

Test Technique: Création d'une application React Native avec l'API Pokémon

Objectif :

Créer une application mobile React Native pour afficher des informations sur des Pokémon en utilisant l'API ouverte de Pokémon.

Exigences :

1. React Native & TypeScript: Votre application doit être développée en React Native en utilisant TypeScript.
2. API Pokémon: Utilisez l'API ouverte de Pokémon pour récupérer des informations sur les Pokémon. Documentation et point d'accès: <https://pokeapi.co>
3. Fetch & Gestion de données: Observez comment vous pouvez faire des requêtes et gérer les données. Comment vous traitez les erreurs lors d'un échec de requête ?
4. Librairie (Optionnel) : Vous pouvez utiliser Redux et RTK query pour gérer l'état de votre application et faire des requêtes, mais ce n'est pas une obligation. Si vous choisissez une autre approche, justifiez votre choix.
5. Composants React Native :
 - Liste de Pokémon: Créez un écran qui affiche une liste de Pokémon. Chaque élément de la liste doit montrer au minimum le nom du Pokémon.
 - Détails du Pokémon: Lorsqu'on clique sur un Pokémon dans la liste, l'application doit montrer un autre écran (ou modal) avec plus de détails sur ce Pokémon (par exemple, son image, type, poids, taille).
6. Styled Components: Utilisez la bibliothèque styled-components pour styliser vos composants. Soyez créatif, le design est libre.
7. Bonnes pratiques: Assurez-vous de suivre les bonnes pratiques en termes de structure de code, de modularité et de propreté. Votre code doit être lisible et bien organisé.

Livraison

Fournissez le code source de votre application ainsi que les instructions pour l'installer et la lancer. Si possible, incluez une brève documentation expliquant votre approche, les choix techniques que vous avez faits et tout autre détail que vous jugez pertinent.

Bonus

- Gestion de la pagination ou du chargement infini lors de la navigation dans la liste de Pokémon.
- Mise en place de tests unitaires pour vos composants.

Évaluation

Nous évaluerons votre test sur la base de la qualité de votre code, de votre capacité à suivre les instructions, de votre compréhension des concepts React Native et de la manière dont vous abordez et résolvez les problèmes.