	Université de Corse - Pasquale PAOLI	
	***Diplôme :* M1 DE et DFS**	**2025-2026**
	Module : Gestion de Bases de données et dataWarehouses **TP2 – Base NoSQL MongoDB** **BD jeux de données scientifiques** **Enseignant : Evelyne VITTORI**	

Dans la première partie du projet (TP1), vous avez conçu une base relationnelle PostgreSQL décrivant l'écosystème de la recherche et stockant les métadonnées des jeux de données (titre, discipline, auteur, licence, date de dépôt, laboratoire, projet associé, etc.).

Ces métadonnées décrivent les jeux de données au niveau global (catalogue), mais elles ne contiennent pas les données effectives elles-mêmes.
Dans cette deuxième partie, il s'agit de créer une base NoSQL sous MongoDB pour stocker les jeux de données effectifs (et les métadonnées techniques spécifiques à ces données) associés aux projets de recherche.

Les jeux de données peuvent être très variés selon les disciplines :

- Sciences médicales : images de radiographie ou IRM (fichiers PNG ou DICOM), avec des métadonnées associées par fichier (dimensions, date de l'examen, identifiant pseudonymisé du patient)
- Chimie/Environnement : fichiers CSV de mesures expérimentales (pH, concentrations de substances), avec métadonnées comme la date et le lieu du prélèvement, la méthode d'analyse, les unités
- Sciences humaines et sociales : fichiers audio (MP3/WAV) d'entretiens ou transcriptions texte (TXT), avec métadonnées comme la durée, la langue, ou le statut d'anonymisation
- Géosciences : fichiers GeoTIFF ou shapefiles avec métadonnées spatiales (système de coordonnées, bounding box, résolution).

Il est essentiel de bien distinguer deux niveaux de métadonnées, qui ne sont pas de même nature :

Métadonnées de catalogue (stockées dans la BD relationnelle – TP1)
Elles décrivent le jeu de données dans son ensemble, de manière globale et normalisée.
Elles servent à identifier le dataset, à le relier à des projets, laboratoires ou contrats, et à gérer les conditions d'accès.
Exemples :

- Identifiant du dataset (ex. DS-2025-001).
- Titre et description.
- Discipline scientifique.
- Auteur / responsable.
- Projet et laboratoire associés.
- Date de création, date de dépôt.

- Licence, conditions d'accès.

Métadonnées techniques (stockées dans la BD NoSQL – TP2)
Elles décrivent les données effectives elles-mêmes : chaque fichier, chaque échantillon ou chaque enregistrement.
Elles varient selon le type de données et ne sont pas normalisées (d'où le recours à MongoDB).

Exemples :
- Pour une image IRM : dimensions (2048×2048), format (PNG ou DICOM), date de l'examen, identifiant pseudonymisé du patient.
- Pour un CSV de mesures : nombre de lignes, séparateur utilisé, liste des colonnes, unités de mesure.
- Pour un fichier audio : durée en secondes, fréquence d'échantillonnage, nombre de canaux.
- Pour un GeoTIFF : système de coordonnées (CRS), bounding box, résolution spatiale.
- Dans tous les cas : taille du fichier, empreinte (hash), version, logs d'acquisition, annotations éventuelles.

👉 Ces métadonnées sont de bas niveau technique et concernent directement le contenu effectif des datasets.

Objectifs Clés du TP
1. **Identifier les données effectives** à stocker pour les jeux de données : préciser les types de fichiers retenus et leurs métadonnées techniques associées (taille, dimensions, colonnes CSV, durée audio, etc.)
2. **Concevoir une base MongoDB** adaptée : proposer une ou plusieurs collections
3. **Générer des données fictives** réalistes : utiliser Python Faker et les identifiants de jeux de données issus de la BD relationnelle (TP1) pour peupler MongoDB
4. **Écrire des requêtes analytiques avec agrégations** : produire des requêtes exploitant les données effectives et leurs métadonnées techniques (statistiques par type de fichier, délais de traitement, taux de complétude…)
5. **Comparer et justifier la conception** : expliquer les choix de modélisation (collection unique, collections multiples, ou hybride) et discuter des avantages/inconvénients de MongoDB par rapport à PostgreSQL.

Rendus
Date rendu : la date de rendu sera fixée pendant les séances
Ce travail doit être réalisé en groupes de 2 à 3 étudiants. Le rendu sera effectué dans l'onglet travaux sur l'ENT sous la forme d'un **seul fichier archive (zip ou rar)** contenant l'ensemble des fichiers demandés :
- Rapport synthétique contenant :
  - Types de jeux de données et attributs techniques choisis
  - Schéma détaillé de la BD MongoDB (collections, structure des documents).
  - Justification des choix de modélisation
  - Réflexion comparative (MongoDB vs PostgreSQL)
  - Notice explicative des scripts
- Scripts MongoDB
  - scripts des requêtes d'insertions de données et des requêtes d'agrégation
- Scripts Python utilisés pour générer les données

Soutenance Orale

- Date prévue (indicative) : 3 décembre
- Présentation de 20 minutes
- Présentation des types de données retenus, du schéma MongoDB et des justifications.
- Démonstration des insertions et requêtes.
- Discussion comparative avec PostgreSQL

Critères d'évaluation

1. Pertinence des types de données et attributs choisis
2. Qualité de la conception (collections, flexibilité, évolutivité)
3. Réalisme et cohérence des données générées
4. Complexité et utilité des requêtes d'agrégation
5. Clarté de la comparaison MongoDB vs PostgreSQL

Partie 1 - Identification des données effectives

À partir des métadonnées définies en PostgreSQL, vous devez préciser ce qui doit être stocké dans MongoDB :

- Liste des types de fichiers (CSV, PNG, WAV, GeoTIFF, etc.).
- Métadonnées techniques spécifiques à chaque type (ex. csv.columns, image.width/height, audio.duration_sec, …)
- Éventuelles informations complémentaires : versions, annotations, …

Partie 2 - Conception des collections

Proposer au moins deux schémas possibles :

- Collection unique
- Plusieurs collections spécialisées

Choisissez une solution, justifiez votre choix et implémentez-la en MongoDB.

Partie 3 - Génération de données fictives

Les données réelles étant soumises à des restrictions de protection des données, vous allez générer des **données fictives** pour remplir votre base de données MongoDB.
Pour cela, vous utiliserez :

- **Les données** (identifiants des jeux de données) de la base PostgreSQL (créée lors de la partie 1 du projet TP1).
- **Python Faker** pour générer des données simulées (nom, taille, formats, métadonnées techniques).

Vous devrez créer suffisamment de données pour que les analyses soient pertinentes, en vous assurant que chaque jeu de données Mongo corresponde à un jeu de données décrit dans PostgreSQL.

Partie 4 - Requêtes avec agrégations

Une fois la base de données remplie, vous devrez créer et exécuter des **requêtes** en utilisant les fonctionnalités d'agrégation de MongoDB et PostgreSQL.
Vous devrez proposer au moins **4 requêtes** dans le style des requêtes suivantes :

- Volume total de données (octets) par type de fichier
- Nombre moyen de fichiers par dataset
- Délai moyen entre l' acquisition de la première donnée et la date de dépôt
- Taux de complétude des métadonnées techniques (ex. % d'images sans dimensions renseignées)

Partie 5 - Réflexion comparative

Pour conclure, vous rédigerez une réflexion pour expliquer :

- Pourquoi MongoDB est adapté pour ces données
- Quelles limites vous identifiez
- Comment PostgreSQL et MongoDB peuvent être complémentaires pour gérer ces données