

Tuto : installer et utiliser l'émulateur Android avec Apache Cordova

Installation Apache Cordova

- **Installer Node.js**
- **Installer et configurer Cordova :**

```
npm install -g cordova
cordova create <project_name>
cd <project_name>
cordova platform add browser
cordova platform add android
```

- **Lancer le projet :**

```
cordova run <platform>
// ça devrait marcher pour Browser, mais pas encore pour Android. Pour pouvoir développer sur
IOS, il faut avoir un mac.
```

Préparer l'émulateur

- **Installer JDK 1.8 ou antérieur :**

Ajouter au PATH et créer la variable d'environnement JAVA_HOME avec le lien vers le dossier d'installation (pas le dossier \bin) du JDK. Vous pouvez tester l'installation avec les commandes suivante dans un terminal windows :

```
java -version
echo %JAVA_HOME%
```

- **Installer Android SDK (par l'intermédiaire d'Android Studio)**

<https://developer.android.com/studio/>

Lancer Android Studio (pas besoin d'importer un projet, on va simplement utiliser l'application pour installer le SDK Android et configurer un AVD (Android Virtual Device)).

More actions > SDK Manager > SDK Tools

> cocher Android SDK Command-line Tools (latest)

> cocher 'Show Package Details > Cliquer sur Android SDK Build-Tools 32-rc1 > décocher 31.00 & cocher 30.0.3

> OK (en bas).

Configurer les variables d'environnement :

More actions > SDK Manager > Android SDK Location

1) Ajouter une nouvelle variable d'environnement ANDROID_SDK_ROOT avec le chemin jusqu'au dossier d'installation du SDK Android ;

2) Ajouter au PATH les chemins vers :

- le dossier d'installation SDK Android ;

- <chemin_SDK_Android>\cmdline-tools\latest\bin
- <chemin_SDK_Android>\emulator
- <chemin_SDK_Android>\platform-tools
- <chemin_SDK_Android>\build\tools

- **Configurer un AVD :**

Toujours dans Android Studio

More actions > ADV Manager > + Create Virtual Device... > sélectionnez un device (de type Phone) de votre choix puis Next > sélectionnez une image système d'un API Level strictement inférieur à 30 > Download > Next > Finish.

- **Installer gradle :**

<https://gradle.org/install/>

Installing manually > Binary-only > à dézipper et déplacer où vous souhaitez sur votre système.

Créer une nouvelle variable d'environnement gradle avec l'emplacement sur votre système jusqu'au dossier bin (<chemin>\gradle-7.3.1-bin\bin).

Ajouter au PATH.

Vous pouvez tester avec la commande suivante dans un terminal windows :

gradle -v

- **Installer Apache Ant 1.9.16 :**

<https://ant.apache.org/bindownload.cgi>

Ajouter le chemin jusqu'au dossier \bin au PATH.

Lancer l'émulateur

Dans un terminal windows, à l'emplacement du projet Cordova, exécutez la commande suivante :

cordova run android --emulator

Le premier lancement peut prendre du temps (de manière général, la commande met un certain temps à s'exécuter).

Annexes :

Message d'erreur « failed to open \qemu.conf, err : 2 » ?

→ <https://stackoverflow.com/questions/59505749/failed-to-open-qemu-conf-err-2>

Warning « unexpected system image feature string, emulator might not function correctly, please try updating the emulator » ?

→ <https://stackoverflow.com/questions/57424137/emulator-unexpected-feature-list-multidisplay-vulkannulloptionalstrings-yuv420>