

Exercice : Développement d'un clone YouTube avec Angular 19

Objectif

Développer une application web "MyYoutube" qui reproduit les fonctionnalités de base de YouTube en utilisant Angular 19. L'application devra stocker les informations utilisateurs et leurs playlists dans le localStorage du navigateur.

Contexte

Vous devez créer une application web qui permet aux utilisateurs de rechercher des vidéos YouTube, de créer un compte, de se connecter et de gérer leurs playlists personnelles.

Fonctionnalités requises

1. Interface utilisateur

- Créer une interface utilisateur responsive avec une barre de navigation contenant le logo "YouTube Clone"
- Implémenter un menu latéral gauche pour afficher les informations de l'utilisateur et ses playlists
- Créer un espace principal pour afficher le contenu (recherche, vidéos, formulaires)

2. Authentification

- Créer un système d'inscription avec les champs : nom d'utilisateur, email et mot de passe
- Implémenter un système de connexion avec nom d'utilisateur et mot de passe
- Ajouter des boutons "S'inscrire" et "Se connecter" dans l'en-tête
- Afficher "Se déconnecter" quand l'utilisateur est connecté

3. Recherche de vidéos

- Créer une page de recherche avec un champ texte et un bouton "Rechercher"
- Afficher les résultats de recherche avec une image miniature, un titre et un bouton "Ajouter" pour chaque vidéo
- Implémenter un indicateur de chargement pendant la recherche

4. Gestion des playlists

- Permettre à l'utilisateur de créer une playlist personnelle
- Afficher la liste des vidéos dans la playlist avec la possibilité de supprimer une vidéo (bouton "x")
- Afficher "Aucune vidéo dans votre playlist" lorsque la playlist est vide

5. Lecture de vidéos

- Créer une page de visionnage de vidéo affichant la vidéo sélectionnée
- Afficher le titre de la vidéo au-dessus du lecteur
- Afficher le nom de l'utilisateur dans le menu latéral

Exigences techniques

Stockage local

- Utiliser le localStorage pour stocker les informations des utilisateurs (noms, emails, mots de passe)
- Stocker les playlists des utilisateurs dans le localStorage
- Gérer l'état de connexion de l'utilisateur en session

Structure Angular

- Créer un minimum de 5 composants : Header, Sidebar, SearchPage, VideoPage, AuthPage
- Implémenter le routage Angular pour naviguer entre les différentes pages
- Utiliser des routes protégées pour les fonctionnalités nécessitant une authentification

Services

- Créer un service d'authentification pour gérer l'inscription, la connexion et la déconnexion
- Implémenter un service de playlist pour gérer l'ajout et la suppression des vidéos
- Développer un service de recherche pour récupérer les vidéos depuis une API externe ou simuler les résultats

Livrables attendus

1. Code source complet de l'application Angular 19
2. Documentation expliquant l'architecture et les choix techniques
3. Instructions d'installation et d'utilisation

Bonus

- Ajouter des fonctionnalités de tri et de filtrage des résultats de recherche
- Implémenter une pagination des résultats
- Ajouter des animations lors des transitions entre les pages
- Permettre la création de plusieurs playlists par utilisateur

Cet exercice vous permettra de mettre en pratique vos connaissances en Angular 19 tout en vous familiarisant avec la gestion du stockage local et la création d'une interface utilisateur dynamique.