

ESCOLA POLITÉCNICA DA UNIVERSIDADE DE SÃO PAULO

CHRISTIAN BRAGA KEDOR
DANIEL LAVEDONIO DE LIMA

Portal de Estágios PCS: Utilizando uma interface web integrada com
WhatsApp para aproximar estudante, empresa e universidade

São Paulo
2020

CHRISTIAN BRAGA KEDOR
DANIEL LAVEDONIO DE LIMA

Portal de Estágios PCS: Utilizando uma interface web integrada com
WhatsApp para aproximar estudante, empresa e universidade

Trabalho de Formatura apresentado à
Escola Politécnica da Universidade de São
Paulo para obtenção do título de bacharel
em Engenharia Elétrica

Orientador:
Prof. Dr. Reginaldo Arakaki

São Paulo
2020

Dedicamos a todos os alunos que passaram pelo árduo processo da graduação.

AGRADECIMENTOS

Aos nossos pais, que nos apoiaram e nos acompanharam durante toda essa jornada.

Ao Professor Reginaldo Arakaki, pela orientação e participação ativa em todas as etapas deste trabalho.

Ao amigo Gabriel da Cunha, pela contribuição no projeto na disciplina de Laboratório de Engenharia de Software II, pela participação em entrevista e por outras contribuições indiretas, as quais foram fundamentais para a aplicação desse projeto.

À amiga Thábata Pontes, também pela contribuição no projeto na disciplina de Laboratório de Engenharia de Software II.

Ao Professor Jorge Risco, que apoiou e participou ativamente do processo, contribuindo com sua experiência, sem a qual possivelmente teríamos um projeto bem diferente do atual.

Ao amigo César Novas, que se prontificou em participar da entrevista que nos ajudou a entender pontos cruciais para o projeto que não havíamos enxergado.

À amiga Letícia Bonella, pela participação na entrevista.

À profissional Ana Paula Neri, que se prontificou em ajudar e que contribuiu em muito com sua visão do mundo de RH para a seção de empresas do portal.

A todos que responderam a pesquisa para ajudar a solidificar o entendimento do público geral de estudantes.

A todos que contribuíram com a comunidade *open-source*, da qual usufruímos em muito seus frutos.

RESUMO

Neste trabalho, desejou-se utilizar toda a gama de conhecimentos adquiridos no decorrer da graduação para realizar um projeto que desse um retorno para a universidade e para os alunos. Foi escolhido reestruturar o Portal de Estágios do PCS, dado que o sistema atual encontra-se defasado e não atende todas as necessidades dos alunos, das empresas e da universidade. Os autores possuem experiência e entendem o que falta para melhorar o processo de estágio do ponto de vista do aluno. Após pesquisas e entrevistas com alunos, representantes de empresas e da faculdade, foram levantados os requisitos necessários e chegou-se a uma solução, a qual foi planejada e executada para satisfazê-los.

O projeto visa aproximar a empresa do aluno por meio de uma plataforma web para para divulgação de vagas, visualização de currículos e outras funcionalidades que melhorem a experiência do processo de estágio na Poli. O sistema foi integrado com um módulo de envio de mensagens automáticas via WhatsApp para fazer um contato com os alunos por um meio mais direto e moderno do que o envio de e-mails.

A arquitetura foi feita de maneira a permitir evoluções do projeto, como a adição de algoritmos inteligentes de recomendações e filtros utilizando os dados do usuário com Machine Learning, interações de busca modernas utilizando processamento de linguagem natural, unificação da plataforma para cuidar também de processos burocráticos e expansão para outras universidades e institutos.

Este documento relata o processo de concepção e implementação do sistema e mostra quais são os próximos passos para melhorar todo o processo de estágios para todas as partes.

Palavras-chave: UI/UX. WhatsApp. Portal de Estágios. Sistema Web. Engenharia de Software.

ABSTRACT

In this work, it was desired to use the full range of knowledge acquired during graduation to carry out a project that would give a return to the university and to the students. It was chosen to restructure the PCS Internship Portal, as the current system is out of date and does not meet all the needs from students, companies and the university. The authors have the experience and understanding of what is needed to improve the internship process from the student's point of view. After surveys and interviews with students, company representatives and college representatives, the necessary requirements were raised and a solution was reached, which was planned and implemented to satisfy them.

The project aims to bring companies closer with the students through a web platform for posting job opportunities, viewing *curricula* and other features that improve the internship process experience at Poli. The system was integrated with a module for sending automatic messages via WhatsApp in order to make contact with students in a more direct and modern way than sending e-mails.

The architecture was made in order to allow the project to evolve, such as the addition of intelligent recommendation algorithms and filters using the user data with Machine Learning, modern search interactions using natural language processing, unification of the platform to also take care of bureaucratic processes and expansion to other universities and institutes.

This document reports on the system design and implementation process and shows what are the next steps to improve the entire internship process for all parties.

Keywords: UI/UX. WhatsApp. Internship Portal. Web System. Software Engineering.

SUMÁRIO

1. INTRODUÇÃO	9
1.1. Objetivo	9
1.2. Motivação	10
1.3. Justificativa	12
1.3.1 Aspectos funcionais	12
1.3.2 Aspectos técnicos	12
2. ASPECTOS CONCEITUAIS^[2]	14
2.1. UX - User experience	14
2.2. Arquitetura web em camadas ^[3]	14
2.3. Automação Web	15
3. TECNOLOGIAS UTILIZADAS	16
3.1. Linguagem de Programação: Python (back-end)	16
3.2. Framework web: Django (back-end)	16
3.3. Linguagens: HTML/CSS e JavaScript (front-end)	17
3.4. Framework web: Bootstrap 4 (front-end)	17
3.5. Framework web: jQuery (front-end)	17
3.6. Banco de Dados PostgreSQL (banco de dados relacional)	17
3.7. Git e GitHub (controle de versionamento)	18
3.8. Framework web: Selenium (back-end)	18
3.9. Add-on Django: Django AllAuth (back-end)	18
3.10. Serviço de Hospedagem: Heroku (PaaS)	19
4. METODOLOGIA DO TRABALHO	20
4.1. Concepção	20
4.2. Projeto	21
4.3. Implementação	21
4.4. Testes	22
5. ESTUDO DE USUÁRIO	23
5.1. Entrevistas	23
5.1.1. Metodologia	23
5.1.2. Entrevista 1: Estudante e gerente do Workshop Integrativo ^[5]	24
5.1.3. Entrevista 2: Prof. Dr. Jorge Luis Risco Becerra ^[7]	26
5.1.4. Entrevista 3: Aluno do PCS, curso semestral ^[8]	27
5.1.5. Entrevista 4: Aluna do curso de Engenharia Elétrica (PEA) ^[9]	28
5.1.6. Entrevista 5: Funcionária do RH de uma empresa ^[10]	29
5.2. Pesquisa	30
5.2.1. Perguntas	30
5.2.2. Respostas	31

6. ESPECIFICAÇÃO DE REQUISITOS DO SISTEMA	39
6.1. Atores	39
6.1.1. <i>Aluno</i>	39
6.1.2. <i>Representante da Empresa</i>	39
6.1.3. <i>Funcionário da POLI</i>	39
6.2. Casos de Uso	39
6.2.1. <i>Cadastrar Empresa no Sistema</i>	39
6.2.2. <i>Postar Vaga de Estágio</i>	40
6.2.3. <i>Cadastrar Aluno no Sistema</i>	41
6.3. Funcionalidades	41
6.3.1. <i>Geral</i>	42
6.3.2. <i>Empresas</i>	42
6.3.3. <i>Alunos</i>	43
6.3.4. <i>Funcionário POLI - Estágios</i>	44
6.4. Diagrama Entidade-Relacionamento	44
6.4.1. <i>App users</i>	46
6.4.2. <i>App companies</i>	47
6.4.3. <i>App experience</i>	50
6.4.4. <i>App education</i>	51
6.4.5. <i>App locations</i>	52
6.4.6. <i>App events</i>	53
6.4.7. <i>App backoffice</i>	54
6.4.8. <i>Relações entre apps</i>	56
7. PROJETO E IMPLEMENTAÇÃO	58
7.1. Administração do Django	60
7.2. Página inicial e Sessões Públicas	60
7.2.1. <i>Página Inicial</i>	61
7.2.2. <i>Página pública “Sou Aluno”</i>	61
7.2.3. <i>Página pública “Sou Empresa”</i>	62
7.2.4. <i>Página pública “Contato”</i>	62
7.3. Login/Sign-Up por meio da conta Google	63
7.4. Cadastro de usuários	66
7.5. Alteração assíncrona de estado no back-end por meio de AJAX	70
7.6. Dashboard do Aluno	70
7.6.1. <i>Dashboard do Aluno - Home</i>	71
7.6.2. <i>Dashboard do Aluno - Vagas</i>	72
7.6.3. <i>Outros</i>	73
7.7. Dashboard da Empresa	74
7.7.1. <i>Dashboard da Empresa - Home</i>	74

7.8. Dashboard Administrativo	75
7.9. Módulo de Mensagens WhatsApp	76
7.9.1. <i>Classes WhatsAppBot e protótipos das funções</i>	77
7.9.2. <i>Uso do módulo WhatsApp</i>	78
7.9.3. <i>Limitações</i>	78
7.10. Integração do Sistema com o WhatsApp	79
7.10.1. <i>Funcionamento das Background-Tasks</i>	79
7.10.2. <i>Task - Envio de nova vaga de estágio</i>	80
8. TESTES E AVALIAÇÃO	82
9. CONSIDERAÇÕES FINAIS	83
9.1. Conclusões do Projeto de Formatura	83
9.2. Perspectivas de Continuidade	83
9.2.1. <i>API WhatsApp e extensão do uso</i>	83
9.2.2. <i>Documentação e Procedimentos Burocráticos</i>	84
9.2.3. <i>Eventos e Notificações</i>	84
9.2.4. <i>Entidades Acadêmicas</i>	84
9.2.5. <i>Utilização inteligente dos dados</i>	85
9.2.6. <i>DialogFlow</i>	86
REFERÊNCIAS	87
GLOSSÁRIO	89
APÊNDICES	90
APÊNDICE A - Diagrama de modelo relacional	90

1. INTRODUÇÃO

1.1. Objetivo

Este trabalho tem como objetivo repensar e implementar um novo portal de estágios para o Departamento de Engenharia de Computação e Sistemas Digitais (PCS) da Escola Politécnica da Universidade de São Paulo (Poli-USP). Este sistema deve atender as necessidades dos alunos, das empresas e dos departamentos, com a finalidade de sanar os problemas existentes e dar uma visão à altura do que este departamento e Escola representam.

Com este trabalho pretende-se também buscar soluções inovadoras para facilitar uma interação atualmente burocrática, a partir de uma interface web simples, intuitiva e eficaz e divulgação de vagas de estágio por WhatsApp diretamente para o aluno.

1.2. Motivação

A plataforma vigente^[1] é antiga e já não supre todas as necessidades dos alunos e empresas e possui falhas técnicas. Além disso, existem muitas funcionalidades que uma plataforma como essa pode ter que seriam úteis tanto para os alunos como para as empresas.

Atualmente as vagas de estágio são divulgadas por meio de e-mails não-padronizados da seção de estágio para o e-mail USP dos alunos. Muitas vezes, estes se perdem em meio a outros comunicados oficiais da faculdade e notificações de plataformas de ensino como o E-Disciplinas. Não existe uma listagem das vagas vigentes, de forma que o aluno possa fazer uma consulta e ver o perfil da empresa e acessar link de contato.

Outro aspecto a ser considerado é que os alunos não possuem um local com relatos e experiências de outros alunos que passaram por determinada empresa, algo que muitas vezes ajuda na escolha.

Já as empresas, muitas vezes, não conseguem contato com o perfil de aluno que deseja, acarretando em possíveis boas oportunidades perdidas por falta de divulgação ou de informações sobre vagas. Também é comum o caso em que o aluno tem uma experiência ruim na empresa, tanto por conta de estar em uma vaga não compatível com seu perfil quanto por não receber o devido valor na condição de potencial talento. Além disso, no portal atual as empresas não possuem acesso aos currículos dos alunos, o que pode ser vantajoso para ambas as partes.

Outro aspecto a ser melhorado é o processo de contratação em si, uma vez que o aluno consegue a vaga. O site pode facilitar a lidar com a burocracia e tornar o processo mais fácil e rápido tanto para a empresa quanto para o aluno, bem como esclarecer dúvidas sem a necessidade de contato direto com a secretaria de estágio.

Um portal bem elaborado pensado para todas as partes tem um grande potencial para corrigir essas deficiências e oferecer uma grande experiência para empresas e possivelmente melhorar o futuro profissional do aluno.

1.3. Justificativa

1.3.1 Aspectos funcionais

O atual portal de estágios não oferece as funcionalidades que o aluno e a empresa precisam para se conectar:

- Uma plataforma oficial de divulgação de vagas
- Listagem de vagas de estágio correntes
- Visualização de perfil das empresas com informações relevantes
- Upload de descrição e currículo do aluno para o aluno ter visibilidade
- Procurar vagas de acordo com o perfil e habilidades desejadas

A empresa não consegue um acesso direto ao aluno e o aluno não tem um lugar centralizado para buscar por vagas, mostrar seu currículo e apresentar sua experiência para as empresas. Os gestores da Poli têm dificuldade de redirecionar as vagas que vêm nos mais variados formatos e os e-mails de divulgação ficam perdidos em meio a comunicados oficiais da faculdade não relacionados ao estágio.

1.3.2 Aspectos técnicos

O portal de estágios atual encontra-se com diversos problemas. Primeiramente, imediatamente ao carregar o site, o visitante é recebido com um visível bloqueio automático do navegador do banner principal, que necessita de um plugin Adobe Flash Player, uma tecnologia defasada e hoje considerada insegura, além de possuir apenas uma versão disponível em HTTP em vez de HTTPS, como pode-se observar na **imagem 1**. Ao começar a usar o site, nota-se rapidamente que ele não é nem um pouco intuitivo, causando uma navegação frustrante e desnecessariamente complexa. Ao validar o estágio para a matéria de estágio obrigatório e, posteriormente, submeter os documentos de Relatório de Estágio e Avaliação do Gestor, o sistema não permite que todos os documentos sejam enviados, e gera problemas com a extração destes para o funcionário encarregado de verificação, causando atritos, desencontros e até situações desconfortáveis,

como pedir para a empresa re-enviar um documento mesmo depois do aluno ter se desligado dela.

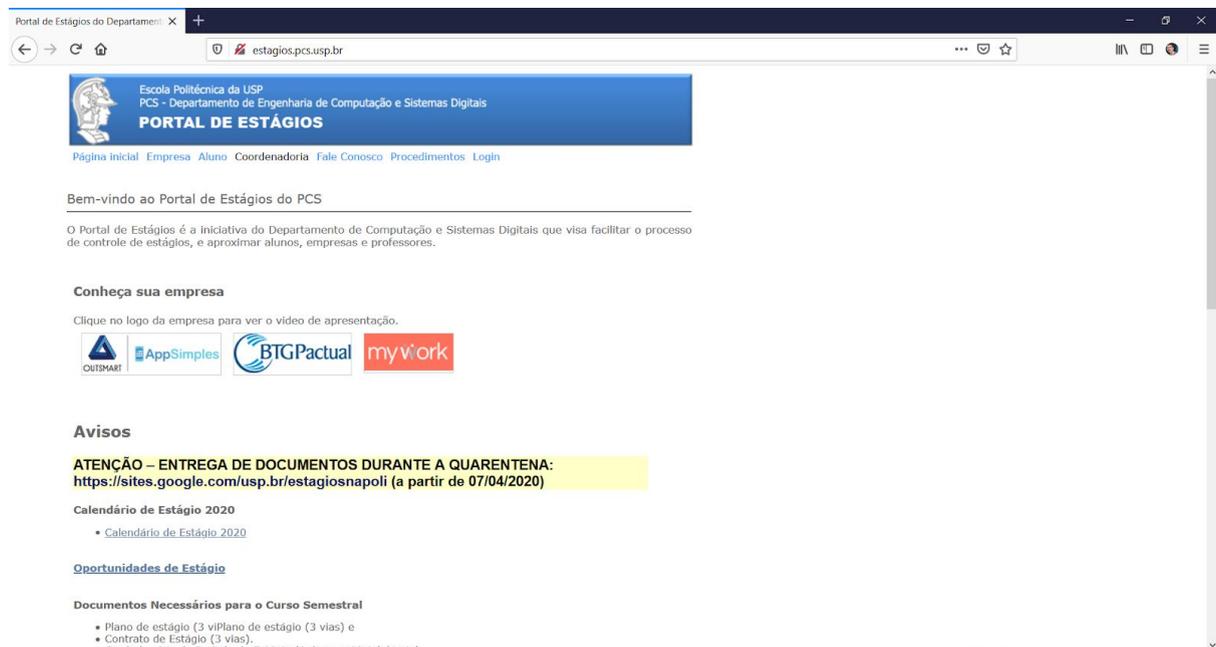


Imagem 1 - Página inicial do portal de estágios do PCS. URL: <http://estagios.pcs.usp.br/>

Acesso em: 14 abr. 2020

2. ASPECTOS CONCEITUAIS^[2]

2.1. UX - User experience

A UX é uma subárea da interação humano-computador que busca não só passar a mensagem corretamente para o usuário do sistema, mas torná-lo confortável com o uso e estimular, no caso tanto o aluno quanto o representante da empresa, a utilizarem o sistema. O UX não cria o valor do produto em si, mas realça e faz com que sua utilização seja mais acessível.

Esta é uma parte fundamental para garantir que o projeto seja de fato útil para todos os diferentes stakeholders.

2.2. Arquitetura web em camadas^[3]

O framework web utilizado utiliza o padrão de projeto MTV - (Model, View, Template). O *Model* é o componente que faz mapeamento do banco de dados para o projeto, o *Template* é responsável pela visualização de páginas (É onde o html é renderizado) e a *View* cuida das regras de negócio do projeto. A **imagem 2** mostra o funcionamento desta arquitetura.

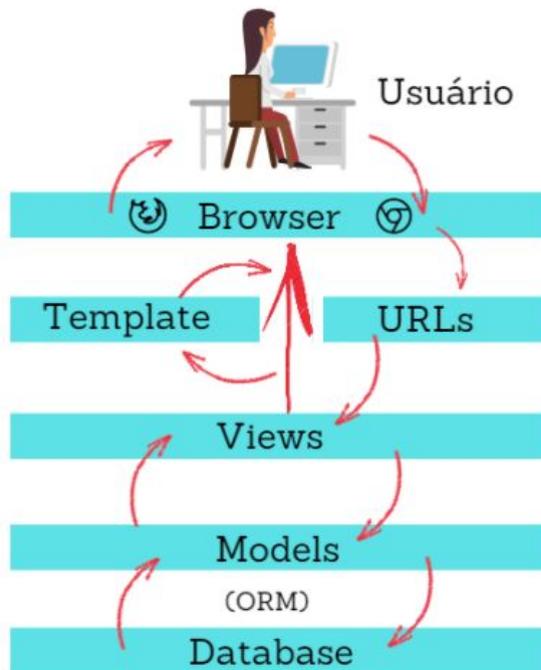


Imagem 2 - Diagrama da arquitetura do Django - fonte:

<https://www.treinaweb.com.br/blog/entendendo-o-mtv-do-django/> - Acesso em 01/12/2020

O usuário faz uma requisição no navegador e a View decide a partir das URLs qual vai ser o template utilizado e as variáveis de contexto. Essas variáveis são puxadas pelas *views* a partir dos *models* que implementam a interface com o banco de dados. Todos esses componentes estão interligados e funcionam junto no contexto da aplicação web.

2.3. Automação Web

O projeto também utilizará o conceito de *scripts* para automação *web*. Esses *scripts* automatizam tarefas pré-programadas em browsers acessando qualquer tipo de aplicação web. Podem ser utilizados para testes de aplicações, *web scraping* (coleta automática de dados) e no contexto deste sistema será utilizado para o envio de mensagens automáticas utilizando o WhatsApp Web.

O *script* de automação necessita de um *webdriver* para fazer a conexão com o navegador. No servidor o browser é executado no modo *headless*, que significa que ele não tem uma interface gráfica.

3. TECNOLOGIAS UTILIZADAS

3.1. Linguagem de Programação: Python (*back-end*)

A linguagem de programação Python é simples de ser implementada e possui um amplo suporte na comunidade *open source*. Também possibilita futuros usos de linguagem de máquina e processamento de linguagem natural com muitas bibliotecas com essa finalidade. O Python também possui suporte para a biblioteca Selenium, que será utilizada para o módulo de mensagens automáticas, discutido mais à frente.

Além disso é uma linguagem madura e de alto nível, que possui grande suporte e adoção, tornando mais fácil e mais duradoura a manutenção do sistema, ao contrário do sistema atual baseado em ASP.NET.

3.2. Framework web: Django (*back-end*)

Para o desenvolvimento do projeto optou-se pelo Django, um framework web Python de alto nível. Django possui código aberto, tem uma ampla comunidade ativa que produz *apps* que podem incrementar sua versatilidade e possui uma documentação robusta. Outros dois pontos fortes desse framework são o foco em segurança, com diversas camadas e travas que asseguram que o projeto final estará protegido contra os mais diversos tipos de ataques comuns na web (*SQL injection*, *cross-site scripting*, *cross-site request forgery*, entre outros), e sua escalabilidade, podendo resultar em um produto final que pode ser utilizado desde apenas a secretaria do PCS, até ser expandido para outros cursos da Poli, da USP ou até de outras universidades que desejam utilizar esse projeto, aproveitando de sua versatilidade.

Para completar, é um framework com grandes períodos de suporte e atualizações de segurança, facilitando sua manutenção.

Além disso, por ser escrito em Python, ele é bastante conveniente para o presente projeto, facilitando a integração do site com os módulos de processamento natural de linguagem (PLN) a serem implementados.

3.3. Linguagens: HTML/CSS e JavaScript (*front-end*)

Como o foco do projeto é produzir um sistema web, não há outra forma se não utilizar as linguagens de marcação HTML/CSS e a de programação JavaScript. Esta última é necessária não só para a interatividade dos elementos de *design* na tela, com o foco de melhorar a experiência do usuário, mas também para a comunicação com o servidor em funcionalidades que não permitem a comunicação padrão baseada em páginas do protocolo HTTP.

3.4. Framework web: Bootstrap 4 (*front-end*)

Para garantir que o requisito da UX seja atendido no projeto, é necessário tornar o código de *front-end* responsivo, a fim de se encaixar bem tanto nos *Desktops* (onde as empresas e os funcionários geralmente irão utilizar o sistema) quanto nos smartphones (onde os alunos geralmente visualizam essas informações).

3.5. Framework web: jQuery (*front-end*)

Em certas partes do sistema, para garantir que o requisito da UX seja atendido no projeto, foi necessário utilizar o jQuery, uma a framework JavaScript, para facilitar as requisições AJAX do projeto. Assim, é possível evitar o carregamento desnecessário de páginas ao fazer requisições assíncronas com o *back-end*.

3.6. Banco de Dados PostgreSQL (banco de dados relacional)

Para o banco de dados foi utilizado o banco PostgreSQL, uma alternativa madura e *open-source* com suas origens em um projeto da *University of California at Berkeley*. A escolha se baseia tanto no amplo suporte como na facilidade de manutenção. Bancos de dados no modelo No-SQL tendem a possuírem cada vez

mais registros não anilhados que podem vir a causar problemas caso haja manutenção que modifique a lógica do projeto; um banco de dados relacional força ao programador que for modificar o código que lide com essas questões no momento da mudança, mitigando esse tipo de problema.

3.7. Git e GitHub (controle de versionamento)

Para garantir que as modificações sejam salvas e exista um controle de versões coerente, foi usado o Git. Esta é a ferramenta mais eficiente e avançada para este propósito, minimizando o espaço ocupado e permitindo que múltiplos programadores possam trabalhar no projeto ao mesmo tempo, com um número reduzido de conflitos.

Para garantir que todos possuam uma maneira de comunicar os avanços no código e garantir que exista um *backup* na nuvem, o GitHub foi escolhido para guardar o código Git. A escolha deste serviço se baseia na possibilidade de gerar projetos privados e do amplo apoio da empresa para estudantes e acadêmicos, tornando-o ideal para o projeto.

3.8. Framework web: Selenium (*back-end*)

O intuito desta framework é criar um método automatizado de permitir com que o servidor utilize o WhatsApp, por meio do WhatsApp Web, usando-o no navegador Google Chrome.

Existe a possibilidade de fazer essa integração via API oficial do WhatsApp, porém ela possui um custo elevado que, dado o contexto, está fora do escopo financeiro do projeto. Com isso, o Selenium será usado para fazer a integração neste projeto.

3.9. Add-on Django: Django AllAuth (*back-end*)

Esse add-on é um app Django que permite conectar o método de autenticação de usuário padrão com autenticações externas de outras plataformas.

O framework Django utilizado permite a adição de diversos *apps* pré-prontos para as mais diversas funcionalidades. Em geral, optou-se por construir tudo in-house, a fim de diminuir as dependências do projeto em repositórios open-source menores que podem ter um suporte futuro rudimentar. Este, porém, é um caso em que usar um *app* pronto permite aumentar a confiabilidade do sistema, pois a maneira que os serviços de terceiros realizam a autenticação está sempre mudando e, utilizando este add-on, manter o processo funcionando torna-se uma questão de atualizá-la assim que for dado o suporte.

No caso, como o e-mail USP utiliza os serviços do G-Suite, foi utilizada também a autenticação via *oauth2* da Google. Com isso, apenas o aluno e administrador que possuem acesso ao e-mail acadêmico podem utilizá-lo para cadastrar na conta, garantindo mais segurança e comodidade para os usuários do sistema.

3.10. Serviço de Hospedagem: Heroku (*PaaS*)

Para a hospedagem do projeto, foi utilizado o Heroku, um serviço que combina diversas funcionalidades e facilita bastante o processo de deploy, principalmente durante o desenvolvimento do projeto.

4. METODOLOGIA DO TRABALHO

Dado que este é um projeto que visa a implementação em um ambiente real, faz-se necessário entender a necessidade de todos os *stakeholders* para então montar o projeto. Uma vez que este processo esteja concluído, visa-se o planejamento e a construção da aplicação web em si, para que possa ser realizado em paralelo com as outras atividades acadêmicas e profissionais dos alunos responsáveis. Por fim, planeja-se realizar testes para verificar a validade do que foi implementado.

Esses passos podem tanto avançar quanto regredir, dependendo do problema encontrado ou resolvido. Isso fica mais claro observando o diagrama da **imagem 2**, que se encontra a seguir:

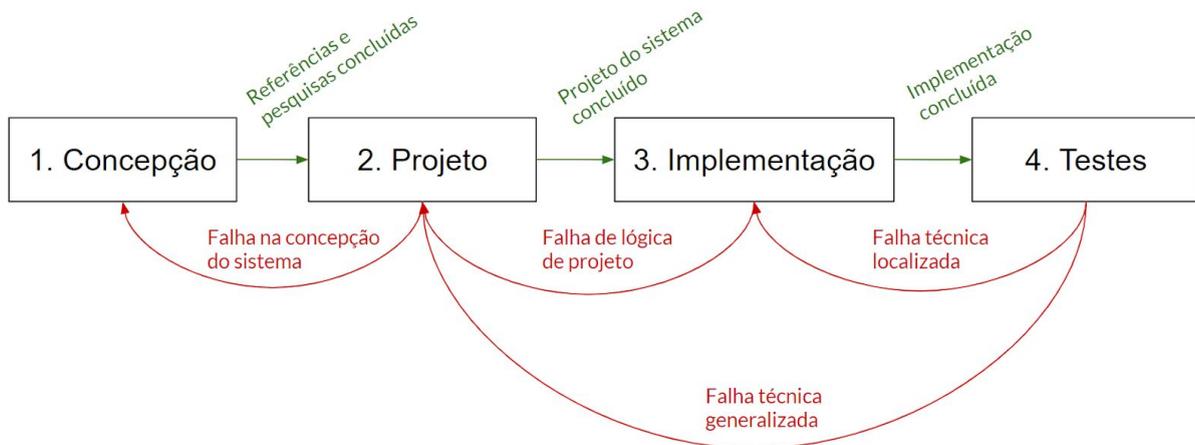


Imagem 3 - Diagrama da Metodologia do Trabalho

Cada etapa está descrita a seguir nos itens deste tópico.

4.1. Concepção

Dadas as motivações e objetivos iniciais, além do fato de que este é um projeto que visa a implementação em um ambiente real, fez-se necessário entender a necessidade de todos os *stakeholders* para então montar o projeto. Para isso, foram feitas diversas entrevistas com diferentes tipos de usuários e montou-se uma

pesquisa para entender melhor o problema identificado. Essas duas abordagens permitiram abordar a questão tanto de forma qualitativa quanto quantitativa (respectivamente). Estes estudos de usuário serão abordados no **item 5** do documento.

A partir disso e da experiência própria dos autores do projeto, foi elencado uma lista de funcionalidades que seriam necessárias para que o projeto atinja as dores dos usuários.

4.2. Projeto

Nesta fase foi feita a documentação e modelagem do projeto. Foram definidas todas as funcionalidades, requisitos funcionais e não-funcionais, criação de diagramas de casos de uso, diagramas de classes, protótipos de layout, diagramas da arquitetura e mapeamento de conteúdo para composição do site.

O resultado disso encontra-se no **item 6** do documento.

4.3. Implementação

Essa etapa é a escrita do código propriamente dito. Os integrantes da dupla irão implementar o projeto conforme a documentação especificada na fase de projeto e com base na tecnologia escolhida. Para implementação será utilizada a ferramenta GIT, com o repositório no site GitHub. Ao iniciar o desenvolvimento de uma funcionalidade, o integrante da dupla responsável irá criar um *branch* e realizar seus *commits* nele. Depois de testar a funcionalidade o código deverá ser *mergeado* com o *master*.

Nessa etapa manteremos um quadro com tarefas e sub-tarefas pendentes e quais estão sendo desenvolvidas e quem da dupla é o responsável. Dessa forma o controle e coordenação do desenvolvimento é facilitada.

Também é importante notar que, conforme o projeto é implementado, decisões de projeto podem ser modificadas e adaptadas com base em novas

necessidades, limitações de ferramentas, recomendações do professor orientador, etc.

O resultado disso encontra-se no **item 7** do documento.

4.4. Testes

Durante a 3ª fase (implementação) também serão realizados testes a cada nova funcionalidade, com testes unitários de funções e testes regressivos. Assim, a cada nova funcionalidade verificaremos se ocorrem erros e se a nova funcionalidade afetou as funcionalidades previamente implementadas. Esses testes serão documentados em uma lista conforme o desenvolvimento, de forma que ao final do projeto pode-se realizar um teste completo com todas as funcionalidades.

O resultado disso encontra-se no **item 8** do documento.

5. ESTUDO DE USUÁRIO

Nesta etapa os potenciais usuários dentro do público-alvo foram abordados de forma a captar informações a partir da ótica dos problemas que enfrentam.

Aqui, o foco foram os estudantes, algo que reflete no decorrer do projeto, por serem os principais usuários desse tipo de sistema. Isso se deve ao fato de que, se os alunos não verem suas necessidades sanadas pelo portal, eles não o utilizarão ou só utilizarão se forem forçados. Sem os estudantes, as empresas não têm motivo para divulgarem suas vagas no portal e o departamento irá manter um sistema essencialmente morto, tal como se dá hoje.

Portanto, a pesquisa foi direcionada apenas para as necessidades dos alunos, enquanto as entrevistas foram, apesar de focadas nos alunos, mais abrangentes e atingiram todos os *stakeholders*.

5.1. Entrevistas

Com as entrevistas, a ideia foi entender para além dos problemas pessoais vivenciados com o sistema atuante e ter diferentes visões de como ele afetava os mais diferentes *stakeholders*.

Foram realizadas 5 entrevistas ao todo, sendo 3 com estudantes, 1 com uma funcionária de RH e 1 com o Prof. Dr. Jorge Risco, responsável pelo processo administrativo da seção de estágios do PCS.

5.1.1. Metodologia

As entrevistas não tiveram um roteiro bem definido, apenas algumas perguntas base. A ideia era que a entrevista fosse em formato de conversa informal, para que o entrevistado se sentisse à vontade e que fosse instigado a desenvolver sobre o tema. Da conversa eram feitas as perguntas, de forma a guiá-lo para apontar os pontos positivos, negativos e o que alteraria no processo.

Para a entrevista com os estudantes, foi abordada a experiência com a busca de estágios: quanto tempo demorou a pesquisa, onde pesquisou e o que foi importante para decidir qual vaga escolher. Além disso, foram feitas perguntas complementares de acordo com as experiências vividas pelo entrevistado.

No caso especial do estudante que foi gerente do Workshop Integrativo, a entrevista teve um foco maior no sistema contratado para o evento que se relaciona muito bem com a proposta deste projeto.

Para a entrevista com o professor responsável pelos estágios no departamento, as perguntas foram focadas nas necessidades do departamento, os problemas que eles enfrentam com o sistema atual e quais seriam as funcionalidades mais importantes para se implementar. Dado que já tinham sido realizadas conversas informais com o professor a respeito do tópico, não foi necessário preparar perguntas previamente, visto que ele já tinha todas essas questões estruturadas.

Para a funcionária do RH, foram feitas perguntas a respeito de como funciona o processo de busca de um candidato, como funcionam as plataformas do ponto de vista do contratante e quais os principais problemas enfrentados. Como se trata de uma realidade distante dos entrevistadores, optou-se por deixar a entrevista mais aberta para que ela contasse o processo.

5.1.2. Entrevista 1: Estudante e gerente do Workshop Integrativo^[5]

A primeira entrevista foi realizada com o estudante César A. M. N., aluno do curso de Engenharia de Minas da Escola Politécnica da USP. Esta foi focada na sua experiência como gerente do 30º Workshop Integrativo (WI), bem como sua visão de um aluno da Poli de um curso que não é de responsabilidade do departamento PCS.

O WI é uma feira de recrutamento focada no público politécnico, mas aberta para outros cursos e faculdades em geral, cujo principal objetivo é aproximar empresas e estudantes. O evento possui um foco em divulgar não apenas os processos seletivos das empresas, mas também ajudar a divulgar a cultura das

empresas, tirar dúvidas dos estudantes a respeito das vagas e ajudar com a escolha de carreira do aluno, via palestras e stands das empresas.

Neste ano de 2020, devido à pandemia de COVID-19, foi necessário transformar o evento que ocorre há 20 anos no estacionamento da Poli em um evento virtual, que permitisse o mesmo tipo de interação. O resultado foi uma plataforma^[6] em que as empresas pudessem postar vagas e os alunos pudessem se candidatar nas mesmas (**imagem 3**). Segundo o César, esse sistema foi criado especialmente para o evento por uma empresa que o gerenciou durante o período do evento.

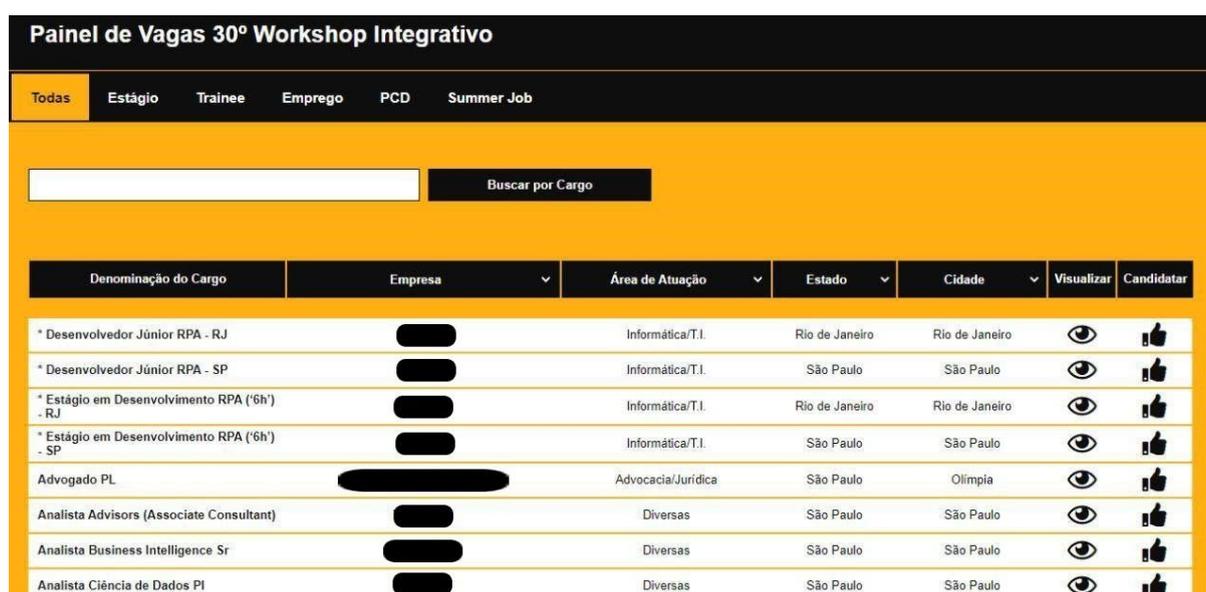


Imagem 4 - Printscreen do painel de vagas usado na 30ª edição do WI. Acesso: 3 set. 2020

Quando questionado sobre a recepção do público, o entrevistado contou que houve um *feedback* geral bem positivo a respeito do Painel de Vagas. Da parte dos estudantes, a facilidade de ter tudo em um só lugar auxiliou a busca muitas vezes exaustiva por um estágio, porém a falta de *summer jobs* foi sentida por muitos. Já da parte das empresas, a possibilidade de entrar em contato com o público politécnico e gerar um relatório de mailing list de todos os participantes foram os aspectos mais elogiados, enquanto o pouco engajamento do público politécnico em geral foi mais sentido.

Este sistema foi, de certa forma, uma prova de conceito da demanda dos alunos por uma plataforma do tipo, focada em estágios e que pudesse ajudar de forma mais constante a busca de estágio dos estudantes. Essa experiência mostra que, principalmente nessa época de pandemia, há uma demanda por um serviço que supra essa necessidade de encontrar um estágio.

5.1.3. Entrevista 2: Prof. Dr. Jorge Luis Risco Becerra^[7]

A segunda entrevista foi realizada com o Prof. Dr. Jorge Risco, responsável pelo processo administrativo da seção de estágios do PCS. Esta foi focada em entender qual a situação atual do Portal de Estágios e como ele é usado pela seção de estágios do PCS, além de buscar uma proposta de intervenção com um novo sistema para substituir o portal atual.

Entendendo melhor como funciona o processo de aprovação de estágios, existem três frentes que eles precisam lidar: o aluno, a empresa e a seção de estágios da Poli. O professor contou que a secretaria do PCS tem uma relação com a COC para analisar se o aluno precisa e/ou cumpre os requisitos para fazer estágio. A função do prof. Risco é certificar a informação e aprovar o processo. Com o setor de estágios, a função é redução de burocracia e diminuição do tempo de resposta. Já o Portal de Estágios atual existe para que tudo esteja alinhado com a empresa. Na pandemia, o setor de estágios criou um sistema bem simples para diminuir o fluxo de papel e saber se o aluno cumpre os requisitos, facilitando um pouco esse processo.

Uma vez mapeado esse processo, foi perguntado mais especificamente sobre o Portal de Estágios atual. Nele, o processo todo se inicia após as contratações, o que invalida o propósito original da plataforma de divulgação das vagas e torna o sistema apenas um passo burocrático para as disciplinas de estágio obrigatório. A divulgação de vagas é feita ou por email ou a partir dos representantes de classe em grupos das turmas no WhatsApp. Com isso, não há um acompanhamento do período de fato de estágio: da evolução do aluno nesse período (o que ele usa e não usa no estágio e o que disso ele aprendeu no curso) ou se a empresa está alinhada com a proposta do curso, principalmente no caso

dos cursos quadrimestrais. Sobre este último tópico, o professor Risco conta que precisa fazer várias reuniões com as empresas para que elas entendam como funciona o regime de estágio cooperativo.

A partir disso, o professor fez uma proposta de criar um sistema para ter um acompanhamento mais perto dos alunos e estágios, principalmente se estes estão alinhados com a proposta do curso quadrimestral, verificando a data de término e carga horária para certificar que a empresa está alinhada com o conceito. Para a divulgação de vagas, a comunicação por WhatsApp é vista como essencial, dado o alto engajamento dos alunos por este meio. Para isso foi sugerido um envio automático para as empresas com email de quem demonstra interesse em suas vagas. Para o professor, o sistema deveria gerar um relatório completo com as tecnologias que estão utilizando nos estágios atualmente, quantas contratações são feitas, o quão rápido o aluno é contratado, entre outras métricas importantes.

Em suma, o que se pode retirar desta reunião é que o objetivo do Portal de Estágios na visão administrativa é medir a evolução dos alunos, verificar os pontos que precisam melhorar e como a graduação precisa se ajustar para atingir as demandas do mercado.

5.1.4. Entrevista 3: Aluno do PCS, curso semestral^[8]

A terceira entrevista foi realizada com o estudante Gabriel C. R., aluno do último ano do curso de Engenharia Elétrica - Ênfase em Computação da Escola Politécnica da USP. Esta foi focada na sua visão de aluno do PCS e como público alvo principal do projeto.

Quando perguntado sobre como conseguiu o estágio atual, ele contou que ele já tinha prestado o processo seletivo 1 ano antes, mas escolheu outra empresa na época. Um tempo depois surgiu a oportunidade de utilizar esse processo seletivo para entrar em uma outra vaga que abriu na primeira empresa. Na época que estava buscando estágio, ele estruturou sua pesquisa para entender primeiro quais empresas e áreas gostaria de trabalhar, buscando palestras e sites de avaliação de empresas para ver se devia investir mais em cada uma. Ele também verificou as

vagas em aberto que tinham na época por meio de uma ferramenta do próprio Google, do LinkedIn, de portais genéricos de estágio e nos próprios sites das empresas.

Um ponto negativo do processo de busca de estágio foi que em muitas vagas o processo seletivo já tinha se encerrado e ele precisou gastar um tempo para filtrar essas que não lhe serviam mais. Ele comentou também que os métodos atuais de notificação de novas vagas por email não é eficiente, pois recebe vagas repetidas, de cursos que não o dele e não há forma fácil de filtrar que tipo de oportunidade é e quais habilidades são requisitadas, já que inclusive muitas vêm em um arquivo anexo ou em imagens.

Em suma, o processo de procura de estágio envolve diversas ferramentas e são usados diversos critérios, tais como qual o tipo de estágio e avaliações da empresa, e os métodos fornecidos atualmente pela faculdade não são suficientes para tal.

5.1.5. Entrevista 4: Aluna do curso de Engenharia Elétrica (PEA)^[9]

A quarta entrevista foi realizada com a estudante Letícia B. R., aluna do penúltimo ano do curso de Engenharia Elétrica - Ênfase em Energia e Automação Elétricas da Escola Politécnica da USP. Esta foi focada na sua visão de uma aluna próxima, mas não necessariamente o público alvo principal do projeto em si.

Seu primeiro estágio foi por indicação em uma área técnica e já foi entrevistada por quem seria sua gestora, em um processo muito rápido do começo ao fim. O segundo estágio, esse sim buscando ativamente, foi através de uma vaga publicada no LinkedIn, onde o processo seletivo foi um pouco mais longo (demorou 3 meses) e teve um projeto a ser desenvolvido no meio. Nesse último, ela também estava cadastrada em mailing-list do Poli Finance, buscava em portais de vagas, como o 99jobs, e acompanhava os emails enviados pela seção de estágio da Poli (apesar deste conter muitas vagas que não se encaixavam com o perfil do curso).

Para ela, a busca basicamente consiste em ver a descrição da vaga, se a interessa e se ela cumpre os requisitos, e busca ver quem dos conhecidos trabalha

naquela empresa, para perguntar sobre como é trabalhar lá. Em processos de empresas do ramo financeiro, principalmente, ela percebeu que grande parte não eram muito estruturados e começavam com o envio de email para uma pessoa específica da área.

Ela se interessou pela proposta do portal de estágios e gostaria que estivesse disponível para além de apenas o PCS.

5.1.6. Entrevista 5: Funcionária do RH de uma empresa^[10]

A quinta entrevista foi realizada com a Ana P. N., profissional de recursos humanos com 7 anos de experiência na área. Esta foi focada na sua visão de uma representante da empresa, que estaria interessada em divulgar vagas e selecionar estudantes para participar em processos seletivos.

A participação nesta entrevista partiu de uma curiosidade dela no tópico do projeto ao comentar sobre o novo portal de estágios que está sendo construído. Ela, assim como o entrevistador e aluno Daniel Lavedonio de Lima, fazem parte de uma empresa de recrutamento onde muitas empresas clientes enfrentam dificuldade de encontrar os estudantes qualificados para as vagas abertas.

Um grande problema do setor de RH é que diferentes empresas têm diferentes nomes para o mesmo tipo de cargo (ex: “customer success” e “gerente de relacionamento” são 2 nomes para o mesmo cargo). Outro problema é que diferentes graduações podem ser usadas para o mesmo tipo de vaga, dependendo do enfoque da empresa, dando um nível de complexidade maior na busca.

Outro fator é que, independentemente da graduação, as experiências profissionais podem variar muito e isso é o que mais conta para muitas empresas (ex: um cientista de dados pode vir de cursos como engenharia, matemática aplicada, estatística, física, etc...). Isso pode não ser um problema tão grande para estagiários, que possuem pouca ou nenhuma experiência profissional prévia, mas do lado contrário impede com que alguns alunos tenham a oportunidade de aparecer no radar dos recrutadores por conta do curso. Uma sugestão dela,

inclusive, foi dar um peso maior para o desejo de carreira do aluno do que o curso dele propriamente dito.

Por fim, ela deu algumas sugestões, como unificar o gênero dos nomes das carreiras, para evitar algum preconceito inerente do entrevistador, criar um filtro para o tipo de empresa que o aluno quer (de *startup* até multinacional) e, claro, convidá-la para testar o sistema assim que ele estiver pronto.

5.2. Pesquisa

Depois de entender qualitativamente, foi realizada uma pesquisa^[11] para entender quantitativamente a abrangência do problema com que foi percebido tanto pelos autores quanto pelos entrevistados.

5.2.1. Perguntas

A pesquisa foi dividida em 4 seções, cada uma contendo um conjunto de perguntas relacionadas com o tema. Nem todas as perguntas apareceram para todos os respondentes, dado que a seção 2 aparecia apenas para quem respondeu “sim” na pergunta “Você já procurou por estágios” enquanto a seção 3 apenas caso a resposta para a mesma pergunta fosse “não”.

A lista completa de perguntas separadas por seção, com as perguntas obrigatórias com asterisco, segue:

1. Informações básicas
 - a. Nome
 - b. Universidade*
 - c. Cidade onde estuda*
 - d. Curso*
 - e. Ano previsto de formatura*
 - f. Você já procurou por estágios?*
2. Histórico de busca de estágios

- a. Quanto tempo você demorou entre o começo da sua pesquisa e conseguir uma vaga? (Se já estagiou mais de uma vez, considere sua experiência mais recente.)*
 - b. Como você normalmente encontra vagas de estágio? (Marque todas as que se aplicam)*
 - c. Como você avalia se uma vaga ou empresa é um bom lugar para se trabalhar? (Marque todas que se aplicam)*
 - d. O que te faz desistir de uma vaga? (Marque todas que se aplicam)*
 - e. Se pudesse melhorar esse processo, o que você mudaria?*
3. Entendendo o comportamento do estudante
 - a. Por qual motivo você não procurou um estágio?*
4. Comentários finais
 - a. Há algo que não foi coberto pela pesquisa que deseja adicionar?
 - b. Alguma sugestão e/ou feedback sobre essa pesquisa?
 - c. Estamos desenvolvendo um sistema para melhorar a forma com que os estudantes procuram estágios. Se gostaria de receber um convite para testar esse sistema quando estiver pronto, coloque seu e-mail abaixo:

5.2.2. Respostas

Com a pesquisa aberta entre os dias 22/11/2020 às 16h07min e 02/12/2020 às 13h30min, foram obtidas 89 respostas. Destas, 3 respostas foram desconsideradas por se tratarem de dados enviados em duplicata, totalizando 86 respostas individuais.

Com os dados limpos, primeiramente foi traçado um recorte do público atingido:

Universidades

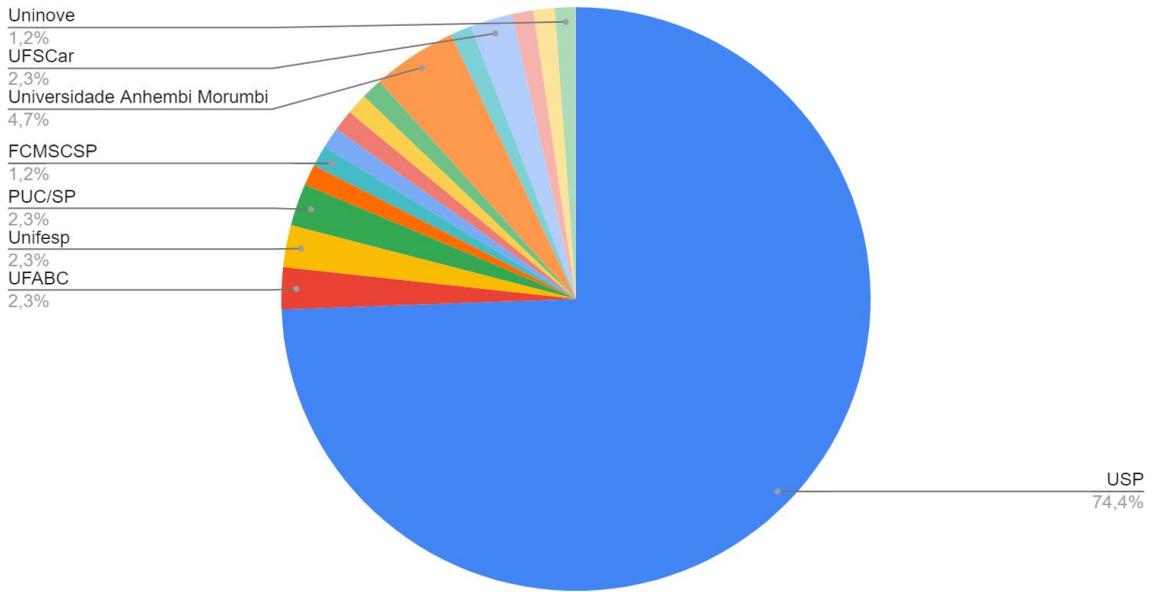


Imagem 5 - Respostas à pergunta 1b da pesquisa

Cidade onde estuda

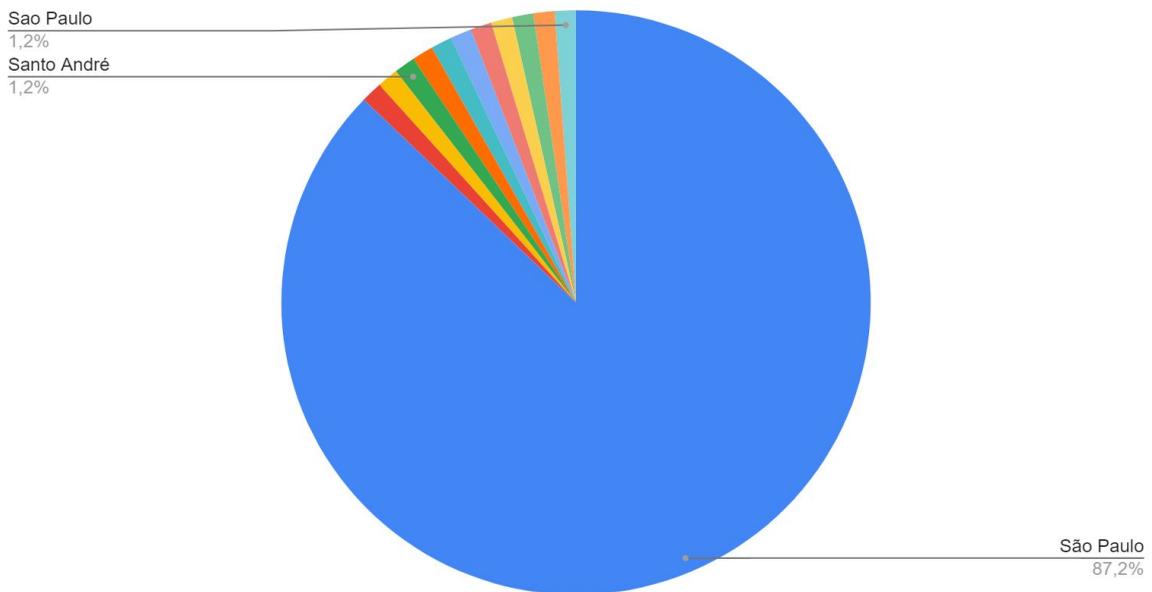


Imagem 6 - Respostas à pergunta 1c da pesquisa

Cursos

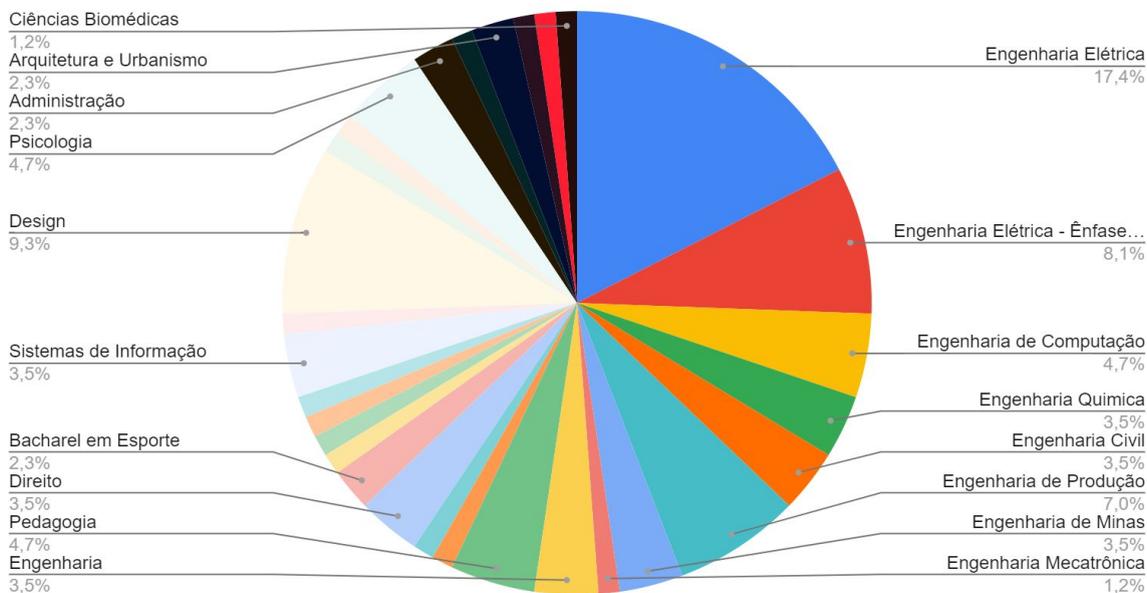


Imagem 7 - Respostas à pergunta 1d da pesquisa

Ano previsto de formatura

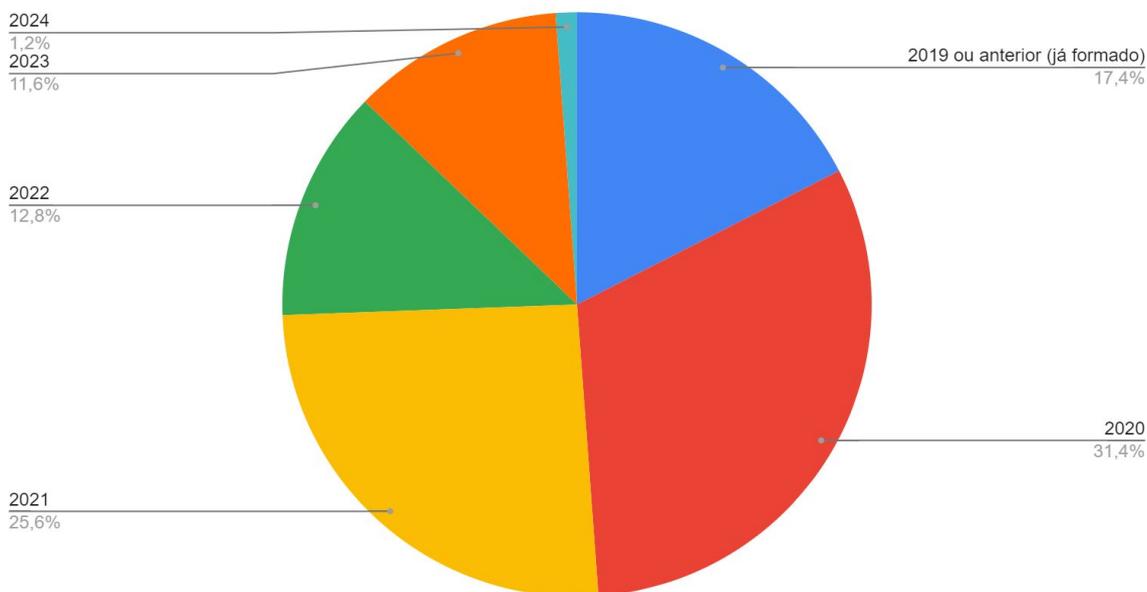


Imagem 8 - Respostas à pergunta 1e da pesquisa

A pesquisa foi divulgada em grupos de WhatsApp e de Facebook da Poli, além de ser pedido para compartilharem com outros grupos se fosse possível. Isso

resultou em um grupo mais diverso do que o usual. Porém, ainda é possível que a grande maioria dos respondentes estudam na USP e em São Paulo, além de um total combinado de 44,19% de alunos da Poli, mais especificamente, enquanto os alunos da Grande Área (GA) Elétrica da Poli reúnem 29,07% dos respondentes. Soma-se também um total combinado de 74,42% dos respondentes ou já estarem formados ou estarem nos últimos 2 anos de graduação, quando geralmente o estágio ocorre. Isso se trata de uma proporção muito favorável para focar nas respostas do público politécnico, mas ao mesmo tempo ter uma visão de como isso pode ser expandido para outros tipos de público.

Entendido o público, foi perguntado se o respondente estagiou.

Você já procurou por estágios?

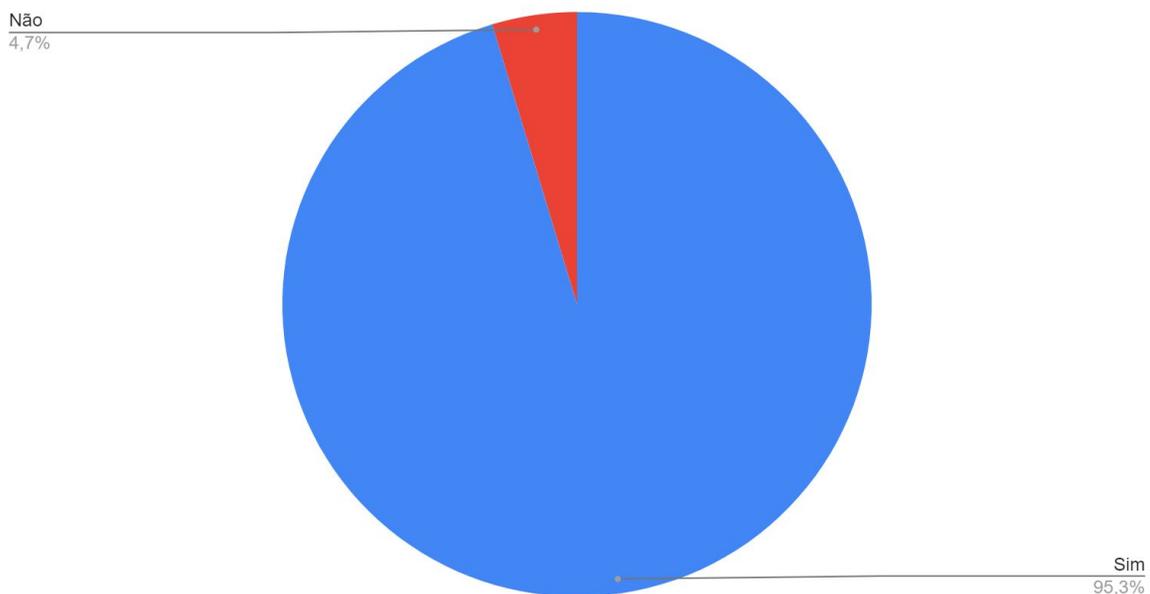


Imagem 9 - Respostas à pergunta 1f da pesquisa

As respostas mostram que é quase unânime a necessidade de estagiar e, se o recorte for feito em respondentes da USP, o número sobe para 100%.

Para os que procuraram estágio, foram feitas perguntas a respeito de como foi o processo:

Quanto tempo você demorou entre o começo da sua pesquisa e conseguir uma vaga? (Se já estagiou mais de uma vez, considere sua experiência mais recente.)

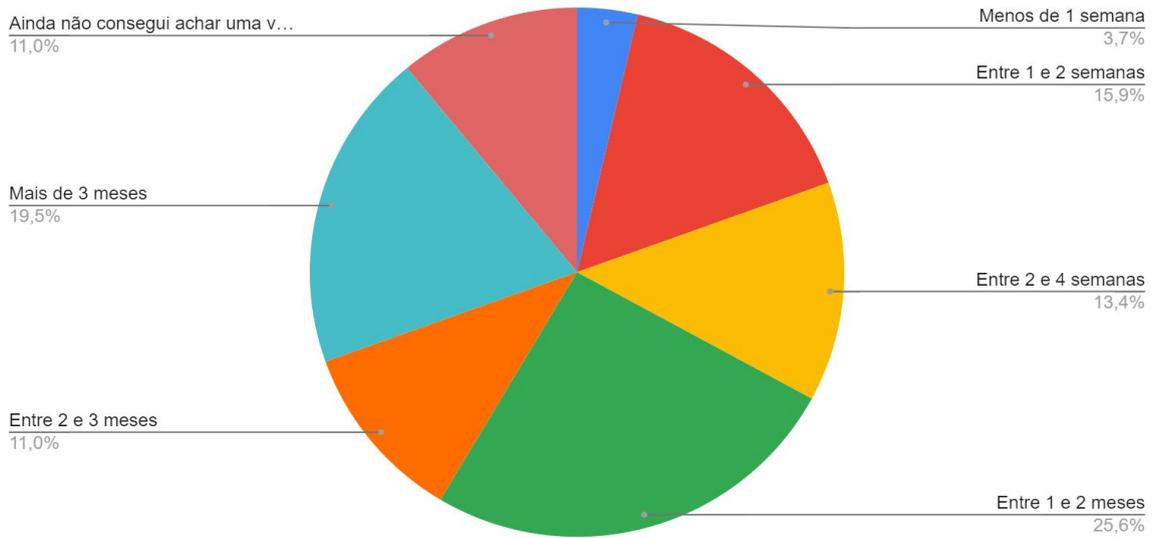


Imagem 10 - Respostas à pergunta 2a da pesquisa

Como você normalmente encontra vagas de estágio? (Marque todas as que se aplicam)

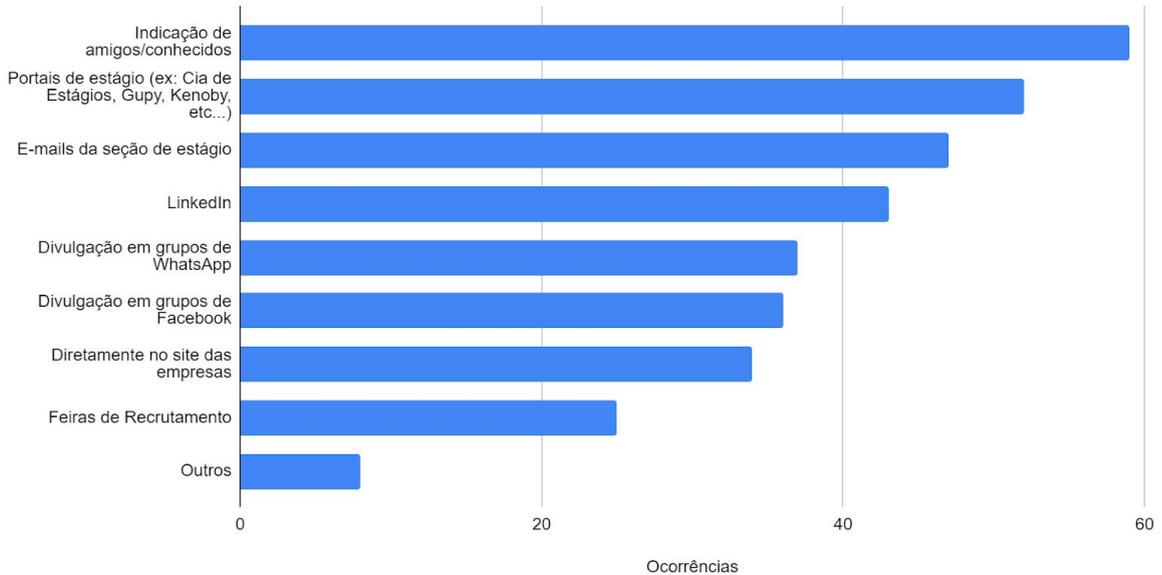


Imagem 11 - Respostas à pergunta 2b da pesquisa

Como você avalia se uma vaga ou empresa é um bom lugar para se trabalhar? (Marque todas que se aplicam)

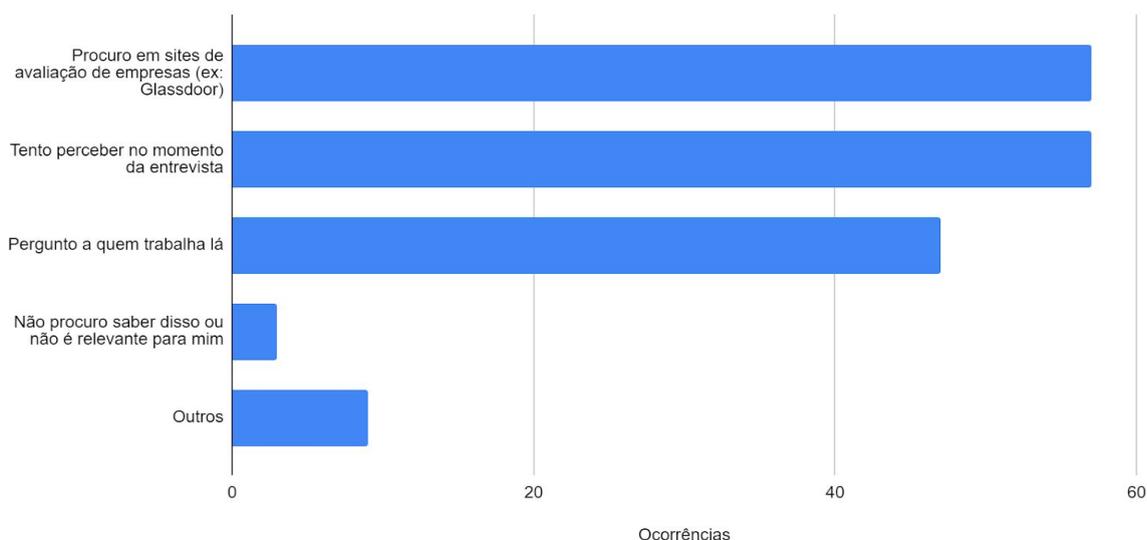


Imagem 12 - Respostas à pergunta 2c da pesquisa

O que te faz desistir de uma vaga? (Marque todas que se aplicam)

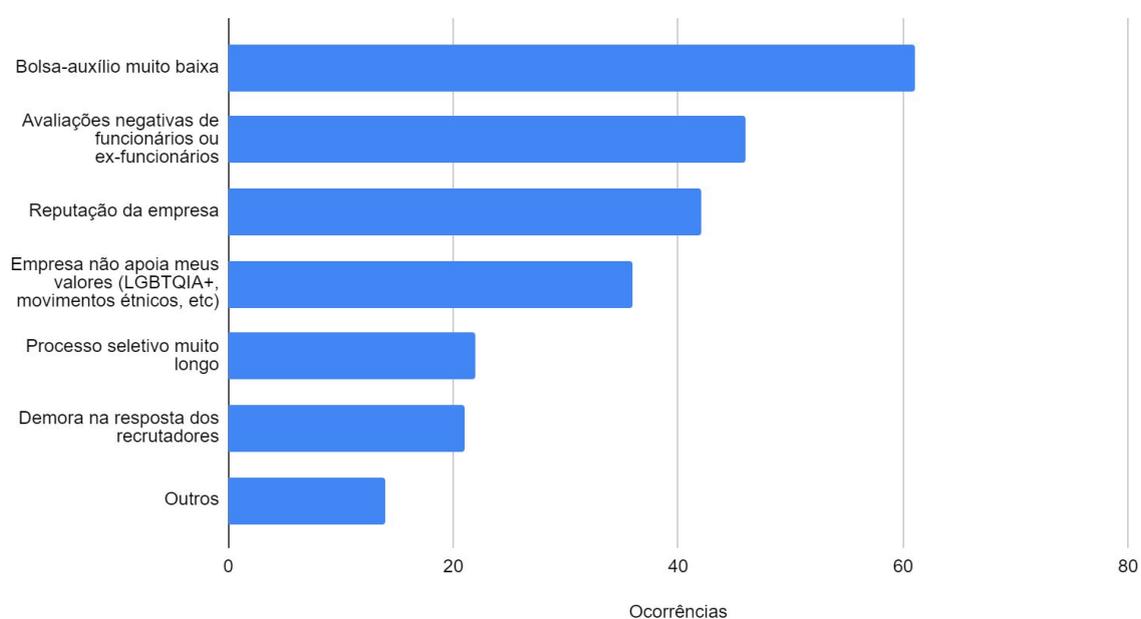


Imagem 13 - Respostas à pergunta 2d da pesquisa

As respostas da primeira pergunta mostram claramente o tamanho do processo para conseguir um estágio, onde 63,95% dos respondentes levaram mais de 1 mês ou ainda não conseguiram achar uma vaga de estágio, mostrando uma

necessidade não atendida desse público. Já o modo com que os alunos encontram vagas de estágio é bem disperso, com uma preferência para a “Indicação de amigos/conhecidos”, mostrando que a pesquisa muitas vezes é difusa e abrangente para tentar encontrar uma vaga que encaixe com o que o estudante procura, o que se encaixa nos depoimentos coletados nas entrevistas acima; nota-se também que a opção “Portais de estágio (ex: Cia de Estágios, Gupy, Kenoby, etc...)” fica bem posicionada, mostrando o potencial que uma plataforma desta tem. Essa forma mais pessoal com que os estudantes procuram estágio também se reflete em como eles avaliam se uma empresa é um bom lugar para se trabalhar, onde um número expressivo comenta que “Pergunta a quem trabalha lá”; neste caso há também outras duas formas mais expressivas de avaliar: “Procuro em sites de avaliação de empresas (ex: Glassdoor)” e “Tento perceber no momento da entrevista”.

Isso mostra que existe espaço para um projeto como esse suprir uma necessidade geral dos estudantes. Na pergunta aberta “Se pudesse melhorar esse processo, o que você mudaria?”, várias das respostas questionam a falta de transparência nos processos seletivos e falta de apoio da faculdade, algo que poderia ser melhorado com o Portal de Estágios.

Para os que não procuraram estágio, foi feita a seguinte pergunta a respeito do motivo pelo qual não houve a procura:

Por qual motivo você não procurou um estágio?

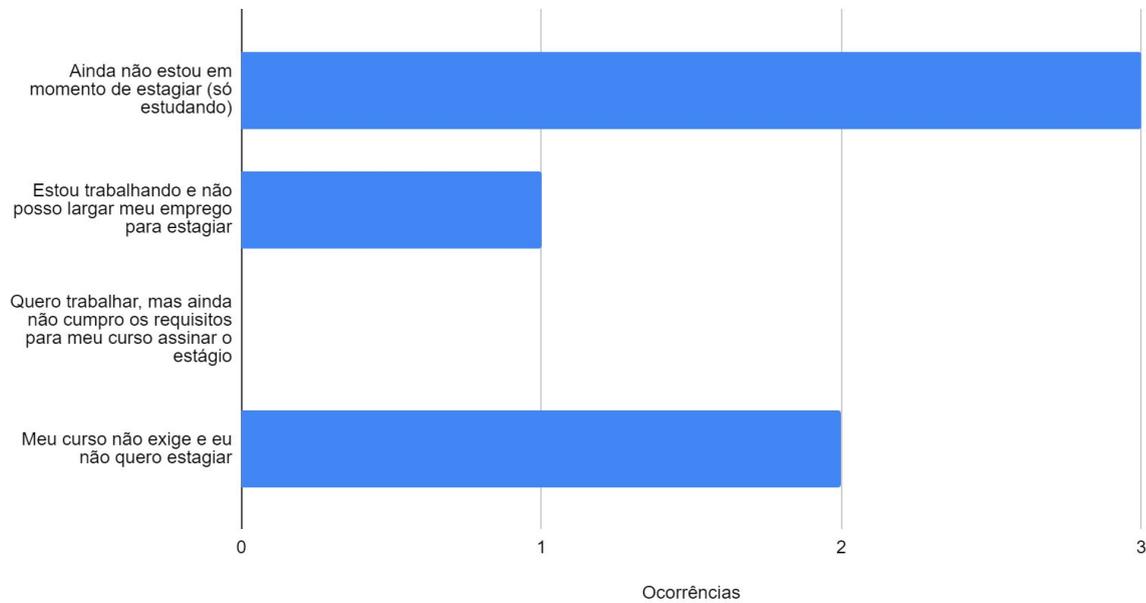


Imagem 14 - Respostas à pergunta 3a da pesquisa

Dada a pouca quantidade de respondentes que não procuraram estágio, não é possível tirar nenhuma conclusão a respeito deste grupo.

Na última seção, reservada a comentários finais e *feedback* da pesquisa, muitos elogiaram e acharam o tópico interessante. Alguns comentários deram sugestões já abordadas nas entrevistas, como a possibilidade de receber vagas pelo WhatsApp, o que reforça o que já foi constatado anteriormente. Com a última pergunta conseguimos o contato de 22 estudantes interessados em testar o sistema, o que será útil para a última etapa do projeto.

6. ESPECIFICAÇÃO DE REQUISITOS DO SISTEMA

Partindo de técnicas de Engenharia de Software^[4] e levando em consideração os problemas levantados nos tópicos anteriores, partiu-se para o planejamento técnico do projeto.

6.1. Atores

O conceito de ator divide os usuários do sistema a partir de características em comum, que definem um objetivo final e modo de usar específico de cada tipo.

6.1.1. Aluno

Um aluno da faculdade onde o software abrange. Neste caso, é um aluno da Escola Politécnica da USP, matriculado nos cursos de graduação sob responsabilidade do Departamento de Engenharia de Computação e Sistemas Digitais: Engenharia de Computação (quadrimestral) ou Engenharia Elétrica com Ênfase em Computação.

6.1.2. Representante da Empresa

Um funcionário de empresa parceira do Departamento de Engenharia de Computação e Sistemas Digitais.

6.1.3. Funcionário da POLI

Um funcionário autorizado da faculdade que irá acompanhar todos os processos e verificar relatórios gerados automaticamente a partir de agregação dos dados de uso ou a partir das avaliações dos alunos.

6.2. Casos de Uso

A partir dos problemas e considerações levantados anteriormente, buscou-se implementar os principais casos de uso do sistema.

6.2.1. Cadastrar Empresa no Sistema

Ator: Representante da Empresa, Funcionário da POLI

Descrição: Representante da Empresa se cadastra e cadastra empresa no sistema; Funcionário da POLI aprova se a empresa for conveniada.

Evento iniciador: Este caso de uso se inicia quando o representante da empresa entra na seção de cadastro do site e inicia o cadastro.

Pré-condição: Representante da Empresa não está cadastrado no sistema.

Pós-condição: Representante da Empresa cadastrado e logado no sistema, junto com todas as informações necessárias para as interações posteriores.

1. Representante da Empresa acessa página de cadastro do sistema.
2. Representante da Empresa escolhe opção de cadastro de empresa.
3. Sistema redireciona o Representante da Empresa para completar seu cadastro.
4. Representante da Empresa preenche suas informações e as da empresa da qual representa.
5. Representante da Empresa envia formulário.
6. Sistema valida as informações. Se houver informações inválidas, volta ao passo 3, senão o formulário é enviado.
7. Funcionário da POLI aprova cadastro se empresa for conveniada com a faculdade.
8. Sistema envia um email para o endereço cadastrado pelo Representante da Empresa, pedindo que ele crie uma senha para a conta.
9. Representante da Empresa cadastra senha e seu acesso ao sistema é liberado.

6.2.2. Postar Vaga de Estágio

Ator: Representante da Empresa

Descrição: Realiza a postagem de vaga de estágio no portal

Evento iniciador: Este caso de uso se inicia quando uma empresa cadastrada inicia a postagem de uma vaga de estágio.

Pré-condição: Empresa conveniada com a Poli e cadastrada no portal.

Pós-condição: Vaga de estágio postada no site, acessível a alunos e envio da vaga

por WhatsApp para alunos.

1. Empresa faz login no portal.
2. Empresa escolhe a opção de postar vaga de estágio.
3. Empresa preenche informações da vaga.
4. Sistema pede confirmação da postagem.
5. Empresa confirma postagem de vaga.

6.2.3. Cadastrar Aluno no Sistema

Ator: Aluno

Descrição: Aluno se cadastra no sistema

Evento iniciador: Este caso de uso se inicia quando o aluno entra na seção de cadastro do site e inicia o cadastro.

Pré-condição: Aluno não está cadastrado no sistema.

Pós-condição: Aluno cadastrado e logado no sistema, junto com todas as informações necessárias para as interações posteriores.

1. Aluno acessa página de cadastro do sistema.
2. Aluno escolhe opção de cadastro de aluno.
3. Sistema pede para Aluno selecionar a conta do Google OAuth.
4. Aluno seleciona a opção para usar seu email USP.
5. Sistema valida cadastro dá acesso para o aluno.
6. Sistema redireciona o Aluno para completar seu cadastro.
7. Aluno preenche suas informações pessoais e profissionais.
8. Aluno envia formulário.
9. Sistema valida as informações. Se houver informações inválidas, volta ao passo 6, senão o Aluno é cadastrado.

6.3. Funcionalidades

A seguir são apresentadas as principais funcionalidades definidas para o projeto.

O site terá quatro seções principais: Geral, Empresas, Alunos e Funcionário POLI - Estágios.

6.3.1. Geral

A seção *Geral* não necessita de login e contém informações gerais sobre o Portal.

a) Informações gerais: O site possuirá informações gerais sobre o funcionamento do portal, apresentado as *features* do site para empresas e alunos.

b) Instruções claras do processo de contratação: Documentos importantes serão colocados em evidência, de forma clara, com um passo-a-passo e avisando quando é necessário preencher algum deles (e-mail para adendo, por exemplo). Haverão links para download de documentos e templates.

6.3.2. Empresas

A seção *Empresas* necessita login e contém as funcionalidades que podem ser usadas pelas empresas.

a) CRUD Empresa: A empresa deverá realizar um cadastro para obter acesso a essa seção. Somente empresas com convênio com a POLI poderão realizar cadastros. Informações de como contatar a POLI para realizar o convênio serão apresentadas na seção *geral*.

b) Empresas postam vagas específicas: As empresas poderão cadastrar vagas de estágio com valor de bolsa-auxílio, benefícios, requisitos, etc. As vagas ficarão disponíveis para busca ou serão recomendadas de forma inteligente para os alunos.

c) Recomendações inteligentes: A recomendação de alunos é inteligente, busca afinidade entre as preferências e perfil do aluno com as características da oferta (Contexto, Perfil, Localização).

d) Informações do perfil: Cada empresa possui um perfil com suas informações cadastradas. As empresas poderão editar esse perfil. (Informações como valores da empresa área de atuação, história, perfis desejados, etc.). Algumas informações serão acessíveis pelos alunos, e outras são dados apenas visíveis para a empresa.

e) Painel de status de alunos: Nesta seção haverá um painel com dados importantes dos alunos que tiverem iniciado alguma relação com a empresa. A empresa poderá ver o status de contratação (processo seletivo, contratado, documentos pendentes, etc) e poderá acessar o perfil do aluno.

f) Procurar alunos: Além de fazer recomendações, a empresa pode procurar por alunos baseado em palavras-chave ou filtros.

6.3.3. Alunos

A seção *Alunos* necessita login e contém as funcionalidades que podem ser usadas pelos alunos.

a) CRUD Aluno: Aluno deverá realizar um cadastro para obter acesso a essa seção. A realização do cadastro e posterior login poderá ser relacionada ao e-mail usp do aluno. Ao realizar o cadastro o aluno poderá digitar informações de perfil mas essas informações podem ser cadastradas e/ou editadas posteriormente.

b) Informações do perfil: Cada aluno possui um perfil com suas informações cadastradas. Os alunos poderão editar esse perfil. (Vagas desejadas, endereço, quanto espera ganhar, áreas de interesse, etc). Algumas informações serão acessíveis pelas empresas e outras são dados apenas visíveis apenas para o aluno.

c) Busca de vagas: O aluno poderá fazer a busca por vagas

d) Avaliação das empresas por alunos: Alunos podem deixar seu relato de como foi trabalhar com uma empresa. Os relatos estarão disponíveis para os alunos e

funcionários.

6.3.4. Funcionário POLI - Estágios

O funcionário da POLI deverá ter privilégios de administrador do site. Ele poderá ver informações tanto de empresas como de alunos.

a) Aceitar Empresa: O funcionário da POLI será responsável por verificar se a empresa possui convênio com a Escola e gera pelo sistema um login e senha para a empresa completar o cadastro.

b) Modificar Status do Aluno: O funcionário pode modificar o status do aluno (por exemplo, para contratado, quando houver uma contratação), ou aceitar relatórios de estágio e documentos enviados.

6.4. Diagrama Entidade-Relacionamento

Uma vez levantadas todas as especificações de funcionalidades e atores, foi concebido o diagrama entidade-relacionamento (DER), para a modelagem do banco de dados do projeto. Padronizou-se para este projeto que todos os nomes de *apps*, classes, atributos e métodos devem ser escritos em inglês, seguindo a convenção do Python.

Dada a complexidade atingida combinação de todos os requisitos levantados, o diagrama foi quebrado em diversas partes. Estas foram divididas em *apps* no Django, de forma a agrupar semanticamente os diferentes objetos do sistema.

Os *apps* e as classes contidas nas *models* dentro de cada um são:

- **backoffice**
 - SiteSetting
 - Message
 - Score
 - Vote
 - SiteItemReview

- Flag
- **companies**
 - Company
 - JobOpportunity
 - Rating
- **education**
 - InstituteAndUniversity
 - Department
 - GraduationCourse
 - AcademicEntity
- **events**
 - Event
 - Round
- **experience**
 - Skill
 - Role
 - Position
 - Experience
- **locations**
 - StateAndCountry
 - City
 - Address
- **users**
 - User
 - Person
 - Student
 - Employer
 - Administrative

O diagrama em sua versão completa encontra-se no **Apêndice A** ao final do documento. Este documento serviu de base para estruturação do back-end e define todos os *models* que serão implementados.

6.4.1. App users

Neste *app* foram definidas todas as funcionalidades e modelagens a respeito dos usuários do sistema. Para isso, utilizou-se como base os atores do sistema levantados anteriormente para gerar 3 diferentes classes de usuário, um para cada. Visto que muitos atributos foram comuns a todos os tipos de usuário, optou-se por criar também uma classe abstrata de pessoa (*Person*), que não gera uma tabela no banco de dados, mas existe como classe na *models* deste *app* no Django.

Para a autenticação no sistema, optou-se por usar a classe *User* do próprio Django, para usufruir de todos os recursos de segurança do *framework*. Para atender as necessidades do processo, esta classe teve sua funcionalidade expandida para acomodar atributos essenciais para o cadastramento: o atributo *user_type* é necessário para ditar qual das 3 classes de usuário que representam os atores devem ser conectada com a que é responsável pela autenticação.

O DER deste *app*, então fica desta forma:

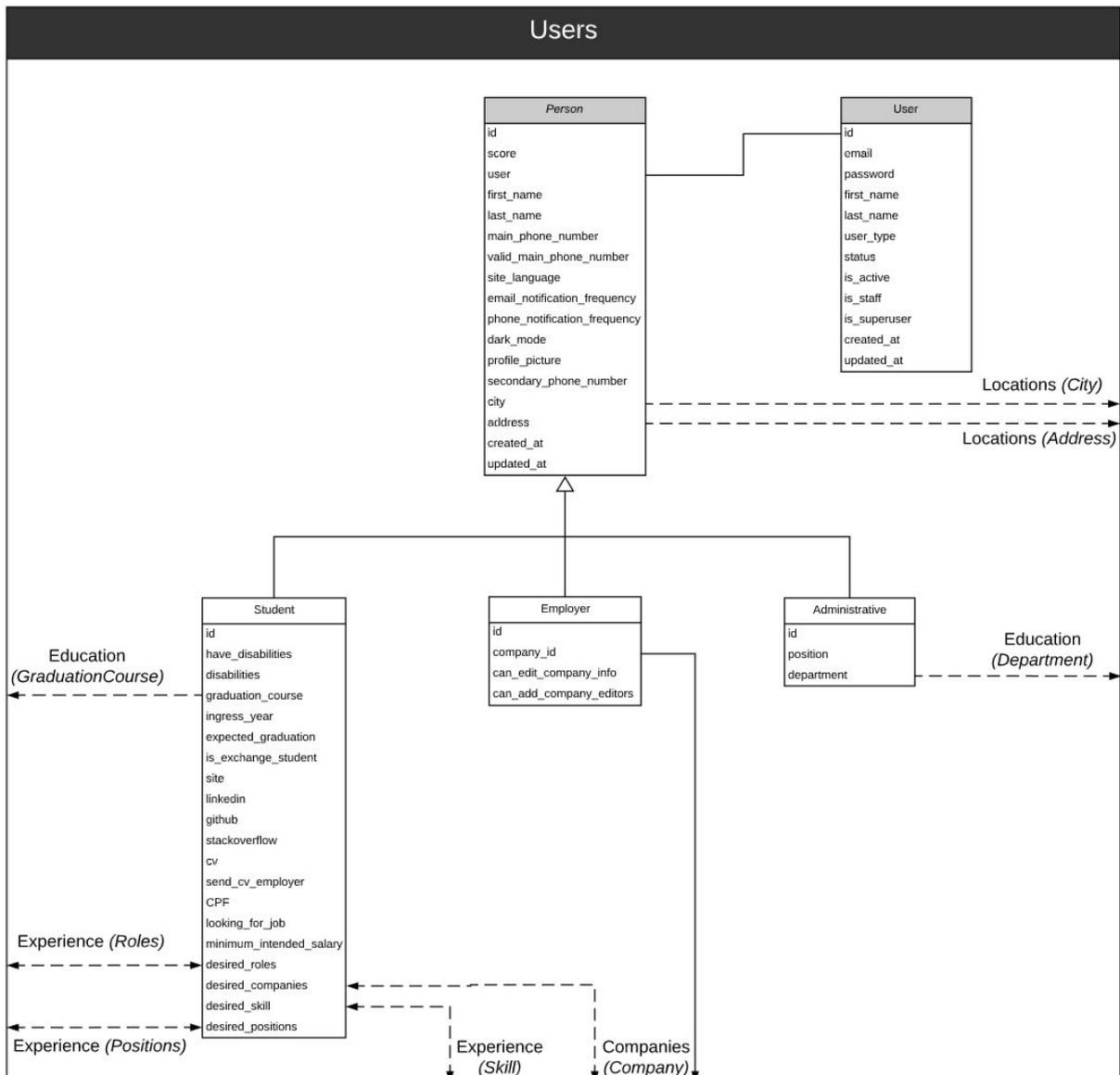


Diagrama 1 - DER do app users

6.4.2. App companies

Uma vez estabelecidos os usuários do sistema, foca-se no app que contém as classes referentes às empresas. Nela está contida a classe para a empresa (*Company*), que serve de agregador de informações para objetos de outras classes (vagas e avaliações, principalmente). Essa classe serve também como forma de identificar os usuários do tipo empregador (*Employer*) e mostrar para eles apenas as informações relevantes para a empresa que representa.

A classe de vaga (*JobOpportunity*) retém todas as informações que foram consideradas importantes e levantadas em entrevistas e na pesquisa. Apesar disso, como forma de incentivar o compartilhamento de vagas na plataforma, poucos campos são obrigatórios, e podem ser preenchidos coletivamente pelos alunos.

A classe de avaliação (*Rating*), apesar de estar no contexto de empresa, está ligado com a classe de experiência (*Experience*). Isso foi uma decisão de projeto para atrelar a avaliação à condição de que o estudante já tenha tido uma experiência anterior. Dessa forma, garante-se que a avaliação possui maior chance de refletir a realidade dentro da empresa e impede que empregadores e administradores consigam inflar os números com suas próprias avaliações.

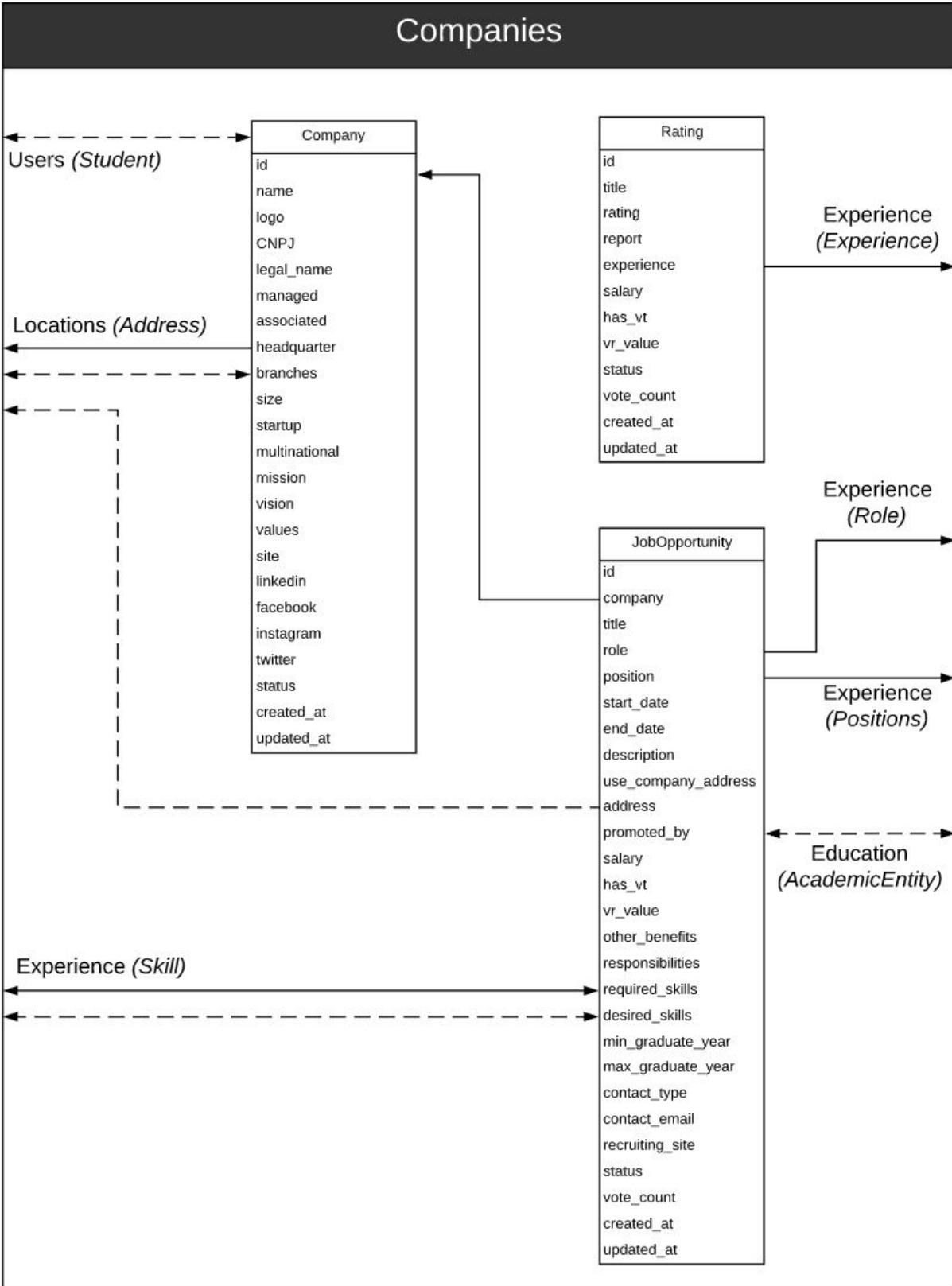


Diagrama 2 - DER do app companies

6.4.3. App experience

Neste app, a principal classe é a homônima: a de experiência (*Experience*). Com ela, buscou-se reproduzir o processo de ter um currículo na plataforma para que as empresas pudessem de forma mais fácil averiguar quais as habilidades e competências que o aluno possui. Isso garante também que o empregador saiba de onde vêm as habilidades dos estudantes, dando credibilidade a estas.

Para fazer essa verificação, há a classe de habilidades (*Skills*), em que os alunos podem colocar todas as competências que usaram durante a experiência. Ela foi feita de forma com que se possa criar categorias de habilidades, aqui chamadas de habilidades abstratas. Isso facilita tanto a busca por vagas que são de interesse para o estudante, quanto a busca de estudantes pelos empregadores.

A classe de posição (*Position*) foi criada para não deixar estático o tipo de vaga permitida na plataforma. Com isso, as vagas podem ser mais do que apenas estágios, como por exemplo estágio de férias, iniciação científica, trainee e até mesmo efetivo. Isso permite também que os alunos possam colocar experiências anteriores que não se encaixam no padrão da Poli (no caso do aluno ter trabalhado antes de entrar na faculdade).

Já a classe de cargo (*Role*) permite que os cargos disponíveis sejam ilimitados. Isso é uma faca de dois gumes, como foi levantado na entrevista com a funcionária de RH, pois ao mesmo tempo que não limita quais vagas podem ser compartilhadas (e, assim, o público alvo da plataforma), também pode gerar duplicatas que dificultam a busca e o mapeamento por parte da plataforma. Com isso, foi adicionado um campo de nome padronizado (*standardized_name*) para que isso possa ser remediado pela intervenção de algum moderador da plataforma.

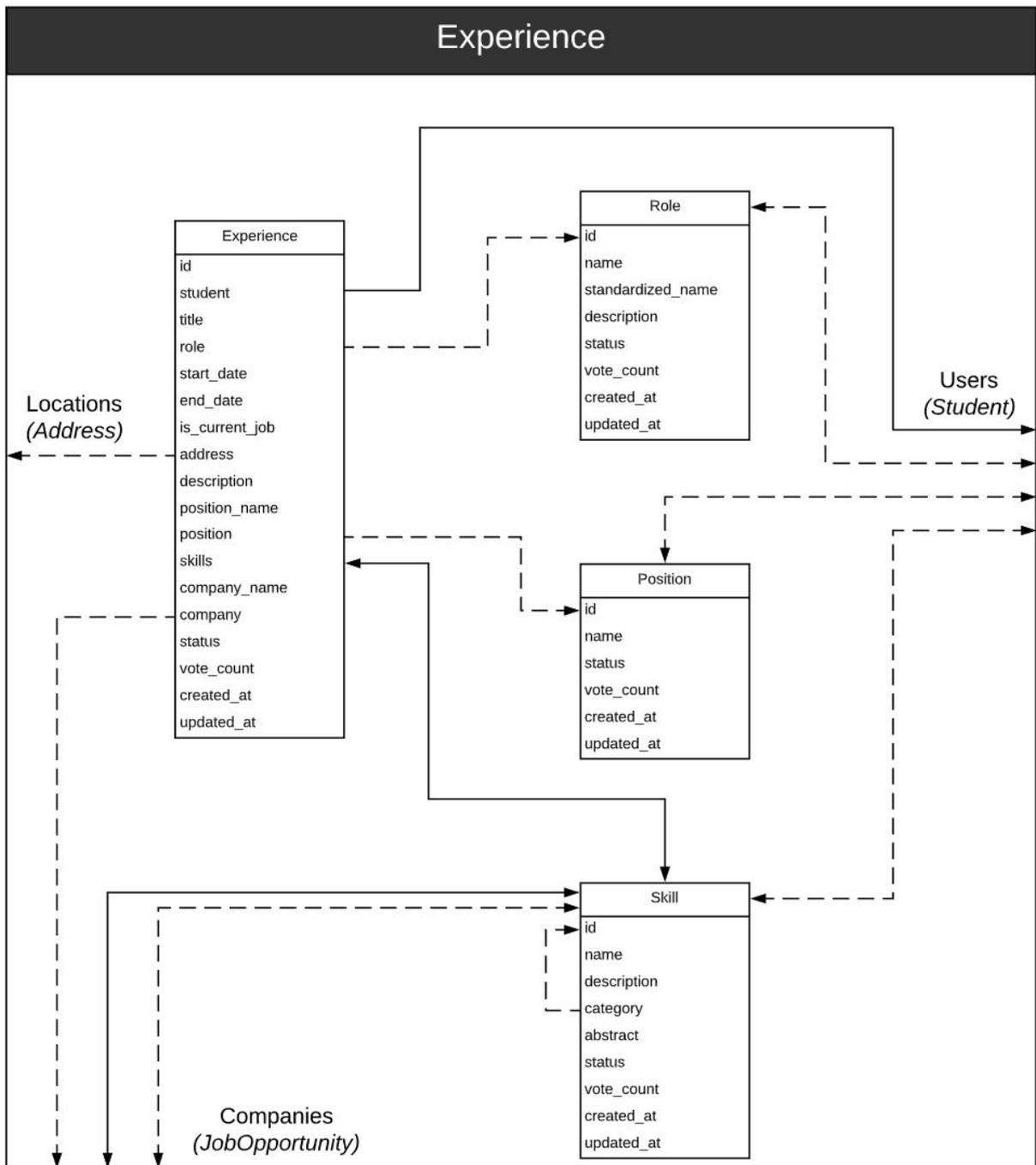


Diagrama 3 - DER do app experience

6.4.4. App education

Neste app, foram colocadas as classes de departamento (*Department*), curso de graduação (*GraduationCourse*) e instituto/universidade (*InstituteAndUniversity*). O intuito de não deixar esses valores estáticos foi deixar com que exista uma possibilidade de expansão de público alvo, caso haja interesse por parte do

departamento, para que este sistema seja uma solução unificada da Poli, USP ou até mais.

A classe de entidade acadêmica (*AcademicEntity*) foi adicionada após a entrevista com o gerente do WI, onde chegou-se à conclusão de que elas potencialmente possuiriam interesse em divulgar vagas promovidas pelas mesmas. Elas podem ser geridas apenas por estudantes e possuem apenas este propósito na plataforma.

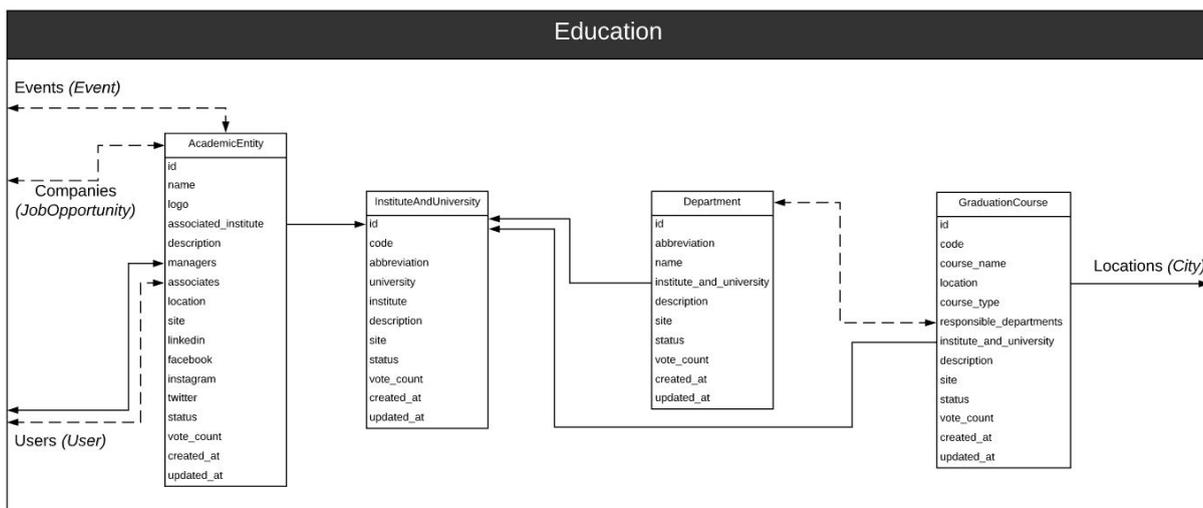


Diagrama 4 - DER do app education

6.4.5. App locations

Este *app* tem 2 objetivos: guardar os endereços e permitir a expansão da plataforma para outros locais além de São Paulo.

O primeiro está relacionado com a classe de endereço (*Address*) que, por ser um conteúdo comum em diversas outras classes da plataforma, foi abstraído para sua própria classe, a fim de padronizar como esses dados são guardados. Isso permitiria a adição de diversas funcionalidades posteriormente, como a de sugerir vagas de empresas de até um certo raio de distância de onde o aluno mora. Este exemplo foi levantado em uma das entrevistas, mas não foi priorizado para este projeto em sua concepção inicial.

O segundo, com as classes de cidade (*City*) e estado/país (*StateAndCountry*), serve para limitar os locais com os quais a plataforma permitiria a operação. Ao criar uma classe em vez de campos fechados, é possível flexibilizar ou não esse critério, ficando sob a decisão do departamento.

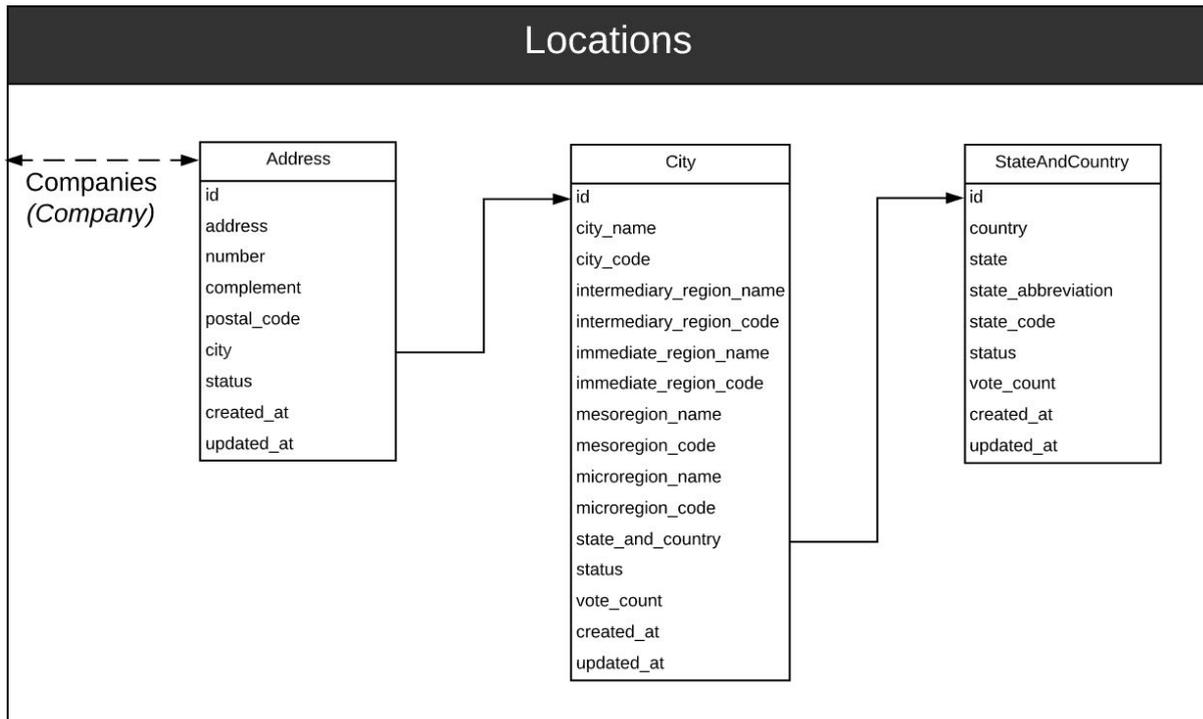


Diagrama 5 - DER do app location

6.4.6. App events

Neste *app*, foram colocadas funcionalidades extras que seriam interessantes para o projeto, mas não prioridade.

A classe de evento (*Event*) foi uma forma de integrar os eventos relacionados com estágio na plataforma. Isso já é comum atualmente, onde o departamento realiza almoços e palestras com as empresas interessadas em conhecer melhor o público politécnico. Esse tipo de evento é muito comum também quando realizado por entidades acadêmicas, que buscam aproximar a realidade do mercado com a Escola.

Já a classe de rodada (*Round*), seria um período curto de tempo em que os alunos e empresas receberiam avisos prévios e seriam instigados a entrar na

plataforma para realizar uma contratação rápida. A ideia por trás é ajudar os alunos dos cursos quadrimestrais a terem um período curto e eficiente para a busca de estágios para o módulo cooperativo.

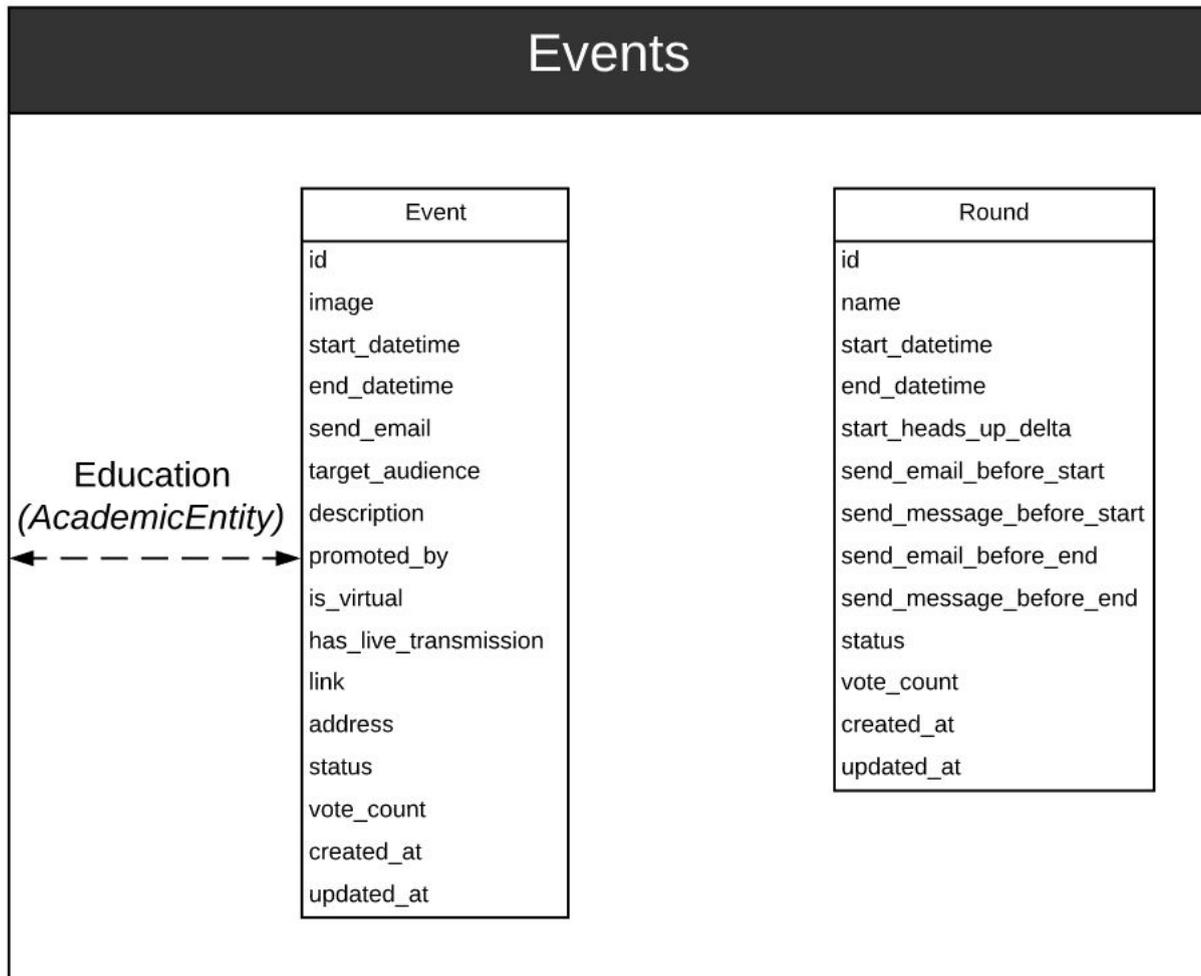


Diagrama 6 - DER do app events

6.4.7. App backoffice

Este *app* contém todas as ferramentas de moderação do site, de forma a garantir ao departamento um melhor controle sobre o uso da plataforma.

A classe de pontuação (*Score*) categoriza, armazena e dá ao usuário pontos por realizar certas tarefas no sistema, como preencher todo seu cadastro, entrar na plataforma, complementar informações de vagas, etc. Essa estratégia de gamificação tornaria a plataforma mais ativa e ajudaria o departamento a moderar e garantir um conteúdo de alta qualidade. A recompensa por esses pontos seria na

ordenação com que esses alunos apareceriam para os empregadores interessados em contratar.

A classe de votação (*Vote*) serve para ajudar a moderar o conteúdo do site, de forma com que vagas, avaliações ou eventos não relevantes sejam “enterrados” ao final da lista quando o usuário for procurar e apareçam mais a frente quando o contrário ocorrer.

A classe de denúncia (*Flag*) serve para que os usuários possam denunciar conteúdo mal-intencionado na plataforma para a ação direta dos administradores do sistema.

A classe de revisão de item do site (*SiteItemReview*) serve para juntar em um só lugar todas as pendências de moderação do site, de forma a agilizar o processo e garantir que nenhum item seja esquecido. Entre as pendências estão as denúncias, edições de conteúdo por um usuário que não o que criou (contribuições) e aprovação de usuários que tiveram seu cadastro não aprovado por algum motivo.

A classe de mensagens (*Message*) serve para enviar notificações aos usuários. Essas notificações podem ser tanto automáticas (como ao ter um pedido de edição de conteúdo aprovado), quanto mensagens diretas dos moderadores para os usuários. Essas mensagens podem ser enviadas também ao WhatsApp do usuário se necessário.

A classe de configurações do site (*SiteSetting*) contém apenas dicionários chave para dar um pouco mais de maleabilidade em configurações específicas do sistema (como quais provedores de email são permitidos para fazer o cadastro na plataforma).

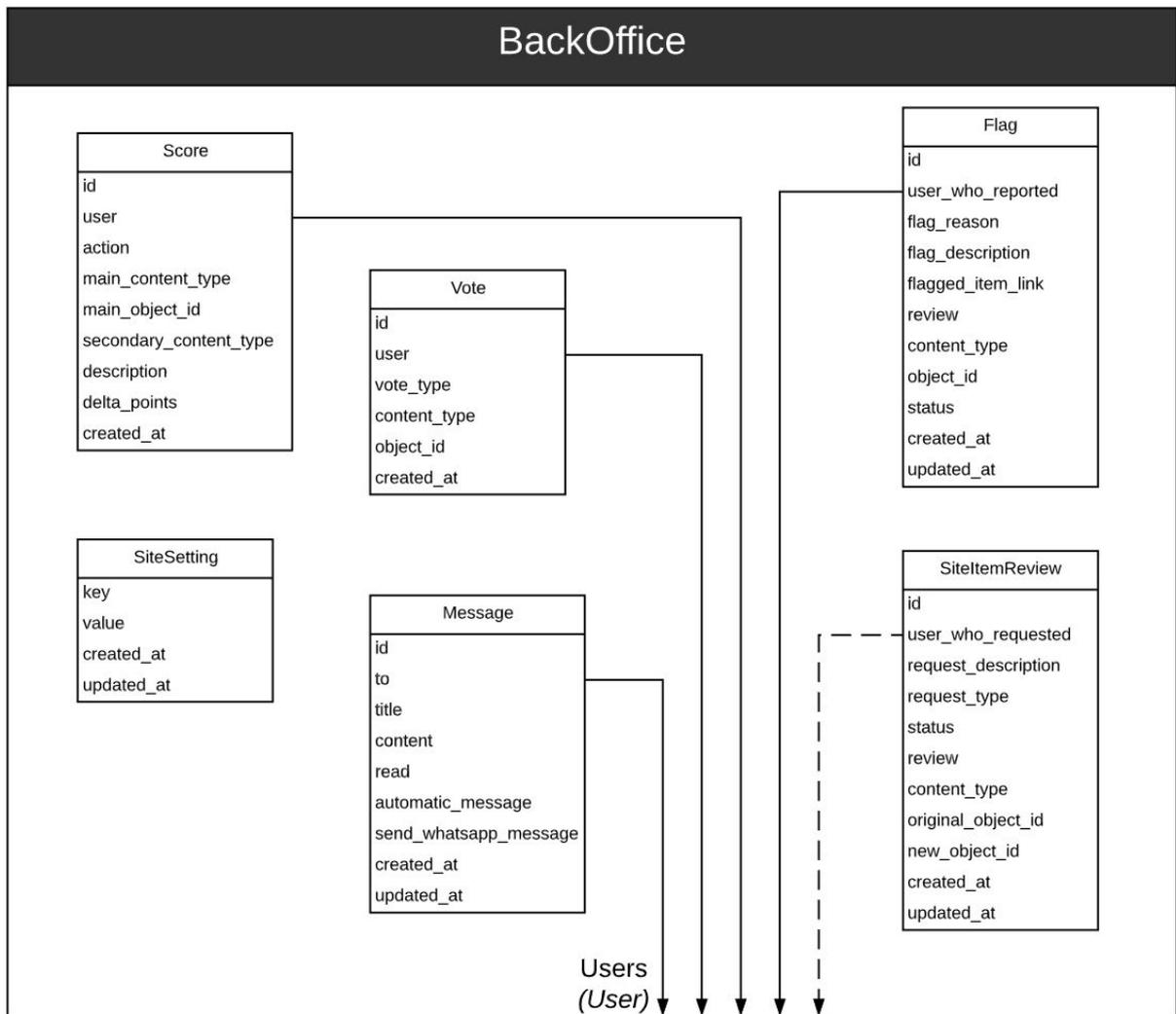


Diagrama 7 - DER do app backoffice

6.4.8. Relações entre apps

Uma vez estabelecidos todos os *apps*, montou-se um diagrama entidade-relacionamento simplificado de forma a explicitar todas as relações entre eles, numa visão macro do sistema.

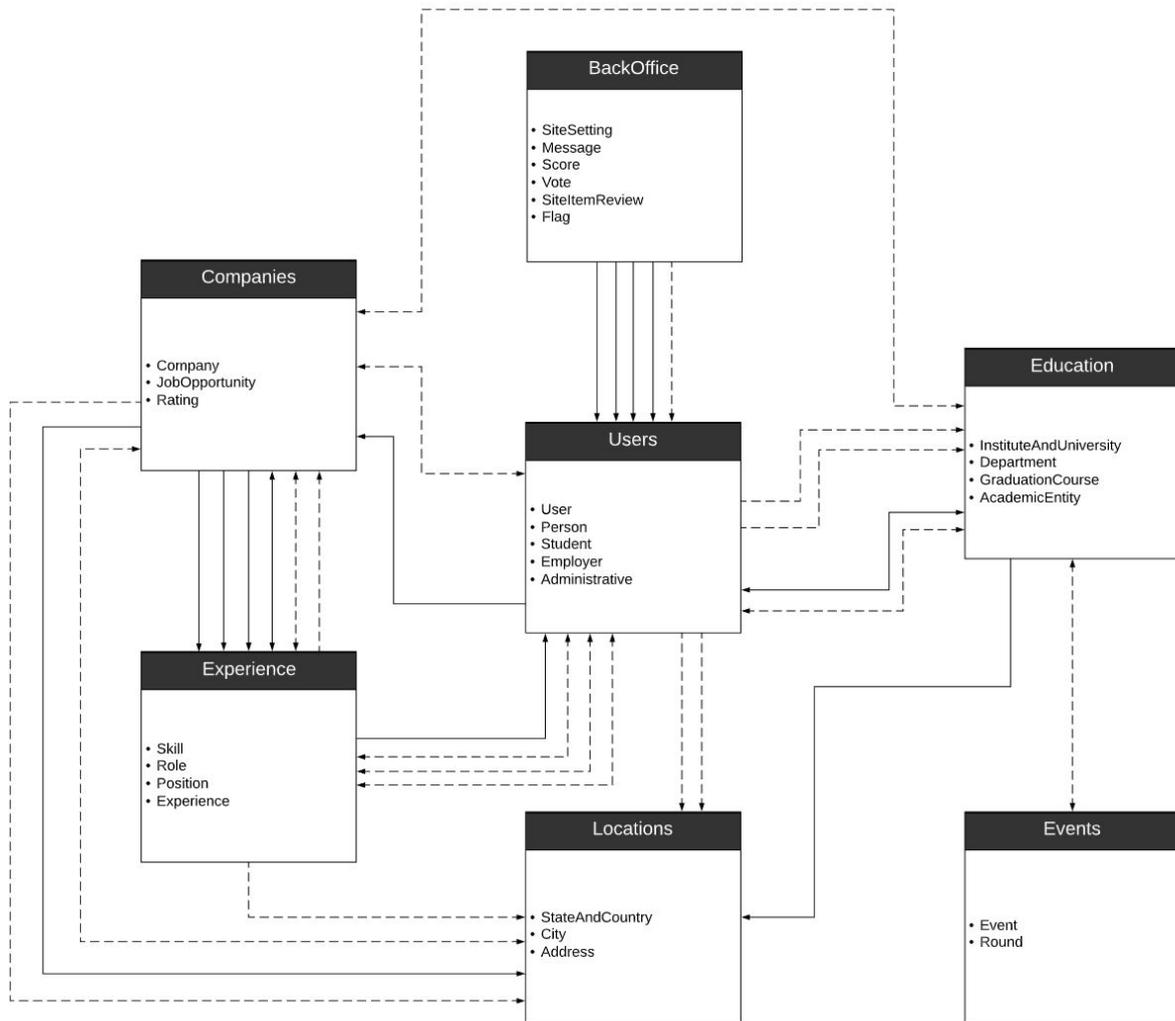


Diagrama 8 - DER simplificado do sistema

7. PROJETO E IMPLEMENTAÇÃO

A partir dos requisitos levantados, optou-se por utilizar a metodologia ágil *Scrum* para coordenar o processo de desenvolvimento de software. Para isso, foi utilizada a framework *Sprint* para dividir a implementação do projeto em etapas com períodos de aproximadamente um mês e entregas bem definidas em cada um deles. As datas foram pensadas com base nas entregas da disciplina de estágio e o esforço necessário para cada uma delas.

O resultado do planejamento realizado encontra-se na tabela a seguir.

Sprint	Período	Entregas
1	01/09 - 30/09	<ul style="list-style-type: none">- Elaboração de um diagrama de modelo relacional do banco de dados.- BE: Estruturação do back-end com todos as entidades e relações estabelecidas nos requisitos. Criação dos <i>models</i>.- Criação de um robô que envia mensagens via WhatsApp Web.- FE: Criação de um layout inicial para <i>Homepage</i> com sessões públicas de apresentação do aluno e empresa.
2	01/10 - 31/10	<ul style="list-style-type: none">- FE e BE: Criação dos Dashboards do aluno e da empresa e páginas relacionadas- Criação de todos os CRUD's necessários e páginas relacionadas. Criação das <i>views</i>.
3	01/11 - 26/11	<ul style="list-style-type: none">- Integração das funcionalidades de criação de vagas e recomendações com o bot

		<p>WhatsApp</p> <ul style="list-style-type: none"> - Integração do sistema com o Dialogflow - Criação da rodada de contratação
4	26/11 - 07/12	<ul style="list-style-type: none"> - Testes, refatorações, reparos de possíveis bugs, refinamentos.

Tabela 1 - Cronograma resultante do planejamento

O acompanhamento das subtarefas que envolvem atingir os objetivos citados estão organizadas em uma ferramenta web chamada *Trello*, que implementa a *framework* utilizada de forma com que facilite o processo de desenvolvimento remoto demandada pelas circunstâncias atuais de um mundo em pandemia.

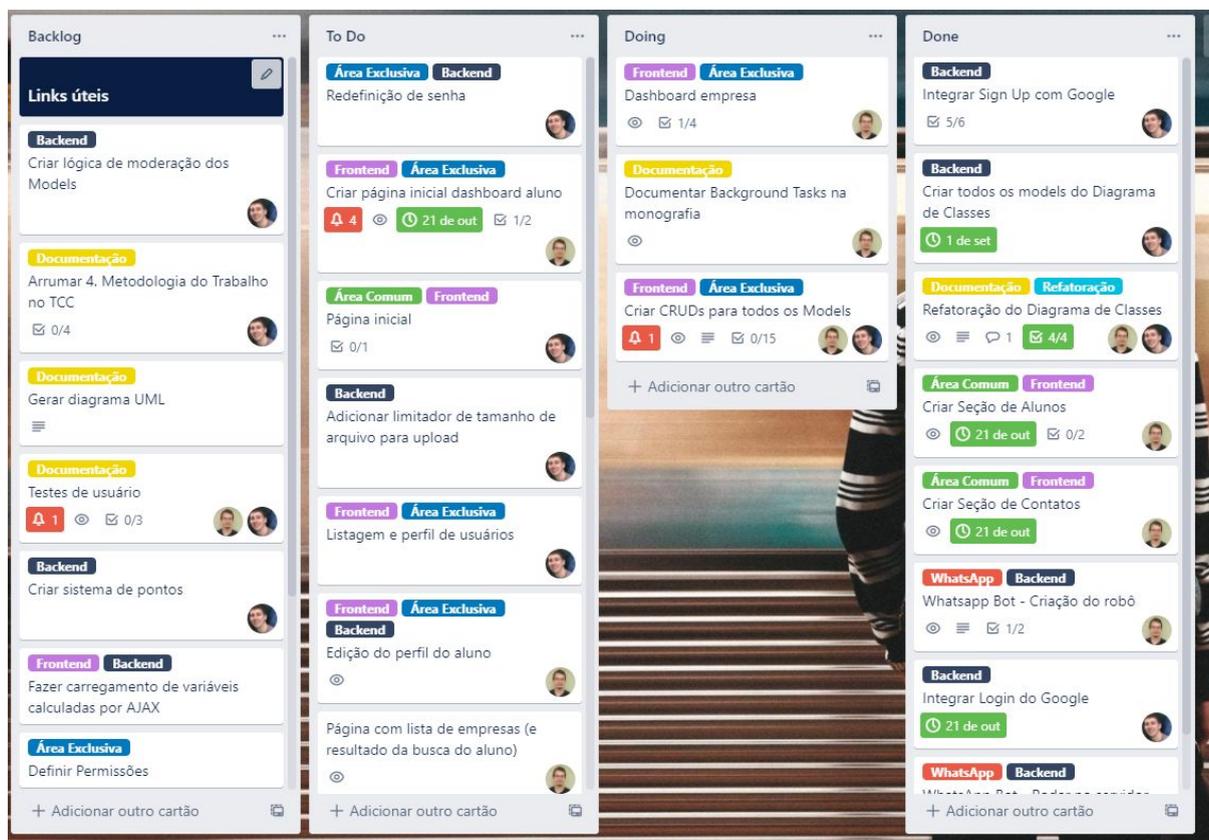


Imagem 15 - Mural do Trello - Acesso em 22 nov 2020

7.1. Administração do Django

A correta estruturação e implementação dos modelos e relações entre eles é de suma importância pois o sistema todo se apoia nessas entidades. Os modelos foram criados com base no diagrama entidade-relacionamento desenvolvido anteriormente.

Para servir de apoio ao administrador que deseja fazer alterações de forma irrestrita no sistema, o Django possui um sistema de gestão de conteúdo (ou CMS) em que as funções básicas de CRUD já estão implementadas. Essa seção foi muito utilizada também para realizar testes durante o processo de desenvolvimento.

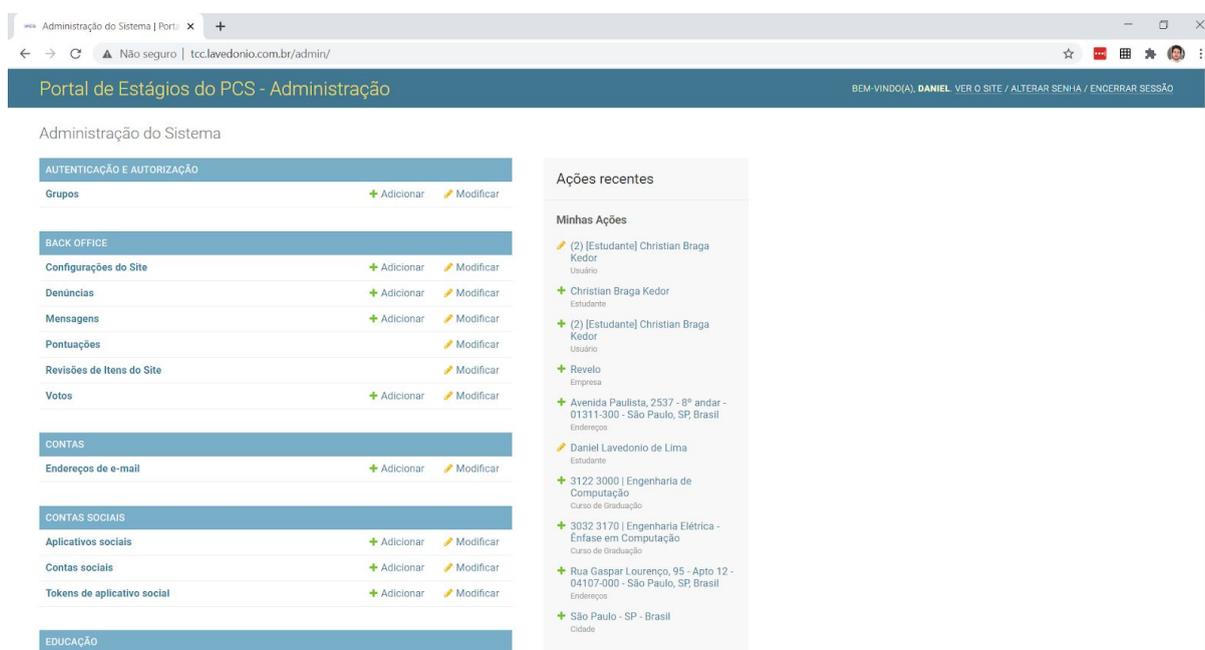


Imagem 16 - Seção de Administração do Django do novo portal de estágios do PCS.

Acesso em: 22 out. 2020

7.2. Página inicial e Sessões Públicas

As sessões públicas do portal são as páginas informativas, e não necessitam de login. Servem para apresentar o que é o portal, que necessidades ele atende e quem pode utilizá-lo, bem como informações de contato.

7.2.1. Página Inicial

A página inicial do portal contém uma barra de navegação que contém o link para acessar as outras partes públicas do portal, bem como o acesso ao login. A *hero-section*, que é o conteúdo principal da página, é uma mensagem de “bem-vindo” com links para entrar ou fazer o cadastro.



Imagem 17 - Homepage do novo portal de estágios do PCS.

Acesso em: 26 nov. 2020

7.2.2. Página pública “Sou Aluno”

A sessão pública “Sou Aluno” tem a intenção de apresentar o que é o portal para o estudante. A página contém informações sobre o que ele vai encontrar no site, quais são as principais *features*, informações sobre estágio na Poli e *link* para fazer o cadastro.



Imagem 18 - Trecho da sessão pública “Sou Aluno”

Acesso em: 26 nov. 2020

7.2.3. Página pública “Sou Empresa”

De forma análoga à sessão pública do aluno, a página para as empresas apresentará o portal e as funcionalidades para empresa, que tipos de alunos vai atingir, vantagens e etc. Qualquer pessoa pode acessar essa página.

7.2.4. Página pública “Contato”

Página para qualquer pessoa contatar a seção de estágios. A página contém um formulário para enviar informações como nome, e-mail, assunto e o campo com a mensagem. A mensagem será enviada para os administradores do site. Nessa seção também estão disponíveis o endereço, telefone e e-mail da secretaria do PCS.

Contato

Tem dúvidas? Não hesite em nos escrever e responderemos suas perguntas o mais rápido possível!

Nome

E-mail

Assunto

Mensagem

Av. Prof. Luciano Gualberto, tv. 3, 158
Butantã - São Paulo - SP
05508-010
+55 (11) 3091-5583
pcs.secretaria.poli@usp.br

Imagem 19 - Trecho da sessão pública “Contato”. Acesso em: 26 nov. 2020

7.3. Login/Sign-Up por meio da conta Google

O processo de login/sign-up na plataforma pode ser feito de duas formas: preenchendo o formulário de acesso ou entrando com uma conta Google. Para o usuário que escolher a segunda opção, ele seguirá os passos a seguir para entrar no sistema.

Passo 1: Ao clicar no botão “Acessar” na barra de navegação da sessão pública, o usuário do sistema irá para uma tela onde poderá escolher em qual categoria se encaixa: “estudante”, “empresa” ou “administrativo”.

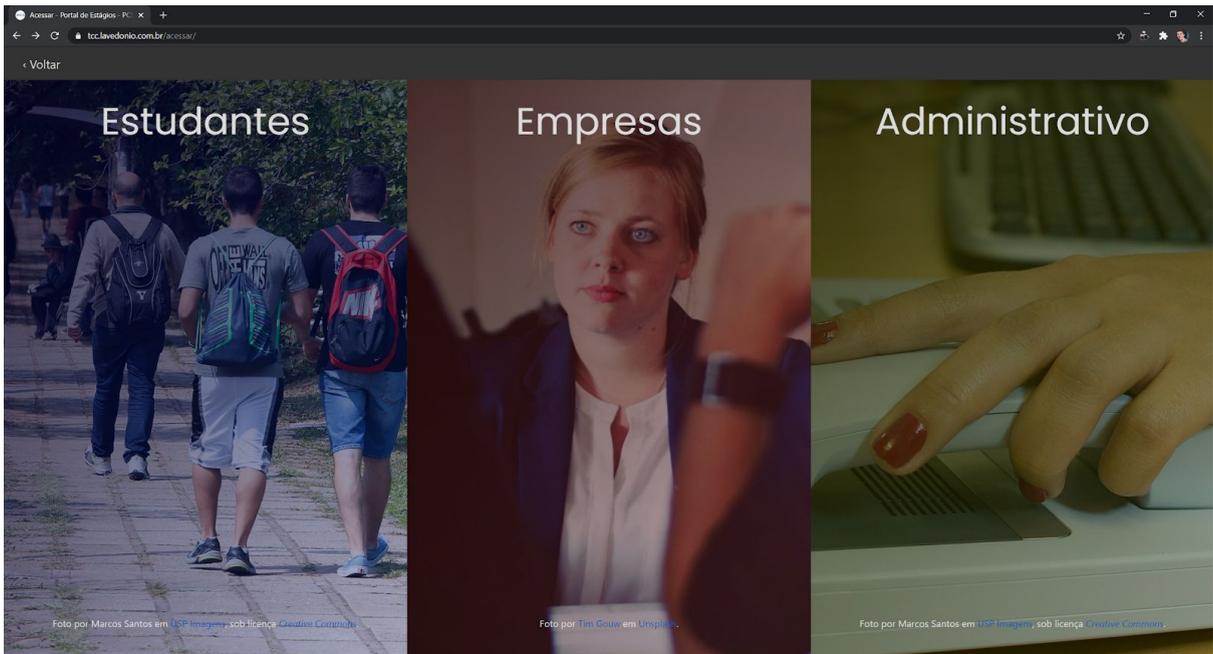


Imagem 20 - Passo 1 do processo de cadastro pela conta Google

Passo 2: Ao fazer a escolha, o usuário que quiser se autenticar com a conta Google pode clicar no botão “Entrar com sua conta Google”.

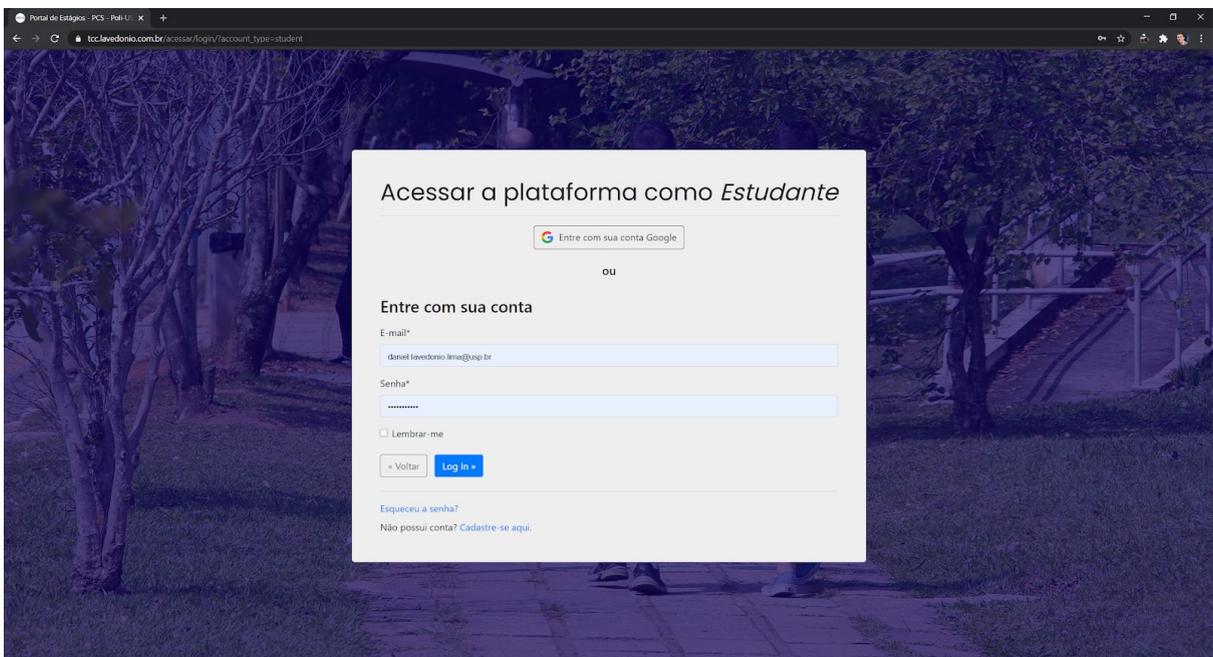


Imagem 21 - Passo 2 do processo de cadastro pela conta Google

Passo 3: O site redireciona o usuário para o processo de autenticação sob tutela do Google, onde ele deverá escolher o e-mail com o qual irá entrar.

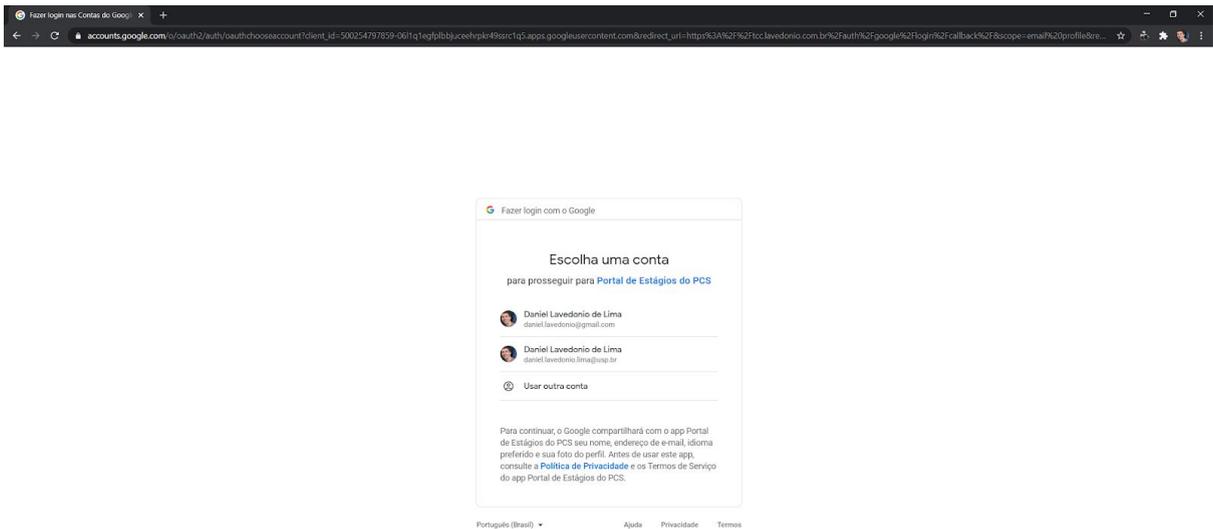


Imagem 22 - Passo 3 do processo de cadastro pela conta Google

Passo 4: O usuário está autenticado e pronto para usar o sistema.

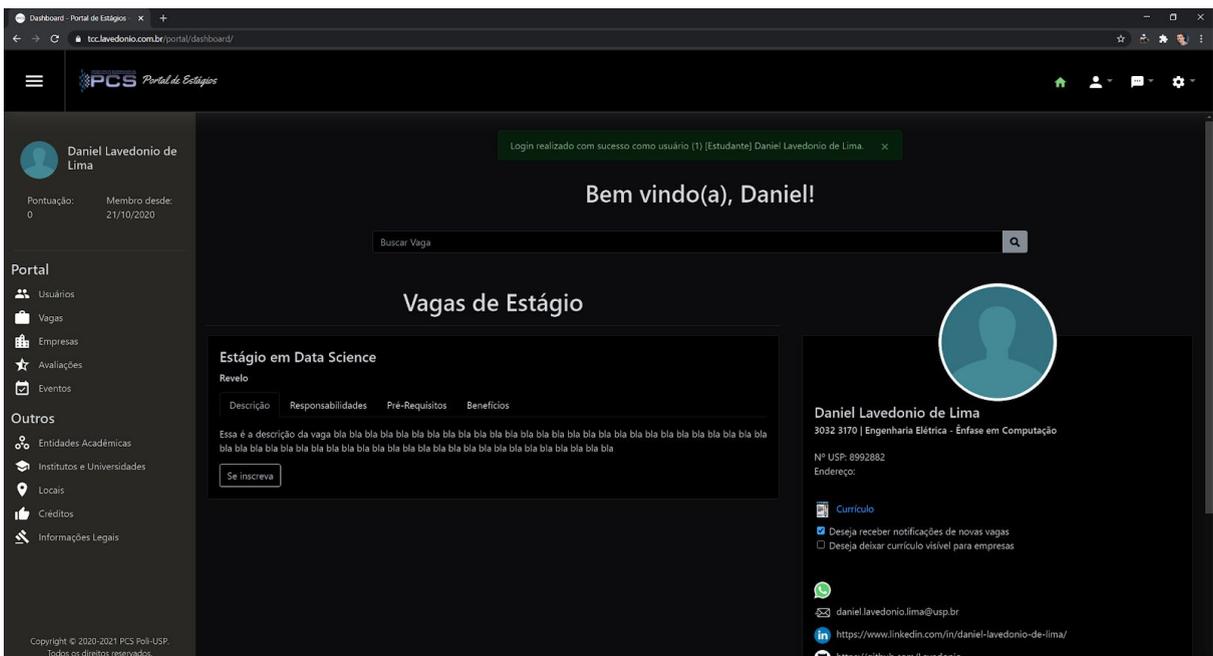


Imagem 23 - Passo 4 do processo de cadastro pela conta Google

No processo de cadastro, o e-mail vindo da conta autenticada pelo Google já é considerado verificado.

7.4. Cadastro de usuários

O cadastro na plataforma pode ser feito de duas formas: preenchendo o formulário de acesso ou entrando com uma conta Google. Para o usuário que escolher a primeira opção, ele seguirá os passos a seguir para entrar no sistema.

Passo 1: Ao clicar no botão “Acessar” na barra de navegação da sessão pública, o usuário do sistema irá para uma tela onde poderá escolher em qual categoria se encaixa: “estudante”, “empresa” ou “administrativo”.

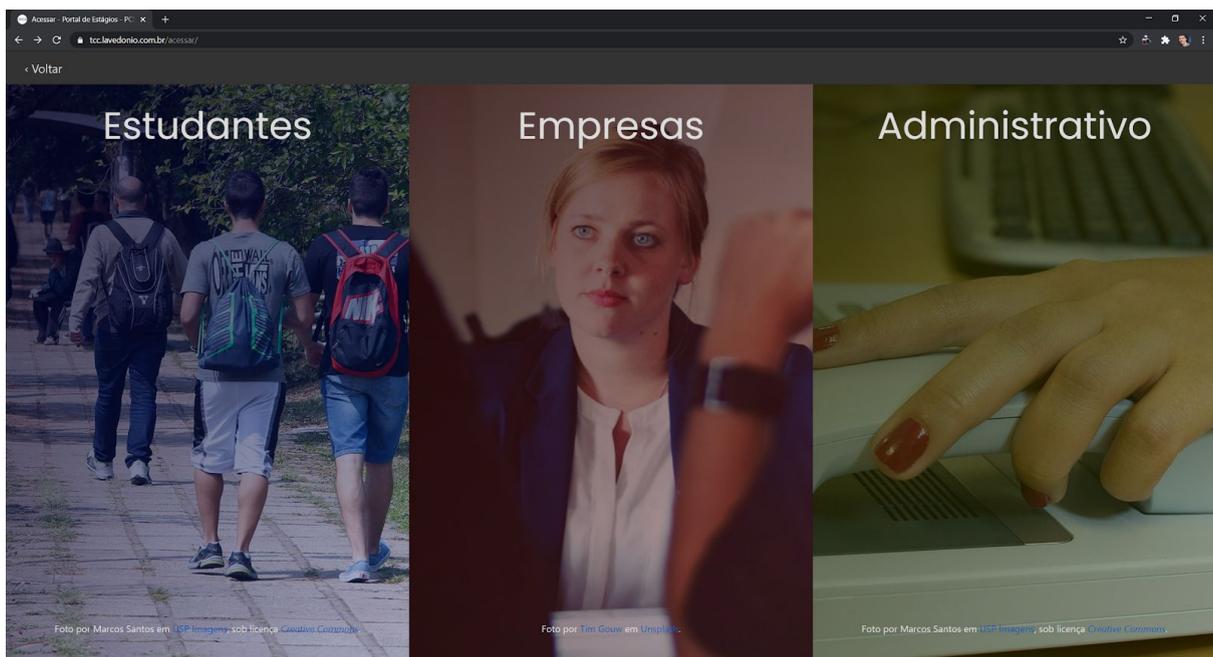


Imagem 24 - Passo 1 do processo de cadastro pelo email

Passo 2: Preencher o formulário de cadastro.

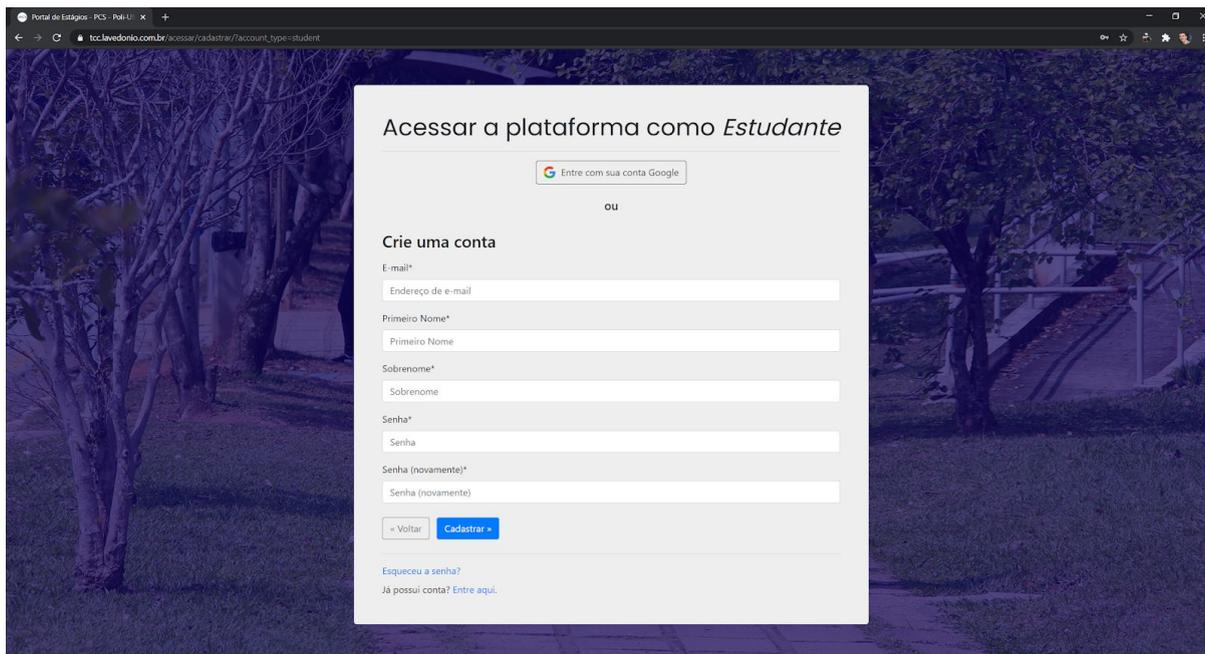


Imagem 25 - Passo 2 do processo de cadastro pelo email

Passo 3: Com as informações corretas, o usuário será redirecionado para uma página avisando-o que deve verificar o email. Enquanto isso não for feito, o usuário não conseguirá acessar o sistema.

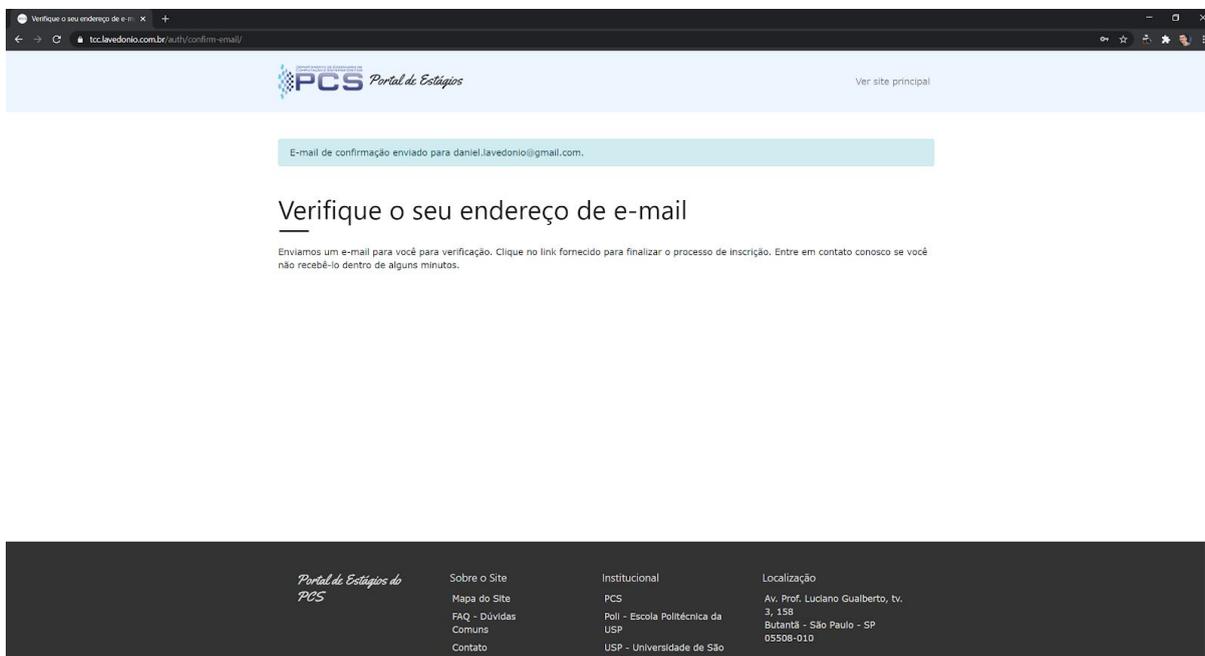


Imagem 26 - Passo 3 do processo de cadastro pelo email

Passo 4: O usuário deverá acessar seu email e verificar se a verificação chegou. Quando ela chegar, o usuário deverá clicar no link para ser redirecionado de volta ao site, para a página de confirmação do e-mail.

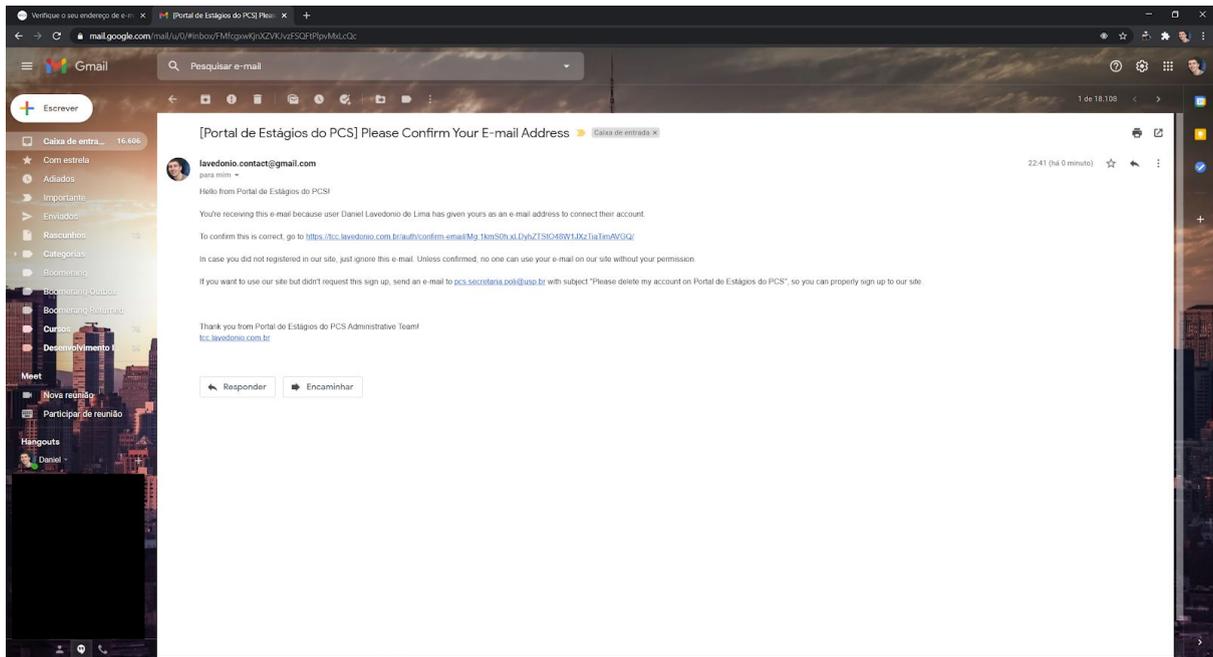


Imagem 27 - Passo 4 do processo de cadastro pelo email

Passo 5: Clicar no botão de confirmação.

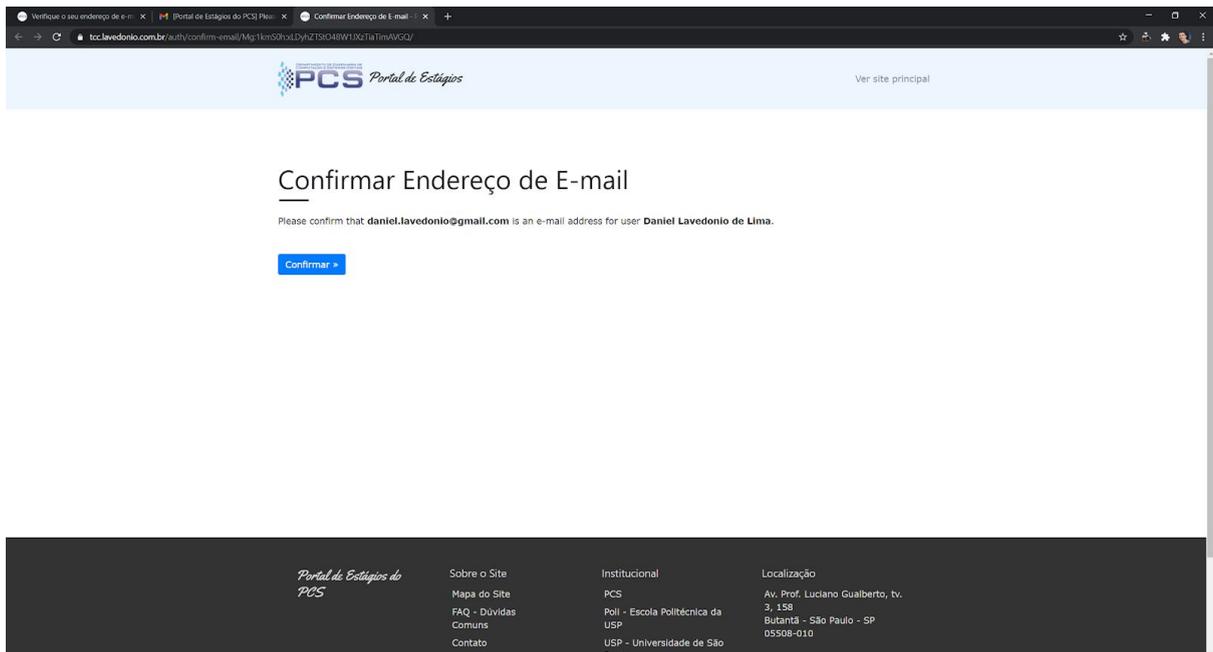


Imagem 28 - Passo 5 do processo de cadastro pelo email

Passo 6: Acessar o tipo de usuário que havia selecionado anteriormente e fazer o login na plataforma.

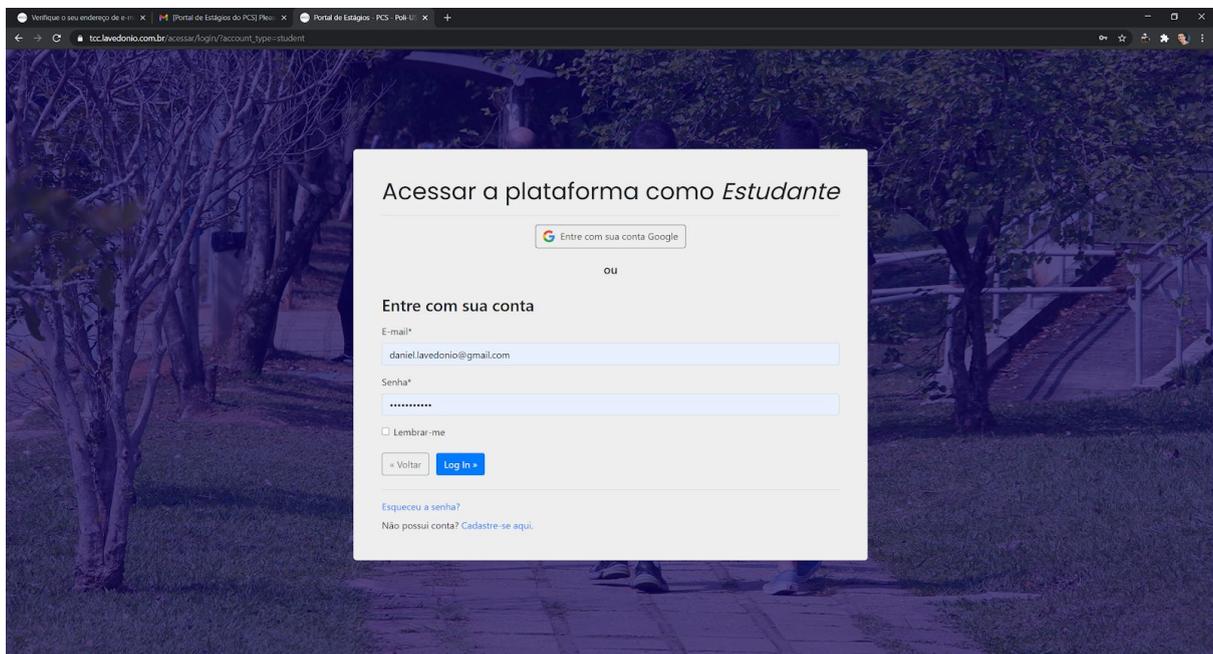
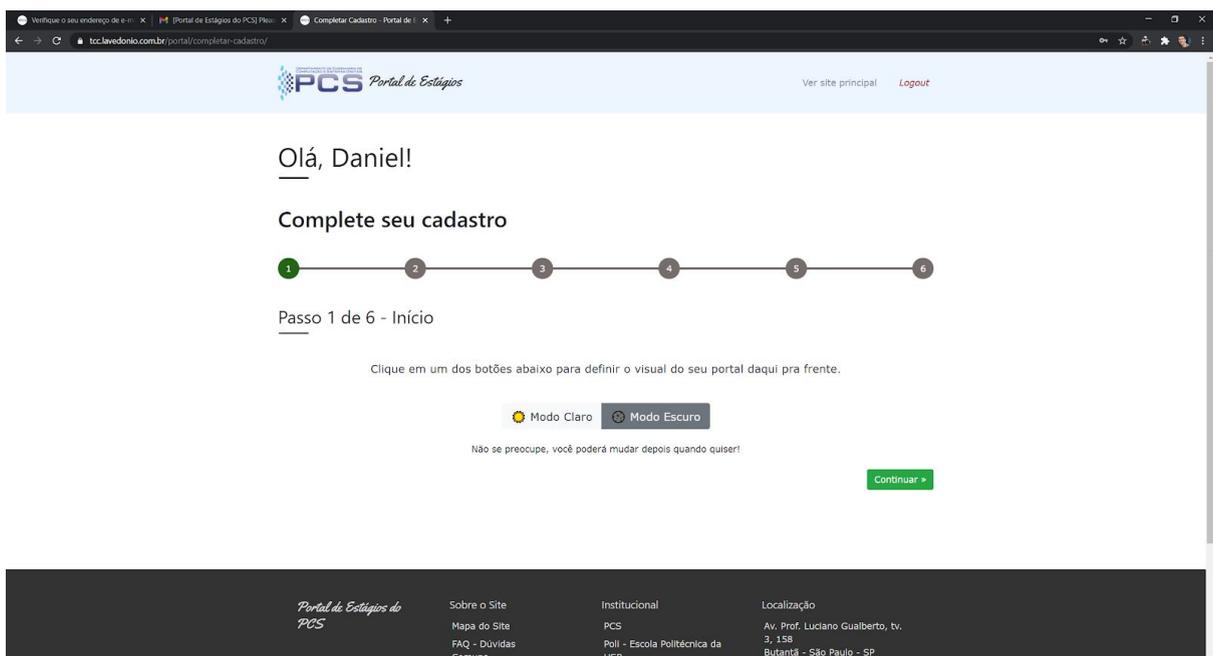


Imagem 29 - Passo 6 do processo de cadastro pelo email

Passo 7: Completar o cadastro na plataforma. Estes próximos passos irão de fato criar um perfil para o usuário, de forma com que ele nem perceba que a autenticação e o perfil de usuário são 2 classes distintas.



7.5. Alteração assíncrona de estado no back-end por meio de AJAX

Algumas alterações são feitas de modo com que os usuários não precisem carregar uma nova página para realizar qualquer ação no sistema. O modo como isso é atingido é por meio de requisições AJAX, em que um payload assíncrono é enviado para o back-end, que processa e retorna com um resultado.

Um exemplo em que se pode ver esse processo em ação é ao ativar e desativar o modo escuro no painel da plataforma. Esta edita a opção do usuário sem que este tenha que recarregar a página.

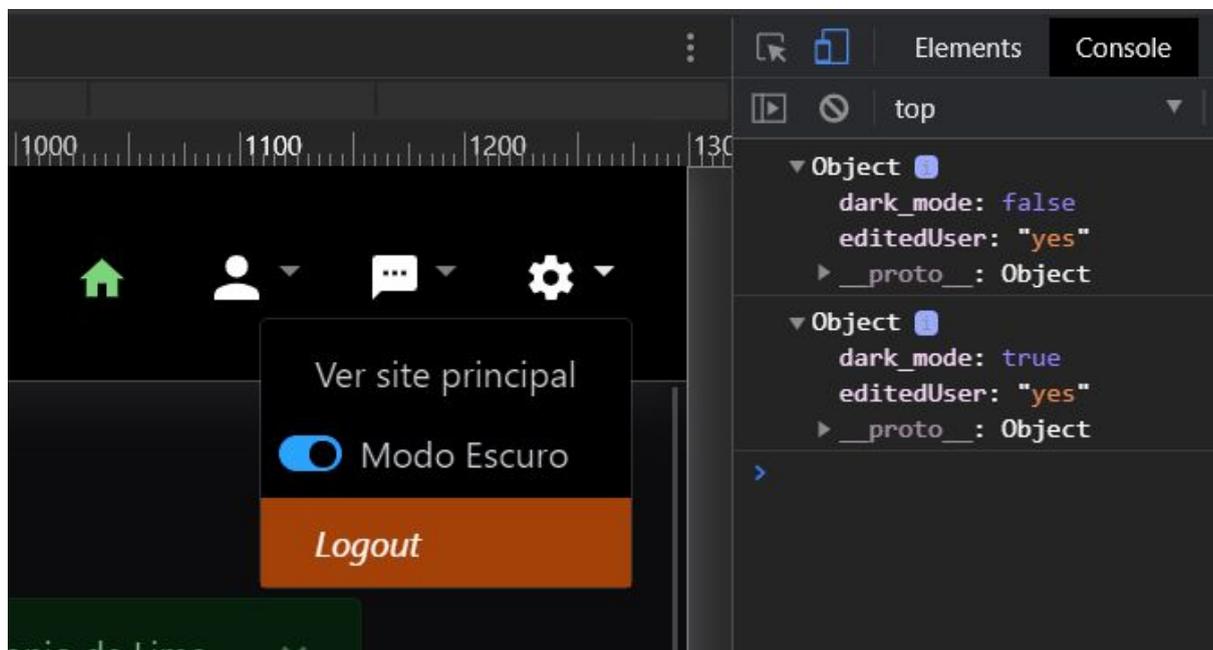


Imagem 31 - Vendo o payload por trás do acionamento do modo escuro

7.6. Dashboard do Aluno

O *dashboard* do aluno é um conjunto de páginas em que o aluno vai realizar buscas de vagas, editar seu perfil, ver perfis de empresas, avaliações, etc. É nessa seção que foram implementados os casos de uso que tem o Aluno como ator.

7.6.1. Dashboard do Aluno - Home

A *home* do dashboard é a primeira página que o aluno visualiza quando realiza o login. O objetivo dessa página é condensar as principais informações e utilidades que o aluno precisa.

a) Barra de busca

A barra de busca serve para o aluno pesquisar vagas e empresas.

b) Perfil

O aluno pode visualizar as informações mais importantes do seu perfil: Nome, foto de perfil, e-mail e Whatsapp, LinkedIn e Github, link para download do currículo e *checkboxes* se deseja receber notificações de novas vagas e se deseja tornar o currículo visível para empresas.

c) Vagas de estágio recomendadas

As vagas de estágio serão mostradas e através de barras de navegação o aluno poderá ver a descrição, responsabilidades, pré-requisitos e benefícios oferecidos. Contém também um botão para a inscrição da vaga, que redireciona para a opção de enviar um e-mail com o currículo ou uma página de recrutamento, dependendo da opção escolhida pela empresa.

d) Avaliações

Algumas avaliações de empresas por alunos serão mostradas nesta seção. A ideia é que o aluno possa ver quais são as melhores empresas para se trabalhar, de acordo com a opinião de ex-alunos que passaram por elas.



Imagem 32 - Dashboard do Aluno - Home Acesso em: 26 nov. 2020

7.6.2. Dashboard do Aluno - Vagas

Para o aluno ver a lista completa de vagas de estágio ele deve clicar em “Vagas”, no menu lateral ou em “Mais vagas”, no dashboard, ou fazer uma busca.

A página de vagas, vista pelo aluno, contém todas as vagas ativas publicadas pelas empresas, com o nome da empresa, título da vaga, salário, e um botão “Info” que o levará para a página da vaga, com mais detalhes sobre ela.



Imagem 33 - Página com vagas de estágio (Vagas e salários meramente ilustrativos)

Acesso em: 26 nov. 2020

A página com os detalhes da vaga conta com as informações relevantes da vaga para o aluno, com descrição, requisitos, habilidades desejadas, benefícios e link para inscrição no processo seletivo.

7.6.3. Outros

a) Empresas - A partir da home do *dashboard* do aluno, o aluno também poderá ver uma lista de empresas, semelhante à lista de vagas, em que ele poderá ver as vagas relacionadas à essa empresa, informações de contato, *website*, descrição, missão, visão e valores e as avaliações de alunos que passaram por essa empresa.

b) Informações sobre estágio - O menu lateral também conta com uma seção informativa, onde são dispostas informações a respeito de todo o processo de estagiar na Escola Politécnica: Documentações, regulamentos, relatórios de estágio e regras sobre que alunos podem estagiar.

c) Avaliações - De forma similar como foi apresentado na seção das empresas, as avaliações das empresas podem ser vistas tanto na home do *dashboard* do aluno como apresentadas em forma de uma lista acessada pelo menu

lateral. O intuito é o aluno conseguir ver quais são as empresas que os alunos mais gostaram de trabalhar e acessar também o perfil das empresas relacionadas.

7.7. Dashboard da Empresa

O *dashboard* da empresa é um conjunto de páginas em que o empregador vai editar o perfil da empresa, suas avaliações, ver perfis de alunos e acessar currículos, etc. É nessa seção que foram implementados os casos de uso que tem o Funcionário da Empresa como ator.

7.7.1. Dashboard da Empresa - Home

Quando o empregador da empresa faz o login, a primeira página que ele verá será a *home* do *dashboard*, que reúne as funcionalidades e informações mais importantes.

Funcionalidades:

- a) **Perfil:** Visualização do perfil da empresa, com logotipo, nome, links de contato e outras informações, bem como um botão para editar esses dados.

- b) **Vagas publicadas:** Uma lista com as vagas que a empresa publicou no Portal com a possibilidade de edição e fechamento da vaga.

- c) **Avaliações:** Lista de avaliações da empresa por alunos que fizeram estágio nela.

- d) **Criação de novas vagas:** A empresa poderá criar novas vagas de estágio que ficarão disponíveis no portal. Ao criar uma vaga, uma notificação via WhatsApp é enviada para os alunos interessados. Esse caso é discutido com mais detalhes na seção 6.7.2.

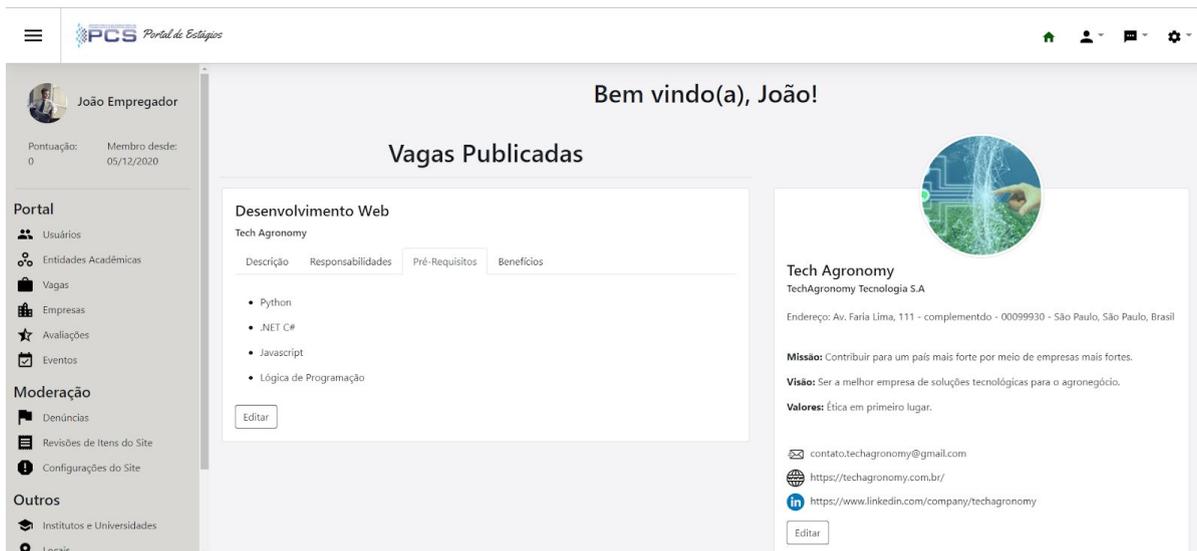


Imagem 34 - Dashboard da empresa - Home - Acesso em: 26 nov. 2020



Imagem 35 - Cards de avaliação dos alunos referente à experiência na empresa

7.8. Dashboard Administrativo

O *dashboard* administrativo foi feito para o funcionário do PCS realizar suas funções no site. É nesse espaço que o funcionário fará aprovação de cadastros de empresas e alunos, visualizar lista de perfis de alunos, empresas, vagas, experiências, universidades, departamentos e avaliações. É nessa seção que foram implementados os casos de uso que tem o Funcionário da Poli como ator.

O administrador terá o papel de moderador, tendo poder de remover perfis de empregadores e alunos, conceder ou bloquear permissões, e terá acesso aos dados relevantes para o PCS. Ele também tem a sua disposição algumas ferramentas para realizar atividades acadêmicas, como enviar mensagens diretamente para o aluno

para solicitar documentações, ou avisar sobre eventos relacionados a estágio. Também poderá mandar mensagens diretas para a empresa.

As mensagens do público geral que são enviadas pela página de contato também poderão ser lidas nesta seção.

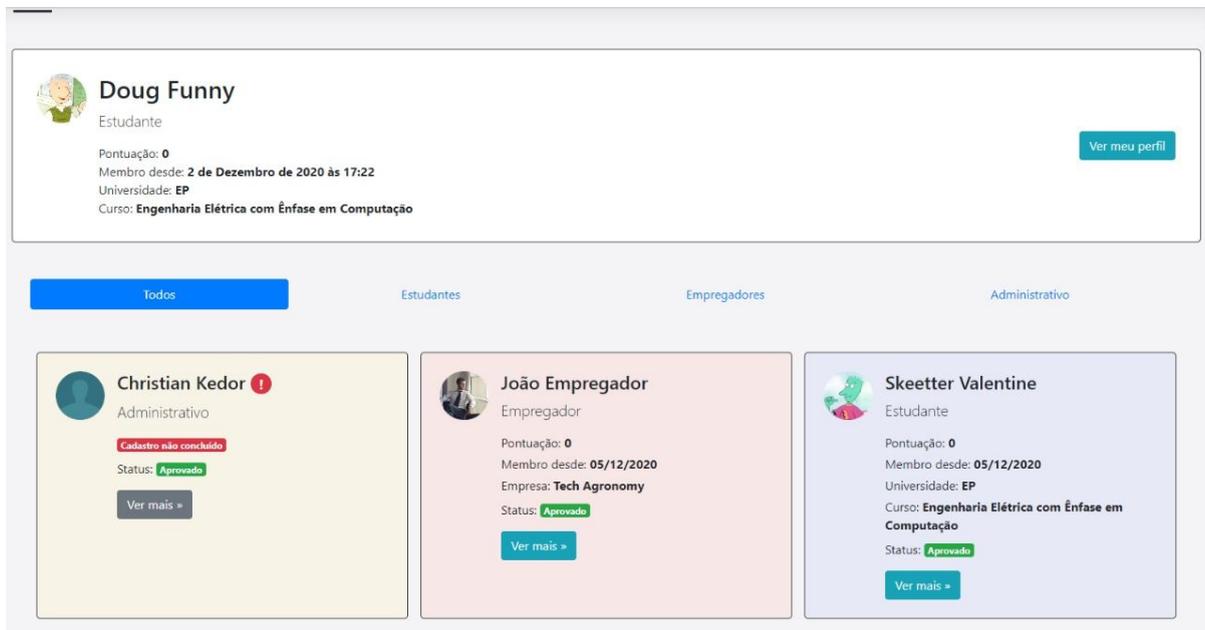


Imagem 36 - Exemplo de página para visualização de usuários pelo administrador.

Na imagem 6.5.1 é possível ver a tela de usuários vista pelo administrador. Diferentemente da situação em que quem está logado é um empregador ou um estudante, o administrador consegue visualizar todos os perfis cadastrados no sistema e a situação de aprovação de cada um. É a partir daí que ele conseguirá aprovar ou não novos cadastros.

7.9. Módulo de Mensagens WhatsApp

Para a criação da *feature* de envios de mensagens automáticas e recebimento de respostas dos alunos foi feita a construção de um módulo python utilizando a biblioteca Selenium para automatizar o acesso ao WhatsApp Web via *browser*. A classe é capaz de realizar envio de mensagens por WhatsApp recebendo como entrada a mensagem e o número do contato.

7.9.1. Classes WhatsAppBot e protótipos das funções

A seguir são apresentadas as interfaces para utilização do módulo WhatsApp.

Protótipo	WhatsAppBot.SetMessages(<i>mensagem</i> , <i>contato</i>)	
Descrição	Armazena o par mensagem/contato em uma lista para posterior envio.	
Parâmetros	<i>mensagem</i> : string	Mensagem que será enviada. A mensagem deverá ser em formato string com o caractere especial “\n” indicando quebra de linha.
	<i>contato</i> : string	Contato para qual a mensagem será enviada
Return	-	

Protótipo	WhatsAppBot.SendMessage()	
Descrição	Envia mensagens que foram previamente configuradas por SetMessages()	
Parâmetros	None	
Return	result: bool	True : o envio foi completado com sucesso False : Houve falha no envio ou não foi setada nenhuma mensagem

7.9.2. Uso do módulo WhatsApp

A utilização da classe **WhatsAppBot** é simples. Basta criar uma instância do objeto **WhatsAppBot**, utilizar a função **SetMessage** para configurar mensagens que serão enviadas (e para qual contato) e fazer o envio utilizando a função **SendMessages**, conforme demonstrado na figura 6.3.1.1

```
from whatsapp_bot import WhatsAppBot

bot = WhatsAppBot()
bot.SetMessage("11981479635", "Mensagem1")
bot.SetMessage("11981479635", "Mensagem2")

bot.SendMessages()
```

Figura 37 - test_bot.py - Arquivo exemplo de uso do módulo WhatsApp

A mensagem deve ser uma string com o caractere especial '\n' indicando quebra de linha.

7.9.3. Limitações

O módulo foi implementado utilizando-se a biblioteca python Selenium para automatizar o envio das mensagens pelo WhatsApp Web, via browser Google Chrome. Essa escolha foi feita porque é uma maneira gratuita de realizar essa tarefa. Para continuidade do projeto, essa função poderá ser trocada por uma API paga. A escolha dessa implementação traz algumas limitações, listadas abaixo:

QR Code - Toda vez que um browser novo (Sem dados de usuários pré-configurados) abre a página do WhatsApp Web, é pedido ao usuário a leitura de um QRCode. Isso significa que o número que irá enviar as mensagens do sistema deveria apontar para o QRCode toda vez que o código é executado. Para solucionar esse problema usamos um parâmetro para indicar qual diretório será utilizado para os dados de usuário do browser ao abrir o webdriver, e subimos esses dados junto com a aplicação ao fazer o *deploy* do sistema. Entretanto, é necessário realizar esse *set* manual uma vez, para gerar esses dados de configuração do usuário. Isso

implica que se for necessário mudar o telefone que gerará as mensagens deverá ser feito esse procedimento na máquina local e fazer um *deploy* do sistema novamente. A geração dos novos dados fica no arquivo `\whatsapp_bot\UserData\Default\LocalStorage\leveldb\000003.log`

Mudanças no WhatsAppWeb - Como não é uma API oficial e o Selenium navega no WhatsAppWeb por meio de classes html, qualquer atualização em uma nova versão deste aplicativo poderá alterar detalhes da página e atrapalhar o funcionamento.

WhatsApp deve estar aberto - O envio de mensagens no WhatsAppWeb só funciona se um dispositivo estiver com o WhatsApp instalado, aberto e rodando com a conta escolhida para o sistema. Por esse motivo a implementação e integração desse módulo com o sistema foi feita de modo “plugável”, podendo, no futuro, ser facilmente substituída por uma API paga para WhatsApp. Esse aspecto será abordado mais à frente, na seção 7, que discute recomendações de continuidade deste projeto.

7.10. Integração do Sistema com o WhatsApp

Nessa seção serão explicados os usos do módulo WhatsApp e como ele se integra no resto do sistema.

Para que o módulo de envio de mensagens automáticas via WhatsApp seja de fato plugável, é a aplicação que monta as mensagens que serão enviadas e depois chamam a interface do módulo Whatsapp para fazer o envio. Como o envio das mensagens pode ser demorado, essa tarefa é agendada e realizada em segundo plano, utilizando uma biblioteca Python chamada *background-tasks*. Isso também é útil para mensagens periódicas, que serão agendadas com dia e horário.

7.10.1. Funcionamento das Background-Tasks

As tarefas que rodam em segundo plano são criadas como funções no arquivo `site_core/background_tasks.py`.

Para criar uma tarefa basta utilizar o decorador `@background(schedule=60)` acima da *task* que deseja-se criar. O argumento *schedule* é o intervalo de tempo *default*, em segundos, que a tarefa será executada depois de ter sido criada.

Para usar a *task* basta chamar a função criada no ponto do sistema que se deseja, passando o argumento *schedule* se deseja agendar para outro intervalo de tempo que não seja o *default*. Quando a *task* é chamada, é criado um objeto do tipo *task* no banco de dados e um processo que roda no servidor em loop processa as *tasks* existentes. Se houver alguma *task* na fila, esse processo irá executar a função no horário agendado.

7.10.2. Task - Envio de nova vaga de estágio

No contexto do Portal, uma das situações de envio de mensagens de WhatsApp para o aluno é a criação de novas vagas de estágio. Para esse caso foi criada a função:

```
def SendJobOpportunityNotification(pk, schedule=10)
```

que é chamada no arquivo de modelos da empresa, **companies/models.py**, no *model JobOpportunity*, quando o objeto que representa a vaga de estágio é criado. Portanto, para cada nova vaga criada por um empregador, uma nova *task* com essa função é agendada para rodar em segundo plano.

A função tem como parâmetro o *pk* (*Primary Key*) que identifica a vaga criada para poder puxar as informações dela no banco de dados. Em seguida, a função gera mensagens para os alunos com descrição, requisitos, responsabilidades, benefícios e link para se candidatarem à vaga. Por fim, a *task* faz o envio das mensagens utilizando o módulo WhatsApp discutido anteriormente.

```

for student in students:
    phone = student.main_phone_number
    name = student.first_name
    message = "Olá, {}!\nA empresa {} abriu uma nova vaga de estágio para {}!\n\n".format(name, company_name, title) + \
        "Descrição da vaga: {}\n\n".format(description) + \
        "Responsabilidades:\n{}\n\n".format(responsabilities) + \
        "Pré-requisitos:\n{}\n\n".format(req_skills) + \
        "Habilidades Desejadas:\n{}\n\n".format(des_skills) + \
        "Data de inicio: {}\n\n".format(start_date) + \
        "Data de termino: {}\n\n".format(end_date) + \
        contact_message

    bot.SetMessage(phone, message)

bot.SendMessages()

```

Imagem 38 - Trecho do código que monta a mensagem e chama o módulo WhatsApp. É importante ressaltar que é nesse ponto que uma possível API do WhatsApp poderia entrar, no futuro, substituindo a função `bot.SetMessage(phone, message)` e `bot.SendMessages()`

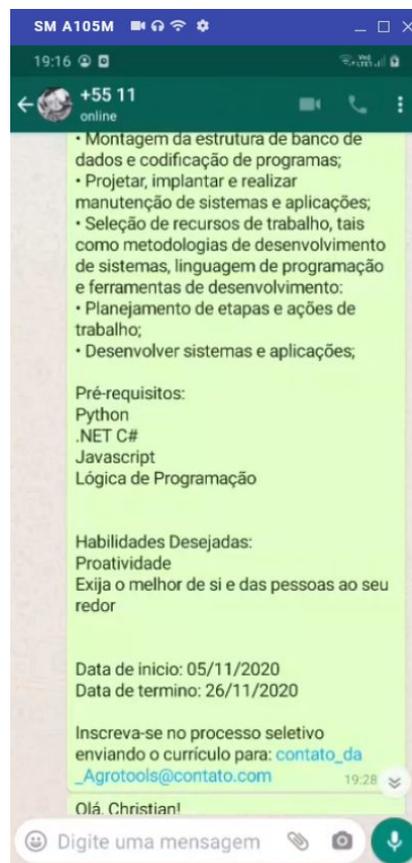


Imagem 39 - Exemplo de mensagem de divulgação de nova vaga de estágio vista no WhatsApp em tela de celular.

8. TESTES E AVALIAÇÃO

Para testarmos a aplicação, além de fazermos testes manuais com os casos de uso e funcionalidades previstas, foi estruturado um teste com usuários reais.

O teste para o aluno será realizado com alunos da Poli que não conhecem a aplicação e um roteiro será fornecido para o aluno realizar atividades no site. Ao fim o aluno poderá explorar o site livremente e fazer considerações.

As seguintes instruções serão passadas para o aluno:

- a) Acesse o site e faça um cadastro como aluno.
- b) Confirme o cadastro e crie um perfil. Coloque seu número real de WhatsApp e permita receber notificações.
- c) Edite o seu perfil para incluir um link no Github
- d) Acesse o website de uma empresa cadastrada no site e visualize seu perfil e explore as vagas postadas
- e) Finja que já fez um estágio em uma das empresas listadas. Crie um relato fictício de sua experiência, com uma avaliação para essa empresa.
- f) Ache sua avaliação e a avaliação de outros alunos para a empresa
- g) Enquanto faz seu teste, uma das empresas do site irá criar uma vaga. Verifique seu WhatsApp para ver se recebeu a mensagem. Verifique se essa vaga apareceu no seu *dashboard*.
- h) Suba o seu currículo no site (Se não tiver um currículo, pode subir um arquivo word ou pdf qualquer) e permita que as empresas o acessem.
- i) Explore o site livremente.

O teste será filmado e trecho dele será apresentado na demonstração prática do projeto. Os testes foram utilizados para achar *bugs* e corrigir problemas de usabilidade não previstos ao longo do projeto.

9. CONSIDERAÇÕES FINAIS

9.1. Conclusões do Projeto de Formatura

Este projeto foi fruto de uma necessidade sentida na pele pelos autores. Ambos os alunos passaram por processos de estágio obrigatório no curso de Engenharia Elétrica com Ênfase em Computação e perceberam vários procedimentos que poderiam ser melhorados. A decisão do projeto também veio de um sentimento de deixar algum legado para a Escola.

Do ponto de vista acadêmico, o sistema desenvolvido foi muito útil não só para aplicar os conhecimentos adquiridos ao longo do curso como também para aprender tecnologias utilizadas hoje em dia no mercado, como framework Django, a linguagem Python e a biblioteca Selenium.

Os autores acreditam que uma plataforma de estágios em uma grande universidade é algo essencial e faz parte da realidade de um mundo digital em que vivemos.

9.2. Perspectivas de Continuidade

Como mencionado anteriormente, o *back-end* foi pensado e concebido para dar suporte a um projeto com uma abrangência maior de funcionalidades do que as que foram implementadas. A seguir são apresentadas algumas ideias de continuidade do projeto para melhorar ainda mais a experiência dos alunos, das empresas e da universidade no processo de estágio.

9.2.1. API WhatsApp e extensão do uso

Como mencionado anteriormente, o módulo de WhatsApp poderá ser facilmente substituído por uma API paga, com pequenas modificações no *back-end*. Esse aspecto foi discutido nas seções 7.6 e 7.7.

Além disso, o uso dessa funcionalidade pode ser expandido. É possível implementar inteligências no sistema utilizando processamento de linguagem natural para não só fazer o envio de notificações mas para ter uma conversa com o

aluno, obtendo resposta do aluno para fazer envio de currículo para empresa, para indicar interesse em vagas e tirar dúvidas do aluno sobre documentação.

9.2.2. Documentação e Procedimentos Burocráticos

Apesar de contar com uma seção com algumas informações sobre o processo de estágio, o foco do projeto não foi nos processos burocráticos de estágio, como emissão de documentos e entrega de relatório.

Uma forma de simplificar o trabalho do funcionário da Poli e melhorar a experiência tanto das empresas quanto dos alunos é unificar tudo que for relacionado à estágio em apenas uma plataforma e realizar procedimentos burocráticos digitalmente. Algumas funcionalidades foram pensadas:

- Criação de template de relatório de estágio e contrato de estágio baseado em dados do aluno.
- Assinatura e troca de documentos oficiais de estágio entre aluno, empresa e faculdade via plataforma (Contrato, termo de rescisão, termos aditivos).
- *Status* do aluno: procurando emprego, em processo de contratação ou contratado
- Acompanhamento do aluno na empresa (Computação de horas trabalhadas, etc).

9.2.3. Eventos e Notificações

Além do processo de procura de vagas em si, o departamento realiza diversos almoços, palestras e outros eventos com o objetivo de aproximar os estudantes e as empresas. Um dos requisitos levantados e que chegou a ser modelado no banco de dados do projeto foi a possibilidade de criar eventos e notificar os estudantes que compõem o público-alvo de tal.

9.2.4. Entidades Acadêmicas

A partir da entrevista com o gerente do WI, notou-se que as entidades acadêmicas possuem grande interesse de ter um ponto em destaque para fazer a divulgação de vagas para os alunos. Muitas delas recebem das empresas

oportunidades exclusivas para divulgação e faltam meios para se conectar mais diretamente com os alunos.

Esta relação chegou a ser mapeada no diagrama de classes e modelada no banco de dados, mas não foi possível implementá-la dada as restrições de tempo. As pendências foram:

- Adição de uma nova Entidade Acadêmica por parte dos alunos;
- Adição de vagas “promovidas” pela Entidade Acadêmica;
- Mudança de comando interno quando houver troca de membro.

9.2.5. Utilização inteligente dos dados

O *back-end* foi estruturado para muitos tipos de dados serem armazenados, uma quantidade muito além do necessário para implementação de funcionalidades do presente projeto. Isso foi feito com o objetivo de expansão em mente. Esses dados poderão trazer muito valor à plataforma com o uso de algoritmos de machine learning, e algumas funcionalidades foram pensadas:

- a) Utilização de dados de localização, interesses, habilidades, experiências anteriores, pretensão salarial e outros para recomendações inteligentes de vagas de estágio. Com essa *feature* os alunos vão ter acesso às vagas que realmente são relevantes para ele e a empresa consegue atingir o perfil que mais se encaixe a suas necessidades.
- b) Utilização de dados de experiências anteriores dos alunos e requisitos para as vagas atuais. O sistema conterà em seu banco de dados muitas informações sobre as principais tecnologias que o mercado pede de um estagiário atualmente e isso pode ser uma importante fonte de informação para a faculdade direcionar o curso de computação e se manter atual e relevante.

9.2.6. DialogFlow

Para a parte de interpretação e análise de fluxo de conversa do usuário, foi feito um pequeno estudo com o DialogFlow, um serviço da Google Cloud Platform que permite com que as chamadas de API sejam mais simples e rápidas para processamento da plataforma.

Esse serviço permite abstrair boa parte do processo de aprendizado de máquina e dá um suporte mais robusto e custo-efetivo ao projeto.

REFERÊNCIAS

- [1] Portal de Estágios do Departamento de Engenharia de Computação e Sistemas Digitais da Escola Politécnica da Universidade de São Paulo, São Paulo. Local de estudo e desenvolvimento da solução. Disponível em: <<http://estagios.pcs.usp.br/>>. Acesso em: 14 abr. 2020.
- [2] Pressman, R. S.; **Software Engineering: A Practitioner's Approach** McGraw-Hill Science/Engineering/Math; 8ª edição - 23 janeiro 2014
- [3] INTERNATIONAL ORGANIZATION FOR STANDARDIZATION. **ISO/IEC 10746-3:2009**: Information technology — Open distributed processing — Reference model: Architecture — Part 3
- [4] Wazlawick, R. S.; **Engenharia de Software – conceitos e práticas**. Ed. Elsevier e Campus. 2013.
- [5] NOVAS, C. A. M. Enquetes e entrevistas: entendendo o Painel de Vagas do 30º WI e como aproveitar os aprendizados. [Entrevista a Christian Braga Kedor e Daniel Lavedonio de Lima]. Google Meets, set. 2020
- [6] Feira Virtual, 30º Workshop Integrativo da Poli Júnior. Disponível em: <<https://verosolutions.6connex.com/event/feiravirtual/wi30/login>>. Acesso em: 3 set. 2020.
- [7] BECERRA, J. L. R. Enquetes e entrevistas: entendendo as necessidades de um portal de estágios. [Entrevista a Christian Braga Kedor e Daniel Lavedonio de Lima]. Google Meets, jun. 2020
- [8] RODRIGUES, G. C. Enquetes e entrevistas: entendendo melhor processo de procura de estágios. [Entrevista a Daniel Lavedonio de Lima]. Google Meets, out. 2020
- [9] RANGEL, L. B. Enquetes e entrevistas: entendendo melhor processo de procura de estágios. [Entrevista a Daniel Lavedonio de Lima]. Google Meets, out. 2020
- [10] NERI, A. P. Enquetes e entrevistas: entendendo melhor a busca de estagiários para o profissional de RH. [Entrevista a Daniel Lavedonio de Lima]. Google Meets, ago. 2020
- [11] KEDOR, C. K.; LIMA, D. L. pesquisa por acesso remoto/ internet: entendendo melhor processo de procura de estágios. Google Forms, nov. 2020
- [12] MELL, P.; TIMOTHY, G. **The NIST Definition of Cloud Computing**. Gaithersburg: National Institute of Standards and Technology, September 2011. (Special Publication 800-145).

- [13] Smith, L. W. **Stakeholder analysis: a pivotal practice of successful projects.** Trabalho apresentado em Project Management Institute Annual Seminars & Symposium, 2000, Houston. Não publicado.

GLOSSÁRIO

As definições a seguir relacionadas foram extraídas do National Institute of Standards and Technology (NIST)^[12] e do Project Management Institute (PMI®)^[13].

Platform as a Service (PaaS) - a capacidade fornecida ao consumidor de implantar na infraestrutura de nuvem aplicativos criados ou adquiridos pelo consumidor, criados usando linguagens de programação, bibliotecas, serviços e ferramentas com suporte do provedor. O consumidor não gerencia ou controla a infraestrutura de nuvem subjacente, incluindo rede, servidores, sistemas operacionais ou armazenamento, mas tem controle sobre os aplicativos implantados e possivelmente as definições de configuração do ambiente de hospedagem de aplicativos.

Stakeholder - Indivíduos e organizações que estão ativamente envolvidos no projeto, ou cujos interesses podem ser positivamente ou negativamente afetados como resultado da execução do projeto ou da conclusão bem-sucedida do projeto.

