

Formation React Native avec Expo

Objectif

Ce module de formation de **30 heures** vous apprendra à développer une application mobile avec **React Native**, **Expo** en prenant comme projet fil rouge : **The Todo List App**. - Visuel de la formation —

Programme détaillé

Séance 1 : Introduction à React Native et mise en place du projet

- Présentation de React Native : avantages, différences avec React web.
- Introduction à Expo : pourquoi l'utiliser ?
- Présentation de ES6 et son intérêt.
- Installation de l'environnement de développement.
- Création du projet **The Todo List App** avec Expo.
- Structure d'un projet React Native.
- Premier écran : affichage d'un simple texte.

Exercice : Afficher un message de bienvenue sur l'écran.

Séance 2 : Composants, State et Props

- Présentation des composants de base (View, Text, Button, TextInput...).
- Différence entre composants fonctionnels et class-based.
- Utilisation du **useState** pour gérer l'état local.
- Passage des données entre composants avec les props.
- Création du premier composant réutilisable : **TodoItem**.

Exercice : Créer un composant **TodoItem** qui affiche une tâche avec un bouton pour la supprimer.

Séance 3 : Styles et mise en forme des composants

- Différentes façons de styler les composants :
 - **StyleSheet.create**
 - Style inline
 - Styled Components (CSS-in-JS)
- Gestion des couleurs et thèmes.
- Flexbox en React Native.
- Ajout d'icônes avec **react-native-vector-icons**.

Exercice : Styliser l'application avec des couleurs et des styles bien organisés.

Séance 4 : Listes et gestion des événements

- Utilisation de `FlatList` et `ScrollView`.
- Gestion des événements avec `onPress`.
- Ajout de nouvelles tâches avec `TextInput`.
- Mise à jour de l'état global.

Exercice : Ajouter une tâche à la liste avec un champ de saisie.

Séance 5 : Navigation avec React Navigation

- Installation et configuration de `react-navigation`.
- Création d'une navigation entre l'écran principal et un écran de détails.
- Passage de paramètres entre écrans.

Exercice : Créer un deuxième écran affichant les détails d'une tâche.

Séance 6 : Gestion du contexte et stockage des données

- Introduction à `Context` API pour la gestion globale de l'état.
- Mise en place d'un `TodoContext`.
- Introduction à `AsyncStorage`.
- Sauvegarde et récupération des tâches stockées localement.

Exercice : Sauvegarder et restaurer les tâches avec `AsyncStorage`.

Séance 7 : Requêtes API et gestion des données externes

- Introduction aux requêtes HTTP en React Native.
- Utilisation de **Fetch API** et **Axios**.
- Récupération d'une liste de tâches depuis une API.
- Gestion des états de chargement (`loading`, `error`).

Exercice : Faire un appel API pour récupérer des tâches et les afficher.

Séance 8 : Hooks avancés et gestion des effets de bord

- Utilisation de `useEffect` pour gérer les effets de bord.
- Gestion des dépendances dans `useEffect`.
- Utilisation de `useReducer` comme alternative à `useState`.
- Optimisation des performances avec `useMemo` et `useCallback`.

Exercice : Sauvegarder automatiquement les tâches avec `useEffect`.

Séance 9 : Animation et interactions avancées

- Introduction à `react-native-reanimated` et `react-native-gesture-handler`.
- Ajout d'une animation sur la suppression d'une tâche.
- Ajout de gestes interactifs (swipe pour supprimer une tâche).

Exercice : Animer la suppression d'une tâche.

Séance 10 : Notifications Push

- Introduction aux notifications push dans les applications mobiles.
- Intégration des notifications push via **Expo Notifications**.
- Gestion des permissions et des paramètres de notification.
- Implémentation d'une notification pour des actions clés de l'application (ajout, modification ou suppression d'une tâche).

Exercice : Implémenter une notification push pour notifier l'utilisateur lors de la modification d'une tâche.

Séance 11 : Internationalisation et Localisation

- Présentation des concepts d'internationalisation (i18n) dans React Native.
- Mise en place d'un système de traduction avec des bibliothèques telles que **react-i18next**.
- Gestion des formats régionaux (dates, nombres, etc.).
- Adaptation de l'interface utilisateur pour une application multilingue.

Exercice : Ajouter la prise en charge de plusieurs langues dans l'application et vérifier l'affichage correct des traductions.

Séance 12 : Test, déploiement et bonnes pratiques

- Introduction aux tests avec Jest et React Native Testing Library.
- Tester un composant et simuler une interaction utilisateur.
- Préparation pour la publication : optimisation et bonnes pratiques.
- Génération d'un build avec Expo et publication sur les stores.

Exercice : Écrire un test pour vérifier l'ajout d'une tâche.

Prêt à coder ? Let's go !