Médiathèque

Ce projet est développé avec le framework Django, et est composé de deux applications. L’application membre permet de visualiser la liste des médias. L’application bibliothécaire, permet la gestion des membres et des médias (création, modification, suppression) et la gestion d’emprunt.

Étude et correctifs du code fourni.

Avec le code fournis, on remarque la création de différentes classes qui vont nous servir de bases pour la création de model pour notre base de données. Les classes livre(), dvd(), er cd() possèdent plusieurs champs en commun, ce qui nous permet de créer un class parent nommé média() qui regroupe ses différents champs. Les class livre(), dvd() et cd() hériteront de la class média().

Mise en place des fonctionnalités :

Pour accéder aux différentes applications, j’ai opté pour une landing page Médiathèque.

Pour l’application membre, l’affichage des différents médias s’effectue sous forme de listes.

Pour l’application bibliothécaire, accessible qu’aux bibliothécaires, j’ai opté d’afficher les différentes listes (cd, dvd, livre, membre, emprunt…) sous formes de tableaux sur la page d’accueil. Chaque fonctionnalité d’ajout, de modification, suppression est séparés et renvoie vers un formulaire différent.

Stratégie de test :

Pour réaliser les tests j’utilise la librairie Pytest.

On retrouve :

1. des tests unitaires qui vérifie l’ajout la modification la suppression d’un média (livre).
2. Des tests intégrations, comme « test\_emprunt\_media » qui simule un emprunt de livre, vérifie les changements en base et affiche le template correspondant.
3. Test de sécurité, avec « test\_connexion\_superuser\_success » qui vérifie l’identifiant et redirige vers la bonne vue. Et « test\_connexion\_superuser\_invalid\_user » qui vérifie l’affichage d’un message d’erreur pour un mauvais identifiant.
4. Des test de gestion d’erreur notamment la limite d’emprunt dépassé ou la tentative d’emprunt avec un emprunt en retard.

Instruction pour lancer le projet :

**1. Cloner le projet**

Dans le terminal d'un nouveau dossier entrer la commande :  
git clone https://github.com/julien696/mediatheque.git

cd mediatheque

**2. Créer un environnement virtuel**

python -m venv env Puis activer.  
Sous Windows :  
env\Scripts\activate  
Sous Mac/Linux :  
source env/bin/activate

**3. Installer les dépendances**

pip install -r requirements.txt

**4. Appliquer les migrations**

python manage.py migrate

**5.Charger les fixtures**

python manage.py loaddata medias.json

**6. Créer un super utilisateur (Accès blibliothécaire)**

python manage.py createsuperuser

**7. Lancer le serveur**

python manage.py runserver

**8. Lancer les test**

python manage.py test