Développement Mobile IONIC

Sequence 1

Introduction

- Deux approches pour le développement des applications:
  - **Développement mobile natif:**
    - Android: Java, Android SDK, Android Studio, Google Play
    - iOS: Objective C, iOS SDK, Xcode, Apple App Store
    - Windows Phone: C#, Windows Phone SDK, Visual Studio, Microsoft Marketplace



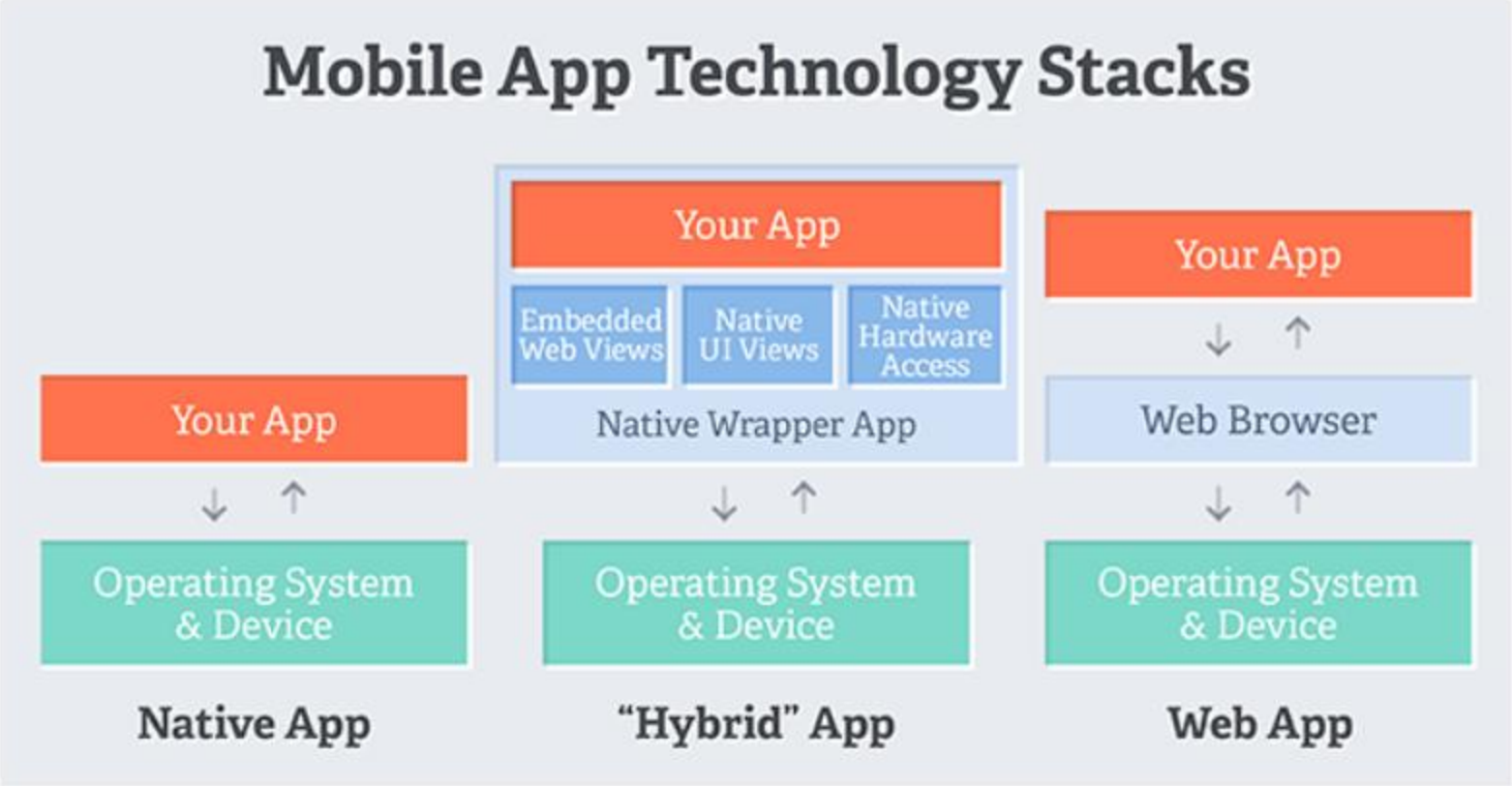
  - **Développement mobile hybride (Cross-platform)**
    - Partie Web: HTML, CSS, JavaScript
    - Partie Native: Apache Cordova
    - Plateformes: Adobe PhoneGap, IONIC, React Native

Approche des plateformes

Application par platform

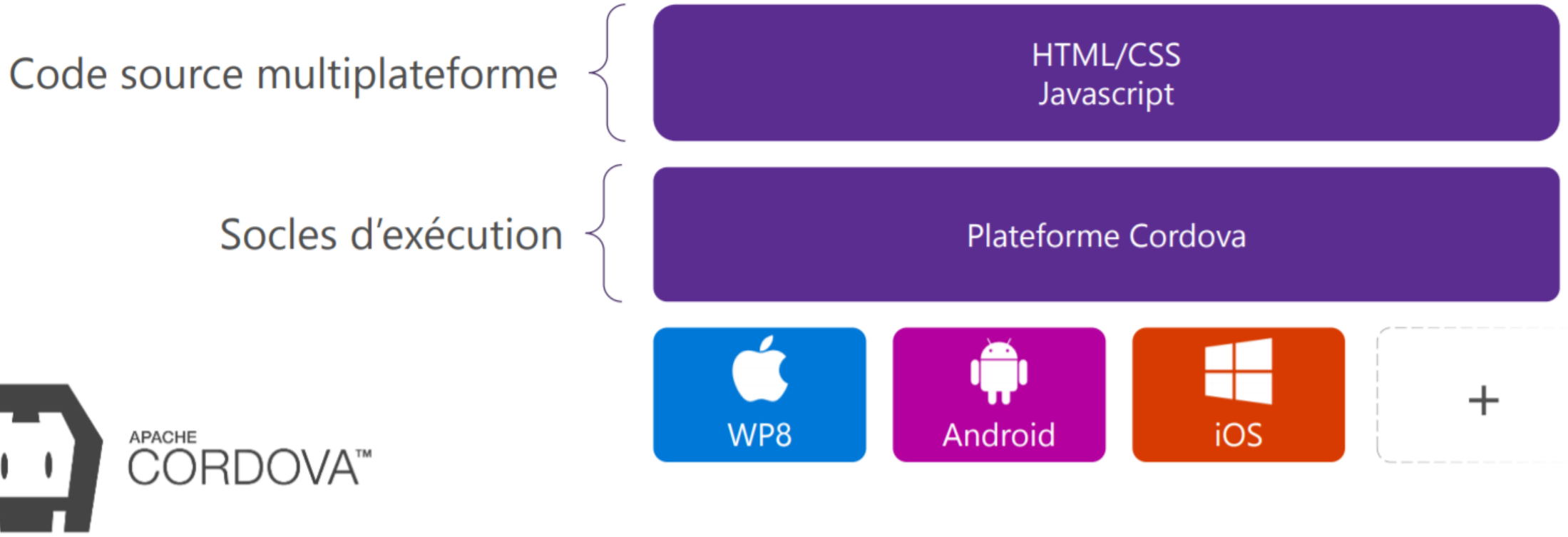


Architecture



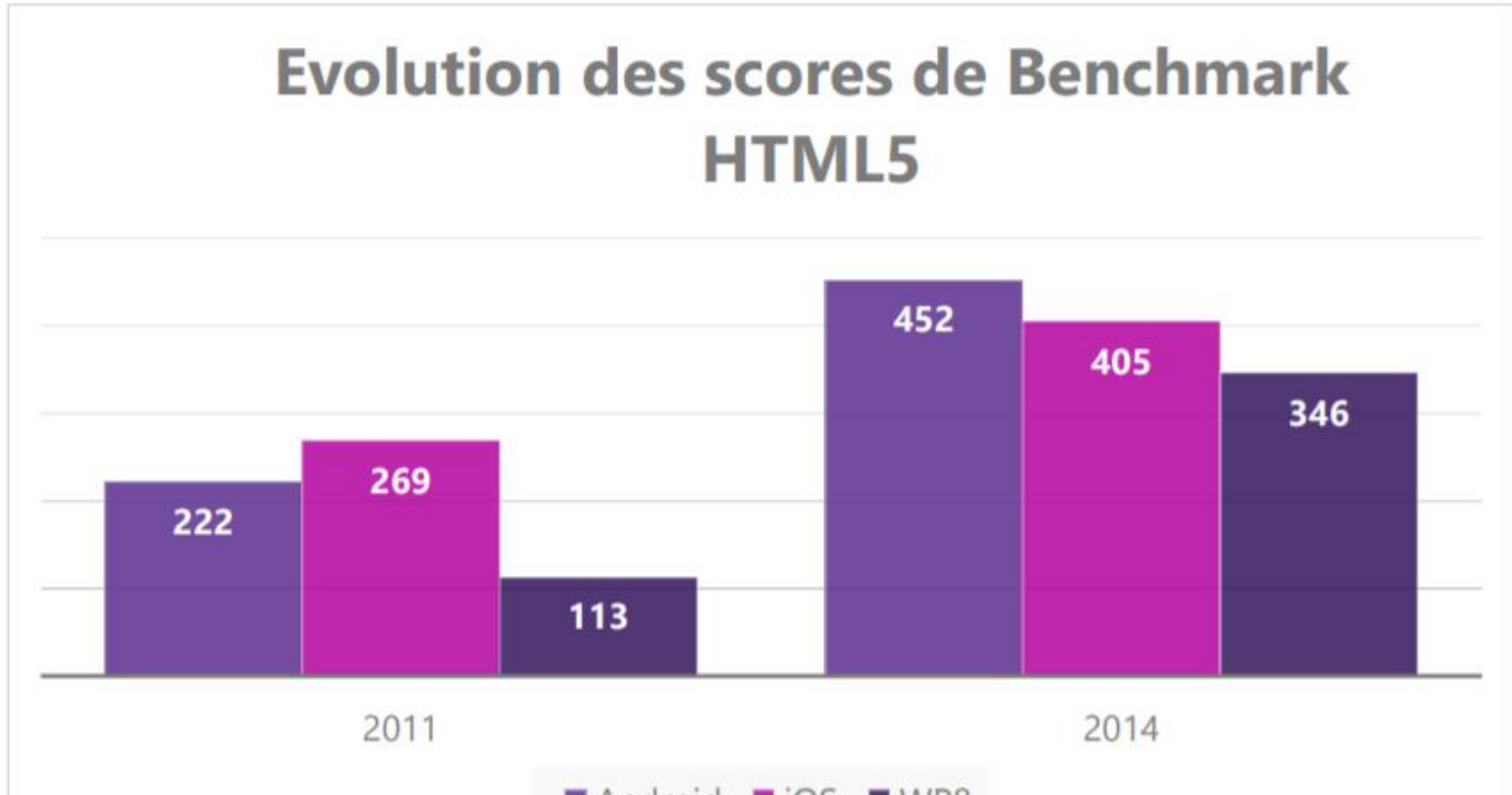
Source: https://hackernoon.com/getting-started-with-ionic-framework-an-overview-6725b687779b

Approche hybride(Cordova)



Performances et support HTML5 en hausse

iOS WKWebView ~+20% de performance
Android 4.4+ intègre Chromium 30



x2

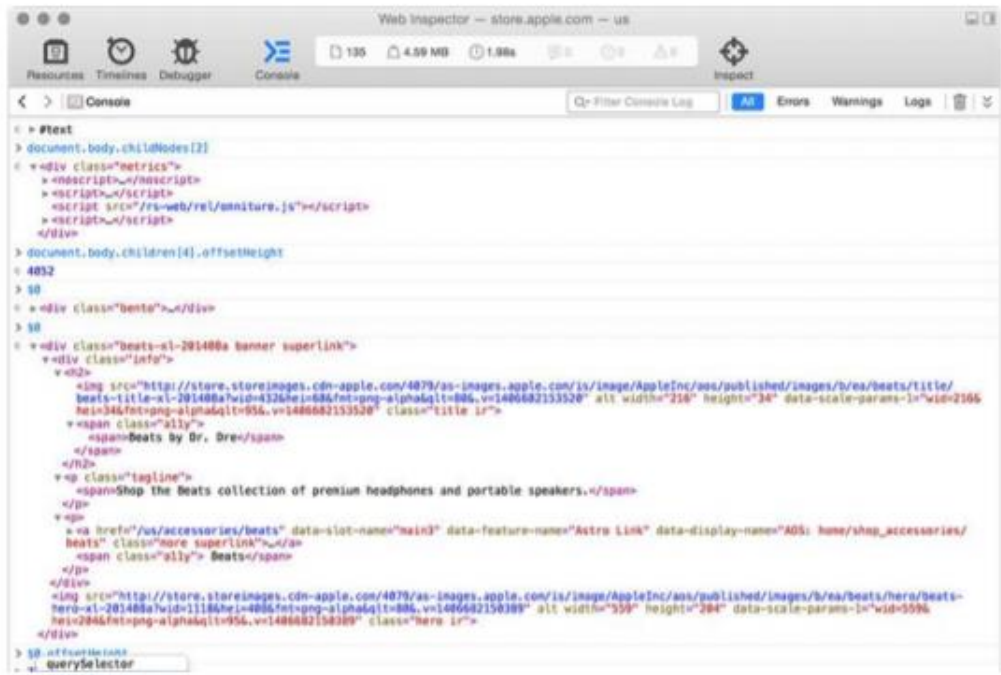
Source: html5test.com

Un outillage orienté mobile

- Framework Web Mobile First
  - Ionic (mobile only!), Bootstrap 3.0+, etc.
  - AngularJS 2.0

- Environnement de développement
  - Web Inspector (Chrome, Safari)
  - Emulateurs (Web, VM)





Application Hybride

Une application est dite ***hybride*** si elle est développée pour les smartphones avec les outils classiques du web.



Cette approche de développement d'application mobile est à la portée des développeurs du web.

Les smartphones Android, iOS ou autre, embarquent dans leur noyau une technologie appellée **WebView** permettant d'executer du code web au sein d'un environnement natif. Ainsi, il est possible de lancer la caméra de votre téléphone (natif), via une simple fonction Javascript (web). Ce qui donnerait ceci par exemple :

Application Hybride

Une application ***hybride*** peut être considérée comme une application native même si en matière de performances elle peut être limitée.



Elle est disponible au niveau des appStrore et téléchargeable

Une application hybride s'exécute dans une sorte de navigateur en mode plein écran, appelé ***WebView***, invisible pour l'utilisateur. Grâce à des plugins natifs personnalisables, elle peut accéder aux fonctionnalités natives d'un périphérique mobile (caméra, contacts, agenda,...), sans que le code saisi par le développeur ne soit directement lié à l'appareil.

Cela signifie que les applications hybrides peuvent fonctionner sur n'importe quelle plate-forme ou périphérique, tout cela à partir d'une base de code unique, tout en offrant une convivialité et un aspect natifs.

Avantages de développement Application Hybride (IONIC)



- Ionic est un Framework mobile open source évolutive qui est actuellement à la version 4.
- Une large communauté grâce à sa facilité de prise en main et une technologie très utilisée pour le développement.
- La puissance de IONIC se base sur les plugins d'apache cardova, de l'Angular de Google, NodeJS, et d'autres fonctionnalités.
- Coder une seule fois et déployer sur plusieurs platefomes





Installation Ionic

- Prérequis: NodeJS
  - NodeJS permet d'executer du code écrit en JavaScript, aussi bien sur un navigateur (côté client), que côté serveur, tout comme des langages comme le Python ou encore le PHP.
  - Avec NodeJS, installer le gestionnaire des packages (NPM) qui permet d'installer les modules développés par les membres de sa communauté
  - Aller sur le site : https://nodejs.org/en/download/ et télécharger NodeJS. Puis installer NodeJS.
  - Télécharger aussi et installer visual code : https://code.visualstudio.com/

Installation Ionic

- Ouvrer votre terminal et lancer la commande :
  - ***npm install –g @ionic/cli***

- Le paramètre « -g » permet de faire une installation globale des outils. Ainsi, vous pourrez utiliser la commande « ionic » et « cordova » dans n’importe quel répertoire.

Cordova et Ionic

- Après avoir installé NodeJS et le gestion de package « ***npm*** », Il faut ouvrir l'invite de commande et taper la commande suivante:
  - "***npm install -g ionic cordova***"
- Cette commande permet d'installer Cordova.
- C'est un outil qui permet de créer des applications mobiles multi-plateformes base sur les langages de (CSS3, JavaScript et HTML5).
- API spécifiques (natives) telles que celles d'Android, iOS ou Windows Phone.
- Le lien entre Cordova et Ionic:
  - **Phonegap** est une version de Cordova mais c'est un produit Adobe.
  - **Ionic Framework** est un ensemble de classes css et une bibliothèque de directives et de modules Javascript, construits au-dessus de Cordova , avec AngularJS.

Création Projet Ionic

- Ouvrir Visual Studio code
- Au niveau du menu en haut cliquer sur File→ Ouvrir un Folder
- Ensuite au niveau du menu en haut clique sur terminal
- Puis taper la commande suivante:

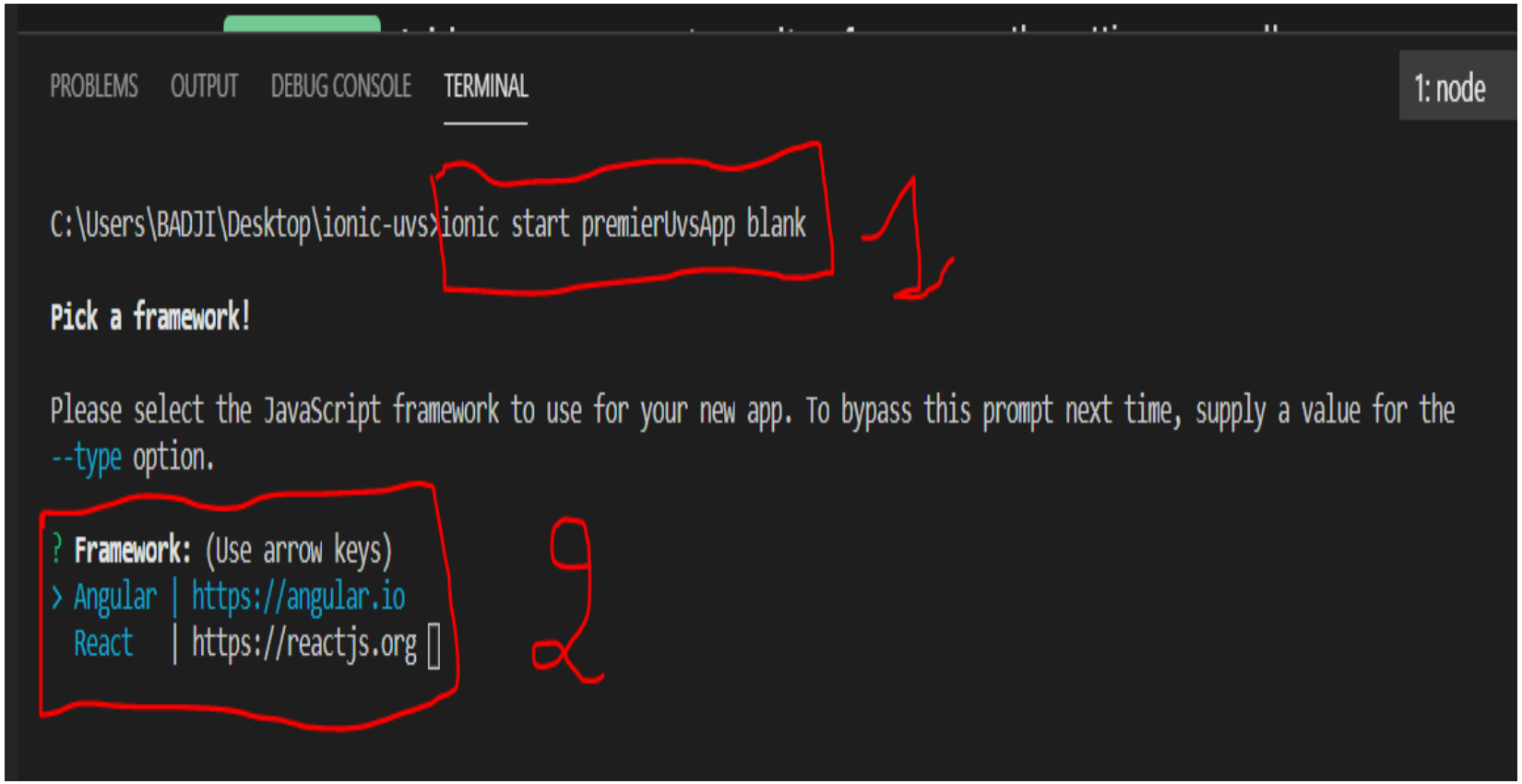
```
ionic start --list
```

- La syntaxe générique de création d'une application:

```
ionic start [<name>] [<template>]
```

- Lancer la commande :
  - ***Ionic start premierAppUvs blank***

Démarrer projet Ionic



1. Initialisation du projet Ionic
2. Choix sur la technologie à utiliser. Dans ce cours nous allons nous baser sous Angular.

Lancer Ionic Projet

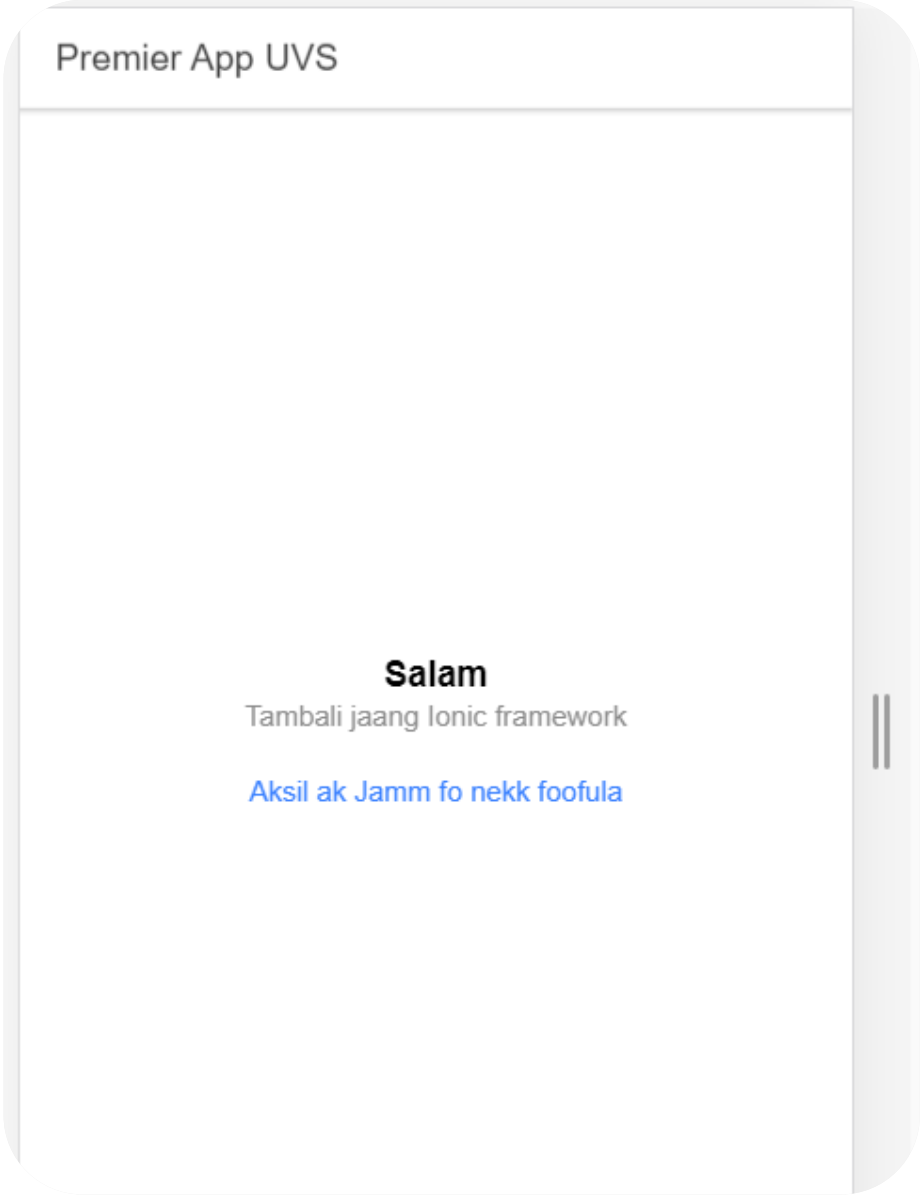
1. Il faut se pointer d'abord sur le répertoire du projet :
   - *cd premierAppUVS*
2. Puis lancer le projet grâce à la commande :
   - *Ionic serve*

```
[INFO] Next Steps:

      - Go to your newly created project: cd .\premierAppUVS
      - Run ionic serve within the app directory to see your app
      - Build features and components: https://ion.link/scaffolding-docs
      - Run your app on a hardware or virtual device: https://ion.link/running-docs
```

Après avoir lancer le projet,
l'adresse suivante est affiché
au niveau de votre navigateur

http://localhost:8100/home



Structure Projet

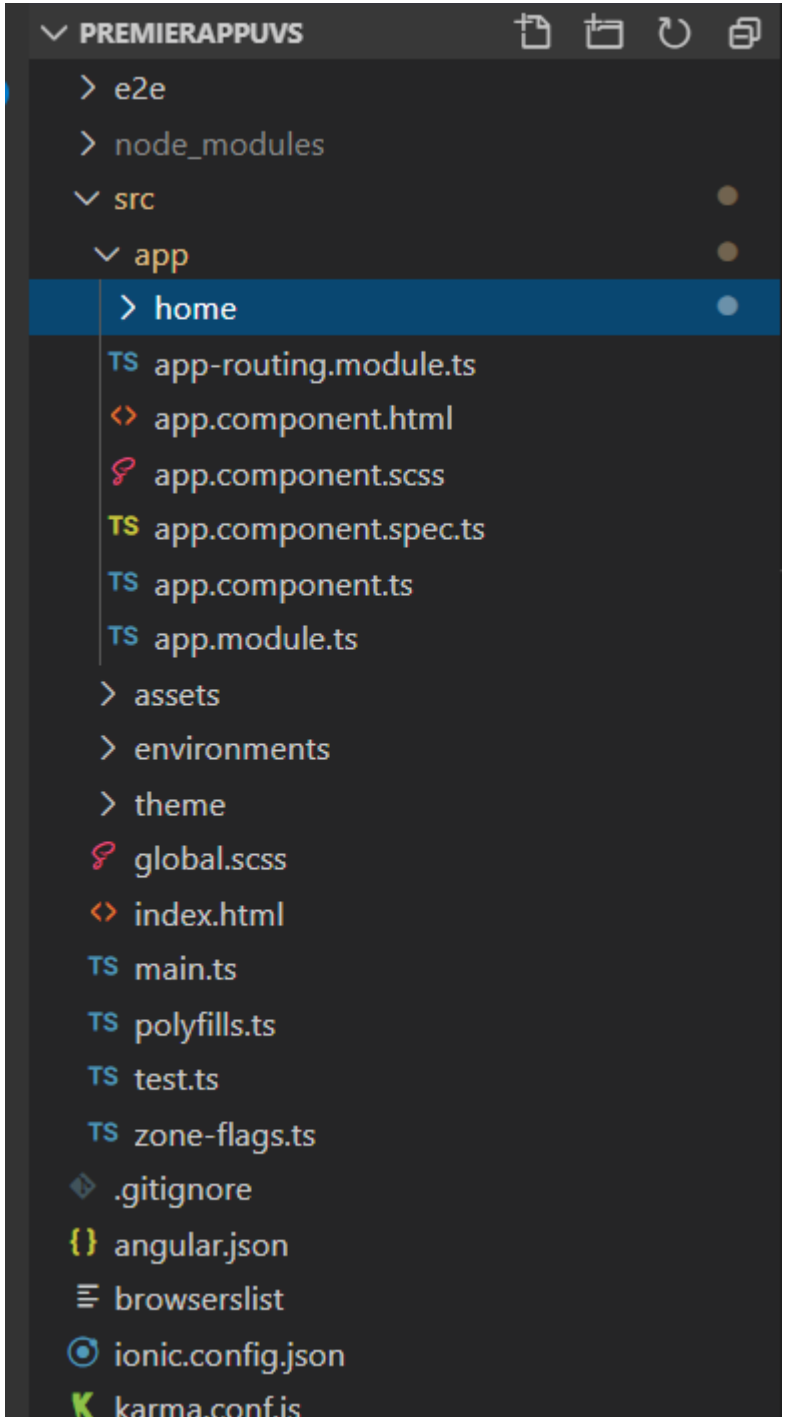
Le dossier ***node_modules*** contient les modules ***Angular, Ionic et Cordova*** intallés

Le dossier ***plateforms*** est pour le déploiement final de votre application. Il contient les plateformes sur lesquelles tu veux deployer ton application.

Le dossier ***plugins*** contient les plugins cordova pour accéder aux ressources des appareils

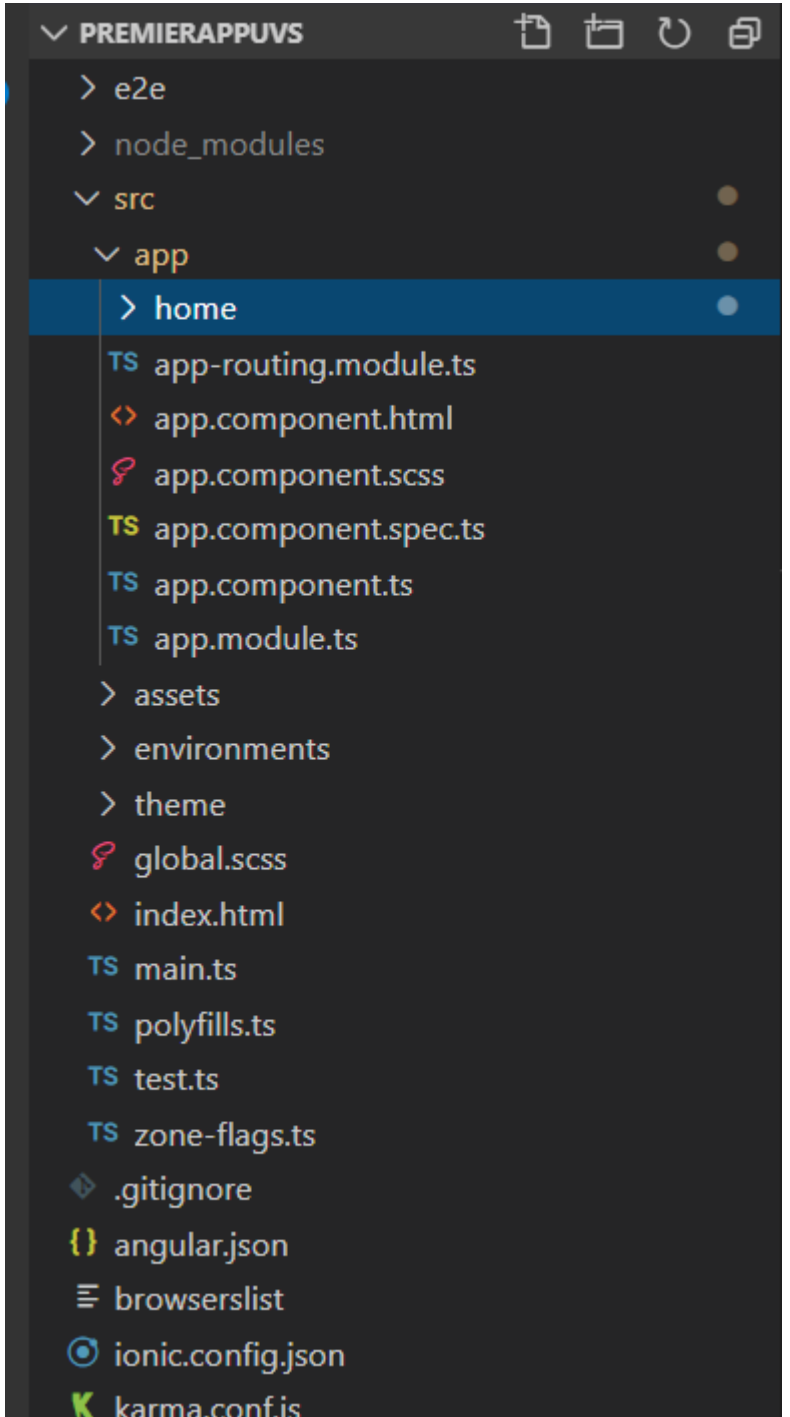
Le dossier ***resources*** les images correspondant à l'icône et au splash screen de votre application.

Le dossier ***www*** est pour la creation des packages finaux de l'application.



Structure Projet

- Le dossier ***src:*** la plus part du temps est passée dans ce dossier pour la réalisation de l'application. Il contient le dossier :
  - **App:**
    - Un component
    - Un module
    - Un template
    - Une feuille de Style
  - **Assets:**
    - contient les assets statiques de l'application comme les audios, images,….
  - **Theme:**
    - Permet de modifier les themes de l'application
  - **Pages:**
    - Contient toutes les pages de l'application



FIN DE LA SEQUENCE 1
Questions

References

- https://ionicframework.com/docs/
- https://openclassrooms.com/fr/courses/5098931-developpez-une-application-mobile-multiplateforme-avec-ionic-3
- https://ionic.mobiletuto.com/v3/chap10/chap10-5.html