente utilizadas: a maioria relatou dificuldades para encontrar oportunidades relevantes e afirmou que o processo é trabalhoso e disperso, devido à falta de centralização das informações. Muitos também gostariam de contar com filtros que permitissem refinar a busca de acordo com critérios específicos (tipo de vaga, área, remuneração etc.).

### Principais achados – Anunciantes de vagas

O grupo de anunciantes é composto, principalmente, por professores e estudantes envolvidos em projetos, que divulgam vagas ocasionalmente. Os principais canais de divulgação também são digitais, com ênfase em e-mails e mensagens diretas para potenciais interessados.

Entre os desafios mais citados estão a dificuldade de atingir o público-alvo desejado e a falta de mecanismos para segmentar a divulgação das vagas. Os anunciantes relataram insatisfação com a dificuldade de organizar as oportunidades em um local centralizado, bem como com o trabalho manual envolvido na divulgação repetitiva por diferentes canais. Além disso, manifestaram o desejo de poder apresentar informações mais completas sobre seus projetos e oportunidades, o que facilitaria o engajamento dos estudantes.

### Conclusões

De maneira geral, tanto candidatos quanto anunciantes expressam insatisfação com as ferramentas e processos atualmente disponíveis para descoberta e divulgação de vagas. A partir da avaliação das respostas, chegou-se uma lista de prioridades de funcionalidades por perfil de usuário, que podem ser resumidas em:

- Prioridades para usuários do tipo candidato:
  - **Centralização da informação:** os usuários querem encontrar as mais diversas oportunidades disponíveis para universitários dentro de uma única plataforma.
  - **Possibilidade de filtrar vagas:** os usuários querem filtrar vagas disponíveis para encontrar apenas aquelas que atendem às suas necessidades.
  - **Possibilidade de se inscrever em vagas:** os usuários querem se candidatar a vagas de forma remota e direta, isto é, de qualquer lugar e sem precisar recorrer a terceiros.
- Prioridades para usuários do tipo anunciante:

- **Centralização da divulgação:** os usuários querem divulgar em um só lugar e atingir o maior número possível de universitários.
- **Possibilidade de divulgação direta para alunos:** os usuários querem divulgar vagas diretamente para universitários sem precisar recorrer a intermediários que repassam seus anúncios.
- **Possibilidade de filtrar público-alvo:** os usuários querem divulgar vagas apenas para grupos específicos de universitários.
- **Possibilidade de divulgar o seu projeto, empresa ou atividade:** os usuários querem poder divulgar informações sobre suas atividades para possíveis candidatos.

Esses achados fundamentaram a priorização das funcionalidades da plataforma desenvolvida neste trabalho, visando atender de forma objetiva às principais demandas identificadas junto aos usuários.

### 3.3 METODOLOGIA DE DESENVOLVIMENTO SCRUM

Para o desenvolvimento da plataforma, foi adotada uma metodologia inspirada no Scrum, um *framework* ágil amplamente utilizado no desenvolvimento de software (SCHWABER; SUTHERLAND, 2020). O Scrum promove a entrega incremental de valor ao usuário por meio de ciclos curtos e iterativos chamados Sprints, com foco na adaptação às mudanças e na colaboração contínua entre os membros da equipe e as partes interessadas.

Diferente dos modelos tradicionais, o Scrum não exige um levantamento fechado de requisitos antes do início do desenvolvimento. Os requisitos evoluem de forma contínua, com base em descobertas realizadas ao longo do projeto. Esse processo teve início com uma etapa de descoberta (detalhada na seção anterior), na qual foram coletadas informações junto aos usuários e convertidas em itens do *Product Backlog*. Segundo Schwaber e Sutherland (SCHWABER; SUTHERLAND, 2020), o *Product Backlog* é uma lista ordenada por prioridade e em constante evolução, contendo todas as funcionalidades necessárias ao produto, sendo a única fonte de trabalho da equipe de desenvolvimento.

Esses itens são geralmente descritos no formato de histórias de usuário, uma técnica comum em metodologias ágeis para representar requisitos do ponto de vista do usuário. Segundo (COHN, 2004), cada história de usuário descreve uma funcionalidade desejada com foco no valor entregue, geralmente utilizando linguagem simples e a estrutura: “Como [tipo de usuário], quero [objetivo] para [benefício].” Esse formato facilita a compreensão, priorização e divisão do trabalho pela equipe de desenvolvimento.

O processo de desenvolvimento pode ser resumido nas seguintes etapas:

1. **Levantamento inicial e construção do *Product Backlog***

- Realização de entrevistas, questionários ou outras técnicas para compreender as necessidades dos usuários.
- Elaboração de histórias de usuário, organizadas em um *Backlog* inicial.

## 2. Planejamento da *Sprint* (*Sprint Planning*)

- Seleção dos itens do *Product Backlog* que serão desenvolvidos durante a *Sprint*.

## 3. Execução da *Sprint*

- Desenvolvimento das funcionalidades, acompanhamento de progresso e resolução de impedimentos.

## 4. Revisão da *Sprint* (*Sprint Review*)

- Apresentação das funcionalidades concluídas para coleta de *feedback* das partes interessadas.

## 5. Retrospectiva da *Sprint* (*Sprint Retrospective*)

- Análise dos pontos fortes e de melhorias para o próximo ciclo.

Durante esse processo, o *Product Backlog* é constantemente refinado para refletir as mudanças nas necessidades do projeto.

## 3.4 APLICAÇÃO DA METODOLOGIA SCRUM NO PROJETO

Neste projeto, inspirou-se na metodologia Scrum para gerenciar o desenvolvimento e levantamento de requisitos para aplicação. A escolha pelo Scrum se deveu à necessidade de flexibilidade no levantamento e refinamento dos requisitos, à possibilidade de entregas incrementais e ao alinhamento com a dinâmica de projetos acadêmicos, onde escopos e prioridades podem evoluir ao longo do tempo (SCHWABER; SUTHERLAND, 2020). Vale ressaltar que, por se tratar de um projeto acadêmico, algumas cerimônias do Scrum foram adaptadas em relação ao *framework* tradicional, priorizando as etapas de planejamento, execução e revisão ao final da *Sprint*, com ciclos ajustados à disponibilidade dos envolvidos.

Na fase inicial, foi realizada uma pesquisa com os principais usuários da plataforma (candidatos e anunciantes) para entender suas necessidades e dificuldades nos processos de descoberta e divulgação de oportunidades. Com base nessas informações, foram definidos objetivos mais amplos do sistema, organizados como épicos no *Product Backlog*.

Cada épico representa um conjunto maior de funcionalidades e, por isso, foi dividido em histórias de usuário menores e mais específicas, que podiam ser desenvolvidas ao longo

das *Sprints*. Esse processo de divisão ajudou a tornar os objetivos mais claros, a facilitar o planejamento e a garantir que as entregas fossem feitas de forma contínua e com foco no que realmente importava para os usuários.

Dessa forma, a adoção da metodologia Scrum permitiu que o desenvolvimento se mantivesse alinhado aos objetivos do projeto e às demandas reais das pessoas que vão usar a plataforma, o que trouxe mais clareza e valor para todos os envolvidos.

### 3.4.1 Construção de *Product Backlog*

Épico: Descoberta de Vagas (Candidato)

- **Visualizar vagas disponíveis:** Como candidato, quero visualizar todas as oportunidades disponíveis em um só lugar para facilitar minha busca por vagas.
- **Filtrar vagas:** Como candidato, quero filtrar as vagas com base em critérios como tipo de vaga, remuneração ou presença de palavra-chave, para encontrar apenas aquelas que atendem às minhas necessidades.
- **Candidatar-se a uma vaga:** Como candidato, quero me inscrever diretamente nas vagas de interesse para simplificar o processo de candidatura.

Épico: Divulgação de Vagas (Anunciante)

- **Criar vaga para divulgação:** Como anunciante, quero cadastrar minhas vagas na plataforma para que estudantes possam se candidatar.
- **Definir público-alvo da vaga:** Como anunciante, quero segmentar o público das minhas vagas para atingir os candidatos mais adequados.
- **Apresentar informações sobre a empresa ou projeto:** Como anunciante, quero disponibilizar informações sobre minha empresa para atrair candidatos alinhados com nossos valores.

### 3.4.2 Backlog da primeira *sprint* e escopo do projeto

Após a definição do *Product Backlog*, o desenvolvimento da plataforma foi organizado em *Sprints*, com o objetivo de promover a entrega incremental de funcionalidades. Embora tenha sido elaborado um planejamento de iterações futuras, este Trabalho de Conclusão de Curso (TCC) concentra-se apenas na primeira Sprint, dedicada à implementação das funcionalidades consideradas essenciais para a validação inicial da proposta.

As funcionalidades selecionadas para essa primeira iteração compõem o *Sprint Backlog* e foram definidas com base em critérios de viabilidade técnica, tempo disponível e valor entregue ao usuário final. A escolha visou garantir que a plataforma já oferecesse, desde

o início, uma experiência funcional para candidatos e anunciantes, mesmo em sua versão inicial. Foram estas:

Épico: Descoberta de Vagas (Candidato)

- **Visualizar vagas disponíveis:** Como candidato, quero visualizar todas as oportunidades disponíveis em um só lugar para facilitar minha busca por vagas.
- **Filtrar vagas:** Como candidato, quero filtrar as vagas com base em critérios como tipo de vaga, remuneração ou presença de palavra-chave, para encontrar apenas aquelas que atendem às minhas necessidades.

Épico: Divulgação de Vagas (Anunciante)

- **Criar vaga para divulgação:** Como anunciante, quero cadastrar minhas vagas na plataforma para que estudantes possam se candidatar.

As histórias escolhidas representam uma Prova de Conceito (POC) da plataforma, possibilitando que candidatos visualizem e filtrem oportunidades e que anunciantes iniciem o processo de divulgação de vagas. Essa entrega inicial permitiu validar, de forma prática, os principais fluxos da aplicação e serviu como base para evoluções nas Sprints futuras planejadas, mesmo que não implementadas neste TCC.

## 4 DESENVOLVIMENTO

O presente capítulo descreve detalhadamente o processo de desenvolvimento do sistema proposto, abrangendo desde as decisões arquiteturais até a implementação das funcionalidades principais. Inicialmente, apresenta-se a arquitetura lógica da solução, com a separação de responsabilidades entre as camadas de apresentação, aplicação e dados. Em seguida, é descrita a arquitetura física adotada, baseada no modelo cliente-servidor e apoiada por uma infraestrutura em nuvem utilizando serviços *serverless* da Amazon Web Services (AWS). São também discutidas as tecnologias selecionadas para cada camada do sistema, com justificativas técnicas e operacionais. Por fim, o capítulo detalha a implementação de funcionalidades centrais do sistema — como visualização, filtragem e cadastro de vagas — por meio de fluxos operacionais, diagramas de sequência e decisões de projeto que evidenciam a aplicação prática da arquitetura proposta.

### 4.1 ARQUITETURA LÓGICA DA SOLUÇÃO

A arquitetura lógica da solução segue o modelo em camadas, uma abordagem amplamente utilizada para promover separação de responsabilidades, modularidade, escalabilidade e manutenibilidade. As três camadas principais são:

- Camada de Apresentação

Responsável pela interação com o usuário final. Essa camada recebe comandos do usuário, apresenta dados e direciona suas entradas para a camada de aplicação. Oferece serviços para capturar ações do usuário e fornecer feedback visual e textual. Consome serviços disponibilizados pela camada de aplicação.

- Camada de Aplicação (ou Lógica de Negócio)

Responsável por implementar as regras de negócio do sistema. Atua como intermediária entre a camada de apresentação e a camada de dados. Recebe requisições da interface do usuário, realiza validações, cálculos ou decisões baseadas em regras de negócio e coordena chamadas à camada de dados. Expõe serviços utilizados pela camada de apresentação.

- Camada de Dados

Responsável pela persistência e recuperação das informações. Armazena dados estruturados e fornece serviços de consulta, inserção, atualização e exclusão. Interage apenas com a camada de aplicação, não sendo acessada diretamente pela interface do usuário.

Essa divisão promove o desacoplamento entre as partes do sistema, permitindo que cada uma evolua ou seja substituída sem impactar diretamente as demais. A separação entre cliente e servidor e as camadas lógicas adotadas está ilustrada na Figura 1.

## 4.2 ARQUITETURA FÍSICA DA SOLUÇÃO

Do ponto de vista físico, o sistema implementa o modelo cliente-servidor descrito por Tanenbaum e Van Steen (TANENBAUM; STEEN, 2025) e tradicionalmente utilizado em aplicações web. Ele é caracterizado por uma separação clara de responsabilidades, onde o servidor oferece serviços acessados sob demanda pelos clientes conectados. Em aplicações web como a deste projeto, o cliente (navegador do usuário) é responsável pela interface e interação com o usuário, enquanto o servidor é responsável por processar as requisições, aplicar as regras de negócio e acessar os dados persistidos.

A arquitetura foi desenvolvida sobre a infraestrutura da Amazon Web Services (AWS), utilizando serviços gerenciados e o modelo de computação *serverless*, no qual os recursos são executados sob demanda e cobrados por uso, sem necessidade de provisionamento ou manutenção manual de servidores. Essa escolha se justifica pela necessidade de baixos custos operacionais humanos e financeiros, especialmente relevantes no contexto acadêmico do projeto. Além disso, optou-se pela AWS por ser uma plataforma amplamente consolidada e já conhecida pelo autor, o que reduziu a curva de aprendizado e acelerou o desenvolvimento da solução.

As três camadas lógicas do sistema — apresentação, aplicação e dados — estão distribuídas entre os componentes do cliente e do servidor, conforme ilustra a Figura 1. A seguir, detalha-se a implementação de cada parte da arquitetura e as tecnologias adotadas.

### 4.2.1 Cliente: camada de apresentação

A camada de apresentação foi implementada como uma *Single Page Application (SPA)* desenvolvida com a biblioteca React<sup>1</sup>, em JavaScript. Nesse modelo, a aplicação é carregada uma única vez no navegador, e as alterações subsequentes de conteúdo são gerenciadas dinamicamente no lado do cliente, sem recarregamento completo da página, conforme ilustrado na parte superior da Figura 1. Isso proporciona uma experiência de navegação fluida e responsiva, semelhante à de aplicativos nativos.

Os arquivos estáticos do *frontend* — como HTML, JavaScript e CSS — são armazenados no Amazon S3<sup>2</sup> e distribuídos globalmente via Amazon CloudFront<sup>3</sup>, que assegura

<sup>1</sup> Biblioteca JavaScript de código aberto para criação de interfaces de usuário, mantida pela Meta Platforms, Inc., focada em componentes reutilizáveis e atualização eficiente da interface. Fonte: (Meta Platforms, Inc., 2024)

<sup>2</sup> Serviço de armazenamento de objetos da Amazon Web Services, utilizado para guardar e recuperar qualquer quantidade de dados, com alta durabilidade e disponibilidade. Fonte: (Amazon Web Services, 2024f)

<sup>3</sup> Rede de distribuição de conteúdo (CDN) da Amazon Web Services que entrega dados, vídeos e apli-

baixa latência e alta disponibilidade. A interface interage com o *backend* por meio de uma API, enviando requisições HTTP enviadas ao Amazon API Gateway<sup>4</sup>, que realiza o roteamento para os serviços apropriados no lado servidor.

#### 4.2.2 Servidor: camadas de aplicação e dados

A lógica de negócio é executada por funções AWS Lambda<sup>5</sup>, escritas em Python, sob o modelo *serverless*. Essas funções são acionadas sob demanda via API Gateway e executam tarefas como validações, inserção de dados e consultas ao banco, com escalabilidade automática e cobrança proporcional ao uso, como mostrado na sequência API Gateway → Lambda → RDS na Figura 1.

A persistência das informações é realizada em um banco de dados relacional PostgreSQL, hospedado no Amazon RDS (Relational Database Service)<sup>6</sup>. A comunicação entre as funções Lambda e o banco é feita por meio da biblioteca Psycopg2<sup>7</sup>, que permite a execução eficiente de comandos SQL, com suporte a transações e controle de exceções.

Essa separação entre lógica de aplicação e armazenamento garante modularidade e facilita a manutenção e evolução dos componentes do sistema de forma independente.

#### 4.2.3 Comunicação assíncrona e eventos

Além da comunicação síncrona via API REST, o sistema adota uma abordagem orientada a eventos para lidar com operações que não exigem resposta imediata, como o envio de e-mails. Nesse caso, a comunicação assíncrona é utilizada exclusivamente para integrar componentes internos da aplicação, conectando a API responsável pelo cadastro de vagas ao serviço de envio de e-mails. Esse fluxo é implementado com o Amazon SNS (Simple Notification Service)<sup>8</sup>, seguindo o padrão de publicação/assinatura.

Quando uma nova vaga é cadastrada, a função AWS Lambda responsável armazena os dados no banco e, em seguida, publica uma mensagem em um tópico SNS (Figura 2). Essa mensagem aciona outra função Lambda, dedicada ao envio de e-mails por meio do

---

cações com baixa latência e alta taxa de transferência. Fonte: (Amazon Web Services, 2024b)

<sup>4</sup> Serviço da Amazon Web Services para criação, publicação, manutenção e segurança de APIs em qualquer escala, permitindo integração com diversos serviços *backend*. Fonte: (Amazon Web Services, 2024a)

<sup>5</sup> Serviço de computação *serverless* da Amazon Web Services que executa código sob demanda, cobrando apenas pelo tempo de execução. Fonte: (Amazon Web Services, 2024g)

<sup>6</sup> Serviço gerenciado de banco de dados relacional da Amazon Web Services, que cuida de tarefas como provisionamento, backups e atualizações de software. Fonte: (Amazon Web Services, 2024c)

<sup>7</sup> Adaptador de banco de dados PostgreSQL para Python, amplamente utilizado para executar comandos SQL, gerenciar transações e interagir de forma eficiente com bancos relacionais. Fonte: (Psycopg Community, 2024)

<sup>8</sup> Serviço de mensageria no modelo publicação/assinatura (*pub/sub*) da Amazon Web Services, que permite a comunicação assíncrona entre sistemas. Fonte: (Amazon Web Services, 2024e)

Amazon SES (Simple Email Service)<sup>9</sup>, que notifica o anunciante de que a vaga foi recebida e será analisada.

Esse encadeamento (Lambda cadastro → SNS → Lambda e-mail → SES) está indicado na Figura 2 e oferece benefícios importantes:

Esse modelo oferece benefícios importantes:

- **Desacoplamento:** produtores e consumidores de eventos operam de forma independente;
- **Tolerância a falhas:** falhas em uma etapa, como o envio de e-mails, não comprometem o restante do fluxo;
- **Escalabilidade:** novos consumidores podem ser adicionados ao fluxo sem necessidade de alterar os produtores existentes.

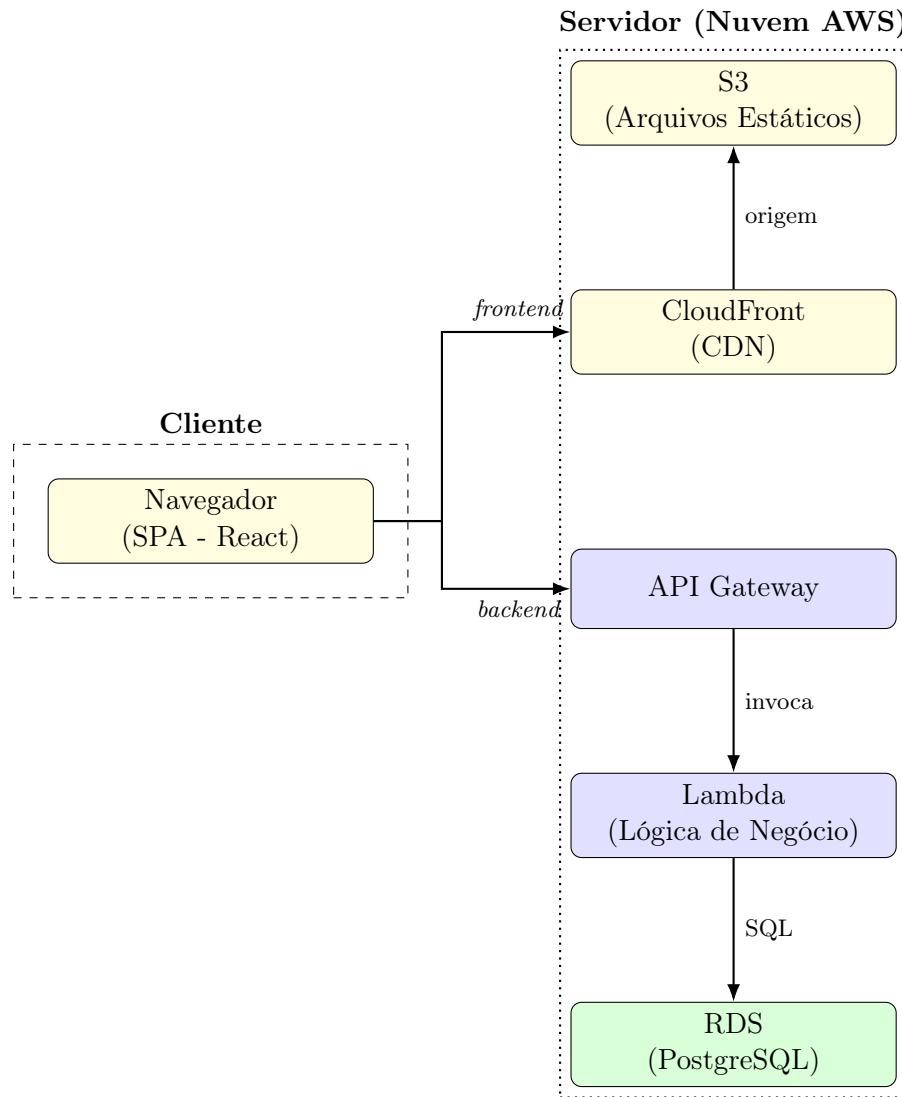
Essa arquitetura híbrida, que combina comunicação síncrona e assíncrona, aumenta a robustez e a flexibilidade da solução, permitindo que ela evolua e escale com mais facilidade.

A Figura 1 apresenta o fluxo de comunicação síncrona entre cliente e servidor, com camadas lógicas destacadas por cores.

---

<sup>9</sup> Serviço de envio de e-mails transacionais e marketing da Amazon Web Services, que permite alto volume de entrega com controle de reputação e taxas de entrega. Fonte: (Amazon Web Services, 2024d)

Figura 1 – Fluxo síncrono cliente–servidor com camadas destacadas: apresentação (amarelo), aplicação (azul) e dados (verde).

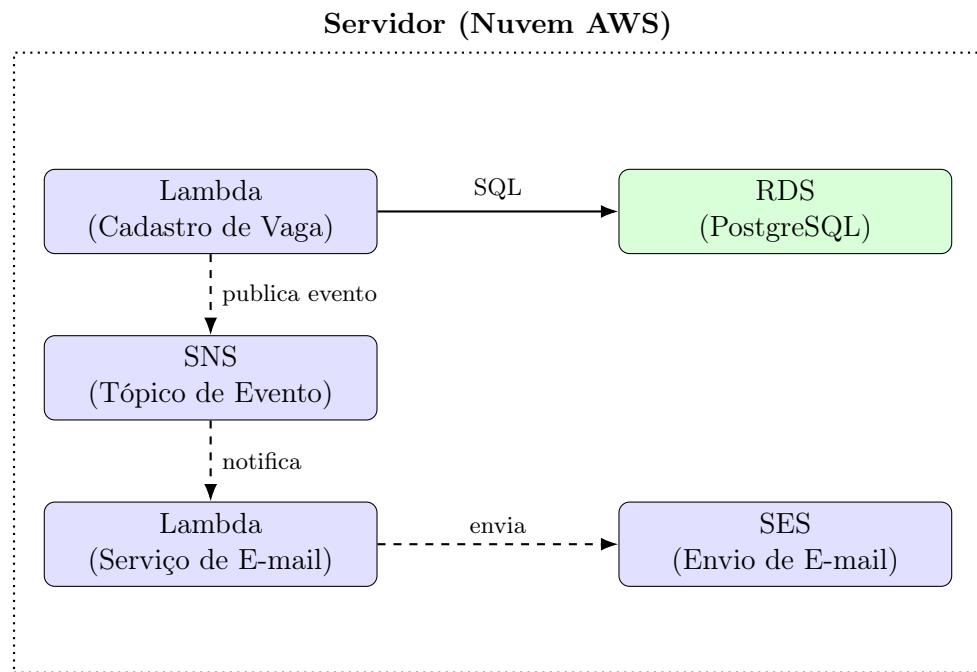


#### Legenda:

- Camada de **Apresentação**
- Camada de **Aplicação**
- Camada de **Dados**
- ..... Contorno pontilhado: **Servidor (nuvem AWS)**
- ----- Contorno tracejado: **lado cliente**
- → Setas: **fluxo síncrono (requisições)**

A Figura 2 mostra o fluxo assíncrono usado para comunicação entre componentes internos do backend.

Figura 2 – Fluxo assíncrono interno (pub/sub) no backend: a lógica de negócios publica um evento (SNS) consumido pelo serviço interno de e-mail (Lambda → SES). Camada de aplicação em azul; dados em verde.



**Legenda:**

- Camada de **Aplicação**
- Camada de **Dados**
- ..... Contorno pontilhado: **Servidor (nuvem AWS)**
- Setas tracejadas: **fluxo assíncrono (eventos pub/sub)**

#### 4.2.4 Resumo das tecnologias utilizadas no sistema

Em resumo, tem-se que, ao longo do desenvolvimento, utilizou-se diversas tecnologias com papéis diferentes. Foram elas:

- **Linguagens de Programação:**

- **Python**

Utilizada na implementação das funções AWS Lambda responsáveis pela lógica de negócios (Python Software Foundation, 2024).

- **JavaScript**

Linguagem utilizada no desenvolvimento da interface do usuário no navegador, por meio da biblioteca React (Ecma International, 2024).

- **Bibliotecas e Frameworks**

- **React**

Biblioteca JavaScript utilizada no desenvolvimento da interface do usuário no navegador. Permite a criação de interfaces gráficas dinâmicas, reutilizáveis e com carregamento assíncrono de dados via API (Meta Platforms, Inc., 2024).

- **Boto3**

Biblioteca oficial da AWS para Python, utilizada para interagir com serviços como SNS (publicação de mensagens) e SES (envio de e-mails) diretamente a partir das funções Lambda (Amazon Web Services, 2024h).

- **Psycopg2**

Biblioteca para interação com PostgreSQL para Python, utilizada para conectar-se ao banco de dados, executar comandos SQL e manipular os resultados dentro das funções Lambda (Psycopg Community, 2024).

- **Banco de Dados**

- **Postgres no AWS Relational Database Service (RDS)**

Serviço gerenciado de banco de dados relacional utilizado para armazenar as informações de vagas (Amazon Web Services, 2024c).

- **Computação e Hospedagem**

- **AWS Simple Storage Service (S3)**

Utilizado para armazenar e hospedar os arquivos estáticos do *frontend* (Amazon Web Services, 2024f).

- **AWS CloudFront**

Serviço de CDN (Content Delivery Network) utilizado para distribuir os arquivos do *frontend* com baixa latência (Amazon Web Services, 2024b).

- **AWS Lambda**

Serviço de computação serverless utilizado para executar as funções do *backend* sob demanda, eliminando a necessidade de gerenciamento de servidores (Amazon Web Services, 2024g).

- **AWS API Gateway**

Responsável por expor os endpoints HTTP da API (POST /jobs, GET /jobs) e rotear as requisições para as funções Lambda correspondentes (Amazon Web Services, 2024a).

- **Interações e Comunicação Assíncrona**

- **AWS Simple Notification Service (SNS)**

Serviço de mensageria utilizado para implementar o padrão pub/sub. Permite que eventos, como o cadastro de uma nova vaga, disparem automaticamente ações em outros componentes, como o envio de e-mails (Amazon Web Services, 2024e).

- **AWS Simple Email Service (SES)**

Serviço utilizado para o envio de e-mails transacionais, como a notificação ao anunciante após o cadastro de uma vaga (Amazon Web Services, 2024d).

## 4.3 IMPLEMENTAÇÃO DAS FUNCIONALIDADES

### 4.3.1 Visualização de Vagas Disponíveis

**Objetivo:** Permitir que o Usuário visualize todas as oportunidades disponíveis de forma paginada.

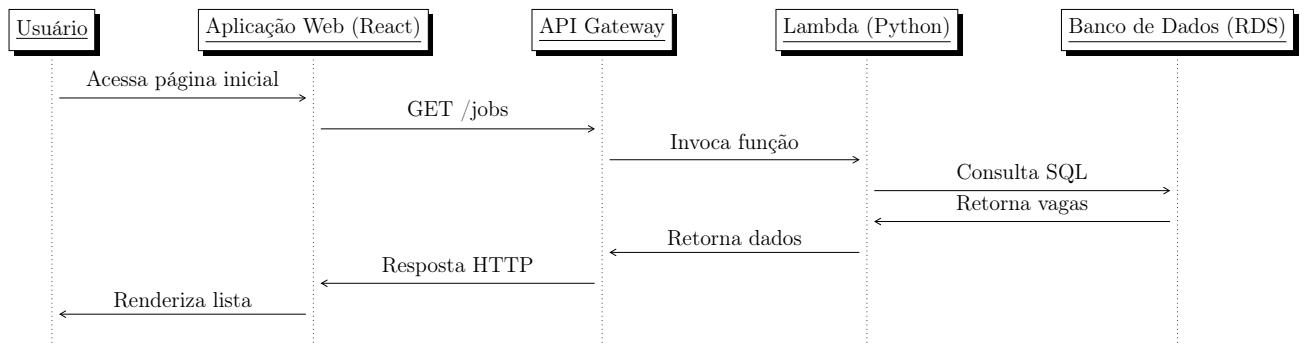
**Critérios de Aceitação Atendidos:**

- O sistema exibe uma lista de vagas disponíveis de forma paginada;
- Cada vaga apresenta detalhes básicos como título, tipo, empregador, localidade, etc.;
- O usuário pode navegar entre páginas da lista de vagas através de botões disponíveis na interface.

**Fluxo de Funcionamento:**

1. O usuário acessa a página inicial da aplicação
2. A aplicação web envia, de forma assíncrona, uma requisição GET /jobs ao Amazon API Gateway
3. O AWS API Gateway aciona uma função AWS Lambda escrita em Python passando como parâmetro a requisição que recebeu da aplicação web
4. A função Lambda executa uma consulta SQL no banco de dados PostgreSQL, hospedado no Amazon RDS, para obter as vagas disponíveis
5. A AWS Lambda retorna a lista de vagas ao AWS API Gateway
6. O API Gateway responde à aplicação web com os dados
7. A aplicação renderiza dinamicamente a lista de vagas e a exibe ao usuário

Figura 3 – Diagrama de sequência da funcionalidade de visualização de vagas.



### Componentes e Tecnologias Envolvidas:

- **Aplicação Web (Interface com Usuário)**

- **Objetivo:** Fornecer uma interface gráfica para visualização das vagas disponíveis.
- **Tecnologias Utilizadas**
  - \* **Linguagem de Programação:** Javascript
  - \* **Bibliotecas:** React, Tailwind CSS
  - \* **Serviços Cloud e Hospedagem:** AWS Cloudfront e AWS S3

- **API**

- **Objetivo:** Buscar as vagas no banco de dados e fornecer à aplicação web.
- **Tecnologias Utilizadas**
  - \* **Linguagem de Programação:** Python
  - \* **Bibliotecas:** Psycopg2
  - \* **Serviços Cloud e Hospedagem:** AWS API Gateway, AWS Lambda

- **Banco de Dados**

- **Objetivo:** Armazenar e recuperar informações sobre as vagas.
- **Tecnologias Utilizadas**
  - \* **Sistema de Gerenciamento de Banco de Dados:** PostgreSQL
  - \* **Serviços Cloud e Hospedagem:** AWS RDS

### Decisões Técnicas Relevantes:

Para otimizar o tempo de carregamento, a exibição da lista de vagas é paginada, limitando o número de registros retornados por requisição. A navegação entre páginas é feita diretamente na interface, com novos dados sendo buscados conforme necessário.

A comunicação com a API é feita de forma assíncrona, utilizando a API nativa do JavaScript (*fetch*). A renderização da lista é feita sem recarregamento da página, aproveitando os recursos de atualização dinâmica do modelo *Single Page Application (SPA)*.

A escolha dos serviços serverless AWS Lambda e API Gateway para desenvolvimento e hospedagem da API foi motivada pela familiaridade do desenvolvedor com a tecnologia e pelo baixo custo estimado de operação e manutenção. Considerando que o sistema foi desenvolvido para atender um instituto com cerca de 1000 alunos; e que cada sessão de uso por um aluno está associada a 20 requisições à API em média; temos que no caso de uso diário por todos os alunos, a API deveria receber cerca de 620.000 requisições mensais. Como a AWS oferece 1 milhão de requisições mensais gratuitas ao API Gateway e Lambda por mês até a presente data, tem-se que não haveria custos para atender o volume de requisições estimado. Ademais, como trata-se de serviços *serverless* e neste modelo o código é executado sob demanda e o provedor (no caso, a AWS) se responsabiliza por toda a infraestrutura necessária para escalar, monitorar e manter os serviços, também não haveria gastos com manutenção.

A hospedagem do *frontend* foi implementada com o uso do Amazon S3, responsável por armazenar os arquivos estáticos da aplicação, e do Amazon CloudFront, que atua como uma *CDN (Content Delivery Network)*, distribuindo esses arquivos a partir de servidores geograficamente próximos aos usuários. A escolha por essa combinação se deu pela baixa complexidade de configuração, alta disponibilidade e excelente desempenho, além da integração nativa com a infraestrutura AWS. Como a AWS oferece atualmente 1 TB/mês de transferência gratuita via CloudFront, a solução permite operação sem custo direto e com mínima necessidade de manutenção.

#### 4.3.2 Filtragem de Vagas

**Objetivo:** Permitir que o usuário filtre a lista de vagas com base em critérios como tipo de vaga, remuneração e modalidade de trabalho, visualizando apenas as oportunidades que atendem às suas preferências.

**Critérios de Aceitação Atendidos:**

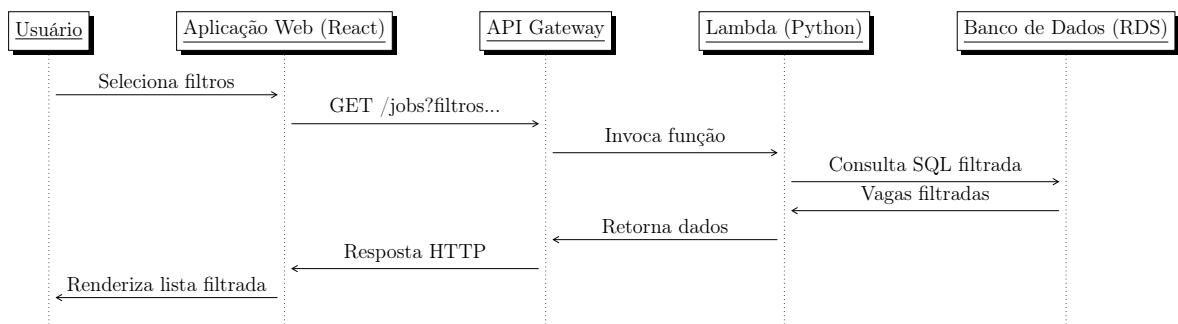
- O sistema permite a filtragem por tipo de vaga (estágio, iniciação científica, etc.), remuneração e modalidade de trabalho;
- Os filtros podem ser aplicados simultaneamente.

**Fluxo de Funcionamento:**

1. O usuário acessa a página inicial da aplicação, onde há a lista de vagas e interage com os filtros disponíveis na interface.

2. A aplicação web envia, de forma assíncrona, uma requisição GET /jobs ao AWS API Gateway, incluindo os filtros selecionados como parâmetros na URL.
3. O AWS API Gateway aciona uma função AWS Lambda escrita em Python, passando como parâmetro a requisição que recebeu da aplicação Web.
4. A função Lambda executa uma consulta SQL no banco de dados PostgreSQL, hospedado no Amazon RDS, para obter as vagas disponíveis que atendem aos critérios selecionados
5. A AWS Lambda retorna a lista de vagas ao AWS API Gateway
6. O API Gateway responde à aplicação web com os dados
7. A aplicação renderiza dinamicamente a lista de vagas e a exibe ao usuário

Figura 4 – Diagrama de sequência da funcionalidade de filtragem de vagas.



### Componentes e Tecnologias Envolvidas:

- **Aplicação Web (Interface com Usuário)**

- **Objetivo:** Permitir a aplicação de filtros de busca e exibição de resultados dinâmicos. Fornecer uma interface gráfica para visualização das vagas disponíveis que atendem aos critérios selecionados.
- **Tecnologias Utilizadas**
  - \* **Linguagem de Programação:** Javascript
  - \* **Bibliotecas:** React, Tailwind CSS
  - \* **Serviços Cloud e Hospedagem:** AWS Cloudfront e AWS S3

- **API**

- **Objetivo:** Interpretar os filtros e buscar e retornar vagas correspondentes do banco de dados.
- **Tecnologias Utilizadas**

- \* **Linguagem de Programação:** Python
- \* **Bibliotecas:** Psycopg2
- \* **Serviços Cloud e Hospedagem:** AWS API Gateway, AWS Lambda

- **Banco de Dados**

- **Objetivo:** Armazenar e consultar dados de vagas com base em critérios filtrados.
- Tecnologias Utilizadas
  - \* **Sistema de Gerenciamento de Banco de Dados:** PostgreSQL
  - \* **Serviços Cloud e Hospedagem:** AWS RDS
  - \* **Recursos do SGBD:** *Full Text Search* para busca eficiente por palavras-chave no título e descrição das vagas.

**Decisões Técnicas Relevantes:**

A filtragem foi implementada no *backend* para garantir a integridade e consistência dos dados retornados, mesmo em casos de múltiplos filtros combinados. Os filtros selecionados pelo usuário se tornam parâmetros de busca na requisição de `GET /jobs`, o que permite que URLs possam ser compartilhadas com filtros já aplicados.

Para a implementação da busca textual por palavras-chave, foi utilizada a funcionalidade de *Full Text Search* do PostgreSQL<sup>10</sup>, que converte o conteúdo textual em um *vetor de termos* (`tsvector`) e interpreta a consulta do usuário como uma *expressão de busca* (`tsquery`). O campo de título e descrição das vagas é indexado com `GIN` (*Generalized Inverted Index*), o que garante consultas rápidas mesmo em bases com milhares de registros.

O fluxo geral dessa busca consiste em:

1. Ao cadastrar ou atualizar uma vaga, o título e a descrição são processados por meio da função `to_tsvector('portuguese', texto)`, que normaliza palavras (remoção de acentos, conversão para minúsculas e redução a radicais).
2. Na consulta, a entrada do usuário é transformada por `to_tsquery('portuguese', consulta)`, permitindo buscas por termos simples, múltiplos e operadores lógicos.
3. A comparação é feita utilizando o operador `@@`, que verifica se o vetor de termos contém os termos de busca.
4. Os resultados podem ser ordenados por relevância usando a função `ts_rank`, que considera a frequência e a posição dos termos no texto.

<sup>10</sup> PostgreSQL Global Development Group. *Full Text Search*. Disponível em: <https://www.postgresql.org/docs/current/textsearch.html>. Acesso em: 10 ago. 2025.

Esse recurso permite que a busca por “*desenvolvedor backend*” retorne resultados contendo variações como “*vaga para desenvolvedores backend em Python*” ou “*desenvolvedor back-end*”, sem a necessidade de correspondência exata de palavras.

Para otimizar o tempo de carregamento, a exibição da lista de vagas é paginada, limitando o número de registros retornados por requisição. A navegação entre páginas é feita diretamente na interface, com novos dados sendo buscados conforme necessário. A comunicação com a API é feita de forma assíncrona, utilizando a API nativa do JavaScript para requisições HTTP (*fetch*). A renderização da lista é feita sem recarregamento da página, aproveitando os recursos de atualização dinâmica do modelo *Single Page Application (SPA)*.

A escolha dos serviços serverless AWS Lambda e API Gateway para desenvolvimento e hospedagem da API foi motivada pela familiaridade do desenvolvedor com a tecnologia e pelo baixo custo estimado de operação e manutenção. Considerando que o sistema foi desenvolvido para atender a uma instituição com cerca de 1000 alunos, e que o volume de requisições estimado permanece dentro da camada gratuita da AWS, a utilização desses serviços garante operação sem custo direto e sem necessidade de manutenção de servidores.

A hospedagem do *frontend* foi implementada com o uso do Amazon S3, responsável por armazenar os arquivos estáticos da aplicação, e do Amazon CloudFront, que atua como uma *CDN (Content Delivery Network)*, distribuindo esses arquivos a partir de servidores geograficamente próximos aos usuários. A escolha por essa combinação se deu pela baixa complexidade de configuração, alta disponibilidade e excelente desempenho, além da integração nativa com a infraestrutura AWS. Como a AWS oferece atualmente 1 TB/mês de transferência gratuita via CloudFront, a solução permite operação sem custo direto e com mínima necessidade de manutenção.

#### 4.3.3 Cadastro de Vagas

**Objetivo:** Permitir que o usuário anunciante solicite a divulgação de uma nova oportunidade na plataforma por meio do preenchimento de um formulário com os dados da vaga.

**Critérios de Aceitação Atendidos:**

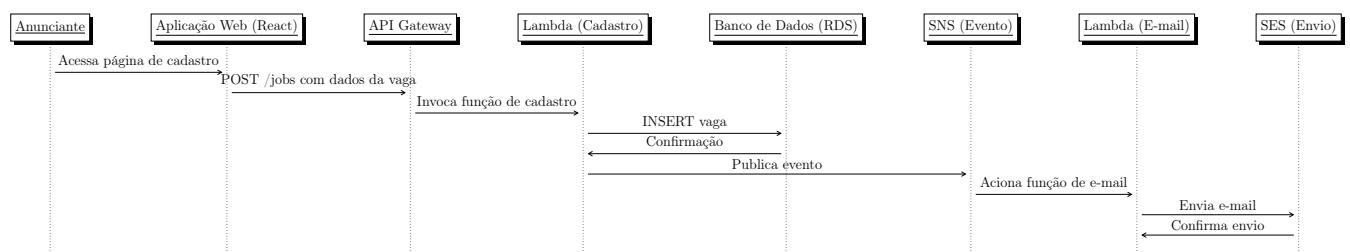
- O anunciantre pode preencher um formulário com dados sobre a vaga a ser divulgada na plataforma como tipo, modalidade de trabalho, descrição, link para inscrição, localidade, etc.;
- O anunciantre recebe um e-mail de confirmação após o cadastro da vaga.

**Fluxo de Funcionamento:**

1. O usuário acessa a página da aplicação dedicada a anunciantes.

2. O anunciente preenche e submete o formulário de cadastro de vagas na aplicação web
3. A aplicação web envia, de forma assíncrona, uma requisição POST /jobs ao Amazon API Gateway contendo os dados da vaga.
4. O API Gateway aciona uma função AWS Lambda escrita em Python, que valida os dados recebidos.
5. A função Lambda valida os dados; executa uma consulta SQL no banco de dados PostgreSQL para nele armazenar os dados; publica um evento no Amazon SNS informando o cadastro da nova vaga.
6. O SNS aciona uma segunda função AWS Lambda
7. A segunda função AWS Lambda envia um e-mail ao anunciente utilizando o Amazon SES como serviço de envio de emails

Figura 5 – Diagrama de sequência da funcionalidade de cadastro de vaga com notificação por e-mail.



### Componentes e Tecnologias Envolvidas:

- **Aplicação Web (Interface com Usuário)**

- **Objetivo:** Permitir o preenchimento de um formulário com informações sobre a vaga a ser divulgada na plataforma.
- **Tecnologias Utilizadas**
  - \* **Linguagem de Programação:** Javascript
  - \* **Bibliotecas:** React, Tailwind CSS
  - \* **Serviços Cloud e Hospedagem:** AWS Cloudfront e AWS S3

- **API**

- **Objetivo:** Receber os dados da vaga, salvá-los no banco de dados e publicar o evento de criação em uma fila AWS SNS.
- **Tecnologias Utilizadas**

- \* **Linguagem de Programação:** Python
- \* **Bibliotecas:** Psycopg2
- \* **Serviços Cloud e Hospedagem:** AWS API Gateway, AWS Lambda
- **Mensageria (Integração Assíncrona)**
  - **Objetivo:** Acionar processos complementares ao cadastro de novas vagas, como o envio de e-mails, de forma desacoplada do fluxo principal.
  - **Tecnologias Utilizadas**
    - \* **Serviços Cloud e Hospedagem:** Amazon SNS
- **Banco de Dados**
  - **Objetivo:** Armazenar os dados das vagas cadastradas.
  - **Tecnologias Utilizadas**
    - \* **Sistema de Gerenciamento de Banco de Dados:** PostgreSQL
    - \* **Serviços Cloud e Hospedagem:** AWS RDS
- **Serviço de E-mail**
  - **Objetivo:** Enviar e-mails de confirmação ao anunciante após o cadastro de uma vaga.
  - **Tecnologias Utilizadas**
    - \* **Serviços Cloud e Hospedagem:** Amazon SES

#### **Decisões Técnicas Relevantes:**

A arquitetura orientada a eventos foi adotada com o uso do Amazon SNS para publicar um evento assim que uma nova vaga é cadastrada. Esse evento é utilizado para acionar, de forma assíncrona, o serviço de envio de e-mails, sem que essa responsabilidade fique acoplada à função principal de cadastro.

Essa abordagem segue o padrão publicador/assinante (pub/sub), no qual um componente (no caso, a função Lambda responsável pelo cadastro) publica um evento em um tópico, e um ou mais componentes independentes assinam esse tópico e reagem a ele de forma automática. No fluxo atual da aplicação, há apenas um assinante: uma segunda função Lambda que envia o e-mail de confirmação via Amazon SES. No entanto, essa arquitetura permite que novos assinantes sejam adicionados a qualquer momento, como, por exemplo, uma função para registrar *logs*, notificar uma equipe interna ou enviar alertas por outros canais — tudo isso sem alterar a lógica principal da API.

Esse modelo traz benefícios importantes, como o desacoplamento entre componentes que facilita a manutenção; a resiliência, já que falhas no envio de e-mail não afetam o cadastro da vaga; e a escalabilidade, pois múltiplos consumidores podem processar o mesmo evento de forma paralela.

O uso do Amazon SES foi escolhido por sua integração nativa com a AWS, confiabilidade e baixo custo.

A escolha dos serviços serverless AWS Lambda e API Gateway para desenvolvimento e hospedagem da API foi motivada pela familiaridade do desenvolvedor com a tecnologia e pelo baixo custo estimado de operação e manutenção. Considerando que o sistema foi desenvolvido para atender a uma instituição com cerca de 1000 alunos e que o volume de uso previsto permanece dentro da camada gratuita da AWS, a aplicação pode ser mantida sem custos de infraestrutura e com mínima necessidade de suporte técnico.

De forma complementar, o *frontend* da aplicação foi hospedado no Amazon S3, com distribuição de conteúdo estático por meio do Amazon CloudFront, que atua como *CDN* (*Content Delivery Network*). Essa configuração foi escolhida por sua alta eficiência, simplicidade de uso e custo reduzido, uma vez que o baixo volume de tráfego estimado para atender ao corpo do Instituto também se encaixa dentro dos limites da gratuidade da AWS.

## 5 AVALIAÇÕES

Neste capítulo, apresenta-se a avaliação da solução desenvolvida, realizada por meio de entrevistas com usuários representativos dos perfis de candidatos e anunciantes de vagas do Instituto de Computação da UFRJ. O objetivo dessa etapa foi validar a efetividade da plataforma proposta, identificar pontos fortes e coletar sugestões de aprimoramento a partir do uso real pelos participantes.

### 5.1 METODOLOGIA DE AVALIAÇÃO

A avaliação da plataforma foi conduzida por meio de **entrevistas semi-estruturadas**, realizadas individualmente com estudantes de graduação e anunciantes de vagas (docentes e pesquisadores). A escolha por entrevistas deste tipo, em vez de questionários, se deu devido à natureza qualitativa e exploratória desta fase, visando aprofundar a compreensão sobre a experiência dos usuários, dificuldades encontradas, opiniões detalhadas e sugestões.

O processo de avaliação foi estruturado em três etapas principais:

1. **Sessão de uso supervisionada:** Cada participante foi convidado a utilizar a plataforma, navegando pelas funcionalidades principais, como:
  - Visualizar a lista de vagas disponíveis;
  - Aplicar filtros para refinar a busca;
  - Cadastrar uma nova vaga (no caso dos anunciantes).
2. **Entrevista individual:** Após a interação com a plataforma, foi realizada uma entrevista semi-estruturada, com perguntas abertas sobre:
  - Facilidade de uso e clareza da interface;
  - Percepção sobre centralização e organização das informações;
  - Dificuldades encontradas durante o uso;
  - Satisfação geral com a solução;
  - Sugestões de melhoria.
3. **Análise dos relatos:** As respostas e observações feitas durante as entrevistas foram transcritas, analisadas qualitativamente e agrupadas por temas, permitindo identificar padrões, pontos de convergência/divergência e exemplos representativos de feedback dos usuários.

## 5.2 PERFIL DOS PARTICIPANTES

Participaram da avaliação 8 usuários, distribuídos entre os seguintes perfis:

- **Candidatos a vagas:** 5 estudantes de graduação do Instituto de Computação, com níveis variados de experiência em busca de oportunidades acadêmicas;
- **Anunciantes de vagas:** 3 líderes de projetos de extensão ou grupos de atividades complementares envolvidos na divulgação de oportunidades em seus projetos.

As entrevistas foram realizadas remotamente, via videoconferência, e tiveram duração média de 30 minutos cada.

## 5.3 RESULTADOS QUALITATIVOS

A análise das entrevistas revelou percepções consistentes sobre o valor da plataforma, além de sugestões práticas para sua evolução. Abaixo são destacados os principais pontos relatados pelos participantes, ilustrados por citações representativas.

### 5.3.1 Pontos Fortes Identificados

- **Centralização e organização das vagas:** Todos os entrevistados destacaram como principal benefício a centralização das oportunidades em um único local. Um estudante relatou:

“Finalmente ficou fácil encontrar todas as vagas de iniciação e estágio sem precisar ficar pulando de grupo em grupo ou procurando e-mail antigo.”

- **Facilidade de uso da interface:** A interface foi descrita como “intuitiva”, “limpa” e “autoexplicativa”. Os participantes apreciaram a navegação fluida e a ausência de etapas desnecessárias.

“Gostei porque é simples, não tem excesso de informação, você já entende como funciona logo na primeira vez.”

- **Filtro de remuneração:** Os estudantes destacaram como ponto positivo o sistema de filtragem, principalmente o filtro por existência de remuneração, que permite filtrar por vagas dentro da Universidade que oferecem algum tipo de auxílio financeiro aos estudantes.

“Nenhum outro lugar permite ver só as vagas de atividades de extensão com bolsa e tem gente que precisa muito.”

### 5.3.2 Dificuldades e Pontos de Melhoria

#### 5.3.2.1 Estudantes

- **Filtros adicionais:** Alguns estudantes sugeriram a inclusão de novos filtros, como “área de atuação”.

“Seria ótimo se tivesse como filtrar por subárea, tipo por desenvolvimento de software, IA, análise de dados, por exemplo.”

- **Candidatura online:** A possibilidade de candidatura direta pela plataforma foi sugerida por mais de um participante, incluindo o acompanhamento do status da inscrição.

“Seria legal se desse pra se inscrever por aqui mesmo, sem ter que ir para outro site.”

- **Feedback visual em casos de erro ou resultados vazios:** Foi sugerido aprimorar as mensagens exibidas quando um filtro não retorna resultados.

#### 5.3.2.2 Anunciantes

- **Logo do Anunciante:** Anunciantes sugeriram a inclusão de suas logos em anúncios.

“Seria muito mais visual se desse para botar a logo do nosso grupo do lado do nome do grupo. As pessoas conseguiram associar bem melhor.”

- **Painel de controle para anunciantes:** Um anunciante sugeriu criar um painel para visualizar vagas cadastradas e acompanhar o status de análise/publicação.

“Seria útil se eu pudesse ver tudo que eu tenho ativo e acompanhar se meus pedidos de anúncio já foram aprovados ou ainda estão em análise.”

- **Newsletter personalizada:** Dois estudantes sugeriram a criação de uma *newsletter* personalizada aos interesses de cada usuário.

“Queria poder selecionar o que tenho interesse e receber isso via email para eu não precisar ter que ficar abrindo o site e procurando.”

### 5.3.3 Satisfação Geral

A satisfação dos participantes foi alta: todos declararam que utilizariam a plataforma no dia a dia e a recomendariam a colegas e também sugeriram que daria para replicar para outros cursos dentro da Universidade. Um estudante resumiu:

“Achei muito melhor que depender de e-mail e grupo de zap. Se todo mundo usar, facilita pra todo mundo.”

#### 5.4 CONSIDERAÇÕES FINAIS

A avaliação qualitativa, realizada por meio de entrevistas, permitiu validar a proposta da solução, identificar seus principais pontos fortes e reunir sugestões concretas para aprimoramento futuro. Os relatos evidenciaram que a plataforma cumpre o objetivo de centralizar e facilitar o acesso às vagas de atividades extracurriculares, sendo bem recebida tanto por estudantes quanto por anunciantes.

As sugestões levantadas pelos usuários reforçam o potencial de evolução do sistema, especialmente com a inclusão de novos filtros, envio de *newsletters* personalizadas, página do anunciantes e aprimoramento das mensagens de feedback. Esses resultados confirmam a relevância da solução desenvolvida, ao mesmo tempo em que orientam seus próximos passos de desenvolvimento.

## 6 CONCLUSÃO

Este trabalho teve como objetivo propor, desenvolver e avaliar uma plataforma web dedicada à centralização da divulgação e descoberta de oportunidades acadêmicas extra-curriculares no contexto do Instituto de Computação da Universidade Federal do Rio de Janeiro (UFRJ). A motivação surgiu da constatação, por meio de pesquisa com estudantes e anunciantes de vagas, de que os métodos tradicionais de divulgação — como e-mails, murais físicos e grupos de mensagens — apresentam limitações significativas de alcance, organização e eficiência.

Para enfrentar esse desafio, foram adotadas metodologias ágeis no planejamento e no desenvolvimento do sistema, com priorização de funcionalidades a partir de demandas reais dos usuários. O processo envolveu desde o levantamento e análise das necessidades dos diferentes perfis de usuários, passando pela modelagem da arquitetura e escolha das tecnologias, até a implementação das principais funcionalidades: visualização, filtragem e cadastro de vagas.

A avaliação da solução foi conduzida por meio de entrevistas semi-estruturadas com estudantes e anunciantes, permitindo a coleta de percepções detalhadas sobre usabilidade, praticidade e relevância do sistema. Os resultados mostraram que a plataforma atendeu aos objetivos iniciais: facilitou o acesso a oportunidades, centralizou informações antes dispersas e proporcionou uma experiência de uso positiva para ambos os públicos.

Entre os principais pontos fortes destacados pelos usuários estão a facilidade de uso da interface, a organização centralizada das vagas e a agilidade no cadastro. Por outro lado, a avaliação também permitiu identificar oportunidades de aprimoramento, como a inclusão de novos filtros de busca, melhorias visuais e a oferta de um painel de controle para anunciantes.

Reconhece-se, contudo, que o sistema ainda possui limitações, especialmente por tratar-se de uma versão inicial (Prova de Conceito) focada em funcionalidades essenciais. Questões como integração com outros sistemas institucionais, moderação de postagens, políticas de privacidade e recursos de acessibilidade e a implementação de funcionalidades sugeridas por usuários devem ser exploradas em desenvolvimentos futuros.

Como trabalhos futuros, recomenda-se:

- Permitir que outros departamentos ou unidades acadêmicas postem suas vagas na plataforma;
- Adicionar novos filtros e critérios de busca;
- Criar um portal do anunciante para centralizar o cadastro e acompanhamento de anúncios;

- Criar um portal do moderador para facilitar o monitoramento, validação e moderação das postagens solicitadas;
- Criar um serviço de envio de notificações sobre novas vagas e personalizadas aos interesses de estudantes;
- Integrar a plataforma com sistemas institucionais de matrícula e autenticação;
- Realizar avaliações quantitativas e qualitativas com amostras maiores e métricas de uso ao longo do tempo;
- Desenvolver uma versão auto-hospedável em infraestrutura local própria da Universidade para usos em casos em que não é possível utilizar níveis públicos;
- Realizar estudos sobre a acessibilidade da plataforma para usuários com deficiências a fim de garantir que a plataforma seja acessível a todos;
- Permitir que empresas anunciem na plataforma;
- Definir uma política de privacidade.

Ademais, reconhece-se que o viés deste trabalho foi predominantemente técnico, com foco no desenvolvimento, implementação e validação de uma solução de software. Assim, recomenda-se que, após a implantação da solução, seja conduzida uma pesquisa que analise em profundidade o impacto organizacional da adoção da plataforma, isto é, como a ferramenta modificou as rotinas de estudantes e anunciantes e a interação entre eles e, por fim, os fluxos de trabalho na Universidade.

Em síntese, o presente trabalho contribuiu ao propor e validar uma solução prática, acessível e adaptada ao contexto universitário brasileiro para a divulgação e descoberta de oportunidades extracurriculares. Espera-se que a plataforma possa evoluir e ampliar seu impacto, facilitando o desenvolvimento acadêmico-profissional dos estudantes e a continuidade dos projetos de ensino, pesquisa e extensão.

## REFERÊNCIAS

Amazon Web Services. **Amazon API Gateway**. 2024. Acesso em: 22 jun. 2025. Disponível em: <https://aws.amazon.com/api-gateway>.

Amazon Web Services. **Amazon CloudFront**. 2024. Acesso em: 22 jun. 2025. Disponível em: <https://aws.amazon.com/cloudfront>.

Amazon Web Services. **Amazon RDS – Relational Database Service**. 2024. Acesso em: 22 jun. 2025. Disponível em: <https://aws.amazon.com/rds>.

Amazon Web Services. **Amazon Simple Email Service (SES)**. 2024. Acesso em: 22 jun. 2025. Disponível em: <https://aws.amazon.com/ses>.

Amazon Web Services. **Amazon Simple Notification Service (SNS)**. 2024. Acesso em: 22 jun. 2025. Disponível em: <https://aws.amazon.com/sns>.

Amazon Web Services. **Amazon Simple Storage Service (S3)**. 2024. Acesso em: 22 jun. 2025. Disponível em: <https://aws.amazon.com/s3>.

Amazon Web Services. **AWS Lambda**. 2024. Acesso em: 22 jun. 2025. Disponível em: <https://aws.amazon.com/lambda>.

Amazon Web Services. **Boto3 Documentation**. 2024. Acesso em: 22 jun. 2025. Disponível em: <https://boto3.amazonaws.com/v1/documentation/api/latest/index.html>.

BORBON, F. V. C.; MARTINS, I. R. L. **Extra \_ currículo: uma proposta de solução digital centrada no usuário para busca e gerenciamento de oportunidades extracurriculares na UFRJ**. Trabalho de Conclusão de Curso (Graduação), 2024. Orientadora: Adriana Santarosa Vivacqua; Coorientadora: Maria Luiza Machado Campos. Disponível em: <https://pantheon.ufrj.br/handle/11422/23613>.

COHN, M. **User Stories Applied: For Agile Software Development**. Boston: Addison-Wesley, 2004.

Conselho Nacional de Educação. Resolução, **Resolução CNE/CES nº 7, de 18 de dezembro de 2018**. 2018. [https://portal.mec.gov.br/index.php?Itemid=30192&alias=104251-rces007-18&category\\_slug=dezembro-2018-pdf&option=com\\_docman&view=download](https://portal.mec.gov.br/index.php?Itemid=30192&alias=104251-rces007-18&category_slug=dezembro-2018-pdf&option=com_docman&view=download). Estabelece as Diretrizes para a Extensão na Educação Superior Brasileira.

Ecma International. **ECMAScript Language Specification**. 2024. Acesso em: 22 jun. 2025. Disponível em: <https://tc39.es>.

INDEED. **Sobre o Indeed**. 2025. Acesso em: 24 maio 2025. Disponível em: <https://www.indeed.com/about>.

LINKEDIN. **Sobre o LinkedIn**. 2025. Acesso em: 24 maio 2025. Disponível em: <https://br.linkedin.com/about-us>.

Meta Platforms, Inc. **React – A JavaScript library for building user interfaces**. 2024. Acesso em: 22 jun. 2025. Disponível em: <https://react.dev>.

OpenAI. **ChatGPT**. 2025. Disponível em: <https://chat.openai.com/>. Acesso em: 25 ago. 2025.

Psycopg Community. **Psycopg – PostgreSQL database adapter for Python**. 2024. Acesso em: 22 jun. 2025. Disponível em: <https://www.psycopg.org>.

Python Software Foundation. **Python Programming Language**. 2024. Acesso em: 22 jun. 2025. Disponível em: <https://www.python.org>.

SCHWABER, K.; SUTHERLAND, J. **O Guia Scrum: o guia definitivo para o Scrum: as regras do jogo**. 2020. Disponível em: <https://scrumguides.org/docs/scrumguide/v2020/2020-Scrum-Guide-PortugueseBR-3.0.pdf>. Acesso em: 24 maio 2024.

TANENBAUM, A. S.; STEEN, M. V. **Distributed Systems**. 4.03. ed. Harlow: Pearson, 2025.

Universidade Federal do Rio de Janeiro. **O que é extensão**. s.d. <https://xn--extenso-2wa.ufrj.br/index.php/o-que-e-extenso>. Acesso em: 21 jun. 2025.

## APÊNDICE A – QUESTIONÁRIOS APLICADOS

### QUESTIONÁRIO A: CANDIDATOS A VAGAS

1. Com que frequência você tem interesse em saber sobre novas oportunidades ou vagas disponíveis na universidade?

- Sempre
- Frequentemente
- Eventualmente
- Raramente
- Nunca

2. Com que frequência seu desejo de saber sobre oportunidades ou vagas disponíveis é motivado por um objetivo pessoal de se candidatar a alguma delas?

- Sempre
- Frequentemente
- Eventualmente
- Raramente
- Nunca

3. Como você fica ou ficou sabendo das vagas disponíveis?

*Marque todas as alternativas que se aplicam.*

- Portal de vagas (ex: Portal da Extensão da UFRJ)
- Site da equipe ou instituição responsável à vaga (ex: site do grupo de extensão)
- Convite direto feito pelos responsáveis pela vaga
- Contato direto: você contactou os responsáveis para saber se havia alguma disponível
- Anúncio em redes sociais (ex: Instagram, Twitter, YouTube, Facebook)
- Anúncio por email (ex: mensagem do SIGA)
- Anúncio por mensagem instantânea (ex: grupo de WhatsApp ou Telegram)
- Anúncio físico (ex: pôster, banner, faixa)
- Eventos ou apresentações
- Plataforma de ensino (ex: Google Classroom, AVA, Moodle)

- Terceiros (ex: colegas, professores, familiares)
  - Agências de notícias (ex: Conexão UFRJ)
  - Administração do seu curso (ex: Coordenação, Direção)
  - Não fiquei sabendo ou não encontrei nenhuma
  - Não busquei por vagas
  - Outros...
4. Normalmente, você tem interesse em descobrir novas vagas em quais tipos de atividades?
- Marque todas as alternativas que se aplicam.*
- Ação de extensão (ex: grupo, ou curso de extensão)
  - Empresa júnior
  - Trabalho voluntário
  - Estágio
  - Oportunidade profissional
  - Freelance ou Oportunidade profissional temporária
  - Atividades complementares
  - Cursos ou apresentações
  - Participação em eventos como ouvinte
  - Participação em eventos como palestrante, staff ou organizador
  - Representação estudantil em colegiados administrativos ou acadêmicos
  - Representação estudantil (ex: vaga em chapa para centro acadêmico, DCE)
  - Iniciação científica
  - Monitoria
  - Pesquisa
  - Mestrado
  - Doutorado
  - TCC
  - Orientação acadêmica
  - Outros...
5. O quão relevantes as vagas disponíveis que descobre ou descobriu são relevantes para você?
- Responda com uma nota de 1 a 5.*

1 2 3 4 5

Nem um pouco do tipo que busco

Exatamente do tipo que busco

6. O quanto fácil é ou foi o processo de encontrar vagas disponíveis?

*Responda com uma nota de 1 a 5.*

1 2 3 4 5

Nem um pouco fácil

## Extremamente fácil

7. Quais foram as dificuldades que você percebeu no processo de descoberta de vagas?

*Se não encontrou nenhuma dificuldade, responda “nenhuma”. Se não quer responder agora, responda “não”.*

**Tipo de resposta:** texto de resposta longa.

8. Na sua opinião, o que uma ferramenta poderia oferecer para facilitar esse processo de descoberta de vagas?

**Tipo de resposta:** texto de resposta longa.

## QUESTIONÁRIO B: ANUNCIANTES DE VAGAS

1. Com que frequência você divulga vagas para universitários?

- Sempre
  - Frequentemente
  - Eventualmente
  - Raramente
  - Nunca

2. Qual tipo de vagas você anuncia ou anunciou?

*Marque todas as alternativas que se aplicam.*

- Ação de extensão (ex: grupo, ou curso de extensão)
  - Empresa júnior
  - Trabalho voluntário
  - Estágio
  - Oportunidade profissional
  - Freelance ou Oportunidade profissional temporária
  - Atividades complementares
  - Cursos ou apresentações

- Participação em eventos como ouvinte
- Participação em eventos como palestrante, staff ou organizador
- Representação estudantil em colegiados administrativos ou acadêmicos
- Representação estudantil (ex: vaga em chapa para centro acadêmico, DCE)
- Iniciação científica
- Monitoria
- Pesquisa
- Mestrado
- Doutorado
- TCC
- Outros...

3. Qual é o seu papel em relação às vagas que divulga ou divulgou?

*Marque todas as alternativas que se aplicam.*

- Intermediário: divulga vagas a pedido de terceiros
- Responsável: divulga vagas nas quais sou responsável de alguma forma
- Outros...

4. Por qual meio você faz ou fez a divulgação de vagas para universitários?

*Marque todas as alternativas que se aplicam.*

- Organizações estudantis (ex: Centro Acadêmico, DCE)
- Lista de transmissão de email (ex: SIGA, Google Groups)
- Contato direto e individual com possíveis interessados
- Redes sociais (ex: Instagram, Youtube, TikTok, Twitter, LinkedIn)
- Lista de transmissão, canais de grupos ou aplicativos de mensagens instantâneas (ex: WhatsApp, Telegram)
- Portais de vagas (ex: Portal da Extensão, Gadzebot, Indeed, Vagas.com.br)
- Agência de notícias (ex: Conexão UFRJ, Extensão UFRJ)
- Terceiros: você pede que algum intermediário divulgue para você
- Coordenação/Direção de cursos ou instituições relevantes
- Anúncio físico (ex: Pôster, banner, faixa)
- Eventos ou apresentações
- Outros...

5. O quanto confiante você está de que as estratégias de divulgação de vagas que utiliza atingem o público desejado como esperado?

*Responda com uma nota de 1 a 5.*

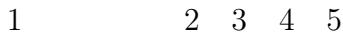


Nem um pouco confiante

Extremamente confiante

6. O quanto fácil é o processo de divulgação de vagas para universitários?

*Responda com uma nota de 1 a 5.*



Nem um pouco fácil

## Extremamente fácil

7. Quais foram as dificuldades que você percebeu no processo de divulgação de vagas?

*Se não encontrou nenhuma dificuldade, responda “nenhuma”. Se não quer responder agora, responda “não”.*

**Tipo de resposta:** texto de resposta longa.

8. Na sua opinião, o que uma ferramenta poderia oferecer para facilitar esse processo de divulgação de vagas?

**Tipo de resposta:** texto de resposta longa.

## APÊNDICE B – CAPTURAS DE TELA DA SOLUÇÃO DESENVOLVIDA

Este apêndice apresenta capturas de tela da aplicação *IC Carreiras*, desenvolvida como parte deste trabalho. As imagens destacam as principais funcionalidades implementadas, incluindo o mecanismo de visualização das vagas, a filtragem por múltiplos critérios, busca por palavra-chave, a paginação de resultados e o formulário de cadastro de novas oportunidades.

### B.1 VISUALIZAÇÃO E FILTRAGEM DE VAGAS

Figura 6 – Página inicial da aplicação com listagem de vagas.

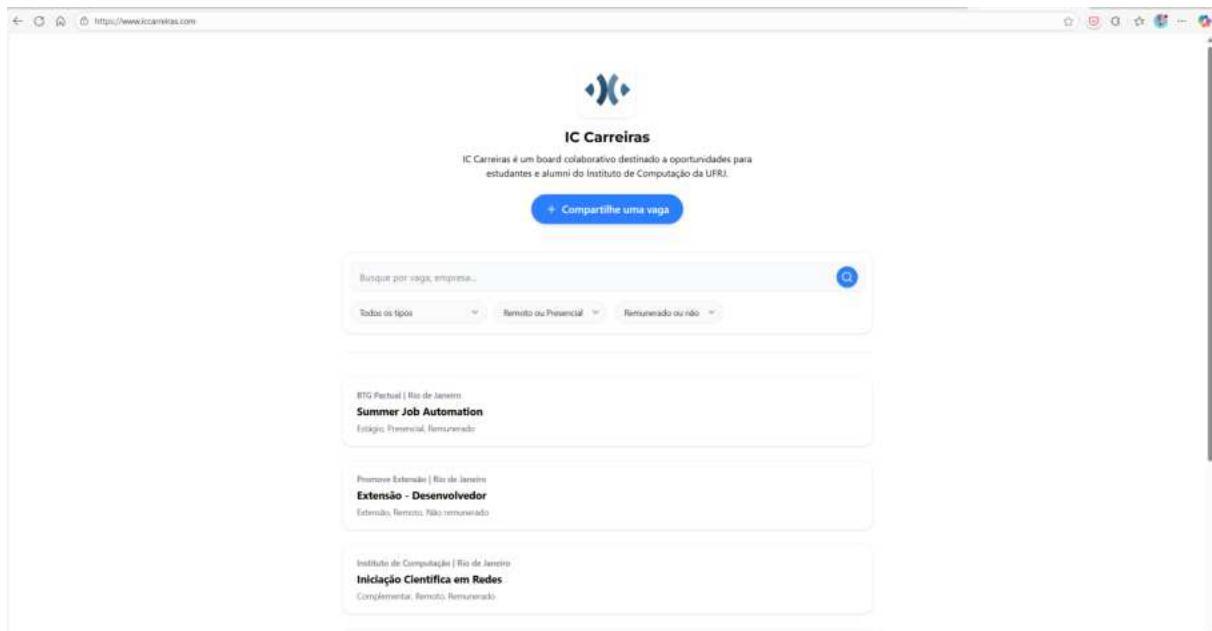


Figura 7 – Página inicial da aplicação com filtragem por tipo de vaga.

The screenshot shows the IC Carreiras homepage. At the top, there is a navigation bar with icons for back, forward, search, and other browser functions. The main header is 'IC Carreiras' with a logo of three blue arrows pointing outwards. Below the header, a sub-header states: 'IC Carreiras é um board colaborativo destinado a oportunidades para estudantes e alumni do Instituto de Computação da UFRJ.' A blue button labeled '+ Compartilhe uma vaga' is visible. The main content area features a search bar with placeholder text 'Busque por vaga, empresa...' and a magnifying glass icon. Below the search bar are three dropdown filters: 'Todos os tipos' (selected), 'Remoto ou Presencial' (selected), and 'Remunerado ou não' (selected). A list of job types is shown: 'Todos os tipos', 'Estendida', 'Complementar', 'Estágio', 'Trabalho em Tempo Integral', 'Trabalho em Tempo Parcial', and 'Pós-graduação'. Below these filters, there are two sections of job listings. The first section is for 'Premissas Extensão | Rio de Janeiro' with a listing for 'Extensão - Desenvolvedor' (Extensão, Remoto, Não remunerado). The second section is for 'Instituto de Computação | Rio de Janeiro' with a listing for 'Iniciação Científica em Redes' (Complementar, Remoto, Remunerado). At the bottom of the page, there is a footer with the text '© 2015 IC Carreiras. Desenvolvido por Inácio Pedro Silveira.'

Figura 8 – Página inicial da aplicação com busca por palavra-chave.

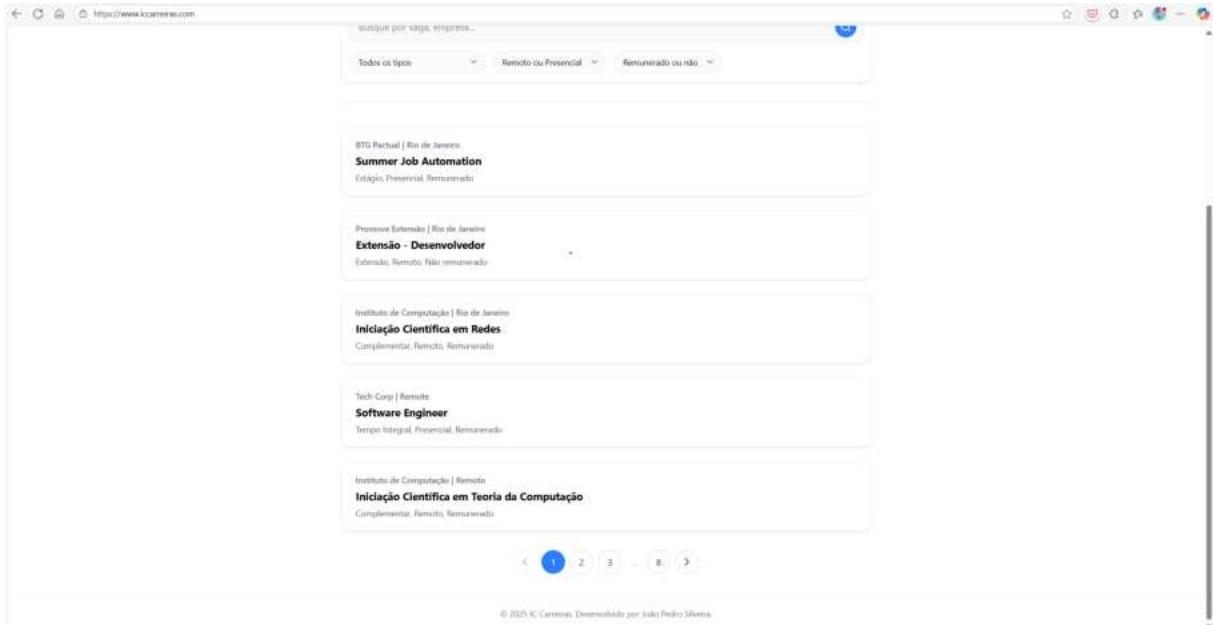
The screenshot shows the IC Carreiras homepage with a search query 'Iniciação' entered into the search bar. The search results are displayed in two sections. The first section is for 'Instituto de Computação | Rio de Janeiro' with a listing for 'Iniciação Científica em Redes' (Complementar, Remoto, Remunerado). The second section is for 'Instituto de Computação | Remoto' with a listing for 'Iniciação Científica em Teoria da Computação' (Complementar, Remoto, Remunerado). The page layout is similar to Figure 7, with the search bar, filters, and footer visible.

A página inicial apresenta, de forma clara e organizada, as oportunidades disponíveis para estudantes e alumni do Instituto de Computação da UFRJ. No topo da listagem, há um conjunto de filtros que permitem ao usuário refinar os resultados de acordo com tipo de vaga (por exemplo, estágio, extensão ou iniciação científica), modalidade de trabalho (presencial, remoto ou híbrido) ou se a vaga oferece alguma remuneração. Além disso,

há um campo de busca por palavras-chave, que possibilita localizar vagas por termos presentes no título ou na descrição.

## B.2 PAGINAÇÃO DE RESULTADOS

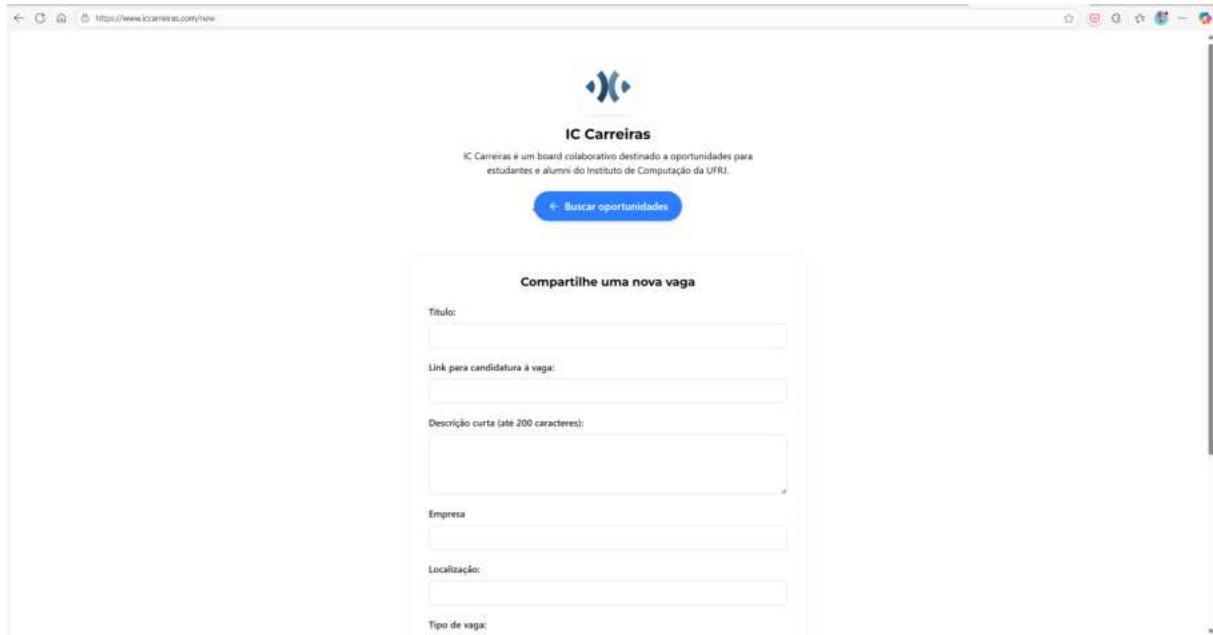
Figura 9 – Paginação da listagem de vagas.



Para otimizar a experiência do usuário e reduzir o tempo de carregamento, a listagem de vagas utiliza paginação. Os controles de navegação, localizados ao final da página, permitem avançar ou retroceder entre conjuntos de resultados, mantendo os filtros e a busca textual aplicados. Essa abordagem evita a sobrecarga de dados na interface e garante que apenas um número limitado de registros seja carregado por requisição.

### B.3 CADASTRO DE NOVAS VAGAS

Figura 10 – Formulário para cadastro de novas vagas.



The screenshot shows a web browser displaying the IC Carreiras website at <https://www.iccarreiras.com/new>. The page features a logo with three blue dots arranged in a triangle. Below the logo, the text "IC Carreiras" is displayed, followed by a subtitle: "IC Carreiras é um board colaborativo destinado a oportunidades para estudantes e alunos do Instituto de Computação da UFRJ." A blue button labeled "Buscar oportunidades" (Search opportunities) is visible. The main content is a form titled "Compartilhe uma nova vaga" (Share a new job). The form fields are: "Título:" (Title), "Link para candidatura à vaga:" (Link to apply for the job), "Descrição curta (até 200 caracteres):" (Short description (up to 200 characters)), "Empresa:" (Company), "Localização:" (Location), and "Tipo de vaga:" (Job type). The form is contained within a white box with a thin gray border.

O sistema inclui um formulário simples e intuitivo para submissão de novas oportunidades. Os campos incluem título da vaga, link para candidatura, descrição curta, nome da empresa, localização e tipo de vaga. Ao ser submetido, aciona uma função no backend que armazena a vaga no banco de dados e aciona, de forma assíncrona, o serviço de notificação por e-mail ao anunciante.