esprit Se former autrement HONORIS UNITED UNIVERSITIES	Année Universitaire : 2025-2026 ANGULAR

Workshop n°1 :

Mise en place de l'environnement

Objectifs

- Installer l'environnement de travail nécessaire pour Angular 18.
- Créer un projet Angular version 18
- Comprendre les différentes configurations possibles d'un projet Angular.

I- Étapes d'installation de l'environnement

1- Téléchargez et installez nodejs depuis le site **:** **https://nodejs.org/en/.**
Node.js Minimum : v18.13.0
Node.js recommandé : **v20.x.x** (LTS — Long Term Support)
   - Pour vérifier que node est bien installé, tapez la commande **node –v** dans l'invite de commande.
   - Pour avoir une idée sur les versions compatibles vous pouvez consulter le lien **https://angular.io/guide/versions**
   - Le node package manager est installé automatiquement avec node. Vous pouvez vérifier la version installée de npm en tapant la commande **npm –v**

2- Installez l'outil CLI version 18 à l'aide de npm en tapant la commande suivante sur l'invite de commande **npm install –g @angular/cli@18**
RQ: Pour verifier la version Angular installée, tapez la commande **ng version**

3- Installez un éditeur/IDE qui vous convient soit Visual Studio Code ou bien Web Storm

4- Créez un répertoire dans lequel vous enregistrez vos projets Angular, par exemple Angular_Workspace.

II- Création d'un projet Angular 100% standAlone

1- Ouvrez la fenêtre cmd dans le répertoire Angular_Workspace et tapez la commande : **ng new projectName**

2- Répondez par CSS à la première question et « Yes » ou « No » à la deuxième question.

```
C:\WINDOWS\system32>ng new mon-projet
? Which stylesheet format would you like to use? CSS             [ https://developer.mozilla.org/docs/Web/CSS
          ]
? Do you want to enable Server-Side Rendering (SSR) and Static Site Generation (SSG/Prerendering)? (y/N) y
```

Figure 1 : Terminal Creation d'un projet Angular

3- Une fois la création est terminée, pour lancer votre projet, il faut accéder au projet crée via la commande **cd projectName** et puis taper la commande **ng serve (ou bien la commande ng serve –open)**

4- Une fois le projet est lancé, tapez l'url **localhost :4200**
Vous obtenez le résultat suivant :

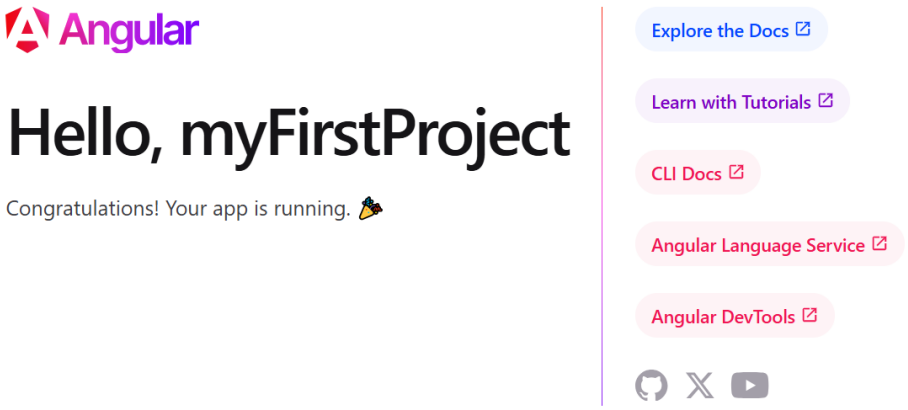

Figure 2 : page d'accueil d'un projet Angular

5- La structure du projet est la suivante:

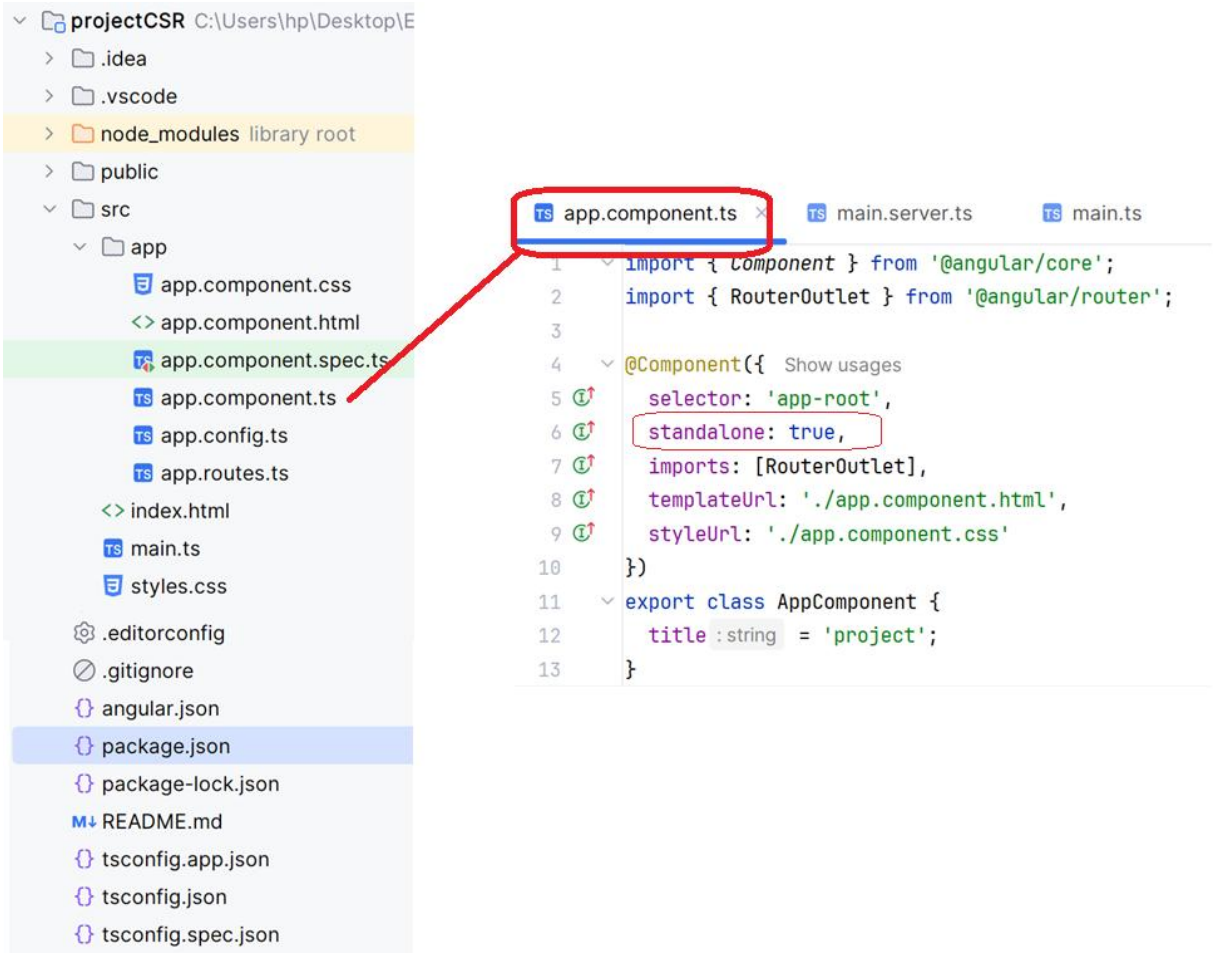

Figure 3- La structure d'un projet Angular 100% standalone

III- Création d'un projet Angular Modulaire

1. Ouvrez la fenêtre cmd dans le répertoire Angular_Workspace et tapez la commande : **ng new projectName --standalone=false**
2. Répondez par CSS à la première question et « Yes » ou « No » à la deuxième question (voir Figure 1).
3. Une fois la création est terminée, pour lancer votre projet, il faut accéder au projet crée via la commande **cd projectName** et puis taper la commande **ng serve (ou bien la commande ng serve –open)**

4. Une fois le projet est lancé, tapez l'url **localhost :4200** (voir figure 2).
5. La structure du projet est la suivante:

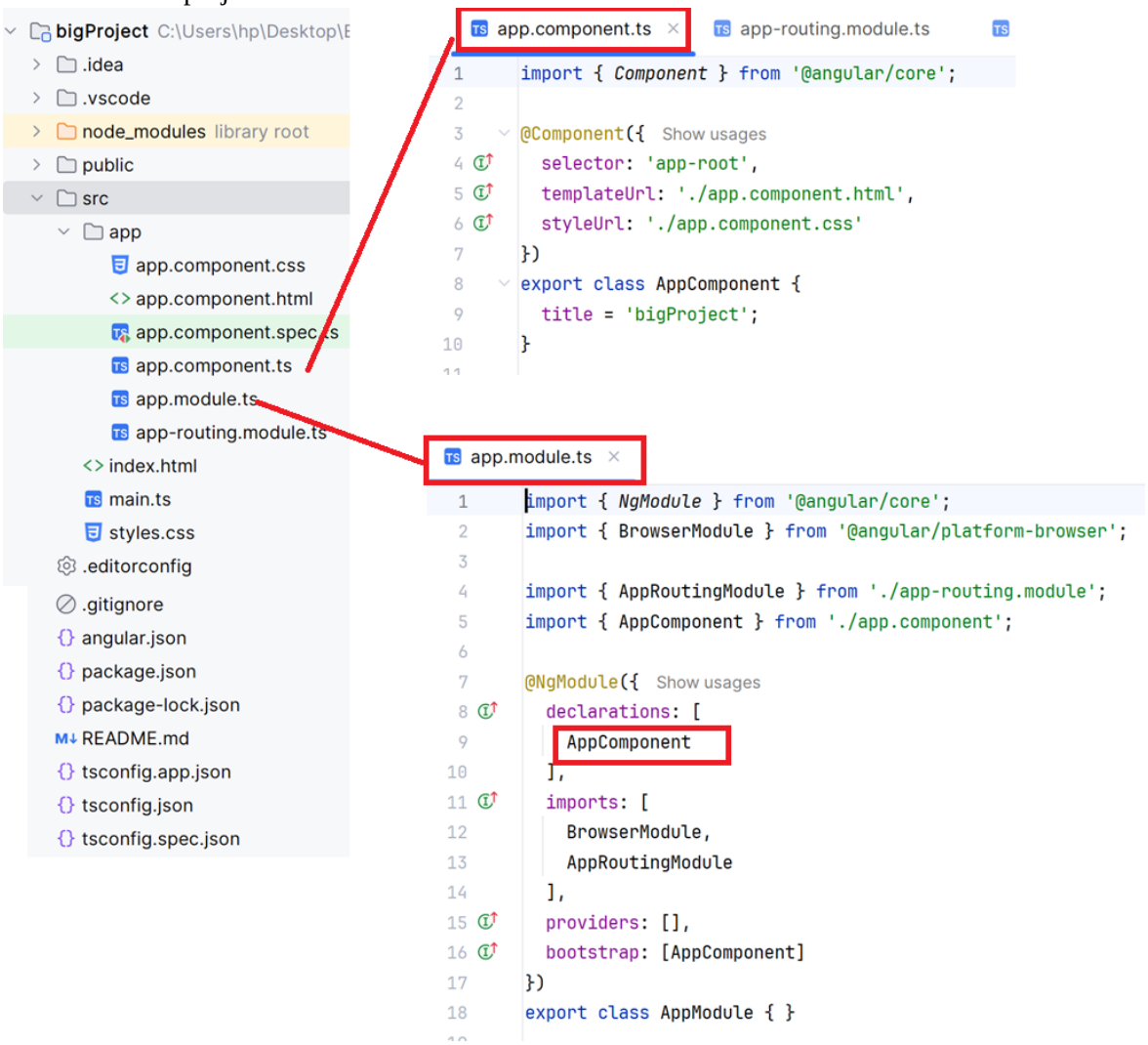

Figure 4- La structure d'un projet Angular Modulaire

IV- Standalone ou ngModule : Règle pratique pour choisir

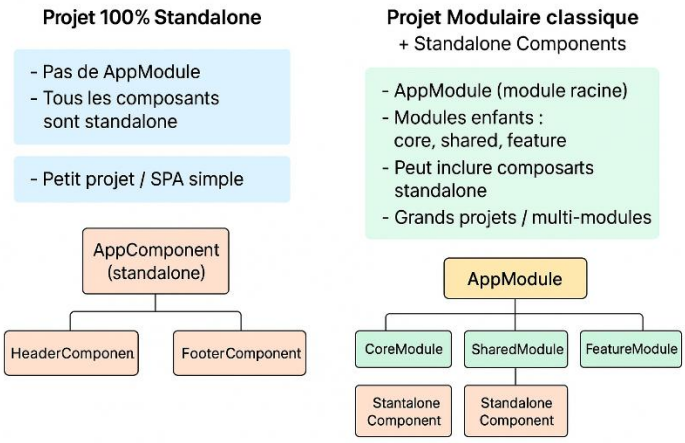

Figure 5 : les configurations possible d'un projet angular

Projet 100% Standalone

Caractéristique :

- Pas de AppModule du tout.
- Tous les composants sont standalone**.**
- Angular 14+ permet de créer des applications entièrement basées sur ces composants.

Quand l'utiliser :

Petits projets / prototypes / MVP : Rapidité de développement. Peu de modules à gérer.

Applications SPA simples : Peu de dépendances et d'interactions complexes.

Expérimentation avec la nouvelle architecture Angular : Tester les standalone components et voir les avantages du tree-shaking.

Avantages :

- Plus léger et moderne.
- Moins de fichiers NgModule à gérer.
- Intégration plus facile avec des librairies modernes et du lazy-loading sur composants standalone.

Limites :

- Pour de très grands projets, la gestion de dépendances et de routing complexe peut devenir plus difficile.

Projet modulaire classique (NgModule)

Caractéristique :

- Basé sur AppModule et d'autres modules.
- Peut inclure des composants standalone si besoin.

Quand l'utiliser :

Grands projets avec plusieurs équipes. Facile de découper en modules : core, shared, feature.

Applications complexes : Plusieurs routes, lazy loading, dépendances entre modules.

Projets qui évoluent dans le temps : Le découpage en modules facilite la maintenance et l'extension.

Mixte : possible d'utiliser des composants standalone à l'intérieur des modules pour profiter du tree-shaking et d'un lazy-loading granulaire.

Avantages :

- Architecture claire et maintenable.
- Bien adapté pour les **projets d'entreprise**.
- Compatible avec SSR, SSG et tous les outils Angular existants.

Limites :

- Un peu plus lourd à mettre en place.
- Plus de fichiers à gérer (NgModule).

Un résumé est donné dans le tableau suivant :

Type de projet	**Approche recommandée**
Petit projet / prototype rapide	100% standalone
Application SPA simple	Standalone ou modulaire légère
Grand projet / projet entreprise / multi-modules	Modulaire classique + possibilité de composants standalone
Besoin de SSR/SSG, routing complexe, lazy-loading	Modulaire + standalone components pour certaines parties

Résumé commandes :

Commande	Rôle	Quand l'utiliser
`node -v`	Vérifier la version de Node.js	Juste après l'installation de Node
`npm -v`	Vérifier la version de npm	Après l'installation de Node (npm vient avec Node)
`npm install -g @angular/cli@18`	Installer Angular CLI (v18) globalement	Une seule fois sur la machine (ou lors d'une mise à jour)
`ng version`	Vérifier les versions Angular/CLI/Node	Après l'installation de la CLI, et en cas de doute
`ng new <projectName>`	Créer un **nouveau projet Angular 100% standalone**	Au démarrage d'un projet
`ng new <projectName> --standalone=false`	Créer un **nouveau projet Angular modulaire**	Au démarrage d'un projet
`cd <projectName>`	Entrer dans le dossier du projet	Juste après la création du projet
`ng serve`	Lancer le serveur de dev	Pour démarrer l'app en local
`ng serve --open`	Lancer et **ouvrir automatiquement** le navigateur	Gain de temps au démarrage

Ressources sont disponibles ici: lien

https://github.com/badi3a/AngularTraining/tree/workshop-01-setup