# Création d’un projet Vue-Native

## Installation de Vue-Native

Afin d’être en mesure de créer un projet Vue-Native, il vous faudra divers outils. Vous devrez commencer par vous assurer d’avoir installé de manière globale les programmes suivants :

* Node (version supérieure à 6.0)
* npm (version supérieure à 4.0)
* Expo CLI ou React Native CLI

Afin d’être en mesure de créer un projet Vue Native, vous aurez besoin de l’outil *Vue Native CLI* qui vous permettra de créer automatiquement un projet Vue Native incorporant pour vous les fichiers requis afin que vos composantes Vue soient interprétées et converties en composants React Native.

Vous pourrez installer cet outil grâce à la commande suivante :

npm install --global vue-native-cli

## Création d’un projet utilisant Expo CLI

L’outil Expo CLI vous permettra de compiler vos projets Vue Native et React Native et lancera un serveur de développement afin que vous puissiez tester vos projets, de différentes manières, que ce soit dans un émulateur Android ou IOS, dans votre navigateur web ou encore en scannant un code QR avec votre téléphone cellulaire.

Vous pourrez installer l’outil Expo CLI grâce à la commande suivante :

npm install --global expo-cli

Ensuite, dirigez-vous dans le dossier qui contiendra votre projet Vue Native et lancez la commande suivante :

vue-native init <Nom du projet>

Un projet Vue Native est désormais créé.

IMPORTANT

Afin que l’outil de packaging reconnaisse vos fichiers Vue, il est important de modifier le fichier *app.json* du projet de la manière suivante :

{  
 "expo": {  
 "name": "Your app's name",  
 "platforms": [  
 "ios",  
 "android",  
 "web"  
 ],  
 "version": "1.0.0",  
 ...  
 "packagerOpts": {  
+ "sourceExts": ["js", "json", "ts", "tsx", "vue"],  
 "config": "metro.config.js"  
 }  
 }  
}

AVANT de lancer l’application, il est également très important d’installer deux autres outils dans votre projet, comme ceci :

npm install vue-native-core vue-native-helper --save  
npm install vue-native-scripts --save-dev

Les outils vue-native-core et vue-native-helper sont nécessaires afin de permettre aux composants Vue Native d’être instanciés lors de l’exécution et l’outil vue-native-scripts est utilisé afin de convertir les *Single file components* Vue en composants React.

## Configuration du packager React Native :

Vous aurez besoin de deux fichiers dans votre projet afin qu’il soit packagé correctement. Ces deux fichiers sont les suivants :

D’abord, le fichier *vueTransformerPlugin.js* conetant le code ci-dessous sera nécessaire.

// For React Native 0.59 and above  
var upstreamTransformer = require("metro-react-native-babel-transformer");  
  
var vueNativeScripts = require("vue-native-scripts");  
var vueExtensions = ["vue"];  
  
module.exports.transform = function({ src, filename, options }) {  
 if (vueExtensions.some(ext => filename.endsWith("." + ext))) {  
 return vueNativeScripts.transform({ src, filename, options });  
 }  
 return upstreamTransformer.transform({ src, filename, options });  
};

Ensuite, assurez-vous d’avoir le fichier *metro.config.js*.

const { getDefaultConfig } = require("metro-config");  
  
module.exports = (async () => {  
 const {  
 resolver: { sourceExts }  
 } = await getDefaultConfig();  
 return {  
 transformer: {  
 babelTransformerPath: require.resolve("./vueTransformerPlugin.js"),  
 getTransformOptions: async () => ({  
 transform: {  
 experimentalImportSupport: false,  
 inlineRequires: false,  
 },  
 })  
 },  
 resolver: {  
 sourceExts: [...sourceExts, "vue"]  
 }  
 };  
})();

IMPORTANT

La création d’un projet Vue Native par l’outil Vue Native CLI comporte un bogue qui fait en sorte qu’un fichier React Native nommé *App.js* est créé dans le projet au même titre que le fichier *App.vue*. Cependant, le projet utilisera le fichier React Native et ignorera complétement le fichier .vue. Il est donc nécessaire de supprimer le fichier *App.js* AVANT de lancer la compilation du projet.

## Lancer le projet Vue Native

Afin de lancer le projet Vue Native, il vous suffit d’utiliser la commande suivante È

npm start