

# Développement Web Front End : **Angular Framework**



**Nadhem Bel Hadj**

[nadhemb@yahoo.com](mailto:nadhemb@yahoo.com)

**ORACLE®**  
Certified Professional  
Oracle Database 10g  
Administrator

**ORACLE®**  
10g Certified Associate

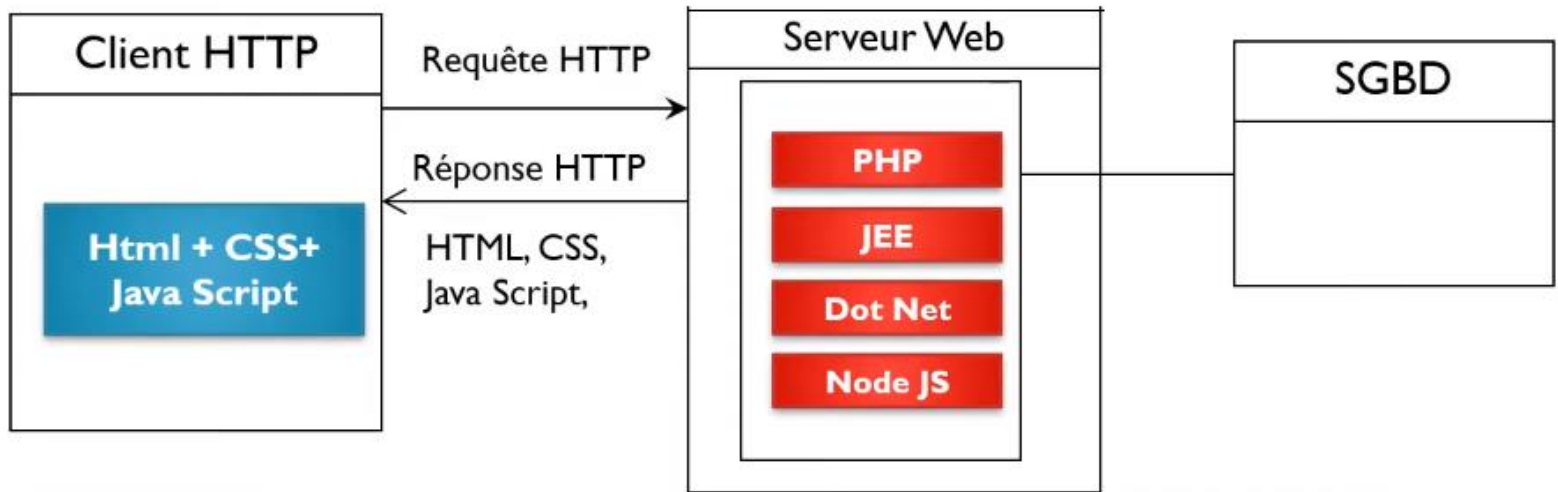
**ORACLE®**  
Certified Associate

**ORACLE®**  
Certified Associate  
Java SE 7 Programmer

# PLAN

- Présentation du Framework Angular,
- Installations de l'environnement de développement,
- Création d'un premier projet Angular,
- Structure d'un projet Angular

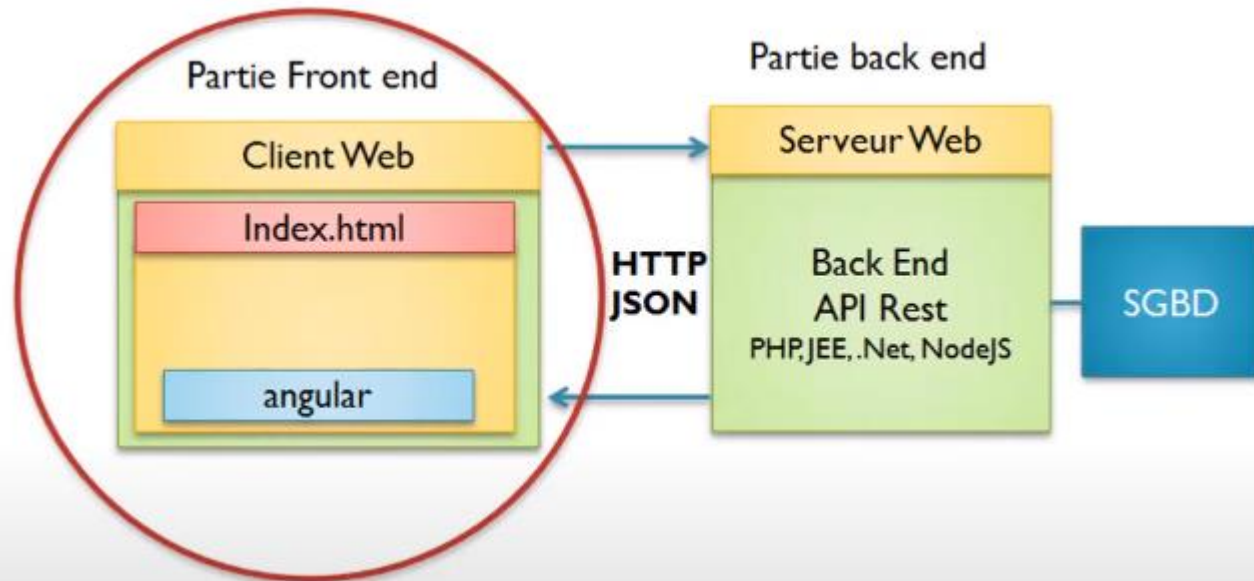
# Architecture Web



- Une application Web se compose de deux parties :
  - Le **Backend** : s'occupe des traitements coté serveur :
    - Technologies : PHP, JEE, .Net, Node JS
  - Le **Frontend** : s'occupe de la présentation coté client :
    - Technologies : HTML, CSS, Java Script

# Angular Framework

- Angular permet de créer la partie frontend des applications web de type SPA(Single Page Application),
- Une SPA contient une seule page HTML (index.html),
- Pour naviguer entre les différentes parties de l'application, JS est utilisé pour envoyer des requêtes http (AJAX) vers le serveur pour récupérer du contenu dynamique au format JSON,
- Ce contenu JSON sera affiché coté client, dans la même page, en utilisant HTML et CSS



# Angular Versions

- Angular 1 (**Angular JS**)
  - Première version de Angular,
  - Architecture MVC, langage utilisé Java Script,
- Angular 2+ (**Angular**)
  - Applications utilisent le langage Type script (langage OO), qui est compilé et traduit en JS avant d'être exécuté les navigateurs,
  - Angular 2+ : la programmation est basée sur les composant Web

# Angular : Installations

- Pour créer des applications Angular, il faut installer les outils suivants :
  - **Node JS** : <https://nodejs.org/en/download/>
  - Node JS installe l'exécutable **npm** (Node Package Manager) qui permet de télécharger et installer des bibliothèques JS,
- Installer ensuite Angular CLI (Command Line Interface) qui permet de créer, compiler, tester et déployer des projets angular : <https://cli.angular.io/>
  - **npm install -g @angular/cli@7**

# Création d'un Projet Angular

- Pour créer un nouveau projet Angular :
  - **ng** new MonProjet
- Cette commande génère les différents fichiers nécessaires à une application basique Angular et installe également les packages requis par le projet.

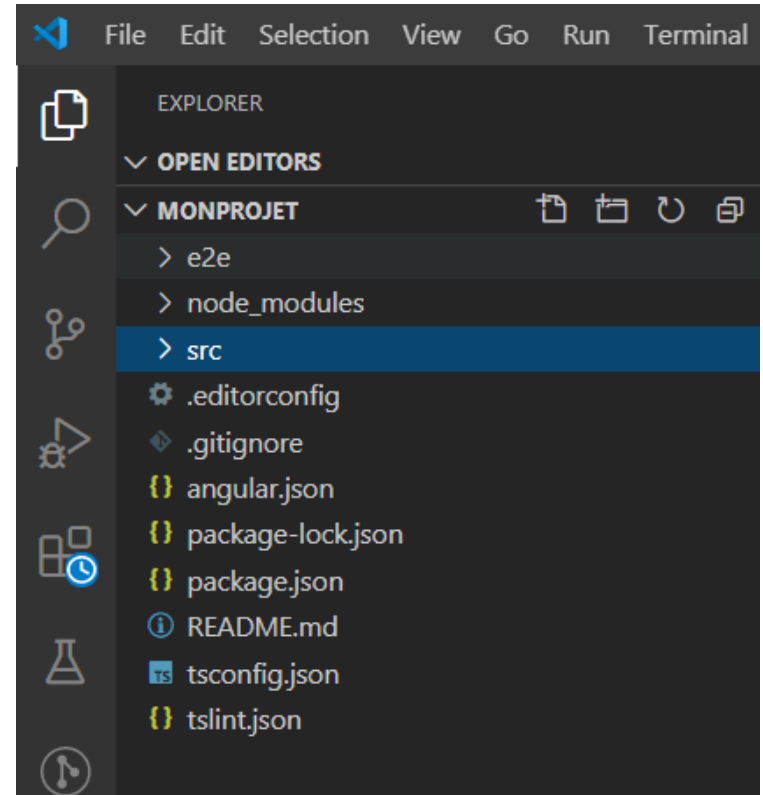
# Exécution d'un Projet Angular

- Pour exécuter un projet Angular, on accède au dossier du projet puis on lance :
  - **ng** serve
- Cette commande compile le code source du projet pour le convertir le code Type Script en JS en plus elle démarre un serveur Web local basé sur NodeJS pour déployer localement l'application,
- Pour tester le projet généré, il suffit de taper :  
`http://localhost:4200`

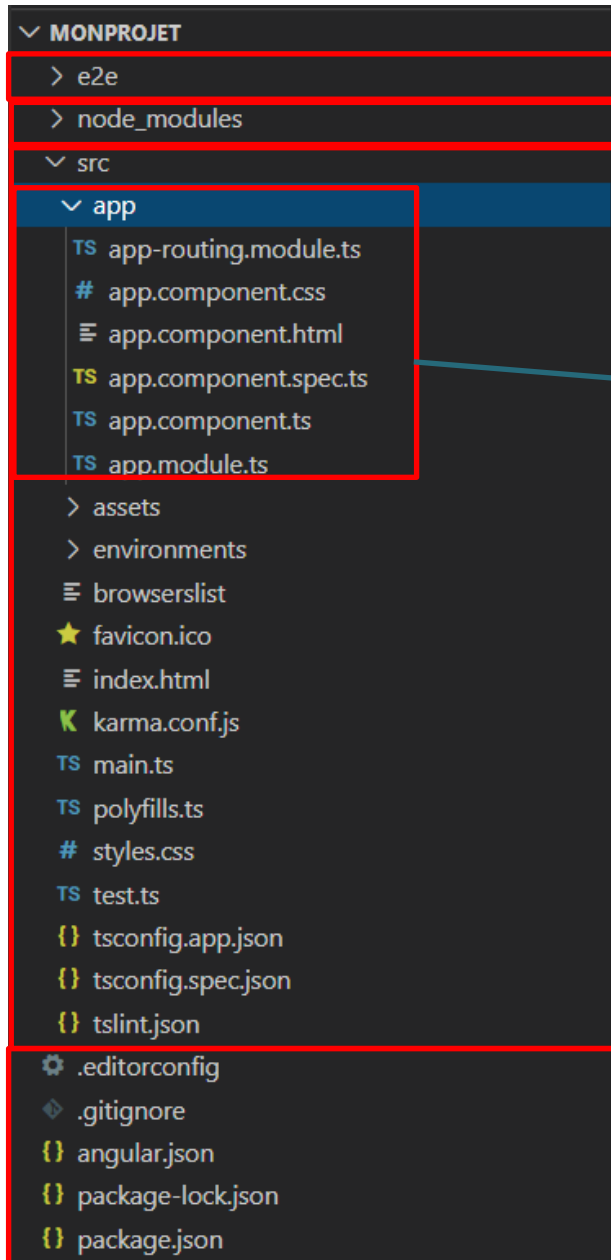


# Editer un Projet Angular

- Installer l'éditeur de code extensible « Microsoft Visual Studio Code » :  
<https://code.visualstudio.com/download>
- Pour éditer un projet Angular avec VS Code, on accède au dossier du projet puis on lance :
  - **Code .**
  - Ou File/Open Folder...



# Structure d'un projet Angular



Fichiers de Configuration du projet

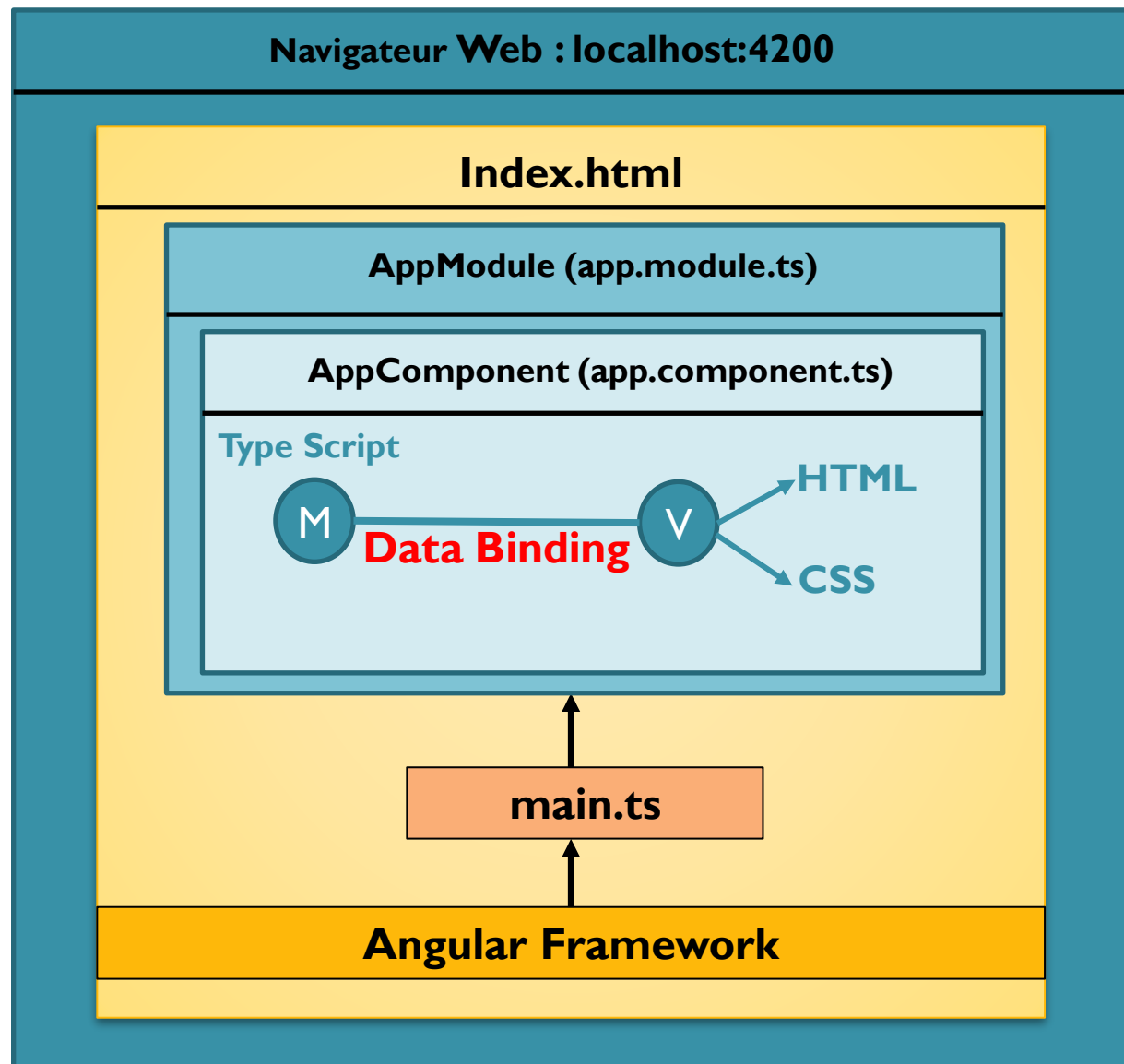
Dépendances externes du projet  
(Librairies JS et CSS)

Logique applicative du projet : Les  
composants, les services,...

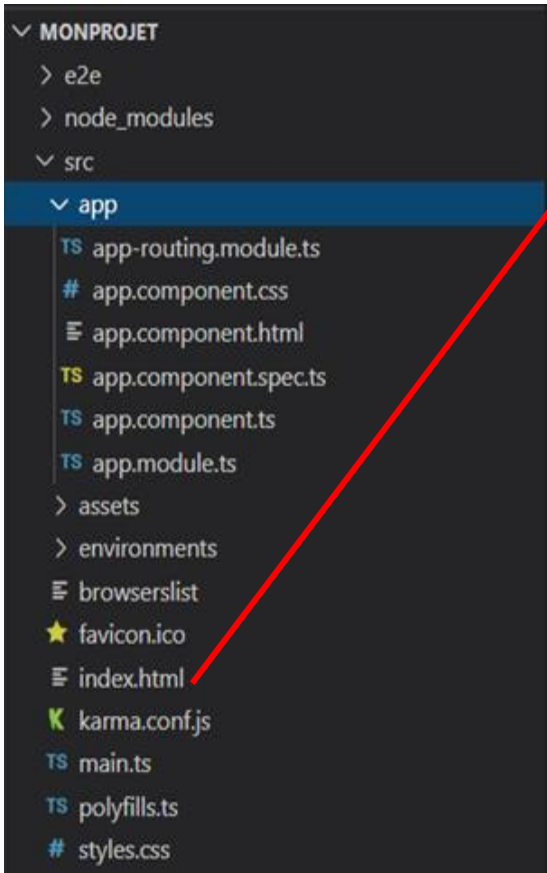
Le code source du projet

Fichiers de Configuration du projet

# Etapes d'exécutions



# Structure d'un projet Angular

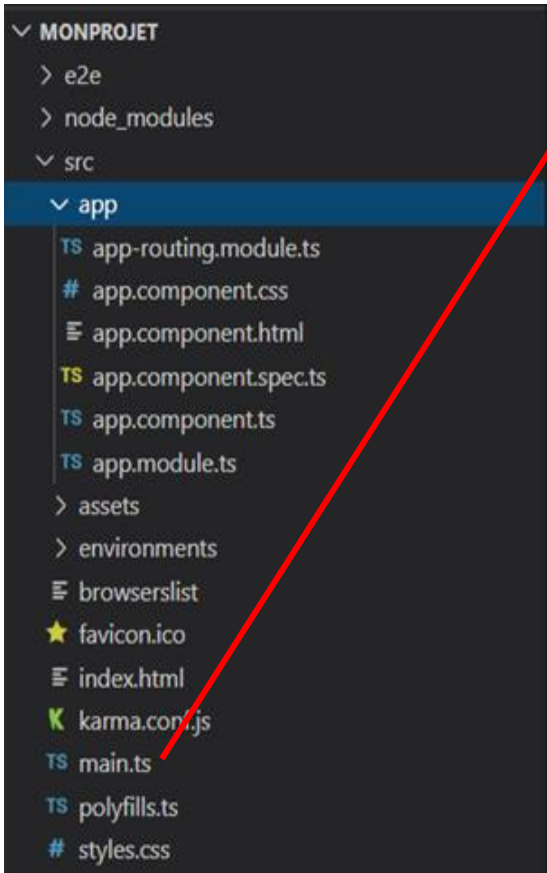


## Index.html

```
<!doctype html>
<html lang="en">
<head>
  <meta charset="utf-8">
  <title>MonProjet</title>
  <base href="/">

  <meta name="viewport" content="width=device-
width, initial-scale=1">
  <link rel="icon" type="image/x-
icon" href="favicon.ico">
</head>
<body>
  <app-root></app-root>
</body>
</html>
```

# Structure d'un projet Angular



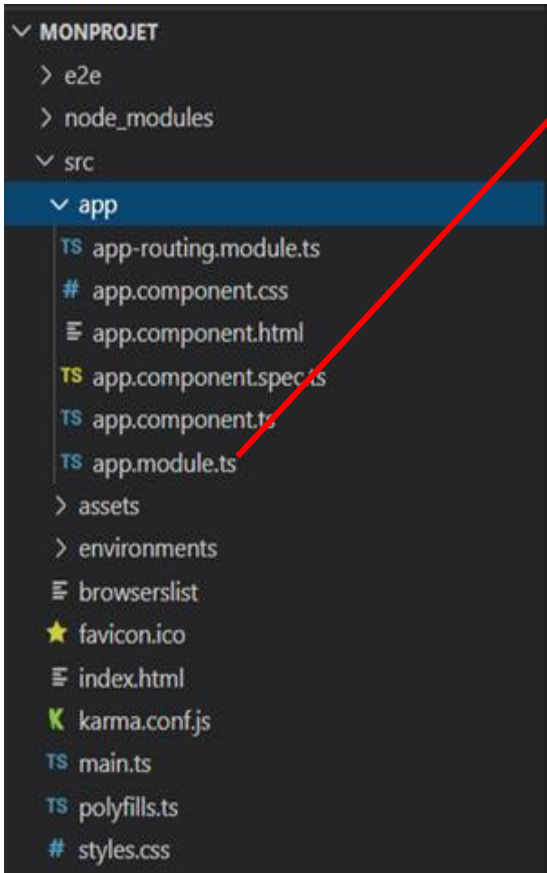
main.ts

```
import { enableProdMode } from '@angular/core';
import { platformBrowserDynamic } from
  '@angular/platform-browser-dynamic';
import { AppModule } from './app/app.module';
import { environment } from './environments/environment';

if (environment.production) {
  enableProdMode();
}

platformBrowserDynamic().bootstrapModule(AppModule)
  .catch(err => console.error(err));
```

# Structure d'un projet Angular

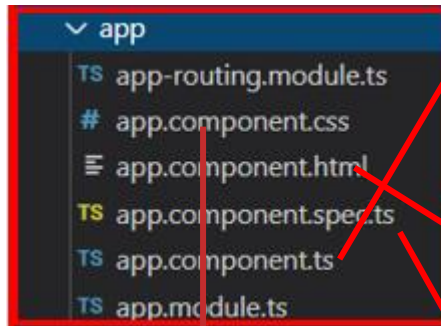
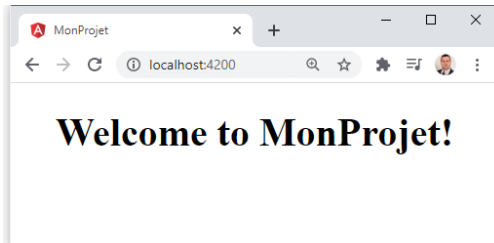


app.module.ts

```
import { BrowserModule } from
 '@angular/platform-browser';
import { NgModule } from '@angular/core';
import { AppRoutingModule } from
 './app-routing.module';
import { AppComponent } from './app.component';

@NgModule({
  declarations: [
    AppComponent
  ],
  imports: [
    BrowserModule,
    AppRoutingModule
  ],
  providers: [],
  bootstrap: [AppComponent]
})
export class AppModule { }
```

# Structure d'un projet Angular



## app.component.ts

```
import { Component } from '@angular/core';

@Component({
  selector: 'app-root',
  templateUrl: './app.component.html',
  styleUrls: ['./app.component.css']
})
export class AppComponent {
  title = 'MonProjet';
}
```

## app.component.html

```
<div style="text-align:center">
  <h1>
    Welcome to {{ title }}!
  </h1>
```

## app.component.ts.spec

Utilisé pour les tests unitaires avec le framework de test java script Jasmine (ng test)

## app.component.css

# Architecture de Angular

- Une application Angular se compose de :
  - un ou de plusieurs modules dont un est principal,
  - chaque module peut inclure :
    - Des **composant Web** : la partie visible de l'appli (IHM),
    - Des **services** qui contiennent la logique applicative. Les composants utilisent les services par le principe des « Injections des dépendances »,
    - Les **Pipes** pour formater l'affichage des données,
    - Les **Directives** permettant d'étendre la puissance de HTML, (Directives prédéfinie \*ngIf, \*ngFor...)

