

Chapitre 5 Projet Application fil rouge : app-music

Vous allez développer une petite application de musique type catalogue de musiques indépendantes avec un petit moteur de recherche.

On développera également un espace sécurisé pour administrer le catalogue en mettant en place un CRUD (create/read/update/delete).

Une fois terminé on verra comment déployer l'application sur un serveur distant.

Cette application sera notre fil rouge pour découvrir les concepts fondamentaux d'Angular.

Installation & configuration

Vous allez tout d'abord installer Angular dans le dossier app-music sur votre bureau. Pour créer un nouveau projet Angular vous allez d'abord récupérer sur votre machine le package node angular-cli, outils en ligne de commande qui nous aidera à créer les fichiers, compiler et déployer l'application :

```
npm install -g @angular/cli
```

Ou plus simplement

```
npm i -g @angular/cli
```

Puis, dans un terminal vous allez taper la ligne suivante cela va créer un dossier et télécharger les sources du framework Angular et de l'application.

Nous allons choisir une installation avec la gestion des scss et sans la gestion des tests unitaires pour l'instant :

```
ng new app-music --style=scss --skip-tests=true
```

Remarques si vous souhaitez faire une installation standard vous ne taperez pas les options de commande

```
ng new app-music
```

Angular-cli est un outils en ligne de commande, il vous permet de créer, builder et mettre en production votre application. C'est un outils indispensable.

Pour lancer l'application et builder les modifications en temps réelles tapez la ligne de commande suivante, Angular lancera un serveur de test sur le port 4200

commande standard

```
ng serve -o
```

commande équivalente option en changeant le port

```
ng serve -o --port 4300
```

Une fois cette commande lancée vous devriez avoir dans votre navigateur par défaut la page suivante qui s'affiche :



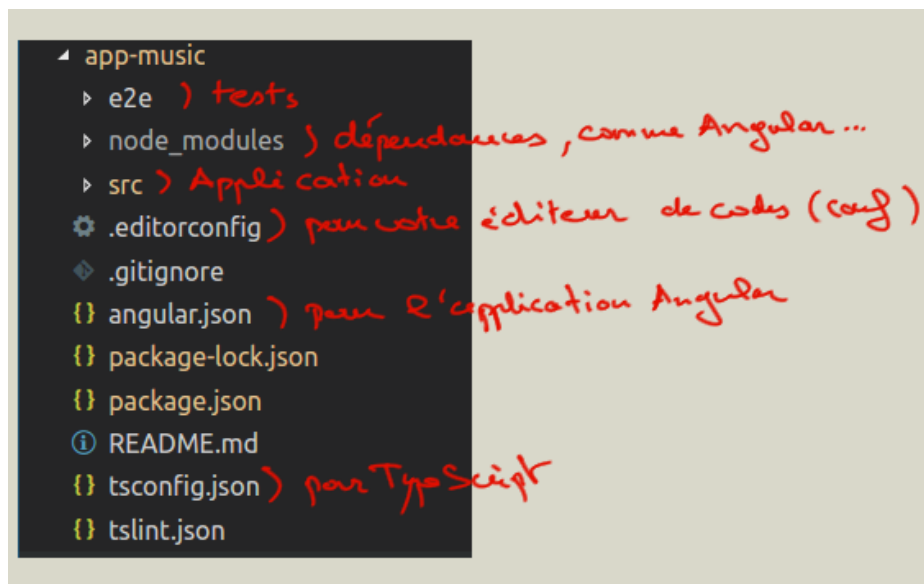
Welcome to app-music!



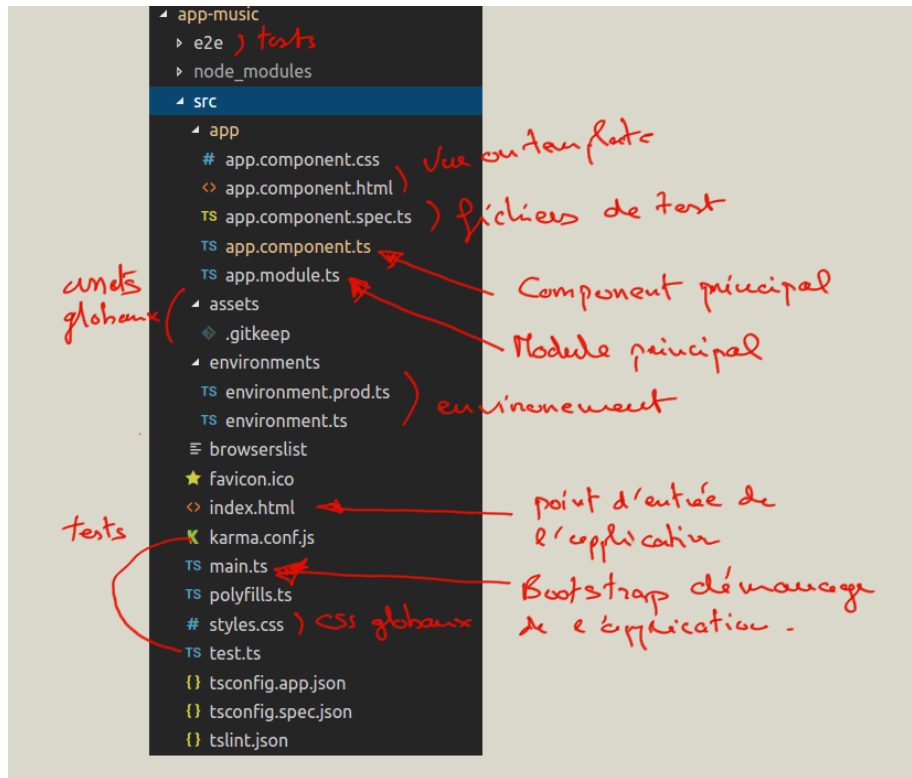
Here are some links to help you start:

- [Tour of Heroes](#)
- [CLI Documentation](#)
- [Angular blog](#)

Présentation des dossiers et fichiers Angular



Les dossiers de l'application src



Architecture de l'application

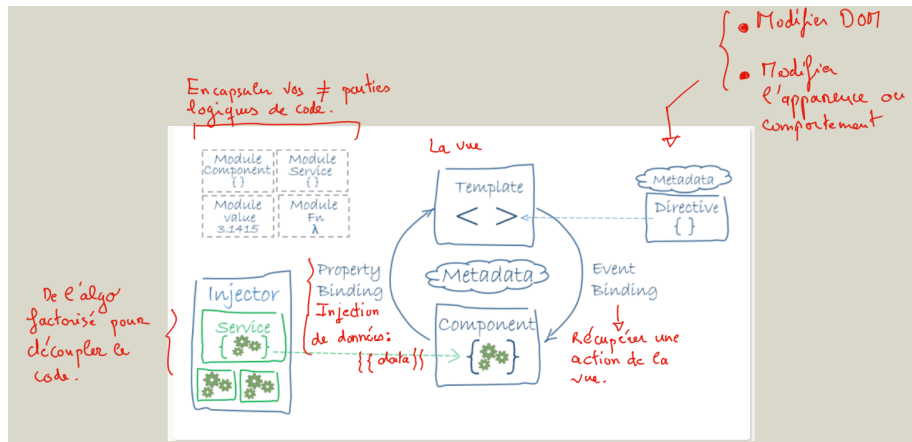
Angular est construit à partir de composants hiérarchiques écrit en TypeScript ce sont les « contrôleurs » de l'application. Chaque Component possède sa propre vue et ses propres CSS qui sont par défaut isolés (portées des CSS) dans le component lui-même.

Les modules encapsules (isolent) les Components dans des unités logiques de votre application : partie publique du site ou partie privée, page d'inscription, etc.

Les services séparent la logique du code des Components, ils sont factorisables et réutilisables dans l'application. On fera de l'injection de dépendance pour consommer un service unique dans les Components.

Les directives agissent sur les données dans les templates.

Les équipes de développement d'Angular propose un schéma détaillant cette architecture :



Installation des CSS ou SCSS

Pour l'instant on ne veut pas se soucier de la mise en page de notre application, nous installerons simplement le Bootstrap :

```
# installation local des CSS (dossier node_modules)
npm install bootstrap --save
```

Puis dans la partie (clé) architect du fichier angular.json du dossier app-music de l'application ajoutez la ligne suivante :

```
"styles": [
  "src/styles.scss",
  "../node_modules/bootstrap/dist/css/bootstrap.min.css"
],
"scripts": []
},
```

Relancez le serveur local de test pour tester, tapez en ligne de commande "Ctrl + c", n'oubliez pas également la commande suivante :

```
ng serve -o
```

Structure de la page principale de notre application

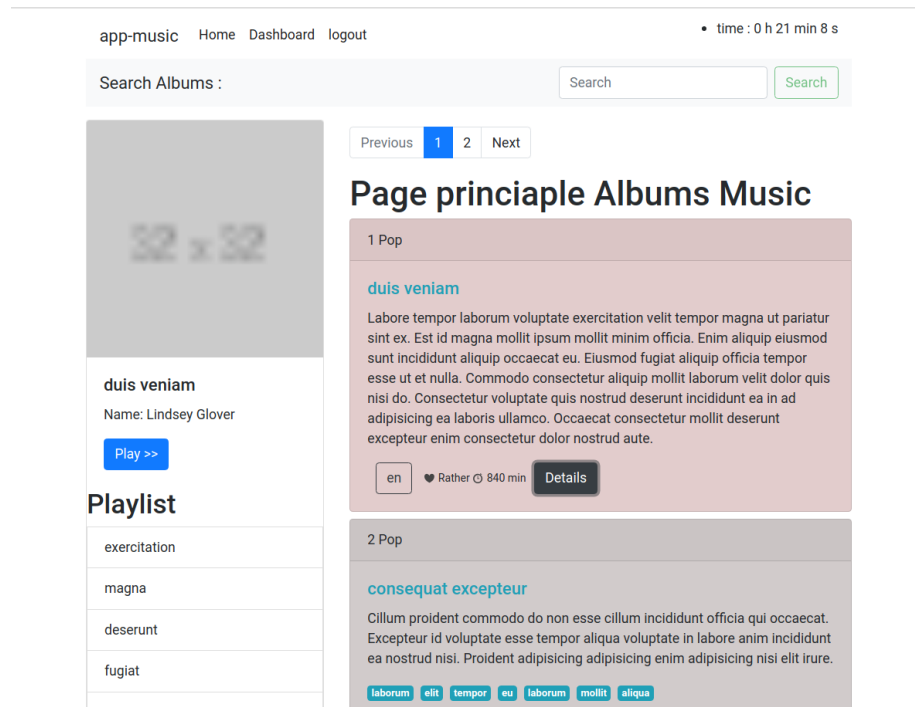
La page principale listera les albums avec la possibilité d'afficher le détail d'un album ou sa description plus détaillée.

Dans le menu principal on pourra aller sur la page de connexion et une fois authentifié on pourra aller sur le dashboard. Un lien de déconnexion “logout” sera utilisable pour se déconnecté.

Récupérez les sources du projet dans ce chapitre.

La partie HTML est à récupérer dans le fichier app.html, elle vous sert d'exemple nous l'utiliserons dans les différents composants qui structure notre application.

Vous l'utiliserez au fur et à mesure pour structurer la/les page(s) ne la recopiez pas directement dans le fichier app.component.html pour l'instant.



Fin de ce chapitre, merci de l'avoir suivi.