

Partie 2 - Gestion de formulaires : application de gestion de budget

Formik, React Form Hook, yup

Introduction :

À travers cet exercice de développement de l'interface utilisateur (UI) d'une application de gestion de budget, vous allez apprendre à utiliser des bibliothèques permettant d'automatiser la gestion des formulaires dans une application React Native.

Dans les applications de finance, la sécurité est au cœur des préoccupations des utilisateurs potentiels, notamment sur les risques de fuites de leurs données personnelles. Il est primordial dans votre activité de développeur de prêter une attention particulière à cela.

L'application développée doit à terme permettre à un utilisateur de gérer les finances de son foyer. Elle doit être offline et elle doit fonctionner sans compte utilisateur.

Écran de l'application :

L'application doit être composée de 3 écrans principaux et de 2 écrans secondaires :

Écrans principaux :

La page « accueil » :

Porte d'entrée de l'application, elle permet à l'utilisateur de retrouver les informations sur le solde son compte, les dernières opérations (débit, crédit) et les liens de navigation pour accéder aux pages d'enregistrement de nouveaux revenus et de nouvelles dépenses.

La page « compte » :

Elle permet d'accéder à la liste des transactions enregistrées. Elle affiche également un récapitulatif des revenus, des dépenses et le solde.

Pour chaque ligne, il faudra afficher le libellé de l'opération, le type de l'opération, la date d'enregistrement et le montant en euros.

La page « statistiques » :

Dans cette page, l'utilisateur a accès à une analyse graphique des transactions de son compte.

- Analyse des revenus et des dépenses.
- Analyse des dépenses par catégories.

Écrans secondaires :

La page « Ajout revenus » :

Elle contient un formulaire permettant à l'utilisateur d'enregistrer un revenu sur son compte.

Le formulaire doit proposer les champs suivants :

- ✿ Un bénéficiaire (nom, prénom)
- ✿ Un montant
- ✿ Une date (date de déclaration)
- ✿ Une catégorie
- ✿ Un champ de commentaires

Les catégories pour les revenus sont les suivantes :

- ✿ Salaire et assimilé
- ✿ Revenu financier
- ✿ Rente
- ✿ Pension alimentaire
- ✿ Allocation chômage
- ✿ Prestations sociales
- ✿ Revenu foncier
- ✿ Revenu exceptionnel
- ✿ Autre revenu

La page « Ajout dépenses » :

Elle contient un formulaire permettant à l'utilisateur d'enregistrer une dépense sur son compte.

Le formulaire doit proposer les champs suivants:

- ✿ Un bénéficiaire (nom, prénom)
- ✿ Un montant
- ✿ Une date (date de déclaration)
- ✿ Une catégorie
- ✿ Un champ de commentaires

Les catégories pour les revenus sont les suivantes :

- ✿ Alimentaires
- ✿ Factures
- ✿ Transport
- ✿ Logement
- ✿ Santé
- ✿ Divertissement
- ✿ Vacances
- ✿ Shopping
- ✿ Autre.

Ce qu'il faut faire :

Vous devez installer et configurer un environnement de développement pour la création d'application avec React Native CLI. Ensuite, il faut créer une application React Native pour l'exercice.

Exercice 1 :

Créer les écrans principaux et mettre en place une navigation tab permettant à l'utilisateur d'accéder à celle-ci.

Exercice 2 :

Créer l'écran d'enregistrement de revenus. Mettre en place le formulaire de recueil des informations. Implémenter le traitement et la validation des données envoyées par l'utilisateur à l'aide d'une librairie dédiée.

Cette vue sera accessible depuis l'écran d'accueil grâce à une navigation stack.

Exercice 3 :

Créer l'écran d'enregistrement de dépenses. Faire le même traitement que celui réalisé pour les revenus.

Exercice 4 :

À l'aide du `json` fourni, finaliser les pages : accueil, compte et statistiques.

Ressources

Setup de l'environnement pour la création d'application React Native

<https://reactnative.dev/docs/environment-setup>

Traitement et validation de formulaire :

<https://react-hook-form.com/get-started#ReactNative>

<https://formik.org/docs/guides/react-native>



Schéma de validation

<https://react-hook-form.com/get-started#SchemaValidation>

<https://formik.org/docs/guides/validation>

Graphique

<https://formidable.com/open-source/victory/docs/native/>

<https://github.com/nihgwu/react-native-pie>

