



Mini-Projet : Cinema

Opened: Thursday, 9 January 2025, 1:30 PM

Due: Thursday, 23 January 2025, 11:59 PM

Mini-Projet : Cinéma

Projet à réaliser seul ou en binôme.

Un cinéma d'un petit village isolé souhaite proposer des films à l'affiche pour attirer plus de jeunes des villages alentours. Pour atteindre cet objectif, les gérants du cinéma décident de vous missionner pour créer une application permettant aux clients de visualiser les films disponibles.

Données :

Les données des films sont disponibles depuis la base de données de [OMDb](#) (The Open Movie Database).

Vous allez avoir besoin de [demander une API Key](#) que vous recevrez par email et qui sera à utiliser dans l'URL pour les requêtes.

Les requêtes auront cette forme : [https://www.omdbapi.com/?apikey=\[votre API Key\]&s=transformers](https://www.omdbapi.com/?apikey=[votre API Key]&s=transformers) (celle-ci fait une recherche pour avoir le premier film qui correspond au titre renseigné).

La base de données est en anglais, il est conseillé de renseigner les noms originaux des films pour faire une recherche. Par exemple : "Guardians of The Galaxy".

Voici le cahier des charges fourni :

Livrables :

Un site web, contenant au moins les 3 pages suivantes :

- index.html : Page d'accueil du site internet montrant les films tendances,
- search.html : Page permettant de rechercher des films,
- movie.html : Page permettant d'afficher les détails d'un film.

Attention :

Il est nécessaire d'utiliser un serveur http pour accéder à vos pages. Vous pourrez utiliser :

- Live Server sur VS CODE
- serve (un paquet NPM) - pour celui-ci, vous pouvez reprendre le script "npm start" des exercices avec batteries de tests

Il est fortement recommandé d'utiliser plusieurs fichiers JS (ou TS) et de les faire traiter par un bundler.

Il est fortement recommandé d'avoir un bundle (ou au moins un fichier JS) par page au lieu d'un seul pour toutes les pages (sauf logique partagé entre pages, comme l'authentification par exemple)

Critères de notation :

index.html (25pts)

- Afficher quelques films (3 minimum) contenus par la base de données que vous considérez en tendance (5 pts)
- Pour chaque film, les informations suivantes doivent être affichées : (5 pts)
 - Son poster
 - Son nom
 - Un lien pour en savoir plus (devra rediriger vers movie.html)
 - Bonus : Un **petit** résumé du film
- Ajouter un bouton pour charger plus de films de 2024

search.html (25pts)

La page devra contenir :

- Une barre de recherche
- Les résultats de la recherche en temps réel (20 pts)
- Interdiction de causer un rechargement de la page pendant ou après la recherche (0 pts sur l'ensemble de la partie si ce n'est pas respecté)
- Un bouton pour charger plus de résultats de recherche

Pour chaque résultat de la recherche affichée : (5 pts)

- Le poster du film
- Le titre du film
- Un lien pour en savoir plus (devra rediriger vers movie.html)

movie.html (25pts)

La page devra contenir :

- Le titre du film
- Le poster du film
- Le **grand** résumé du film
- Le genre du film
- Les acteurs
- Bonus :
 - Les notes obtenues par le film
 - La date de sortie en DVD (formatée en français : jj/mm/aaaa)

Git (25 pts)

- La bonne gestion du dépôt distant sur GitHub (10 pts)
- L'organisation des branches (10 pts)
- Le nom des messages de commit (5 pts)

Le rendu se fera dans une archive ZIP (ou autre) depuis Moodle, seul le rendu sur Moodle sera prit en compte.
Pour évaluer la partie Git, il faudrait que vous renseigniez le lien vers le dépôt GitHub dans un README.md (ou autre).

Grading summary

Visible groups Cohorte Paris B.Eng.1 2024-2025


Hidden from students	Yes
Participants	34
Submitted	0
Needs grading	0
Time remaining	14 days 10 hours

Contact us



Follow us



 Contact site support

You are logged in as [Gabriel Pichot](#) ([Log out](#))

[Reset user tour on this page](#)