

Ionic 4.0

Pré-requis pour le développement

1. Node JS : <https://nodejs.org/en/download/>
2. NPM : Package manager permettant d'installer les noeuds (outils / librairies) nécessaire à votre développement.

Dans la console, tapez : `npm install -g npm@latest`

Le -g permet d'installer globalement NPM sur votre machine

3. Ionic Cordova

Dans la console, tapez : `npm install -g ionic cordova`

La commande ionic est à présent disponible dans votre console.

Ionic Native

Depuis la version 4.0.0 du framework Ionic, Cordova ne doit plus être installé comme package à part.

Ionic Native fournit un ensemble de plugins natifs pour interagir avec les fonctions natives des smartphones. Ionic Native encapsule Cordova.

IDE

Pour vos projets Ionic, vous pouvez utiliser l'IDE de votre choix. Voici une liste non exhaustive :

- WebStorm (payant)
- Visual Studio Code
- Atom
- Sublime Text

Pour les développeurs C# un template Ionic existe pour Visual Studio. Il ne sera pas traité durant ce cours. <https://docs.microsoft.com/en-us/visualstudio/cross-platform/tools-for-cordova/first-steps/get-started-with-ionic2?view=toolsforcordova-2017>

Créer un nouveau projet Ionic

A l'endroit où vous souhaitez créer votre projet, exécuter la commande suivante :

```
ionic start mon-projet-ionic blank
```

Cette commande permet de créer un projet sans contenu pré-généré.

Il existe d'autres options pour générer un nouveau projet :

- **tabs** : crée un projet avec 3 tabs
- **sidemenu** : crée un projet avec un menu s'ouvrant sur le côté

Lancer le serveur de développement

Pour lancer le serveur de développement, entrez dans le créé pour votre projet et exécutez **ionic serve**.

```
cd mon-projet-ionic
```

```
ionic serve
```

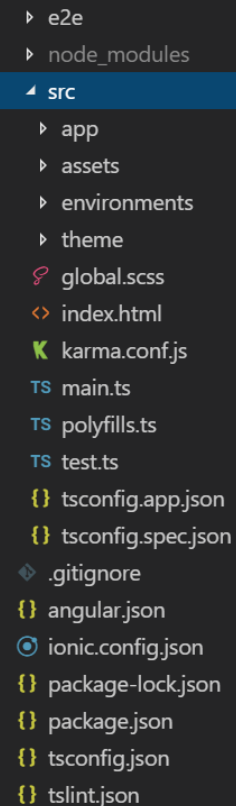
Note : **ionic serve** est l'équivalent de la commande **ng serve** pour Angular.

L'url pour naviguer sur l'application est la suivant : localhost:8100

Structure d'un projet Ionic

La structure d'un projet Ionic est adapté au développement d'application mobile.

Elle n'est donc pas tout à fait similaire à celle d'un projet web Angular.



```
▸ e2e
▸ node_modules
▾ src
  ▸ app
  ▸ assets
  ▸ environments
  ▸ theme
  🔗 global.scss
  <> index.html
  📄 karma.conf.js
  📄 main.ts
  📄 polyfills.ts
  📄 test.ts
  {} tsconfig.app.json
  {} tsconfig.spec.json
  💎 .gitignore
  {} angular.json
  🕒 ionic.config.json
  {} package-lock.json
  {} package.json
  {} tsconfig.json
  {} tslint.json
```

Détail ?

Structure d'un projet Ionic

La majorité de l'application sera développée dans le répertoire **src/app/**

Ce répertoire contient les éléments tels que le fichier **index.html**, les fichiers de configuration pour les tests, un dossier d'**assets** pour les images.

```
└─ src/  
  └─ app/  
  └─ assets/  
  └─ environments/  
  └─ theme/  
    global.scss  
    index.html  
    karma.conf.js  
    main.ts  
    polyfills.ts  
    test.ts  
    tsconfig.app.json  
    tsconfig.spec.json
```

Structure d'un projet Ionic

Le répertoire `src/app/` contient le composant et le module racine de l'application.

C'est également ici que nous trouverons les répertoires des différentes fonctionnalités de l'application (pages, composants, services).

```
▲ src /  
  ▲ app /  
    app-routage.module.ts  
    app.component.html  
    app.component.spec.ts  
    app.component.ts  
    app.module.ts
```

Références

- La documentation officielle : <https://ionicframework.com/docs>
- Tutoriels vidéos : <https://app.pluralsight.com/library/courses/ionic-cli/table-of-contents>
<https://app.pluralsight.com/library/courses/ionic2-angular2-typescript-mobile-apps/table-of-contents>
<https://app.pluralsight.com/library/courses/building-mobile-apps-ionic-framework-angularjs/table-of-contents>
-