teste-engenheiro-de-dados-observatorio-da-industria 1

Configuração Adequada do Docker-Compose

Windows 11

Durante o processo de configuração do Docker-Compose no sistema operacional Windows 11, encontrei diversas dificuldades. Ao tentar executar o Docker-Compose pelo WSL, fui enviado para uma tela que apenas funcionaria ao utilizar o WSL2 com default. Percebi que isso exigiria um tempo adicional para ser resolvido.

Buscando alternativas, optei por utilizar o Docker Desktop e o PowerShell do Windows. No entanto, encontrei várias incompatibilidades ao tentar subir o Docker-Compose, especialmente ao definir a variável de ambiente com o comando `echo -e "AIRFLOW_UID=$(id -u)" > .env`.

Mesmo após conseguir subir o Airflow corretamente, percebi que a configuração de um ambiente no Windows exigiria muito esforço, considerando que estou mais acostumado ao universo Linux

Linux

1. Construir e Iniciar os Containers

```
docker-compose up -d
```

Durante a execução, recebi um `WARNING` que por si só não quebrava o ambiente, mas indicava possíveis problemas a serem resolvidos posteriormente. Além disso, um `ERROR` foi informado, indicando que a imagem necessária não foi encontrada.

```
support@support:~/git/teste-engenheiro-de-dados-observatorio-da-industria$ docker-compose up -d
WARNING: The AIRFLOW_UID variable is not set. Defaulting to a blank string.
Creating network "teste-engenheiro-de-dados-observatorio-da-industria_default" with the default driver
Pulling postgres (postgres:)...
latest: Pulling from library/postgres
f11c1adaa26e: Pull complete
76ce212b9153: Pull complete
919ca406a058: Pull complete
6b7a1245fe71: Pull complete
8064ffe06c65: Pull complete
4b5c59f2d82c: Pull complete
fe72764b9070: Pull complete
6ef8e2c0f4d9: Pull complete
e71fe9d7ff11: Pull complete
f3225d69190d: Pull complete
2bf90d17afc8: Pull complete
d3aee49eb079: Pull complete
e1e856658919: Pull complete
95c2c2ef9f02: Pull complete
Digest: sha256:0aafd2ae7e6c391f39fb6b7621632d79f54068faebc726caf469e87bd1d301c0
Status: Downloaded newer image for postgres:latest
Pulling airflow-init (local-cluster-airflow:1.0.0)...
ERROR: The image for the service you're trying to recreate has been removed. If you continue, volume data could be lost. Consider backing up your data before co

Continue with the new image? [yN]n
ERROR: pull access denied for local-cluster-airflow, repository does not exist or may require 'docker login': denied: requested access to the resource is denied
support@support:~/git/teste-engenheiro-de-dados-observatorio-da-industria$
```

2. Construir a imagem local-cluster-airflow

`docker build -t local-cluster-airflow:1.0.0 .`

O build foi finalizado com sucesso so apresentou alguns `warnings` apos isso realizar o `docker-compose up -d` para subir todos os containers.

```
support@support:~/git/teste-engenheiro-de-dados-observatorio-da-industria$ docker build -t local-cluster-airflow:1.0.0 .
[+] Building 1.1s (10/10) FINISHED
 => [internal] load build definition from Dockerfile
 => => transferring dockerfile: 783B
 => [internal] load metadata for docker.io/apache/airflow:slim-2.8.3-python3.11
 => [internal] load .dockerignore
 => => transferring context: 2B
 => [internal] load build context
 => => transferring context: 38B
 => [1/5] FROM docker.io/apache/airflow:slim-2.8.3-python3.11@sha256:09b7216cd768d6ea48c3f1102fc800caa012b5e2bac10b0e89263c780c364345
 => CACHED [2/5] RUN set -eux;     apt-get update &&     apt-get install --yes --no-install-recommends     wget     openjdk-17-jdk
 => CACHED [3/5] RUN wget https://dlcdn.apache.org/hadoop/common/hadoop-3.3.6/hadoop-3.3.6.tar.gz && tar xf hadoop-3.3.6.tar.gz && mv hadoop-3.3.6 /opt && rm -rf hadoop-3.3.6.tar.gz
 => CACHED [4/5] COPY requirements.txt /
 => CACHED [5/5] RUN set -eux;     pip install --no-cache-dir "apache-airflow==2.8.3" -r /requirements.txt
 => exporting to image
 => => exporting layers
 => => writing image sha256:f2d785b0a5dcb1e913b0dc6ae747dafeab5a0850456349158296914d0fdc70be
 => => naming to docker.io/library/local-cluster-airflow:1.0.0

 5 warnings found (use --debug to expand):
 - LegacyKeyValueFormat: "ENV key=value" should be used instead of legacy "ENV key value" format (line 3)
 - LegacyKeyValueFormat: "ENV key=value" should be used instead of legacy "ENV key value" format (line 4)
 - LegacyKeyValueFormat: "ENV key=value" should be used instead of legacy "ENV key value" format (line 5)
 - LegacyKeyValueFormat: "ENV key=value" should be used instead of legacy "ENV key value" format (line 6)
 - LegacyKeyValueFormat: "ENV key=value" should be used instead of legacy "ENV key value" format (line 7)
support@support:~/git/teste-engenheiro-de-dados-observatorio-da-industria$ ~
```

`docker-compose up -d`

Houve um problema indicando que o serviço `airflow-init` não conseguia ler. Isso ocorreu porque meu ambiente utiliza uma distribuição Linux, e o módulo SELinux bloqueia esse tipo de operação ao mapear as pastas do contêiner com as da minha máquina.

Obs.: Esta parte foi especialmente difícil de corrigir. Precisei eliminar várias possibilidades, como permissões do próprio contêiner, até descobrir essa particularidade do SELinux. Para desabilitar temporariamente e corrigir esse problema, use o comando `setenforce 0`.

```
root@support:/home/support/git/teste-engenheiro-de-dados-observatorio-da-industria# podman-compose up
84c90e95e8ef2d10e3beb0e5bcfdb716f6424bdccd01a30fda80fbe8b902a6e3
cdd229cafb7e871c0bc9c7d5fa2e16d0ebb4a2fe5ef794e7d9c6eee2f16b736a
c24371038 6b00192b6bcb4eb8792a9dd8ee8b04e6eb93309d1c6b38b533711fa
4dc96e834dc0c6b53855d166bdc9e3ef320d120dd86335e12e1544af1908a544
55af9ac28411c11ecb8a3261620ff67c587734a87362bf9344e687de8a00ca32
[postgres]          | The files belonging to this database system will be owned by user "postgres".
[postgres]          | This user must also own the server process.
[postgres]          |
[postgres]          | The database cluster will be initialized with locale "en_US.utf8".
[postgres]          | The default database encoding has accordingly been set to "UTF8".
[postgres]          | The default text search configuration will be set to "english".
[postgres]          |
[postgres]          | Data page checksums are disabled.
[postgres]          |
[postgres]          | fixing permissions on existing directory /var/lib/postgresql/data ... ok
[postgres]          | creating subdirectories ... ok
[postgres]          | selecting dynamic shared memory implementation ... posix
[postgres]          | selecting default max_connections ... 100
[postgres]          | selecting default shared_buffers ... 128MB
[postgres]          | selecting default time zone ... Etc/UTC
[postgres]          | creating configuration files ... ok
[airflow-init]      | chown: cannot read directory '/sources/logs': Permission denied
[airflow-init]      | chown: cannot read directory '/sources/dags': Permission denied
[airflow-init]      | chown: cannot read directory '/sources/plugins': Permission denied
[airflow-init]      | The container is run as root user. For security, consider using a regular user account.
[postgres]          | running bootstrap script ... ok
[postgres]          | performing post-bootstrap initialization ... ok
[postgres]          | syncing data to disk ... ok
[postgres]          |
```

Apos isso funcionou corretamente.

```
support@support:~/git/teste-engenheiro-de-dados-observatorio-da-industria$ docker-compose up -d
WARNING: The AIRFLOW_UID variable is not set. Defaulting to a blank string.
Creating teste-engenheiro-de-dados-observatorio-da-industria_postgres_1 ... done
Creating teste-engenheiro-de-dados-observatorio-da-industria_airflow-init_1 ... done
Creating teste-engenheiro-de-dados-observatorio-da-industria_airflow-webserver_1 ... done
Creating teste-engenheiro-de-dados-observatorio-da-industria_airflow-scheduler_1 ... done
support@support:~/git/teste-engenheiro-de-dados-observatorio-da-industria$
```

3. Validadando se o ambiente está executando corretamente

`docker-compose ps`

O contêiner `airflow-init` não iniciou corretamente, como indica o estado `exit 1`. Isso mostra que houve um problema durante a inicialização dos contêineres. Vamos renovar todos os volumes para zerar o status e iniciar um ambiente limpo.

```
support@support:~/git/teste-engenheiro-de-dados-observatorio-da-industria$ docker-compose ps
                              Name                                            Command                  State        Ports
-------------------------------------------------------------------------------------------------------------------------------
teste-engenheiro-de-dados-observatorio-da-industria_airflow-init_1        /bin/bash -c mkdir -p /sou ...   Exit 0
teste-engenheiro-de-dados-observatorio-da-industria_airflow-scheduler_1   /usr/bin/dumb-init -- /ent ...   Exit 1
teste-engenheiro-de-dados-observatorio-da-industria_airflow-webserver_1   /usr/bin/dumb-init -- /ent ...   Exit 1
teste-engenheiro-de-dados-observatorio-da-industria_postgres_1            docker-entrypoint.sh postgres    Up (healthy)   5432/tcp
support@support:~/git/teste-engenheiro-de-dados-observatorio-da-industria$
```

Troubleshoot:

```
docker compose down --volumes --remove-orphans
docker rm $(docker ps -a -q)
```

Seguir a documentação do airflow para ver algo que precisa ser feito
https://airflow.apache.org/docs/apache-airflow/stable/howto/docker-compose/index.html

```
# Set a variavel para construçãode permissão dos containers nas pastas
echo -e "AIRFLOW_UID=$(id -u)" > .env

# Inicialiar o databse primeiro para realizar do migrate
docker compose up airflow-init
```

```
# Rodar os conatainers
docker compose up -d
```

```
airflow-init-1 exited with code 0
root@support:/home/support/git/teste-engenheiro-de-dados-observatorio-da-industria# docker compose up -d
[+] Running 4/4
 ✔ Container teste-engenheiro-de-dados-observatorio-da-industria-postgres-1           Healthy
 ✔ Container teste-engenheiro-de-dados-observatorio-da-industria-airflow-init-1       Exited
 ✔ Container teste-engenheiro-de-dados-observatorio-da-industria-airflow-webserver-1  Started
 ✔ Container teste-engenheiro-de-dados-observatorio-da-industria-airflow-scheduler-1  Started
root@support:/home/support/git/teste-engenheiro-de-dados-observatorio-da-industria#
```

Apos o *troubleshot* mostra que todos os containers estão rodando. Vamos iniciar o ambiente do airflow no browser em http://localhost:11000. Funcionando perfeitamente.

```
root@support:/home/support/git/teste-engenheiro-de-dados-observatorio-da-industria# docker-compose ps
                              Name                                            Command                  State                            Ports
-------------------------------------------------------------------------------------------------------------------------------------------------------------
teste-engenheiro-de-dados-observatorio-da-industria_airflow-init_1        /bin/bash -c if [[ -z "0"  ...   Exit 0
teste-engenheiro-de-dados-observatorio-da-industria_airflow-scheduler_1   /usr/bin/dumb-init -- /ent ...   Up (health: starting)   8080/tcp
teste-engenheiro-de-dados-observatorio-da-industria_airflow-triggerer_1   /usr/bin/dumb-init -- /ent ...   Up (health: starting)   8080/tcp
teste-engenheiro-de-dados-observatorio-da-industria_airflow-webserver_1   /usr/bin/dumb-init -- /ent ...   Up (health: starting)   0.0.0.0:11000->8080/tcp,:::11000->8080/tcp
teste-engenheiro-de-dados-observatorio-da-industria_airflow-worker_1      /usr/bin/dumb-init -- /ent ...   Up (health: starting)   8080/tcp
teste-engenheiro-de-dados-observatorio-da-industria_postgres_1            docker-entrypoint.sh postgres    Up (healthy)            5432/tcp
teste-engenheiro-de-dados-observatorio-da-industria_redis_1               docker-entrypoint.sh redis ...   Up (healthy)            6379/tcp
root@support:/home/support/git/teste-engenheiro-de-dados-observatorio-da-industria#
```

```
root@support:/home/support/git/teste-engenheiro-de-dados-observatorio-da-industria# docker-compose ps
                              Name                                            Command                  State                        Ports
-------------------------------------------------------------------------------------------------------------------------------------------------------
teste-engenheiro-de-dados-observatorio-da-industria-airflow-init-1        /bin/bash -c mkdir -p /sou ...   Exit 0
teste-engenheiro-de-dados-observatorio-da-industria-airflow-scheduler-1   /usr/bin/dumb-init -- /ent ...   Up             8080/tcp
teste-engenheiro-de-dados-observatorio-da-industria-airflow-webserver-1   /usr/bin/dumb-init -- /ent ...   Up             0.0.0.0:11000->8080/tcp,:::11000->8080/tcp
teste-engenheiro-de-dados-observatorio-da-industria-postgres-1            docker-entrypoint.sh postgres    Up (healthy)   5432/tcp
root@support:/home/support/git/teste-engenheiro-de-dados-observatorio-da-industria#
```

Apos executar todos os passos corretamente e resolver problemas que foram aparecendo e todos os containers sendo executado corretamente algo na aplicação não subiu.

Ooops!

```
Something bad has happened. For security reasons detailed information about the error is not logged.

  * You should check your webserver logs and retrieve details of this error.

  * When you get the logs, it might explain the reasons, you should also Look for similar issues using:

     * GitHub Discussions
     * GitHub Issues
     * Stack Overflow
     * the usual search engine you use on a daily basis

    All those resources might help you to find a solution to your problem.

  * if you run Airflow on a Managed Service, consider opening an issue using the service support channels

  * only after you tried it all, and have difficulty with diagnosing and fixing the problem yourself,
    get the logs with errors, describe results of your investigation so far, and consider creating a
    bug report including this information.

Python version: redact
Airflow version: redact
Node: redact
-------------------------------------------------------------------------------
Error! Please contact server admin.
```

Para otimizar e esclarecer que este passo não inclui a intervenção no container para corrigir possíveis problemas no *nginx* ou no *gunicorn* que estão servindo a aplicação, eu preferi consultar a documentação. Ela fornece as imagens e serviços necessários para executar o Airflow. Veja mais detalhes em: https://airflow.apache.org/docs/apache-airflow/stable/howto/docker-compose/index.html"

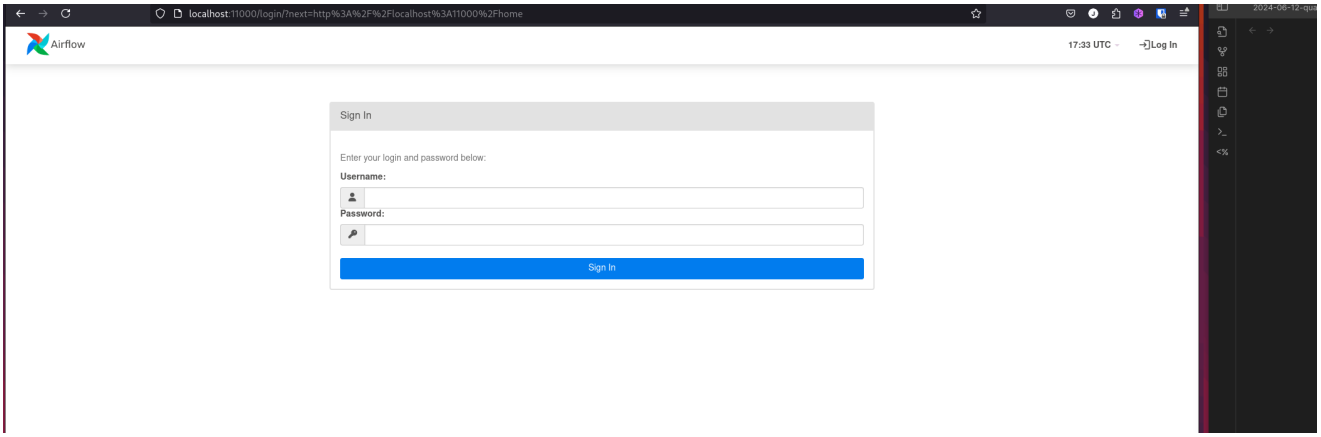
Após consultar a documentação e subir com a imagem estabelecida por lá os serviços estão funcionando corretamente. Chuto aqui dizer que isso possa de funcionar o primeiro e não funcionar na máquina a questão da incompatibilidade com a versão do linux.

```
root@support:/home/support/git/teste-engenheiro-de-dados-observatorio-da-industria# docker-compose ps
                                Name                                              Command               State                    Ports
-------------------------------------------------------------------------------------------------------------------------------------------------------
teste-engenheiro-de-dados-observatorio-da-industria_airflow-init_1       /bin/bash -c if [[ -z "0"  ...   Exit 0
teste-engenheiro-de-dados-observatorio-da-industria_airflow-scheduler_1   /usr/bin/dumb-init -- /ent ...   Up (healthy)   8080/tcp
teste-engenheiro-de-dados-observatorio-da-industria_airflow-triggerer_1   /usr/bin/dumb-init -- /ent ...   Up (healthy)   8080/tcp
teste-engenheiro-de-dados-observatorio-da-industria_airflow-webserver_1   /usr/bin/dumb-init -- /ent ...   Up (healthy)   0.0.0.0:8080->8080/tcp,:::8080->8080/tcp
teste-engenheiro-de-dados-observatorio-da-industria_airflow-worker_1      /usr/bin/dumb-init -- /ent ...   Up (healthy)   8080/tcp
teste-engenheiro-de-dados-observatorio-da-industria_postgres_1            docker-entrypoint.sh postgres    Up (healthy)   5432/tcp
teste-engenheiro-de-dados-observatorio-da-industria_redis_1               docker-entrypoint.sh redis ...   Up (healthy)   6379/tcp
root@support:/home/support/git/teste-engenheiro-de-dados-observatorio-da-industria#
```

```
root@support:/home/support/git/teste-engenheiro-de-dados-observatorio-da-industria# sudo netstat -tuln | grep LISTEN
tcp        0      0 127.0.0.54:53           0.0.0.0:*               LISTEN
tcp        0      0 0.0.0.0:5355            0.0.0.0:*               LISTEN
tcp        0      0 0.0.0.0:8080            0.0.0.0:*               LISTEN
tcp        0      0 0.0.0.0:27500           0.0.0.0:*               LISTEN
tcp        0      0 127.0.0.53:53           0.0.0.0:*               LISTEN
tcp        0      0 127.0.0.1:631           0.0.0.0:*               LISTEN
tcp6       0      0 :::5355                 :::*                    LISTEN
tcp6       0      0 :::8080                 :::*                    LISTEN
tcp6       0      0 ::1:631                 :::*                    LISTEN
root@support:/home/support/git/teste-engenheiro-de-dados-observatorio-da-industria#
root@support:/home/support/git/teste-engenheiro-de-dados-observatorio-da-industria#
```

Subiu na porta *8080* vou realizar alternação para porta 11000 para validar e concluir essa parte do problema proposto.

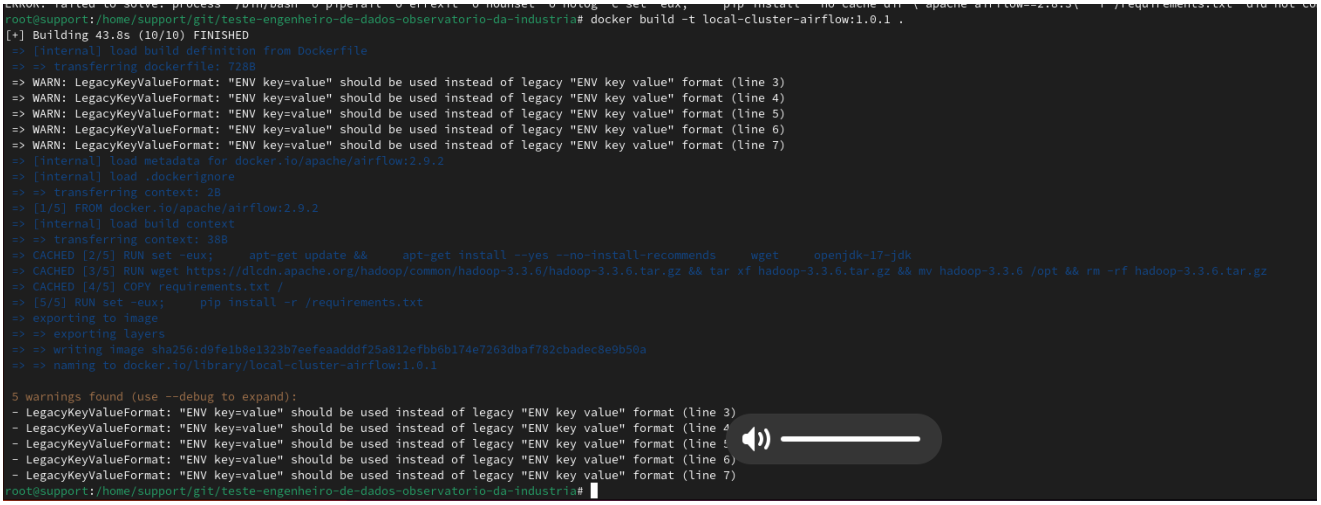
```
root@support:/home/support/git/teste-engenheiro-de-dados-observatorio-da-industria# docker-compose ps
                                Name                                              Command                  State                        Ports
-------------------------------------------------------------------------------------------------------------------------------------------------------------
teste-engenheiro-de-dados-observatorio-da-industria_airflow-init_1       /bin/bash -c if [[ -z "0"  ...   Exit 0
teste-engenheiro-de-dados-observatorio-da-industria_airflow-scheduler_1   /usr/bin/dumb-init -- /ent ...   Up (health: starting)   8080/tcp
teste-engenheiro-de-dados-observatorio-da-industria_airflow-triggerer_1   /usr/bin/dumb-init -- /ent ...   Up (health: starting)   8080/tcp
teste-engenheiro-de-dados-observatorio-da-industria_airflow-webserver_1   /usr/bin/dumb-init -- /ent ...   Up (health: starting)   0.0.0.0:11000->8080/tcp,:::11000->8080/tcp
teste-engenheiro-de-dados-observatorio-da-industria_airflow-worker_1      /usr/bin/dumb-init -- /ent ...   Up (health: starting)   8080/tcp
teste-engenheiro-de-dados-observatorio-da-industria_postgres_1            docker-entrypoint.sh postgres    Up (healthy)             5432/tcp
teste-engenheiro-de-dados-observatorio-da-industria_redis_1               docker-entrypoint.sh redis ...   Up (healthy)             6379/tcp
root@support:/home/support/git/teste-engenheiro-de-dados-observatorio-da-industria#
```



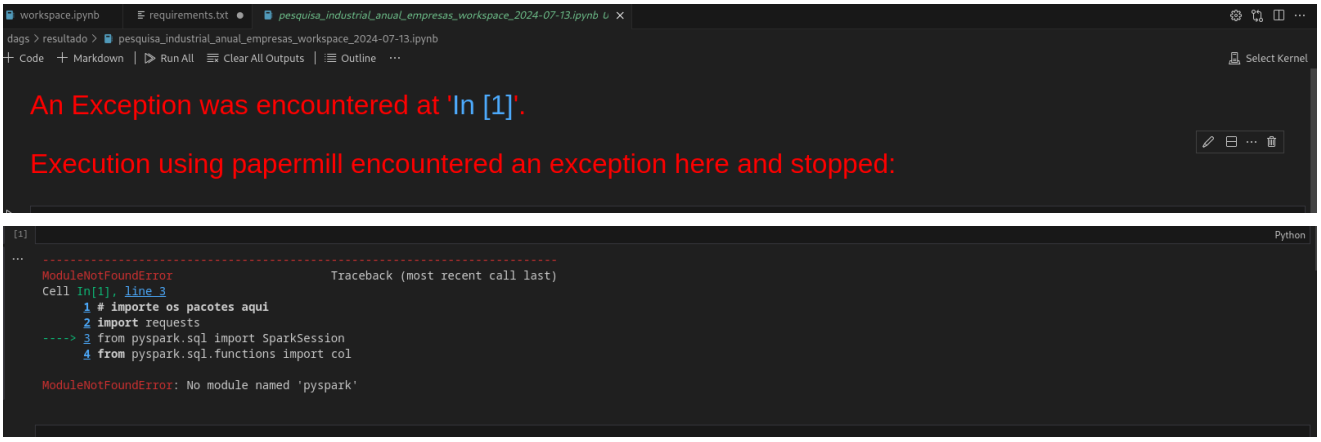
Extrair as informações da Pesquisa Industrial Anual - Empresas do ibge

Ao entrar na aplicação com *usuario: airflow* e *senha: airflow* mostra que houve um problema de importação devido o *ModuleNotFound* no pacote chamado *airflow.providers.papermil*

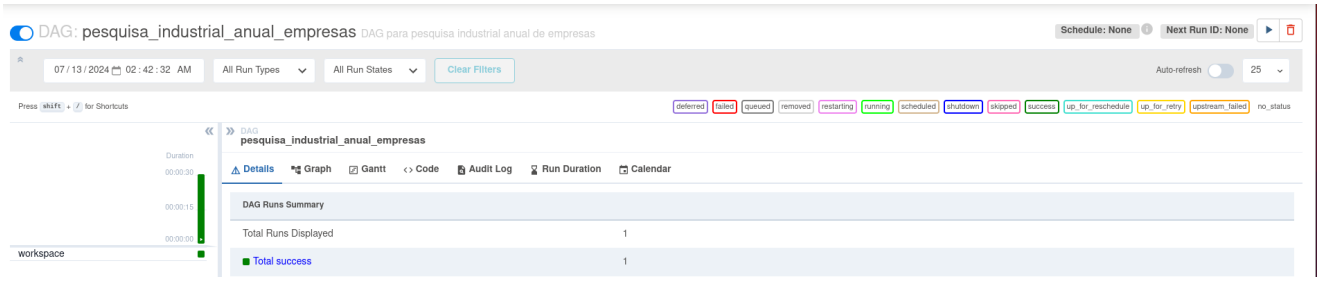
Troubleshoot:
Preciso realizar o build novamente da imagem do `local-cluster-airflow:1.0.0.1` utilizando a image base `apache/airflow:2.9.2` para instalar o novo pacote. Como os contêineres Docker permanecem em estado de execução, é necessário criar a imagem novamente com o novo pacote instalado.



Ao rodar a dag ainda informou o operator papermill notificava um erro que tinha erro na execução do notebook que não encontra a modulo pypspark para resolver esse problema preciso instalar em uma nova imagem, realizar o build e depois rodar novamente a DAG.



Após instalar o módulo `pyspark`, o DAG foi executado perfeitamente



Conclusão

Apesar de ter progredido até o passo anterior, deparei-me com diversos desafios na ambientação com o Airflow. Minha inexperiência com a ferramenta me levou a buscar materiais introdutórios para compreendê-la melhor. No entanto, minha familiaridade com o Python e Linux se mostrou um ponto forte nesse processo.

Referencias

https://airflow.apache.org/docs/apache-airflow/stable/index.html
https://papermill.readthedocs.io/en/latest/usage-parameterize.html
https://www.astronomer.io/docs/learn/execute-notebooks
https://github.com/apache/airflow/blob/providers-papermill/3.7.0/tests/system/providers/papermill/example_papermill.py