

# RAPPORT pour le projet de développement web

Movies & Series



rédigé par

LELIEVRE Saidou - DA CRUZ Mathis



Licence d'Informatique deuxième année

## Table des matières

Remerciements.....	3
Introduction.....	4
1.1 Objectif du projet.....	4
1.2 Contexte du projet.....	4
1.3 Gestion du projet.....	5
2 Conception et réalisation du projet.....	6
2.1 Les Choix Techniques.....	6
2.2 Contrainte du projet.....	7
2.2 Amélioration du projet.....	8
Conclusion.....	9

## Remerciements

Tout d'abord, nous tenons à remercier sincèrement M. Lemaire, qui, en tant que professeur de notre module de développement web, s'est toujours montré à l'écoute et disponible pour notre groupe tout au long de ce semestre. En effet grâce à sa dévotion en tant que superviseur, il nous a fournis beaucoup d'aide durant nos cours de TD, particulièrement utile en ce moment de confinement. Nous remercions également toute l'équipe des Professeurs de l'université de Cergy pour nous avoir suivie et soutenue jusqu'ici.

## Introduction

### 1.1 Objectif du projet

Le but du projet est de faire un site web qui met en œuvre l'ensemble des éléments techniques de l'UE « Développement Web » : *HTML 5 / CSS 3 / PHP 7* dans le cadre d'une réalisation en binôme. Le site offrira une interface de recherche de film ou série en fonction de critères à définir. L'internaute pourra sélectionner des options, choisir dans une liste, avoir une barre de recherche...etc. L'affichage permettra d'obtenir le maximum d'informations sur le film ou la série recherchée. Les échanges se feront impérativement entre votre serveur Web et l'API (éventuellement plusieurs à agréger, cf. Extensions possibles). Les réponses utiliseront le format XML et / ou JSON afin de nous permettre de maîtriser les 2 principaux flux d'échanges des API disponibles dans le contexte du Web.

### 1.2 Contexte du projet

Tout d'abord, On doit choisir une ou plusieurs API (*Application Programming Interface*) afin de pouvoir récupérer les informations des films et séries sur le format d'échange JSON. Nous devons récupérer le plus d'informations possible comme par exemple « Titre, Genre, Année de parution ». Afin de réaliser un stockage côté serveur, sur fichier, des films consultés par l'ensemble des utilisateurs du site. Ensuite une barre de recherche doit être disponible pour faciliter l'utilisation de l'internaute. Et aussi le site doit contenir une partie « statistiques » permettant de visualiser sous forme graphique l'utilisation du site : notamment le nombre total de visiteurs ainsi que les films ou séries les plus fréquemment consultés. Le site obtenu doit être esthétique et ergonomique « indentation, rangement des fichiers dans des dossiers...etc ».

Finalement Afin d'améliorer l'ergonomie et l'expérience utilisateur , ajouter le stockage du dernier film consulté coté client via un mécanisme de cookie (*film ou série consultée, date de la consultation, etc.*) .

### 1.3 Gestion du projet

Tableau 1: Répartitions des taches

Répartition des Taches	
<b>Lelievre Saidou</b>	<ul style="list-style-type: none"> <li>- Développement HTML/CSS (Mise en page général du site)</li> <li>- Développement PHP (Récupération des données de IMDB)</li> <li>- Développement PHP (Fichier CSV / Film – Série – Anime Aléatoire)</li> <li>- Développement PHP (TMDB API récupération des films Populaire et des Upcoming movies)</li> </ul>
<b>Da Cruz Mathis</b>	<ul style="list-style-type: none"> <li>- Développement PHP (intégration des données dans une barre de recherche Globale)</li> <li>- Développement PHP (intégration des données dans une barre de recherche par série et récupération des données d'épisode)</li> <li>- Développement PHP (Mise en page des panorama et Album)</li> <li>- Développement PHP (Diagramme de Statistique et Sauvegarde d'un WatchList)</li> </ul>

Tableau 2: Planning

PLANNING	
<b>1 mars – 7 mars</b>	Réflexion sur le contexte du projet /mise en binôme/ <b>BrainStorming</b>
<b>7 mars – 28 mars</b>	Début du projet/ Choix d'une <i>API</i> et récupération <b>des données</b> des films par ID
<b>29 mars – 4 Avril</b>	Intégration d'une <b>barre de recherche</b> par <i>nom et date</i> / Films et série Aléatoire grâce au fichier CSV
<b>5 avril – 18 avril</b>	Travail en Autonomie / Chacun travail pour l'amélioration du site et pour récupérer plus de données / données des <i>épisode et barre de recherche</i> <b>Série</b>
<b>19 avril – 2 mai</b>	Statistique et <b>WatchList</b> /Récupération des données de TMDB api / <b>Popular Films et Upcoming Films</b>
<b>2 mai – 9 mai</b>	<i>Rapport du projet</i> et des petites <i>amélioration</i> du site

## 2 Conception et réalisation du projet

### 2.1 Les Choix Techniques

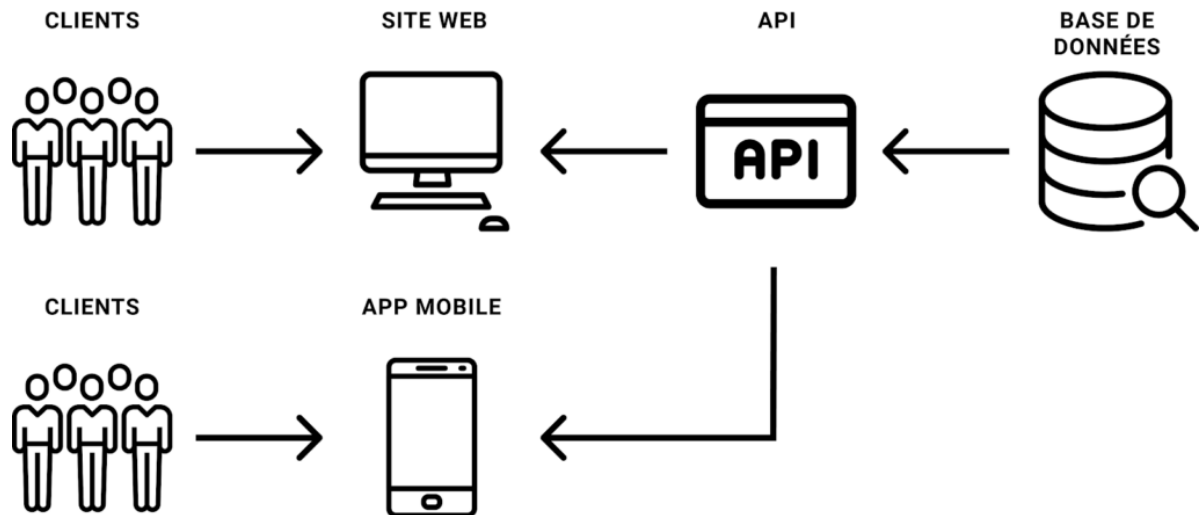


Figure 1: Fonctionne de l'API

Tout d'abord nous avons choisis l'API (*Application Programming Interface*) *OMDB*, un service Web gratuit pour obtenir des informations de « Film,Série ». Tout le contenu et les images du site sont apportés et entretenus par les utilisateurs. L'utilisation est très simple, il suffit de faire une demande de requêtes par « Titre ou ID » et nous avons une réponse au format JSON que nous décodons avec la fonction « **Json\_decode** » (php 7).

Nous avons d'abord créer une « **API\_key** » pour nous authentifier en tant qu'utilisateur, ensuite nous avons choisis de récupérer les données par « **ID** » (*numérotation qui sert d'identifiant pour chaque film ou série*). En effet *OMDB* est une API avec plusieurs paramètres nous permettant de préciser « l'URL » de notre page sous la forme d'une chaîne de caractère.

#### EXEMPLE d'URL

<http://lelievre-cyu.alwaysdata.net/projet/action.php?i=tt4154796>

↑  
**Protocole**

↑  
**Domaine**

↑  
**Paramètre par ID**

Figure 2: URL avec paramètre ID

Avec les fonctions « **json\_decode** » et « **file\_get\_contents** » nous avons modifié les paramètres de l'URL. Par *exemple* avec « **t** » pour Title et « **i** » pour ID.. Nous avons décodé le

contenue au format JSON. Exemple: « <http://lelievre-cyu.alwaysdata.net/projet/action.php?i=tt0988824> » nous affiche la page de l'animé « Naruto : Shippuden » grâce a son ID.

Tableau 3: Les paramètres D'OMDB

OMDB API param	omdb.py Param	Value	Description
<b>s</b>	search	string (optional)	titre du média rechercher
<b>i</b>	imdb_id	string (optional)	un IMDb ID valide
<b>t</b>	title	string (optional)	Renvoie le titre du media
<b>y</b>	year	year (optional)	Année du media
<b>page</b>	page	page (optional)	Renvoie une page
<b>Season</b>	season	season (optional)	Nombre de saison
<b>Episode</b>	episode	episode (optional)	Nombre d'épisode
<b>type</b>	media_type	string (optional)	Renvoie le type du media (Films, épisode, ou series)

Ensuite nous avons fait la barre de recherche classique d'OMDB qui figure sur le site de l'API avec comme paramètre « *Titre et Année* ». Cette barre affiche avec précision l'œuvre paru à tels date par exemple si on écrit « harry potter, 2001 » il nous affiche « Harry Potter à l'école des sorcier ». Après nous avons une barre de recherche plus Globale avec comme paramètre « *titre du média rechercher et le type* » qui nous affiche toute les media du liée au Titre.

Et pour finir la dernière barre de recherche par Série avec comme paramètre « *le type série, la saison, les épisodes* » elle nous affiche que des séries avec toute les données comme par exemple « *le nombre de saison, le titre des épisodes...etc* ». Toutes les fonctions figure dans notre fichier « fonctions.inc ».

The image shows a web interface with three distinct search sections, each with a title and a search bar:

- Rechercher des films, séries, animés...**: Includes a text input field with placeholder text "Recherche par Titre", a small input field for "Année", and a "search" button.
- Recherche globale**: Includes a text input field with placeholder text "Recherche par Titre", a "Type" selector with radio buttons for "Movie" (selected) and "Série", and a "search" button.
- Recherche série**: Includes a text input field with placeholder text "Recherche par Titre" and a "search" button.

Figure 3: Barre de recherche du Site

Nous avons en second plan fait le choix de mettre dans chaque onglet, un affichage automatique d'une œuvre au aléatoire pour que l'internaute puisse découvrir de lui même des œuvres qu'il ne connaît pas de toute les époques allant presque de « 1980 à 2021 ». Nous récupérons aléatoirement l'*ID* d'un « films, Séries, Animes » que nous avons stockés dans des fichiers CSV «movies.csv, series.csv, Animes.csv » et à chaque fois qu'on rafraîchis la page, une nouvelle œuvre qui figure dans ce fichier est générée.

Pour finir nous avons fait un diagramme statistique des 10 films les plus populaire du site tout simplement grâce à un algorithme de compteur qui compte chaque visite sur le site dans un fichier «*compteur.txt* » et ensuite on récupère l'*ID* et de chaque film dans un fichier « *stats.csv* » pour vérifier si le « films, Série » rentre dans le classement.

## 2.2 Contrainte du projet

Ce projet était finalement aisé en niveau de difficulté car il est relativement intéressant grâce au contenu du sujet. En effet le fait de travailler sur un site qui traite de Films et Série nous a tout de suite donné plus de motivation. De plus en plus on avançait dans nos recherches de plus en plus nous étions fiers de nos découvertes et de notre contenu. Mais nous avons quand même eu quelque difficulté car au début du projet, nous avions une avancée plutôt lente. Mais après avoir mis en accord notre site avec le thème en fixant notre « notre barre de recherche » nous avons commencé à être productif dans ce travail. Malgré cela nous avons quand même des regrets car nous avons entamés trop tardivement l'utilisation d'une seconde API « *TMDB* » qui nous a paru beaucoup plus intéressante autant dans son contenu que dans sa mise en forme.

Nous sommes insatisfait aussi du fait de ne pas avoir intégralement mis les cookies sur notre site car nous avons mis d'autres objectifs en priorité. Nous pensons aussi que la mise en forme de notre site aurait pu avoir des améliorations pour que l'internaute ait une utilisation plus optimale.



## 2.2 Amélioration du projet



Figure 4: Plan du Site

Notre site contient une amélioration inédite c'est la WatchList, cette onglet stocke un ou plusieurs films de l'internaute en cliquant sur le bouton « plus » sans qu'il ai besoin de se connecter à un portail. Dans l'aspect technique on récupère l'adresse\_ip de l'internaute et ensuite on crée un fichier « .txt » du même nom et donc a chaque fois que l'internaute clique sur le bouton vert, il ajoute l'ID du « film, série ou animés » ensuite à chaque fois qu'il revient sur l'onglet WatchList il y'a une lecture du fichier texte avec le nom de l'adresse IP. De même qu'il peut supprimer les media déjà présent dans sa liste. Cette onglet est très utile pour ceux et celle qui veulent absolument garder en mémoire les films qu'ils ont déjà vue ou qu'ils veulent voir.

## Conclusion

En conclusion nous tenons à dire que nous avons que ce projet est sûrement le plus intéressant de toute l'année, il nous a permis de travailler sur un sujet qui avons beaucoup apprécié car tout les jours, nous allons navigué sur des sites de streaming qui nous apporte le même genre de contenu. En effet cela nous a beaucoup motivé et cela nous a enfin permis de nous améliorer sur le plan technique et l'aspect visuel du développement web. Même nous cela nous a été utile car nous avons fait des découvertes pendant notre exploration. Finalement cela a rendu le travail en binôme lucratif et cela facilité l'échange, nous sommes très content du rendu.