

Angular

Détail du cours :

- **Date:** 01/2023
- **Angular** CLI: 15.0.4
- **Node:** 18.12.1
- **Package Manager:** npm 8.19.2
- **OS:** win32 x64
- **Inspiration** : <https://www.youtube.com/watch?v=DTIYVffhJuU>

Un framework est un ensemble d'outil servant à reproduire rapidement des fonctionnalités que l'on utilise sur de nombreux projets différents. AngularJS (Angular jusqu'à la version 2) était développé sous Javascript. (plus soutenu) Angular (La version actuelle) est développé sous Typescript. Le principe d'Angular est de créer tout un tas de composant que l'on va assembler pour faire une SPA.

Installation de Angular

Pour utiliser Angular nous allons avoir besoin de NodeJS et NPM Comme beaucoup de projet JS.

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

On pourra vérifier si angular est bien installé avec:

```
ng version
```

Ensuite rendez vous bien à la racine de votre projet et entrer la commande :

```
ng new nomDuProjet
```

Il nous demandera ensuite si on souhaite un routeur, disons lui "Oui" Puis il nous demandera comment on souhaite gérer notre CSS, là c'est à votre choix. On se retrouve alors avec un dossier de plus de 40 000 fichiers.

Contenu d'un projet Angular

On y trouvera un dossier node_modules qui contient toute les dépendances utilisé par Angular. Un dossier src qui contiendra tous le code source de notre projet.

- Dans src on trouvera un dossier app qui sera 90% de notre travail avec Angular.
 - Dans app on trouvera :
 - app-routing.module.ts qui contiendra tout notre routing
 - app.component.html qui contient notre html par défaut.
 - app.component.scss (.css ou autre) qui contient le style de notre composant

- `app.component.spect.ts` qui sert au débogage, on aura pas à y toucher.
- `app.component.ts` qui gère le fonctionnement de base de notre composant.
- `app.module.ts` qui gère les modules utilisés sur notre projet. Il y a aussi un dossier `assets` qui contiendra toutes les assets (image, video...)

Puis un dossier `environments` qui contient nos variables d'environnements.

```
# Devenu optionnel depuis Angular CLI 15.1
ng generate environments
```

- `environment.ts` indique actuellement que notre projet n'est pas en production.
- `environment.prod.ts` indique qu'on est en production.

Le premier sera utilisé et on le remplacera par le second quand on sera en production.

- `.browserslistrc` n'est que pour gérer de la configuration avec les navigateurs.
- `.gitignore` qui permet de ne pas avoir 40 000 fichiers à envoyer sur notre projet.
- `angular.json` qui indique toute la configuration de notre projet Angular.
- `README.md` qui décrit notre projet et fait des rappels sur certaines commandes de Angular.
- Trois fichiers de configuration de Typescript

Si on ouvre notre fichier `"src/app/app.module.ts"` on trouvera différents imports

- `NgModule` permet le fonctionnement des modules Angular.
- `BrowserModule` permet le fonctionnement d'Angular.
- `AppRoutingModule` on l'a vu plus haut, contient nos routes.
- `AppComponent` est le composant racine de notre projet.
- Ensuite on utilisera `NgModule` dans un décorateur (les décorateurs sont représentés par un "@" et apporte des fonctionnalités à une classe.)
 - `declarations` indique les différents composants utilisés.
 - `imports` indique les différents modules utilisés.
 - `providers` permet d'injecter des dépendances.
 - `bootstrap` indique quel est le composant racine.

Peut être ajouter à la config ts:

```
"strictPropertyInitialization": false,
```

Pour lancer notre serveur de développement on utilisera la commande :

```
ng serve
```

Depuis le dossier de notre projet.

Par défaut on pourra retrouver notre site sur : "http://localhost:4200/"