

Développement d'application Django

Antonin MOREL

Avril 2024

Abstract

Le sujet de cette évaluation est de livrer une application Python / Django répondant au cahier des charges suivant. Le but de l'application est de permettre à un utilisateur de rechercher un objet dans une base de données et d'afficher le(s) résultat(s). Si aucun résultat n'est connu, l'application va alors chercher les résultats dans une API tierce.

1 Énoncé

Pour cela nous allons utiliser l'API suivante : <http://www.omdbapi.com/> . Les comportements attendus sont les suivants:

- Au lancement de l'application, un formulaire est présenté à l'utilisateur, celui-ci peut alors entrer un nom de film (exemple : star wars) dans une barre de recherche et valider le formulaire.
- L'application doit répondre avec les résultats correspondants trouvés dans la base de données
- Si aucun résultat n'a été trouvé dans la base de donnée l'application va alors rechercher les résultats dans l'API et les afficher à l'utilisateur. Ces résultats sont enregistrés en base pour pouvoir être réutilisés si besoin.

Ainsi une première recherche avec le mot clé "star wars" utilisera l'API, enregistrera et affichera les résultats. Une deuxième recherche avec le même mot clé répondra directement avec les résultats contenus dans la base de données. Il faut afficher au minimum:

- Le titre du film
- L'année
- Les genres

2 Rendu et notation

Un projet git contenant tous les fichiers nécessaires au bon fonctionnement de l'application doit être envoyé avant le XX. Vous devrez:

- Bien intégrer tous les fichiers nécessaires au fonctionnement de l'application
- Utiliser des comptes git séparés afin que les commits soient identifiables (pensez à utiliser git tout au long de la vie du projet)
- Rédiger des messages de commit clair
- Écrire un texte de 200 mots minimum **personnel** décrivant ce que vous avez fait dans ce projet, et ce que vous en avez appris. Les textes seront inclus dans le repo git.

La grille de notation suivante sera utilisée:

- *Compréhension et respect du sujet - 8 points* : Pensez à bien relire le sujet et à implémenter tous les comportements demandés.
- *Qualité du code - 7 points* : Le code respecte les standards évoqué en cours, le code est clair et compréhensible. Il pourrait être maintenu facilement par un(e) camarade de classe ou un(e) futur collègue. Pensez à linter votre code.
- *Tests - 3 points*: Le projet présente une suite de tests en accord avec les fonctionnalités demandées.
- *Documentation - 2 points* : Le projet comporte toutes les instructions qui permettent d'installer et de lancer le projet.

Points bonus (note maximum de 20):

- Un effort de design et d'intégration a été fait - 1.5 point
- L'application est hébergée et disponible en ligne (merci de fournir l'URL) : 1.5 points
- L'historique git est clair et les commit bien écrit : 1 point