**Documentation du Projet Angular**

**Table des matières**

1. [Introduction](https://chat.openai.com/c/f2771c56-c09a-4e25-9824-c09e3a230428#introduction)
2. [Structure du Projet](https://chat.openai.com/c/f2771c56-c09a-4e25-9824-c09e3a230428#structure-du-projet)
3. [Composants](https://chat.openai.com/c/f2771c56-c09a-4e25-9824-c09e3a230428#composants)
   * [AppComponent](https://chat.openai.com/c/f2771c56-c09a-4e25-9824-c09e3a230428#appcomponent)
   * [PersonneComponent](https://chat.openai.com/c/f2771c56-c09a-4e25-9824-c09e3a230428#personnecomponent)
   * [PersonneFormComponent](https://chat.openai.com/c/f2771c56-c09a-4e25-9824-c09e3a230428#personneformcomponent)
   * [FootComponent](https://chat.openai.com/c/f2771c56-c09a-4e25-9824-c09e3a230428#footcomponent)
4. [Services](https://chat.openai.com/c/f2771c56-c09a-4e25-9824-c09e3a230428#services)
   * [PersonneService](https://chat.openai.com/c/f2771c56-c09a-4e25-9824-c09e3a230428#personneservice)
   * [AuthService](https://chat.openai.com/c/f2771c56-c09a-4e25-9824-c09e3a230428#authservice)
5. [Modèles](https://chat.openai.com/c/f2771c56-c09a-4e25-9824-c09e3a230428#modeles)
   * [Modèle Personne](https://chat.openai.com/c/f2771c56-c09a-4e25-9824-c09e3a230428#modele-personne)
6. [Routing](https://chat.openai.com/c/f2771c56-c09a-4e25-9824-c09e3a230428#routing)
7. [Styles](https://chat.openai.com/c/f2771c56-c09a-4e25-9824-c09e3a230428#styles)
8. [Utilisation](https://chat.openai.com/c/f2771c56-c09a-4e25-9824-c09e3a230428#utilisation)
9. [Conclusion](https://chat.openai.com/c/f2771c56-c09a-4e25-9824-c09e3a230428#conclusion)

**Introduction**

Ce projet Angular, appelé Training Studio, est conçu pour gérer les adhésions à la salle de sport et les programmes d'entraînement. L'application comprend des fonctionnalités pour afficher, ajouter, mettre à jour et supprimer des informations sur les membres.

**Structure du Projet**

Le projet est structuré avec les répertoires et fichiers clés suivants :

* **src/app** : Contient les composants Angular, les services, les modèles et d'autres fichiers liés à l'application.
* **src/assets** : Contient des ressources statiques telles que des images et des styles.
* **angular.json** : Fichier de configuration Angular.
* **tsconfig.json** : Fichier de configuration TypeScript.

**Composants**

**AppComponent**

Le **AppComponent** est le composant principal qui sert de point d'entrée de l'application. Il comprend la barre de navigation et le conteneur de contenu.

**Template :**

htmlCopy code

<!-- app.component.html --> <!-- Navigation principale et conteneur de contenu -->

**Logique :**

typescriptCopy code

// app.component.ts @Component({ selector: 'app-root', templateUrl: './app.component.html', styleUrls: ['./app.component.css'] }) export class AppComponent { // Logique du composant ici }

**PersonneComponent**

Le **PersonneComponent** gère l'affichage d'une liste de membres de la salle de sport.

**Template :**

htmlCopy code

<!-- personne.component.html --> <!-- Afficher la liste des membres de la salle de sport -->

**Logique :**

typescriptCopy code

// personne.component.ts @Component({ selector: 'app-personne', templateUrl: './personne.component.html', styleUrls: ['./personne.component.css'] }) export class PersonneComponent { // Logique du composant ici }

**PersonneFormComponent**

Le **PersonneFormComponent** gère le formulaire pour ajouter et mettre à jour les informations des membres.

**Template :**

htmlCopy code

<!-- personne-form.component.html --> <!-- Formulaire pour ajouter et mettre à jour les informations des membres -->

**Logique :**

typescriptCopy code

// personne-form.component.ts @Component({ selector: 'app-personne-form', templateUrl: './personne-form.component.html', styleUrls: ['./personne-form.component.css'] }) export class PersonneFormComponent { // Logique du composant ici }

**Services**

**PersonneService**

Le **PersonneService** gère les opérations CRUD liées aux membres de la salle de sport.

**Méthodes :**

* **getPersonne(): Observable<Personne[]>** : Récupérer une liste des membres de la salle de sport.
* **addPersonne(personne: Personne): Observable<Personne[]>** : Ajouter un nouveau membre de la salle de sport.
* **deletePersonne(id: number): Observable<Personne[]>** : Supprimer un membre de la salle de sport par ID.
* **updatePersonne(updatedPersonne: Personne): Observable<Personne>** : Mettre à jour les informations du membre de la salle de sport.
* **getPersonneById(id: number): Observable<Personne>** : Récupérer un membre de la salle de sport par ID.

**AuthService**

Le **AuthService** fournit des fonctionnalités d'authentification.

**Méthodes :**

* **login(username: string, password: string): boolean** : Authentifier l'utilisateur en fonction du nom d'utilisateur et du mot de passe.

**Modèles**

**Modèle Personne**

Le modèle **Personne** représente la structure des informations sur les membres de la salle de sport.

**Propriétés :**

* **id: number** : Identifiant unique du membre.
* **first\_name: string** : Prénom du membre.
* **last\_name: string** : Nom de famille du membre.
* **email: string** : Adresse e-mail du membre.
* **address: string** : Adresse du membre.
* **sport\_type: string** : Type de sport ou programme d'entraînement.

**Routing**

L'application utilise le routeur Angular pour la navigation. Les routes clés incluent :

* **/** : Page d'accueil avec des informations générales.
* **/list** : Affiche la liste des membres de la salle de sport.
* **/admin** : Tableau de bord admin pour la gestion de l'application.

**Styles**

La stylisation de l'application est gérée à l'aide de fichiers CSS, assurant une mise en page réactive et visuellement attrayante.

**Utilisation**

Pour exécuter le projet localement, suivez ces étapes :

1. Clonez le dépôt : **git clone <url-du-depot>**
2. Installez les dépendances : **npm install**
3. Lancez le serveur de développement : **ng serve**

**Conclusion**

Cette documentation offre un aperçu de la structure du projet Angular, des composants, des services et des instructions d'utilisation. Pour plus de détails, consultez des fichiers spécifiques et leur documentation.