

CINEZONE — PLATEFORME DE GESTION DE CATALOGUE DE FILMS

Institution : Mediaschool Nice — IRIS

1. CONTEXTE ET OBJECTIF

Dans le cadre du BTS SIO option SISR, ce projet vise à concevoir et développer une plateforme web de consultation et de gestion de catalogue de films. L'objectif est de fournir aux utilisateurs finaux une interface moderne permettant de rechercher des films, d'afficher leurs informations détaillées et de naviguer dans un catalogue dynamique alimenté par une API REST externe.

Le projet a été réalisé en binôme et couvre les aspects suivants :

- Analyse du besoin et conception de la solution
- Intégration Front-End (HTML, CSS, JavaScript)
- Consommation d'une API REST (TMDB — The Movie Database)
- Déploiement sur un serveur web Apache
- Sécurisation des accès

2. ARCHITECTURE GLOBALE

L'application repose sur une architecture client-serveur simple. Le Front-End est hébergé sur un serveur Apache et communique avec l'API TMDB pour récupérer les données des films en temps réel.

Composant Rôle

Navigateur web Interface utilisateur — rendu HTML/CSS/JS

Serveur Apache Serveur web hébergeant les fichiers statiques

API TMDB Source de données externe — films, affiches, synopsis

Pare-feu Contrôle des accès entrants/sortants vers le serveur

Flux de données :

Navigateur → Serveur Apache (fichiers HTML/CSS/JS) → API TMDB (requêtes HTTP REST) → Données JSON retournées → Affichage dynamique

3. TECHNOLOGIES UTILISEES

Technologie Rôle

HTML5 Structure des pages web

CSS3 Mise en forme et design de l'interface

JavaScript Logique applicative côté client et appels API

API REST TMDB Source de données cinématographiques

Apache HTTP Serveur web d'hébergement

Pare-feu Sécurisation des accès réseau au serveur

4. DESCRIPTION DES COMPOSANTS

4.1 Interface utilisateur (Front-End)

L'interface est développée intégralement en HTML, CSS et JavaScript sans framework. Elle propose :

- Un catalogue de films affiché dynamiquement
- Une fonctionnalité de recherche avancée par titre ou genre
- L'affichage des informations détaillées d'un film sélectionné (synopsis, note, affiche, casting)
- Une présentation responsive adaptée à différentes tailles d'écran

4.2 Intégration API TMDB

L'application consomme l'API REST de TMDB (The Movie Database) via des appels JavaScript asynchrones (fetch). Les données retournées au format JSON sont parsées et injectées dynamiquement dans le DOM.

Endpoints utilisés :

- Récupération des films populaires
- Recherche de films par titre
- Récupération des détails d'un film

4.3 Serveur web Apache

Le serveur Apache héberge les fichiers statiques de l'application (HTML, CSS, JS, images). La configuration du pare-feu permet de restreindre les accès au port 80 (HTTP) uniquement.

5. FONCTIONNEMENT TECHNIQUE

5.1 Chargement de la page

Au chargement de la page principale, un appel JavaScript vers l'API TMDB est déclenché pour récupérer les films populaires du moment. Les résultats sont affichés dynamiquement dans le catalogue sans rechargement de page.

5.2 Recherche de films

La barre de recherche déclenche un appel à l'endpoint de recherche de l'API TMDB avec le terme saisi par l'utilisateur. Les résultats sont filtrés et affichés en temps réel.

5.3 Affichage des détails

Au clic sur un film, un appel supplémentaire à l'API récupère les informations détaillées du film sélectionné. Ces données sont affichées dans une vue dédiée ou une fenêtre modale.

6. SECURITE

- La clé d'API TMDB est intégrée dans le code côté client JavaScript. Ce point constitue une limitation de sécurité inhérente à l'architecture Front-End uniquement.
- Le serveur Apache est configuré avec un pare-feu limitant les accès aux ports nécessaires
- Les accès au serveur sont restreints aux protocoles HTTP/HTTPS

Note : une évolution vers une architecture avec un back-end dédié permettrait de sécuriser la clé API côté serveur et de ne pas l'exposer dans le code source client.

7. DEPLOIEMENT ET CONFIGURATION

Pré-requis :

- Serveur Apache installé sur la machine d'hébergement
- Clé d'API TMDB valide (obtenue après inscription sur themoviedb.org)
- Règles de pare-feu configurées pour autoriser le port 80

Procédure de déploiement :

1. Installer le serveur web Apache

2. Configurer les règles de pare-feu pour autoriser le trafic HTTP

3. Déposer les fichiers de l'application (HTML, CSS, JS) dans le répertoire web d'Apache (/var/www/html/ ou équivalent)

4. Renseigner la clé API TMDB dans le fichier JavaScript de configuration

5. Vérifier l'accessibilité de l'application via le navigateur

L'application est accessible à l'adresse : https://walo2428.github.io/Cine_pro/index.html

8. TESTS ET VALIDATION

Tests réalisés :

- Chargement du catalogue de films depuis l'API TMDB : succès
- Recherche d'un film par titre avec affichage des résultats : succès
- Affichage des informations détaillées d'un film sélectionné : succès
- Vérification de la réactivité de l'interface sur différentes tailles d'écran : succès
- Accessibilité du serveur Apache après configuration du pare-feu : succès

Durée de réalisation : 40 jours