

Rapport PC3R

Guillaume RIDEY Laetitia PHAM Maxence BRUNET

Mars 2020

1 Description de notre application

Nous avons décidé de créer une application de gestion de sortie de films et de séries. Cette application doit ressembler à un calendrier des sorties avec en plus comme fonctionnalités: la recherche de films ou séries selon différents critères, l'ajout dans les favoris, la gestion de celles-ci et la possibilité de noter un film ou une série. De plus, l'utilisateur aura accès à un système d'inscription, de connexion et de déconnexion.

2 API choisi

Nous avons choisi l'API "*the movie DB*" car cette API dispose d'une base de films, séries ainsi qu'un catalogue des acteurs. De plus, l'API est régulièrement mise à jours avec les nouvelles sorties, elle possède aussi un système de vote pour les films et séries. Ainsi notre application utilisera cette API comme base de données, les requêtes pour la recherche de films ou de séries seront directement effectués sur cette API.

3 UseCase

Les Use-case sont :

- *s'inscrire*:

Bob va sur le site. Il clique sur *connexion*. Il arrive sur la page de connexion, il clique sur *créer un compte*. Il arrive sur la page d'inscription, il rentre un nom de compte, un mot de passe et une confirmation de son mot de passe puis valide en appuyant sur *validation*. Il est renvoyé sur la page de profil.

- *connexion*:

Bob va sur le site. Il clique sur *connexion*. Il arrive sur la page de connexion, il rentre son nom de compte et son mot de passe puis clique sur *validé*, il est renvoyé sur la page de profil.

- *déconnexion*:

Bob est connecté sur le site et est sur la page d'accueil. Il clique sur *déconnexion*, il est renvoyé sur la page d'accueil.

- *ajouter un favori*:

Bob est connecté sur le site et est sur la page d'un film ou d'une série. Il clique sur *ajouter aux favoris*, il est renvoyé sur la page de description mais le bouton *ajouter aux favoris* est vert.

- *supprimer un favori*:

Bob est connecté sur le site et est sur sa page de profil. Il clique sur *Favoris*. Il est envoyé sur sa page des Favoris. Il clique sur la croix à côté du favori qu'il veut supprimer. Il est renvoyé sur sa page des Favoris sans le favori qu'il vient de supprimer.

- *recherche de films ou de séries*:

Bob est sur la page d'accueil. Il tape le titre "the blind side" dans la barre de recherche puis valide. Il reçoit une liste de film correspondant à sa recherche, il clique sur celui qui l'intéresse. Il est renvoyé sur la page de description du film qu'il a choisi.

- *ajouter un commentaire*:

Bob est connecté sur le site et est sur la page d'un film ou d'une série. Il tape le commentaire "Super film" dans la barre de commentaire puis valide. Il est renvoyé sur la page du film mais son commentaire est maintenant dans la liste des commentaires sous le synopsis.

Les fonctionnalités de l'application sont:

- *s'inscrire*: l'utilisateur pourra s'inscrire en choisissant son nom d'utilisateur dans l'application et son mot de passe.

- *se connecter*: l'utilisateur pourra se connecter en rentrant son nom d'utilisateur et son mot de passe dans l'application.

- *se déconnecter*: si l'utilisateur est connecté, il pourra se déconnecter.

- *recupérer le calendrier*: l'utilisateur (connecté ou non) pourra avoir accès au calendrier des sorties des films et séries.

- *ajouter un favori*: si l'utilisateur est connecté, il pourra aller sur la page d'un film ou d'une série l'ajouter à sa liste de favoris.

- *supprimer un favori*: si l'utilisateur est connecté, il pourra aller sur la page de ses favoris pour en supprimer.

- *recherche de films ou de séries*: l'utilisateur (connecté ou non) aura accès à la barre de recherche des films et séries selon les mots-clefs ou les critères sélectionnés (genre, acteur, catégorie, date de sortie, popularité,...).

4 Base de données du serveur

Dans la base de données du serveur, nous stockerons les informations sur les comptes des utilisateurs c'est-à-dire le nom de l'utilisateur, son mot de passe et sa liste de favoris, en plus des commentaires de chaque films ou séries.

4.1 Schémas de tables:

Les tables sont les suivantes:

- Utilisateurs (Apache)
id_user: identifiant d'un utilisateur. password : le mot de passe associé à un utilisateur.

clef de session : clef unique permettant de savoir si l'utilisateur est connecté ou pas (attribuer lors de la connexion et supprimer lors de la déconnexion).

- Favoris (MongoDB)
id_user: identifiant d'un utilisateur. liste_favoris: liste des favoris d'un utilisateur. (films/séries)

- Commentaires_Films (MongoDB)
-titre: Le titre d'un film -commentaire: Commentaires associé au film ou la série. Un commentaire sera de la forme (id, com)

- Commentaires_Series (MongoDB)
-titre: Le titre d'une serie -timeline: Les commentaires associé à une série selon la saison et l'épisode.Cela se présentera sous la forme de {saison: {num, (id,com)}}, episode {num, (id,com)} }

Nous aurons quatre tables de données, une avec les données de l'utilisateur, une autre avec sa liste de favoris et enfin une table avec les commentaires de chaque films et séries.

4.2 Mise a jour et appelle de l'API

Nous effectuerons des mises à jour dans les bases de données lorsque:

- Un nouvelle utilisateur s'inscrive (insert dans la table *utilisateur*).

- Un nouvelle utilisateur se connecte ou se déconnecte (insert de la clé de session dans la table *utilisateur*).
 - Un utilisateur ajoute un favoris (insert dans la table *favoris*).
 - Un utilisateur supprime un favoris (delete dans la table *favoris*).
 - On affiche la liste des favoris d'un utilisateur (recherche dans la table *favoris*).
 - Un utilisateur ajoute un commentaire (insert dans la table *Commentaire_Film* pour un film ou *Commentaire_Serie* pour une série).
 - Un utilisateur supprime un commentaire (delete dans la table *Commentaire_Film* pour un film ou *Commentaire_Serie* pour une série).
 - On affiche la liste des commentaires d'un utilisateur (recherche dans la table *Commentaire_Film* pour un film ou *Commentaire_Serie* pour une série).
- Pour ce qui en est des recherches de films et de séries, du calendrier des sortis ou des descriptions, nous appellerons l'API externe.

5 Description du Serveur

On utilisera des Services qui seront appelés par nos différents servlets.

Les Composants (Servlet) sont:

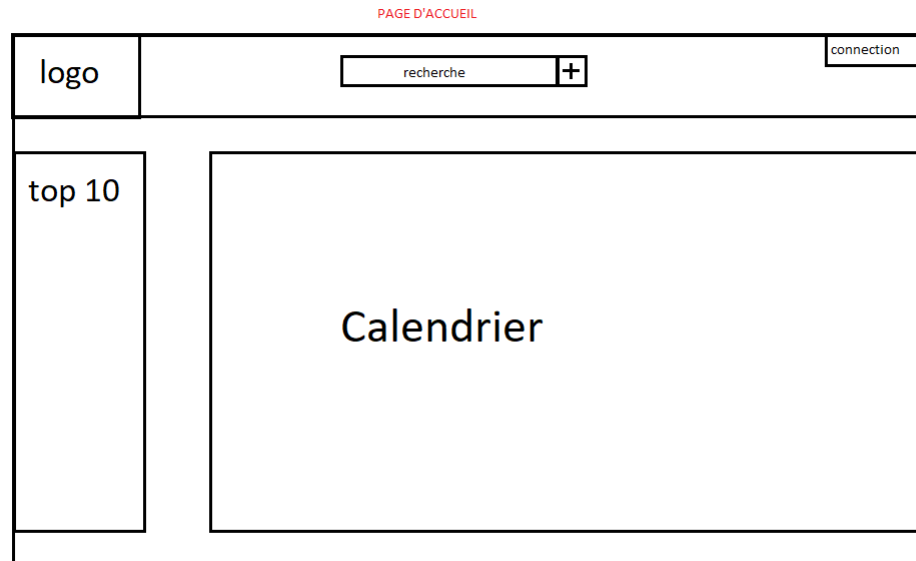
- LOGIN (connexion):
 - Accède à la base de données des utilisateurs.
- LOGOUT (déconnexion):
 - Accède à la base de données des utilisateurs.
- REGISTER (inscription):
 - Accède à la base de données des utilisateurs.
- ADDFAVORIS (ajout Favoris)
 - Requête paramétré par une chaîne de caractère (film ou une série).
 - Met à jour la liste des favoris d'un utilisateur par un ajout.
 - Réponse de contenue vide.
- DELETEFAVORIS (supprimer un commentaire)
 - Requête paramétré par une chaîne de caractère (film ou une série).
 - Met à jour la liste des favoris d'un utilisateur par une suppression.
 - Réponse de contenue vide .
- LISTFAVORIS (Liste des favoris)
 - Requête paramétré par une chaîne de caractère correspondant à un utilisateur.
 - Consulte la base de données (requête pour récupérer les objets correspondant à la chaîne).
 - Il fabrique une réponse correspondant contenant la liste des favoris d'un utilisateur.
- ADDCOMMENT (ajout commentaire)
 - Requête paramétré par une chaîne de caractère et un identifiant d'un utilisateur.
 - Met à jour la liste des commentaires d'un film (ou d'une série) par un ajout.

- Réponse de contenu vide.
- DELETECOMMENT (supprimer Commentaire)
 - Requête paramétré par une chaîne de caractère et un identifiant d'un utilisateur.
 - Met à jour la liste des commentaires d'un film (ou d'une série) par une suppression.
 - Réponse de contenu vide.
- LISTCOMMENT (Listes des commentaires)
 - Requête paramétré par une chaîne de caractère correspondant à une série ou un film.
 - Consulte la base de données (requête pour récupérer les objets correspondant à la chaîne).
 - Il fabrique une réponse correspondant contenant la liste de commentaire associé au film (ou la série).

6 Description du Client

Le contenu des écrans est le suivant:

- *La page d'accueil*: le bouton connexion permettra d'accéder à la page d'authentification. La barre de recherche fera appel à l'API externe lorsque des mots-clefs auront été entré. Le logo permet d'accéder à la page d'accueil. Le "top 10" et le calendrier des sorties fera appel à l'API externe et affichera le résultat obtenue.



- *La page de profil*: le bouton de déconnexion fera appel au composant "Logout" du serveur. La barre de recherche fera appel à l'API externe lorsque des

mots-clés auront été entré. La liste des favoris fera appel au composant "ListFavoris" dont nous passerons en requête l'identifiant de l'utilisateur. Finalement, le calendrier personnel fera appel à l'API externe.

page Profil

logo	<input style="width: 90%;" type="text" value="recherche"/> <input style="width: 10%;" type="button" value="+"/>	deconnection
favori	<h2 style="margin: 0;">Calendrier perso</h2>	

- *La page de connexions*: Le bouton "validé" réalisera la connexion qui fera appel au composant "Login". Le bouton "crée un compte" permettra d'accéder à la page d'inscription. Le logo permet d'accéder à la page d'accueil.

connection

logo	
<div style="border: 1px solid black; width: 60%; margin: 0 auto; padding: 10px;"> <input style="width: 90%;" type="text" value="nom utilisateur"/> <input style="width: 90%;" type="text" value="mots de passe"/> <input style="width: 20%;" type="button" value="validé"/> <input style="width: 40%;" type="button" value="cree un compte"/> </div>	

- *La page d'inscription*: Le bouton "validation" réalisera l'inscription qui fera appel au composant "Register". Le logo permet d'accéder à la page d'accueil.

inscription

logo	<div style="border: 1px solid black; padding: 10px; margin: 0 auto; width: 60%;"> <input style="width: 90%;" type="text" value="nom utilisateur"/> <input style="width: 90%;" type="text" value="mots de passe"/> <input style="width: 90%;" type="text" value="confirmation mot de passe"/> <input style="width: 30%; margin-top: 10px;" type="button" value="validation"/> </div>
------	--

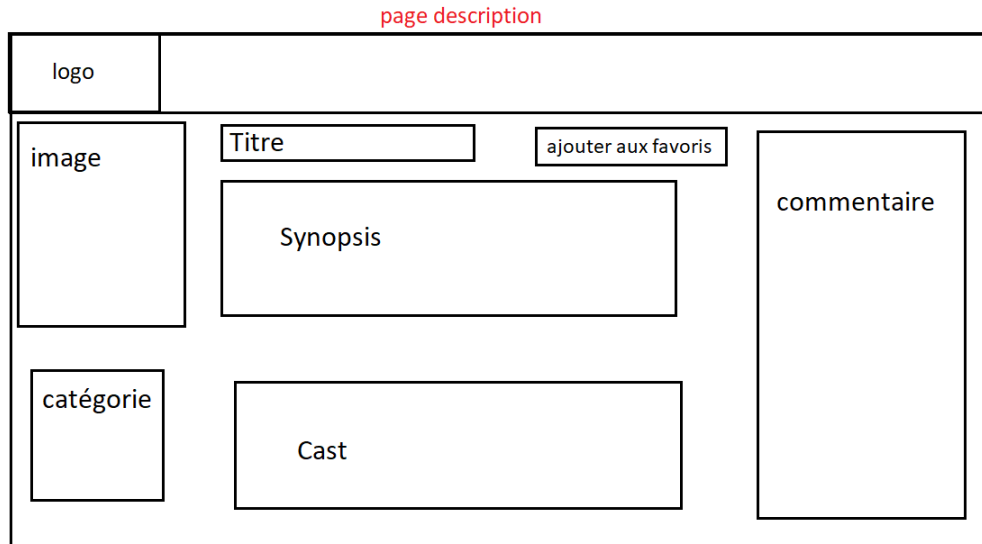
- *La page de recherche avancée:* Le bouton "recherche" permettra d'effectuer la recherche avancé et fera appel à l'API externe. Les boutons "action" "aventure" ... permettent de spécifier les informations de la recherche.

recherche avancer

logo	<div style="display: flex; justify-content: space-between;"> <div style="width: 45%;"> <input style="width: 95%;" type="text" value="nom du film"/> <input style="width: 95%;" type="text" value="acteur"/> <input style="width: 95%;" type="text" value="date"/> </div> <div style="width: 45%; border: 1px solid black; padding: 5px;"> <table style="width: 100%; border-collapse: collapse;"> <tr> <td style="padding: 2px 5px;">action</td> <td style="width: 40px; height: 20px; border: 1px solid black;"></td> <td style="width: 40px; height: 20px; border: 1px solid black;"></td> </tr> <tr> <td style="padding: 2px 5px;">aventure</td> <td style="width: 40px; height: 20px; border: 1px solid black;"></td> <td style="width: 40px; height: 20px; border: 1px solid black;"></td> </tr> <tr> <td style="padding: 2px 5px;">etc</td> <td style="width: 40px; height: 20px; border: 1px solid black;"></td> <td style="width: 40px; height: 20px; border: 1px solid black;"></td> </tr> </table> </div> </div> <div style="text-align: center; margin-top: 20px;"> <input style="width: 100px;" type="button" value="rechercher"/> </div>	action			aventure			etc		
action										
aventure										
etc										

- *La page de description:* L'image, la synopsis, le "casting" et la catégorie auront été récupéré grâce à l'API externe, tandis que les commentaires auront été récupéré via la composant "ListComment". La rédaction d'un commentaire est disponible si seulement l'utilisateur est connecté et cela appellera le composant "AddComment" (ou "DeleteComment" l'utilisateur veut supprimer son commentaire). De plus, le bouton "ajout aux favoris" fera appel au composant

”AddFavori”.



- *La page des favoris*: la récupération des favoris c’est faite avec le composant ”ListFavoris”. De plus, les boutons de suppression de favoris feront appel au composant ”DeleteFavori”.

7 Descriptions des requêtes

Les principales requêtes utilisées sur notre applications :

- **DEMANDE DE CONNEXION** : Lorsqu’un utilisateur souhaite se connecter, nous effectuons directement une requête en mode *POST* pour permettre la protection du mot de passe.

- **AJOUT DES FAVORIS** : Lorsqu’un utilisateur souhaite ajouter un film/série en favoris à la base de donnée, nous effectuons une requête en mode *GET*. On reçoit une confirmation sous la forme d’un *JSON* pour confirmer si la requête à réussi (ou échoué).

- **AJOUT DES UTILISATEURS** : Lors de l’inscription d’un utilisateur, on effectue une requête en mode *GET* pour ajouter l’utilisateur à la base de donnée. Le serveur renvoie un *JSON* pour confirmer que la requête à réussi (ou échoué).

- **LISTER LES FAVORIS D’UN UTILISATEUR** : Lorsque l’on souhaite voir les films et séries d’un utilisateur, on envoie une requête en mode *GET*. On récupère via la requête un *JSON* qui contient la liste des favoris lié à un utilisateur.

- AJOUT DE COMMENTAIRES : Lorsqu'un utilisateur souhaite commenter un film ou une série, on effectue une requête en mode *GET* pour stocker ce commentaire dans une base de donnée

Les différentes requêtes *JSON* auront cette structure (Liste non exhaustive):

- POUR LA CONNEXION : { "connexion" : { "id": "David", "mdp" : "Good-enough" } }

- POUR LES FAVORIS : { "favoris" : { "id": "David", "favoris" : ["Avengers", "Scooby-Do"] } }

- POUR LES COMMENTAIRES DE FILMS: { "AllMovies" : { "titre": "Jurassic Park", "Synopsis": "Film de dinosaures super cool", "catégorie": ["Action","Aventure"], "commentaires" : ["Trop bien !", "Je préfère Dinotopia lol"], "cast": [{ "Nom":Neill "Prenom": Sam},...] } }

- POUR LES COMMENTAIRES DE SERIES: { "AllSeries" : { "titre": "How I Met Your Mother", "synopsis": "C'est l'histoire de Ted Mosby qui raconte à ses enfants comment il a rencontré leur mère" "saison": 4, "episode": 1, "commentaires" : ["Great Speech at 19:30", "Nice episode"] } }

- POUR LES CONFIRMATION OU ERREUR LIÉ À LA REQUÊTE :

{"Ok"} ou {"Error" : 404}

8 Schéma global du système

