

Desenvolvimento de uma Plataforma para Agregação de Vagas de Emprego no Brasil com Web Scraping e Classificação por Google Natural Language AI

Development of a Platform for Aggregating Job Vacancies in Brazil with Web Scraping and Classification by Google Natural Language AI

Romário Araújo Carlos

rac3@discente.ifpe.edu.br

Rodrigo Cesar Lira da Silva

rodrigo.lira@paulista.ifpe.edu.br

RESUMO

Este trabalho descreve o desenvolvimento de uma plataforma web para a busca por oportunidades de emprego, utilizando de técnicas de web scraping e a utilização da ferramenta de classificação de linguagem natural Google Natural Language AI. A aplicação visa facilitar o processo de pesquisa por uma vaga no mercado de trabalho, classificando as ofertas disponíveis por meio de palavras-chave que representam a área de interesse do candidato. Para viabilizar essa classificação, é aplicada a técnica de web scraping, que permite a extração automatizada de dados de sites que listam vagas de emprego. Os dados capturados são então processados e classificados pelo Google Natural Language AI, etapa na qual ocorre a identificação das palavras-chave relevantes que são vinculadas às respectivas vagas. Já na aplicação web, as vagas são listadas com suas palavras-chave e outros dados relevantes como faixa salarial, modalidade de contratação e local de trabalho, permitindo ao usuário a filtragem por sua área de interesse de forma simplificada. Consequentemente, o processo de busca por oportunidades profissionais torna-se mais eficiente, uma vez que os candidatos passam a receber informações mais alinhadas ao seu perfil e interesses, reduzindo o tempo gasto em buscas manuais.

Palavras-chave: Aplicação web, Web scraping, Linguagem natural, Emprego

ABSTRACT

This paper aims to describe the development of a web platform for searching job opportunities, using web scraping techniques and the Google Natural Language AI natural language classification tool. The application aims to facilitate the job search process by classifying available offers using keywords that represent the candidate's area of interest. To enable this classification, the web scraping technique is applied, which allows the automated extraction of data from websites that list job vacancies. The captured

data is then processed and classified by Google Natural Language AI, a stage in which the identification of relevant keywords that are linked to the respective vacancies occurs. In the web application, the vacancies are listed with their keywords and other relevant data such as salary range, hiring modality, and workplace, allowing the user to filter by their area of interest in a simplified way. Consequently, the process of searching for professional opportunities becomes more efficient, since candidates start to receive information that is more aligned with their profile and interests, reducing the time spent on manual searches.

Keywords: Web application, Web scraping, Natural language, Employment

1 INTRODUÇÃO

A internet veio revolucionar o processo de recrutamento, devido ao seu imediatismo e na facilidade de as pessoas interagirem, trouxe mais agilidade às empresas e às pessoas (Duarte Neto; Bandeira; Macêdo, 2020). Além desta característica, o recrutamento *on-line* deixou o processo menos custoso (Reche, 2011). Nesse formato de recrutamento, a única ou principal forma de contato com os candidatos é realizada pela internet, muitas vezes eliminando os anúncios em jornais, agências terceirizadas e outras formas mais ultrapassadas (Guimarães, 2024).

No cenário digital, a internet tem se tornado uma ferramenta aliada na busca por vagas de emprego, ela potencializa o alcance para os candidatos e traz vantagens significativas no processo de recrutamento e seleção. As empresas também passam a se beneficiar, com a diminuição dos custos para contratar, como gastos com divulgação de vagas, tempo despendido em triagens de currículos e processos seletivos prolongados, além da diversificação de competências e habilidades da sua força de trabalho.

Diante desta situação, portais *on-line* de vagas de emprego e divulgação de currículos passaram a desempenhar um papel fundamental no processo de recrutamento e seleção, facilitando a conexão entre empresas e candidatos. Estes portais oferecem soluções práticas para consulta de vagas e permitem que as empresas filtrem talentos de maneira mais rápida e assertiva. Além disso, os candidatos contam com um amplo leque de oportunidades e, muitas das vezes, sem limitações geográficas.

Entretanto, a grande quantidade de portais disponíveis na internet leva a um cenário em que os candidatos acabam dedicando um tempo excessivo à adaptação aos diferentes métodos de filtragem das oportunidades mais relevantes e, conseqüentemente, perdem um dos principais benefícios da busca por vagas *on-line*: a agilidade.

Dessa forma, este projeto descreve a criação de uma aplicação *web* para o desenvolvimento de um portal de vagas de emprego, que permite a filtragem conforme a área de interesse do candidato, agregando anúncios de diferentes portais com o objetivo de simplificar a busca por oportunidades. Utilizando técnicas de *web scraping*, a aplicação coleta anúncios de vaga em diferentes portais e as classifica por meio da

*Natural Language AI*¹ da Google, economizando tempo na filtragem e maximizando a eficácia na busca por emprego.

O trabalho está estruturado da seguinte maneira: na Seção 2 são descritos trabalhos relacionados, que de alguma forma se assemelham ao projeto. Na Seção 3 são detalhadas as etapas da metodologia utilizada. Na Seção 4 são exibidas as funcionalidades do sistema e os resultados obtidos. Na Seção 5 são apresentadas as considerações finais.

2 TRABALHOS RELACIONADOS

Nesta seção são analisados trabalhos de acordo com os tópicos abordados, como *Web scraping* e Plataformas de agregação de vagas. A seguir, na Tabela 1, é apresentado um resumo com as contribuições de cada um desses estudos, facilitando a comparação entre as abordagens dos trabalhos obtidos.

Tabela 1 – Comparação de Trabalhos Relacionados

Trabalho	Scraping	Plataforma Web	Classificação	Notificações
Cocon et al. (2023)	✓	☐	☐	☐
Kumar et al. (2023)	✓	✓	☐	☐
Sinha et al. (2021)	✓	✓	☐	☐
Kumar et al. (2024)	✓	✓	☐	✓
Este trabalho (2025)	✓	✓	✓	(Sugerido para trabalhos futuros)

Em (Cocón et al., 2023), demonstram a utilização de técnicas de extração de dados em portais públicos que fornecem informações sobre profissões, com comparações entre algumas ferramentas que permitem *web scraping*. As ferramentas são usadas para compilar informações acerca dos empregos das distintas carreiras oferecidas no México. Por fim, o trabalho conclui elegendo a ferramenta WebHarvy² como a mais adequada, destacando as funcionalidades de *Append* e a possibilidade de omitir colunas desnecessárias.

Em (Kumar; Kinshuk; Jaspreet, 2023), é proposto o desenvolvimento de uma plataforma de notificação de vagas de emprego alinhadas ao interesse do candidato. As informações das propostas de trabalho são obtidas através de técnicas de *web scraping*, a linguagem de programação escolhida para o desenvolvimento é Python, com o auxílio das bibliotecas de manipulação de dados *Octa-parse* e *Parse-hub*. O autor destaca a agregação e centralização dos dados como pontos positivos para o usuário.

¹<https://cloud.google.com/natural-language/docs/reference/rest>

²<https://www.webharvy.com/>

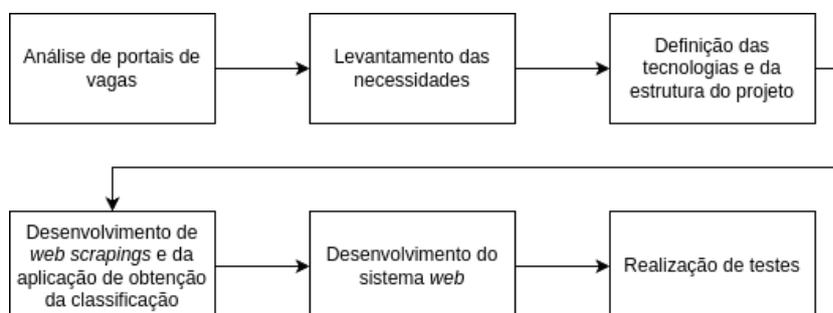
Como visto em (Sinha; Saxena; Garg, 2021), a grande quantidade de informações presente na internet é favorável, mas também pode ser hostil. O artigo também descreve que 27,9% da internet utilizada por indivíduos é para fins de emprego e completa: com o crescimento da Internet, a quantidade de sites de busca de emprego se desenvolveu rapidamente, e isso incentiva as pessoas a procurarem por vagas de forma adequada através da internet. Os usuários dedicam grandes quantidades de energia procurando por empregos viáveis. A pesquisa propõe a utilização de técnicas de *web scraping* para a realização da extração de dados dos portais *Naukri*³ e *Indeed*⁴ e, depois, a exibição em um portal web de forma simplificada. O projeto conta com a utilização da linguagem de programação Python e das bibliotecas *BeautifulSoup* e *Selenium*.

Em (Kumar; Chauhan; Grewal, 2024), afirmam que pode ser desafiador para os engenheiros recém-formados encontrar uma oportunidade de trabalho que se adeque à sua formação acadêmica e nível de habilidade. Acrescentam que os portais de vagas de emprego *online* são a forma mais rápida de se comunicar, permitindo alcançar um público amplo com apenas um clique. Por isso, propõem o desenvolvimento de uma plataforma de listagem de vagas de emprego, que agrega anúncios de diversos portais, eliminando a necessidade de múltiplos cadastros e personalizando recomendações com base nas habilidades e preferências do usuário. O sistema de *web scraping* utiliza a linguagem Python, combinada com as bibliotecas *BeautifulSoup*, *Selenium* e *Scrapy*, além da construção do portal com Django e o envio de e-mails com notificações das vagas via protocolo SMTP (*Simple Mail Transfer Protocol*). O artigo conclui que, ao realizar o agrupamento de vagas de emprego, o sistema facilita a busca para os candidatos, tornando o processo mais simples.

3 METODOLOGIA

Nesta seção são apresentadas as etapas e procedimentos da metodologia de desenvolvimento que foi utilizada para a construção deste trabalho. A seguir, serão listadas as atividades que correspondem a etapa de desenvolvimento de toda a estrutura criada, como é possível observar na Figura 1.

Figura 1 – Diagrama de fluxo de desenvolvimento da estrutura do sistema.



Fonte: Própria.

³<https://www.naukri.com/> - Portal indiano de busca por vagas de emprego

⁴<https://www.indeed.com/> - Portal de busca por vagas de emprego

1. **Análise de portais de vagas:** Nessa etapa, foi realizado o estudo de portais de vagas de emprego aptos à *web scraping*.
2. **Levantamento de atividades necessárias:** Criação das tarefas de desenvolvimento no Kanban.
3. **Definição das tecnologias e da estrutura do projeto:** Nesse estágio, foi definida a linguagem JavaScript e o banco de dados MongoDB para a construção do projeto.
4. **Desenvolvimento de *web scrapings* e da aplicação de obtenção da classificação:** Nessa etapa, foram desenvolvidas as aplicações responsáveis pela coleta de dados das páginas de vagas de emprego e da aplicação responsável pela classificação por meio da Natural Language AI.
5. **Desenvolvimento do sistema web:** Nessa etapa, foi desenvolvido o portal *web*, responsável pela listagem das vagas já classificadas com suas respectivas áreas de interesse e do painel administrativo para gerenciamento de automações e controle da exibição das vagas. Além disso, as aplicações criadas foram adicionadas ao Docker, garantindo a escalabilidade e portabilidade.
6. **Realização de testes:** Testes funcionais em toda a estrutura criada na etapa de desenvolvimento foram realizados nesta etapa.

Para realizar a gestão das atividades do projeto, foi escolhida a estratégia *Kanban*⁵, uma abordagem ágil amplamente utilizada para gestão visual do andamento de fluxos de trabalho. O Kanban permite o acompanhamento das tarefas de forma transparente, prezando pela melhoria contínua do fluxo e na gestão do tempo.

A plataforma escolhida para a implementação do Kanban foi o *Notion*⁶, que oferece um ambiente de produtividade flexível para criação de painéis, tabelas e listas. No *Notion*, foi desenvolvido um quadro para a implementação deste projeto onde as atividades levantadas foram organizadas nas colunas que representam os estágios de trabalho: “Não iniciada”, “Em progresso” e “Finalizada”.

A aplicação foi construída no formato *web*, possibilitando ao usuário o acesso multipataforma, com isto, não importando o sistema operacional que o usuário utilize, bastando apenas possuir um navegador *web* para a utilização. Para o portal implementado, adotou-se a designação VAGEX, sendo disponibilizado no *repositório Vagex*⁷.

Para o desenvolvimento do *software*, foram utilizadas tecnologias gratuitas ou que oferecem cotas de uso gratuito, como HTML, Node.js, o banco de dados MongoDB, a IDE Visual Studio Code, a plataforma de criação de contêineres Docker e, para a classificação das oportunidades de emprego, a API⁸ *Google Natural Language AI*.

O Google Natural Language AI fornece tecnologias de compreensão de linguagem natural, como análise de sentimento, reconhecimento de entidade, análise de sentimento de entidade e outras anotações de texto, para desenvolvedores (Google, 2025).

⁵<https://asana.com/pt/resources/what-is-kanban>

⁶<https://www.notion.com/>

⁷<https://github.com/romaaraujo/vagex>

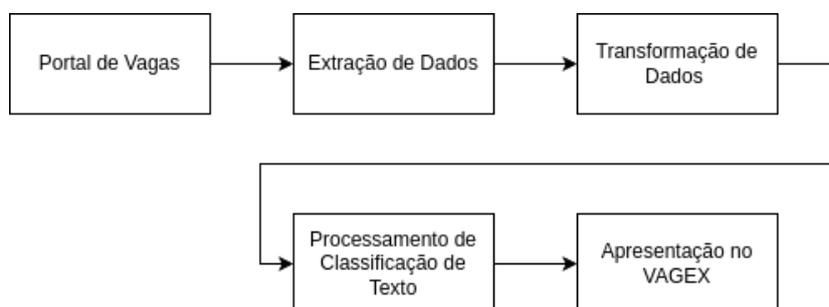
⁸<https://www.redhat.com/en/topics/api/what-are-application-programming-interfaces>

No contexto deste projeto, foi implementada a *API Google Natural Language AI* com o objetivo de classificar vagas de emprego. Para isso, foi utilizado o método *Classify-Text*⁹, no qual um objeto contendo a descrição da vaga é enviado, e a API retorna uma classificação denominada categoria. Essa classificação apresenta dois campos principais: “*name*”, que representa o nome da categoria, e “*confidence*”, cujo valor varia de 0.0 (indicando um nível de confiança mais baixo) a 1.0 (indicando que a classificação é extremamente precisa).

As plataformas Portal ABRE, Solides.com.br e Vagas.com.br foram utilizadas para extração de dados dos anúncios das vagas de emprego. Essas plataformas foram escolhidas por sua relevância no universo dos portais de vagas *on-line* e pela viabilidade do *web scraping*, uma vez que não possuem barreiras que impeçam a extração de dados. Além disso, o Portal ABRE se destaca por ser voltado a oportunidades de estágio, o que representou um diferencial para a implementação desse processo. A construção do sistema de *web scraping* seguiu uma abordagem modular, possibilitando a integração de novos algoritmos para a extração de dados de outros portais de vagas conforme necessário, ampliando continuamente a base de anúncios.

A Figura 2 ilustra o processamento sequencial do fluxo de dados do sistema, representando todas as etapas, desde a coleta até a apresentação. As etapas são explicadas da seguinte forma:

Figura 2 – Diagrama de fluxo de dados do sistema.



Fonte: Própria.

1. **Portal de Vagas:** Representa o ponto de partida na obtenção dos dados que serão utilizados ao longo do processo. Nesta etapa, as informações sobre vagas de emprego a serem classificadas são buscadas.
2. **Extração de Dados:** Os dados mais relevantes dos portais de vagas são coletados por meio de *web scraping*.
3. **Transformação de Dados:** Após a extração, os dados passam por um processo de normalização, limpeza e estruturação antes de serem armazenados no banco de dados.
4. **Processamento de Classificação de Texto:** Os dados normalizados são submetidos ao processo de classificação por meio da Natural Language AI, resultando na obtenção das áreas de interesse para cada vaga de emprego.

⁹<https://cloud.google.com/natural-language/docs/reference/rest/v2/documents/classifyText>

5. **Apresentação no VAGEX:** Na etapa final, os dados obtidos e classificados são exibidos na página de vagas do Portal VAGEX, representando a interface de saída.

3.1 Web scraping

A extração de dados dos portais de vagas *on-line* é essencial para o desenvolvimento da aplicação, pois assegura que, na etapa seguinte, teremos as informações necessárias para realizar o processamento e a classificação. Considerando a diversidade de portais, cada um com suas particularidades na lógica de obtenção de dados, é fundamental desenvolver algoritmos de extração específicos para cada plataforma, levando em conta as regras de acesso e as estruturas próprias de cada site.

Após a extração dos dados, inicia-se o processo de normalização, que, assim como a extração, exige a implementação de algoritmos específicos para cada portal. Por exemplo, enquanto alguns sites não fornecem informações sobre o salário nas vagas, outros oferecem esse dado. Nesse caso, é necessário tratar a informação, convertendo-a para o formato adequado, como um número, garantindo que o dado seja salvo corretamente em seu tipo apropriado, como numeral, para assegurar a consistência e integridade das informações armazenadas.

Os dados normalizados são então salvos no banco de dados MongoDB, seguindo a estrutura que é possível observar na Figura 3.

Figura 3 – Estrutura do modelo de vagas

```
{
  id : string
  url : string
  titulo : string
  descricao : string
  local : string
  empresa : string
  hierarquia : string
  ▾ salario : {
    min : numero
    max : numero
  }
  regime : string
}
```

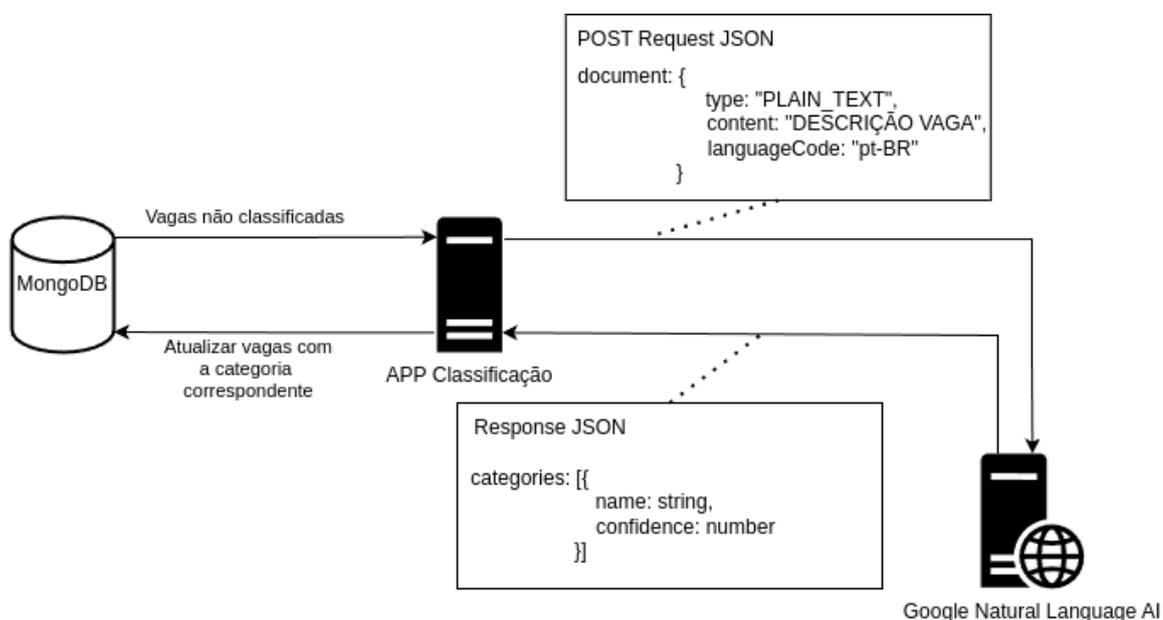
Fonte: Própria.

3.2 Classificação

Na etapa de classificação, o processo começa com uma busca no banco de dados para identificar as vagas que ainda não foram classificadas. Em seguida, é construído o objeto de requisição a ser enviado à API de classificação, com base na estrutura definida: TÍTULO - EMPRESA - HIERARQUIA - REGIME - DESCRIÇÃO, informações essas extraídas na etapa anterior. Esse formato garante que os dados sejam enviados de maneira organizada e padronizada, otimizando a classificação das vagas.

Considerando a limitação da *API Natural Language AI*, que não permite a classificação em lote, as vagas são processadas individualmente. Cada vaga é enviada, uma por uma, via requisição *HTTP POST*. Assim que a classificação é recebida, o banco de dados é atualizado imediatamente, incorporando as *tags* de classificação retornadas pela API. O processo descrito está representado na Figura 4.

Figura 4 – Diagrama do fluxo de requisição de classificação por meio da *Natural Language AI*.



4 RESULTADOS

Nesta seção são apresentados os resultados obtidos com a construção do sistema. A premissa central foi desenvolver uma aplicação capaz de auxiliar o candidato na busca por uma vaga no mercado de trabalho, garantindo celeridade e eficiência na escolha.

Os resultados foram avaliados com base nos critérios de precisão na classificação, fluidez do sistema e desempenho do *web scraping* durante a extração dos dados nos portais de vagas. Para a apresentação dos resultados deste trabalho, as informações serão organizadas em subseções que compõem a descrição da solução.

4.1 Descrição da Solução: VAGEX

O sistema *web* desenvolvido está dividido em duas partes principais. A primeira é a parte não autenticada, destinada aos candidatos, onde é possível filtrar vagas de emprego por áreas de interesse de forma rápida, facilitando o acesso às oportunidades. A segunda parte, acessível apenas na área autenticada, é voltada para os administradores do sistema. Nessa seção, o administrador pode acionar funcionalidades importantes, como a busca automatizada de vagas através de algoritmos de *web scraping* nos portais de vagas *on-line*, além de realizar a classificação das vagas obtidas.

4.1.1 Página Vagas

As vagas classificadas são apresentadas na página “Vagas”, onde é possível visualizar rapidamente as principais características das oportunidades de emprego. Conforme ilustrado na Figura 5, no lado esquerdo do site, as vagas são exibidas em um formato de cartão que contém informações essenciais, como o nome da empresa contratante, o valor salarial, a localização e a hierarquia do cargo. Além disso, cada cartão apresenta o título e a descrição resumida da vaga. Ao clicar no botão “Leia Mais”, a descrição completa da vaga é exibida, permitindo ao usuário acessar mais detalhes de forma prática e intuitiva. Para acessar a vaga pretendida, o usuário deve clicar no cartão, onde será direcionado ao site originário da vaga.

Figura 5 – Página de Vagas: exibe a listagem de oportunidades disponíveis, com informações relevantes sobre cada vaga, além de filtros por senioridade e categoria.

A captura de tela mostra a interface do sistema VAGEX. No topo, há o nome do sistema 'VAGEX' e links para 'Vagas', 'Sobre' e 'Admin'. A página exibe uma listagem de vagas em cartões. Cada cartão contém:

- Nome da empresa (ex: CELTA SERVICOS TEMPORARIOS LTDA, EVIAS TECNOLOGIA EM INFRAESTRUTURA VIARIA LTDA)
- Salário (ex: R\$0, R\$1600, R\$2100)
- Localização (ex: Manaus - AM, São Simão - GO, Marília - SP)
- Nível de senioridade (ex: Sênior)
- Botão 'Reportar'
- Título da vaga (ex: Analista de DP, Gerente Comercial, Auxiliar de Topografia Eco Cerrado, Auxiliar de Laboratório Marília)
- Descrição resumida da vaga
- Botão 'Leia Mais'
- Tags de categoria (ex: Recursos Humanos, Direito Trabalhista e Emprego, Gestão, Engenharia Civil, Construção e Manutenção/Outros)

À direita, há um painel de filtros:

- Botão 'Limpar Filtros'
- Seção 'Senioridade' com uma lista suspensa atualizada para 'Todos'.
- Seção 'Categorias' com uma lista de checkboxes para seleção de áreas de interesse.

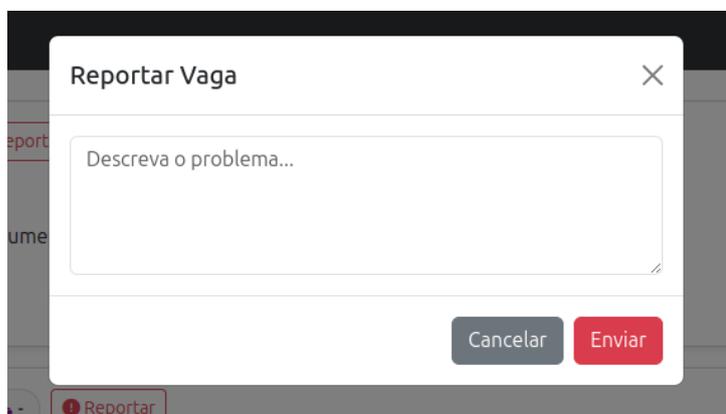
Fonte: Própria.

Do lado direito da página, encontra-se o filtro de senioridade, no qual é possível selecionar entre Sênior, Pleno, Júnior, Especialista e Estagiário, além de manter a seleção em “Todos”, para englobar mais vagas. Além disso, há outro filtro com as

categorias classificadas. Ao selecionar uma categoria, as vagas exibidas são automaticamente atualizadas para corresponder à categoria escolhida. Caso mais de uma categoria seja selecionada, são exibidas todas as vagas que pertencem a pelo menos uma das categorias selecionadas. No canto superior direito, há um botão denominado “Limpar Filtros”, que, ao ser acionado, desmarca todas as categorias selecionadas, permitindo ao usuário redefinir a pesquisa de maneira prática.

Por fim, o usuário pode reportar a vaga à administração do portal, utilizando o botão contornado em vermelho, chamado “Reportar”. Ao clicar, será aberta uma janela onde é possível descrever brevemente a observação a ser enviada para análise. O intuito do botão é permitir ao usuário denunciar eventuais problemas com a vaga, como não estar mais disponível, página de candidatura inacessível ou algum outro tipo de problema. A janela pode ser observada na Figura 6.

Figura 6 – Reportar Vaga: permite ao usuário descrever problemas relacionados a uma vaga específica, por meio de uma caixa de texto.



Fonte: Própria.

4.1.2 Página Sobre

A Figura 7 ilustra a página “Sobre”, que apresenta um resumo claro e objetivo sobre o portal, destacando sua finalidade, principais funcionalidades e os objetivos que busca alcançar.

4.1.3 Página de Autenticação Administrativa

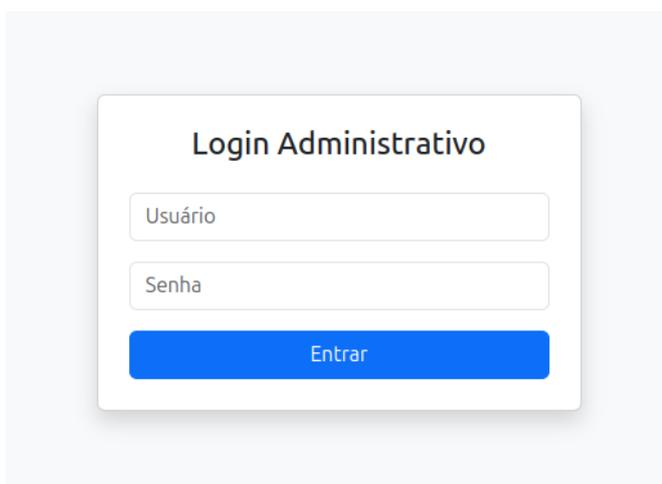
O sistema possui uma página dedicada à autenticação de administradores, acessada ao clicar no item “Admin” no menu. Caso o usuário não esteja autenticado previamente, será redirecionado para a página de *login*, conforme ilustrado na Figura 8. Nessa página, o administrador deve inserir suas credenciais de usuário e senha no formulário de autenticação. Após a autenticação bem-sucedida, o sistema direciona o usuário para o painel autenticado, onde será possível realizar todas as ações e operações exclusivas da administração do sistema.

Figura 7 – Página Sobre: apresenta informações explicativas acerca do funcionamento e objetivos do sistema.



Fonte: Própria.

Figura 8 – Formulário de Autenticação do administrador do sistema.

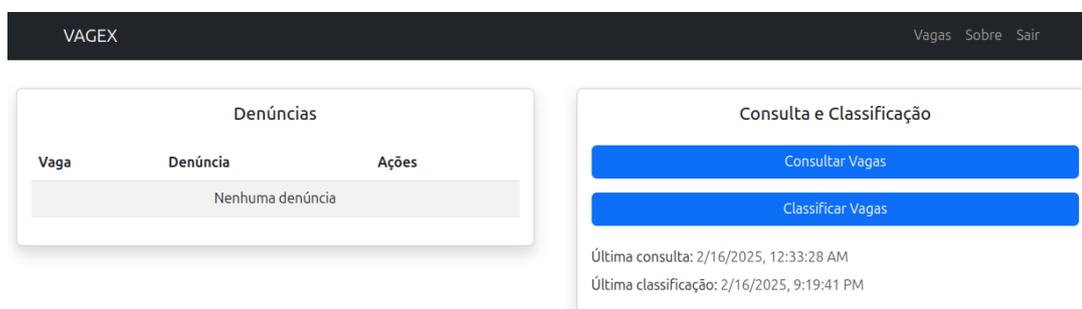


Fonte: Própria.

4.1.4 Página Administrativa

Na página administrativa, acessível apenas após a autenticação do usuário, estão disponíveis dois botões de ação: “Consultar Vagas” e “Classificar Vagas”. Ao clicar em qualquer um deles, o sistema inicia automaticamente o respectivo processo, criando um registro no banco de dados para acionar as etapas de *web scraping* ou classificação de vagas. Durante a execução do processo, o botão correspondente é desativado, garantindo que nenhuma ação duplicada seja realizada. Além disso, abaixo dos botões, são exibidas as datas e horas da última consulta ou classificação realizada, permitindo um controle interno eficiente. A Figura 9 apresenta essa página em detalhes.

Figura 9 – Página Administrativa: oferece ao administrador ferramentas para gerenciar denúncias de usuários e acessar ações relacionadas à consulta e classificação de vagas.



Fonte: Própria.

Como é possível visualizar na Figura 10, no lado esquerdo da página, em “Denúncias”, o administrador pode gerenciar as mensagens reportadas pelos usuários na página de vagas da plataforma. O administrador visualiza o título da vaga denunciada, o texto da denúncia e, nos botões de ação, pode decidir entre “Dispensar”, “Acessar” ou “Excluir”. Ao clicar em “Dispensar”, o administrador ignora a denúncia feita pelo usuário. Clicando em “Acessar”, o administrador será direcionado ao endereço *web* da vaga denunciada. Já ao clicar em “Excluir”, o administrador acata a denúncia e remove o anúncio da vaga da página principal.

Figura 10 – Página Administrativa: exibe denúncias de usuários e disponibiliza ações para o administrador.



Fonte: Própria

5 CONSIDERAÇÕES FINAIS

O sistema desenvolvido neste trabalho propõe a facilitação do processo de busca por vagas no mercado de trabalho. Considerando a abundância de portais *on-line* que oferecem listagens de vagas, os usuários frequentemente enfrentam dificuldades

durante a procura, precisando navegar por diversos sites até encontrar os anúncios que melhor se adequam ao seu perfil profissional.

Para a construção do sistema, foram utilizadas tecnologias amplamente difundidas no contexto da programação atual, que são gratuitas ou possuem versões gratuitas, como: HTML, Node.js, MongoDB e a IDE Visual Studio Code. Além disso, foram empregadas bibliotecas que auxiliaram na criação do website e na extração de dados por meio de *web scraping*: Express, Axios e Cheerio.

Com o auxílio da classificação das vagas por meio da API *Natural Language AI* do Google, foi possível categorizá-las, facilitando a filtragem por área de interesse do candidato. Além disso, o portal desenvolvido proporciona uma experiência de navegação intuitiva e prática.

Para trabalhos futuros, recomenda-se o desenvolvimento de novos algoritmos que possibilitem a extração de dados de um número maior de portais de vagas, enriquecendo a base de anúncios disponível. Além disso, a implementação de uma funcionalidade de notificação por e-mail, que alerta o candidato assim que uma vaga for classificada como pertencente à sua área de interesse, aumentando a praticidade e, por fim, a avaliação da usabilidade da solução pelo usuário final.

Por fim, a construção deste trabalho resultou em um sistema prático e eficiente que auxilia o candidato na busca por uma vaga de trabalho. Além disso, representou uma experiência enriquecedora no âmbito pessoal, proporcionando o aprofundamento em metodologias e tecnologias amplamente utilizadas no mercado de trabalho.

REFERÊNCIAS

COCÓN, Felipe et al. Web scraping: Uso de plataformas de extracción de datos aplicadas a un sitio web sobre profesiones en México. **RISTI - Revista Ibérica de Sistemas e Tecnologias de Informação**, p. 61–73, 2023. DOI: 10.17013/risti.52.61-73.

DUARTE NETO, A.; BANDEIRA, P. S. R. de S.; MACÊDO, M. E. C. Novas Ferramentas para encontrar Talentos:Recrutamento e Seleção On-Line. **Revista Multidisciplinar e de Psicologia**, v. 14, n. 50, p. 964–974, 2020. DOI: 10.14295/online.v14i50.2507.

GOOGLE. **Cloud Natural Language API**. [S.l.: s.n.], 2025. <https://cloud.google.com/natural-language/docs/reference/rest>. [Online; acessado em 12 de janeiro de 2025].

GUIMARÃES, Bruna. **Recrutamento Online: entenda as vantagens e desvantagens**. [S.l.: s.n.], 2024. <https://www.gupy.io/blog/recrutamento-online>. [Online; acessado em 13 de janeiro de 2025].

KUMAR, Ashutosh; CHAUHAN, Kinshuk; GREWAL, Jaspreet Kaur. **Web scraping job portals**. [S.l.: s.n.], abr. 2024. Disponível em: <<https://doi.org/10.56155/978-81-955020-7-3-25>>.

KUMAR, Ashutosh; KINSHUK, Chauhan; JASPREET, Grewal. Web Scraping Job Portals. **Computing and Intelligent Systems, SCRS, India**, p. 291–303, 2023. DOI: 10.56155/978-81-955020-7-3-25.

RECHE, Grazielli. **E-recrutamento: a internet como ferramenta no recrutamento e seleção**. [S.l.: s.n.], 2011. <https://www.administradores.com.br/artigos/e-recrutamento-a-internet-como-ferramenta-no-recrutamento-e-selecao>. [Online; acessado em 11 de janeiro de 2025].

SINHA, Saumya; SAXENA, Rashee; GARG, Hitesh Kr. Smart Job Search Engine Using Web Scrapping. **International Journal of Advances in Engineering and Management (IJAEM)**, p. 586–591, 2021. DOI: 10.35629/5252-0306586591.