

P2022 : Sunshare  
L'HOMMEAU Nicolas

## Installation React Native

### Table des matières

<b>1 - PRÉ-REQUIS.....</b>	<b>2</b>
1.1 - NODEJS.....	2
1.1.1 - Présentation.....	2
1.1.2 - Installation.....	2
1.1.3 - Vérification.....	2
1.2 - NPM.....	3
1.2.1 - Présentation.....	3
1.2.2 - Installation.....	3
1.2.3 - Vérification.....	3
1.3 - JDK.....	3
1.3.1 - Présentation.....	3
1.3.2 - Installation.....	3
1.3.3 - Vérification.....	4
1.4 - ANDROID STUDIO (FACULTATIF).....	5
1.4.1 - Présentation.....	5
1.4.2 - Installation.....	5
1.4.3 - Configuration.....	5
<b>2 - INSTALLATION REACT NATIVE.....</b>	<b>6</b>
2.1 - PRÉSENTATION.....	6
2.2 - INSTALLATION REACT NATIVE CLI.....	6
2.3 - INSTALLATION EXPO CLI.....	6
<b>3 - PREMIÈRE APPLICATION.....</b>	<b>6</b>
3.1 - CRÉATION ET CONFIGURATION.....	6
3.2 - LANCEMENT.....	7
3.2.1 - Configuration d'un émulateur.....	7
3.2.2 - Lancement de l'émulateur.....	8
3.2.3 - Activation débogage.....	9
3.2.4 - Lancement de l'application.....	9

## 1 - Pré-requis

Toutes les actions s'effectueront dans le terminal de commandes et en tant qu'administrateur, l'utilisation de **sudo** est indispensable. Pour l'ouvrir, allez dans la barre de recherche dans Ubuntu puis chercher cmd, cliquez ensuite sur Émulateur de terminal.

### 1.1 - NodeJS

#### 1.1.1 - Présentation

NodeJS est une plateforme logicielle libre permettant l'exécution de JavaScript côté serveur. Elle utilise la machine virtuelle V8 (moteur JavaScript open-source pour les navigateurs Google Chrome et Chromium).



#### 1.1.2 - Installation

Il existe plusieurs façons d'installer NodeJS, nous allons utiliser ici les dépôts officiels d'Ubuntu qui est la manière la plus simple de procéder. À noter que cette manière ne permet pas d'installer forcément la version désirer de NodeJS, cependant nous n'avons pas besoin d'une version spécifique de NodeJS pour cette installation.

Pour l'installation, seulement 2 lignes de commandes seront nécessaires, tout d'abord on va mettre à jour nos paquets :

```
> sudo apt-get update
```

Puis, il ne reste plus qu'à installer nodejs :

```
> sudo apt-get install nodejs
```

```
local@1116-PC11-SNIR:~$ sudo apt-get install nodejs
Lecture des listes de paquets... Fait
Construction de l'arbre des dépendances
Lecture des informations d'état... Fait
Le paquet suivant a été installé automatiquement et n'est plus nécessaire :
  libfwupdplugin1
Veuillez utiliser « sudo apt autoremove » pour le supprimer.
Les paquets suivants seront mis à jour :
  nodejs
1 mis à jour, 0 nouvellement installés, 0 à enlever et 53 non mis à jour.
Il est nécessaire de prendre 25,4 Mo dans les archives.
Après cette opération, 1 024 o d'espace disque supplémentaires seront utilisés.
Réception de :1 https://deb.nodesource.com/node_14.x focal/main amd64 nodejs amd64 14.19.3-deb-1nodesource1 [25,4 MB]
25,4 Mo réceptionnés en 2s (10,9 Mo/s)
(Lecture de la base de données... 261878 fichiers et répertoires déjà installés.)
Préparation du dépaquetage de .../nodejs_14.19.3-deb-1nodesource1_amd64.deb ...
Dépaquetage de nodejs (14.19.3-deb-1nodesource1) sur (14.19.2-deb-1nodesource1) ...
Paramétrage de nodejs (14.19.3-deb-1nodesource1) ...
Traitement des actions différées (« triggers ») pour man-db (2.9.1-1) ...
```

#### 1.1.3 - Vérification

Enfin, pour vérifier la version de NodeJS installée par la commande précédente, il suffit de faire :

```
> node --version
```

Ou :

```
> node -v
```

## 1.2 - NPM

### 1.2.1 - Présentation

NPM est un gestionnaire de paquets pour l'environnement d'exécution NodeJS. Celui-ci permet principalement l'installation de paquets publics stockés dans une base de données en ligne, mais il permet également d'en publier/développer.



### 1.2.2 - Installation

Comme l'installation de NodeJS, on utilisera ici les dépôts officiels d'Ubuntu afin d'installer NPM. Mettre à jour nos paquets une seconde fois n'est pas nécessaire, il n'y a donc qu'une ligne à exécuter :

```
> sudo apt-get install npm
```

```
local@1116-PC11-SNIR:~$ sudo apt-get install npm
[sudo] Mot de passe de local :
Lecture des listes de paquets... Fait
Construction de l'arbre des dépendances
Lecture des informations d'état... Fait
Certains paquets ne peuvent être installés. Ceci peut signifier
que vous avez demandé l'impossible, ou bien, si vous utilisez
la distribution unstable, que certains paquets n'ont pas encore
été créés ou ne sont pas sortis d'Incoming.
```

### 1.2.3 - Vérification

Enfin, pour vérifier la version NPM installée, on utilise le même procédé que NodeJS, c'est-à-dire :

```
> npm --version
```

Ou :

```
> npm -v
```

## 1.3 - JDK

### 1.3.1 - Présentation

JDK (Java Development Kit) est un ensemble de bibliothèques logicielles de base de Java. On en a besoin ici pour le développement de l'application, il est indispensable pour travailler sur des projets React Native.



### 1.3.2 - Installation

L'installation ici requiert une commande en plus par rapport aux autres installations :

```
> sudo add-apt-repository ppa:openjdk-r/ppa
```

```
local@1116-PC11-SNIR:~$ sudo add-apt-repository ppa:openjdk-r/ppa
[sudo] Mot de passe de local :

Plus d'info : https://launchpad.net/~openjdk-r/+archive/ubuntu/ppa
Appuyez sur [ENTRÉE] pour continuer ou sur Ctrl-c pour annuler l'ajout.

Réception de :1 http://ppa.launchpad.net/openjdk-r/ppa/ubuntu focal InRelease [23,8 kB]
Atteint :2 http://fr.archive.ubuntu.com/ubuntu focal InRelease
Réception de :3 http://security.ubuntu.com/ubuntu focal-security InRelease [114 kB]
Réception de :4 http://fr.archive.ubuntu.com/ubuntu focal-updates InRelease [114 kB]
Atteint :5 https://deb.nodesource.com/node_14.x focal InRelease
Ign :6 http://ppa.launchpad.net/webupd8team/java/ubuntu focal InRelease
Ign :7 http://www-fourier.ujf-grenoble.fr/~parisse/debian stable InRelease
Réception de :8 http://fr.archive.ubuntu.com/ubuntu focal-backports InRelease [108 kB]
Err :9 http://ppa.launchpad.net/webupd8team/java/ubuntu focal Release
404 Not Found [IP : 185.125.190.52 80]
Atteint :10 http://www-fourier.ujf-grenoble.fr/~parisse/debian stable Release
Réception de :11 http://ppa.launchpad.net/openjdk-r/ppa/ubuntu focal/main amd64 Packages [14,4 kB]
Réception de :12 http://ppa.launchpad.net/openjdk-r/ppa/ubuntu focal/main i386 Packages [14,3 kB]
Atteint :13 https://download.sublimetext.com apt/stable/ InRelease
Réception de :14 http://ppa.launchpad.net/openjdk-r/ppa/ubuntu focal/main Translation-en [1 480 B]
Réception de :15 http://security.ubuntu.com/ubuntu focal-security/main amd64 DEP-11 Metadata [40,7 kB]
Réception de :16 http://security.ubuntu.com/ubuntu focal-security/universe amd64 DEP-11 Metadata [66,6 kB]
Réception de :17 http://security.ubuntu.com/ubuntu focal-security/multiverse amd64 DEP-11 Metadata [2 464 B]
Réception de :18 http://fr.archive.ubuntu.com/ubuntu focal-updates/main amd64 DEP-11 Metadata [278 kB]
Réception de :20 http://fr.archive.ubuntu.com/ubuntu focal-updates/universe amd64 DEP-11 Metadata [391 kB]
Réception de :21 http://fr.archive.ubuntu.com/ubuntu focal-updates/multiverse amd64 DEP-11 Metadata [944 B]
Réception de :22 http://fr.archive.ubuntu.com/ubuntu focal-backports/main amd64 DEP-11 Metadata [8 000 B]
Réception de :23 http://fr.archive.ubuntu.com/ubuntu focal-backports/universe amd64 DEP-11 Metadata [30,7 kB]
Lecture des listes de paquets... Fait
E: Le dépôt http://ppa.launchpad.net/webupd8team/java/ubuntu focal Release n'a pas de fichier Release.
N: Les mises à jour depuis un tel dépôt ne peuvent s'effectuer de manière sécurisée, et sont donc désactivées par défaut.
N: Voir les pages de manuel d'apt-secure(8) pour la création des dépôts et les détails de configuration d'un utilisateur.
```

Celle-ci permet d'ajouter le package openjdk qui constitue l'implémentation de référence officielle et libre de Java. Suite à cela, il est préférable de remettre à jour les paquets pour être sûr d'avoir les bonnes versions :

```
> sudo apt-get update
```

Enfin, il ne reste plus qu'à installer JDK :

```
> sudo apt-get install openjdk-8-jdk
```

```
local@1116-PC11-SNIR:~$ sudo apt-get install openjdk-8-jdk
Lecture des listes de paquets... Fait
Construction de l'arbre des dépendances
Lecture des informations d'état... Fait
openjdk-8-jdk est déjà la version la plus récente (8u312-b07-0ubuntu1~20.04).
Le paquet suivant a été installé automatiquement et n'est plus nécessaire :
  libfwpdplugin1
Veuillez utiliser « sudo apt autoremove » pour le supprimer.
0 mis à jour, 0 nouvellement installés, 0 à enlever et 53 non mis à jour.
```

A noter : ici JDK étant déjà installé et n'ayant pas de nouvelles versions, aucunes installations ne s'est effectuée. Ceci peut également se produire sur les autres installations.

### 1.3.3 - Vérification

La vérification de la version reste cependant la même :

```
> java -version
```

```
local@1116-PC11-SNIR:~$ java -version
openjdk version "11.0.15" 2022-04-19
OpenJDK Runtime Environment (build 11.0.15+10-Ubuntu-0ubuntu0.20.04.1)
OpenJDK 64-Bit Server VM (build 11.0.15+10-Ubuntu-0ubuntu0.20.04.1, mixed mode, sharing)
```

## 1.4 - Android Studio (facultatif)

### 1.4.1 - Présentation

Android Studio est un environnement de développement pour développer des applications mobiles pour Android. Il utilise le moteur de production Gradle. Dans notre cas, on utilise Android Studio uniquement pour son émulateur permettant de faire des tests directement sur l'application et voir les résultats de ces tests en temps réel.



### 1.4.2 - Installation


Pour installer Android Studio, il suffit de se rendre sur le site officiel d'Android Studio et de télécharger la version proposée par celui-ci (le bouton vert "Download Android Studio"). Une fois téléchargé, ouvrez le fichier obtenu puis extrayez le dossier "android-studio" où vous le souhaitez (exemple : dans le répertoire /Documents). Une fois fait, rendez-vous dans l'émulateur de terminal et déplacez-vous dans le dossier "bin" dans les fichiers d'Android Studio :

```
> cd Documents/android-studio/bin
```

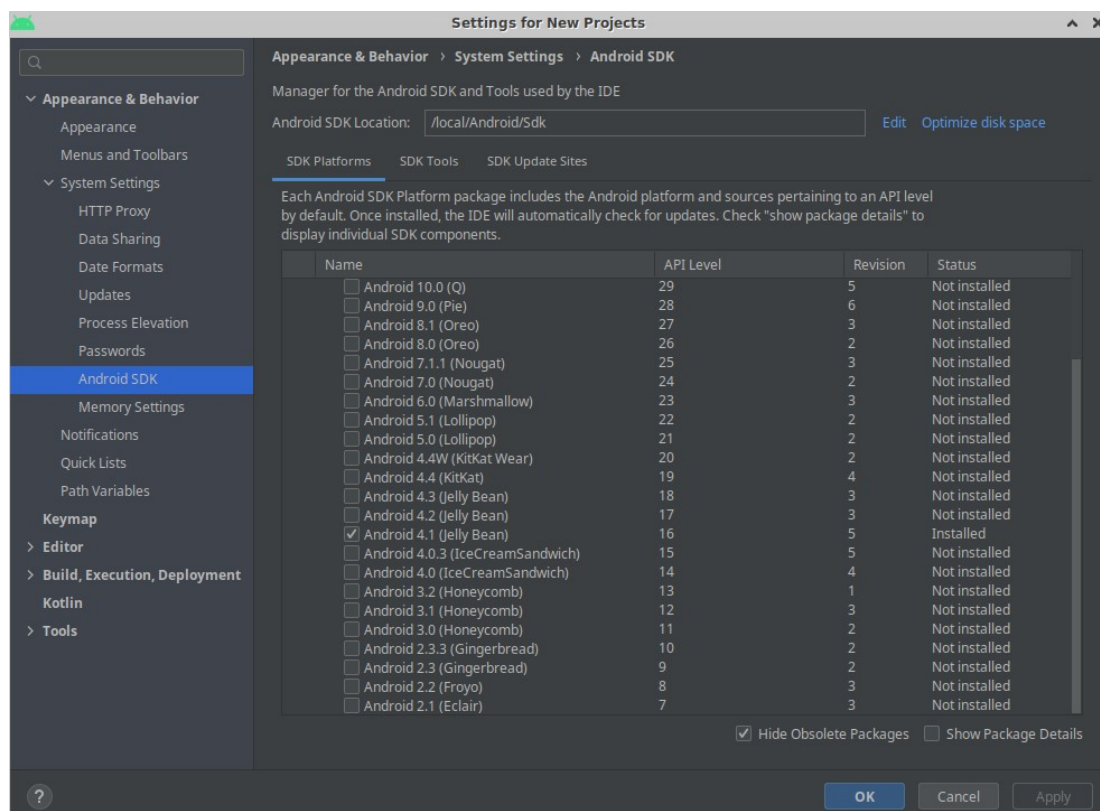
Puis exécutez le script **studio** afin de lancer l'application :

```
> ./studio.sh
```

### 1.4.3 - Configuration

Une fois l'application lancée, cliquez sur le bouton représentant une boîte fléchée  dans le coin supérieur gauche de l'application. Une fois dans ce menu, rendez-vous dans l'onglet "SDK Tools" et vérifiez que les options "Android SDK Build-Tools", "Android Emulator" et "Android SDK Platform-Tools" sont bien cochées.

Quand ceci est fait, il ne reste plus qu'à installer une version d'Android. Pour que l'application soit compatible avec le plus de smartphones possibles, l'API 16 (4.1 - Jelly Bean) est recommandée. Pour l'installer il suffit de se rendre dans l'onglet "SDK Platforms", sélectionnez "Android 4.1 (Jelly Bean)" puis cliquez sur **Apply**, et **OK**. Laissez l'installation se terminer et cliquez sur **Finish**.





Pour finir, retournez dans un terminal, et entrez la commande :

```
> sudo nano $HOME/.bash_profile
```

Cela va créer un fichier, et dans celui-ci, rajoutez ces lignes :

```
export ANDROID_HOME=$HOME/Android/Sdk
export PATH=$PATH:$ANDROID_HOME/emulator
export PATH=$PATH:$ANDROID_HOME/tools
export PATH=$PATH:$ANDROID_HOME/tools/bin
export PATH=$PATH:$ANDROID_HOME/platform-tools
```

## 2 - Installation React Native

### 2.1 - Présentation

React Native est un framework d'applications mobiles open source créé par Facebook. Il permet de développer des applications pour Android, iOS et UWP.



### 2.2 - Installation React Native CLI

Pour installer React Native, on utilise le même procédé que pour les autres installations :

```
> sudo npm install -g react-native-cli
```

### 2.3 - Installation Expo CLI

Expo est une plateforme open-source permettant la création et la gestion des applications. Elle permet notamment de tester son application directement sur son téléphone à l'aide d'un QR Code et d'une application (Expo Go). Ici, même technique pour l'installer :

```
> sudo npm install -g expo-cli
```

```
local@1116-PC11-SNIR:~$ sudo npm install -g expo-cli
[sudo] Mot de passe de local :
npm WARN deprecated graphql-tools@3.0.0: This package has been deprecated and now it only exports makeExecutableSchema. And it will no longer receive updates. We recommend you to migrate to scoped packages such as @graphql-tools/schema, @graphql-tools/utils and etc. Check out https://www.graphql-tools.com to learn what package you should use instead
npm WARN deprecated uuid@3.4.0: Please upgrade to version 7 or higher. Older versions may use Math.random() in certain circumstances, which is known to be problematic. See https://v8.dev/blog/math-random for details.
npm WARN deprecated svgo@1.3.2: This SVGO version is no longer supported. Upgrade to v2.x.x.
npm WARN deprecated source-map-resolve@0.5.3: See https://github.com/lydell/source-map-resolve#deprecated
npm WARN deprecated source-map-url@0.4.1: See https://github.com/lydell/source-map-url#deprecated
npm WARN deprecated urix@0.1.0: Please see https://github.com/lydell/urix#deprecated
npm WARN deprecated resolve-url@0.2.1: https://github.com/lydell/resolve-url#deprecated
npm WARN deprecated querystring@0.2.0: The querystring API is considered Legacy. new code should use the URLSearchParams API instead.
npm WARN deprecated chokidar@2.1.8: Chokidar 2 will break on node v14+. Upgrade to chokidar 3 with 15x less dependencies.
npm WARN deprecated fsevents@1.2.13: fsevents 1 will break on node v14+ and could be using insecure binaries. Upgrade to fsevents 2.
/usr/bin/expo -> /usr/lib/node_modules/expo-cli/bin/expo.js
/usr/bin/expo-cli -> /usr/lib/node_modules/expo-cli/bin/expo.js
npm WARN optional SKIPPING OPTIONAL DEPENDENCY: fsevents@2.3.2 (node_modules/expo-cli/node_modules/chokidar/node_modules/fsevents):
npm WARN notsup SKIPPING OPTIONAL DEPENDENCY: Unsupported platform for fsevents@2.3.2: wanted {"os":"darwin","arch":"any"} (current: {"os":"linux","arch":"x64"})
npm WARN optional SKIPPING OPTIONAL DEPENDENCY: fsevents@1.2.7 (node_modules/expo-cli/node_modules/watchpack-chokidar2/node_modules/chokidar/node_modules/fsevents):
npm WARN notsup SKIPPING OPTIONAL DEPENDENCY: Unsupported platform for fsevents@1.2.7: wanted {"os":"darwin","arch":"any"} (current: {"os":"linux","arch":"x64"})
npm WARN optional SKIPPING OPTIONAL DEPENDENCY: fsevents@1.2.7 (node_modules/expo-cli/node_modules/webpack-dev-server/node_modules/chokidar/node_modules/fsevents):
npm WARN notsup SKIPPING OPTIONAL DEPENDENCY: Unsupported platform for fsevents@1.2.7: wanted {"os":"darwin","arch":"any"} (current: {"os":"linux","arch":"x64"})
npm WARN ajv-keywords@3.5.2 requires a peer of ajv@^6.9.1 but none is installed. You must install peer dependencies yourself.

+ expo-cli@5.4.6
added 1449 packages from 689 contributors in 72.931s
```

## 3 - Première application

### 3.1 - Création et configuration

Pour créer notre première application, rendez-vous dans le dossier où vous souhaitez créer votre application :

```
> cd Documents/MesApplications
```

Ensuite, on initialise l'application :

```
> expo init MonApp
```

Ici on sélectionne une application vide en TypeScript à l'aide des flèches :

```
local@1116-PC11-SNIR:~/Documents/MesApplications$ expo init MonApp

There is a new version of expo-cli available (5.4.6).
You are currently using expo-cli 5.0.3
Install expo-cli globally using the package manager of your choice;
for example: `npm install -g expo-cli` to get the latest version

? Choose a template: > - Use arrow-keys. Return to submit.
  ----- Managed workflow -----
  blank                               a minimal app as clean as an empty canvas
> blank (TypeScript) same as blank but with TypeScript configuration
  tabs (TypeScript) several example screens and tabs using react-navigation and TypeScript
  ----- Bare workflow -----
  minimal                             bare and minimal, just the essentials to get you started
```

Attendez un peu moins d'une minute et votre application sera créée :

```
✓ Choose a template: > blank (TypeScript) same as blank but with TypeScript configuration
✓ Downloaded and extracted project files.
📦 Using npm to install packages.
✓ Installed JavaScript dependencies.

✓ Your project is ready!

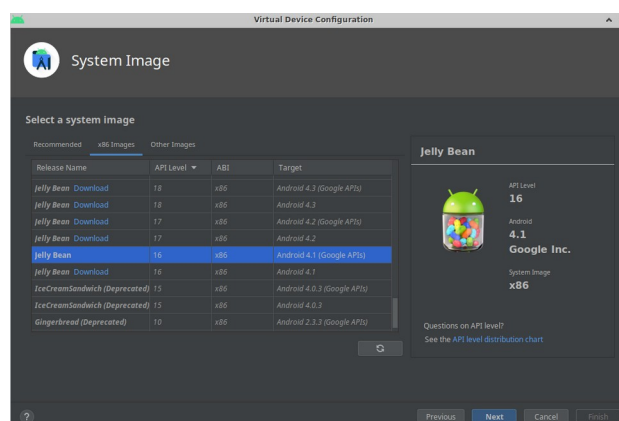
To run your project, navigate to the directory and run one of the following npm commands.

- cd MonApp
- npm start # you can open iOS, Android, or web from here, or run them directly with the commands below.
- npm run android
- npm run ios # requires an iOS device or macOS for access to an iOS simulator
- npm run web
```

## 3.2 - Lancement

### 3.2.1 - Configuration d'un émulateur

Maintenant que l'application est créée, on peut la lancer. Tout d'abord sur Android Studio, cliquez sur le bouton à gauche de la flèche verte "Run", puis sélectionnez "AVD Manager". Une fois la fenêtre ouverte, cliquez sur **Create Virtual Device...**, choisissez un modèle dans la catégorie "Phone" et cliquez sur **Next**. Enfin, sélectionnez "x86 Images" puis prenez une version dont le nom dans la catégorie "Target" comporte "(Google APIs)", cliquez sur [Download](#), **Finish**, sélectionnez la version en question et cliquez sur **Next**, puis **Finish**.



### 3.2.2 - Lancement de l'émulateur

La configuration d'émulateur est maintenant créée, retourner sur la page principale d'Android Studio et sélectionnez à nouveau le bouton à côté du bouton "Run". Maintenant cliquez sur l'émulateur que vous venez de créer (ici Pixel XL API 16) et sélectionnez le bouton "Run". Une fois l'émulateur complètement lancé, retournez sur le terminal se trouvant dans le dossier MesApplications, rendez-vous dans votre dossier d'application :

```
> cd MonApp/
```

Et lancez enfin l'application :

```
> sudo npm start
```

Ou :

```
> sudo expo start
```

```
local@1116-PC11-SNIR:~/Documents/MesApplications$ cd MonApp/  
local@1116-PC11-SNIR:~/Documents/MesApplications/MonApp$ sudo expo start  
[sudo] Mot de passe de local :
```

```
There is a new version of expo-cli available (5.4.6).  
You are currently using expo-cli 5.0.3  
Install expo-cli globally using the package manager of your choice;  
for example: `npm install -g expo-cli` to get the latest version
```

```
Starting project at /local/Documents/MesApplications/MonApp  
Developer tools running on http://localhost:19002  
Starting Metro Bundler
```



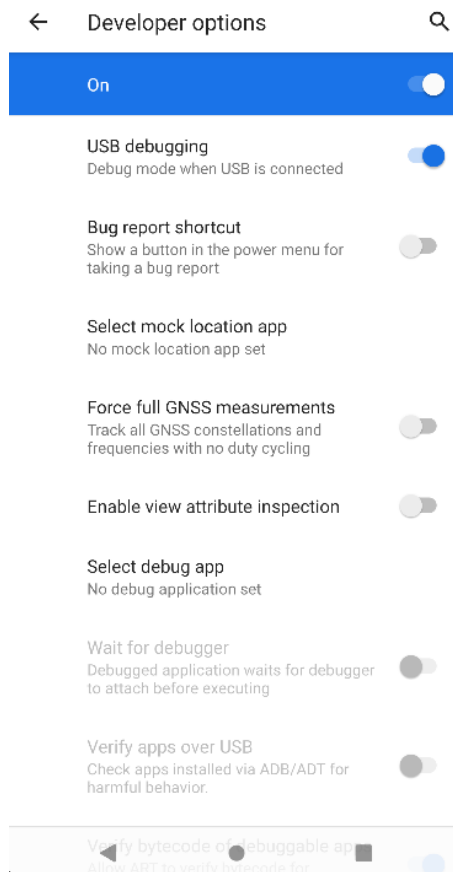
```
> Metro waiting on exp://172.31.6.111:19000  
> Scan the QR code above with Expo Go (Android) or the Camera app (iOS)  
  
> Press a | open Android  
> Press w | open web  
  
> Press r | reload app  
> Press m | toggle menu  
> Press d | show developer tools  
> shift+d | toggle auto opening developer tools on startup (disabled)  
  
> Press ? | show all commands  
  
Logs for your project will appear below. Press Ctrl+C to exit.
```



### 3.2.3 - Activation débogage

Le débogage doit d'abord être activé sur l'émulateur :

- Dans les paramètres "A propos" puis "Informations sur le logiciel", appuyer sur le numéro de build 7 fois
- Retourner dans le menu des paramètres, entrer dans les options de développeurs tout en bas de la page puis activer le débogage



### 3.2.4 - Lancement de l'application

Sur le terminal où l'application a été exécutée avec **sudo expo start**, appuyez sur 'a', cela va installer l'application Expo Go sur l'émulateur et lancer l'application par la suite.