

Projet type:Développement d'une application Angular – Gestion d'une Clinique

Contexte :

Développer une application web côté client en Angular permettant de gérer une clinique, avec 3 rôles distincts :

- Administrateur
- Médecin
- Patient

Chaque rôle dispose d'un dashboard spécifique après connexion et peut consulter, ajouter, modifier et supprimer des données selon ses droits.

Connexion à la base de données

L'application consommera une API simulée grâce à **JSON Server**, dont la base de données est définie dans le fichier db.json

Structure des pages principales

/login

/admin/dashboard (+ composant Patients + Médecins + Rendez-vous)

/medecin/dashboard (+ composant Rendez-vous médecin)

/patient/dashboard (+ composant Rendez-vous patient)

Structure des routes de l'application

Rôle	Route
Login commun	/login
Admin	/admin/dashboard
Médecin	/medecin/dashboard
Patient	/patient/dashboard

Structure des modèles (db.json)

Table : admins

Champ	Type
id	: string
nom	: string
email	: string
password	: string

Table : medecins

Champ	Type
id	: string
nom	: string
email	: string
password	: string
telephone	: string

Champ	Type
specialite	: string

Table : patients

Champ	Type
id	: string
nom	: string
email	: string
password	: string
telephone	: string

Table : rendezvous

Champ	Type
id	: string
date	: string
heure	: string
patientId	: string
medecinId	: string
etat	: string

Travail à réaliser et gestion du temps

1. Authentification

Créer une page /login permettant :

- De saisir email + mot de passe
- De vérifier les identifiants dans les collections admins, medecins et patients (appel API sur JSON Server)
- D'enregistrer dans localStorage :
 - id
 - rôle
- De rediriger vers le dashboard du rôle détecté :
 - Admin → /admin/dashboard
 - Médecin → /medecin/dashboard
 - Patient → /patient/dashboard

2. Dashboard Administrateur

Route : /admin/dashboard

L'admin peut gérer :

Les patients :

- Afficher la liste des patients (nom, email, téléphone)
- Bouton pour ajouter un patient via formulaire, modifier un patient existant et supprimer un patient

Les médecins :

- Afficher la liste des médecins (nom, email, téléphone, spécialité)
- Bouton pour ajouter un médecin via formulaire, modifier un médecin existant et supprimer un médecin

Les rendez-vous :

- Afficher la liste de tous les rendez-vous (date, heure, médecin, patient, état (en attente / confirmé / annulé))

3. Dashboard Médecin

Route : /medecin/dashboard

Le médecin peut consulter **uniquement ses rendez-vous (filtrés par medecinId)** :

- Date
- Heure
- Patient concerné
- État (en attente, confirmé, annulé)

4. Dashboard Patient

Route : /patient/dashboard

Le patient peut consulter **uniquement ses rendez-vous (filtrés par patientId)** :

- Date
- Heure
- Médecin concerné
- État (en attente, confirmé, annulé)