

TEC Code 2020

Projet Cod’Flix

Présentation du projet

Pour un lancement prévu en Mai 2021, Cod’Flix souhaite se faire une place sur le marché de la VOD (Video On Demand). La concurrence est intense avec des acteurs comme Netflix, Disney+ ou encore OCS by Orange.

Vous venez de rejoindre cette startup et votre objectif est de travailler sur le projet web de l’entreprise. A ce jour, la plateforme est en développement et nécessite l’ajout et la correction de plusieurs fonctionnalités. Disponible en Java ou en PHP, les fonctionnalités suivantes ont été implémentées :

- S’identifier à la plateforme via une adresse email et un mot de passe.

Les identifiants sont les suivants :

- **Email** : coding@gmail.com
- **Mot de passe** : 123456
- Afficher la liste des films disponible sur la plateforme.
- Rechercher un film par son titre.
- Déconnexion de l’utilisateur.
- Le HTML et le CSS ont été implémentés sur les pages suivantes :
 - Accueil
 - Connexion
 - Inscription
 - Dashboard
 - Liste des films

La base de données se composent des tables et colonnes suivantes :

- **user**
 - id - *Clé primaire*
 - email - *Adresse email*
 - password - *Mot de passe*
- **genre**
 - id - *Clé primaire*

- name - Nom du genre (Action, Aventure, etc...)
- **media - Le média est la fiche du contenu**
 - id - Clé primaire
 - genre_id - Clé étrangère avec la table "genre" et sa colonne "id"
 - title - Titre du média
 - type - Type de média (film ou série)
 - status - Média publié ou en brouillon
 - release_date - Date de sortie
 - summary - Résumé
 - trailer_url - Lien vers le trailer
- **history - Historique des medias visionnés**
 - id - Clé primaire
 - user_id - Clé étrangère avec la table "user" et sa colonne "id"
 - media_id - Clé étrangère avec la table "media" et sa colonne "id"
 - start_date - Date de début de visionnage du contenu
 - finish_date - Date de fin de visionnage du contenu
 - watch_duration - Minutes actuelles de visionnage (exemple : 34ème minutes)

Installation du projet

Vous avez le choix entre Java et PHP pour faire votre développement. Voici les URLs de chaque repo sur GitHub :

- <https://github.com/synapticvoid/ec-code-2020-codflix-java>
- <https://github.com/synapticvoid/ec-code-2020-codflix-php>

Chaque repo contient l'intégralité du projet (Backend / Frontend / script SQL). Les instructions d'installation de votre machine de dev sont décrites dans le README.md de chaque projet.

Vous devez copier un de ces projets dans un repo à vous (via un Fork sur GitHub ou une copie manuelle) et c'est sur ce nouveau repo que vous ferez votre développement.

Consignes

Général

- Vous devez créer votre repo GitHub à partir de celui que vous avez choisi (PHP ou Java)
- Libre à vous de changer de techno en cours de route
- Vous ne devez rendre qu'un seul repo à partir d'un seul langage choisi (Java ou PHP)

- Le rendu final sera un lien vers votre repo GitHub qui sera consultable par les P.O (vérifiez qu'il est accessible en public). **Vous avez jusqu'au jeudi 25/06 à 17h pour envoyer le lien de votre repo**
- **Vous devez déposer l'URL du repo dans un fichier texte sur BlackBoard**
- **Chaque heure supplémentaire pour envoyer votre lien fera un malus de 2 points sur la note finale (ramenée sur 20 points)**

Code

- L'indentation, nommage des variables, qualité des commentaires sont importants et pris en compte dans la notation
- Cod'Flix vise l'international, vous devez donc favoriser l'anglais (code, commentaires, commit git) pour que des développeurs non français puissent intervenir dans le projet
- Veillez à faire des messages de commit git explicites (Pas juste un "Fix", "Coucou", etc)
- Cod'Flix n'a pas encore d'application mobile. Le design des pages doit donc être compatible avec des smartphones. **Chaque page doit être responsive et supporter correctement l'affichage sur mobile**
- **Toute modification de la structure de la base de données (ajout de table, colonne, autre) doit être incluse dans le fichier codflix.sql se trouvant à la racine du repo**
- Le site sera ouvert au grand public, ce qui implique un risque de sécurité majeur. Vous devez donc vérifier toutes les informations transmises par un client (validation de formulaire, validation des types de paramètres, etc). Ces vérifications doivent être faites aussi bien côté client (JavaScript) que côté serveur (Java ou PHP)

Ajout de bibliothèque

Vous pouvez ajouter la bibliothèque de votre choix dans n'importe quelle partie (backend, front end, SQL). Gardez en tête que chaque ajout d'une bibliothèque est un risque pour Cod'Flix car :

- Il faudra mettre à jour la bibliothèque régulièrement
- Il peut y avoir des bugs dans la bibliothèque qu'il faudra contourner / corriger
- Il faut que les autres développeurs de Cod'Flix apprennent à l'utiliser et soient à l'aise avec elle

Chaque ajout devra être justifié par écrit dans le repo. Pour se faire, vous devez créer un fichier TECH.md à la racine de votre repo listant les bibliothèques que vous avez ajouté avec une explication pour chacune. Voici un exemple du fichier TECH.md :

Bootstrap v4.5.0

Bootstrap est un composant permettant de simplifier le développement en front web (HTML / CSS) en structurant le rendu des pages. J'ai décidé d'inclure Bootstrap car cette bibliothèque apporte :

- *L'uniformisation du rendu des pages en découpant par colonne*

- Le support du responsive design et donc un rendu sur mobile correct
- Un affichage propre des boutons, formulaires, modales, etc

Bootstrap est une bibliothèque Open-Source (licence MIT). Cela permet à Cod’Flix de modifier librement le code si le besoin se présente. Il n’y a pas de risques de sécurité à priori concernant cette bibliothèque puisqu’elle ne concerne que l’affichage pur des éléments en HTML.

Cette bibliothèque est très utilisée et la communauté est active :

- Utilisée par 21% de tous les sites web (source : <https://w3techs.com/technologies/details/js-bootstrap>)
- 142 000 stars sur GitHub (<https://github.com/twbs/bootstrap>)
- Plus de 350 000 questions sur StackOverflow
- Le développement reste actif avec des commits datant du mois de juin 2020

Votre justification doit donc contenir les éléments suivants :

- La version de la bibliothèque incluse
- A quoi sert la bibliothèque en quelques mots
- Quelle est la licence utilisée par la bibliothèque et quelles libertés / restrictions pour Cod’Flix
- Est-ce qu’il y a des risques de sécurité en partant sur cette bibliothèque (failles connus, etc)
- Est-ce qu’elle est très utilisée, quelle est la taille de la communauté, son activité pour savoir si on trouvera facilement des informations pour son utilisation / ses bugs
- Est-ce que la bibliothèque est toujours en développement

Décider de ce que vous affichez

Cod’Flix est encore une petite startup et toute l’équipe touche à tous les aspects du produit. Vous aurez parfois la demande de développer des fonctionnalités sans forcément avoir le détail tout ce qui doit être affiché. Ceci n’est pas un oubli dans l’énoncé mais un moyen de vous pousser à réfléchir au produit “au delà” du code.

Faites preuve d’initiative :

- Mettez vous à la place de l’utilisateur : quelle information pertinente a-t-il envie de voir sur telle page ? Que faut-il mettre en avant ?
- Regardez la concurrence : que font les autres plateformes de VODs pour la même fonctionnalité ? Pourquoi ont-ils fait ces choix ?

Fonctionnalités communes - L1, L2 et TP Reco

[Afficher le détail d'un média](#)

Actuellement le site affiche uniquement la liste des médias disponibles.

Vous devez donc créer la route, ainsi que la page HTML permettant de fournir à l'utilisateur le détail d'un média. Le CSS devra également être créé afin d'afficher correctement le contenu.

Le rendu doit être correct sur mobile également.

[Support des séries](#)

A ce jour, le site permet uniquement d'afficher la liste des films disponibles.

- Créer la fonctionnalité permettant de voir apparaître des séries dans la liste des médias
- Lorsque l'utilisateur ira vers le détail d'une série, il aura la possibilité de choisir parmi les épisodes disponibles
- Les épisodes seront eux mêmes organisés par saison
- Lorsque l'utilisateur cliquera sur un épisode, la lecture se lancera de la même façon qu'un film. On parle ici d'un "stream"
- Pour simplifier le développement, on considère que visionner un stream revient à regarder une vidéo YouTube (intégrée dans la page du stream)
- Attention : la liste des médias doit toujours afficher des informations de médias. Il ne faut pas casser l'existant mais le compléter
- Chaque "stream" doit avoir au minimum :
 - Un titre
 - Une durée
 - Une courte description
 - Une URL du média (pointant sur une vidéo YouTube par exemple)
- L'affichage des médias (liste et détail) doit inclure la durée total du média au format HHhmm (exemple: 06h56)
- L'historique de visionnage doit gérer les streams

Le rendu doit être correct sur mobile également.

Formulaire d'inscription

Le template de la page d'inscription est déjà existante.

- Créer la fonctionnalité permettant à l'utilisateur de s'inscrire. Pour rappel, une inscription correspond à un enregistrement dans la base de données
- Créer la fonctionnalité permettant à l'utilisateur de confirmer son adresse email

Le processus d'inscription est le suivant :

1. L'utilisateur saisit son email et un mot de passe
2. Un email est envoyé avec un lien unique
3. L'utilisateur clique sur le lien
4. Son compte est mis à jour et est noté comme étant activé

Bien entendu, vous travaillez en local, les mails ne partiront pas. Néanmoins, la fonctionnalité sera créée même si elle n'a pas été testée.

Historique de visionnage

- Créer la page pour afficher l'historique (HTML et CSS)
- Afficher l'historique avec les informations de chaque stream
- Possibilité de supprimer un élément de l'historique
- Possibilité de supprimer l'intégralité de l'historique via un bouton "Supprimer mon historique"

Le rendu doit être correct sur mobile également.

Corriger la recherche

Actuellement la fonction de recherche ne fonctionne pas, mais le développeur qui a travaillé sur cette fonctionnalité est parti. Vous devez faire en sorte que la recherche fonctionne sur le titre d'un média.

Une partie du titre doit pouvoir être saisie pour retrouver un média.

Exemple :

Pour "Les" nous aurons comme résultats : "Les indestructibles" et "Les misérables"

Pour "Les ind" nous aurons comme résultat : "Les indestructibles"

Améliorer la sécurité des mots de passe

- Les mots de passe devront être hachés en SHA-256. Vous reprendrez donc les fonctionnalités de login et d'inscription afin de garantir aux utilisateurs une sécurité optimale
- Si vous voyez d'autres moyens d'améliorer la sécurité des mots passe, faites-le en expliquant en commentaire dans le code votre démarche

Afficher la date de sortie d'un film sur les preview

- Ajouter la date de sortie du média pour chaque média sur la page de recherche

Fonctionnalités bonus L1 - Obligatoire pour les L2 et TP Reco

Améliorer la recherche

Vous améliorerez la recherche afin de proposer à l'utilisateur la possibilité de rechercher un média avec les critères suivants :

- Par genre
- Par date de sortie (par exemple, tous les films sortie avant 2001)
- Par type de média (film ou série)

Autre fonctionnalité à ajouter :

- Les médias affichés sur la page seront différenciés auront un style différent suivant le type de média. Par exemple, les films et les séries seront différenciable d'un coup d'oeil. **Libre à vous de les différencier de la façon qu'il vous convient**
 - Les requêtes de recherche se feront en AJAX afin d'éviter un rechargement de page
 - Les résultats de recherche s'affinent à mesure où l'utilisateur saisira sa recherche
- Exemple :

Pour "Les" nous aurons comme résultats : "Les indestructibles" et "Les misérables"

Pour "Les ind" nous aurons comme résultat : "Les indestructibles"

Ajouter une page nous contacter

- Créer une page nous contacter (HTML et CSS)
- Créer la fonctionnalité permettant d'envoyer un mail à l'adresse "contact@codflix.com"
- Ajouter le lien sur l'écran d'accueil du site (/)

Cette page doit être accessible sans avoir une session utilisateur connectée.

Le rendu doit être correct sur mobile également.

Bien entendu, vous travaillez en local, les mails ne partiront pas. Néanmoins, la fonctionnalité sera créée même si elle n'a pas été testée.

Fonctionnalités L2 / TP Reco

MAJ de l'historique à partir du visionnage

- Si un utilisateur quitte un média en cours de lecture, il sera enregistré dans l'historique la minute à laquelle il se trouvait
- Reprendre le média à la minute à laquelle l'utilisateur s'est arrêté
- Ces informations seront enregistrées via une requête AJAX

Ajouter une page profil utilisateur

Vous créerez les fonctionnalités suivantes :

- Une page "Mon profil" (HTML et CSS)
- Modification du mot de passe (avec confirmation par l'ancien mot de passe)
- Modification de l'adresse email
- Suppression complète du compte avec confirmation par le mot de passe. Ceci entraînera la suppression de toutes les données associées à l'utilisateur

Astuce : Vérifiez vos clés étrangères.

Le rendu doit être correct sur mobile également.

Fonctionnalités bonus L2 et TP Reco

Proposer des médias en lien avec l'historique

- A partir des médias visionnés vous proposerez des médias à l'utilisateur.
Vous étudierez les critères suivants :
Regarde t'il plus de série ou de film ?
 - Quel genre regarde t il le plus souvent ?

- Quel genre de film/série a t il regardé mais pas terminé ? Dans ce cas là vous excluez ces genre de film et série de vos propositions
- Les médias proposés seront affichés sur l'accueil, juste au dessus du formulaire de recherche. Vous afficherez médias sous le titre "Médias susceptibles de vous plaire".

Récupérer des informations depuis une API tierce

- Pour détailler la fiche d'un média, vous utiliserez l'API suivante :

<https://www.themoviedb.org/documentation/api>

Vous afficherez un maximum d'informations sur le média provenant de votre API.

Implémenter une liste de souhaits

- Ajouter une icône "favoris" à côté de chaque média. Dans la recherche et également sur la fiche du média.
- L'utilisateur aura accès à ses favoris sur la page d'accueil, au dessus du formulaire de recherche sous le titre "Vos médias à regarder"