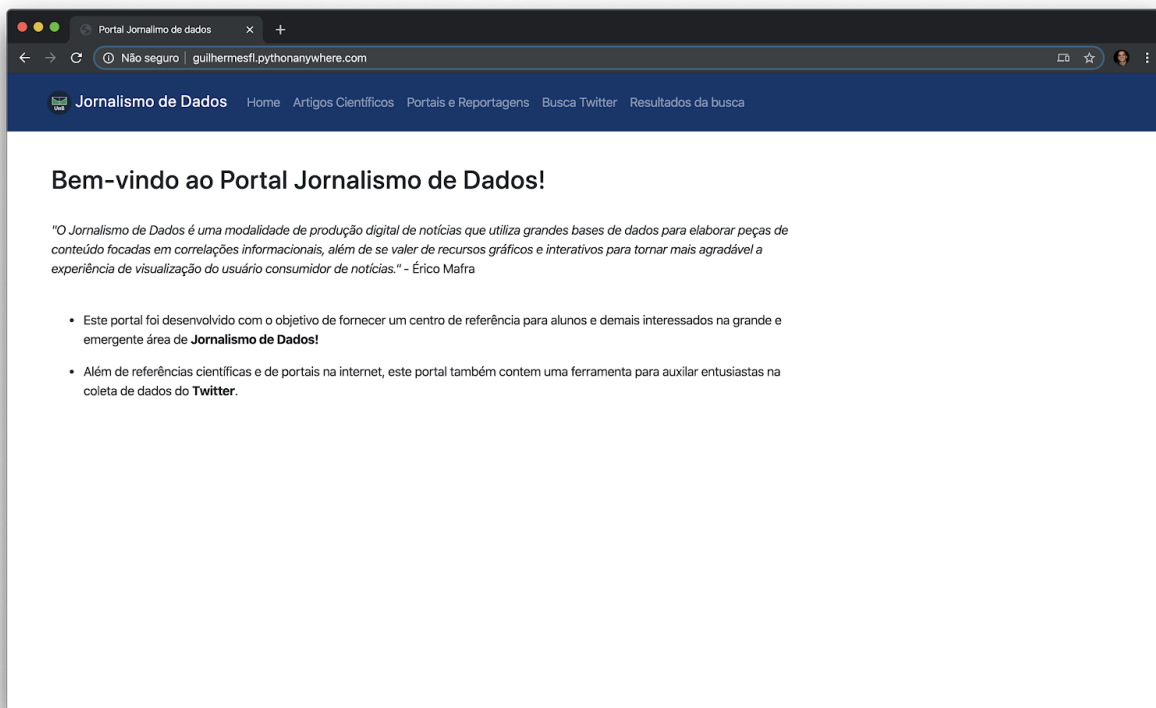


Documentação - Portal Jornalismo de Dados

Tópicos Avançados em Computadores - Programação Web



Guilherme da Silva Fontes Lopes - 150128215
Alex Nascimento Souza - 150143672

Sumário

1. Motivação
2. Estrutura do Projeto
3. Funcionamento
4. Melhorias futuras
5. Referências

1. Motivação

"O Jornalismo de Dados é uma modalidade de produção digital de notícias que utiliza grandes bases de dados para elaborar peças de conteúdo focadas em correlações informacionais, além de se valer de recursos gráficos e interativos para tornar mais agradável a experiência de visualização do usuário consumidor de notícias." - Érico Mafra

O principal objetivo deste projeto é desenvolver um portal de referência para jornalistas de dados. O portal contará com referências para artigos científicos e fontes de informações sobre jornalismo de dados. O portal também conterá uma ferramenta que auxilie jornalistas de dados a coletarem dados de interesse do Twitter.

2. Estrutura do Projeto

O projeto contém dois componentes principais que são interligados: o **Portal Jornalismo de Dados** e a **Ferramenta de Raspagem do Twitter**. Por meio do portal, é possível utilizar ambos os sistemas ao mesmo tempo.

2.1 Portal Jornalismo de Dados

O Portal Jornalismo de Dados contém um acervo de artigos científicos, portais e reportagens para auxiliar o desenvolvimento do profissional de jornalismo de dados. Além disso, fornece uma ferramenta capaz de coletar dados de Tweets públicos para que sejam feitos estudos posteriores pelo jornalista.

2.1.1 Descrição Técnica

O projeto foi desenvolvido utilizando as seguintes ferramentas:

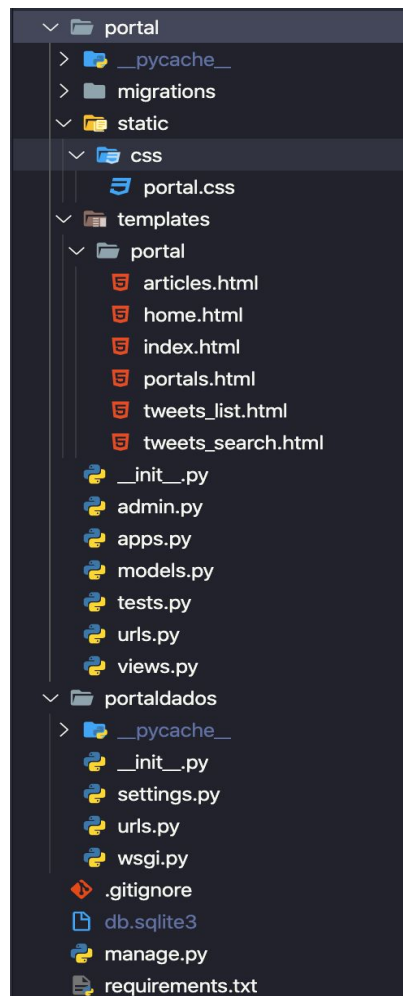
- **Python versão 3.7.3**
- **Django versão 2.2.4**
- **HTML5**
- **CSS3**

O projeto inteiramente desenvolvido utilizando as ferramentas disponíveis no Django.

Após a geração do projeto utilizando o comando

```
django-admin startproject portaldados .
```

O projeto seguiu a estrutura de arquivos proposta no Django e teve como resultado a seguinte organização:



Cada pasta na estrutura é responsável por armazenar e separar algum componente do sistema. Temos então o seguinte objetivo por pasta:

/static

- Armazena os arquivos estáticos do sistema. No nosso caso, somente foi necessário armazenar o arquivo de estilo **portal.css**

/templates

- Armazena os arquivos HTML do projeto. O arquivo HTML contém a estrutura que será apresentada no sistema.
- O arquivo **index.html** contém a estrutura geral, e os demais arquivos contém as estruturas das abas de navegação do sistema.

arquivos.py

- Os arquivos .py contém a maioria das configurações do próprio Django. Somente os arquivos **models.py**, **urls.py** e **views.py** foram alterados para que o Django fosse capaz de renderizar os templates e os dados de forma correta.

Dentro do arquivo **views.py**, estão contidas as funções **tweets_search** e **query_constructor**, responsáveis por realizar a lógica que monta a query para busca no Twitter. As demais funções são responsáveis por renderizar os templates de acordo com a url de busca.

Adição de novos itens ao acervo

Para adicionar itens seja para o acervo de artigos científicos ou portais e reportagens, é necessário utilizar o **Django Admin**. O Django admin é acessado por meio da url:

https://nome_do_site_de_hospedagem/admin/

Acessando esse link e adicionando as credenciais de administrador, o usuário será redirecionado para a seguinte página:

Administração do Django

Administração do Site

AUTENTICAÇÃO E AUTORIZAÇÃO		
Grupos	+ Adicionar	 Modificar
Usuários	+ Adicionar	 Modificar
PORTAL		
Articles	+ Adicionar	 Modificar
References	+ Adicionar	 Modificar
Tweets	+ Adicionar	 Modificar

Após isso, basta selecionar o que se deseja modificar, e então a nova referência será automaticamente adicionada ao acervo geral e exibida no porta.

Hospedagem

- O sistema foi hospedado no site de hospedagem gratuito <https://www.pythonanywhere.com/>.
- Para encontrar uma descrição completa de como realizar o deploy do sistema, basta olhar o excelente tutorial disponibilizado em: <https://tutorial.djangogirls.org/pt/deploy/>

2.2 Ferramenta de Raspagem do Twitter

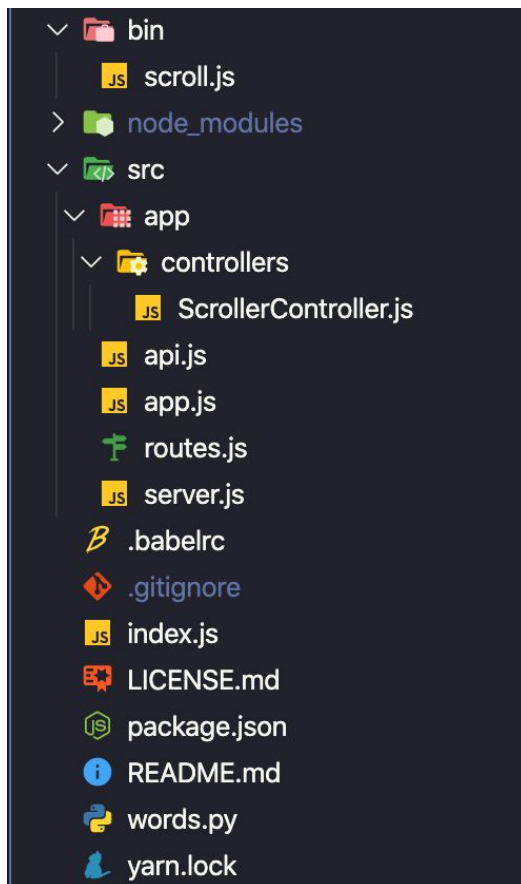
A ferramenta de raspagem é uma aplicação desenvolvida utilizando o recurso de busca avançada do Twitter com o intuito de fornecer um mecanismo de busca de dados para que os jornalistas possam encontrar informações de interesse de uma forma mais simples.

2.2.1 Descrição Técnica

O projeto foi desenvolvido utilizando as seguintes ferramentas:

- **NodeJs versão 10.16.3**
- **Express versão 4.17.1**
- **Adaptação do scroller público** <https://github.com/eyqs/twitter-scroller>

Essa parte do sistema contém um servidor que recebe como requisição do portal a busca avançada a qual deseja-se que seja realizada no Twitter e retorna os Tweets como resposta. A estrutura do projeto ficou da seguinte forma:



Assim como no portal, cada pasta ou arquivo é responsável por armazenar e separar um tipo de funcionalidade ou objetivo do sistema. A seguir é fornecido a descrição dos principais arquivos e pastas:

/controllers

- Contém o arquivo responsável por obter os parâmetros da busca avançada, passar os mesmos para o algoritmo de busca no Twitter e retornar os Tweets de resposta.

routes.js

- Arquivo contém as rotas disponíveis do servidor

server.js

- Arquivo responsável por iniciar o servidor na porta 3333

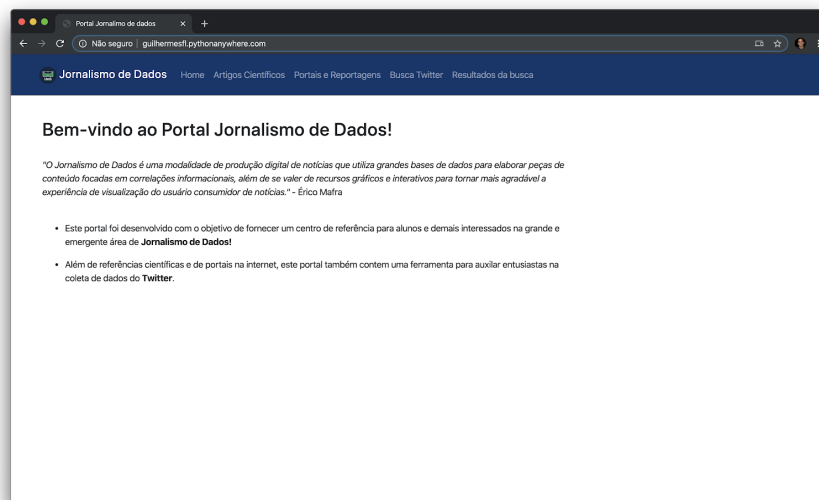
index.js

- É nesse arquivo que de fato é realizada a busca no Twitter. A busca consiste em se gerar um navegador controlado pelo programa e realizar o scroll na página para que todos os tweets sejam raspados.

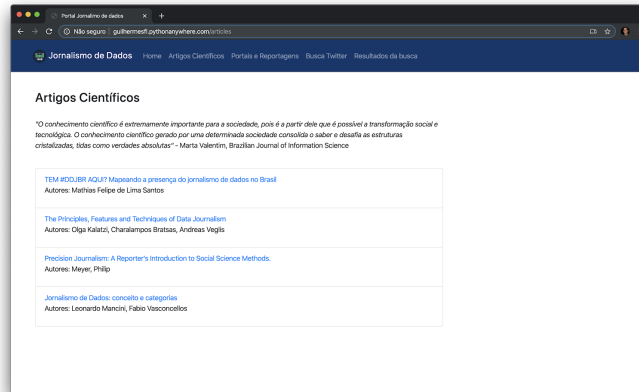
3. Funcionamento

O sistema é utilizado através do **Portal Jornalismo de Dados**. Sendo assim, é necessário que ambos o portal e o servidor de raspagem estejam rodando. Com o portal rodando, é possível navegar entre as abas para acessar as funcionalidades do sistema. A seguir seguem as abas disponíveis:

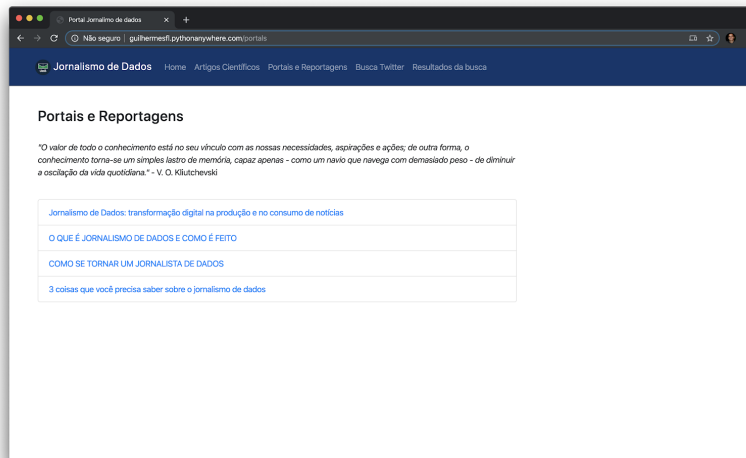
Home Screen



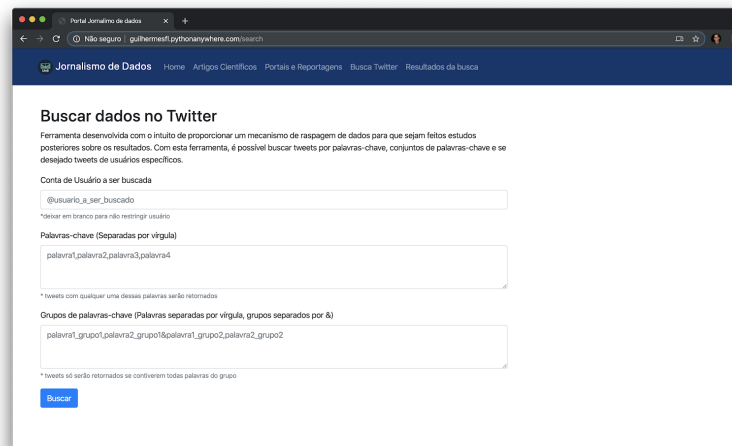
Acervo de Artigos Científicos



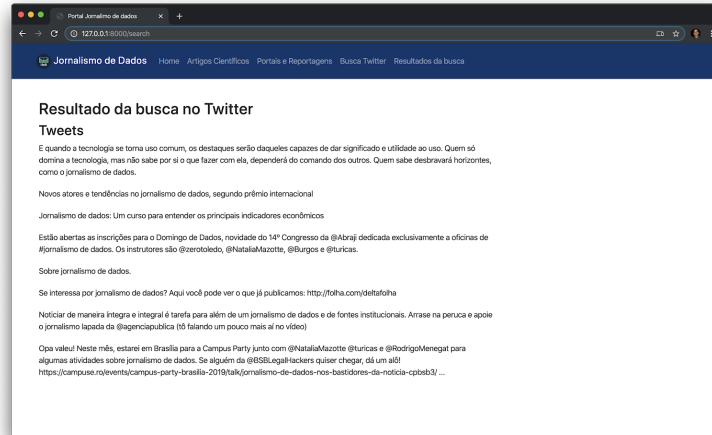
Acervo de Portais e Reportagens



Ferramenta de Raspagem



Resultados da Raspagem



Para utilizar a ferramenta de raspagem, basta definir as **palavras-chave** ou **conjunto de palavras-chave** que se deseja utilizar, a **data de início** e **data de fim** da busca, e o **usuário** específico, caso seja de interesse.

Após definidos os parâmetros da busca, basta clicar em **BUSCAR** e esperar que os resultados sejam retornados.

4. Melhorias Futuras

Visto que o sistema foi desenvolvido ao longo de um curto semestre e dada a enorme quantidade de informações e refinamentos que ainda podem ser feitos no sistema, aqui estão algumas possibilidades de melhorias que futuros desenvolvedores podem realizar:

- Melhorar a performance da busca no twitter.

A busca ainda tem limitações de performance, logo, dependendo da informação de interesse, é possível que se demore um pouco para retornar os resultados.

- Site responsivo

O site foi desenvolvido visando-se a utilização em computadores de mesa(Desktops) o que torna a visualização em dispositivos menores não otimizada. Tornar o site responsivo possibilitaria o uso em celulares, por exemplo.

- Popular os artigos e referências

O acervo contido ainda é escasso, e deve ser populado rotineiramente para manter o portal sempre atualizado.

- Download dos resultados

Atualmente, os resultados somente são retornados em forma de texto para o portal. Seria interessante que fosse possível baixar um arquivo contendo os resultados.

- Expansão da ferramenta de raspagem para outras plataformas

O acervo do portal atualmente é populado manualmente, caso fossem desenvolvidas novas ferramentas, esse processo poderia ser otimizado. Também poderia ser feita uma busca no Facebook, para se variar a fonte de dados.

- Adicionar ferramentas e guias para análise dos dados coletados

Adição de aba contendo acervo para ferramentas e guias de análise de dados

- Adicionar referências para cursos ou até criação de curso

Adição de aba contendo acervo para cursos de jornalismo de dados ou até mesmo integração de um curso na plataforma.

5. Referências

As seguintes referências podem auxiliar aos futuros desenvolvedores que desejem incrementar a plataforma:

1. Tutorial de Django - <https://tutorial.djangogirls.org/pt/>
2. Tutorial de Express - https://developer.mozilla.org/en-US/docs/Learn/Server-side/Express_Nodejs
3. Hospedagem de sites - <https://help.pythonanywhere.com/pages/>
4. APIs Rest - <https://www.codementor.io/olatundegaruba/nodejs-restful-apis-in-10-minutes-q0sgsfhbd>

5. Busca avançada Twitter -

<https://www.techtudo.com.br/dicas-e-tutoriais/noticia/2015/10/como-fazer-uma-busca-avancada-no-twitter.html>