

React Native

Projet développement mobile

Formateur : Djemai Samy

PARTIE 1: Introduction et Initiation:

1/ React Native	1
1.1/ Framework React Native:	1
1.2/ DOM réel et virtuel	1
2/ Expo:	2
2.1/ Expo:	2
2.2/ EXPO SDK:	3
2.3/ EXPO CLI	3
2.4/Expo devtools:	3
2.5/Expo Client app:	3
2.6/Publication:	4
Tester l'App:	4
Publication de l'App:	4
Expo OTA:	4
Release Channel:	4
3/ Installations:	5
3.1/ Node.js:	5
WINDOWS:	6
MAC:	6
3.2/ Expo cli:	7
4/ Lancement du Projet:	7
4.1/ Initialiser le projet en utilisant Expo cli:	7
Sur navigateur:	7
Sur téléphone:	9

1/ React Native

1.1/ Framework React Native:

React native est une librairie qui permet de créer des applications android et IOS en utilisant Javascript et du JSX.

C'est un outil qui a été développé par Facebook en 2015 dans le but d'accélérer le développement et la maintenance de ses applications mobiles.

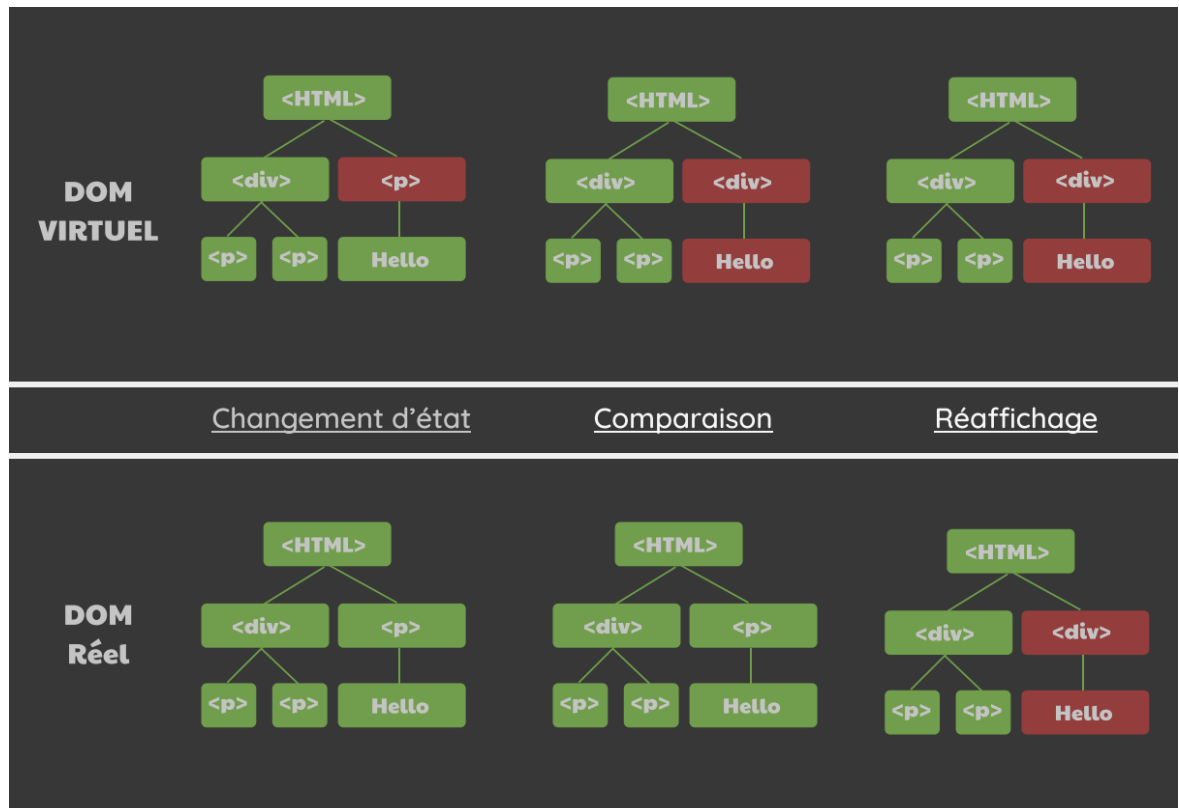
1.2/ DOM réel et virtuel

React utilise un DOM (Document Object Model) virtuel. Le DOM (réel) est l'arbre de nœuds, la représentation textuelle de toute votre application web. Quand nous effectuons un changement d'état au sein de notre application, l'intégralité du DOM est recréée.

Le DOM virtuel quant à lui est un second arbre, correspondant à l'état réel de la globalité de votre application.

Si nous effectuons une modification d'un composant (couleur, texte, etc.). Avec React, seul le DOM virtuel sera modifié. React va alors réaliser une comparaison avec des algorithmes spécifiques entre le DOM virtuel fraîchement actualisé, et le DOM réel.

En fonction des différences entre les deux arbres, seuls les nœuds qui ont un nouvel état, seront redessinés dans le DOM réel. Cela permet de redessiner seulement un seul nœud.



2/ Expo:

2.1/ Expo:

Expo est un framework et une plateforme pour les applications React.

Il s'agit d'un ensemble d'outils et de services construits autour de React Native et de plateformes natives qui nous aident à développer, créer, déployer et itérer rapidement sur des applications IOS, Android et Web à partir de la même codebase JavaScript.

2.2/ EXPO SDK:

Le Expo SDK est une librairie qui nous donne accès à une grande variété d'APIs native sur android et ios. Nous avons accès par exemple au FileSystem du device pour lire et écrire des fichiers, à la caméra pour prendre des photos, etc.

2.3/ EXPO CLI

L'expo cli (command line tool) est un outil phare dans le processus de développement d'une application react native avec Expo.

C'est un ensemble d'instructions en ligne de commande qui nous aide à développer facilement et rapidement nos projets.

Avec Expo cli nous pouvons :

- Démarrer un projet Expo avec la commande : expo init.
- Exécuter notre app et commencer le développement : expo start
- Publier notre app pour la faire tester par nos clients/ beta-users etc.: expo publish
- Créer nos binaires qui vont être soumis aux stores: expo build:android , expo build:ios , expo build:web etc

2.4/Expo devtools:

Expo devtools c'est le côté UI de expo. Nous avons presque toutes les fonctionnalités du Expo cli en version web où nous pouvons cliquer sur des boutons pour faire des actions au lieu d'écrire des commandes.

2.5/Expo Client app:

C'est une app dans laquelle nous pourrions tester et avoir un rendu en temps réel de notre application, sur un appareil. (ex: votre téléphone).

2.6/Publication:

- **Tester l'App:**

Une fois l'App terminée. Nous voulons la faire tester par notre client ou par quelques utilisateurs. Nous n'avons pas besoin de la mettre sur les stores pour ça, Expo nous fournit une commande qui nous permet de rendre notre app publique dans l'app Expo client.

- **Publication de l'App:**

Pour déployer, une app sur Playstore nous devons soumettre un fichier APK, sur appstore ça sera un IPA. Expo nous donne une commande très simple pour générer ces fichiers :

Expo build:android pour générer l'APK

Expo build:ios pour générer le IPA

- **Expo OTA:**

Une fois sur le store, si nous voulons modifier notre code et une mise à jour. Le processus normal est de créer un nouveau binaire (apk et ipa) puis de le soumettre encore une fois aux stores, cela peut prendre du temps.

Le OTA (over the air) update vous permet de mettre à jour votre app sans pour autant soumettre un nouveau binaire dans le store. Juste avec la commande expo publish, votre application est mise à jour.

- **Release Channel:**

Si nous voulons ajouter une nouvelle fonctionnalité. Nous voudrions la tester avec notre client ou utilisateurs. En utilisant expo publish tous les utilisateurs qui utilisent la version prod vont recevoir notre mise à jour alors que nous voulons juste un environnement de test pour notre client.

Releases channels nous permet de publier différentes versions de notre app.

Nous allons donc avoir une channel Staging pour tester nos nouvelles fonctionnalités en utilisant la commande : `expo publish --release-channel staging`.

Les utilisateurs qui utilisent la version prod ne verront pas nos mises à jour. En réalité quand vous faites `expo publish`, expo ajoute implicitement une channel default.

3/ Installations:

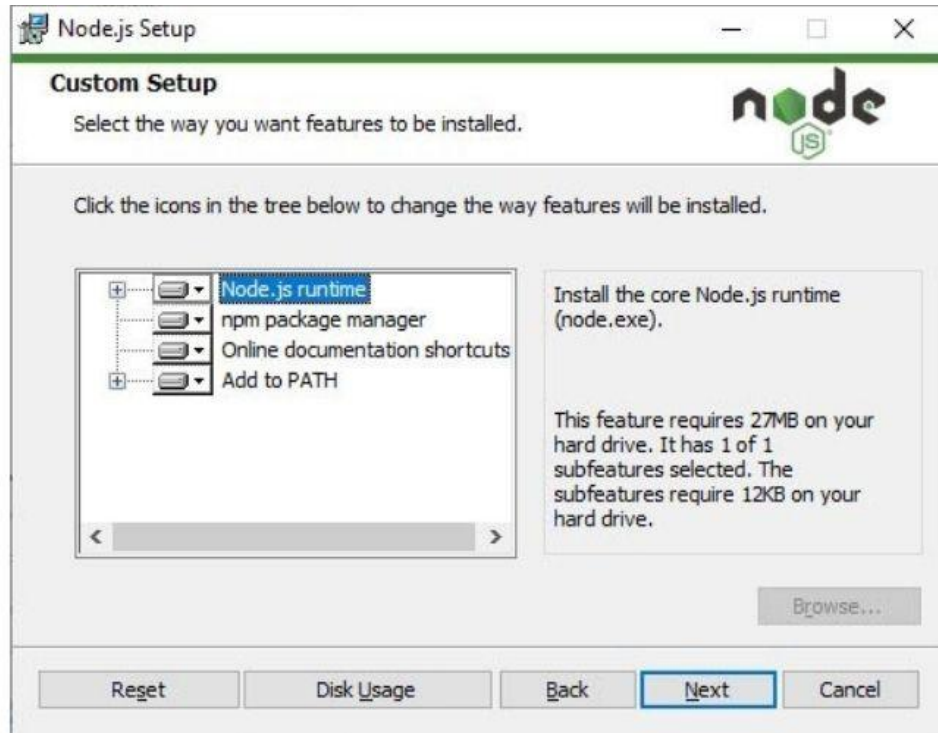
3.1/ Node.js:

Avant d'installer Node.js, utilisez la commande : `node-v` pour vérifier si Node.js est installé sur votre machine.

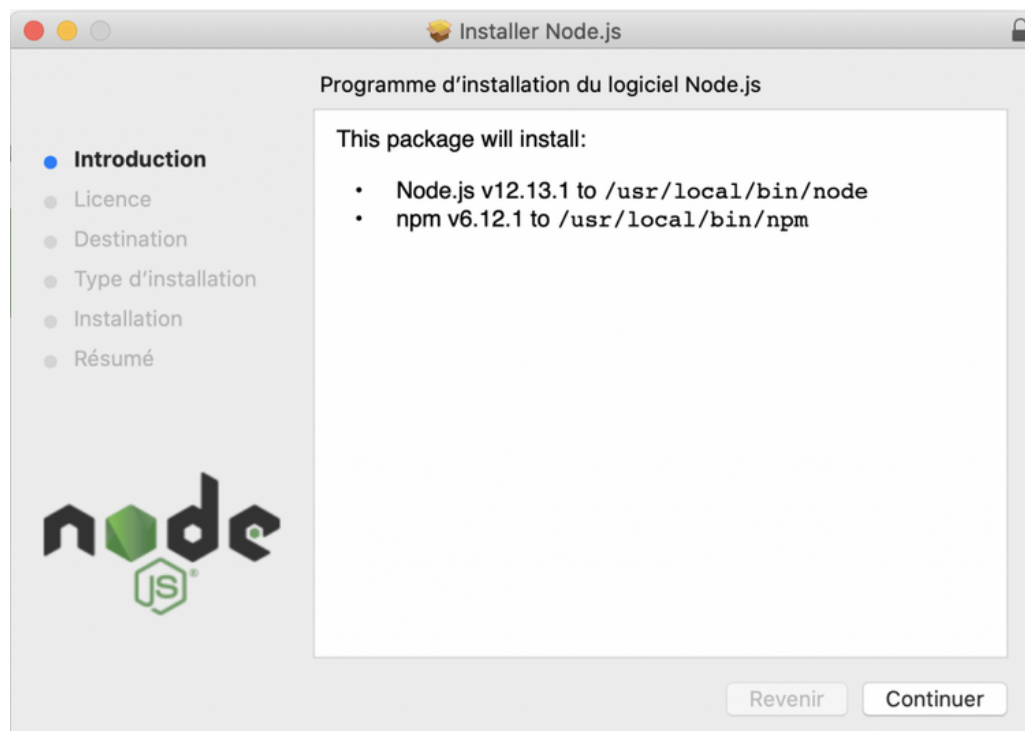
Pour télécharger node.js, ouvrez la page: www.nodejs.org et téléchargez la version LTS.

Après les habituelles étapes d'acceptation de la licence et du choix du dossier d'installation, le programme d'installation vous propose de choisir vos fonctionnalités et leur méthode d'installation.

WINDOWS:



MAC:



3.2/ Expo cli:

Pour installer Expo-cli globalement utilisez la commande :

```
npm install --g expo-cli
```

Pour vérifier que l'installation s'est bien utilisée la commande :

```
expo --version
```

Si vous rencontrez l'erreur : 'expo command not found' vous devez ajouter le dossier au PATH

4/ Lancement du Projet:

4.1/ Initialiser le projet en utilisant Expo cli:

Ouvrez un terminal dans le dossier désiré, puis entrez la commande :

```
npx create-expo-app app
```

Cette commande crée un dossier nommé monapp, contenant tous les fichiers nécessaires au lancement de l'application.

Une fois l'installation terminée, utilisez cette commande pour lancer l'application:

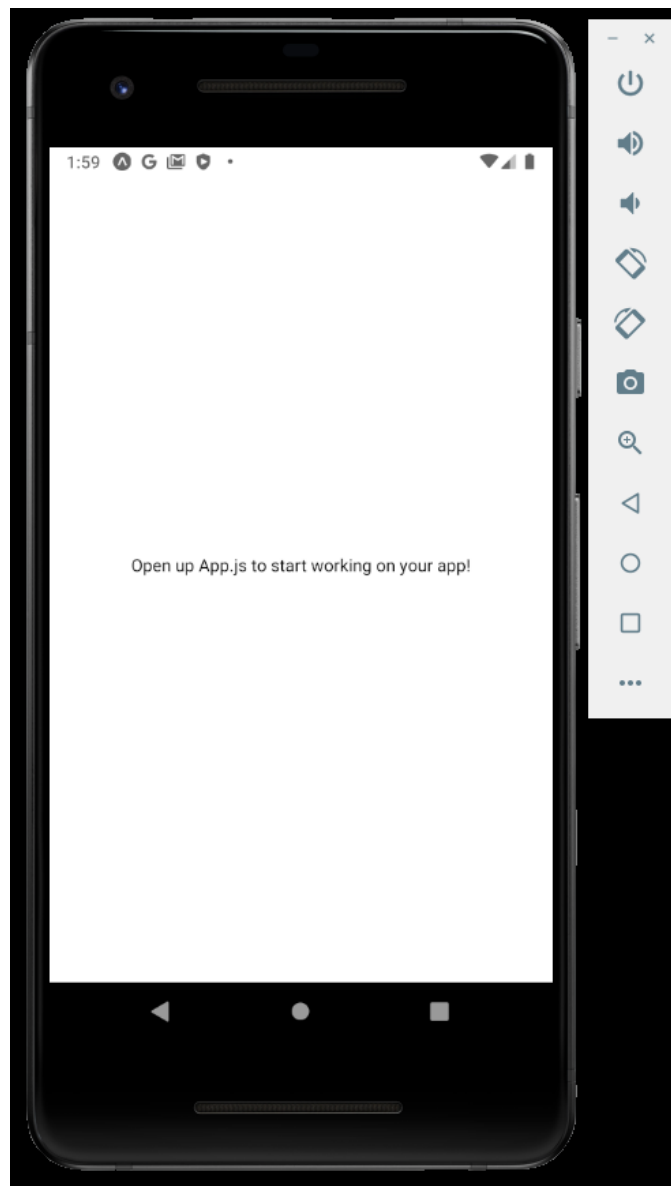
```
cd app
```



```
npx expo start
```

- **Sur téléphone:**

Téléchargez l'application Expo Go sur le store. Puis scannez le QR code, pour ouvrir l'application.



- **Sur ordinateur:**

D'abord, il faut installer les dépendances :

```
npx expo install react-native-web  
react-dom @expo/webpack-config
```

Une fois l'application lancée, entrez la touche 'w' dans le terminal pour ouvrir l'application sur le navigateur.