Travail pratique #1 : Laravel API

Pondération : 28 %

Travail devant être réalisé **individuellement**

Remise : Au plus tard **dimanche le 7 mars** avant minuit

1 Objectif

Développer une API REST sécuritaire, incluant des tests et de la documentation, en utilisant Laravel.

2 Conditions de réalisation du travail

Votre remise consiste à livrer toutes les routes accompagnées de tests et de documentation.

Vous devez utiliser les données fournies en SQL comme données de départ de votre application.

Le modèle de la base de données est disponible en annexe. Le nom de votre BD doit respecter le format **TP_Laravel_VotreNom.**

De plus, vous devrez travailler en utilisant **GitLab** en continu et de façon adéquate.

Éléments importants de l'API REST :

- Contrôleurs
- Modèles et entité(s)
    - Relations appropriées
- Ressource (DTO)
- Validation (*Request*) pour les actions *store* et *update*
- Structure et données de tests (migrations et *seeds*)
- Documentation de l'API avec Swagger
- Tests de la base de données
- Tests suivants pour toutes les actions :

- Les requêtes valides (200 et 201)
- Les requêtes avec données invalides (422)

3 API – Routes à développer

- **Opérations CRUD** pour un film :
  - **Consultation** des films (sans critiques et sans acteurs)
  - **Consultation** d'un certain film avec ces critiques
- **Consultation** de tous les **acteurs** d'un certain **film**
- **Recherche** de films
  - Selon les critères suivants :
    - Mot-clé (dans *title* et *description*)
    - Classification (*rating*)
    - Durée minimale
    - Durée maximale
  - Une **limite de 20 films** doit être retourné par requête (les 20 premiers correspondant aux critères)
    - Il est possible de préciser la **page** pour accéder à 20 autres films
    - Tous les **critères** sont **optionnels** et si aucun n'est fourni, on doit simplement retourner les 20 premiers films
- **Opérations CRUD** pour un *user* :
  - **Ajout** d'un nouveau *user*

4 Critères d'évaluation

Éléments	Pondération
• **Routes d'API demandées** o Planification des routes, requêtes, validations et réponses	40%
• **Qualité du code** o Bonne utilisation des patrons de conception ▪ MVC, DTO, Entité, Modèles, Relations, Request, etc. o Bien découpé et indenté o Code clair et simple o Principes SOLID o Respecte la structure de Laravel et le standard des routes enseigné o Documentation avec *Swagger*	20%
• **Qualité des tests** o Données de tests suffisantes (*seeds*) o Tests suffisants o Teste la fiabilité et la robustesse o Tests pour les codes suivants : 200, 201, 204, 422, 405	30%
• **Utilisation adéquate de GitLab** o Branches par fonctionnalité, participation régulière, etc.	10%

Pour chaque faute de français, 0,5% sera soustrait de la note finale (pour un maximum de 20%).

5 Documents à réaliser et à remettre

Veuillez remettre sur « LÉA » une archive « zip » contenant tous les fichiers du projet :

- Lien GitLab avec accès pour l'utilisateur **filiat00**
- Code (sans les dépendances – **sans le dossier *vendor***)

Bon succès!

Annexe – Base de données