**École Supérieure en Sciences et Technologies de l’Informatique**

**Et du Numérique (ESTIN)**

**Amizour, Béjaïa**

**2ème année cycle supérieure en Informatique**





| **Base de Données Distribuée :**  Projet TP |
| --- |

Réalisé par :

* Zerouk Mohamed Lamine

Groupe :

* 2CS2

Encadré par :

* Pr.K.SOUADIH

2023/2024

**Documentation technique de l'architecture de la base de données distribuée :**

**1. Introduction**

Le système de gestion des étudiants utilise une architecture de base de données distribuée qui combine MySQL et MongoDB. Cette approche hybride permet de tirer parti des forces de chaque système de base de données pour gérer efficacement différents types de données.

**2. Architecture de la base de données distribuée**

L'architecture de la base de données distribuée est composée de deux systèmes de gestion de bases de données distincts :

**2.1. MySQL**

MySQL est un système de gestion de base de données relationnelle (SGBDR) utilisé pour stocker les données structurées des étudiants. Il offre un schéma de données bien défini et des fonctionnalités robustes pour gérer les transactions, les contraintes d'intégrité et les requêtes complexes.

**2.2. MongoDB**

MongoDB est un système de gestion de base de données NoSQL orienté documents. Il est utilisé pour stocker les données non structurées, comme les photos de profil des étudiants. MongoDB offre une grande flexibilité pour stocker et interroger des données non structurées, ainsi qu'une mise à l'échelle horizontale efficace.

**3. Choix de conception**

Le choix d'utiliser MySQL et MongoDB dans cette architecture distribuée repose sur les critères suivants :

**3.1. Données structurées vs. non structurées**

Les données relatives aux informations des étudiants (nom, numéro d'inscription, cours, département, etc.) sont structurées et suivent un schéma bien défini. Ces données sont mieux gérées par un SGBDR comme MySQL. En revanche, les photos de profil sont des données non structurées et binaires, mieux adaptées à une base de données NoSQL comme MongoDB**.**

**3.2. Performances et évolutivité**

MySQL offre des performances optimales pour les opérations ACID (Atomicité, Cohérence, Isolement, Durabilité) et les requêtes complexes impliquant des jointures et des agrégations. MongoDB, quant à lui, est conçu pour une évolutivité horizontale efficace et des performances élevées pour les lectures et écritures de données non structurées.

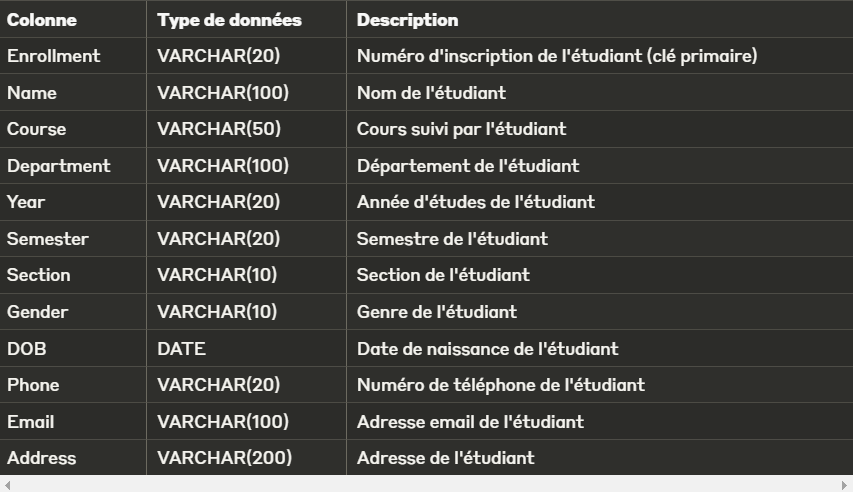
**3.3. Flexibilité du schéma**

Alors que MySQL nécessite un schéma de données rigide, MongoDB offre une flexibilité du schéma, ce qui facilite le stockage et la modification des données non structurées sans avoir à redéfinir le schéma.

**4. Schémas de données**

**4.1. Table MySQL student**

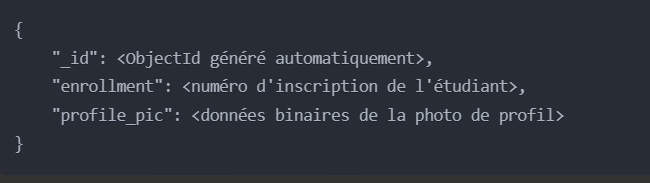
La table student dans MySQL stocke les informations structurées des étudiants. Voici son schéma :

****

**4.2. Collection MongoDB profile\_pics**

La collection profile\_pics dans MongoDB stocke les photos de profil des étudiants

sous forme de données binaires. Voici son schéma :

****

Dans ce schéma, \_id est un identifiant unique généré automatiquement par MongoDB pour chaque document. Le champ enrollment correspond au numéro d'inscription de l'étudiant, qui est utilisé pour établir une relation avec la table student dans MySQL. Le champ profile\_pic contient les données binaires de la photo de profil de l'étudiant.

**5. Intégration de MySQL et MongoDB**

L'intégration de MySQL et MongoDB est réalisée au niveau de l'application. Lorsqu'un nouvel étudiant est ajouté ou qu'un étudiant existant est mis à jour, les informations structurées sont enregistrées dans la table student deMySQL, tandis que la photo de profil est enregistrée dans la collection profile\_pics de MongoDB.

Lors de la récupération des données d'un étudiant, l'application interroge d'abord la table student dans MySQL pour récupérer les informations structurées. Ensuite, elle interroge la collection profile\_pics dans MongoDB pour récupérer la photo de profil correspondante en utilisant le numéro d'inscription comme clé de recherche.

Cette architecture distribuée offre les avantages suivants :

* **Gestion efficace des données structurées et non structurées :** MySQL gère efficacement les données structurées avec des opérations ACID et des requêtes complexes, tandis que MongoDB gère efficacement les données non structurées et binaires.
* **Évolutivité :** MongoDB offre une évolutivité horizontale efficace pour gérer d'importants volumes de données non structurées.
* **Flexibilité du schéma** : MongoDB permet de stocker et de modifier facilement les données non structurées sans avoir à redéfinir le schéma.

**6. Fonctionnalités clés**

**6.1. Interface Utilisateur Unique** Le système est doté d'une interface utilisateur graphique développée avec la bibliothèque Tkinter de Python. Cette interface permet aux utilisateurs de saisir facilement diverses informations sur les étudiants, telles que leur nom, leur numéro d'inscription, leur cours, leur département, etc. L'interface fournit également des fonctionnalités pour afficher, mettre à jour et supprimer les informations des étudiants.

**6.2. Répartition des Données** La répartition des données entre MySQL et MongoDB est effectuée en fonction de la nature des données :

* **Données structurées et interconnectées** : Les informations structurées des étudiants (nom, numéro d'inscription, cours, département, etc.) sont stockées dans la table student de la base de données MySQL.
* **Données semi-structurées ou non structurées** : Les photos de profil des étudiants, qui sont des données binaires non structurées, sont stockées dans la collection profile\_pics de la base de données MongoDB.

**6.3. Gestion des Transactions**

Pour assurer la cohérence des données entre MySQL et MongoDB, le système utilise une approche de transaction distribuée. Lorsqu'un nouvel étudiant est ajouté ou qu'un étudiant existant est mis à jour, une transaction est ouverte dans MySQL pour insérer ou mettre à jour les informations structurées. Si cette opération réussit, une opération d'insertion ou de mise à jour correspondante est effectuée dans MongoDB pour la photo de profil. En cas d'échec dans l'une des bases de données, la transaction est annulée dans les deux pour maintenir la cohérence des données.

**6.4. Sécurité**

Le système met en place diverses mesures de sécurité pour protéger les données sensibles stockées dans MySQL et MongoDB :

* **Authentification et contrôle d'accès** : Des mécanismes d'authentification et de contrôle d'accès sont mis en place pour limiter l'accès aux bases de données uniquement aux utilisateurs autorisés.
* **Journalisation des activités** : Toutes les activités critiques, comme les ajouts, les mises à jour et les suppressions de données, sont journalisées pour faciliter le suivi et l'audit.

**6.5. Migration des Données** Le système inclut un mécanisme de migration des données pour déplacer les informations des étudiants entre MySQL et MongoDB si nécessaire. Cette fonctionnalité peut être utile dans les cas suivants :

* **Changement de schéma de données** : Si le schéma de données dans MySQL doit être modifié, les données existantes peuvent être migrées vers MongoDB, modifiées, puis réintégrées dans MySQL avec le nouveau schéma.
* **Archivage des données** : Les données historiques des étudiants peuvent être déplacées de MySQL vers MongoDB à des fins d'archivage, libérant ainsi de l'espace dans MySQL pour de nouvelles données.

Lors de la migration des données, le système s'assure que l'intégrité et la cohérence des données sont maintenues. Des vérifications supplémentaires peuvent être effectuées pour détecter et résoudre tout conflit ou incohérence potentielle entre les deux bases de données.

**7. Technologies utilisées**

Le système de gestion des étudiants est développé en utilisant les technologies suivantes :

* **Python** : Le langage de programmation Python est utilisé pour développer l'application principale.
* **Tkinter** : La bibliothèque Tkinter de Python est utilisée pour créer l'interface utilisateur graphique.
* **MySQL** : MySQL est le système de gestion de base de données relationnelle utilisé pour stocker les données structurées des étudiants.
* **MongoDB** : MongoDB est le système de gestion de base de données NoSQL utilisé pour stocker les données non structurées, comme les photos de profil des étudiants.
* **mysql-connector-python** : Cette bibliothèque Python est utilisée pour interagir avec la base de données MySQL.
* **pymongo** : Cette bibliothèque Python est utilisée pour interagir avec la base de données MongoDB.
* **os** : Le module os de Python est utilisé pour interagir avec le système d'exploitation, notamment pour la gestion des chemins de fichiers et des variables d'environnement.