

Margaux BRULIARD - Nahida BENHAFFAF

DODOCINÉ

PROJET PROGRAMMATION WEB



20 Mai 2021

TABLE DES MATIÈRES

Table des matières	1
Introduction	2
Présentation du projet	2
1 Architecture du site web	4
1.1 Mise en place de la base de données	4
2 Accueil et Navigation	5
2.1 Les composants de la page d'accueil	5
2.2 La barre de navigation	6
2.2.1 mise en place d'une recherche auto-complétée	6
3 Accès utilisateurs	7
3.1 Inscription ou Connexion sur DodoCiné	7
3.2 L'espace membre associé à un compte	9
3.2.1 Rappels des notes et des messages postés	10
3.2.2 Paramètres du compte	10
3.2.3 Favoris	11
4 Présentation des films	12
4.1 Présentation des films disponibles sur le site	12
4.2 Page de présentation d'un film	13
4.2.1 Construction du système de notation des films	13
4.2.2 Gestion du forum relatif au film	14

INTRODUCTION

Présentation du projet

De nos jours, l'arrivée en masse des plateformes de "streaming" sur Internet a permis de populariser l'accès aux films de tous types et de toutes générations. Afin d'aiguiller et d'aider les cinéphiles en herbes que nous sommes, nombre de sites nous aident aujourd'hui à découvrir des films qui correspondent à nos préférences. Parmi ces sites, nous retrouvons notamment le site Allociné qui présente des films ainsi que les acteurs et les réalisateurs y ayant participé.

Dans le cadre du module **Programmation Web** de la licence d'Informatique de l'Université Sorbonne Paris Nord, nous nous sommes penchés sur le fonctionnement de ce type de site web et avons choisi de construire un "mini-clone" du site d'Allociné, que nous appellerons en cette période de pandémie **DodoCiné** afin de parfaire nos connaissances sur la programmation web avec les langages de programmation PHP¹ et Ajax côté serveur et les langages HTML5/CSS3/Javascript pour le côté client.

Mise en production

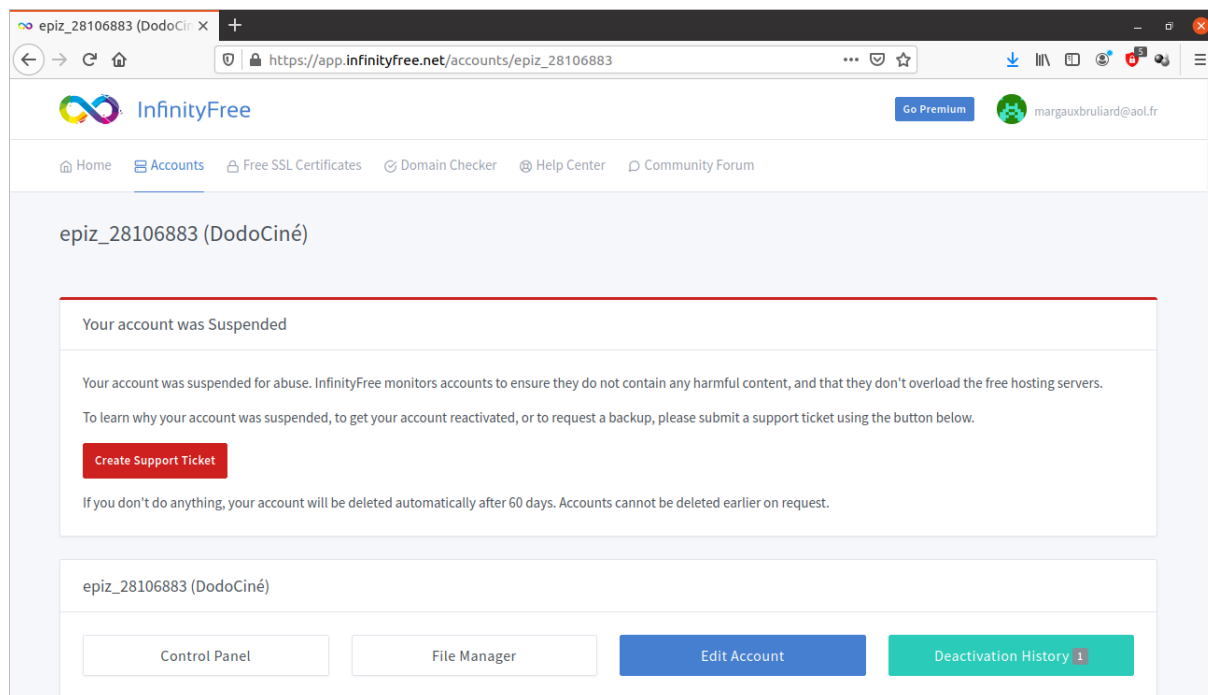
*Il est possible de visualiser notre site **DodoCiné** directement en cliquant sur ce lien : <http://dodocine.infinityfreeapp.com/>.*

Cependant, notre hébergement étant gratuit, il est à noter que la protection de la base de données n'est pas assurée. En conséquence, nous ne vous conseillons pas d'utiliser votre véritable adresse email si vous souhaitez créer un compte

Téléchargement de l'archive

Vous pouvez retrouver l'ensemble de nos fichiers sources dans notre repository Git : <https://github.com/MBruliard/dodocine> ou simplement télécharger l'archive via le lien suivant https://drive.google.com/file/d/1pcoIUpSAnw7-lmUuXKf4hSuKBiMXp_nk/view?usp=sharing

1. nous utiliserons ici la version de PHP7



Malheureusement, en vérifiant que l'ensemble du site fonctionnait, j'ai eu la mauvaise surprise de découvrir que mon compte avait été suspendu... Donc mon bel hébergement a été bloqué juste avant de rendre notre projet. Je suis navrée pour le désagrément. Nous avons fait une demande pour que le site soit de nouveau actif mais je ne sais pas si cela sera réglé au moment de votre visite.

1. ARCHITECTURE DU SITE WEB

Pour ce projet, nous avons essayé de tirer parti du dynamisme de PHP en créant de multiples fichiers annexes appelés à de multiples reprises dans différentes pages. Ces fichiers sont regroupés dans le dossier *elements*. On y retrouve notamment des parties de codes réutilisées pour chaque page de notre site

- *elements/head.php* contient la balise 'head' que nous rappelons pour chaque page de notre site. Nous lui avons ajouté deux variables, *\$title_page* qui permet de changer le titre de la page au besoin ainsi que *\$css_addon*, qui prend une chaîne de caractères pour ajouter des liens vers des fichiers css supplémentaires.
- *elements/footer.php* contient le footer, répété dans chacune de nos pages.
- *elements/js_files.php* appelle l'ensemble des scripts Javascript/JQuery dont nous avons besoin. De la même façon que pour *elements/head.php*, nous avons construit une variable *\$js_addon* pour ajouter des scripts supplémentaires dans certaines pages.
- *elements/navigation.php* la barre de navigation

1.1 Mise en place de la base de données

Nous avons fait le choix de créer des scripts PHP contenant l'ensemble des fonctions qui interagiront avec la base de données via le système PDO de PHP. Ces scripts sont regroupés dans le dossier *controller* :

- *controller/authentication.php* pour les fonctions relatives aux utilisateurs
- *controller/datafilms.php* pour les fonctions relatives aux films et à la notation de ceux-ci
- *controller/dataindividus.php* pour les fonctions permettant d'obtenir des informations sur les acteurs/réalisateurs contenus de la base de données.
- *controller/dataforum.php* pour la gestion du forum.

Pour ce projet, nous avons fait le choix d'utiliser le système de gestion de base de données nommé SQLite et non pas MySQL, principalement afin de pouvoir avoir un hébergement gratuit de notre site pour sa présentation. Cependant, si SQLite est adaptée pour un projet universitaire, il engendre des ralentissements de navigation lorsque la taille de la base de données ou le trafic de visites dépassent un certain seuil.

On remarque, que dans l'écriture de requêtes SQL avec PDO, SQLite ferme automatiquement la connexion à la base de données contrairement à MySQL.

2. ACCUEIL ET NAVIGATION

Commençons par vous présenter la page centrale de notre site : la page d'accueil, que les visiteurs voient en premier c'est pourquoi nous l'avons construite de manière épurée et fonctionnelle.

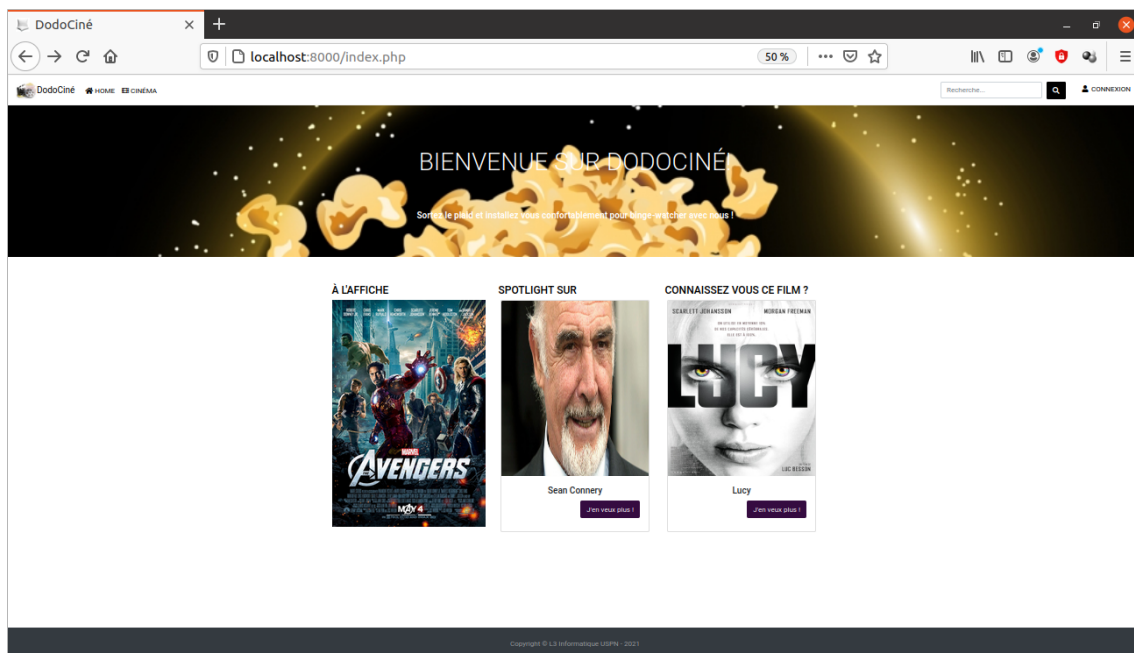


FIGURE 2.1 – Page d'accueil de notre site

2.1 Les composants de la page d'accueil

On souhaite que, dès son arrivée, le visiteur puisse avoir un aperçu des derniers films présentés sur le site. Pour ce faire, nous utilisons un "carousel", c'est-à-dire une bannière animée, pour les mettre en avant.

Afin de simplifier son inclusion et son interaction avec l'ensemble de la page, nous avons fait le choix d'utiliser l'élément "carousel" intégré dans le framework Bootstrap, et qui gère de façon automatique les actions dynamiques autour du "carousel".

En plus de cela, nous mettons en avant un acteur/réalisateur ainsi qu'un film apparaissant de manière aléatoire avec la possibilité de visiter leur fiche pour plus d'informations.

2.2 La barre de navigation

Nous avons décomposé la barre de navigation en 2 parties : sur la gauche, le visiteur peut naviguer entre la page d'accueil et la liste de films disponibles sur le site (via l'onglet "Cinéma"). Sur la droite, le visiteur peut se connecter, ou s'inscrire (en créant son compte).



FIGURE 2.2 – notre barre de navigation vu par un simple visiteur

Si le visiteur s'est connecté, alors la barre de navigation change : à la place du bouton de connexion, il voit apparaître son pseudonyme et peut accéder à son espace personnel, ou se déconnecter.



FIGURE 2.3 – notre barre de navigation lorsque le visiteur est connecté

2.2.1 mise en place d'une recherche auto-complétée

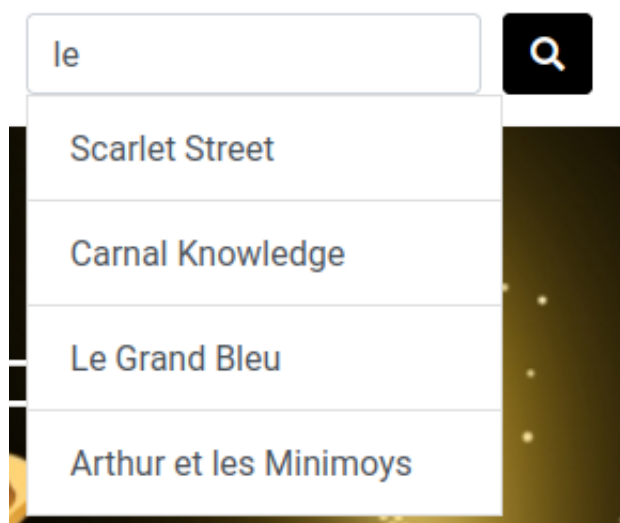


FIGURE 2.4 – Recherche auto-complétée

Il est possible à tout le moment d'effectuer une recherche à propos d'un film ou d'un acteur/réalisateur en tapant son nom dans la barre de recherche du menu de navigation. Au fur et à mesure des entrées utilisateurs, nous proposons à l'utilisateur des noms de films ou d'acteurs. En cliquant sur l'un d'eux, il accède directement à la fiche du film/de l'acteur. À l'inverse, si le visiteur ne trouve pas dans les propositions ce qui lui convient, il peut simplement cliquer sur le bouton de recherche qui le renverra sur une nouvelle page contenant le résultat de sa recherche.

3. ACCÈS UTILISATEURS

Afin de filtrer et maîtriser les informations circulant sur le site, ainsi que pour fidéliser les visiteurs les plus assidus, nous construisons un espace membre permettant aux visiteurs de créer et/ou de se connecter pour avoir accès à des fonctionnalités supplémentaires du site.

3.1 Inscription ou Connexion sur DodoCiné

Les visiteurs du site peuvent se connecter à tout moment, en utilisant le bouton **Connexion** situé à droite dans la barre de navigation.

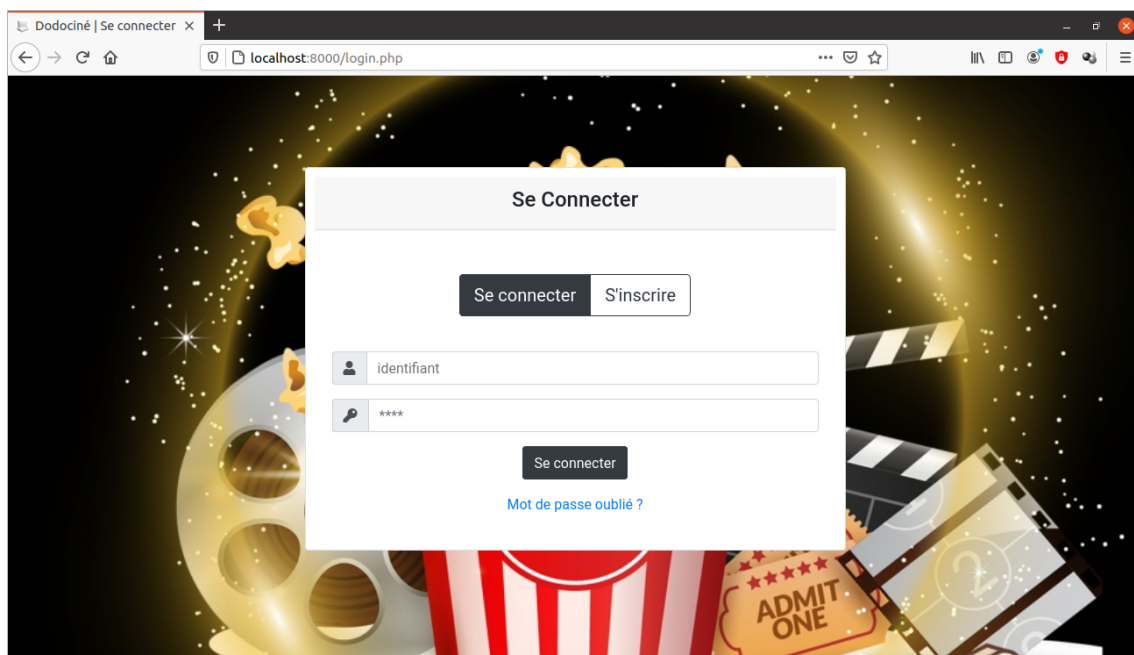


FIGURE 3.1 – Page de connexion/inscription sur DodoCiné

Si le visiteur ne possède pas encore son propre compte, il peut s'inscrire en se rendant sur l'onglet **S'inscrire** et remplir un formulaire d'inscription. Pour plus de confidentialité, on demande au futur membre de choisir un pseudonyme : ce pseudonyme servira à l'identifier sur le site tout au long de sa visite auprès des autres membres du site.

Fichiers utilisés pour cette partie du site

- *login.php* qui est la page appelée lorsque l'on clique sur le bouton "Connexion"
- *elements/login-form.php* et *elements/signup-form.php* contiennent les formulaires de connexion et d'inscription. Nous avons fait le choix de les écrire dans des fichiers séparés avant d'améliorer la visibilité de notre fichier *login.php*.
- *static/css/login.css*
- *static/js/login.js* est le script JQuery/Javascript permettant d'ajouter du dynamisme à la page.
- *login-response.php* contient le script PHP "pur" qui répond aux demandes Ajax du fichier *static/js/login.js*

Présentation de la page

En nous inspirant des dernières tendances, nous avons construit une unique page dans laquelle il est possible au choix de se connecter ou de s'inscrire. Le passage d'un formulaire à l'autre se fait via une fonction JQuery qui, selon le bouton sélectionné cache un formulaire et fait apparaître le second.

Vérification des informations renseignées dans les formulaires

Afin d'offrir une meilleure expérience aux utilisateurs du site, nous avons fait le choix d'effectuer toutes les vérifications d'informations à l'aide de Ajax et d'un script PHP. Lorsque l'utilisateur appuie sur le bouton **S'enregistrer** ou **S'inscrire**, nous effectuons un appel Ajax qui appelle à son tour notre script PHP *login-response.php* par la méthode POST. A l'intérieur de ce script, nous vérifions tout d'abord si tous les champs sont bien renseignés puis si les informations entrées sont correctes vis-à-vis du contenu de la base de données (en particulier la table **users** ici). Si des erreurs surviennent, celles-ci sont renvoyées afin que l'on puisse afficher une indication à l'utilisateur. A l'inverse, en l'absence d'erreur l'utilisateur est connecté et est renvoyé sur la page d'accueil du site.

Une fois connecté, le visiteur peut remarquer que son pseudonyme se substitue au bouton de connexion dans la barre de navigation. En cliquant dessus, il peut alors accéder à son espace personnel¹ ou tout simplement se déconnecter.

Problèmes rencontrés lors de la programmation :

- Suite à l'utilisation d'AJAX pour passer nos requêtes, nous constatons que l'utilisation de la fonction "header('Location : index.php')" ne fonctionne plus. Après recherches dans la documentation de PHP, nous avons découvert que cela était dû au fait que cette fonction ne fonctionne que si aucun code html n'est encore affiché sur la page. Pour effectuer cette redirection, nous choisissons dorénavant d'utiliser la méthode Javascript "window.location.replace(url_redirection) ;"

1. voir plus loin

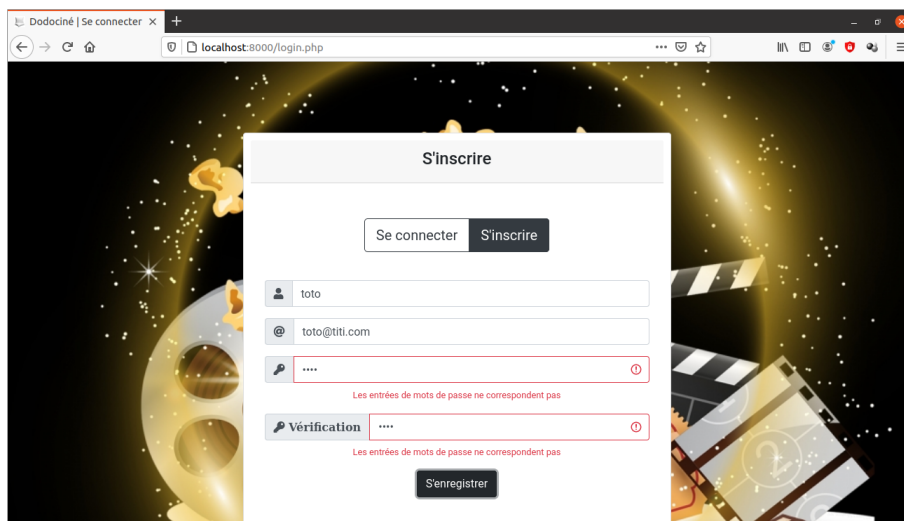


FIGURE 3.2 – Affichage d'indications d'erreurs sur les formulaires d'inscription et de connexion

- Lors de l'écriture d'un script JQuery, l'utilisation d'un argument *event* dans la fonction `$(document).ready(function(event) {})`; nécessitait l'appel de la méthode `event.preventDefault()`; afin d'empêcher le rechargement automatique de la page. Cependant, si *function* ne prend aucun argument, alors l'utilisation de la méthode n'est plus indispensable et la page ne se recharge pas.

3.2 L'espace membre associé à un compte

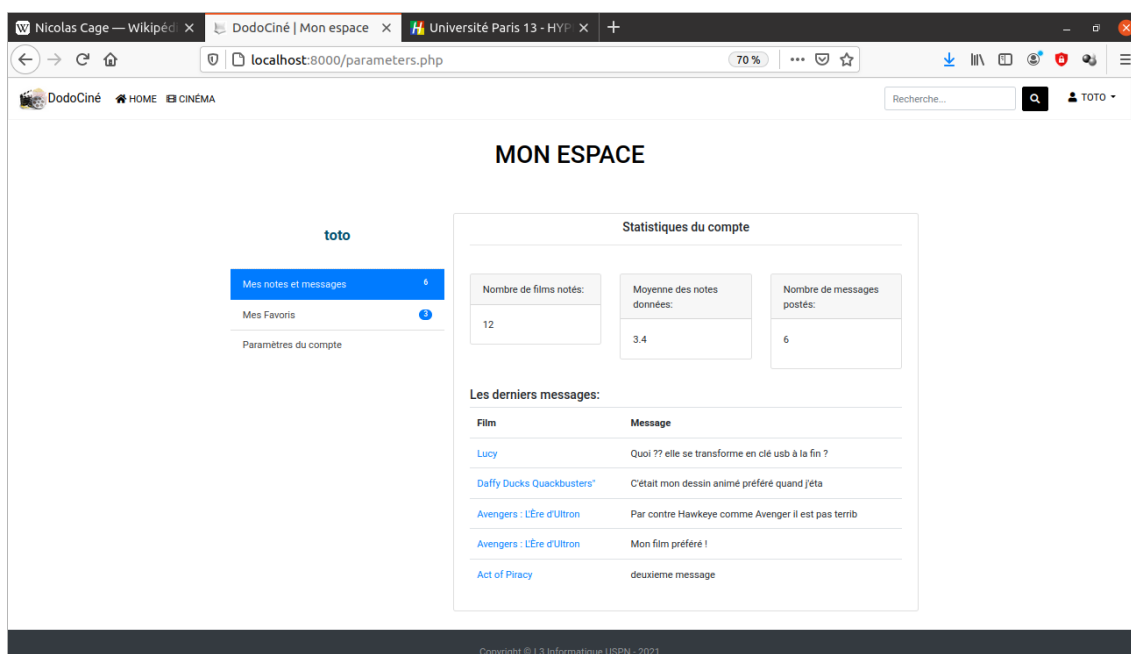


FIGURE 3.3 – Espace Membre/Paramètres dans DodoCiné

Dans cet espace, l'utilisateur peut modifier les paramètres de son compte, tel que changer son mot de passe, son email, ou même se désinscrire (entraînant la suppression de son compte), ou bien accéder plus rapidement aux messages et aux notes qu'il a postés sur le site.

Fichiers utilisés pour la partie 'espace membre'

- *parameters.php* qui est la page principale.
- *architecture des onglets de la page* :
 - *parameters-config.php* pour modifier ses informations personnelles
 - *parameters-fav.php* pour accéder à sa liste de films favoris
 - *parameters-rating.php* est l'onglet rappelant à l'utilisateur les notes qu'il a données à différents films ainsi que ses derniers messages dans différents forum.

Chacun des onglets est caché/affiché à l'aide de fonctions définies dans le script JQuery/Javascript.

- *static/css/parameters.css*
- *static/js/parameters.js* est le script JQuery/Javascript permettant d'ajouter du dynamisme à la page.
- *parameters-response.php* contient le script PHP "pur" qui répond aux demandes Ajax du fichier *static/js/login.js*

3.2.1 Rappels des notes et des messages postés

Nous proposons à l'utilisateur, dans l'onglet "mes notes et messages", de retrouver quelques statistiques sur ces interventions sur le site :

- le nombre de messages postés
- le nombre de films notés
- la moyenne des notes données

De plus, il peut voir les derniers messages qu'il a postés et avoir un lien direct pour rejoindre le fil de discussion et continuer à échanger avec d'autres utilisateurs.

Les messages sont présentés dans un tableau, avec le début du message ainsi que le film auquel se rapporte le message. En cliquant sur le nom du film, l'utilisateur retourne directement sur sa fiche et a accès au forum. Nous faisons le choix de n'afficher que les 5 derniers messages et ce pour éviter, le cas échéant, l'affichage d'un nombre trop important de messages.

3.2.2 Paramètres du compte

Dans cette section, nous avons créé deux formulaires distincts, l'un permettant de modifier le mot de passe du compte et le second, l'email. Nous avons construit ces formulaires et les appels associés de la même façon que pour la connexion/inscription d'un utilisateur : à l'aide d'Ajax, nous vérifions les erreurs potentielles entrées par l'utilisateur et les affichons le cas échéant. En cas de succès, une fenêtre modale apparaît pour indiquer que les informations ont bien été modifiées.

Cette fenêtre modale est un composant intégré du framework Bootstrap, ce qui nous permet de l'utiliser de façon simplifiée sans nous inquiéter de la partie Javascript qui est pré-programmée. De plus, ces "modals" sont très facilement modulables ce qui les rend très simples d'utilisation.

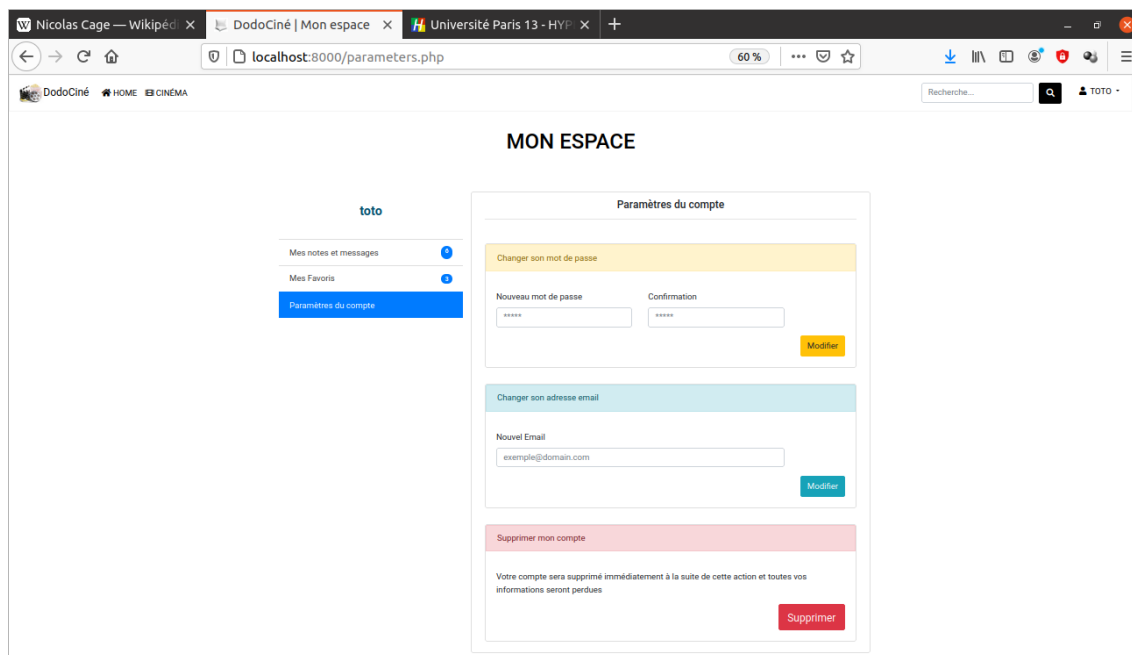


FIGURE 3.4 – Section Paramètres du compte dans l'espace membre

Enfin, le bouton **Suppression** permet à l'utilisateur, s'il le souhaite, de supprimer de façon définitive son compte. En cas de sélection, le compte est supprimé, et les informations concernant cet utilisateur seront supprimées. Néanmoins, les notes données, ainsi que les messages postés sont conservés car devenus publics.

3.2.3 Favoris

Nous avons souhaité ajouter une petite fonctionnalité à notre site en offrant à nos utilisateurs la possibilité de "sauvegarder" les films qui l'intéressent, par exemple pour créer une liste de films à visionner, ou de films qui lui plaisent particulièrement. Pour cela, il lui suffit de cliquer le bouton "Ajouter aux favoris" disponible sur la fiche d'un film. Tous les films ajoutés sont ensuite présentés dans la section **Mes Favoris** de son espace Membre.

4. PRÉSENTATION DES FILMS

4.1 Présentation des films disponibles sur le site

Nous proposons dans l'onglet "Cinéma" de notre menu de navigation, la liste complète des films disponibles sur notre site. De cette façon, le visiteur peut découvrir au gré de ses envies de nouveaux films.

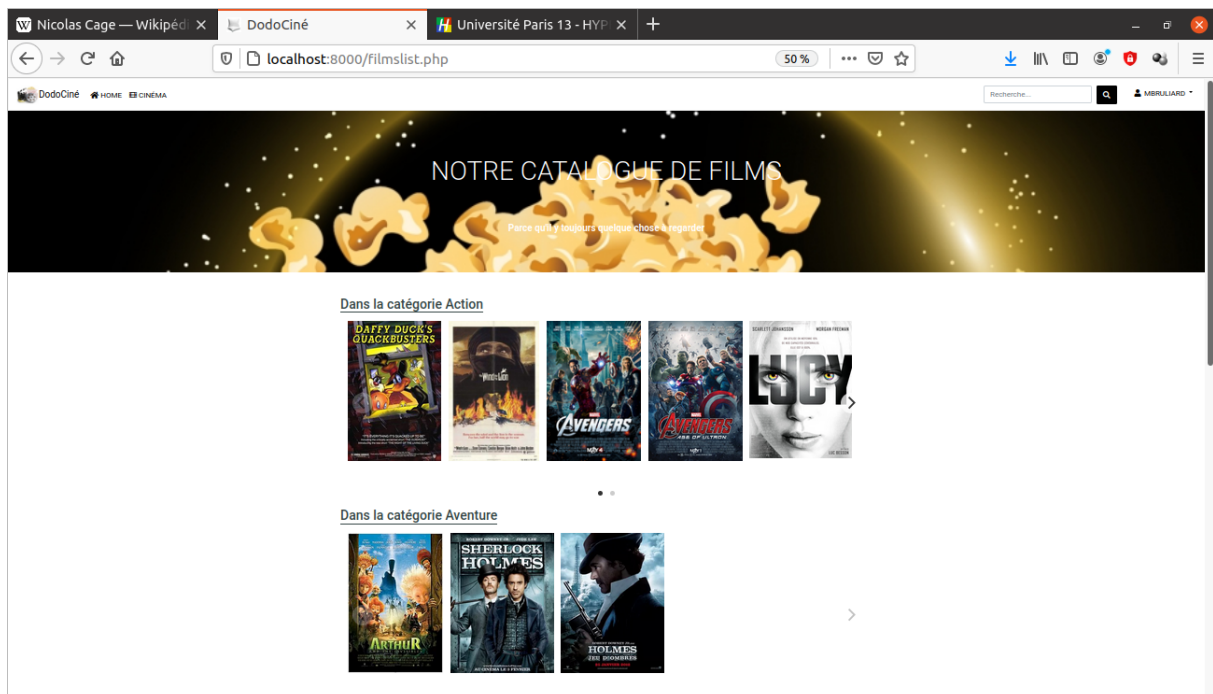


FIGURE 4.1 – Page Cinéma de notre site

Les films sont ici regroupés par catégorie et présentés sous la forme d'un "slider carrousel". Nous avons notamment trouvé un framework <https://flickity.metafizzy.co> open-source (licence GPLv3) pour les projets personnels et universitaires que nous avons choisi ici d'utiliser. Ce framework est "responsive" et donc s'adapte parfaitement à notre site.

L'affichage des "slider carrousel" est géré automatiquement : pour chaque catégorie, nous récupérons via une requête PDO l'ensemble des films correspondant à la catégorie dans un tableau (array en PHP). Nous appelons ensuite le script PHP `elements/films-slider.php` auquel on applique le tableau de films.

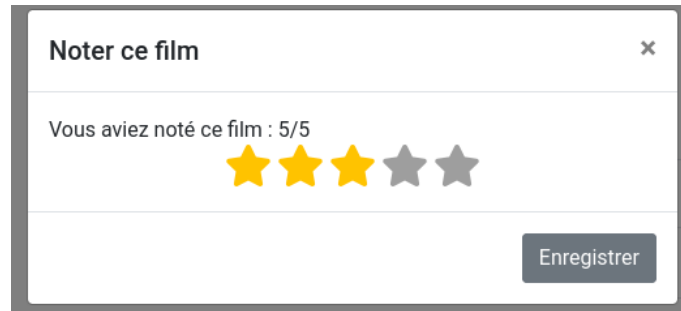


FIGURE 4.2 – Fenêtre modale permettant de noter un film

4.2 Page de présentation d'un film

En cliquant sur un film de notre liste, le visiteur peut afficher une fiche descriptive du film sur laquelle sont renseignées des informations telles que l'année de sortie du film, son réalisateur, les acteurs principaux y ayant participés, etc.

En plus de cela, on propose à l'utilisateur des films en lien avec le film sélectionné :

- un film de même catégorie
- un film fait par le même réalisateur
- un autre film dans lequel a joué un des acteurs

Pour cela on va utiliser des requetes SQL (via PDO de PHP) pour récupérer les résultats puis nous tirerons au hasard un des ses résultats pour l'afficher à l'aide d'une fonction incluse dans PHP : "array_rand". L'algorithme se révèle en réalité très semblable à celui utilisé pour l'affichage aléatoire d'un acteur et d'un film sur la page d'accueil du site.

Enfin, nous proposons aux utilisateurs ayant un compte sur notre site plusieurs fonctionnalités supplémentaires :

- noter un film
- ajouter un film à ses favoris¹
- écrire dans le forum

4.2.1 Construction du système de notation des films

Pour ce faire, nous utilisons un système de notation très à la mode en ce moment : la notation par étoiles (5 étoiles maximum). Les films sont alors notés sur 5.

Font Awesome

*Les étoiles, ainsi que la plupart des icônes présentes sur les boutons de notre site proviennent du framework **Font Awesome**.*

La note du film est la moyenne de toutes les notes données par les utilisateurs du site. De plus, un utilisateur connecté peut noter/modifier sa note sur un film en cliquant simplement sur le bouton "Noter ce film". Une fenêtre modale (une fenêtre modale intégrée à Bootstrap) apparait alors pour que l'utilisateur entre sa note.

1. c.f. section sur l'espace membre

L'enregistrement de la note de l'utilisateur, et la mise à jour de la note globale du film se fait ensuite via AJAX.² On utilise une méthode POST qui renvoie 3 choses : l'ID de l'utilisateur, l'ID du film à noter et la note.

Lorsque la fenêtre modale se referme, la fonction AJAX met directement à jour les informations sur la note.

4.2.2 Gestion du forum relatif au film

Commentaires:



FIGURE 4.3 – Exemple d'espace commentaire (ici pour le film *Lucy*)

Dans ce projet, on considère que les utilisateurs connectés peuvent écrire dans le forum du film pour discuter entre eux. Si un visiteur du site n'a pas de compte, il peut lire les messages postés mais ne peut pas écrire ou répondre à des messages.

Un utilisateur connecté peut :

- écrire un message dans le forum du film
- supprimer un message qu'il a posté auparavant

Liste des fichiers utilisés

- *elements/forum-message.php* : pour écrire un message
- *elements/forum".php* pour la gestion du forum. Il sera appelé pour chaque film
- *static/css/forum.css* pour le style
- *static/js/forum.js* pour la gestion des evenements
- *controller/dataforum.php* pour les fonctions faisant appel a la base de données
- *forum-response.php* pour les appels ajax

Finalement le fichier *elements/forum-message.php* n'est pas utilisé dans la construction du forum car nous ne sommes pas parvenu à faire l'appel succesif. A la place, nous écrivons directement la partie HTML à l'aide de l'appel de *forum-response.php* via AJAX. Ce n'est probablement pas la méthode la plus simple, ou la plus efficiente mais il s'agit de la seule méthode que nous avons trouvée...

2. Nous avons fait le choix de n'utiliser que le format JSON pour le transfert de données via AJAX car celui-ci est moins lourd et plus simple à manier

En plus d'afficher le message, son auteur et la date à laquelle il a été posté, nous créons un bouton permettant à un utilisateur de supprimer son propre post. Ce bouton n'est bien sûr visible que par lui lorsqu'il est connecté à son compte.