

PROJETO TCC – BRAZIL CAREER HUB

Carlos Eduardo dos Santos Ferreira
Gabriel Menezes de Antonio
Gustavo Henrique dos Santos Faria
Mayara Marques Pereira de Souza
Elio Caruso Filho – Orientador

RESUMO

O objetivo do projeto é desenvolver um site para auxiliar na inserção de estrangeiros e brasileiros no mercado de trabalho do Brasil. No desenvolvimento do método para a criação do website, a primeira decisão foi definir o tipo de arquitetura cliente e servidor, em seguida as linguagens de programação utilizadas, sendo elas o framework React com a linguagem Typescript no front-end, uma API REST Django e a linguagem Python no back-end na parte do servidor junto com as regras de negócio e o MySQL como sistema de Gerenciamento de Banco de Dados, sendo todos eles gerenciados por um repositório privado no GitHub. A criação do protótipo do design foi feita com a ferramenta Figma, pensando na melhor experiência para o usuário e o website inteiro foi feito em cima dessas decisões. Como resultado, o site de vagas de emprego Brazil Career Hub obteve uma estrutura completa com todas as informações necessárias para a inserção no mercado de trabalho brasileiro para qualquer pessoa interessada, mas com foco no público estrangeiro, trazendo páginas com explicações sobre legislações e normas específicas do país, promovendo a imigração de mão de obra qualificada no país. Em conclusão, após revisão de dados obtidos em buscas de sites oficiais do governo brasileiro, desenvolveu-se o website Brazil Career Hub para contribuição no preenchimento com mão de obra qualificada de vagas de emprego no Brasil, independente se o candidato à vaga reside ou não no país.

Palavras-chave: Imigrantes; Website; Emprego; Refugiados; Mão de obra qualificada.

1. INTRODUÇÃO

1.1. Objetivo Geral

O presente trabalho é sobre a busca e preenchimento de vagas de emprego e tem como objetivo a criação de um website para as pessoas que estão procurando emprego no território brasileiro, em que elas possam identificar vagas, leis e normas relacionadas a documentações, independente de já residirem no país ou não, tendo como foco maior os imigrantes. Desta forma o website terá como principal funcionalidade a listagem de vagas, ademais a sua diversidade de dados auxiliará na redução de desemprego, resultando em geração de oportunidade para todos e preenchendo vagas com mão de obra qualificada. Com a crescente globalização, muitas pessoas estão buscando oportunidades de trabalho em países estrangeiros, e a falta de informação e barreiras culturais podem dificultar esse processo. Além disso, muitas vezes empresas têm dificuldades em encontrar talentos com habilidades específicas.

1.2. Objetivo Específico

Os objetivos específicos do projeto são os listados:

- Criação de um website com a arquitetura cliente e servidor
- Desenvolver um website utilizando frameworks React, Django e REST
- Auxiliar imigrantes a encontrarem empregos no Brasil por meio de um website acessível para todos
- Normalizar o entendimento referente a legislações e normas brasileiras

1.3 Justificativa

Existem alguns sites que focam em empregos para imigrantes, porém poucos citam vagas sobre o Brasil. Segundo o artigo de Freitas (2023), atualmente nos Estados Unidos possuem aproximadamente 28 milhões de imigrantes, o que é um valor muito grande comparado ao número no Brasil representando apenas 181.385 imigrantes registrados em 2020 (TST, 2023). Com um website específico para imigração para o Brasil, poderíamos facilitar a imigração trazendo múltiplos benefícios, como por exemplo a diminuição do desemprego que, segundo o IBGE (2022), a taxa se encontra em 7,9% no 4º trimestre de 2022. Essa taxa abaixaria, pois o nosso site seria mais uma fonte de empregos.

Além disso, informações sobre processos do nosso dia a dia podem se mostrar difíceis de ser encontradas, como por exemplo a declaração do imposto de renda, em 2022 oitenta mil goianos ainda não havia declarado seu imposto de renda com apenas 1 dia para o fim do prazo (Alexandria, 2022). Sendo assim, acreditamos que o site poderá ajudar pessoas a nível nacional e internacional, diminuindo o desemprego e combatendo a desinformação.

2. DESENVOLVIMENTO

2.1. Github

Para mantermos um controle de versão e suportar o desenvolvimento simultâneo do projeto utilizamos o GitHub como o principal local para armazenar e organizar o código. O GitHub é um serviço baseado em nuvem que hospeda um sistema de controle de versão (VCS) chamado Git, o qual é utilizado para controlar o histórico de alterações de arquivos e principalmente de projetos, permitindo mais flexibilidade no fluxo de trabalho, segurança e desempenho. De todas as funcionalidades que o serviço disponibiliza, as principais utilizadas foram: workflows, branches, issues, pull requests, projects e controle de dependência automatizada.

Um workflow é um conjunto de processos e ferramentas automatizadas que auxiliam no controle de qualidade do código, o projeto utiliza três principais workflows. Um para a avaliação do ambiente de frontend, outro para a avaliação do código Python escrito e outro para a avaliação do ambiente de backend.

O workflow para avaliação do ambiente de frontend consiste em compilar o código utilizando a própria biblioteca do Node. Já o fluxo para avaliação do código Python consiste na utilização de uma biblioteca de padrão de código chamado Pylint, e por fim, o workflow para validação do ambiente de backend consiste na atualização da estrutura do banco de dados, coleta de arquivos estáticos, avaliação do código e execução de casos de testes.

As branches são ramificações do projeto que possuem o objetivo de isolar desenvolvimentos. O nosso projeto possui duas principais ramificações, uma para desenvolvimento e outra para produção. A ramificação de desenvolvimento é destinada para o uso cotidiano, onde todas as ramificações pessoais (utilizado para o desenvolvimento ativo) são baseadas nessa branch, permitindo um controle ainda maior nas alterações. Já a ramificação de produção tem o objetivo de centralizar o ambiente e o código que estará publicamente disponível para o usuário final.

Os issues (do Inglês, problemas) são “tickets” abertos para controle de tarefas e problemas que devem ser corrigidos, neles são descritas todas as informações necessárias para sua correção. Esses issues são os principais agentes para o controle de tarefas e pendências do projeto.

Conforme comentado anteriormente, projeto possui algumas ramificações para permitir o isolamento do desenvolvimento. Quando o desenvolvimento é finalizado, as

alterações dessas ramificações devem ser mescladas com as alterações existentes da ramificação de origem, para realizar essa mesclagem é necessária a abertura de um pull request, essa requisição é um método para permitir que outros desenvolvedores revisem o novo código antes de ser oficializado numa branch geral, neles também são executados os workflows, os quais validam a qualidade do código de forma automatizada para que a revisão por outros desenvolvedores possa ser mais rápida e fácil.

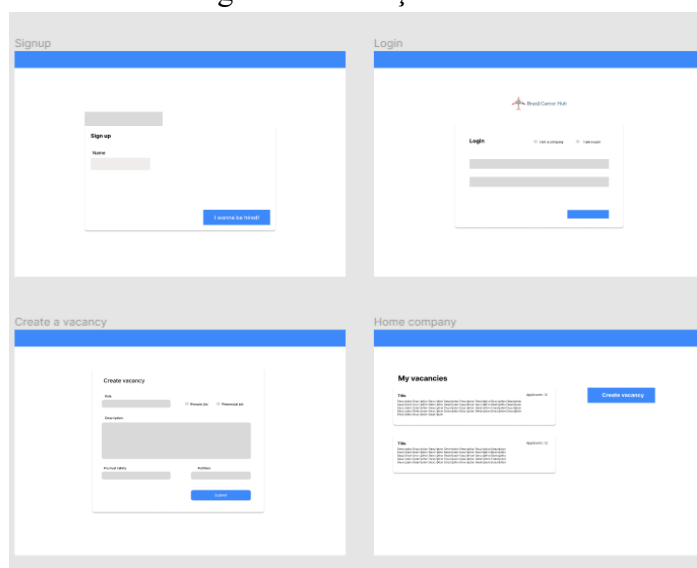
A funcionalidade de projects é algo que foi recentemente adicionado ao GitHub, permitindo uma exibição, ordenação e gerenciamento melhor, mais otimizado e persistente dos issues, conforme comentado anteriormente, os issues são constantemente utilizados para manter um controle referente aos problemas e pontos pendentes do projeto, permitindo criar diversos recursos visuais e automatizados para facilitar esse controle. O nosso projeto utiliza um board geral, dois específicos (para cada ambiente) e um roadmap, permitindo que o desenvolvedor tenha uma melhor visão de suas tarefas em andamento, e pendentes.

Por fim, mas não menos importante, temos o controle de dependência automatizada. O autômato cria pull requests ao detectar uma nova versão de alguma biblioteca ou dependência, além disso, também aponta falhas de segurança e outros problemas relacionados a dependências e pacotes externos utilizados, mantendo uma maior qualidade, atualidade e segurança no projeto.

2.2. Front-End

No desenvolvimento do nosso projeto do TCC, um dos problemas que tivemos que pensar foi a interface do sistema, com o nosso propósito de construir um site de empregos a nível mundial focado para vagas do Brasil, precisaríamos de uma interface simples de se usar, fácil de se aprender e sem muita burocracia para que os candidatos não desistam facilmente de criar sua conta e aplicar-se para uma vaga criada no nosso site. Então, para planejarmos sobre, criamos alguns esboços das telas utilizando o software Figma, com ele pudemos decidir as cores principais do site e alguns layouts de tela para as funcionalidades que o usuário irá interagir conforme figura 1.

Figura 1 – Esboço de telas



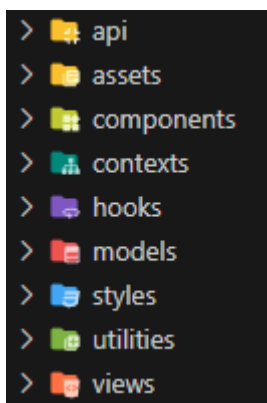
Fonte: Autoria própria

Para o desenvolvimento da interface decidimos usar o React, pela sua facilidade de criação de telas e quantidade de bibliotecas que podem ser integradas com o intuito de facilitar ainda mais a criação de pequenos componentes já com padronizados. O React é uma biblioteca open source criada pela empresa Meta com a ajuda da comunidade, lançado em 29 de maio de 2013.

Sobre as bibliotecas que utilizamos com o React, a primeira delas é a “styled-components”, uma biblioteca que permite utilizarmos estilização para componentes em específico sem que seja utilizada uma estilização global importando arquivos CSS, como é feito por padrão no React sem qualquer biblioteca. Também utilizamos react-router-dom, essa biblioteca nos permite mapear rotas do navegador para componentes do React, por exemplo, quando o usuário acessar o domínio com o caminho /home, um componente gerado será renderizado. A biblioteca mais importante utilizada no nosso projeto foi a Material UI, ela nos fornece diversos componentes prontos, a possibilidade de reutilização de componentes auxiliou muito no tempo de desenvolvimento.

A estrutura do projeto front-end foi pensada visando separar cada responsabilidade do código fonte, para que seja fácil escalar e implementar novas funcionalidades.

Figura 2 - Estrutura front-end



Fonte: Autoria própria

Na pasta de api temos todas as funções que foram implementadas para realizar requisições para o back-end, cada sub-diretório contém chamadas de acordo com sua entidade. A pasta de assets do projeto contém todos os arquivos de mídia utilizados no projeto, como imagens, áudios, vídeos, gif etc.

Já a pasta de components contém todos os componentes que podemos criar utilizando React, esses componentes são trechos de código HTML, CSS e Javascript que podem ser reutilizados em diversas partes do nosso sistema, sendo extremamente poderoso e simples de ser feito, os componentes também podem receber propriedades, tornando-o personalizável para cada parte do código.

A pasta de contexts possui contextos que podem ser utilizados em qualquer parte da aplicação, permitindo salvar dados e funções que serão reutilizados, evitando passar dados via componentes. Os contextos permitiram a redução da complexidade do projeto e a melhor manutenibilidade do mesmo, gerando mais segurança e estabilidade ao front-end.

A pasta de hooks contém funcionalidades que podem ser reutilizadas sobre os componentes, hooks são funções facilitadas que podem ser utilizadas para diversos princípios, como buscar dados, configurar variáveis, executar funcionalidade em qualquer ciclo de um componente, etc. No nosso projeto criamos um para que certas páginas do site só fossem acessadas após o usuário ou empresa terem sido autenticadas.

A pasta de models contém todas as estruturas de dados que foram mapeadas durante o desenvolvimento da aplicação, possuindo também algumas funcionalidades sobre a própria estrutura. Essa parte do código fonte está relacionada com diversas outras partes do sistema, como por exemplo páginas que simbolizam essas estruturas, chamadas de API que retornam estruturas mapeadas, criação de modelos no back-end etc.

A pasta de styles contém tudo o que é relacionado com a estilização global da aplicação, dentro do arquivo global.css podem conter códigos CSS que irão ser utilizados no projeto inteiro. Já o arquivo theme.ts contém a estilização global relacionada ao framework Material UI, o mesmo além de nos fornecer diversos componentes prontos, ele também nos dá a possibilidade de estilizarmos o mesmo, como alterar cores, tamanho de fontes, etc.

A pasta de utilities possui todos os arquivos que contém funcionalidades úteis que possam ser reutilizadas ao longo do projeto como por exemplo, formatação de datas, formatação de campos, utilidades para textos, etc.

Por fim, a pasta de views contém todas as páginas do site, nela são armazenadas diversos componentes, tais como as próprias páginas. Cada página deve ter uma rota que deve ser mapeada no arquivo “router.ts” que se encontra na raiz do projeto, dentro desse arquivo contém uma lista com vários itens nos quais devem mapear um componente com sua rota.

2.3. Back-End

O ambiente backend é responsável por armazenar os dados de todas as empresas, usuários, vagas, currículos e por disponibilizar esses dados para a interface. A aplicação do backend é uma API utilizando o framework Django codificado na linguagem de programação Python. A aplicação também implementa o framework REST, encapsulado como uma biblioteca otimizada para ser utilizada juntamente com o framework do Django. A aplicação faz uso de um banco de dados relacional MySQL. para facilitar o processo e reduzir os recursos necessários durante o desenvolvimento, utilizamos uma variação independente do MySQL, o sqlite, o qual permite persistir os dados em um banco de dados relacional não precisando hospedar ou se conectar a um servidor.

A API disponibiliza 19 endpoints com 15 deles possuindo operações CRUD completas, para facilitar o desenvolvimento o backend possui uma documentação completa de todas as APIs com uma interface gráfica utilizando o Swagger, o qual é um conjunto de ferramentas de API na qual a especificação OpenAPI é baseada.

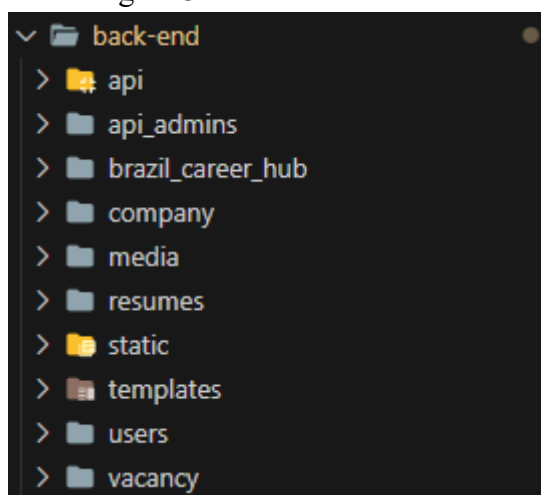
Além da interface gráfica, também implementamos uma coleção completa de endpoints hospedada no Postman, uma plataforma de API para desenvolvedores. Permitindo realizar chamadas para os endpoints do backend e avaliar as respostas, os parâmetros e outros pontos necessários para o desenvolvimento de uma API. A aplicação possui um método de autenticação do tipo Bearer Token, o qual é um tipo de token de acesso em uma API utilizado

para autenticação e autorização. Os tokens Bearer são gerados com base em protocolos e especificações como OAuth e JWT (JSON Web Token).

Toda requisição para o ambiente de backend e seus endpoints devem possuir um token, caso não seja informada, o acesso não será autorizado. Para gerar um token, o usuário deve possuir uma conta de administrador no ambiente de backend e solicitar a geração de um token de acesso a partir de um endpoint único, o qual deve se autenticar utilizando usuário e senha.

Toda a camada de endpoint, autenticação e resposta é tratada pelo próprio framework do Django com a ajuda do framework do rest. A arquitetura do ambiente de backend foi modularizada e seccionada cuidadosamente para criar uma arquitetura que forneça uma melhor manutenibilidade e escalonamento.

Figura 3 - Estrutura back-end



Fonte: Autoria própria

A pasta `api` possui os arquivos responsáveis pela informação geral da API. A pasta `api_admins` possui os arquivos responsáveis pelo armazenamento e gerenciamento de usuários administrativos, os quais poderão gerar seus tokens para realizar acessos remotos via endpoints. A pasta `company` é responsável pela lógica, armazenamento e gerenciamento de todos os dados (conta e perfil) das empresas cadastradas no site.

Assim como a `company`, a pasta `users` é responsável pela lógica, armazenamento e gerenciamento de todos os dados (conta e perfil) dos usuários cadastrados no site. A pasta `vacancy`, assim como o nome define, é responsável pelo tratamento de todas as vagas criadas. Por fim, a pasta `resumes` é responsável pela lógica, armazenamento e gerenciamento de todos

os currículos gerados pelos usuários, os quais serão utilizados para se candidatar nas vagas, as quais por sua vez, serão publicadas pelas empresas.

A estrutura de urls e endereços da API são definidas em arquivos específicos dentro do framework do Django, nomeados de `urls.py`. Nesses arquivos, são definidos os endereços os quais serão acessados pelos usuário através dos endpoints, essas definições irão vincular a parte lógica da API (também conhecidas como views) e a área externa da aplicação (os endpoints).

As views, por sua vez, possuem uma abordagem orientada a objetos, também conhecida como class-based views, sendo ela uma das abordagens disponibilizada pelo framework do Django. A criação de uma API consiste em duas classes, um schema, o qual será responsável por produzir as informações necessárias para gerarmos a documentação do Swagger, e uma classe onde será responsável pelo endpoint.

A classe responsável pelo endpoint deverá conter os métodos das requisições que deverão ser suportadas pelo endpoint, ou seja, se o endpoint permite uma requisição no método GET, uma função `get` deve existir dentro da classe pelo fato de que ela será executada ao receber uma requisição desse método.

Essa função também deverá retornar uma resposta que possa ser serializada e enviada via HTTP, permitindo uma comunicação entre APIs sem a renderização de uma página. Essa resposta deverá receber um dicionário que irá conter os dados que deverão ser retornados na resposta, assim como um código de status.

Outro modelo de resposta que também é muito utilizado é o retorno de objetos do banco de dados, porém, diferente de um texto comum, um objeto que seria uma classe na linguagem Python não pode ser serializado automaticamente pelo framework do rest. Para permitir a serialização de um objeto, é necessário definir um serializador que realizará essa conversão de um objeto para algo que possa ser emitido via protocolos HTTP.

Figura 4 - Serializador de objetos

```
You, 27 seconds ago | 1 author (You)
class VacancyModelSerializer(serializers.ModelSerializer):
    addresses = serializers.SerializerMethodField()
    company_name = serializers.SerializerMethodField()

    def get_addresses(self, obj):
        if obj.address:
            return obj.address.__getattr__("serialize")

        return []

    def get_company_name(self, obj: models.VacancyModel):
        return obj.created_by.fantasy_name

You, 27 seconds ago | 1 author (You)
class Meta:
    model = models.VacancyModel
    fields = ["pk", "created_by", "company_name", "role", "description",
             "modality", "created_at", "salary", "addresses", "resumes"]
```

Fonte: Autoria própria

O exemplo mostrado na figura 4 é o serializador de objetos de vagas, ele é responsável por converter uma entrada do banco de dados em um objeto que possa ser serializado e transmitido para o requisitante via protocolo HTTP. Em resumo, o serializador possui uma meta classe que irá definir duas principais variáveis, a variável `model` será responsável por definir de qual tabela os dados deverão ser buscados, já a variável `fields` informará quais informações (colunas) devem ser selecionadas e serializadas.

Além dessas duas variáveis, o exemplo exibe também algumas variáveis computadas, essas variáveis são únicas e geradas durante a serialização, geralmente utilizadas para obter informações de outras tabelas que estão vinculadas por chaves estrangeiras.

No framework Django, o banco de dados é tratado completamente pelo próprio framework, sendo necessária a criação de um modelo. O modelo é uma classe a qual possui uma estrutura específica que pode ser interpretada pelo framework e manipulada como tabelas em um banco de dados relacional.

Figura 5 - Modelo da tabela de vagas

```
You, last month | 1 author (You)
class VacancyModel(models.Model):
    created_by = models.ForeignKey(CompanyAccountModel,
                                   verbose_name="Created by",
                                   on_delete=models.CASCADE)
    role = models.CharField(verbose_name="Role", max_length=255)
    description = models.TextField(verbose_name="Description")
    modality = models.CharField(verbose_name="Modality", max_length=255)
    created_at = models.DateTimeField(verbose_name="Created At",
                                      auto_now_add=True,
                                      editable=False)
    salary = models.PositiveIntegerField(verbose_name="Salary")
    address = models.ForeignKey(VacancyAddress,
                               verbose_name="Address",
                               on_delete=models.SET_NULL,
                               null=True,
                               blank=True)
    resumes = models.ManyToManyField(ResumeModel,
                                     verbose_name="Resumes",
                                     blank=True)

    def __str__(self) -> str:
        return self.role

You, 2 months ago | 1 author (You)
class Meta:
    verbose_name = 'Job vacancy'
    verbose_name_plural = 'Job vacancies'
```

Fonte: Autoria própria

O exemplo mostrado na figura 5, é o modelo da tabela das vagas criadas por uma empresa, o modelo herda de uma classe específica do framework que permitirá que o Django interprete suas variáveis, métodos e metaclasses e gere comandos SQL a partir dessas interpretações. As variáveis declaradas na classe são as variáveis que irão definir as colunas das tabelas, os seus valores são os tipos / atributos nos quais os valores dessa coluna serão tratados. Assim como a herança da classe, os valores das variáveis são classes específicas do framework, os quais permitirão a interpretação e manipulação conforme necessário.

Devido a responsabilidade de manipulação do banco de dados ser completamente do framework, o Django disponibiliza nativamente uma interface de administração para que os dados do banco possam ser facilmente acessados e modificados sem a necessidade uma conexão manual com o banco. O registro de um modelo para exibição na página de

administração do framework é realizado por um arquivo específico de sua estrutura, nele são criadas classes que irão fornecer informações necessárias para a renderização das páginas.

3. METODOLOGIA

A metodologia de pesquisa utilizada foi bibliográfica, a análise de dados sobre imigrantes foi feita em cima de dados oficiais obtidos de documentos do governo brasileiro. Com o objetivo de criar o site, os métodos utilizados foram definir a arquitetura cliente e servidor, onde toda parte visual do site foi desenvolvida utilizando o framework React com a linguagem TypeScript e as regras de negócio junto aos dados foram estabelecidas na parte do servidor, com uma API REST feita com Django e a linguagem Python. O sistema foi desenvolvido separado por funcionalidade, implementando e unificando as alterações a cada funcionalidade finalizada.

Sobre a parte de UI do sistema, foi criado um protótipo utilizando a ferramenta Figma para dar a possibilidade de ajuste de layout e design de todo site com facilidade de visualização, visando oferecer uma boa experiência aos usuários. Os dados foram salvos em um banco de dados único que utiliza a convenção relacional. O sistema foi dividido em tabelas e o sistema de gerenciamento escolhido foi o MySQL. Todo código está hospedado em um repositório privado no GitHub, onde todos os desenvolvedores envolvidos têm acesso.

O material de estudo foi obtido a partir de notícias, artigos, fóruns e livros que foram publicados nos últimos 5 anos. O conteúdo coletado foi utilizado para identificar as dificuldades e necessidades dos imigrantes nesse processo de transição, tendo como objetivo criar e fornecer ferramentas para ajudá-los a superar essas dificuldades. O controle de qualidade do site foi feito de forma manual durante o desenvolvimento e aplicação das ferramentas e funcionalidades, com o intuito de mitigar erros e melhorar a experiência do usuário durante a sua utilização.

4. REFERENCIAL TEÓRICO

No Brasil, o número de estrangeiros buscando oportunidades de emprego tem aumentado cada vez mais, isso se deve principalmente devido a crises em outros lugares do mundo. Segundo o relatório do Observatório das Migrações Nacionais (2022), o volume de estrangeiros trabalhando no Brasil foi de 62.423 para 187.985 de 2011 para 2021, aumentando em mais de três vezes no período de uma década.

Além dos trabalhadores com visto legal, o Brasil tem muitos refugiados e o marco legal para o início dessa imigração foi em 22 de julho de 1997 com a Lei nº 9.474 que define que será reconhecido como refugiado todo indivíduo que:

- I – devido a fundados temores de perseguição por motivos de raça, religião, nacionalidade, grupo social ou opiniões políticas encontre-se fora de seu país de nacionalidade e não possa ou não queira acolher-se à proteção de tal país;
- II – não tendo nacionalidade e estando fora do país onde antes teve sua residência habitual, não possa ou não queira regressar a ele, em função das circunstâncias descritas no inciso anterior;
- III – devido a grave e generalizada violação de direitos humanos, é obrigado a deixar seu país de nacionalidade para buscar refúgio em outro país. (LEI Nº 9.474, DE 22 DE JULHO DE 1997).

Segundo os dados do Observatório de Migrações Nacionais (2022) entre os anos de 1997-2009 o Brasil recebeu 2.488 solicitações de refúgio e entre os anos 2010-2021 esse número aumentou mais de 100 vezes, saltando para 298.331 solicitações, dos quais 59% foram venezuelanos e 13,3% haitianos. Este fato se deve principalmente devido as crises humanitárias tanto na Venezuela quanto no Haiti, que fez com que parte da população buscasse refúgio em outros lugares do mundo.

Tabela 1 – Número de solicitações de reconhecimento da condição de refugiado, segundo principais países de nacionalidade ou de residência habitual, Brasil, 2016-2021

Principais países	Número de Solicitações
Total	258.241
VENEZUELA	174.945
HAITI	33.533
CUBA	11.425
ANGOLA	6.048
CHINA	5.369
BANGLADESH	2.852
SENEGAL	2.504
SÍRIA	1.963
NIGÉRIA	1.725
COLÔMBIA	1.171
GUINÉ-BISSAU	1.047
ÍNDIA	1.045
GANÁ	948
PAQUISTÃO	852
REPÚBLICA DEMOCRÁTICA DO CONGO	843
LÍBANO	827
MARROCOS	789
FILIPINAS	551
GUINÉ	545
CAMARÕES	505
OUTROS PAÍSES	8.754

Fonte: Relatório Anual 2022 (OBMigra)

Além de todos os dados já levantados, existem muitos refugiados e imigrantes que estão ilegalmente no país e acabam não entrando nos dados oficiais. É evidente que o Brasil tem um volume considerável de imigrantes tentando iniciar uma vida aqui e todos os dias novas vagas de emprego aparecem tentando ser preenchidas. Tendo em vista o crescimento do uso da tecnologia na vida das pessoas, ter uma plataforma online disponível para auxiliar na busca de vagas e normas é muito positivo.

As tecnologias sem fio, como os celulares e as formas de conexão Wi-Fi à internet, têm criado novas práticas de mobilização social nas metrópoles contemporâneas. A era da conexão relaciona assim tecnologia digital, comunicação, massa, multidão, mobilidade e conexão (LEMOS, 2005).

Com todos esses aspectos, é necessário criar um facilitador para os imigrantes que já vivem no Brasil e os que tem interesse em imigrar, que contenha desde informações sobre documentação até busca de vagas de emprego de confiança, para que seja evitada a exposição

a golpes e vagas falsas, visto que de acordo com Léa Lima (2019) cerca de 42% das empresas utilizaram a internet para recrutar em 2017.

Para isso o nosso site tem como objetivo auxiliar residentes e estrangeiros com interesse de trabalhar no Brasil, também os que já estão no país e precisam de um emprego e empresas que buscam expandir suas buscas de profissionais com aptidões para suas vagas abertas.

Existem diversos meios de divulgação de empregos no Brasil, com vagas disponíveis para os interessados se inscreverem e tentarem preencher. Os sites e aplicativos estão cada vez mais sendo utilizados como ponte para essa conexão entre empresas e pessoas buscando vagas. O recrutamento por meio dessas plataformas também possibilita a facilidade de envio de currículos, facilitando que empresas encontrem potenciais candidatos para preencher suas vagas de emprego, pois segundo Chiavenato (2010) “Este é o papel do recrutamento: divulgar no mercado as oportunidades que a organização pretende oferecer para as pessoas que possuam determinadas características desejadas”.

Foi utilizado como base de estudo sites que de acordo com o portal Guia da Carreira, estão entre os 11 melhores sites de busca de emprego do Brasil. Dentro deles, existem sites com vagas mais gerais, para qualquer tipo de pessoa e existem outros mais focais, com vagas específicas para estagiários, pessoas com deficiência, áreas específicas ou para empregos fora do Brasil.

Figura 6 – Site LinkedIn



Fonte: LinkedIn, 2023

O mais conhecido é o LinkedIn, lançado na Califórnia em 2003 ele funciona como uma rede social, currículo online e plataforma de empregos, muito utilizado para criar conexões com outras pessoas no mercado de trabalho, mesmo não tendo tantas ofertas de vagas quanto em outros concorrentes. É gratuito e tem a possibilidade de assinatura para amplificar a visibilidade do seu currículo.

Figura 7 – Site Catho



Fonte: Catho, 2023

Existe também a Catho, uma das mais tradicionais e com mais ofertas de emprego no Brasil, o serviço tem a opção de teste gratuito por 7 dias, mas é pago após esse tempo. Pessoas com deficiência conseguem usar gratuitamente.

Figura 8 – Site Indeed



Fonte: Indeed, 2023

Já o Indeed é um agregador de vagas totalmente gratuito, nele é possível buscar vagas em diversos sites de empregos diferentes ao mesmo tempo. Dessa forma poupa tempo do usuário em ficar entrando em diversos sites diferentes para realizar novas buscas.

Figura 9 – Trovit



Fonte: Trovit, 2023

O Trovit é um site internacional muito utilizado na Europa e na América Latina, mas diferente dos outros, ele não só tem vagas de emprego como também funciona como um classificado, tendo imóveis e carros também.

Figura 10 – Site CIEE



Fonte: CIEE, 2023

Para os estagiários, existe o Centro de Integração Empresa-Escola (CIEE), o qual é completamente gratuito e tem vínculo com diversas empresas que divulgam seus programas de estágio pela sua plataforma, facilitando a busca de quem está na faculdade querendo iniciar a vida profissional com estágio na sua área.

Figura 11 – Deficiente Online



Fonte: Deficiente Online, 2023

E para as pessoas com deficiência, foi criado o Deficiente Online, conhecido como PCD.com.br, dá a possibilidade de o usuário filtrar pelo tipo de deficiência que possui, além de possuir também guias sobre documentação para facilitar a organização dos candidatos. De todas as opções citadas anteriormente, essa é a mais próxima do que é necessário para os imigrantes no Brasil.

5. RESULTADOS

O Brazil Career Hub foi um projeto que facilitou muito a aplicação para vagas de todos que pretendem trabalhar no Brasil, sua interface trouxe uma experiência fácil e intuitiva, sem fazer com que o usuário passe muito tempo procurando por um emprego, a interação visual do site com o servidor ficou muito boa, com vários pontos dinâmicos, sem problemas de performance etc. O projeto se tornou uma boa oportunidade de investimento para trazer mão de obra qualificada de qualquer lugar do mundo para o Brasil, assim como promover o trabalho para brasileiros residentes, no qual resulta em geração de empregos.

O projeto também resolve o problema de falta de emprego em cidades pequenas ao disponibilizar a possibilidade de criação de vagas remotas. Um outro problema que o site facilita é a comunicação, o cadastro do usuário permite registrar informações para que a empresa possa vir a se comunicar, como e-mail, telefone, website etc. A empresa também pode fornecer alguns dados na qual o usuário pode pesquisar mais sobre ela.

6. CONCLUSÃO

Após o término do projeto concluímos que os objetivos iniciais foram alcançados com o desenvolvimento do projeto Brazil Career Hub, uma vez que suas funções iniciais foram desenvolvidas com êxito e os testes iniciais foram um sucesso, assim imigrantes e refugiados podem se utilizar do projeto desenvolvido para procurar vagas de emprego e entrar no mercado de trabalho brasileiro. Ademais tanto imigrantes quanto os próprios brasileiros podem se informar de leis e processos burocráticos do Brasil. Porém visto que não serão possíveis testes em grande escala ou a publicação do projeto ao público por conta de seu intuito acadêmico não pudemos ver os resultados iniciais refletidos em grande escala.

7. REFERÊNCIAS

ALEXANDRIA, K. **A um dia do fim, 80 mil goianos ainda não declararam o Imposto de Renda.** O Popular. Disponível em: <https://opopular.com.br/economia/a-um-dia-do-fim-80-mil-goianos-ainda-n-o-declararam-o-imposto-de-renda-1.2464589>. Acesso em 25 de março de 2023.

Brasil. **Lei no 9.474, de 22 de julho de 1997. Define mecanismos para a implementação do Estatuto dos Refugiados de 1951, e determina outras providências.** Brasília, DF:

Presidência da República. Disponível em: https://www.planalto.gov.br/ccivil_03/leis/19474.htm. Acesso em 7 de maio de 2023.

CAVALCANTI, Leonardo; OLIVEIRA, Antonio Tadeu; TONHATI, Tânia (Orgs.) **A Inserção dos Imigrantes no Mercado de Trabalho Brasileiro**. Cadernos OBMigra, Ed. Especial, Brasília, 2015.

CAVALCANTI, L; OLIVEIRA, T.; SILVA, B. G. **Relatório Anual OBMigra 2022**. Série Migrações. Observatório das Migrações Internacionais; Ministério da Justiça e Segurança Pública/ Conselho Nacional de Imigração e Coordenação Geral de Imigração Laboral. Brasília, DF: OBMigra, 2022.

CHIAVENATO, I. **Gestão de Pessoas: O novo papel dos recursos humanos nas organizações**. 3 ed. Rio de Janeiro: Elsevier, 2010.

FREITAS, E. **O número de imigrantes nos Estados Unidos**. Brasil Escola. Disponível em: <https://brasilescola.uol.com.br/geografia/o-numero-imigrantes-nos-estados-unidos.htm>. Acesso em 25 de março de 2023.

GUIA DA CARREIRA. **Descubra quais são os melhores sites de emprego do Brasil**. Disponível em: <https://www.guiadacarreira.com.br/blog/os-melhores-sites-de-emprego>. Acesso em 7 de maio de 2023.

IBGE. **Desemprego**. Disponível em: <https://www.ibge.gov.br/explica/desemprego.php>. Acesso em 25 de março de 2023.

LEMO, A.. **Cibercultura e Mobilidade: A Era da Conexão**. In: CONGRESSO BRASILEIRO DE CIÊNCIAS DA COMUNICAÇÃO, 28. Anais. Rio de Janeiro: Universidade do Estado do Rio de Janeiro, 2005.

LIMA, Léa. **Como as plataformas de emprego constroem o mercado de trabalho: uma comparação de classificados de vagas no setor da limpeza em seis portais de emprego brasileiro**. 19º Congresso Brasileiro de Sociologia, Jul 2019, Florianópolis, Brazil.

SILVA, L. **Os imigrantes no Brasil, sua vulnerabilidade e o princípio da igualdade**. Disponível em: https://www.researchgate.net/profile/Leda-Maria-Silva/publication/320203829_OS_IMIGRANTES_NO_BRASIL_SUA_VULNERABILIDADE_E_O_PRINCIPIO_DA_IGUALDADE/links/5dc589ff299bffa47b23d708/OS-IMIGRANTES-NO-BRASIL-SUA-VULNERABILIDADE-E-O-PRINCIPIO-DA-IGUALDADE.pdf. Acesso em 7 de maio de 2023.

TST. **Brasil tem mais de 180 mil imigrantes no mercado de trabalho formal**. Disponível em: [https://www.tst.jus.br/-/brasil-tem-mais-de-180-mil-imigrantes-no-mercado-de-trabalho-formal#:~:text=Em%20uma%20década%2C%20o%20volume,das%20Migrações%20Nacionais%20\(OBMigra\)](https://www.tst.jus.br/-/brasil-tem-mais-de-180-mil-imigrantes-no-mercado-de-trabalho-formal#:~:text=Em%20uma%20década%2C%20o%20volume,das%20Migrações%20Nacionais%20(OBMigra)). Acesso em 25 de março de 2023.