



**Universidade do Minho**  
Escola de Ciências

## **JOB API EXPLORER**

Análise do conteúdo da API do itjobs.pt. Criação de uma CLI (Command Line Interface)

**Discentes:**

Filipe Araújo(A112467)

Marta Rocha (A112387)

Olga Marynch (A79235)

**Docentes:**

Prof. Tiago Baptista

Prof. Nuno Macedo

Prof. Pedro Rangel Henriques

Outubro, 2025

# **Índice**

Introdução .....	3
1 - Organização dos dados na API do ITjobs .....	4
2 – CLI – descrição dos comandos .....	5
2.1 Comando “dump” .....	5
2.2 Comando “top” .....	5
2.3 Comando “regime” .....	6
2.4 Comando “procurar” .....	6
2.5 Comando “skills” .....	7
2.6 Comando “search” .....	8
2.7 Comando search com o parâmetro --csv .....	8
CONCLUSÃO .....	9

## Introdução

Este projeto tem como objetivo analisar os dados disponibilizados pela API do site *itjobs.pt*, focando-se na recolha, armazenamento e processamento de anúncios de emprego na área das tecnologias de informação. Para facilitar o uso e garantir uma interação simples e eficiente, foi desenvolvida uma CLI (Command Line Interface) em Python, recorrendo aos módulos *typer*, *csv*, *requests*, *json*, *re* e *datetime*.

A CLI permite evitar chamadas excessivas à API, extrair informações úteis, filtrar dados relevantes e contar skills técnicas e comportamentais apresentadas nos anúncios. No fundo: automatizar o trabalho e transformar dados brutos em informação clara.

No primeiro capítulo será abordada a organização dos dados na API trabalhada.

Os capítulos seguintes descrevem os comandos implementados e incluem capturas de ecrã que ilustram a sua execução.

## 1 - Organização dos dados na API do ITjobs

A API do itjobs.pt segue uma estrutura organizada, pensada para fornecer informação sobre ofertas de emprego na área tecnológica. Cada pedido devolve dados em JSON, organizados em blocos que representam tanto o estado da resposta como o conteúdo da mesma. No centro está sempre um objeto onde encontramos metadados, tal como o número total de resultados disponíveis, a página atual e o tamanho da paginação. Logo a seguir surge o campo relevante para a exploração feita neste projeto: uma lista chamada *results*, que contém cada oferta de emprego individual.

Cada entrada dessa lista é um dicionário com tudo o que caracteriza a vaga: o ID da oferta, o título, a empresa, a localização, a data de publicação, os tags associados e a descrição completa do anúncio. Para completar o pacote, a API também inclui links úteis, como o endereço direto da candidatura e, às vezes, informação adicional sobre contacto ou requisitos mais específicos.

Por fim, a própria API disponibiliza endpoints diferentes conforme o tipo de pesquisa: pode-se pedir todas as ofertas, filtrar por empresa, por localização ou por tecnologias. Em todos os casos, o formato mantém-se consistente, o que facilita a análise automática — seja através de scripts simples, seja com ferramentas mais avançadas de exploração de dados.

## 2 – CLI – descrição dos comandos

### 2.1 Comando “dump”

Cria ou atualiza o ficheiro “*empregos.json*” com os dados mais recentes obtidos da API. Isto evita chamadas repetidas e ajuda a ultrapassar o limite de requisições da API.

*Exemplo de execução:*

```
(base) C:\Users\lenovo\OneDrive\2 ano-1 semestre\Ambientes e linguagem de programação\TP1>python jobscli.py dump
Arquivo 'empregos.json' criado/atualizado com sucesso!
```

### 2.2 Comando “top”

Recebe um número inteiro **n** e devolve as **n** ofertas mais recentes.

Apresenta o ID da vaga, o título do anúncio, a empresa e a data de publicação.

Foi por nós adicionada uma opção *opcional* `--csv` ao comando `top`, permitindo exportar os N empregos mais recentes para um ficheiro CSV. O CSV contém os campos principais de cada anúncio: título da oferta, empresa, descrição, data de publicação, salário e localização. Esta funcionalidade mantém o output em JSON no terminal, garantindo que a visualização imediata dos dados não é perdida mesmo quando o CSV é gerado. A geração do CSV ocorre apenas se o utilizador indicar a opção `--csv`, sendo implementada de forma compatível com Typer através de `typer.Option`. Esta abordagem facilita a análise posterior dos dados em ferramentas externas, como Excel ou bibliotecas Python de processamento de dados, permitindo integrar os resultados da API em workflows de análise de forma prática e eficiente.

*Exemplo de execução:*

```
(base) C:\Users\Asus\Downloads>python jobscli.py top 3 --csv -TOP3.csv
[
  {
    "ID da oferta": 507057,
    "Oferta de emprego": "Quality Manager (AI/Automation)",
    "Empresa": "Otonomee",
    "Data da publicação": "2025-11-16 16:33:39"
  },
  {
    "ID da oferta": 507058,
    "Oferta de emprego": "Product Owner/Project Manager",
    "Empresa": "Aubay",
    "Data da publicação": "2025-11-16 16:33:06"
  },
  {
    "ID da oferta": 507033,
    "Oferta de emprego": "IT Security Consultant",
    "Empresa": "Noesis Portugal",
    "Data da publicação": "2025-11-16 16:32:54"
  }
]
CSV exportado com sucesso para '-TOP3.csv'
```

## 2.3 Comando “regime”

Recebe o ID de um anúncio e devolve o tipo de regime associado: remoto, híbrido ou presencial. Para efeitos de teste, podem ser utilizados os ID devolvidos no comando anterior – comando “top”.

*Exemplo de execução:*

```
(base) C:\Users\lenovo\OneDrive\2 ano-1 semestre\Ambientes e linguagem de programação\TP1>python jobscli.py regime 507035  
Remoto
```

## 2.4 Comando “procurar”

Utilizado por nós como um comando intermédio para obter descrições de anúncios que incluíssem as seguintes expressões: *"Experiência"*, *"Conhecimentos"*, *"Forte"*, *"valoriza"*, *"Domínio"*, *"skills"*, *"experience"*, *"expertise"*, *"looking for"*. Motivo: redução do campo de análise de expressões regulares associadas a *skills*, uma vez que nem todos os anúncios incluem as *skills* pretendidas. Objetivo: criação de uma lista concreta de *skills* (tecnológicas e comportamentais). Dado que o output desta pesquisa permaneceu extenso, foi utilizado o apoio da Inteligência Artificial para filtrar as palavras de interesse de forma eficiente.

*Exemplo de execução:*

```
(base) C:\Users\Asus\Downloads>python jobscli.py procurar  
[  
    "<p>About The Role</p><p>Job Title Quality Manager </p><p>Location Remote within Portugal </p><p>Terms  
is role, you'll bring a strong blend of technical expertise, leadership, and a passion for continuous improvement. You  
paced environment. </p><ul><li>Analytical Mindset: You approach problems with logic and data. You're skilled at inter-  
rm decisions and continuously seek ways to improve processes through data-driven strategies.</li><li>Excellent Communi-  
ces—from frontline agents to senior leadership and clients. You're confident in presenting findings, facilitating dis-  
ncies and errors, and you take pride in maintaining high standards. Whether reviewing audit results or refining SOPs,  
roach: You see the bigger picture and understand how quality impacts business outcomes. While you can design long-ter-  
and support teams directly.</li><li>Collaborative and Influential: You build strong relationships across departments  
artnership and shared goals.</li><li>Self-Motivated and Independent: You thrive in a remote work environment, managing  
h on commitments, and proactively seek out opportunities to add value.</li><li>Tech-Savvy and Innovation-Driven: You  
and quality monitoring tools, and you actively explore new solutions to enhance efficiency and scalability.</li><li>  
vering exceptional experiences to clients and end-users. You advocate for the customer in every decision and ensure th  
years of experience in quality assurance or operations management, preferably within the BPO or customer service ind  
in leading cross-functional teams and managing quality programs across multiple clients.</li><li>Strong background in  
Remote work based in Portugal</li><li>A competitive salary </li><li>Flexible benefits</li><li>Equipment provided</li>  
al growth</li><li>Fun company events and team outings</li><li>Autonomy and Responsibility </li></ul><p>Send y
```

## 2.5 Comando “skills”

Recebe duas datas no formato AAAA-MM-DD e conta quantas vezes cada skill aparece nos anúncios publicados dentro desse intervalo. Para tornar os testes mais eficientes, recomenda-se utilizar datas próximas da atual, dado que novos anúncios são publicados diariamente.

*Exemplo de execução:*

```
(base) C:\Users\lenovo\OneDrive\2 ano-1 semestre\Ambientes e linguagem de programação\TP1>python jobscli.py skills 2025-11-14 2025-11-15
{
    "Python": 6,
    "Análise": 2,
    "Inovação": 2,
    "NAS": 2,
    "JavaScript": 1,
    "SQL": 3,
    "SQL Server": 1,
    "C": 2,
    "CSS": 1,
    "Flexibilidade": 1,
    "Gestão": 1,
    "Comunicação": 3,
    "Problem-solving": 2,
    "Communication skills": 3,
    "Frameworks": 2,
    "Desenho": 1,
    "Troubleshooting": 1,
    "Workflows": 1,
    "Data Platform": 1,
    "Databricks": 1,
    "Monitoring": 2,
    "Apache": 1,
    "Java": 1,
    "Git": 1,
    "NoSQL": 2,
    "Data engineering": 1,
    "Data modeling": 1,
    "APIs REST": 1,
    "Embedded systems": 1,
    "Software architecture": 1,
    "Software development": 1,
    "Leadership": 1,
    "Autonomy": 1,
    "Planning": 1,
    "Training": 2,
    "Organization": 1,
    "Containers": 1,
    "CI/CD": 1,
    "SaaS": 1,
```

## 2.6 Comando “search”

O comando "search" permite pesquisar ofertas de emprego a tempo parcial (part-time) com base numa localidade e empresa específicas. O utilizador deve indicar três argumentos para usar este comando: a localidade, o nome da empresa e o número máximo de anúncios. A estrutura do comando é a seguinte: `python jobscli.py search localidade empresa n`. Se nenhum anúncio for encontrado, o sistema retorna a mensagem "Nenhum resultado encontrado".

Caso contrário, os anúncios são apresentados em formato JSON no terminal, com a substituição tag HTML “</p> por duplo espaço e as restantes eliminadas.

```
(base) C:\Users\lenovo\OneDrive\2 ano-1 semestre\Ambientes e linguagem de programação\TP1>python jobscli.py search Lisboa "Master.D" 10
[
  {
    "titulo": "Formador na área de Inteligência Artificial",
    "empresa": "Master.D",
    "descrição": "É apaixonado/a por tecnologia, inovação e ensino? O futuro está a ser escrito com Inteligência Artificial, e queremos que faça parte dessa jornada! O nosso grupo internacional na área da formação procura um/a Formador/a de IA para liderar um curso abrangente de IA, onde são abordados des de os fundamentos de Python e Machine Learning até tópicos avançados como Deep Learning e a aplicação prática da IA em diferentes setores. O que procura mos: Formação em Engenharia Informática, Ciências da Computação, Matemática, Estatística ou áreas relacionadas; Certificações em Machine Learning / Deep Learning (valorizadas); Experiência em projetos de IA, desenvolvimento de software ou análise de dados; Domínio de Python e experiência em Machine Learning e Deep Learning (CNNs, RNNs, etc.); CCP (Certificado de Competências Pedagógicas) - obrigatório; Experiência comprovada como formador/a e gosto pel a partilha de conhecimento. O que oferecemos: Integração num ambiente dinâmico, inovador e internacional; Contrato de prestação de serviços em regim e part-time; Teletrabalho; Um ambiente colaborativo, com espaço para crescer e inovar. Se gosta de ensinar e inspirar, e quer contribuir para o futu ro da formação em Inteligência Artificial, envie o seu CV para recrutamento.formacao.pt@gmail.com com a referência IA/TUT. ",
    "data de publicação": "2025-11-14 21:57:31",
    "salário": null,
    "localização": "Lisboa"
  }
]
```

## 2.7 Comando search com o parâmetro --csv

A partir do comando “`search`” é acrescentado o parâmetro `--csv`, deixando a estrutura do comando como: `python jobscli.py search localidade empresa n --csv`. Permitindo, então, a criação do ficheiro csv “`search_results.csv`” com os dados dos n(ou menos) anúncios que cumprem os requisitos do comando.

```
(base) C:\Users\lenovo\OneDrive\2 ano-1 semestre\Ambientes e linguagem de programação\TP1>python jobscli.py search Lisboa "Master.D" 10 --csv
Resultados exportados para 'search_results.csv'
```

## CONCLUSÃO

A aplicação desenvolvida permite analisar de forma eficiente os conteúdos disponibilizados pela API do itjobs.pt, evitando requisições desnecessárias e facilitando a extração de informação relevante. A CLI torna o processo simples, modular e reutilizável — desde a recolha dos dados até à contagem de *skills* técnicas e comportamentais.

O resultado é uma ferramenta prática que ajuda a compreender tendências, requisitos e padrões das ofertas de emprego em tecnologia.

Durante o desenvolvimento, enfrentámos algumas dificuldades significativas. A primeira foi o atingimento do limite de requisições da API, uma vez que inicialmente utilizámos ciclos de repetição para recolher os dados. Para contornar este problema, adaptámos o código com a ajuda do professor, implementando o *user\_agent* e criando um ficheiro local com os dados, de forma a evitar múltiplas chamadas à API. Para além disso, surgiu a dificuldade de identificar expressões regulares que representassem *skills*, dado que cada anúncio utiliza descrições distintas e em duas línguas, português e inglês. Para lidar com esta complexidade, procedemos a uma filtragem intermédia e, posteriormente, recorremos à inteligência artificial para gerar a lista final de *skills* a partir do conteúdo previamente filtrado.

Apesar das dificuldades, estas experiências revelaram-se extremamente úteis para o desenvolvimento das nossas competências técnicas e da capacidade de adaptação a desafios inesperados. O trabalho contribuiu ainda para aumentar a nossa familiaridade e segurança na utilização do GitHub, fortalecendo boas práticas de versionamento e colaboração em projetos de programação.