

1 INFORMATIONS GENERALES

Apprenti :	Nom:	Prénom :
Lieu de travail :	ETML / Vennes / 1004 LAUSANNE	
Client :	Nom:	Prénom: -
Dates de réalisation :	Semaines 15 à 22 Du 7.03.2024 au 26.05.2025	
Temps total	24 périodes	

2 PROCÉDURE

- Tous les apprentis réalisent le projet sur la base d'un cahier des charges.
- Le cahier des charges est présenté, commenté et discuté.
- Les apprentis sont entièrement responsables de la sécurité de leurs données.
- En cas de problèmes graves, les apprentis avertissent leur chef de projet au plus vite.
- Les apprentis ont la possibilité d'obtenir de l'aide externe, mais ils doivent le mentionner.
- Ce projet est individuel.
- Un mini-rapport est demandé à chaque apprenti de manière individuelle
- En cas d'imprévu, le temps total doit être géré par l'apprenti.

3 SUJET

Implémenter et utiliser une base de données MongoDB

4 MATÉRIEL ET LOGICIEL À DISPOSITION

- Un PC ETML
- Environnement docker avec container MongoDB (serveur et mongosh)
- Environnement docker avec container Redis-Stack (serveur et redis-cli)
- Interface de gestion MongoDB compass + VS Code avec l'extension MongoDB
- Application Node.js / Vue3.js : todo-app-mysql
- Accès à Internet

5 PRÉREQUIS

Module I165

6 LIVRABLES

- **Première partie** : une « release » GitHub de l'application **todo-app** dûment modifiée.
- **Deuxième partie** : Un rapport de projet au format PDF avec une mise en page soignée comprenant les commandes et les explications correspondantes. Nom du fichier : **X-165-P-DB-NoSQL-<NomPrénom>.pdf**
- Un journal de travail (selon model fournit par votre enseignant)
- Date de remise des livrables : jour de la dernière séance, 23h59 sur le « Devoir Teams ».

7 CAHIER DES CHARGES :

7.1 Première partie :

Une application de gestion de tâches « todo-app » avec un backend Node.JS / Express offre une API pour le frontend développé avec Vue3.js. L'application utilise actuellement une base de données MySQL.

- <https://github.com/matinfo/todo-app-mysql/archive/refs/tags/v1.0.1.zip>

Le client désire faire une migration de l'application afin d'utiliser à la place MongoDB.

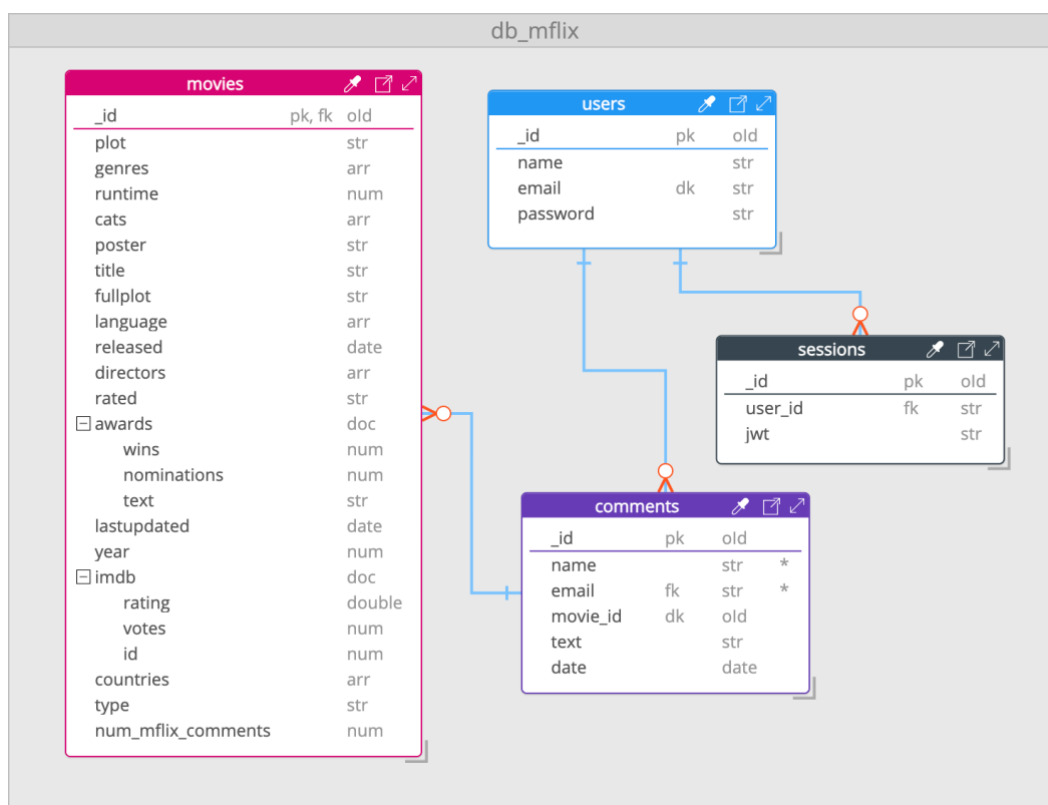
Vous avez pour mission de modifier l'application pour :

- Utiliser MongoDB à la place de MySQL
- Implémenter correctement la recherche de Todo avec MongoDB
- Ajouter du cache applicatif Redis pour optimiser les appels API

Note : vous devez mettre votre code de l'application sur GitHub dans un repo privé partagé avec votre enseignant. Attention à appliquer des commit fréquent et avec des commentaires structurés.

7.2 Deuxième partie

Vous avez à disposition un fichier ayant pour nom **db_mflix.gz**, contenant les collections de la base de données « db_mflix ». Ce fichier est un dump fait avec **mongodump** de type *archive* et *compressé*, dont voici le MPD :



Ce MPD exprime les collections ainsi que les liaisons entre les collections avec leurs cardinalités selon la notation dite « **Crow's Foot** », [patte d'oie].

A- Importer les données et le schéma de la base de données

Restaurez les collections dans une base de données nommée « db_mflix » dans une instance de MongoDB.

Travail demandé :

- Donner et expliquer en détail la commande utilisée.

B- Gestion des utilisateurs

Trois types d'utilisateurs interagiront avec la base de données « db_mflix ». Pour ce faire, chaque type d'utilisateur a des besoins d'accès et de manipulation de données différents :

1. Administrateur :
 - 1.1. Peut créer ou supprimer des collections ;
 - 1.2. Peut créer, lire, mettre à jour et supprimer (CRUD) n'importe quels documents des collections ;
 - 1.3. Gérer les indexes pour toutes les collections ;
 - 1.4. Gérer les rôles (et donc les utilisateurs) et leurs privilèges de cette base de données.
2. Utilisateur :
 - 2.1. Lire les informations sur les films ;
 - 2.2. Lire, ajouter ou supprimer un ou des commentaires.
3. Gestionnaire :
 - 3.1. Lire les informations sur tous les utilisateurs (pour savoir qui a fait un commentaire) ;
 - 3.2. Mettre à jour, lire et supprimer des films ou des commentaires ;

Travail demandé :

- Donnez les commandes effectuées pour créer les rôles selon les informations ci-dessus.
- Donnez les commandes effectuées pour créer un utilisateur pour chacun des rôles.

C- Backup/Restore

Nous souhaitons réaliser une sauvegarde complète de notre base de données « db_mflix », de façon qu'elle prenne le moins de place possible. Ensuite, nous aimerions procéder à sa restauration.

Travail demandé :

- Donnez et expliquez en détail chaque commandes effectuées.

8 EVALUATION :

1. Une auto-évaluation challengée par le client basé sur des éléments observables inscrits dans la planification initiale à environ 80% du temps imparti pour le projet sera effectuée.
2. L'évaluation comprendra une revue de code challengé par votre enseignant.
3. Une évaluation finale sera effectuée par le client après la livraison final du projet.
4. Le recours à des outils en ligne d'intelligence artificielle (ex. : Chat GPT) doit être mentionné et ne peut servir que d'inspiration à la réalisation. En cas d'abus, l'évaluation du projet en tiendra compte.