



**POLYTECHNIQUE
MONTRÉAL**

UNIVERSITÉ
D'INGÉNIERIE

Travail pratique #4: Applications et Bases de données

École polytechnique de Montréal

Trimestre : Hiver 2020

Équipier1 : Joe *ABDO* *1 939 689*

Équipier2 : Hugo *PERRONNET* *1 885 263*

Présenté à : Mathieu Châteauvert

Polytechnique Montréal

Remis le 19 avril 2020

Table des matières

Introduction	3
Modèle conceptuel UML	4
Hypothèses	5
Modèle relationnel en syntaxe abstraite	6
Détails sur les requêtes.	7
Algèbre relationnelle	7
Présentation de l'application	8
Présentation de l'application	8
Administrateur	8
Utilisateur	11
Aspects Novateurs	13
Raccourcis	13
Sécurité	14
Originalité	14
Guide d'installation	15
Base de données	15
Serveur	15
Client	15
Proxy	15
Tests	16
Administrateur	16
Subscribed User	16
Pay Per View User	16

Introduction

Le TP4 vient sous forme d'un mini-projet, qui regroupe les différents aspects d'une base de données, de la création, à sa population tout en touchant à sa gestion et en effectuant des requêtes à partir d'une application.

Dans notre cas, le but est de créer une application web avec Angular et Node JS qui permet à un utilisateur de regarder ou de commander un film. Cependant derrière ces deux fonctionnalités se posent beaucoup de zones grises. Comme savoir où l'utilisateur s'est arrêté dans son visionnement? L'utilisateur a-t-il le droit de visionner? Combien est-ce que la livraison du DVD va coûter?

Mais aussi d'un autre côté il faut donner un accès simplifié à un administrateur qui va se charger d'ajouter des films à cette base de données mais aussi des utilisateurs. De ce côté aussi beaucoup de questions se posent pour pouvoir arriver à une bonne solution parmi ces questions les suivantes: Comment savoir si l'utilisateur n'existe pas déjà?

Comment relier des cérémonies et des participants au film? Comment savoir si ces participants n'existent pas déjà et ne pas avoir à les dupliquer dans la base de données?

Ces questions et bien d'autres que nous nous sommes posées au début de notre travail nous aideront à réaliser le travail avec clarté, et ainsi établir une architecture et un plan de maquette dès le début.

Modèle conceptuel UML

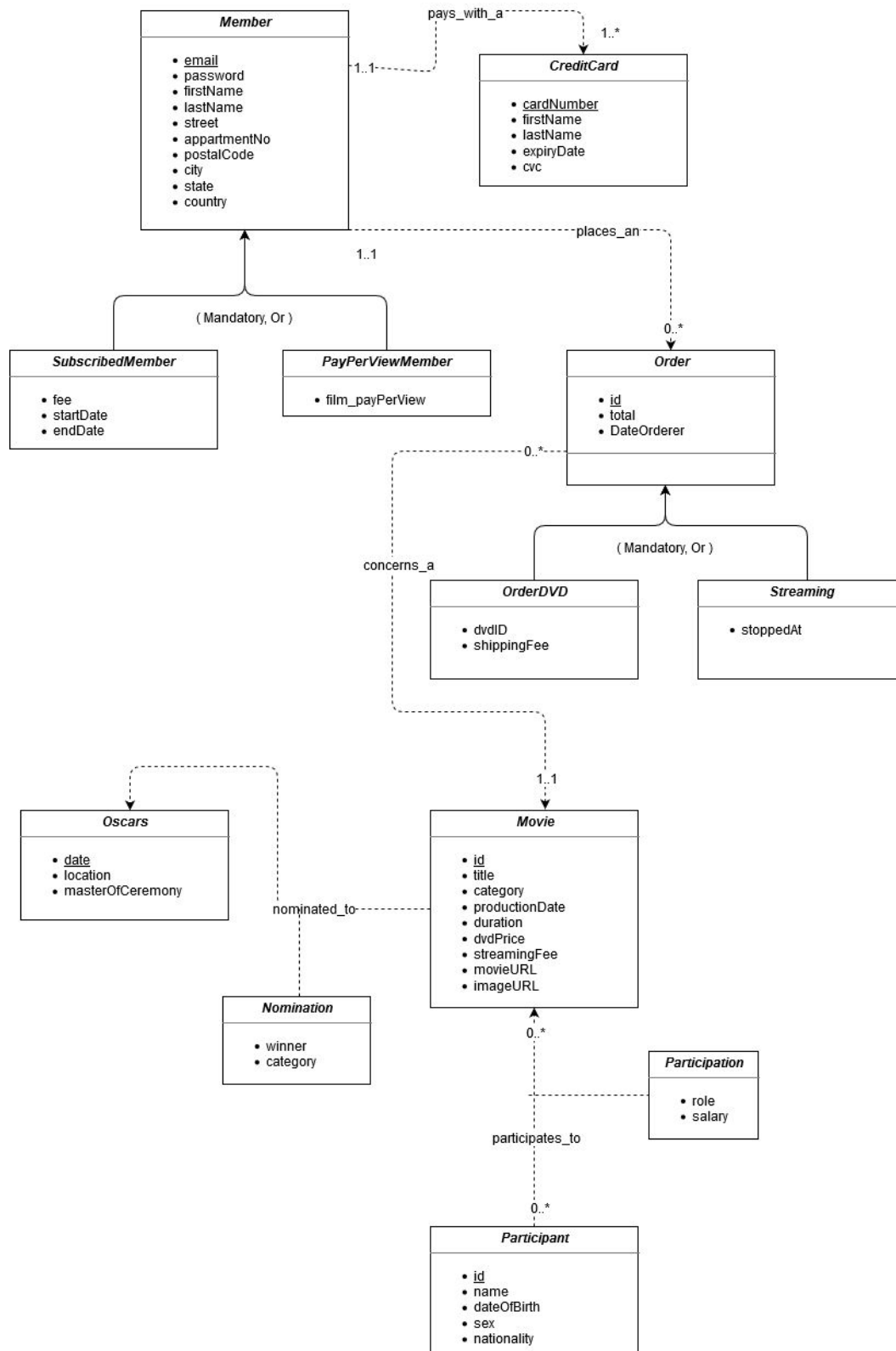


Schéma 1: Modèle de Netflix_Poly, base de donnée

Hypothèses

Ceci est la représentation de l'étude de cas présentée nous supposons que:

- Deux cérémonies d'oscars ne peuvent pas avoir lieux au même moment, doit le choix de la clef primaire qui est la date de la cérémonie.
- Un client est obligatoirement inscrit sur le site pour visionner, il doit donc obligatoirement avoir un forfait que ce soit, un abonné ou qu'il paie par visionnement.
- Une commande (order) est singulière, c'est à dire une commande est un achat de film ou d'un DVD et ne peut pas contenir plus d'un item.

Modèle relationnel en syntaxe abstraite

Member (email, password, firstName, lastName, street, apartmentNo, postalCode, city, state, country)

SubscribedMember (email, fee, startDate, endDate)
FK: email References Member(email)

PayPerViewMember (email, film_payPerView)
FK: email References Member(email)

CreditCard (cardNumber, firstName, lastName, expiryDate, cvc, ownerID)
FK: ownerID References Member(email)

Order (id, dateOrder, movieID, clientID)
FK: movieID References Movie(id)
FK: clientID References Member(email)

OrderDVD (id, dvdID, shippingFe)
FK: id References Order(id)

Streaming (id, stoppedAt)
FK: id References Order(id)

Movie (id, title, category, productionDate, duration, dvdPrice, streamingFee, movieURL, imageURL)

Participant (id, name, dateOfBirth, sex, nationality)

Participation (movieID, participantID, role, salary)
FK: movieID References Movie(id)
FK: participantID References Participant(id)

Oscars (date, location, masterOfCeremony)

Nomination (dateOfCeremony, movieID, winner, category)
FK: dateOfCeremony References Oscars(date)
FK: movieID References Movie(id)

Détails sur les requêtes.

Pour la 3ème requête, nous avons décidé d’afficher tous les membres ayant le plus de visionnements. Par conséquent, si plusieurs membres ont le nombre de visionnement le plus élevée, nous les affichons tous.

Algèbre relationnelle

1) *Affichez toutes les informations sur un film spécifié par l'utilisateur (selon le titre)*

Soit x le titre spécifié par l'utilisateur :

- $\sigma_{\text{titre}=x}(\text{movie})$

2) *Pour chaque genre de film, listez tous les titres de films ainsi que la dernière date à laquelle un film a été acheté (DVD) ou visionné*

- $M \leftarrow \pi_{\text{title, category, movieId}}(\text{movie})$

- $O \leftarrow \pi_{\text{movieId, dateOrder}}(\text{Order})$

- $\pi_{\text{category, title, } \mathfrak{J}\text{MAX}_{(\text{dateOrder})}}(M \bowtie O)$

3) *Trouvez les noms des membres dont le coût total d’achat de DVD est plus élevé que la moyenne*

- $\text{TotalCost} \leftarrow \rho_R(\text{email, firstname, lastname, sumcost})_{\text{email, firstname, lastname,}}$

$\mathfrak{J}\text{SUM}_{(\text{dvdprice} + \text{shippingfee})}(\text{orderdvd} \bowtie \text{order} \bowtie \text{movie} \bowtie \text{member})$

- $\text{AverageCost} \leftarrow \mathfrak{J}\text{AVG}_{(\text{sumcost})}(\text{TotalCost})$

- $\pi_{\text{firstname, lastname}}(\sigma_{\text{sumcost} > \text{AverageCost}}(\text{TotalCost}))$

Présentation de l'application

Présentation de l'application

Administrateur











ADMINISTRATOR

NETFLIXPOLY

Tasks

- Home
- Add Movie
- Add User

Filter

Title	Genre	Release Date	Duration (s)	DVD Price	Streaming Fee	Delete
Fast and Furious 9	Action	2020-04-01T04:00:00.000Z	230	70	5	
The Transporter 3	Crime	2008-11-26T05:00:00.000Z	125	30	2	
Spenser Confidential	Action	2020-03-06T05:00:00.000Z	177	60	5	
Salt	Drama	2010-07-23T04:00:00.000Z	118	30	3	
Taken	Thriller	2008-01-30T05:00:00.000Z	112	20	2	
Skyfall	Mystery	2012-11-09T05:00:00.000Z	153	25	3	
Central Intelligence	Crime	2016-06-17T04:00:00.000Z	147	40	3	
Ride Along	Comedy	2016-01-15T05:00:00.000Z	154	40	4	
Top Gun	Action	1986-04-12T05:00:00.000Z	88	10	1	
Titanic	Drama	1997-11-18T05:00:00.000Z	117	10	1	

Items per page: 10 1 - 10 of 10 < >

Capture d'écran 1: Page d'accueil, qui montre les films déjà disponibles

ADMINISTRATOR
NETFLIXPOLY

Tasks
Home
Add Movie
Add User

Register a New Movie

Title

Title is required

Category

Category is required

Release Date

Duration (s)

Duration is required

DVD Price (\$)

DVD Price is required

Streaming Fee (\$)

Streaming Fee is required

Movie URL

Movie URL is required
Valid Movie URL is required

Image URL

Image URL is required
Valid image URL is required

Participant's Name

Name is required for a Participant

Date Of Birth

Capture d'écran 2: Formulaire pour l'ajout d'un film

Image URL is required
Valid image URL is required

Participant's Name

Date Of Birth

Nationality

Sexe

Role

Salary (\$)

Add Participant

Name	Role	Delete
Participant 1	actor	

Host's Name

Name is required for a Host

Ceremony's Date

Ceremony's Location

Capture d'écran 3: Formulaire pour l'ajout d'un film, partie d'ajout de participant

Le formulaire d'ajout d'un film, permet d'ajouter un film avec des participants et des nominations. En effet si un participant existe déjà l'application ne l'ajoute pas encore mais ajoute son rôle dans le nouveau film.

ADMINISTRATOR
NETFLIXPOLY

Tasks

- Home
- Add Movie
- Add User

Register a New User

First Name
Utilisateur

Last Name
1

Email

Email is required
Email must be a valid email address

Password

Password is required
Password must be at least 6 characters
Password must contain at least one special character
Password must contain at least one upper case character
Password must contain at least one number

Address
rue labelle

Number
530

Postal Code

Postal Code is required
Postal Code must be a valid postal code

City
Laval

Province
Quebec

Country
Canada

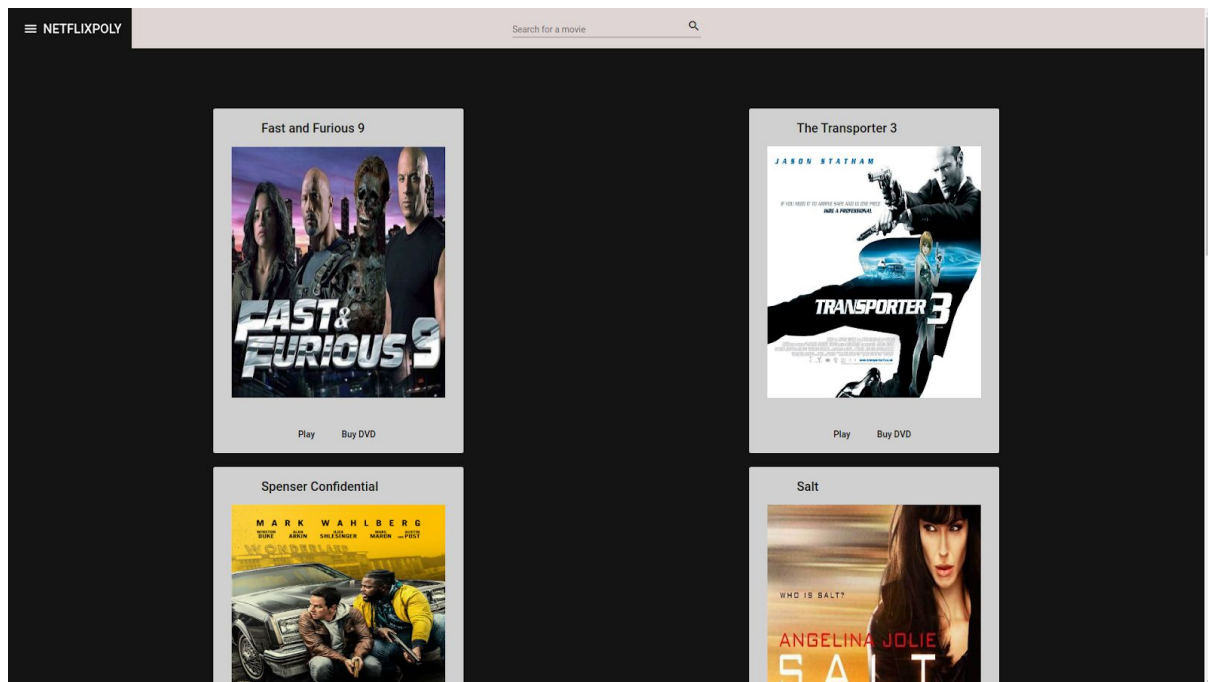
Subscription

Capture d'écran 4: Formulaire d'ajout d'un utilisateur

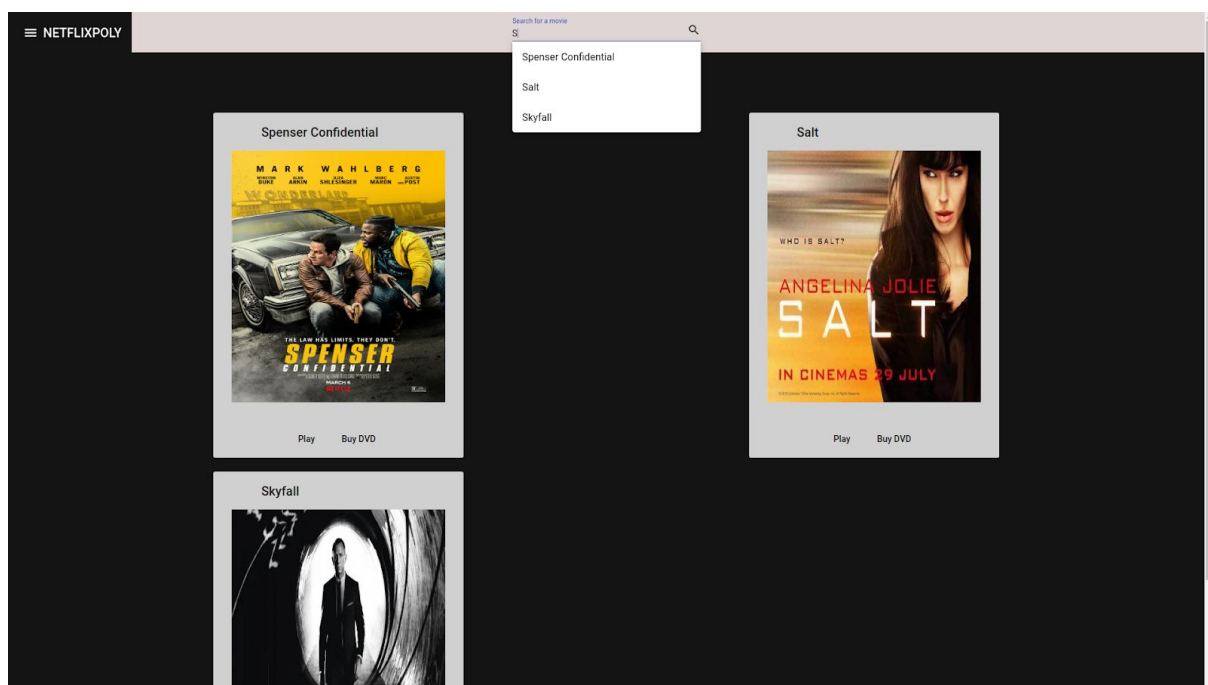
Le formulaire d'ajout d'un utilisateur permet d'ajouter un utilisateur en fonction de son type d'abonnement.

Les formulaires présentent des contrôles d'erreurs, certains conçus avec des expressions régulières, de plus ces contrôles d'erreurs ne sont pas conçus pour prévention de toutes les erreurs vu qu'ils se trouvent du côté administrateur on s'attend à une certaine attention de la part de l'administrateur.

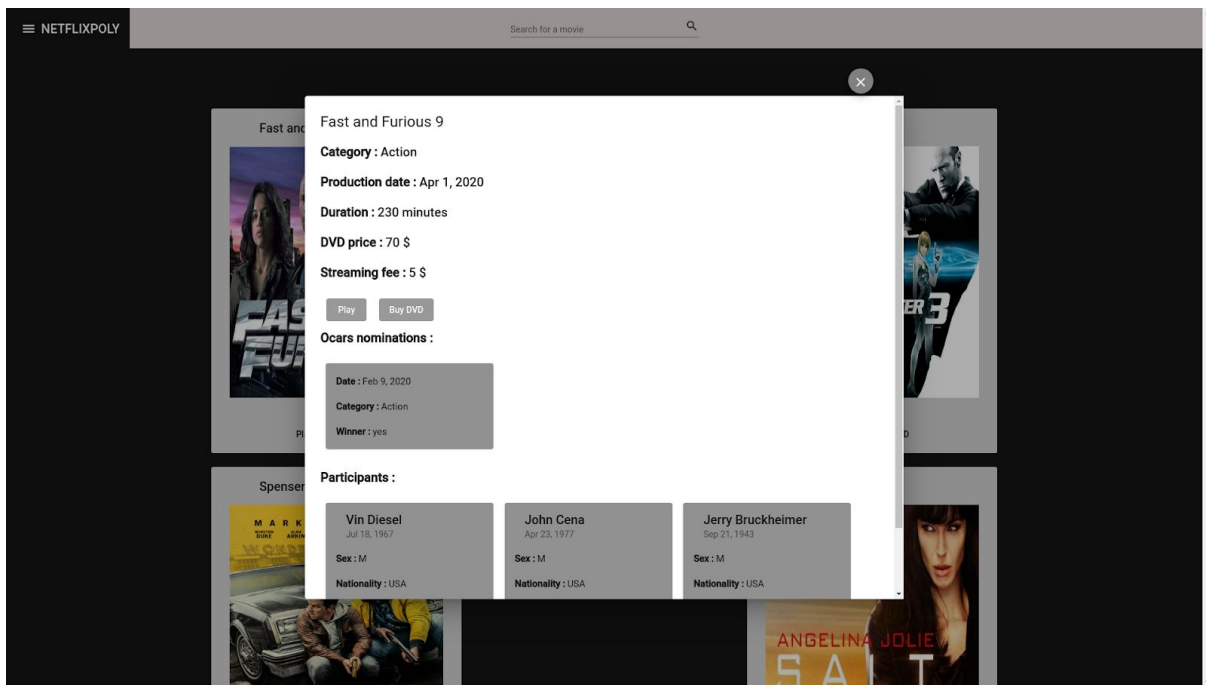
Utilisateur



Capture d'écran 5: Page d'accueil, qui montre les films disponibles.

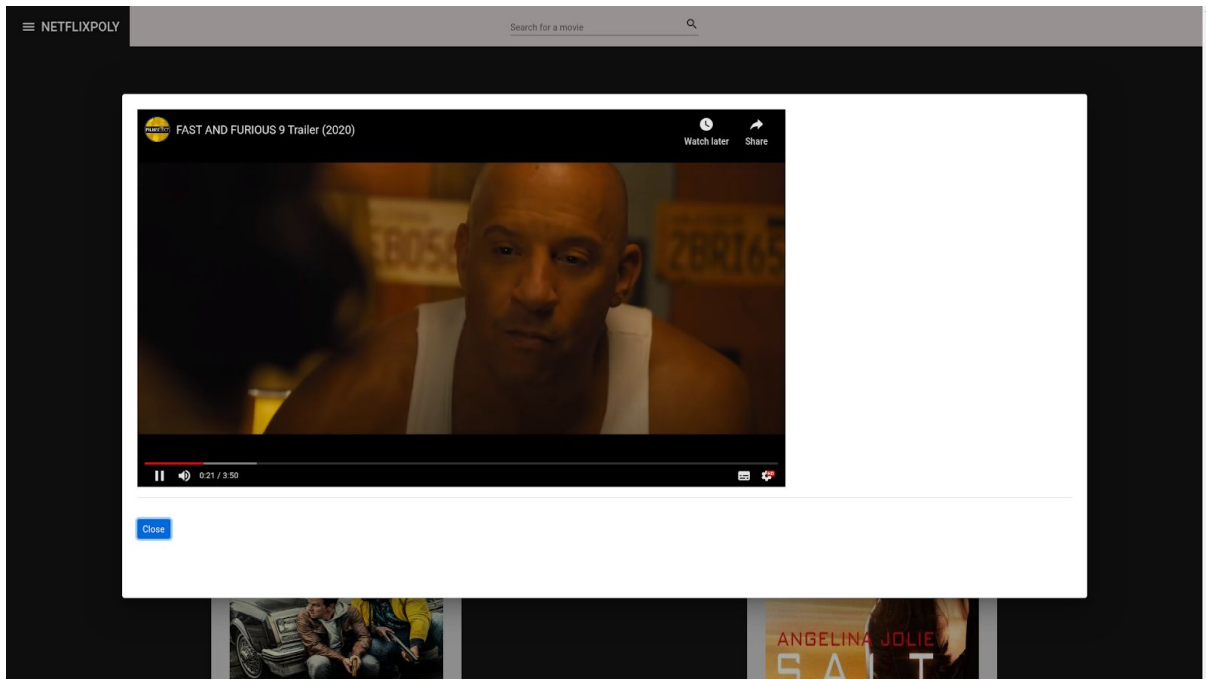


Capture d'écran 6: Filtrage des films en utilisant la barre de recherche



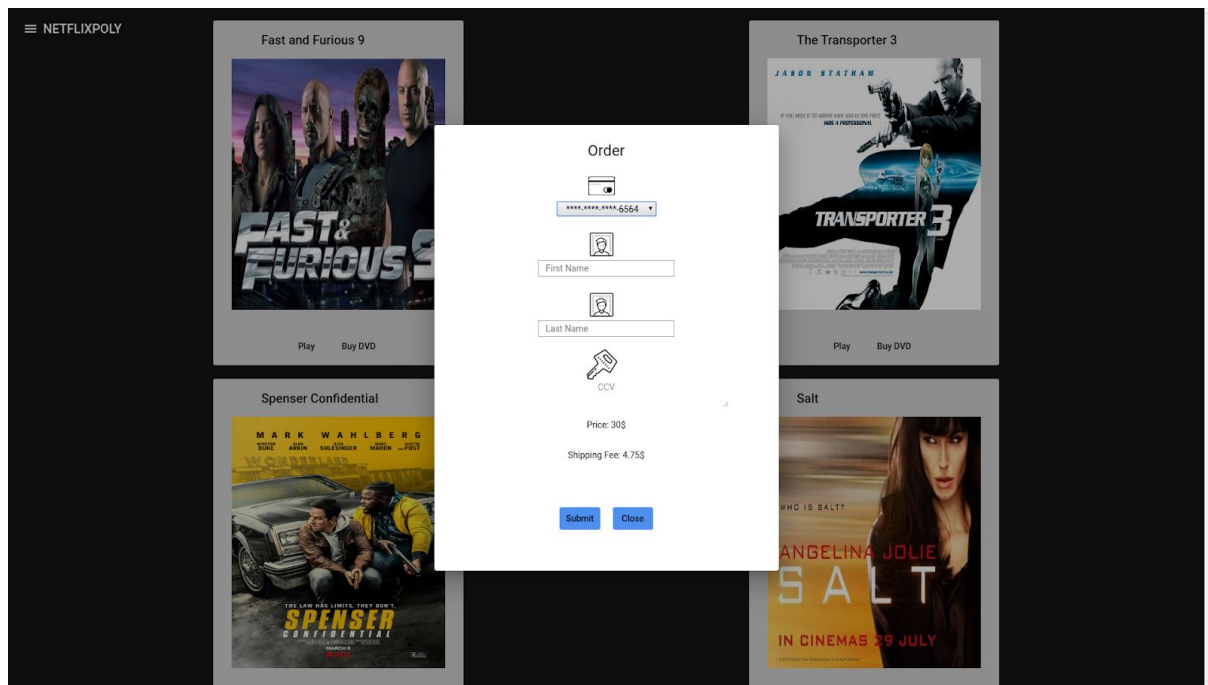
Capture d'écran 7: Affichage des informations concernant un film

On peut afficher les informations concernant un film c'est à dire les participant, le prix du dvd, la date de production, le prix pour le visionnage en ligne, la catégorie et le nom en appuyant sur la photo de couverture du film.



Capture d'écran 8: Jouer un film

On joue un film en appuyant sur le bouton play qui va ouvrir un modal de visionnement.



Capture d'écran 9: Menu pour acheter un dvd

Si on veut acheter un film en copie physique (DVD), on appuis sur le bouton buy DVD qui ouvre une page d'achat.

C'est cette même page qui est utilisée quand on essaye de jouer un film en diffusion en ligne mais qu'on ne l'a pas acheté encore.

Aspects Novateurs

Notre application est particulière dans le sens ou elle apporte plusieurs fonctionnalités dont

Raccourcis

- Control + m permet d'aller vers le formulaire d'ajout d'un film (Administrateur)
- Control + u permet d'aller vers le formulaire d'ajout d'un utilisateur (Administrateur)
- Control + h permet de revenir à la page d'accueil (Administrateur)
- + permet d'afficher plus de film dans la pagination de la table d'accueil des films déjà présents
- - permet d'afficher moins de film dans la pagination de la table d'accueil des films déjà présents

- La touche escape permet à tout moment de se déconnecter (Administrateur et Utilisateur)
- Control + F (dans la page home) permet d'aller vers la barre de filtrage des films (Administrateur et Utilisateur)

Sécurité

Notre application utilise des JWT qui sont générée avec un algorithme très sécuritaire et qui permettent une connection de maximum 2 heures, après laquelle l'utilisateur sera redirigé vers la page d'authentification s'il essaie de faire des manipulations sur l'application. En effet, chaque requête vers le serveur commence par valider le Token et sa validité, sans laquelle l'utilisateur n'a pas accès à la requête.

Dans le cadre de notre application, un film correspond à un trailer (≈3 min). Si nous avions intégré de vrais films (≈2 heures), nous aurions dû prolonger la validité du token pour améliorer l'expérience utilisateur.

Originalité

Notre application est différente des autres. En effet chez nous tout les films peuvent être visionnés, c'est plutôt TrailerPoly que NetflixPoly. On utilise l'API de youtube pour gérer les bandes d'annonce et continuer ou on s'était arrêté.

Guide d'installation

Base de données

Contrairement à ce que beaucoup on fait, nous n'avons pas de bouton create and populate DB, notre base de donnée est automatiquement créée quand le schéma n'existe pas et est peuplée en conséquent.

Pour ce il faut donc simplement, démarrer le serveur.

Sur certains systèmes (Linux) et dépendamment de l'utilisateur utilisée pour la base de donnée il faut parfois donner les privilèges requis, avec les commandes suivantes:

```
GRANT USAGE ON SCHEMA netflixpoly TO your_user;
```

```
GRANT SELECT ON ALL TABLES IN SCHEMA netflixpoly TO your_user;
```

```
GRANT UPDATE ON ALL TABLES IN SCHEMA netflixpoly TO your_user;
```

```
GRANT DELETE ON ALL TABLES IN SCHEMA netflixpoly TO your_user;
```

```
GRANT EXECUTE ON ALL FUNCTIONS IN SCHEMA netflixpoly TO your_user;
```

Serveur

1. npm install
2. npm start

Client

1. npm install
2. npm start

Proxy

C'est le serveur qui permet de traiter les requêtes vers google maps api, pour ne pas générer une erreur de cors. Ce serveur a une license qui donne tout les droit à toute personne détenant le liens.

1. npm install
2. npm start

Tests

Administrateur

admin@netflixpoly.com

password1

Subscribed User

email1[@gmail.com](mailto:email1@gmail.com)

password1

Pour achat de film : (numero de carte / cvc)

Jean Lebeau

Carte 1 : *5151909080876564 / 555*

Carte 2 : *348930913082441 / 874*

Pay Per View User

email5[@gmail.com](mailto:email5@gmail.com)

password5

Pour achat de film : (numero de carte / cvc)

Hugo Boss

Carte 1 : *5485667163502783 / 673*