PROJET 5

OPENCLASSROOMS: FORMATION

INGÉNIEUR DATA

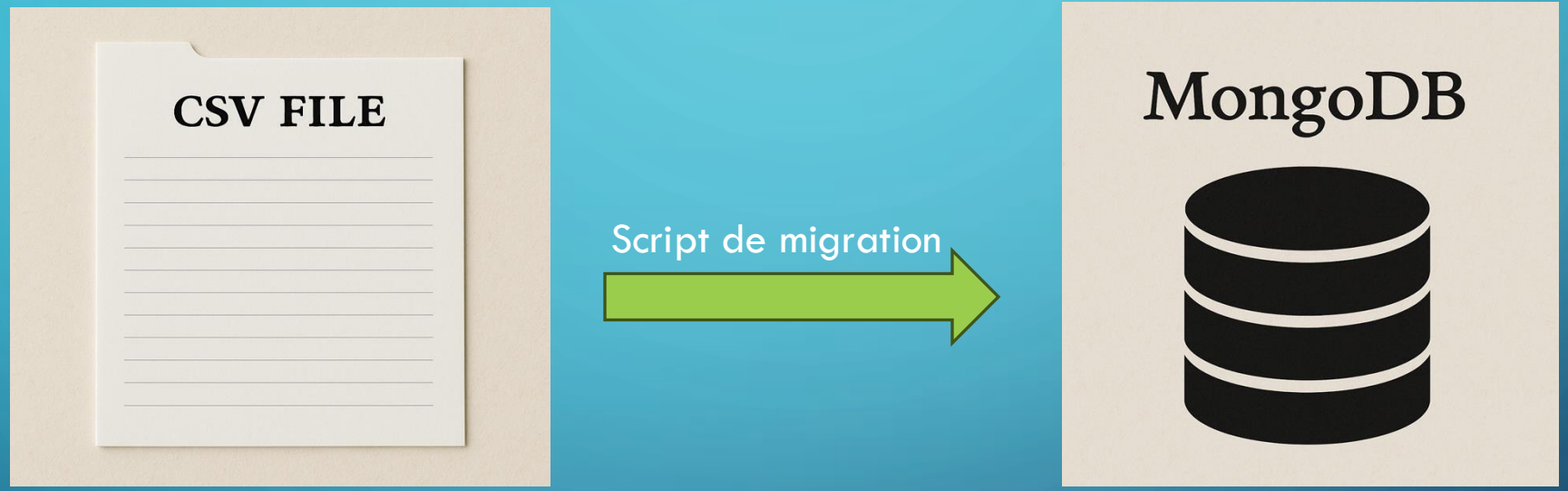
Contexte:

La société DataSoluTech est spécialisée dans la fourniture de solutions de gestion de données et d'analyse pour tous types d’entreprises.

But du projet:

Proposer une solution Big Data évolutive horizontalement pour un de ces clients.

CSV TO MONGO



https://github.com/nau81000/csv_to_mongo

CSV TO MONGO

- Conçu pour être utilisable pour différents jeux de données
- Facilement paramètrable par des variables d'environnement

GRANDES ÉTAPES DE LA MIGRATION AVEC PYTHON

Dotenv

Lecture des variables d'environnement:

- URI base MongoDB
- Nom base, nom de la collection, indexes
- Date pattern
- Schéma de données
- Utilisateurs

Pandas

- Lecture du fichier CSV
- Suppression des doublons

Pymongo

- Suppression de l'ancienne collection, indexes et utilisateurs
- Création de la collection, insertion des documents
- Création indexes
- Ajout des utilisateurs

TESTS

PYTEST

2 tests:

1. Vérification qu'il n'y a pas de valeurs manquantes
2. Vérification que le nombre de lignes du fichier CSV est égal au nombre de documents importés dans la collection

CAS D'ÉCOLE

Un jeu de données médicales

CRÉATION D'UNE BASE DES DONNÉES MONGODB

HEALTH

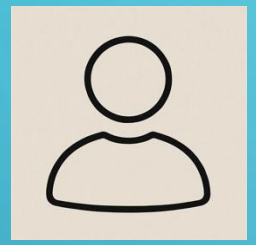
1 collection: PATIENTS

54966 documents

Schéma de données de la collection PATIENTS

```
{
   "_id"                    : "ObjectId",
   "Name"                   : "string",      "Age"                  : "int",
   "Gender"                 : "string",      "Blood Type"           : "string",
   "Medical Condition"      : "string",      "Insurance Provider"   : "string",
   "Admission" : {
      "Date of Admission"   : "date",        "Discharge Date"       : "date",
      "Hospital"            : "string",      "Doctor"               : "string",
      "Admission Type"      : "string",      "Room Number"          : "int",
      "Test Results"        : "string" ,     "Medication"           : "string",
     "Billing Amount"       : "double",
      }
}
```

2 UTILISATEURS RELIÉS À LA BASE HEALTH



admin

- Administrateur
- Droits de lecture écriture uniquement

user

- Utilisateur lambda
- Droits de lecture uniquement

SYSTÈMES D'AUTHENTIFICATION

Authentification	Utilisation principale	Mot de passe ?	SSO / External IDP	Sécurité	Remarques
SCRAM-SHA-1/256	Par défaut (utilisateurs MongoDB)	✅ Oui	❌ Non	🔒 Élevée (SHA-256)	Standard, sécurisé, local
x.509	Serveurs et clients avec certificats	❌ Non	❌ Non	🔒 Très élevée	Basé sur SSL/TLS, sans mot de passe
LDAP	Intégration avec Active Directory / LDAP	✅ Oui	⚠️ Partiel (via proxy)	🔒 Élevée	Centralisé, bon pour entreprises
Kerberos	Authentification réseau / entreprise	❌ Non	✅ Oui	🔒 Élevée	Permet le SSO, complexe à configurer
AWS IAM (Atlas)	Auth IAM via rôle (EC2, ECS, etc.)	❌ Non	✅ Oui	🔒 Élevée	Spécifique à MongoDB Atlas
OIDC (Atlas)	Auth via fournisseurs externes (OAuth)	❌ Non	✅ Oui	🔒 Élevée	Moderne, supporte Google, Okta, etc.

DÉPLOIEMENT

DESCRIPTION DU DOCKER-COMPOSE.YML

2 services

1 réseau

docker-compose.yml

```
services:
    db:
        container_name: mongo_db

        image: mongo:latest

        volumes:
            - ./mongo_data:/data/db

        restart: on-failure

        networks:
            - frontend

        ports:
            - "27017:27017"

        environment:
            MONGO_INITDB_DATABASE: health
            MONGO_INITDB_ROOT_USERNAME: ADMIN
            MONGO_INITDB_ROOT_PASSWORD: ADMIN
```

docker-compose.yml

```
services:
 mongo_migration:
      container_name: mongo_migration
      depends_on:
        - db
      volumes:
       - "./csv_data:/data/csv"
      restart: on-failure
      networks:
        - frontend
      build:
     context: .
     dockerfile_inline: |
            FROM ubuntu:latest
            RUN apt-get update -y && apt-get install git python3-pymongo python3-pandas python3-dotenv -y
            RUN mkdir -p /var/opt
        command:
          - bash
          - -c
          - >
                  python3 /var/opt/csv_to_mongo/migration.py;
                  sleep infinity;
```

docker-compose.yml

```
services:
  mongo_migration:
        environment:
          CSV_DATASET_FILENAME: /data/csv/healthcare_dataset.csv
          DB_SERVER: mongodb://ADMIN:ADMIN@db:27017
          DB_NAME: health
          COLLECTION_NAME: patients
      INDEXES: "GENDER&AGE,BLOOD TYPE"
          DB_SCHEMA:
"{'Name': '', 'Age': '', 'Gender': '', 'Blood Type': '', 'Medical Condition': '', 'Insurance Provider': '', 'Hospital': 'Admission', 'Date of Admission':
    'Admission', 'Admission Type': 'Admission', 'Doctor': 'Admission', 'Room Number': 'Admission', 'Discharge Date': 'Admission', 'Medication':
    'Admission', 'Test Results': 'Admission', 'Billing Amount': 'Admission'} »
          DATE_PATTERN: "%Y-%m-%d"
          USER_ACCOUNTS: "[{'username': 'admin', 'password': 'admin', 'role': 'readWrite'},
        {'username': 'user', 'password': 'user', 'role': 'read'}]"
```

```
networks:
    frontend:
        driver: bridge
```