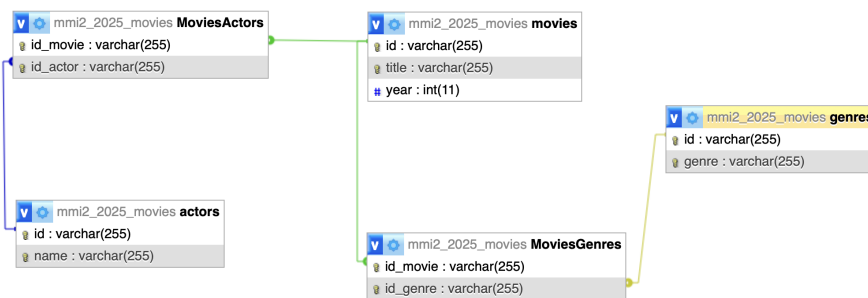


SAE4D05 – S4

1 Objectifs

Vous allez travailler avec une base de données relatives à des films américains sortis en salle entre 1980 et 2023.

Les données vous sont fournies sous la forme d'une base de données SQL correspondant au schéma suivant :



Vous devrez créer une API écrite en NodeJS, documentée et testée afin de pouvoir accéder à ces données. Ensuite vous devrez créer un site web permettant à un utilisateur de découvrir ces données de manière ludique.

1.1 API pour interagir avec les données

Votre serveur NodeJS devra permettre l'accès aux différentes tables de la base de données avec des requêtes de type GET.

Vous fournirez également une documentation SWAGGER explicite de votre API.

Vous mettrez en œuvre des tests de toutes les routes proposées par votre API.

1.2 Client Web pour découvrir et manipuler les données

Votre client web devra prendre la forme d'un site web optimisé pour ordinateur.

L'objectif est de proposer une découverte ludique des données aux utilisateurs. Ainsi, lorsqu'un utilisateur utilisera votre site, il pourra réaliser les actions suivantes :

- Obtenir quelques statistiques sur les données : nombre de films, nombre de films / année, nombre d'acteurs, nombre moyen d'acteurs / film. . .
- Démarrer une nouvelle partie d'un jeu qui consistera à explorer les données de manière ludique.
 - Au démarrage d'une nouvelle partie, votre site web devra proposer de manière aléatoire un film ou un acteur.
 - Si un film a été proposé, son titre sera visible et l'ensemble des acteurs ayant joué dans le film devront être découverts par l'utilisateur.
 - Les données seront affichées sous la forme d'un graphe interactif (vous pourrez utiliser le package D3¹) et plus spécifiquement, la visualisation de graphes².
 - L'utilisateur devra alors trouver quels acteurs jouent dans le film. Pour chaque acteur découvert, le graphe sera mis à jour et proposera les films (à découvrir) dans lesquels ce nouvel acteur joue.

¹<https://d3js.org>

²<https://observablehq.com/@d3/force-directed-tree>

- Interaction avec l'utilisateur : vous pourrez utiliser tout type de formulaire pour interagir avec l'utilisateur, par exemple le package SweetAlert³.
- Sauvegarde locale sur le navigateur : un utilisateur pourra, s'il le souhaite, recharger la dernière partie effectuée. Celle-ci sera sauvegardée en local sur le navigateur (`localStorage`) et pourra être chargée par défaut.
- Vous êtes fortement encouragés à créer un projet github dédié pour cette SAÉ.
- Extensions possibles :
 - Utilisation de l'API wikipedia pour nodeJS⁴ vous permettra de récupérer la photo des acteurs, au fur et à mesure, ainsi que l'affiche des films (lorsque tous les acteurs d'un film auront été trouvés)
 - Vous pouvez mettre en place un système de points, obtenus lors de la découverte de films, d'acteurs ou de l'intégralité du casting d'un film, ou de l'intégralité de la filmographie d'un acteur. Ces points pourraient être utilisés pour déclencher des bonus ou obtenir des indices sur les informations à trouver
 - Vous pouvez imaginer un mode simplifié ou une partie des informations à trouver sont révélées (lettres composant le titre du film ou le nom de l'acteur...)

2 Évaluation

Votre travail sera évalué selon 4 aspects :

- La documentation de votre API
- Les tests réalisés pour tester votre API
- Les fonctionnalités proposées au niveau du jeu
- Une soutenance orale de 15 minutes pour vérifier votre niveau de maîtrise du code

Vous pouvez utiliser des outils d'assistance à la création de code, mais vous devez impérativement avoir la maîtrise du code écrit.

Concernant les fonctionnalités proposées dans votre projet, elles seront évaluées de la manière suivante :

- Originalité : ce critère sera mesuré de manière objective. Une fonctionnalité présente dans de nombreux projets aura moins d'originalité (et donc de points associés) qu'une fonctionnalité présente uniquement dans votre projet. Le nombre de points accordé à l'originalité sera donc inversement proportionnel au nombre de fois où la fonctionnalité est présente dans tous les projets rendus. Les fonctionnalités de l'API, qui sont indispensables au bon fonctionnement du projet, et notamment celles permettant l'interaction avec la base de données, auront donc un poids faible car tous les projets vont devoir les proposer.
- Complexité de mise en œuvre de la fonctionnalité. Une fonctionnalité simple, par exemple une page about rapportera moins de points qu'une fonctionnalité plus complexe.
- Valeur ajoutée en terme de jouabilité. Par exemple, un habillage graphique soigné mais n'apportant rien au jeu rapportera peu de points.

Vous devrez compléter le fichier présent sur Moodle pour lister et décrire toutes vos fonctionnalités.

3 Réalisation de la SAÉ

Vous pourrez réaliser la SAÉ seul ou en binôme. Compte tenu de l'organisation des soutenances orales le vendredi 4 avril après-midi, il ne pourra pas y avoir plus de 12 groupes (4 soutenances / heure).

³<https://sweetalert2.github.io/>

⁴<https://www.npmjs.com/package/wikipedia>