

Documentation Technique

Ajout d'une carte recensant les parcours à proximité dans l'application

1. Implémentation

Une nouvelle page a été créée pour accueillir la carte côté application mobile. Pour le voir apparaître dans l'application, il suffit de décommenter le bloc de code dans app/routes/Navigation.js associé :

```
/*Décommenter pour activer l'onglet*/  
/*  
<Tab.Screen  
  name="Carte"  
  component={MapPage}  
  options={{  
    headerShown: false,  
    tabBarIcon: () => (  
      <Icon name="map" color={theme.PRIMARY_COLOR} size={30}/>  
    ),  
  }}  
>  
*/
```

Ensuite vous pouvez aller dans le composant Map.component.js et faire vos modifications pour implémenter la map.

Solutions Techniques

1. MapView

<https://github.com/react-native-maps/react-native-maps>

MapView a le mérite d'être la solution la plus couramment utilisée par les applications en React Native, et est aussi celle qui propose le plus de fonctionnalités. Elle se base sur l'utilisation d'une carte liée à Google Maps, (ou à Carte sur iOS). De nombreux tutoriels sont également disponibles et renseignent très bien la façon d'implémenter différentes fonctionnalités.

Cependant MapView nécessite l'utilisation de l'API Google Maps Platform et de son SDK. Cela implique donc une facturation possible. Google Cloud Platform propose 300\$ de crédit gratuit au total et Google Maps Platform propose 200\$ de crédit gratuit par mois (en gros c'est une sorte de limite : au-delà d'un certain nombre de chargement de cartes, comptabilisé en « crédits », il faut payer). Voici une explication sur les tarifs donnée par Google :

Premier compte

Si le premier compte de facturation Cloud que vous créez est utilisé pour un projet dans lequel les API ou les SDK Google Maps Platform sont activés, l'essai gratuit de 300 \$ de Google Cloud Platform et le crédit mensuel récurrent de 200 \$ de Google Maps Platform s'appliquent tous les deux.

Le principe est le suivant : pendant l'essai gratuit, les frais sont d'abord déduits du crédit mensuel récurrent de 200 \$ de Google Maps Platform. Si les frais dépassent 200 \$ au cours d'un mois donné, le montant supplémentaire est déduit du fonds restant des 300 \$ d'essai gratuit de Google Cloud Platform.

Comme indiqué ci-dessus, à la date de fin de l'essai gratuit au plus tard, vous devez mettre à niveau votre premier compte de facturation Cloud vers un compte payant. Une fois la mise à niveau effectuée, le crédit mensuel de 200 \$ continuera à être appliqué à votre compte de facturation Cloud, même après la fin de l'essai gratuit.

En ce qui concerne les tarifs Google Maps Platform, les 200\$ de crédit par mois permettent 28 571 chargements de carte gratuits par mois. Au-delà, il faut payer pour les chargement en plus.

La solution de MapView semble donc offrir une bonne base gratuite, tout en devenant payante au-delà d'un certain cap.

2. MapBox

<https://github.com/rnmapbox/maps/blob/main/docs/GettingStarted.md>

De la même manière, MapBox permet de construire des cartes et d'y placer des points. L'implémentation a l'air plus simple que MapView, mais le degré de personnalisation de la carte a l'air moins important aussi.

En ce qui concerne la facturation, MapBox est aussi gratuit jusqu'à un certain seuil (pas d'information sur le seuil mais il doit être de l'ordre de celui de Google Maps).

3. Autres solutions

Diverses autres solutions existent aussi, qui semblent gratuites d'utilisation. Cependant, nous avons l'impression qu'elles ne fonctionnent pas toutes (erreurs à la compilation). Cela peut tout de même être des pistes à explorer :

<https://github.com/reggie3/react-native-webview-leaflet> (erreur à la compilation)

<https://www.npmjs.com/package/react-native-yandexmapkit> (non testé)

<https://docs.maptiler.com/react-native/> (erreur à la compilation)